**General information on dataset:**

**1:Name of dataset: Breast Cancer**

**Wisconsin (Diagnostic) Data Set**

(Predict whether the cancer is benign or malignant)

**About Dataset**

Features are computed from a digitized image of a fine needle aspirate (FNA) of a breast mass. They describe characteristics of the cell nuclei present in the image.  
n the 3-dimensional space is that described in: [K. P. Bennett and O. L. Mangasarian: "Robust Linear Programming Discrimination of Two Linearly Inseparable Sets", Optimization Methods and Software 1, 1992, 23-34].

This database is also available through the UW CS ftp server:  
ftp ftp.cs.wisc.edu  
cd math-prog/cpo-dataset/machine-learn/WDBC/

Also can be found on UCI Machine Learning Repository: <https://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+%28Diagnostic%29>

Attribute Information:

1) ID number  
2) Diagnosis (M = malignant, B = benign)  
3-32)

Ten real-valued features are computed for each cell nucleus:

a) radius (mean of distances from center to points on the perimeter)  
b) texture (standard deviation of gray-scale values)  
c) perimeter  
d) area  
e) smoothness (local variation in radius lengths)  
f) compactness (perimeter^2 / area - 1.0)  
g) concavity (severity of concave portions of the contour)  
h) concave points (number of concave portions of the contour)  
i) symmetry  
j) fractal dimension ("coastline approximation" - 1)

The mean, standard error and "worst" or largest (mean of the three  
largest values) of these features were computed for each image,  
resulting in 30 features. For instance, field 3 is Mean Radius, field  
13 is Radius SE, field 23 is Worst Radius.

All feature values are recoded with four significant digits.

Missing attribute values: none

Class distribution: 357 benign, 212 malignant

2: The number of columns: 33, 30 feature, 1 classification

3: number of samples :570 sample

4: the number of samples used in training and testing .

The number of samples used in training, validation, and testing depends on the split ratio specified in the **train\_test\_split** function. In the provided code, the test size is set to 0.30, which means 30% of the data will be used for testing, and the remaining 70% will be used for training.

Here's the breakdown of the samples:

* Training set: 70% of the total data
* Testing set: 30% of the total data

Given that the total number of samples (full dataset size) is 570, and the test size is set to 0.30 (30%), we can calculate the number of samples in each set:

1. Training set: 570×0.70=399570×0.70=399 samples
2. Testing set: 570×0.30=171570×0.30=171 samples

So, in this scenario:

* The training set contains 399 samples.
* The testing set contains 171 samples.

Implementation details:

Number of feature :30

Names of feature :

1. radius\_mean
2. texture\_mean
3. perimeter\_mean
4. area\_mean
5. smoothness\_mean
6. compactness\_mean
7. concavity\_mean
8. concave points\_mean
9. symmetry\_mean
10. fractal\_dimension\_mean
11. radius\_se
12. texture\_se
13. perimeter\_se
14. area\_se
15. smoothness\_se
16. compactness\_se
17. concavity\_se
18. concave points\_se
19. symmetry\_se
20. fractal\_dimension\_se
21. radius\_worst
22. texture\_worst
23. perimeter\_worst
24. area\_worst
25. smoothness\_worst
26. compactness\_worst
27. concavity\_worst
28. concave points\_worst
29. symmetry\_worst
30. fractal\_dimension\_worst

* **Batch Size**: Batch size is set to 1, as specified in **model.fit(X\_train, y\_train, batch\_size=1, epochs=50)**.
* **Number of Epochs**: 50 epochs are used for training the model.

Additionally:

* **Loss Function**: Binary cross-entropy loss is used, specified as **loss="binary\_crossentropy"**.
* **Metrics**: Model performance is evaluated using accuracy, specified as **metrics=["accuracy"]**.

Top of Form

A graph with blue dots

Description automatically generated

A graph with blue lines and white text

Description automatically generated