|  |  |
| --- | --- |
| Algorithm: Linear Kernel SVM | |
| USN: 1MS18CS024 | NAME: Aravind S |
| USN: 1MS18CS029 | NAME: Arun J Kennedy |

**Description of the Algorithm: <<Write 2-3 Paragraphs about the Algorithm>>**

The objective of the support vector machine algorithm is to find a hyperplane in an N-dimensional space(where N is the number of features) that distinctly classifies the data points.

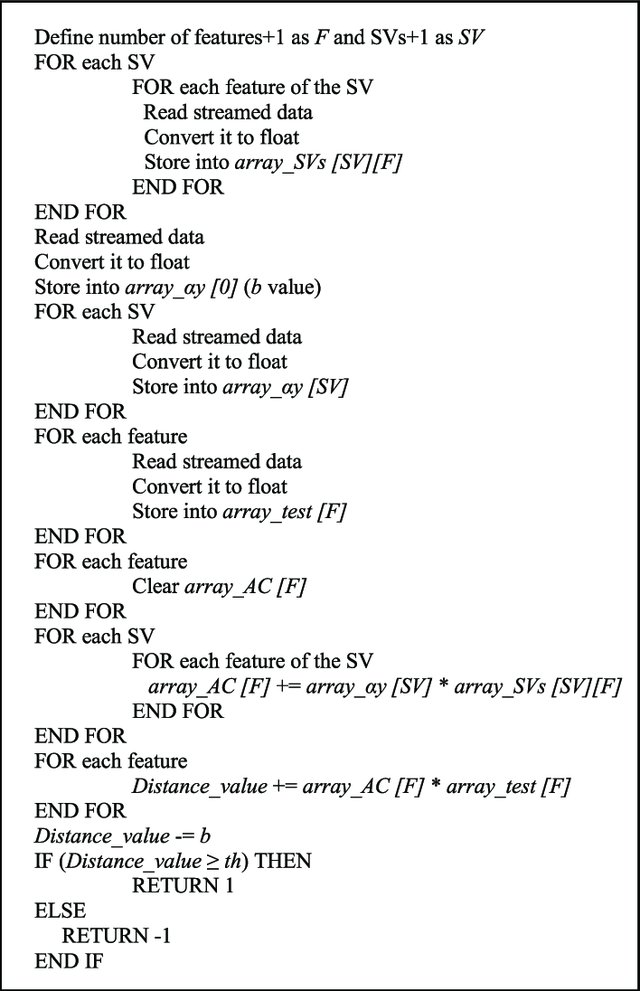
Hyperplanes are decision boundaries that help classify the data points. Data points falling on either side of the hyperplane can be attributed to different classes.

SVM maps training examples to points in space so as to maximize the width of the gap between the two categories. New examples are then mapped into that same space and predicted to belong to a category based on which side of the gap they fall.

Possible Hyperplanes Optimal hyperplane

**Algorithm Pseudocode:**



**Data set Used: (Attach Screen shot of the few rows and also the Kaggle/Dataset link)**

The data points were generated using np.random.

The following function provides a distribution of N points in k clusters of Income vs Age.

def createClusteredData(N, k):  
 np.random.seed(1234)  
 pointsPerCluster = float(N)/k  
 X = []  
 y = []  
 for i in range (k):  
 incomeCentroid = np.random.uniform(20000.0, 200000.0)  
 ageCentroid = np.random.uniform(20.0, 70.0)  
 for j in range(int(pointsPerCluster)):  
 X.append([np.random.normal(incomeCentroid, 10000.0), np.random.normal(ageCentroid, 2.0)])  
 y.append(i)  
 X = np.array(X)  
 y = np.array(y)  
 return X, y

