**Experiment No.5**

**Title:** SVM Classification Algorithm

**Batch: 3 Roll No.: 16010421119 Experiment No.:5 Aim:** Design and implement of SVM classification algorithm.

**Resources needed:** Python 3.6 onwards, RapidMiner

# Theory:

The Support Vector Machine (SVM) is a Supervised Learning algorithm , it is powerful and widely used for classification and regression tasks in machine learning. It's particularly effective for binary classification tasks, where the goal is to separate data points of two different classes by finding an optimal hyperplane that maximizes the margin between them.

# SVM Classification Algorithm:

**Data Preparation:** Prepare your labelled training data. Each data point should have a set of features (attributes) and a corresponding class label indicating its category (e.g., 0 or 1, A or B).

**Feature Scaling:** It's essential to scale the features so that they have a similar range. Common scaling techniques include normalization or standardization.

**Kernel Selection:** Choose a kernel function to transform the data into a higher-dimensional space. SVM uses the kernel trick to find non-linear decision boundaries. Commonly used kernels are:

**Linear Kernel:** For linearly separable data (the default kernel when no other kernel is specified).

**Polynomial Kernel:** Suitable for data with polynomial patterns.

**Radial Basis Function (RBF) Kernel (Gaussian Kernel):** Suitable for non-linear data and is the most popular kernel choice.

**Sigmoid Kernel:** Useful for neural network-inspired architectures.

**Parameter Selection:** SVM has two crucial parameters to be set:

**C (Cost parameter):** Controls the trade-off between maximizing the margin and minimizing the classification error on the training data. A smaller C allows for a wider margin but may result in misclassifying some training examples, while a larger C aims to classify all examples correctly, potentially leading to a narrower margin.

**Kernel-specific parameters (e.g., gamma for RBF kernel):** The effect and interpretation of these parameters depend on the selected kernel. Proper tuning is essential for optimal performance.

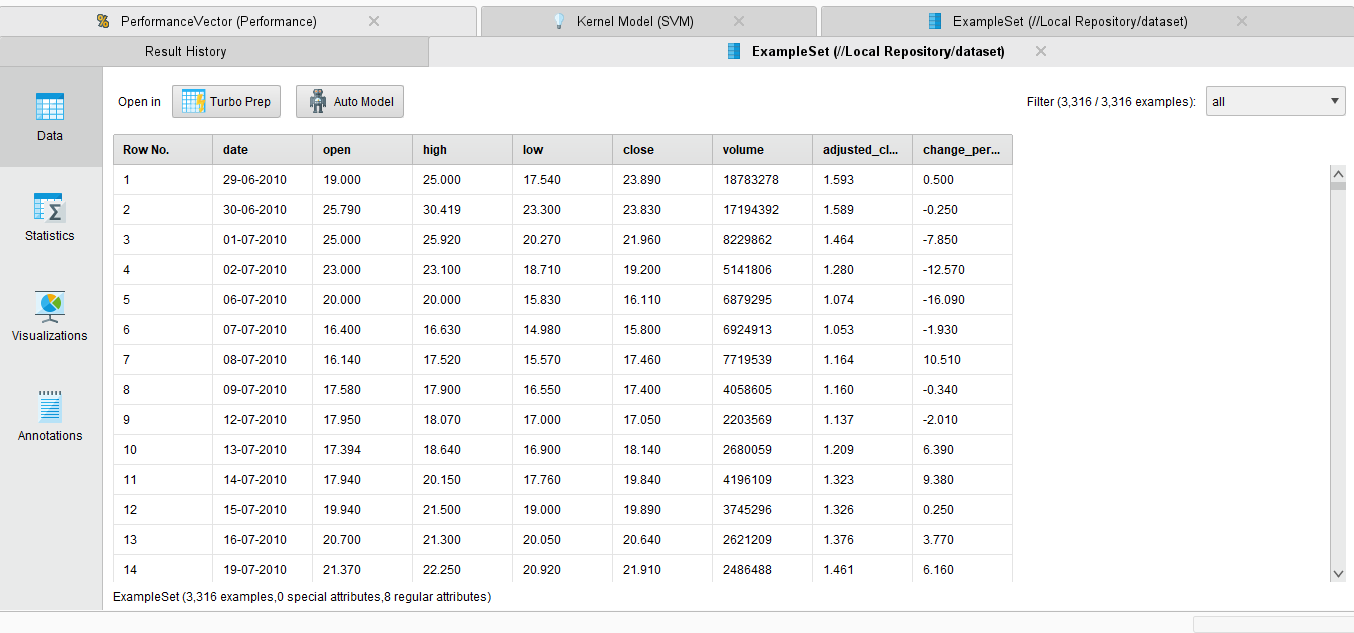
**Model Training:** Feed the scaled and transformed data into the SVM algorithm, which optimizes the position of the hyperplane to achieve the best separation between classes. This process involves solving a convex optimization problem to find the optimal weight vector and bias term that define the hyperplane.

