

#Breast Cancer Classification — ML Pipeline Project

##Project Overview

This project builds a Machine Learning model to classify whether a breast tumor is Malignant or Benign using the Breast Cancer Wisconsin Dataset.

We implement a complete ML pipeline including preprocessing, scaling, model training, and evaluation.

##Dataset Information

- Source: `sklearn.datasets.load_breast_cancer()`
- Total Samples: 569
- Features: 30 Numerical Features
- Target Classes:
 - 0 → Malignant
 - 1 → Benign

##Technologies Used

- Python
- Pandas
- NumPy
- Scikit-learn

##Project Workflow

1. Load dataset
2. Convert to DataFrame
3. Split Features & Target
4. Train-Test Split
5. Apply Feature Scaling
6. Build ML Pipeline
7. Train Logistic Regression Model
8. Make Predictions
9. Evaluate Performance

##Machine Learning Pipeline

Pipeline Steps:

- **Preprocessing:** StandardScaler
- **Model:** Logistic Regression

Pipeline automates:

- Data scaling
- Model training
- Prediction process

##Model Evaluation Metrics

- Accuracy Score
- Precision
- Recall
- F1-Score

Example Result:

Accuracy \approx 97%

##Code Structure

Import Libraries

Load Dataset

Create DataFrame

Split X and y

Train-Test Split

Scaling using StandardScaler

Pipeline Creation

Model Training

Prediction

Evaluation

##How to Run

1. Install required libraries:

```
pip install pandas numpy scikit-learn
```

2. Run the Python script / Jupyter Notebook.

##Key Learning Outcomes

- Understanding ML Pipeline
- Feature Scaling Importance
- Logistic Regression Implementation
- Model Evaluation Techniques

##Conclusion

The pipeline model successfully classifies breast cancer tumors with high accuracy. Using pipelines improves code organization, reduces errors, and ensures consistent preprocessing during training and testing.