**The below theory has to be written in ruled pages.**

**All diagrams on white page.**

**In white pages of the record, take the print of code and visualized graphs and output and paste the code**

### **1. Regression**

#### ****About the Algorithm****:

Regression is a statistical and machine learning method to model and analyze relationships between dependent (target) and independent (predictor) variables. It predicts continuous numerical outcomes.

#### ****Types****:

1. **Linear Regression**: Predicts using a straight-line relationship.
2. **Logistic Regression**: For binary classification; uses a sigmoid function.
3. **Polynomial Regression**: Fits non-linear data using polynomial terms.
4. **Ridge Regression**: Adds an L2L2-norm penalty to regularize.
5. **Lasso Regression**: Uses L1L1-norm for feature selection.

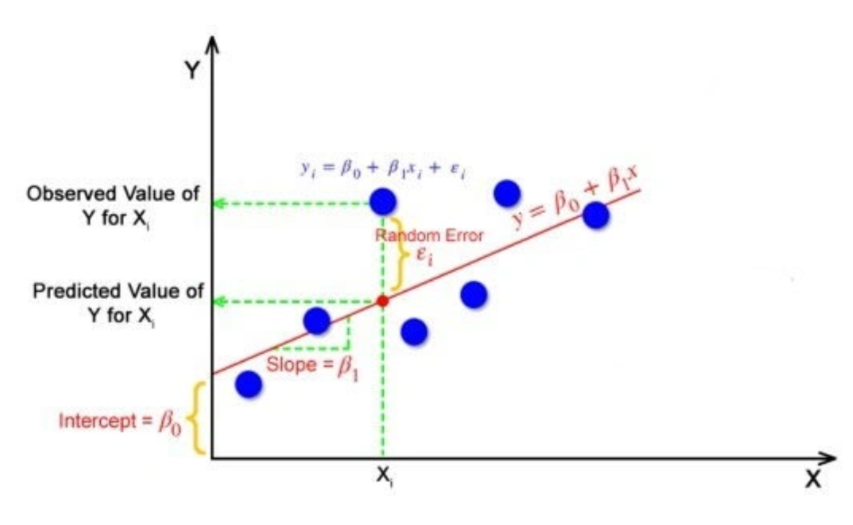


Fig: Simple linear regression

#### 

#### ****Applications****:

* Predicting sales, stock prices, or energy consumption.
* Analyzing trends and relationships in business and healthcare.

#### ****Limitations****:

* Assumes linearity.
* Sensitive to outliers.
* Prone to multicollinearity when predictors are correlated.

#### ****How to Overcome****:

* Use regularization techniques (e.g., Ridge or Lasso).
* Normalize data and remove outliers.
* Use polynomial regression for non-linear relationships.

### **2. Support Vector Machine (SVM)**

#### ****About the Algorithm****:

SVM is a supervised learning method for classification and regression tasks. It finds the hyperplane that separates data into distinct classes with maximum margin.

#### ****Types****:

1. **Linear SVM**: Used for linearly separable data.
2. **Non-linear SVM**: Applies kernel tricks (e.g., polynomial, RBF) for non-linear data.

