

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

- The dataset is “**Chronic Kidney Disease**”. By using this data need to predict the Kidney disease based on the record.
- Step 1 – Dataset contains **Numeric data** with ordinal values. So, the domain is **Machine Learning**.
- Step 2 – **Learning** – Here input and output are clearly given so it comes under “**Supervised Learning**”.
- Step 3 – It is Supervised Learning by using the possibility of data in output column (i.e.) classification of disease yes/no so, it is a **classification problem**.

2.) Tell basic info about the dataset (Total number of rows, columns)

- Total number of rows, columns: **399 rows, 28 columns**.

```
In [3]: dataset.shape
```

```
Out[3]: (399, 25)
```

- For independent (i.e.) input columns: **399 rows, 27 columns**.
- For dependent (i.e.) input columns: **399 rows, 1 column**.

```
In [36]: print(independent.shape)
          print(dependent.shape)
```

```
(399, 27)
```

```
(399, 1)
```

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

- In dataset some columns contain **categorical value** so in pre-processing step did **one-hot encoding** using **get_dummies**.

```
In [32]: #changing categorical value to numerical value
dataset = pd.get_dummies(dataset, drop_first=True)
dataset=dataset.astype(int)
dataset
```

```
Out[32]:
```

	age	bp	al	su	bgr	bu	sc	sod	pot	hrmo	...	pc_1	pcc_1	ba_1	htn_1	dm_1	cad_1
0	2	76	3	0	148	57	3	137	4	12	...	0	0	0	0	0	0
1	3	76	2	0	148	22	0	137	4	10	...	1	0	0	0	0	0
2	4	76	1	0	99	23	0	138	4	12	...	1	0	0	0	0	0
3	5	76	1	0	148	16	0	138	3	8	...	1	0	0	0	0	0
4	5	50	0	0	148	25	0	137	4	11	...	1	0	0	0	0	0
...
394	51	70	0	0	219	36	1	139	3	12	...	1	0	0	0	0	0
395	51	70	0	2	220	68	2	137	4	8	...	1	0	0	1	1	1
396	51	70	3	0	110	115	6	134	2	9	...	1	0	0	1	1	1
397	51	90	0	0	207	80	6	142	5	8	...	1	0	0	1	1	1
398	51	80	0	0	100	49	1	140	5	16	...	1	0	0	0	0	0

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.

Developed a classification model: -

- Random Forest classifier
- Decision Tree Classifier
- KNN
- SVM
- Naïve Bayes classifier
- Logistic regression

Random Forest and **Logistic Regression** is giving Higher Accuracy of 99%.

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

S. No	Algorithm	Recall Yes-1	Recall No-0	Precision Yes-1	Precision No-0	F1 Score Yes-1	F1 Score No-0	ROC_AUC	Accuracy
1	Random Forest classifier	0.99	1.00	1.00	0.98	0.99	0.99	0.999	0.99

2	Decision Tree Classifier	0.93	0.98	0.99	0.89	0.96	0.93	0.953	0.95
3	SVM	0.98	1.00	1.00	0.96	0.99	0.98	0.987	0.98
4	Logistic regression	0.99	1.00	1.00	0.98	0.99	0.99	0.993	0.99
5	KNN	0.68	0.84	0.88	0.62	0.77	0.72	0.763	0.74
6	Gaussian NB	0.96	1.00	1.00	0.94	0.98	0.97	0.945	0.98
7	Complement NB	0.73	0.98	0.98	0.69	0.84	0.81	0.856	0.83
8	Multinomial NB	0.73	0.98	0.98	0.69	0.84	0.81	0.856	0.83

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

S. No	Algorithm	Recall Yes-1	Recall No-0	Precision Yes-1	Precision No-0	F1 Score Yes-1	F1 Score No-0	ROC_AUC	Accuracy
1	Random Forest classifier	0.99	1.00	1.00	0.98	0.99	0.99	0.999	0.99
4	Logistic regression	0.99	1.00	1.00	0.98	0.99	0.99	0.993	0.99

- Here Logistic regression and Random Forest Giving better accuracy than other model.
- While comparing random forest and Logistic regression above ROC_AUC is higher in RF than Logistic regression.