ASSIGNMENT 6

ROHAN GHOSH DASTIDAR
22CH30028
rgdastidar2069@kgpian.iitkgp.ac.in

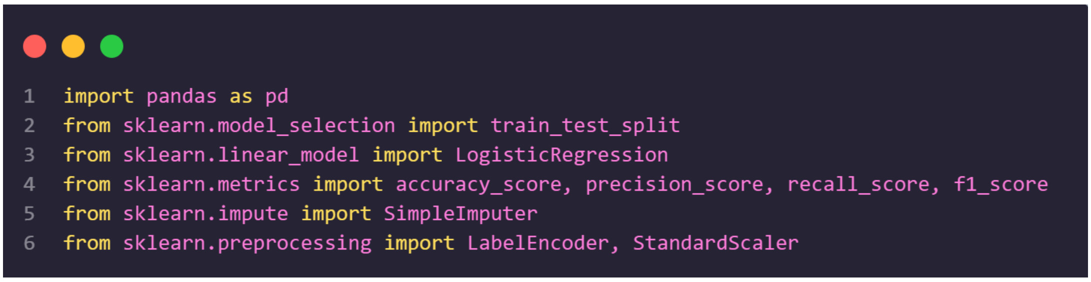
OBJECTIVE:

This assignment focuses on understanding and comparing the performance of different machine learning models for predicting cancer types (Benign/Malignant) based on a given dataset.

CODE OVERVIEW:

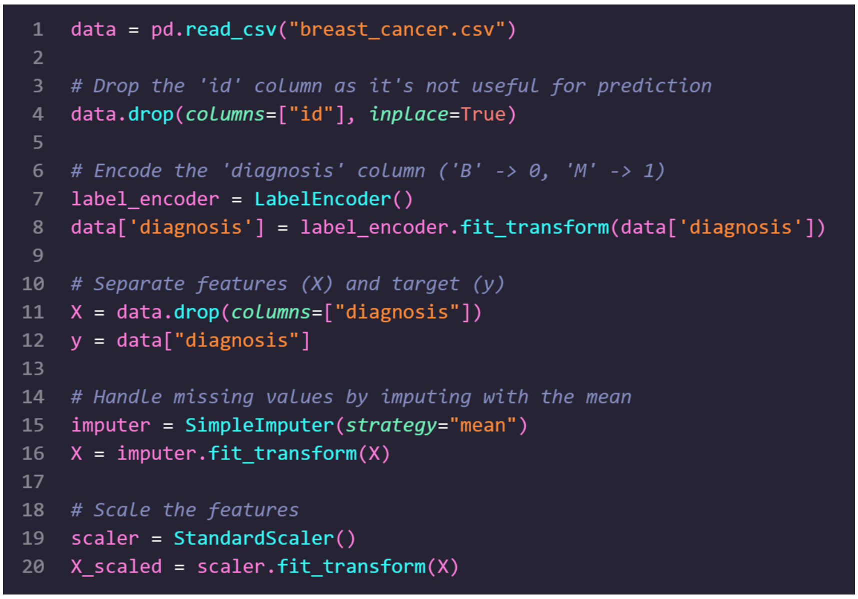
Identify the main sections for data import, manipulation, model training, and result storage.

Import dependencies:



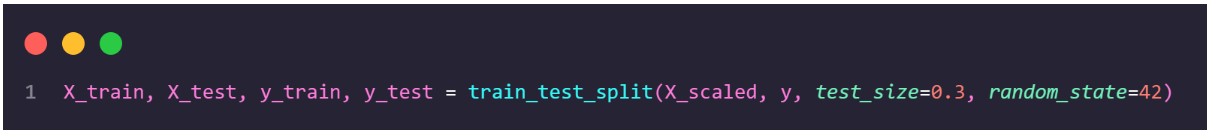
```python
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import accuracy_score, precision_score, recall_score, f1_score
from sklearn.impute import SimpleImputer
from sklearn.preprocessing import LabelEncoder, StandardScaler
```

Data import and data pre-processing:



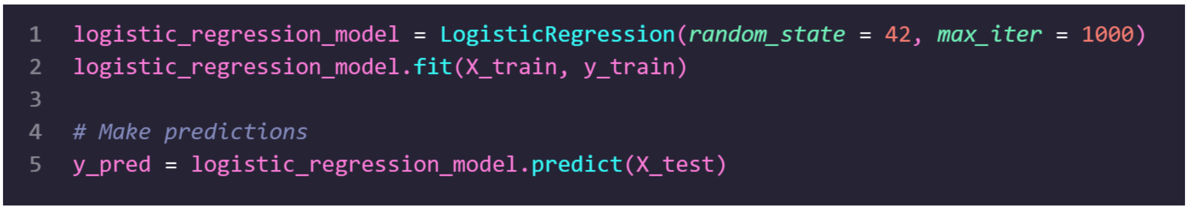
```python
data = pd.read_csv("breast_cancer.csv")

# Drop the 'id' column as it's not useful for prediction
data.drop(columns=["id"], inplace=True)

# Encode the 'diagnosis' column ('B' -> 0, 'M' -> 1)
label_encoder = LabelEncoder()
data['diagnosis'] = label_encoder.fit_transform(data['diagnosis'])

# Separate features (X) and target (y)
X = data.drop(columns=["diagnosis"])
y = data["diagnosis"]

# Handle missing values by imputing with the mean
imputer = SimpleImputer(strategy="mean")
X = imputer.fit_transform(X)

# Scale the features
scaler = StandardScaler()
X_scaled = scaler.fit_transform(X)
```