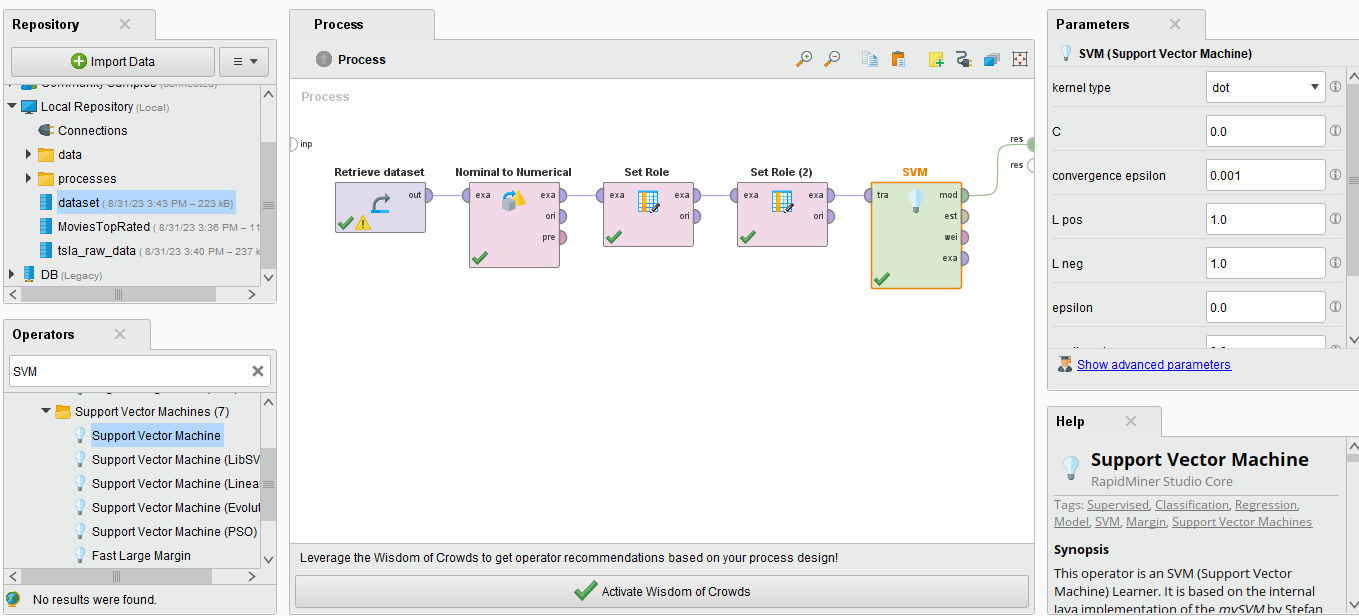
# Procedure / Approach /Algorithm / Activity Diagram:

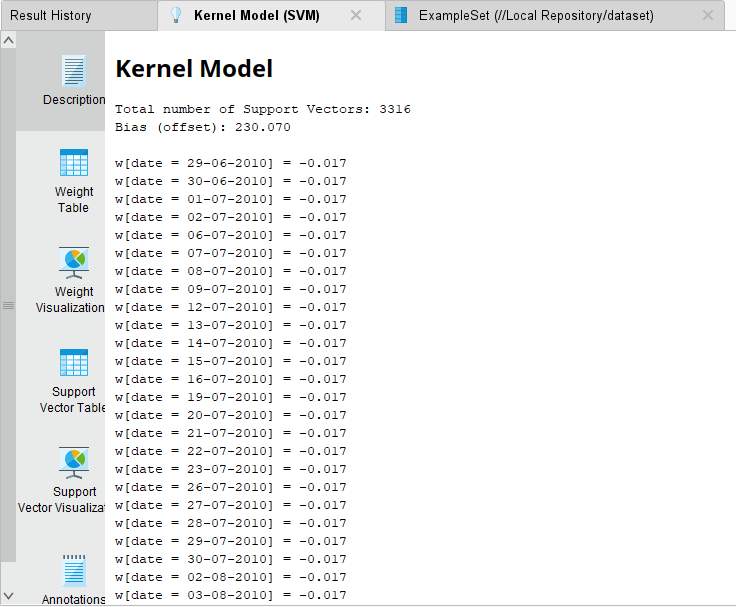
1. Identify attributes suitable for applying classification algorithm
2. Implement SVM on your dataset using Python and RapidMiner.
3. Apply SVM to classify unknown tuple.