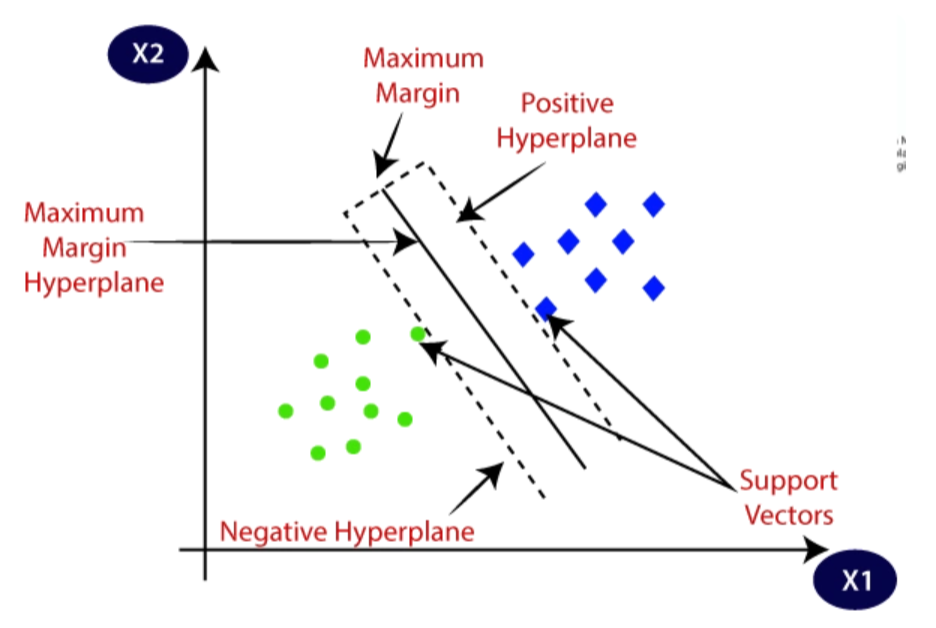


Fig: Support Vector Classifier

#### 

#### ****Applications****:

* Handwriting recognition.
* Bioinformatics (protein classification).
* Email spam detection.

#### ****Limitations****:

* Slow for large datasets.
* Requires careful kernel selection.

#### ****How to Overcome****:

* Use approximate SVM solvers like SVM-Light.
* Perform hyperparameter tuning.
* Employ dimensionality reduction techniques.

### **3. K-Means Clustering**

#### ****About the Algorithm****:

An unsupervised learning method for grouping data points into K clusters by minimizing intra-cluster variance.

#### ****Types****:

1. **K-means++**: Improves centroid initialization to speed up convergence.
2. **Mini-Batch K-means**: Processes data in smaller batches for efficiency.

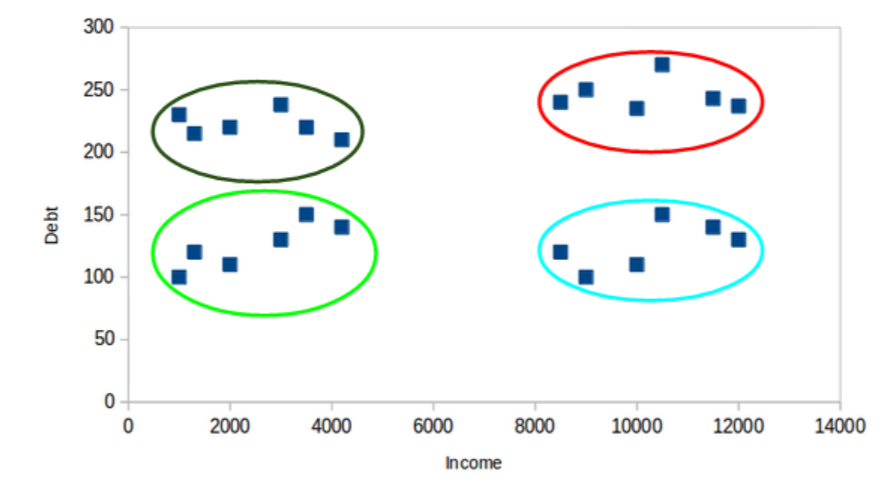


Fig: Grouping based on Income Vs Debt using KNN

#### ****Applications****:

* Customer segmentation.
* Image compression.
* Topic modeling in text analysis.

#### ****Limitations****:

* Sensitive to initialization.
* Assumes clusters are spherical and equally sized.

#### ****How to Overcome****:

* Use K-means++ for better initialization.
* Use silhouette analysis to determine the optimal K.
* Switch to density-based clustering for complex shapes.

### **4. K-Nearest Neighbors (KNN)**

#### ****About the Algorithm****:

KNN is a simple, instance-based learning algorithm. It classifies data points based on the majority class among their K nearest neighbors.

