

# Assignment\_3

April 1, 2023

## 0.1 Assignment 3

### IMPLEMENTATION ON CLASSIFICATION Using R

Name	USN	Class
Ajay Nagaraj M	2SD19CS008	8TH Sem CSE B Div

```
[ ]: #Loading library
library(mlbench)
#Using BreastCancer dataset
data(BreastCancer, package = "mlbench")
breast_canc = BreastCancer[complete.cases(BreastCancer),]
#Displaying the information related to dataset with the head() function.
head(breast_canc)
```

A data.frame: 6 × 11

	Id	Cl.thickness	Cell.size	Cell.shape	Marg.adhesion	Epith.c.size	Bare.nu
	<chr>	<ord>	<ord>	<ord>	<ord>	<ord>	<fct>
1	1000025	5	1	1	1	2	1
2	1002945	5	4	4	5	7	10
3	1015425	3	1	1	1	2	2
4	1016277	6	8	8	1	3	4
5	1017023	4	1	1	3	2	1
6	1017122	8	10	10	8	7	10

```
[ ]: library(caret)
library(dplyr)
```

Loading required package: ggplot2

Loading required package: lattice

Attaching package: ‘dplyr’

The following objects are masked from ‘package:stats’:

filter, lag

The following objects are masked from ‘package:base’:

intersect, setdiff, setequal, union

```
[ ]: data1 <- select(breast_canc, -Id)
```

```
[ ]: #Dividing dataset into training and test dataset.
set.seed(100)
#Creating partitioning.
Training_Ratio <- createDataPartition(data1$Class, p=0.7, list = F)
#Creating training data.
Training_Data <- data1[Training_Ratio, ]
head(Training_Data)
#Creating test data.
Test_Data <- data1[-Training_Ratio, ]
head(Test_Data)
```

		Cl.thickness <ord>	Cell.size <ord>	Cell.shape <ord>	Marg.adhesion <ord>	Epith.c.size <ord>	Bare.nuclei <fct>	Bl.c <fct>
A data.frame: 6 × 10	1	5	1	1	1	2	1	3
	2	5	4	4	5	7	10	3
	3	3	1	1	1	2	2	3
	4	6	8	8	1	3	4	3
	6	8	10	10	8	7	10	9
	8	2	1	2	1	2	1	3
		Cl.thickness <ord>	Cell.size <ord>	Cell.shape <ord>	Marg.adhesion <ord>	Epith.c.size <ord>	Bare.nuclei <fct>	Bl.c <fct>
A data.frame: 6 × 10	5	4	1	1	3	2	1	3
	7	1	1	1	1	2	10	3
	9	2	1	1	1	2	1	1
	10	4	2	1	1	2	1	2
	11	1	1	1	1	1	1	3
	13	5	3	3	3	2	3	4

```
[ ]: model <- train(Class ~ ., method='knn', data = Training_Data, metric='Accuracy')
```

```
[ ]: dim(Test_Data)
```

1. 204 2. 10

```
[ ]: predict <- predict(model, Test_Data)
```

```
[ ]: predict <- as.factor(predict)
      truth <- as.factor(Test_Data$Class)
```

```
[ ]: precision <- posPredValue(truth, predict)
      precision
```

0.984962406015038

```
[ ]: cm <- confusionMatrix(predict, Test_Data$Class)
      cm
```

#### Confusion Matrix and Statistics

	Reference	
Prediction	benign	malignant
benign	131	13
malignant	2	58

Accuracy : 0.9265  
95% CI : (0.8816, 0.9583)  
No Information Rate : 0.652  
P-Value [Acc > NIR] : < 2.2e-16

Kappa : 0.8319

McNemar's Test P-Value : 0.009823

Sensitivity : 0.9850  
Specificity : 0.8169  
Pos Pred Value : 0.9097  
Neg Pred Value : 0.9667  
Prevalence : 0.6520  
Detection Rate : 0.6422  
Detection Prevalence : 0.7059  
Balanced Accuracy : 0.9009

'Positive' Class : benign

#### Testing Sample Data

```
[ ]: sample <- Test_Data[1,]
```

```
[ ]: predict(model, sample)
```

benign *Levels:* 1. 'benign' 2. 'malignant'

```
[ ]:
```