Splitting the dataset into TRAINING and TESTING:



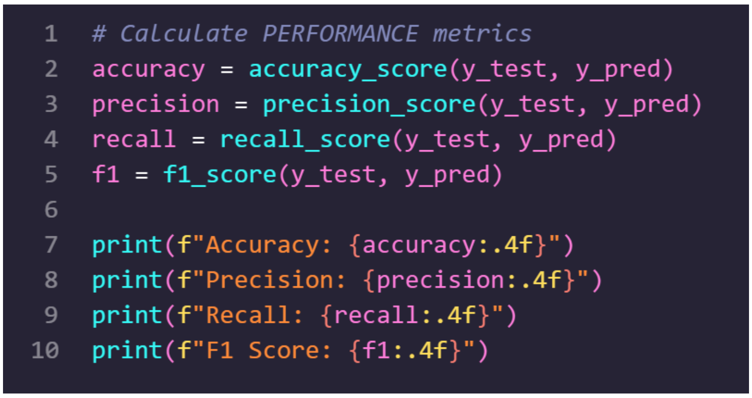
```python
X_train, X_test, y_train, y_test = train_test_split(X_scaled, y, test_size=0.3, random_state=42)
```

Training the model and making prediction on the TESTING subset:



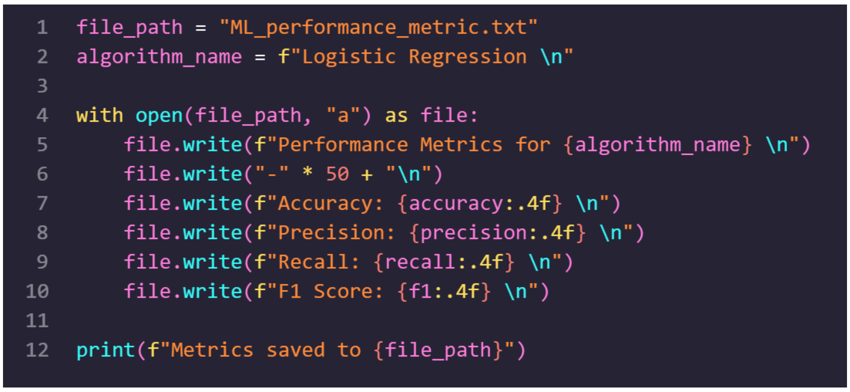
```python
logistic_regression_model = LogisticRegression(random_state = 42, max_iter = 1000)
logistic_regression_model.fit(X_train, y_train)

# Make predictions
y_pred = logistic_regression_model.predict(X_test)
```

Calculating the PERFORMANCE metric and printing it:



```python
# Calculate PERFORMANCE metrics
accuracy = accuracy_score(y_test, y_pred)
precision = precision_score(y_test, y_pred)
recall = recall_score(y_test, y_pred)
f1 = f1_score(y_test, y_pred)

print(f"Accuracy: {accuracy:.4f}")
print(f"Precision: {precision:.4f}")
print(f"Recall: {recall:.4f}")
print(f"F1 Score: {f1:.4f}")
```

Storing the performance metric values in .txt file:



```python
file_path = "ML_performance_metric.txt"
algorithm_name = f"Logistic Regression \n"

with open(file_path, "a") as file:
    file.write(f"Performance Metrics for {algorithm_name} \n")
    file.write("-" * 50 + "\n")
    file.write(f"Accuracy: {accuracy:.4f} \n")
    file.write(f"Precision: {precision:.4f} \n")
    file.write(f"Recall: {recall:.4f} \n")
    file.write(f"F1 Score: {f1:.4f} \n")

print(f"Metrics saved to {file_path}")
```

DATA DESCRIPTION:

Dataset – *breast_cancer.csv*

COLUMNS:

Features: ***All contain numerical values***

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.

Target: ***Categorical column – contains "B" for Benign & "M" for Malignant – depicting cancer type***

diagnosis

Redundant column(s):

id – *has no significance in prediction of the target*

ANALYSIS OF THE SVM MODEL:

SVM's role in cancer type prediction.