#### ****Types****:

1. **Weighted KNN**: Assigns higher weight to closer neighbors.
2. **Distance-based KNN**: Uses distances like Euclidean or Manhattan.

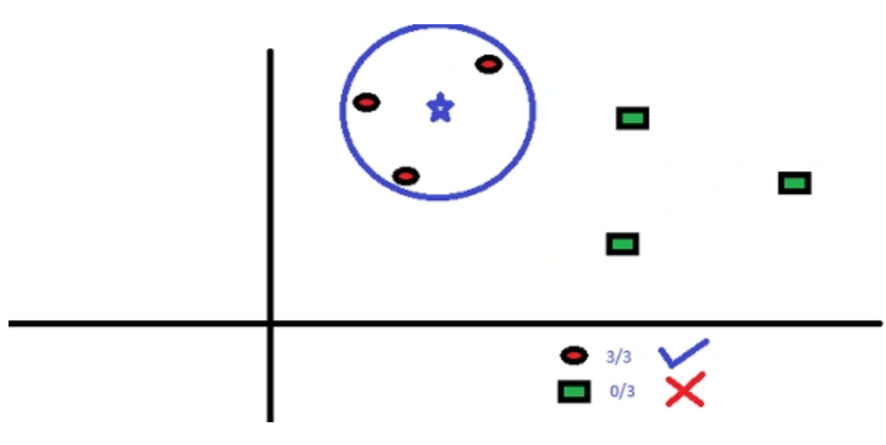


Fig: Classifying new data point blue star based on KNN

#### ****Logic or Maths****:

#### ****Applications****:

* Recommender systems.
* Disease classification (e.g., diabetes prediction).
* Fraud detection.

#### ****Limitations****:

* High computational cost for large datasets.
* Sensitive to irrelevant features.

#### ****How to Overcome****:

* Use dimensionality reduction techniques like PCA.
* Leverage KD-trees or Ball trees for faster neighbor search.

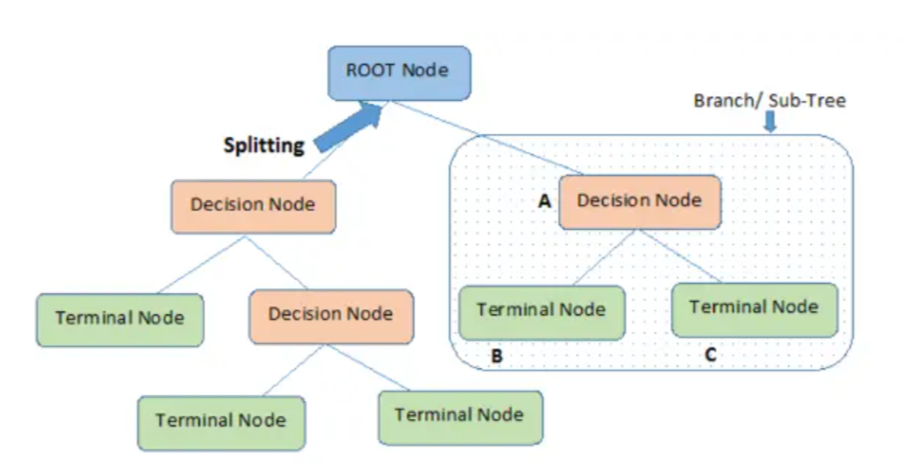
### **5. ID3 Decision Tree**

#### ****About the Algorithm****:

ID3 constructs decision trees by selecting features with the highest information gain at each node.

#### ****Types****:

1. **C4.5**: An improvement of ID3, handles continuous data.
2. **CART**: Uses Gini impurity for splitting.



#### ****Applications****:

* Customer segmentation.
* Medical diagnosis.
* Fraud detection.

#### ****Limitations****:

* Prone to overfitting.
* Biased towards features with more levels.