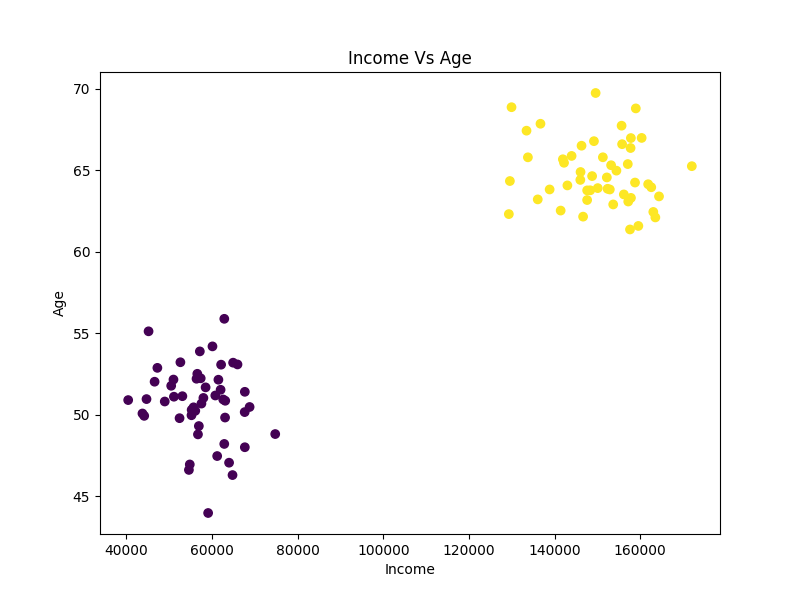


Figure 1: Scatter plot showing how the datapoints are distributed

**Challenges faced during the implementation of the program:**

Due to age being in the range of 0 to 100 normally and Income being in the range of 1000s the SVM is more sensitive to difference in income.

To make the SVM unbiased we scale the features between -1 to 1.

**Advantages & Disadvantages of the Algorithm:**

|  |  |
| --- | --- |
| Advantages | Disadvantages |
| SVM works relatively well when there is a clear margin of separation between classes. | SVM algorithm is not suitable for large data sets. |
| SVM is more effective in high dimensional spaces. | SVM does not perform very well when the data set has more noise i.e., target classes are overlapping. |
| SVM is effective in cases where the number of dimensions is greater than the number of samples. | In cases where the number of features for each data point exceeds the number of training data samples, the SVM will underperform. |
| SVM is relatively memory efficient | As the support vector classifier works by putting data points, above and below the classifying hyperplane there is no probabilistic explanation for the classification. |

**Applications of the Algorithm:**

* SVMs are helpful in text and hypertext categorization, as their application can significantly reduce the need for labeled training instances in both the standard inductive and transductive settings. Some methods for shallow semantic parsing are based on support vector machines.
* Classification of images can also be performed using SVMs
* Hand-written characters can be recognized using SVM.
* The SVM algorithm has been widely applied in the biological and other sciences. They have been used to classify proteins with up to 90% of the compounds classified correctly.
* Permutation tests based on SVM weights have been suggested as a mechanism for interpretation of SVM models.
* Posthoc interpretation of support-vector machine models in order to identify features used by the model to make predictions is a relatively new area of research with special significance in the biological sciences.

