**Loan Approval Prediction**

**ABSTRACT: -**

Loan Approval Prediction is a significant project addressing the crucial task of predicting whether loan applications will be approved or denied. With financial institutions facing the challenge of efficiently evaluating numerous loan applications, machine learning offers a promising solution. This project focuses on implementing two machine learning algorithms: Support Vector Machine (SVM) as the proposed algorithm and Random Forest as the existing algorithm. SVM is chosen for its ability to handle high-dimensional data and effectively classify applicants into approved or denied categories, while Random Forest serves as a benchmark for comparison due to its robustness and scalability. The system processes various applicant features such as credit history, income, employment status, and loan amount, extracting meaningful patterns to predict loan approval outcomes. By training the models on historical loan data and evaluating their performance using metrics like accuracy, the project aims to provide financial institutions with valuable insights to streamline their loan approval process, reduce risk, and improve decision-making efficiency. Through accurate prediction of loan outcomes, this project contributes to enhancing the overall efficiency and effectiveness of the lending industry.

|  |  |
| --- | --- |
| **EXSISTING SYSTEM** | **PROPOSED SYSTEM** |
| * Random Forest is a powerful ensemble learning algorithm used for both classification and regression tasks. It was introduced by Leo Breiman and Adele Cutler in 2001 and has since become one of the most popular and widely used machine learning algorithms. Random Forest is based on the concept of decision trees, where multiple decision trees are built during training and predictions are made by aggregating the results of individual trees. * Random Forest is known for its robustness and ability to handle high-dimensional data with ease. It performs well on both structured and unstructured data and is less prone to overfitting compared to individual decision trees. Additionally, Random Forest provides an estimate of feature importance, allowing users to understand which features contribute most to the predictions. | * Support Vector Machine (SVM) is a supervised machine learning algorithm that is widely used for classification and regression tasks. Developed by Vladimir Vapnik and his colleagues in the 1990s, SVM is based on the concept of finding the optimal hyperplane that best separates data points belonging to different classes in a high-dimensional space. It is known for its ability to handle both linear and non-linear classification problems efficiently. * SVM can classify new data points by examining which side of the hyperplane they fall on. Data points on one side of the hyperplane are classified as one class, while those on the other side are classified as the other class. |
| **EXISTING ALGORITHM**   * Random forest | **PROPOSED ALGORITHM: -**   * Support Vector Machine (SVM) |
| * Random Forest works by constructing a multitude of decision trees during training. Each tree is trained on a subset of the training data, sampled with replacement (bootstrap sample), and a subset of features randomly selected at each node. This randomness ensures that each tree in the forest learns different patterns from the data. * During prediction, the output of each decision tree is aggregated to make the final prediction. For classification tasks, the most common class among the predictions of individual trees is selected as the final prediction. For regression tasks, the average of the predictions from all trees is taken as the final prediction. | **ALGORITHM DEFINITION: -**   * SVM works by mapping input data points into a higher-dimensional space using a mathematical function called a kernel. In this space, SVM tries to find the hyperplane that maximizes the margin, which is the distance between the hyperplane and the nearest data points (support vectors) from each class. By maximizing the margin, SVM aims to achieve better generalization and robustness to unseen data. * For linearly separable data, SVM finds the optimal hyperplane that separates the classes with the largest margin. However, in real-world scenarios where data may not be linearly separable, SVM can still perform well by using different kernel functions such as polynomial, radial basis function (RBF), or sigmoid to map the data into a higher-dimensional space where separation is possible. |
| **DRAWBACKS: -**   * Random Forest can be computationally expensive and memory-intensive, especially when dealing with large datasets and a large number of trees. * While Random Forest provides accurate predictions, it is often considered a black box model, meaning it can be challenging to interpret the reasoning behind individual predictions. | **ADVANTAGES: -**   * SVM performs well in high-dimensional spaces, making it suitable for tasks with many features, such as text classification, image recognition, and gene expression analysis. * SVM supports various kernel functions, allowing it to handle both linear and non-linear classification problems effectively. |

**SYSTEM ARCHITECTURE:**

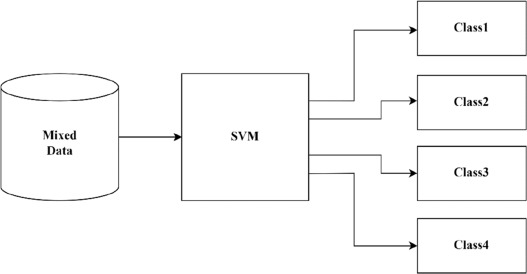


Fig:- proposed model

**MINIMUMSYSTEM REQUIREMENTS**

**HARDWARE REQUIREMENTS**

* PROCESSOR : Pentium i3 Processor
* RAM : 2GB DD RAM
* HARD DISK : 250 GB

**SOFTWARE REQUIREMENTS**

* BACK END : PYTHON
* OPERATING SYSTEM : WINDOWS 7
* IDE : Spyder3