|  |
| --- |
| Project Based Learning Report on    Implementation of SVM of artificial neural network in python for wine dataset    Submitted in the partial fulfillment of the requirements For the Project based learning in Fuzzy Logic, Neural Networks and Generic Algorithms course.    In branch    ELECTRONICS AND COMMUNICATION ENGINEERING    By    **Name of the student PRN**  Dhruv Gupta 2114110456  Vaibhav Gupta 2214110453  Pranjal kumar Jha 2114110462      Under the guidance of Course In charge.  Prof. V. P. KADUSKAR    Department of Electronics & Communication Engineering  Bharati Vidyapeeth  (Deemed to be University)  College of Engineering,  Pune – 411043 |

**Bharati Vidyapeeth**

**(Deemed to be University)**

**College of Engineering,**

**Pune – 411043**

**DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING**

# CERTIFICATE

Certified that the Project Based Learning report entitled, “Implementation of SVM of artificial neural network in python for wine dataset.” is a bonafied work done by

**Name of the student PRN**

**Dhruv Gupta 2114110456**

**Vaibhav Gupta 2214110453**

**Pranjal Kumar Jha 2114110462**

in partial fulfillment of the requirements for the award of credits for Project Based Learning (PBL) in Fuzzy Logic, Neural Network and Generic Algorithms Course of Bachelor of Technology Semester V, in Electronics and Communication Branch.

Date:

Prof. V. P. Kaduskar Prof. Dr. A. A. Shinde

Course Incharge Head of Department

**INDEX**

|  |  |  |
| --- | --- | --- |
| SR NO | TITLE | PAGE NO |
| 1 | PROBLEM STATEMENT | 1 |
| 2 | SUPPORT VECTOR MACHINE | 2 |
| 3 | PROGRAM & OUTPUT | 3 - 4 |
| 4 | ALGORITHM | 5 |
| 5 | CONCLUSION | 6 |

# PROBLEM STATEMENT

**Q**) Implementing a Support Vector Machine (SVM) and an Artificial Neural Network (ANN) in Python to classify the Wine dataset. Our goal is to compare the performance machine learning models on the Wine dataset and determine which one provides better accuracy in classifying the wine flowers into their respective species.

**Solution.**

**Dataset description:**

The Wine dataset is a well-known dataset commonly used for classification tasks in machine learning. It is often used for practicing classification algorithms and evaluating their performance. The Wine dataset contains information about different varieties of wine, specifically, three different classes of wine. Here's a description of the Wine dataset:

**Implementation steps:**

* Import the necessary libraries of python which is going to help while performing the SVM like sklearn and different libraries from sklearn.
* Load the wine dataset using datasets.load\_datasetname()
* Dataset Processing:

1. Split the dataset into training and testing sets to evaluate the models.
2. Normalize or standardize the feature values if necessary.

* Support Vector Machine (SVM) implementation:

1. Implement SVM classifier using library SVC.
2. Train the SVM model on training data.
3. Predict the output on testing data.
4. Calculate the accuracy by accuracy\_score method.

# 2. SUPPORT VECTOE MACHINE

Support Vector Machine (SVM) is a powerful and widely used supervised machine learning algorithm for classification and regression tasks. It is particularly effective in solving classification problems where the data is separable into distinct classes.

SVM is known for its ability to handle high-dimensional data and its versatility in handling both linear and non-linear classification problems. However, it may require careful parameter tuning, and the choice of the appropriate kernel function can greatly impact its performance. SVM is a valuable tool in machine learning when used judiciously for the right type of problems.

Characteristics of SVM are:

* **Maximum Margin:** SVM is known for its ability to find a hyperplane that maximizes the margin between two classes. The margin is the distance between the hyperplane and the nearest data points (support vectors) from each class. This maximization of the margin helps SVM generalize well to unseen data and is a key characteristic of the algorithm.
* **Support Vectors:** Support vectors are the data points that are closest to the decision boundary (hyperplane). They play a crucial role in defining the hyperplane's position and the margin. SVM focuses on these support vectors during training, which makes it memory-efficient and robust to outliers.
* Kernel Trick: SVM can handle non-linearly separable data by applying a kernel function. The kernel function maps the data into a higher-dimensional space, where it becomes linearly separable. This allows SVM to solve complex classification problems that wouldn't be solvable with a linear model alone.
* **Optimization**: The training of SVM involves solving a quadratic optimization problem, which can be efficiently solved using specialized optimization techniques.
* Outliers Robustness: SVM is relatively robust to outliers in the training data because it focuses on the support vectors, which are the closest points to the decision boundary. Outliers far from the boundary have little influence.

