

```
#Importing the data analysis modules and their dependancies
#The pandas library provides many functions and methods to expedite the data analysis
import pandas as pd
#The numpy library provides all the functions you need to prepare and transform your data
import numpy as np
# The matplotlib a 2-D plotting library that helps in visualizing figures
import matplotlib.pyplot as plt
#The seaborn library provides a high-level interface for drawing attractive and informative plots
import seaborn as sns
# The scikit learn library creates various models and prepare and evaluate data or models
from sklearn.linear_model import LogisticRegression,LinearRegression
from sklearn.model_selection import train_test_split
from sklearn import metrics
from sklearn.preprocessing import LabelEncoder
from sklearn.metrics import accuracy_score,mean_squared_error,mean_absolute_error,

#loading the data
test_data = pd.read_csv("test_100r.csv")
train_data = pd.read_csv('train_100r.csv')
```

```
#previewing the data
test_data
```



```
train_data
```

```
# previewing first 5 records  
train_data.head()
```

```
# previewing first 5 records  
test_data.head()
```

```
#sampling data
train_data.sample()
```

```
#sampling data
test_data.sample()
```

```
#visulizing the columns of the data set
train_data.columns
```

```
Index(['id', 'diagnosis', 'radius_mean', 'texture_mean', 'perimeter_mean',
       'area_mean', 'smoothness_mean', 'compactness_mean', 'concavity_mean',
       'concave points_mean', 'symmetry_mean', 'fractal_dimension_mean',
       'radius_se', 'texture_se', 'perimeter_se', 'area_se', 'smoothness_se',
       'compactness_se', 'concavity_se', 'concave points_se', 'symmetry_se',
       'fractal_dimension_se', 'radius_worst', 'texture_worst',
       'perimeter_worst', 'area_worst', 'smoothness_worst',
       'compactness_worst', 'concavity_worst', 'concave points_worst',
       'symmetry_worst', 'fractal_dimension_worst'],
      dtype='object')
```

```
#visulizing the columns of the data set
test_data.columns
```

```
Index(['id', 'radius_mean', 'texture_mean', 'perimeter_mean', 'area_mean',
       'smoothness_mean', 'compactness_mean', 'concavity_mean',
       'concave points_mean', 'symmetry_mean', 'fractal_dimension_mean',
       'radius_se', 'texture_se', 'perimeter_se', 'area_se', 'smoothness_se',
       'compactness_se', 'concavity_se', 'concave points_se', 'symmetry_se',
       'fractal_dimension_se', 'radius_worst', 'texture_worst',
       'perimeter_worst', 'area_worst', 'smoothness_worst',
       'compactness_worst', 'concavity_worst', 'concave points_worst',
       'symmetry_worst', 'fractal_dimension_worst'],
      dtype='object')
```

```
#previewing target variable
train_data['diagnosis'].value_counts().plot(kind='pie',radius=(2))
```

from the above it can be seen that the two strains of breast cancer are evenly distributed

```
#checking information about data
train_data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 25524 entries, 0 to 25523
Data columns (total 32 columns):
#   Column                                Non-Null Count  Dtype
---  -
0   id                                     25524 non-null  int64
1   diagnosis                             25524 non-null  int64
2   radius_mean                           25524 non-null  float64
3   texture_mean                           25524 non-null  float64
4   perimeter_mean                         25524 non-null  float64
5   area_mean                             25524 non-null  float64
6   smoothness_mean                       25524 non-null  float64
7   compactness_mean                      25524 non-null  float64
8   concavity_mean                        25524 non-null  float64
9   concave points_mean                   25524 non-null  float64
10  symmetry_mean                          25524 non-null  float64
11  fractal_dimension_mean                 25524 non-null  float64
12  radius_se                              25524 non-null  float64
13  texture_se                             25524 non-null  float64
14  perimeter_se                           25524 non-null  float64
15  area_se                                25524 non-null  float64
```

```

16 smoothness_se 25524 non-null float64
17 compactness_se 25524 non-null float64
18 concavity_se 25524 non-null float64
19 concave points_se 25524 non-null float64
20 symmetry_se 25524 non-null float64
21 fractal_dimension_se 25524 non-null float64
22 radius_worst 25524 non-null float64
23 texture_worst 25524 non-null float64
24 perimeter_worst 25524 non-null float64
25 area_worst 25524 non-null float64
26 smoothness_worst 25524 non-null float64
27 compactness_worst 25524 non-null float64
28 concavity_worst 25524 non-null float64
29 concave points_worst 25524 non-null float64
30 symmetry_worst 25524 non-null float64
31 fractal_dimension_worst 25524 non-null float64
dtypes: float64(30), int64(2)
memory usage: 6.2 MB

```

```

#checking information about data
test_data.info()