#### ****How to Overcome****:

* Use tree pruning techniques.
* Combine with ensemble methods like Random Forest.

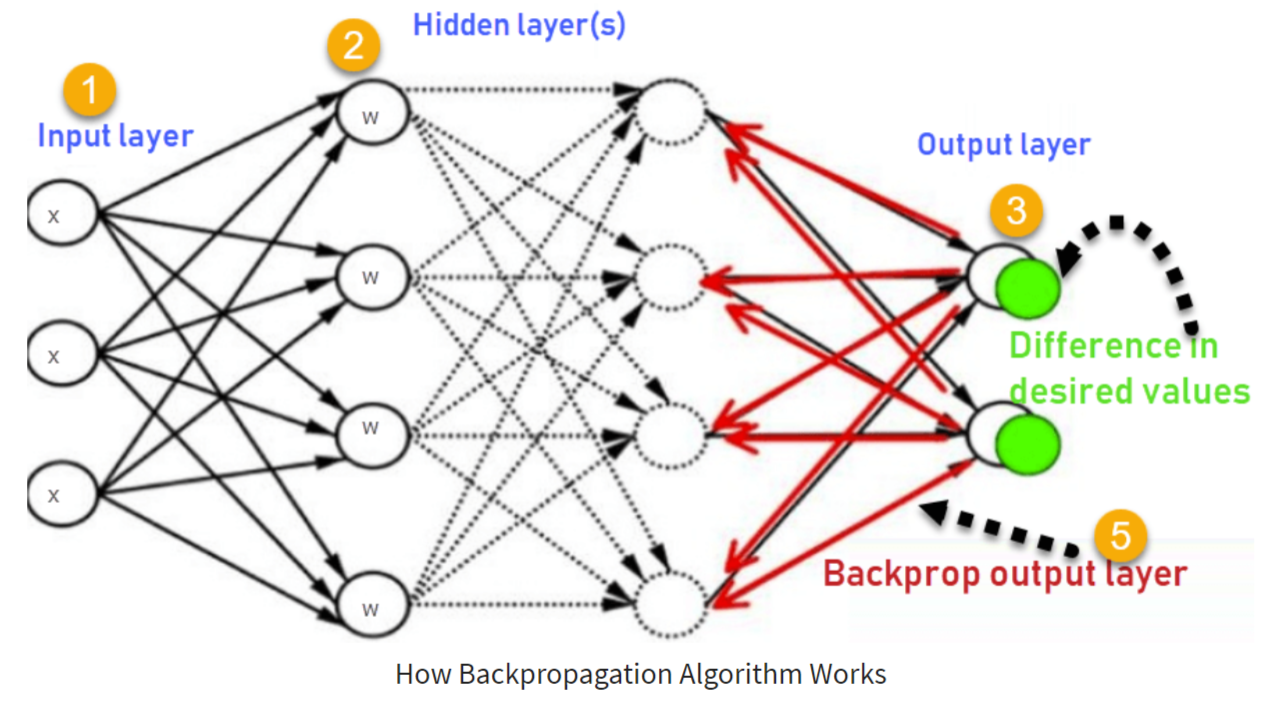
### **6. Backpropagation Algorithm**

#### ****About the Algorithm****:

Backpropagation optimizes the weights of neural networks by propagating errors backward and using gradient descent.

#### ****Types****:

1. **Stochastic Gradient Descent**: Updates weights after each example.
2. **Batch Gradient Descent**: Updates weights after all examples.
3. **Mini-batch Gradient Descent**: Updates weights after a batch of examples.



#### ****Applications****:

* Speech recognition.
* Image classification.
* Natural Language Processing.

#### ****Limitations****:

* Can get stuck in local minima.
* Suffers from vanishing or exploding gradients.
* Requires large datasets.

#### ****How to Overcome****:

* Use advanced optimizers like Adam or RMSProp.
* Use ReLU activations to combat vanishing gradients.
* Apply batch normalization to stabilize training.