

# Hope Artificial Intelligence



## Classification Assignment

### Problem Statement or Requirement:

A requirement from the Hospital, Management asked us to create a predictive model which will predict the Chronic Kidney Disease (CKD) based on the several parameters. The Client has provided the dataset of the same.

- 1.) Identify your problem statement
- 2.) Tell basic info about the dataset (Total number of rows, columns)
- 3.) Mention the pre-processing method if you're doing any (like converting string to number – nominal data)
- 4.) Develop a good model with good evaluation metric. You can use any machine learning algorithm; you can create many models. Finally, you have to come up with final model.
- 5.) All the research values of each algorithm should be documented. (You can make tabulation or screenshot of the results.)
- 6.) Mention your final model, justify why u have chosen the same.

# **Assignment – Classification Algorithm**

## **Problem Statement or Requirement:**

A client's requirement is, predictive model which will predict the Chronic Kidney Disease (CKD) based on the several parameters.

## **Solution :**

### **1.) Identify your problem statement**

- Stage 1: **Machine Learning** because input has majority of excel numerical data
- Stage 2: **Supervision** because we got clear input and output data's
- Stage 3: **Classification** because we have to predict the Chronic Kidney Disease.

### **2.) Tell basic info about the dataset**

We have totally 399 inputs (Rows) with different of 13 numerical and 12 categorical (totally 25 columns) input data.

### **3. Mention the pre-processing method if you're doing any (like converting string to number – nominal data)**

- Yes, we have to preprocess the input data that categorical input of sg,rbc,pc,pcc,ba,htn,dm,cad,appet,pe,ane column to be convert into meaningful number.
- Because Python cannot handle categorical input like health disease nominal data as input.
- So here, Classification column (Dependent variable) output, rest all are (Independent variable) inputs

### **4. Develop a good model with good evaluation metric. You can use any machine learning algorithm; you can create many models. Finally, you have to come up with final model.**

**5.) All the research values of each algorithm should be documented. (You can make tabulation or screenshot of the results.)**

1. Random Forest classification:

	precision	recall	f1-score	support
0	0.96	1.00	0.98	51
1	1.00	0.98	0.99	82
accuracy			0.98	133
macro avg	0.98	0.99	0.98	133
weighted avg	0.99	0.98	0.99	133

2. Decision Tree classification:

	precision	recall	f1-score	support
0	0.92	0.96	0.94	51
1	0.97	0.95	0.96	82
accuracy			0.95	133
macro avg	0.95	0.96	0.95	133
weighted avg	0.96	0.95	0.96	133

3. SVM Linear Classification:

	precision	recall	f1-score	support
0	1.00	0.96	0.98	51
1	0.98	1.00	0.99	82
accuracy			0.98	133
macro avg	0.99	0.98	0.98	133
weighted avg	0.99	0.98	0.98	133

#### 4. SVM Kernal:

	precision	recall	f1-score	support
0	0.68	0.92	0.78	51
1	0.94	0.73	0.82	82
accuracy			0.80	133
macro avg	0.81	0.83	0.80	133
weighted avg	0.84	0.80	0.81	133

---

#### 5. Logistic classification:

	precision	recall	f1-score	support
0	1.00	0.96	0.98	51
1	0.98	1.00	0.99	82
accuracy			0.98	133
macro avg	0.99	0.98	0.98	133
weighted avg	0.99	0.98	0.98	133

---

#### 6. KNN:

	precision	recall	f1-score	support
0	0.71	0.92	0.80	51
1	0.94	0.77	0.85	82
accuracy			0.83	133
macro avg	0.83	0.84	0.82	133
weighted avg	0.85	0.83	0.83	133

### 7.A. Navies Bayes: (MultinomialNB)

	precision	recall	f1-score	support
0	0.70	0.96	0.81	51
1	0.97	0.74	0.84	82
accuracy			0.83	133
macro avg	0.83	0.85	0.83	133
weighted avg	0.87	0.83	0.83	133

```
[[49  2]
 [21 61]]
```

### 7.B. Navies Bayes: (BernoulliNB)

	precision	recall	f1-score	support
0	0.86	1.00	0.93	51
1	1.00	0.90	0.95	82
accuracy			0.94	133
macro avg	0.93	0.95	0.94	133
weighted avg	0.95	0.94	0.94	133

```
[[51  0]
 [ 8 74]]
```

### 7.C. Navies Bayes: (BernoulliNB)

	precision	recall	f1-score	support
0	0.70	0.96	0.81	51
1	0.97	0.74	0.84	82
accuracy			0.83	133
macro avg	0.83	0.85	0.83	133
weighted avg	0.87	0.83	0.83	133

```
[[49  2]
 [21 61]]
```

**6.) Mention your final model, justify why u have chosen the same.**

- ✓ **Random Forest, SVM linear, and Logistic classification** are giving best result of 98%
- ✓ Navies bayes with Bernoulli algorithm gives good results 94% accuracy
- ✓ I will choose **Random forest algorithm** for this particular project, because when I tried with Traditional method as well as Grid search method, Random forest gives best result (Traditional RF 98% and Grid search RF gives 99%)
- ✓ Another big plus for using Random Forest is that it's really good at handling noisy data and avoiding overfitting. Instead of relying on just one decision tree, especially when the dataset is a bit messy or has lots of features.