Name: Adithisri Vishwanath
Register number: 2022116013
Department: Biomedical Engineering

Project Report: Breast Cancer Classification using MLP and Streamlit

Predicting Malignant vs. Benign Breast Cancer Cases Using MLP Classifier

---

🔹 Objective

To build a machine learning model that predicts whether a breast cancer tumor is malignant or benign using diagnostic image-derived features and deploy it via a web app for interactive predictions.

🔹 Steps and What I Did

1. Data Loading & Understanding

- Loaded the Breast Cancer Wisconsin Diagnostic dataset from sklearn.datasets.
- Examined the shape, feature names, target classes, and distribution.
- Understood that the dataset contains 569 samples and 30 numerical features related to cell nuclei.

2. Data Preprocessing

- Split the dataset into training (80%) and testing (20%) sets using train_test_split.
- Applied StandardScaler to normalize the input features — critical for MLP model convergence.

3. Model Building

- Initialized and trained an MLPClassifier with two hidden layers (64 and 32 neurons).
- Chose ReLU activation and Adam optimizer.
- Set training iterations to 500 to allow convergence.

4. Model Evaluation

- Evaluated performance using metrics like precision, recall, accuracy, and F1-score.
- Created a confusion matrix to understand prediction distribution.
- Achieved a test accuracy of ~97%, with strong precision/recall values.

5. Saving the Model

- Used joblib to save both the trained model (mlp_2layer_model.pkl) and the scaler (scaler.pkl) for later use.

6. Building the Web App

- Developed a **Streamlit app** with 30 input fields, one for each feature.
- Loaded the model and scaler to make real-time predictions based on user input.
- The app provides clear feedback: **Benign or Malignant** prediction.
- https://qnq9i8mfkcpgin95dkvfxh.streamlit.app/" url for the app

---

🔹 **Key Learnings from the Project**

- **MLP Classifier Effectiveness**: MLP can yield high accuracy for classification tasks if preprocessing and architecture are well-tuned.
- **Importance of Scaling**: Feature scaling with StandardScaler is essential for neural network models to avoid convergence issues.
- **Model Deployment**: Streamlit offers a powerful and simple way to turn Python models into interactive apps.
- **Evaluation Matters**: Relying only on accuracy is misleading — metrics like precision, recall, and F1-score provide deeper insights, especially in medical applications where false negatives can be dangerous.

🔹 **Key Takeaways from the Course**

- A clear **ML pipeline**—from data cleaning to model deployment—is crucial for any machine learning solution.
- **Scikit-learn's versatility** with preprocessing, modeling, and evaluation tools simplifies model development.
- **Model interpretability and evaluation** should be prioritized alongside accuracy.
- Hands-on projects help build **real-world problem-solving skills** beyond theory.
- Confidence gained in using libraries like Streamlit, joblib, and understanding the logic behind model selection, training, and testing.

🔹 **Conclusion**

This project not only improved my technical skills in machine learning but also strengthened my ability to deliver end-to-end solutions — from understanding raw data to deploying an accessible app for users. It also emphasized the importance of ethical AI in healthcare by minimizing prediction errors, especially false negatives.