**Output: (Screen shots)**

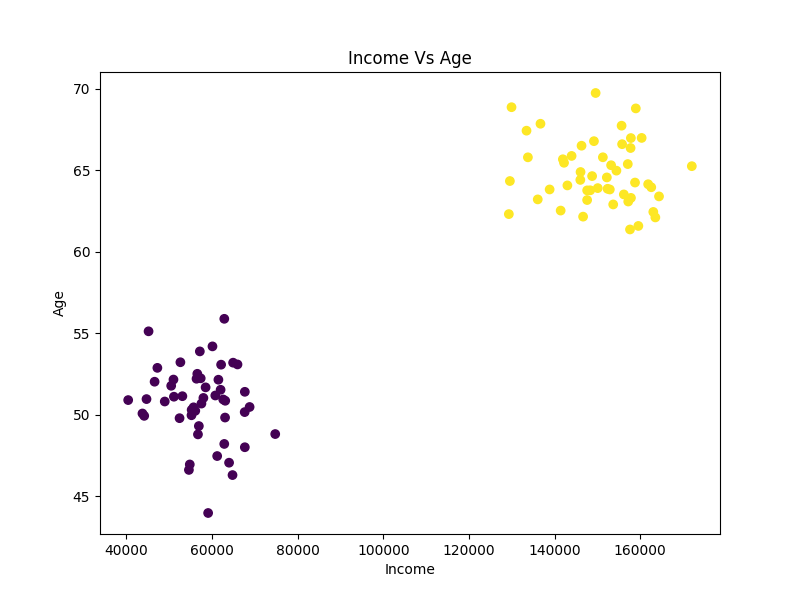


Figure 2: Income Vs Age

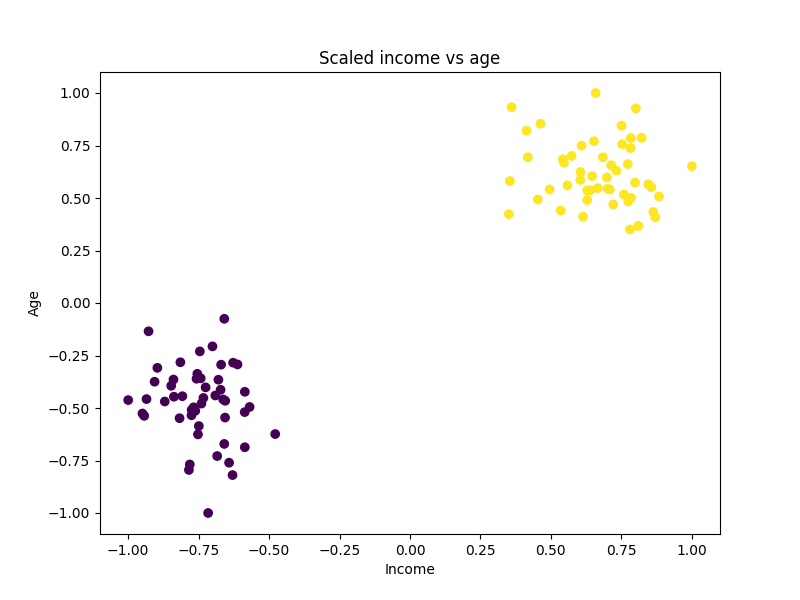


Figure 3: Scaled income vs age

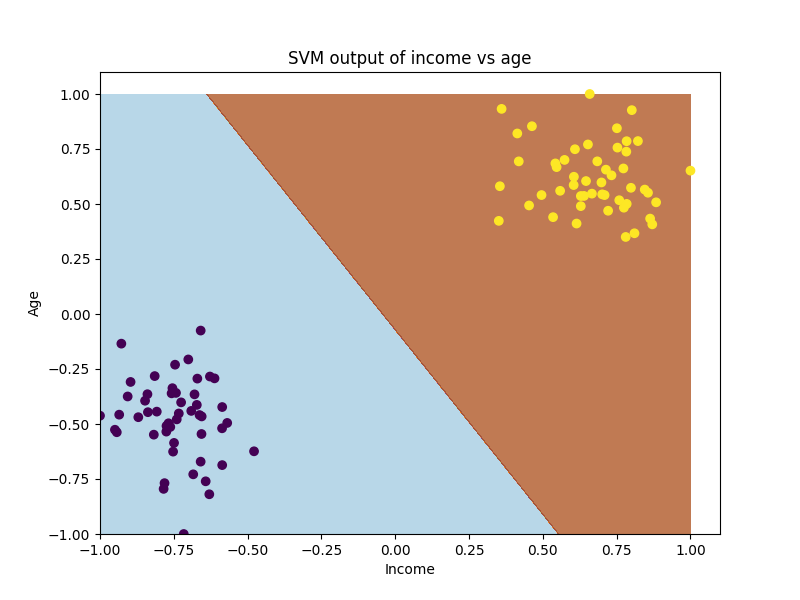


Figure 4: SVM model

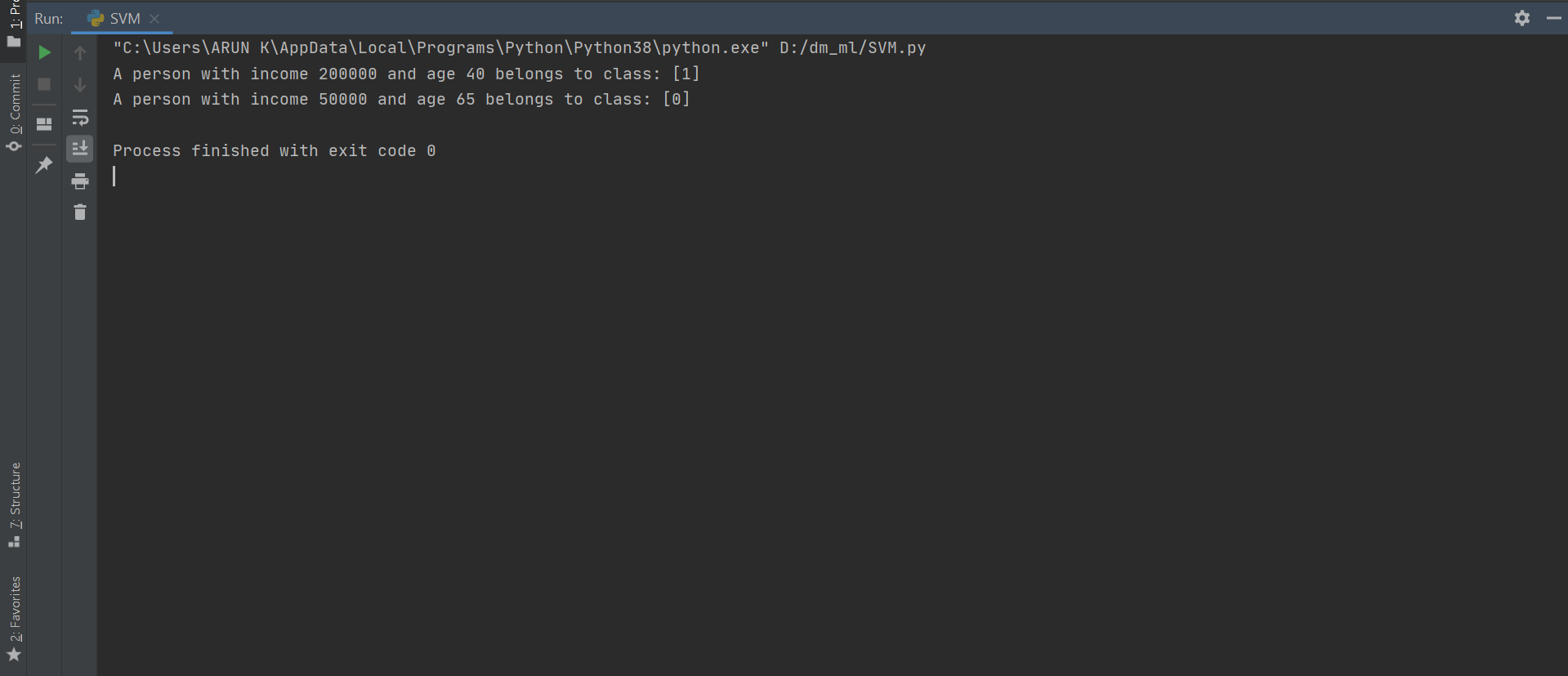


Figure 5: Console output

**References:**

<https://en.wikipedia.org/wiki/Support-vector_machine>

<https://towardsdatascience.com/support-vector-machine-introduction-to-machine-learning-algorithms-934a444fca47>

<https://www.w3schools.com/python/matplotlib_labels.asp>

<https://www.w3schools.com/python/matplotlib_scatter.asp>

[1]Noble, W.S., 2006. What is a support vector machine?. Nature biotechnology, 24(12), pp.1565-1567.

[2]Joachims, Thorsten. "Svmlight: Support vector machine." *SVM-Light Support Vector Machine http://svmlight. joachims. org/, University of Dortmund* 19, no. 4 (1999).