Practical no.6

**Aim:** Train an SVM regress or on the California Housing Dataset

# Theory

SVM is a powerful algorithm that can be used for both classification and regression tasks. In regression, the goal is to predict a continuous target variable, as opposed to a categorical target variable in classification. The California Housing Dataset is a real-world dataset that contains information on housing prices in California. By applying SVM regression on the California Housing Dataset, we can learn how to use SVM for regression tasks, and how to fine-tune the hyper parameters of the model to achieve better performance.

# Material

* sklearn

# Program

from sklearn.datasets import fetch\_california\_housing from sklearn.model\_selection import train\_test\_split from sklearn.preprocessing import StandardScaler from sklearn.svm import SVR

from sklearn.metrics import mean\_squared\_error

# Load the California Housing dataset housing = fetch\_california\_housing()

# Split the data into training and testing sets

X\_train, X\_test, y\_train, y\_test = train\_test\_split(housing.data, housing.target, test\_size=0.2, random\_state=42)

# Scale the data using StandardScaler scaler = StandardScaler()

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

# Train an SVM regressor on the training data svr = SVR(kernel='linear', C=1.0, epsilon=0.1) svr.fit(X\_train, y\_train)

# Evaluate the model on the testing data y\_pred = svr.predict(X\_test)

mse = mean\_squared\_error(y\_test, y\_pred) print("Mean squared error:", mse)

# Output

