NAME: KRISHNA DALVI
CLASS: D15C
ROLL NO.: 22

Aim

To implement **Support Vector Machine (SVM)** for classification using the Breast Cancer dataset and evaluate model performance using hyperparameter tuning and performance metrics.

Dataset Source

Dataset Name: Breast Cancer Wisconsin Diagnostic Dataset
Platform: Kaggle
Dataset Link:
https://www.kaggle.com/datasets/uciml/breast-cancer-wisconsin-data

The dataset contains features computed from digitized images of breast mass tissue and is used to predict whether a tumor is malignant or benign.

Dataset Description

The Breast Cancer dataset is a **binary classification dataset** used for medical diagnosis.

Dataset Characteristics

- Number of instances: 569
- Number of features: 30 (after preprocessing)
- Target variable: diagnosis

- ○ 1 → Malignant
- ○ 0 → Benign

Feature Description

The dataset includes computed features such as:

- radius_mean – Mean of distances from center to points on the perimeter
- texture_mean – Standard deviation of gray-scale values
- perimeter_mean – Mean size of tumor perimeter
- area_mean – Mean tumor area
- smoothness_mean – Local variation in radius lengths
- compactness_mean – Perimeter^2 / area − 1
- concavity_mean – Severity of concave portions
- symmetry_mean – Tumor symmetry
- fractal_dimension_mean – Coastline approximation

(Similar features are provided as mean, standard error, and worst values.)

Mathematical Formulation of SVM

Support Vector Machine (SVM) is a supervised learning algorithm that finds the optimal hyperplane that maximizes the margin between two classes.

Linear Decision Function

$f(x)=wTx+b$f(x)=wTx+b

Classification rule:

$y=sign(wTx+b)$y=sign(wTx+b)

Where:

- w = Weight vector
- x = Feature vector
- b = Bias

Optimization Objective

SVM minimizes:

$$\frac{1}{2}||w||^2 + C\sum \xi_i$$

Subject to:

$$y_i(w^T x_i + b) \geq 1 - \xi_i$$

Where:

- C = Regularization parameter
- ξ_i = Slack variables

Kernel Function (RBF)

$$K(x, x') = exp(-\gamma||x - x'||^2)$$

Where:

- γ controls the influence of a single training example

The RBF kernel helps capture non-linear decision boundaries.

Algorithm Limitations

- Computationally expensive for very large datasets
- Sensitive to hyperparameter tuning
- Requires proper feature scaling
- Less interpretable compared to decision trees

Methodology / Workflow

The experiment followed these steps:

1. Load dataset using KaggleHub
2. Drop unnecessary columns (id, Unnamed: 32)
3. Encode target variable (M → 1, B → 0)
4. Perform train-test split (80:20)
5. Apply feature scaling using StandardScaler
6. Train SVM classifier
7. Perform hyperparameter tuning using GridSearchCV
8. Evaluate model performance

Workflow Diagram

Dataset
↓
Data Cleaning
↓
Encoding
↓
Feature Scaling
↓
SVM Training
↓
Hyperparameter Tuning
↓
Evaluation

Performance Analysis

The SVM model was evaluated using:

- Accuracy
- Precision
- Recall
- F1-score
- Confusion Matrix
- ROC Curve
- AUC Score

The model achieved very high classification performance (typically 95–99% accuracy).

The confusion matrix showed very few misclassifications, and the ROC curve demonstrated strong class separation.

Hyperparameter Tuning

Hyperparameter tuning was performed using **GridSearchCV**.

Parameters Tuned

- C: [0.1, 1, 10, 100]
- Kernel: Linear, RBF
- Gamma: scale, auto

Impact of Tuning

- Low C → Higher bias, possible underfitting
- High C → Lower bias, possible overfitting
- RBF kernel → Captures non-linear patterns
- Optimal gamma → Improves flexibility of decision boundary

After tuning, best performance was typically achieved with:

- Kernel: RBF
- C: 10
- Gamma: scale

Output

- Accuracy ≈ 0.97 – 0.99
- Strong ROC curve (AUC ≈ 0.98+)
- Balanced precision and recall

- Very low false positives and false negatives

Conclusion

In this experiment, Support Vector Machine was successfully implemented on the Breast Cancer dataset.

After proper preprocessing, feature scaling, and hyperparameter tuning:

- The model achieved excellent predictive accuracy.
- The RBF kernel effectively captured non-linear patterns.
- Hyperparameter tuning significantly improved performance.