```

```

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 158 entries, 0 to 157
Data columns (total 31 columns):
#   Column                                Non-Null Count  Dtype
---  -
0   id                                     100 non-null    float64
1   radius_mean                           100 non-null    float64
2   texture_mean                           100 non-null    float64
3   perimeter_mean                         100 non-null    float64
4   area_mean                              100 non-null    float64
5   smoothness_mean                        100 non-null    float64
6   compactness_mean                       100 non-null    float64
7   concavity_mean                         100 non-null    float64
8   concave points_mean                    100 non-null    float64
9   symmetry_mean                          100 non-null    float64
10  fractal_dimension_mean                  100 non-null    float64
11  radius_se                               100 non-null    float64
12  texture_se                              100 non-null    float64
13  perimeter_se                            100 non-null    float64
14  area_se                                 100 non-null    float64
15  smoothness_se                          100 non-null    float64
16  compactness_se                          100 non-null    float64
17  concavity_se                            100 non-null    float64
18  concave points_se                       100 non-null    float64
19  symmetry_se                             100 non-null    float64
20  fractal_dimension_se                    100 non-null    float64
21  radius_worst                            100 non-null    float64
22  texture_worst                           100 non-null    float64
23  perimeter_worst                         100 non-null    float64
24  area_worst                              100 non-null    float64
25  smoothness_worst                       100 non-null    float64
26  compactness_worst                       100 non-null    float64
27  concavity_worst                         100 non-null    float64
28  concave points_worst                    100 non-null    float64
29  symmetry_worst                          100 non-null    float64
30  fractal_dimension_worst                 100 non-null    float64

```

```
dtypes: float64(31)
memory usage: 38.4 KB
```

```
#checking shape or number of record in the data
train_data.shape
```

```
(25524, 32)
```

```
#checking shape or number of record in the data
test_data.shape
```

```
(158, 31)
```

```
#checking datatypes of data
train_data.dtypes
```

id	int64
diagnosis	int64
radius_mean	float64
texture_mean	float64
perimeter_mean	float64
area_mean	float64
smoothness_mean	float64
compactness_mean	float64
concavity_mean	float64
concave points_mean	float64
symmetry_mean	float64
fractal_dimension_mean	float64
radius_se	float64
texture_se	float64
perimeter_se	float64
area_se	float64
smoothness_se	float64
compactness_se	float64
concavity_se	float64
concave points_se	float64
symmetry_se	float64
fractal_dimension_se	float64
radius_worst	float64
texture_worst	float64
perimeter_worst	float64
area_worst	float64
smoothness_worst	float64
compactness_worst	float64
concavity_worst	float64
concave points_worst	float64
symmetry_worst	float64
fractal_dimension_worst	float64
dtype:	object

```
#checking datatypes of data
test_data.dtypes
```

id	float64
radius_mean	float64
texture_mean	float64
perimeter_mean	float64

```

area_mean          float64
smoothness_mean    float64
compactness_mean    float64
concavity_mean      float64
concave points_mean float64
symmetry_mean       float64
fractal_dimension_mean float64
radius_se          float64
texture_se          float64
perimeter_se        float64
area_se            float64
smoothness_se       float64
compactness_se      float64
concavity_se        float64
concave points_se   float64
symmetry_se         float64
fractal_dimension_se float64
radius_worst        float64
texture_worst        float64
perimeter_worst     float64
area_worst          float64
smoothness_worst    float64
compactness_worst    float64
concavity_worst      float64
concave points_worst float64
symmetry_worst       float64
fractal_dimension_worst float64
dtype: object

```

▼ Data Preparation and Analysis

We will be using the training data set

```

#checking missing data
train_data.isna().sum()

```

```

id          0
diagnosis   0
radius_mean 0
texture_mean 0
perimeter_mean 0
area_mean    0
smoothness_mean 0
compactness_mean 0
concavity_mean 0
concave points_mean 0
symmetry_mean 0
fractal_dimension_mean 0
radius_se    0
texture_se    0
perimeter_se 0
area_se       0
smoothness_se 0
compactness_se 0
concavity_se 0

```

```

concave_points_se      0
symmetry_se            0
fractal_dimension_se   0
radius_worst           0
texture_worst          0
perimeter_worst        0
area_worst             0
smoothness_worst       0
compactness_worst      0
concavity_worst        0
concave_points_worst   0
symmetry_worst         0
fractal_dimension_worst 0
dtype: int64

```

```

#checking missing data
test_data.isna().sum()

```

```

id                58
radius_mean       58
texture_mean      58
perimeter_mean    58
area_mean         58
smoothness_mean   58
compactness_mean  58
concavity_mean    58
concave_points_mean 58
symmetry_mean     58
fractal_dimension_mean 58
radius_se         58
texture_se        58
perimeter_se      58
area_se           58
smoothness_se     58
compactness_se    58
concavity_se      58
concave_points_se 58
symmetry_se       58
fractal_dimension_se 58
radius_worst      58
texture_worst     58
perimeter_worst   58
area_worst        58
smoothness_worst  58
compactness_worst 58
concavity_worst   58
concave_points_worst 58
symmetry_worst    58
fractal_dimension_worst 58
dtype: int64

```

► The test data has 58 missing values

[] ↪ 3 cells hidden

▶ MODELING

[] ↪ 2 cells hidden

▶ Logistic Regression or Binary Classification

[] ↪ 6 cells hidden

The above data set is the comparison between the test values and the predcited values

▶ Deep neural network

[] ↪ 1 cell hidden

▶ Building and compiling

[] ↪ 2 cells hidden

▶ Fitting,predicting and evaluating

[] ↪ 2 cells hidden

▶ Deep neural networking model 2

[] ↪ 2 cells hidden

▶ Fitting,predicting and evaluating

[] ↪ 2 cells hidden

▶ Performance and Evaluation

↪ 1 cell hidden