# 3. PROGRAM & OUTPUT

from sklearn import datasets

from sklearn.model\_selection import train\_test\_split

from sklearn.preprocessing import StandardScaler

from sklearn.svm import SVC

from sklearn.metrics import accuracy\_score, classification\_report

# Load the Wine dataset

wine = datasets.load\_wine()

X = wine.data

y = wine.target

# Split the data into training and testing sets

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.3, random\_state=42)

# Standardize features by removing the mean and scaling to unit variance

scaler = StandardScaler()

X\_train = scaler.fit\_transform(X\_train)

X\_test = scaler.transform(X\_test)

# Create an SVM classifier

svm\_classifier = SVC(kernel='linear', C=1.0, random\_state=42)

# Train the SVM classifier

svm\_classifier.fit(X\_train, y\_train)

# Make predictions on the test set

predictions = svm\_classifier.predict(X\_test)

# Evaluate the model

accuracy = accuracy\_score(y\_test, predictions)

report = classification\_report(y\_test, predictions, target\_names=wine.target\_names)

# Print the results

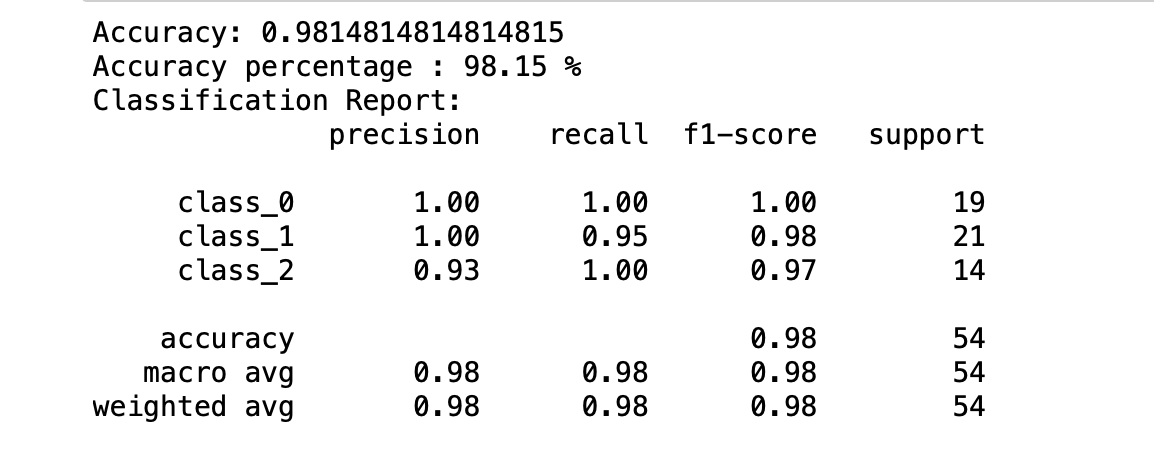
print("Accuracy:", accuracy)

print('Accuracy percentage :',round((accuracy\*100),2),"%")

print("Classification Report:")

print(report)

**OUTPUT:**



# ALGORITHM:

1. Data Loading: It loads the Wine dataset from scikit-learn's built-in datasets.
2. Data Splitting: The dataset is split into training and testing sets using train\_test\_split() to evaluate the model's performance.
3. Data Standardization: Feature scaling is performed using StandardScaler() to standardize the feature values.
4. SVM Classifier Creation: An SVM classifier with a linear kernel and C=1.0 is created using SVC() from scikit-learn.
5. Model Training: The SVM classifier is trained on the standardized training data using fit().
6. Model Prediction: The trained model is used to make predictions on the test data using predict().
7. Model Evaluation: Accuracy and a classification report are computed using accuracy\_score() and classification\_report(), respectively.
8. Results Printing: The accuracy and classification report are printed to the console.

# 5. CONCLUSION

* The SVM model's performance can be summarized based on the accuracy and the information provided in the classification report.
* The SVM classifier, as implemented in this code, is a solid choice for the Wine dataset classification task, given its ability to maximize the margin and generalize well to unseen data.
* The classification report provides insights into the model's precision, recall, and F1-score for each class, helping to understand its performance on each class separately.