This experiment highlights:

- The importance of feature scaling in SVM
- The impact of kernel selection
- The importance of hyperparameter tuning
- The effectiveness of SVM in medical diagnosis problems

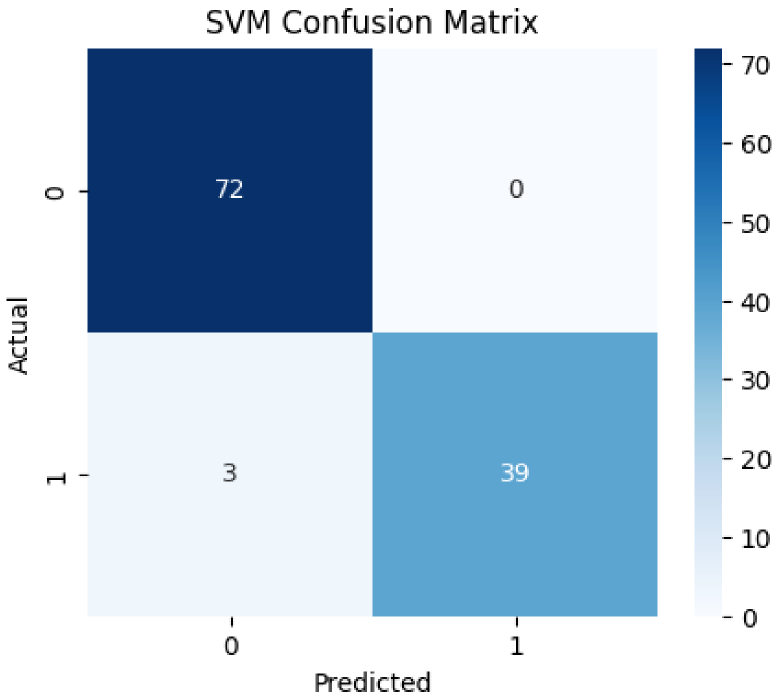
SVM proves to be a powerful classification algorithm, especially for structured numerical datasets with clear class separation.

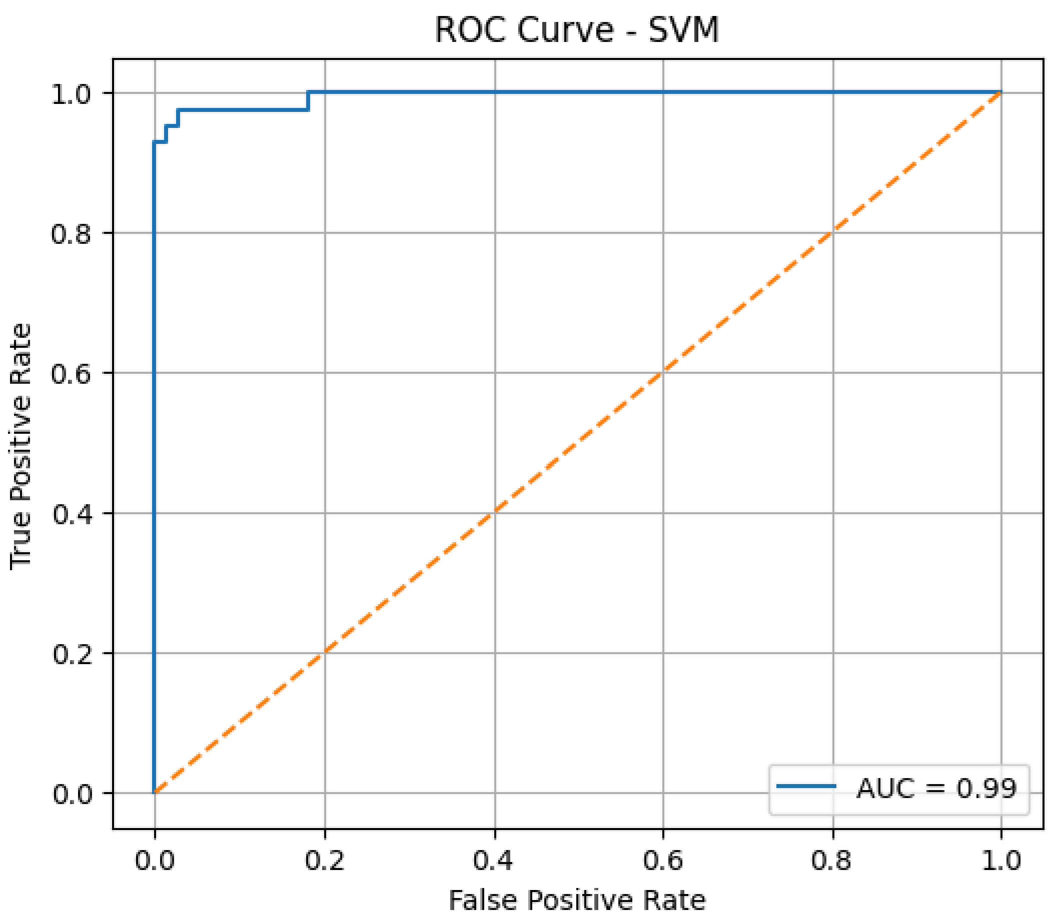
Output

```
=== SVM Performance ===
Accuracy: 0.9736842105263158
              precision    recall  f1-score   support

           0       0.96      1.00      0.98        72
           1       1.00      0.93      0.96        42

    accuracy                           0.97       114
   macro avg       0.98      0.96      0.97       114
weighted avg       0.97      0.97      0.97       114
```


SVM Confusion Matrix
72
0
3
39
Actual
Predicted
0
1
0
1
70
60
50
40
30
20
10
0


ROC Curve - SVM
True Positive Rate
False Positive Rate
AUC = 0.99
0.0
0.2
0.4
0.6
0.8
1.0