

Confusion Matrix Creation & Accuracy Calculation for Disease Diagnosis using KNN Model

```
#install.packages("class") #install.packages("gmodels")
```

Loading dataframe

```
prc <- read.csv("C:/Users/Arup/Documents/DS_DiseaseDiagnose.csv", stringsAsFactors = FALSE)
str(prc)

## 'data.frame':    100 obs. of  10 variables:
## $ id           : int  1 2 3 4 5 6 7 8 9 10 ...
## $ diagnosis_result : chr  "M" "B" "M" "M" ...
## $ radius        : int  23 9 21 14 9 25 16 15 19 25 ...
## $ texture        : int  12 13 27 16 19 25 26 18 24 11 ...
## $ perimeter      : int  151 133 130 78 135 83 120 90 88 84 ...
## $ area           : int  954 1326 1203 386 1297 477 1040 578 520 476 ...
## $ smoothness     : num  0.143 0.143 0.125 0.07 0.141 0.128 0.095 0.119 0.127 0.119 ...
## $ compactness    : num  0.278 0.079 0.16 0.284 0.133 0.17 0.109 0.165 0.193 0.24 ...
## $ symmetry       : num  0.242 0.181 0.207 0.26 0.181 0.209 0.179 0.22 0.235 0.203 ...
## $ fractal_dimension: num  0.079 0.057 0.06 0.097 0.059 0.076 0.057 0.075 0.074 0.082 ...
```

Removing the ID variable (not required) from the data set

```
prc <- prc[-1]
```

The data set contains patients who have been diagnosed with either Malignant (M) or Benign (B) cancer

```
prc$diagnosis <- factor(prc$diagnosis_result, levels = c("B", "M"), labels = c("Benign", "Malignant"))
round(prop.table(table(prc$diagnosis)) * 100, digits = 1) # Percentage form rounded to 1 decimal place

##
##   Benign Malignant
##      38         62
```

Normalizing numeric data

```
normalize <- function(x) {  
  return ((x - min(x)) / (max(x) - min(x))) }  
  
prc_n <- as.data.frame(lapply(prc[2:9], normalize)) # Normalize everything except the result  
  
summary(prc_n) # Checking if normalized  
  
##      radius      texture      perimeter      area  
## Min.   :0.0000 Min.   :0.0000 Min.   :0.0000 Min.   :0.0000  
## 1st Qu.:0.1875 1st Qu.:0.1875 1st Qu.:0.2542 1st Qu.:0.1639  
## Median :0.5000 Median :0.4062 Median :0.3500 Median :0.2637  
## Mean   :0.4906 Mean   :0.4519 Mean   :0.3732 Mean   :0.2989  
## 3rd Qu.:0.7500 3rd Qu.:0.7031 3rd Qu.:0.5188 3rd Qu.:0.4266  
## Max.   :1.0000 Max.   :1.0000 Max.   :1.0000 Max.   :1.0000  
## smoothness compactness symmetry fractal_dimension  
## Min.   :0.0000 Min.   :0.0000 Min.   :0.0000 Min.   :0.0000  
## 1st Qu.:0.3219 1st Qu.:0.1384 1st Qu.:0.2189 1st Qu.:0.1364  
## Median :0.4384 Median :0.2622 Median :0.3254 Median :0.2273  
## Mean   :0.4484 Mean   :0.2889 Mean   :0.3442 Mean   :0.2657  
## 3rd Qu.:0.5753 3rd Qu.:0.3876 3rd Qu.:0.4379 3rd Qu.:0.3636  
## Max.   :1.0000 Max.   :1.0000 Max.   :1.0000 Max.   :1.0000
```

Traininig and Testing the Dataset

```
prc_train <- prc_n[01:070,]  
prc_test  <- prc_n[71:100,]  
  
prc_train_labels <- prc[01:070, 1] # Target Variable is in Column-1 as Label  
prc_test_labels  <- prc[71:100, 1]
```

KNN Modelling and Confusion Matrix

```
library(class)  
  
## Warning: package 'class' was built under R version 4.1.3  
  
prc_test_pred <- knn(train = prc_train, test = prc_test, cl = prc_train_labels, k=10)  
  
library(gmodels)
```

```
## Warning: package 'gmodels' was built under R version 4.1.3
```

```
CrossTable(x=prc_test_labels, y=prc_test_pred, prop.chisq=FALSE)
```

```
##
```

```
##
```

```
## Cell Contents
```

```
## |-----|
## |                N |
## |      N / Row Total |
## |      N / Col Total |
## |      N / Table Total |
## |-----|
```

```
##
```

```
##
```

```
## Total Observations in Table:  30
```

```
##
```

```
##
```

prc_test_labels	prc_test_pred		Row Total
	B	M	
B	7	8	15
	0.467	0.533	0.500
	1.000	0.348	
	0.233	0.267	
M	0	15	15
	0.000	1.000	0.500
	0.000	0.652	
	0.000	0.500	
Column Total	7	23	30
	0.233	0.767	

```
##
```

```
##
```

```
#Measuring Accuracy
```

```
percentAccuracy <- 100*(7+15)/30; #TN+TP, whereas FN=0 & FP=8
```

```
percentAccuracy
```

```
## [1] 73.33333
```