- SVM's are useful for binary classification as well as multi class classification (for predicting multiple cancer types) with high accuracy and precision.
- SVM can handle the complexity of non-linear input data by using different *kernels*.
- RBF kernel is the best for handling highly non linear data as it maps the input data into a higher dimensional space allowing the hyperplane decision boundary to categorize the data effectively.

Why Support Vector Machines should be used:

ADVANTAGES:

- Accurate
- Works well on smaller datasets
- Efficient to train on a subset of the whole data

DISADVANTAGES:

- NOT suited for large datasets as training time is too high
- Less effective on noisy data

ANALYSIS OF THE NEURAL NETWORK REGRESSION MODEL:

Significance of neural networks in cancer type prediction:

- Neural networks excel at processing high-dimensional data (with lots of features) and can identify intricate patterns within it thereby producing highly reliable output.
- Neural networks, particularly Deep Neural Networks have the ability to automatically learn relevant features from the raw input data, reducing the need for manually performing *feature engineering*.

Explain the Grid Search process for neural network parameters.

- Grid search is a *hyperparameter tuning technique* used to find the best combination of hyperparameters for a machine learning model.
- Hyperparameters control the learning process but are not directly obtained from the input data
- Hyperparameters involved in the training of Neural Networks:
  - Learning rate: Controls how quickly the model updates its weights during training.
  - Number of Hidden layers: Determines the depth of the neural network.
  - Number of neurons per layer: Determines the width of each hidden layer.
  - Activation Function: The function used to convert the input data into a non-linear version *ReLu, sigmoid, tan(h)*
  - Epochs: Number of times the entire dataset is propagated through the neural network.
  - Solver (Optimizer): The algorithm used to minimize the loss function (*adam*, *sgd*, *rmsprop*).

CONTRAST THE PERFORMANCE OF THE THREE MODELS:

List of Performance parameters for each ML model ordered in decreasing performance

MODEL	**Accuracy**	**Precision**	**Recall**	**F1 Score**
Logistic Regression	0.9825	0.9688	0.9841	0.9764
Random Forest Classifier	0.9708	0.9672	0.9365	0.9516
SVM – Kernel: *Linear*	0.9649	0.9672	0.9365	0.9516
Neural Network Regression	0.9649	0.9672	0.9365	0.9516
KNN	0.9591	0.9516	0.9365	0.9440
SVM – Kernel: *Poly*	0.9415	0.9818	0.8571	0.9153
SVM – Kernel: *RBF*	0.9357	1.0000	0.8254	0.9043
SVM – Kernel: *Sigmoid*	0.4678	0.1818	0.1270	0.1495

Most suitable model in terms of accuracy –
Logistic Regression . . . and **Random Forest** (which are of similar accuracy)

DISCUSSION:

Broader implications of accurate cancer type prediction:

- Accurate diagnosis helps in improving patient outcomes and prevent misdiagnosis.
- Accurate classification of cancer also helps in the development of drugs aimed at targeting specific cancer types
- By correctly identifying aggressive cancers, healthcare providers can prioritize the correct treatment plan, thereby reducing the overall costs associated with treatment
  .

Real-world applications of the machine learning models' performance:

- Reliability of machine learning models depend on their ability to make reliable predictions and adapt to various conditions depending on the nature and source of the input dataset used.
- Healthcare applications – *Logistic Regression*, *Random Forest* algorithms are used to classify cancer subtypes, identify patterns in X-rays, MRI’s, CT scans to detect anomalies.
- Business and Industry – *KNN models* are useful in e-commerce for interpreting customer behaviour and enabling personalized product recommendations; *Random Forests* are used by banks to detect fraudulent transactions by analyzing patterns in financial data with high accuracy and F1 Scores.
- Environmental Monitoring – *Neural Networks* are used to analyze patterns in climate data (temperature, humidity, CO2) to predict weather events.

CONCLUSION:

This project involved the evaluation and performance analysis of multiple machine learning models for classification tasks, particularly in the context of prediction of breast cancer. The results demonstrated that models such as **Logistic Regression**, **Random Forest**, and **Neural Networks** excel in accuracy, precision, recall, and F1 scores, making them highly suitable and reliable for real-world applications. While models like SVM with certain kernels (such as *linear*) and K-Nearest Neighbours showed competitive performance, other kernels such as *sigmoid* were less effective, highlighting the importance of model selection and hyperparameter tuning.

The results of the project also highlight the potential of machine learning in solving complex classification problems, especially in important domains such as healthcare. These advanced techniques can achieve improved decision-making, enhance efficiency, and positively impact people’s lives.

In conclusion, selecting the right model depends not only on its performance metrics but also on the specific requirements of the task, and the constraints associated with it. Each model has its benefits and trade offs and it is the responsibility of the ML engineer to choose the correct model and its parameters to suit the task.