# Results: (Program printout with output / Document printout as per the format)

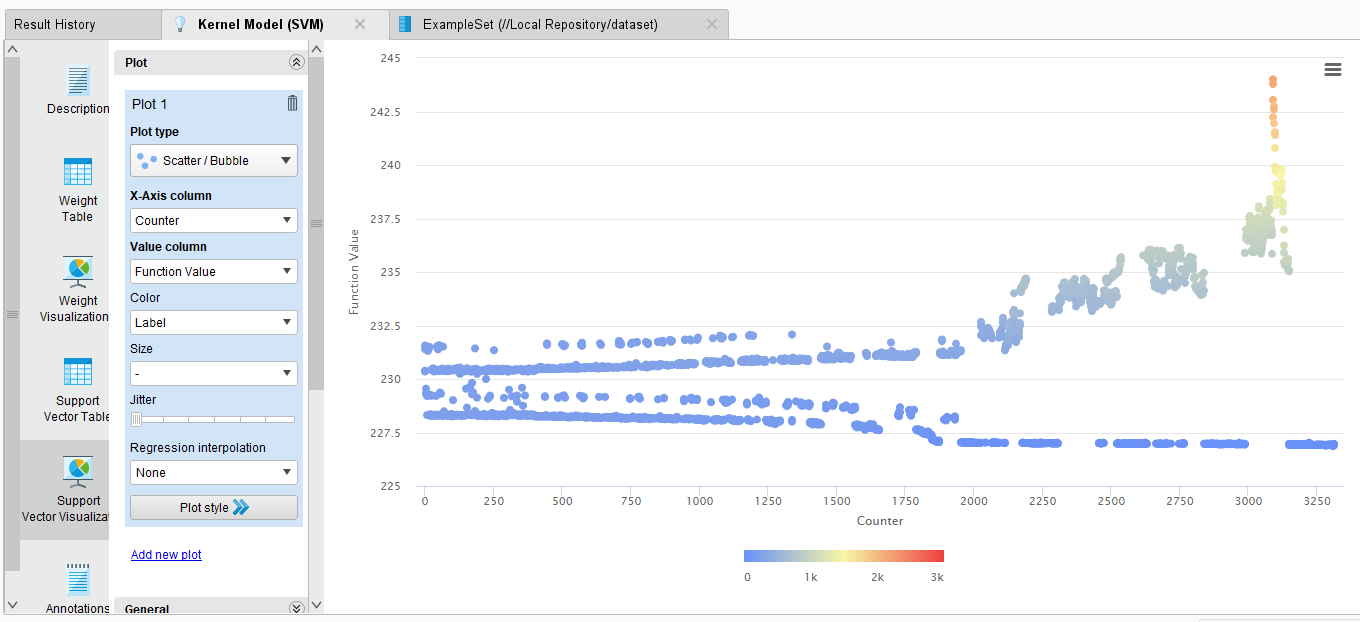
Dataset:



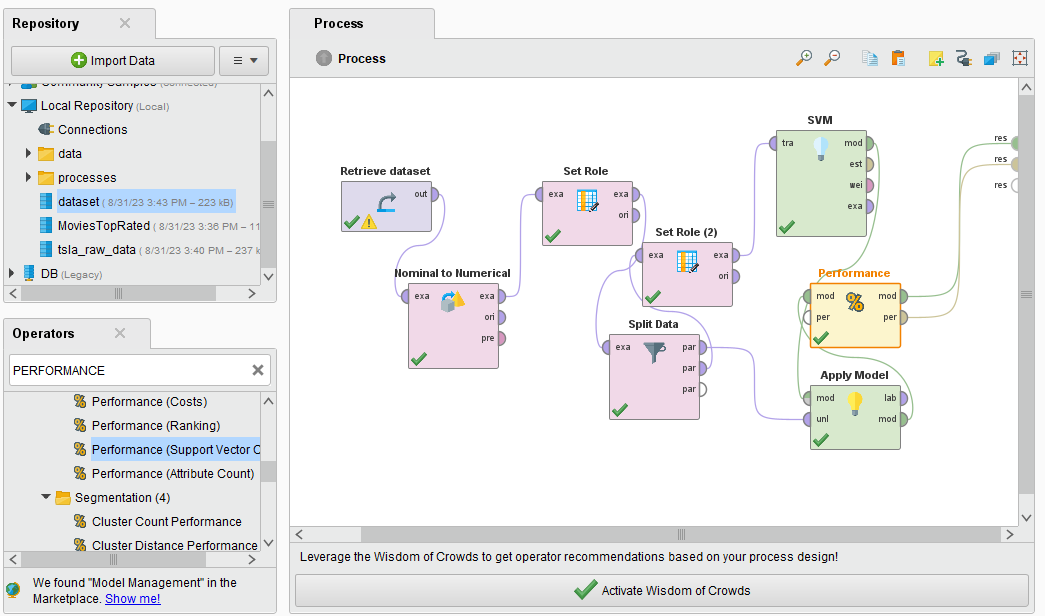


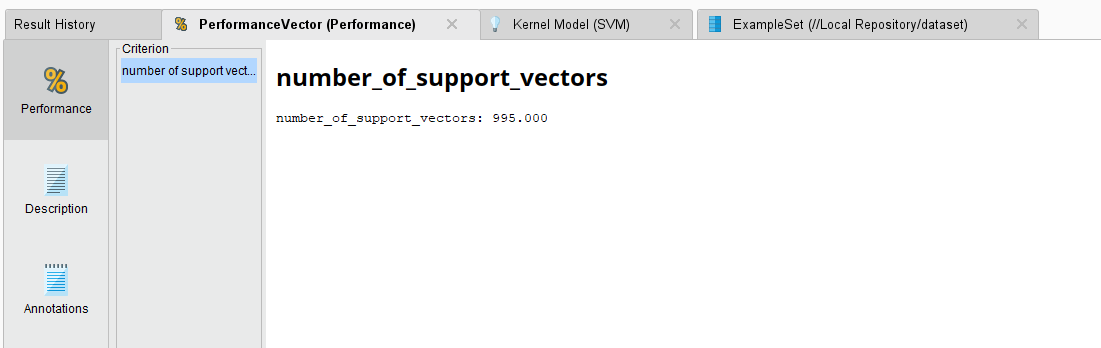






Calculating Performance





import numpy as np

from sklearn import datasets

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

from sklearn.svm import SVC

from sklearn.metrics import accuracy\_score

iris = datasets.load\_iris()

X = iris.data[:, :2] # Use only the first two features for simplicity

y = iris.target

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

svm\_classifier = SVC(kernel=&#39;linear&#39;, C=1) # Linear kernel, regularization parameter C=1

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

y\_pred = svm\_classifier.predict(X\_test)

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

print(f&#39;Accuracy: {accuracy \* 100:.2f}%&#39;)

**Questions:**

1. What are advantages and disadvantages of SVM?

ANS:

**Advantages of SVM:**

1. Effective in High-Dimensional Spaces: SVMs perform well in high-dimensional spaces, making them suitable for tasks with a large number of features.

2. Robust to Overfitting: SVMs are less prone to overfitting compared to some other machine learning algorithms. They aim to find a hyperplane that maximizes the margin between classes, which helps generalize well to new data.

3. Versatile: SVMs can be used for both classification and regression tasks, as well as for outlier detection.

**Disadvantages of SVM:**

1. Computational Complexity: SVMs can be computationally expensive, especially when working with large datasets. The training time can be slow, and the memory requirements can be significant.

2. Sensitivity to Hyperparameters: SVMs have several hyperparameters to tune, such as the choice of kernel, regularization parameter (C), and kernel-specific parameters. The performance of SVM can be highly sensitive to these hyperparameters, and finding the right combination can be challenging.

3. Lack of Interpretability: SVMs often provide accurate predictions, but the resulting models can be difficult to interpret, especially when non-linear kernels are used.

1. Explain over fitting problem in Machine Learning.

ANS: Overfitting is a common problem in machine learning that occurs when a model learns the training data too well, to the extent that it captures not just the underlying patterns but also the noise and randomness in the data. In other words, an overfit model performs extremely well on the training data but fails to generalize its predictions to unseen or new data, which is the primary goal of any machine learning model. Overfitting is a situation where the model fits the training data too closely, becoming overly complex and, as a result, losing its ability to make accurate predictions on new, unseen data.

# Outcomes:

# CO3 Comprehend radial-basis-function (RBF) networks and Kernel learning method

**Conclusion**: In this experiment we designed and implemented of SVM classification algorithm.

**Grade: AA / AB / BB / BC / CC / CD /DD**

Signature of faculty in-charge with date

# References:

Books/ Journals/ Websites:

* 1. Han, Kamber, "Data Mining Concepts and Techniques", Morgan Kaufmann 3nd Edition