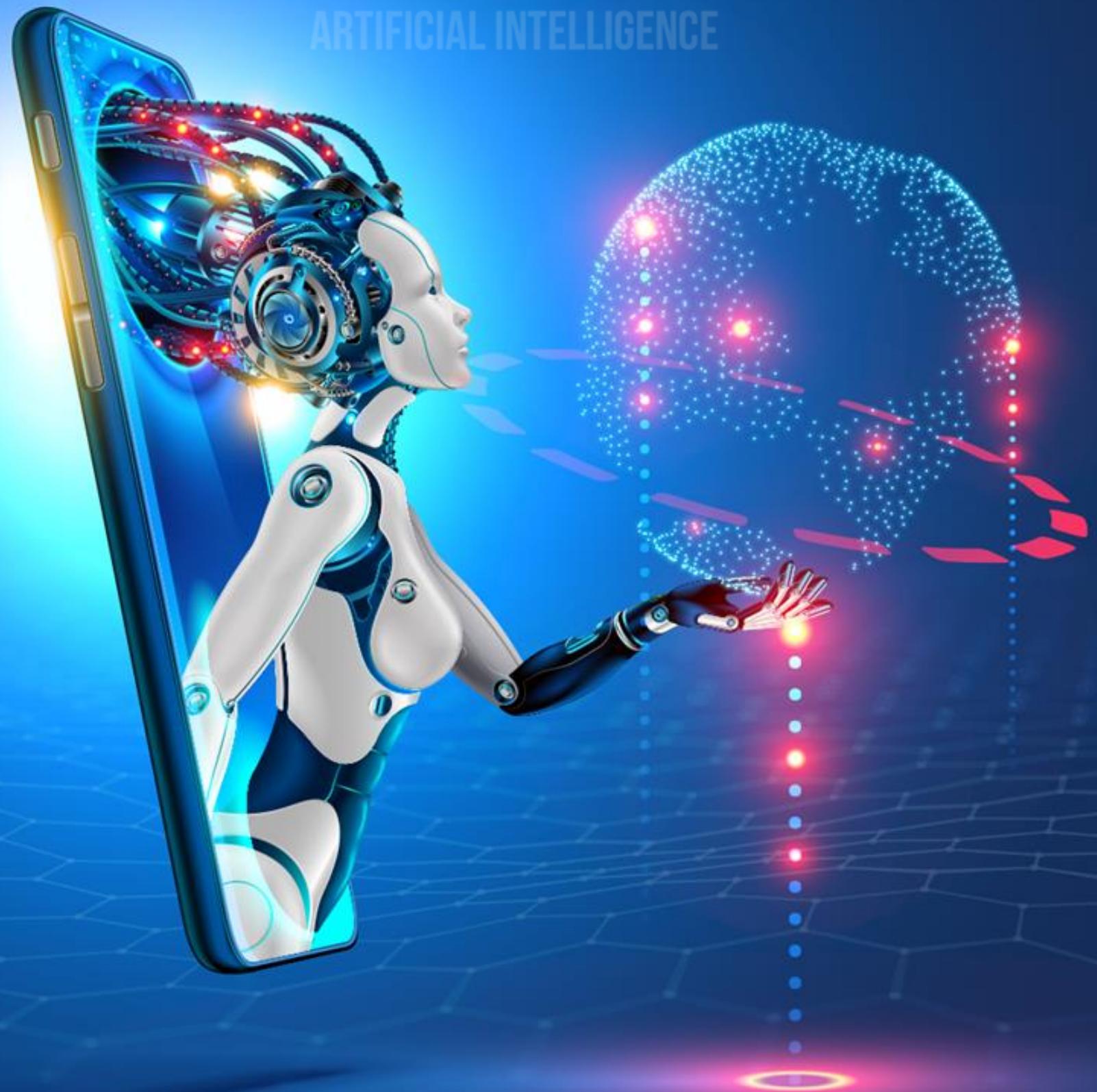


DATA AND ARTIFICIAL INTELLIGENCE



Data Analytics with R

DATA AND ARTIFICIAL INTELLIGENCE



Classification

Business Scenario

- George works in customer service department of a BPO. His KRA includes decreasing the waiting time for incoming service calls.

Approach: George needs to identify the specific behavioral trends of the customers calling and identify the calls that might be for minor issues. To identify such calls, George needs to develop a classification model classifying the priority of calls.



Learning Objectives

By the end of this lesson, you will be able to:

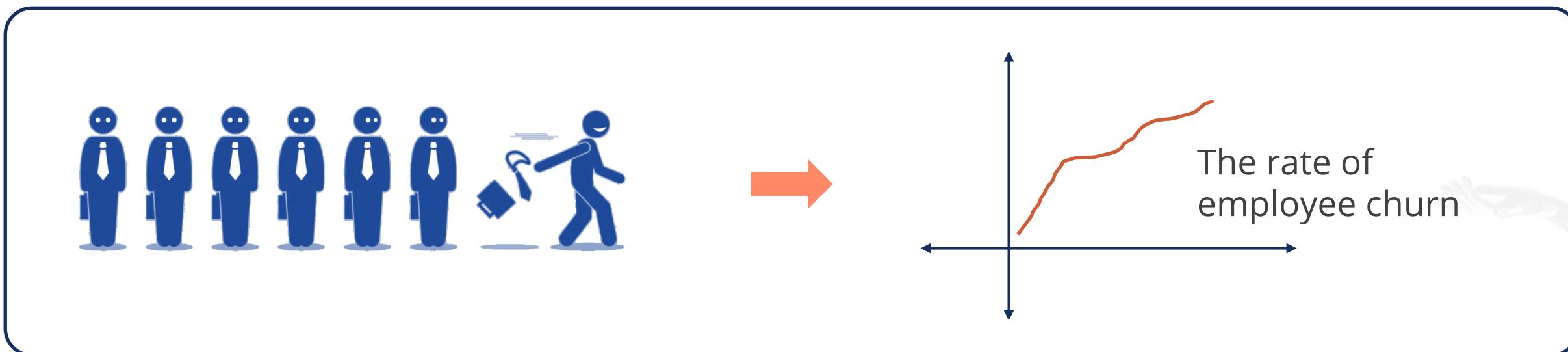
- Perform classification
- Apply logistic regression algorithm to solve classification problem
- Analyze the k-nearest neighbors algorithm
- Implement decision tree and random forest algorithm
- Use support vector and Naïve Bayes models for classification



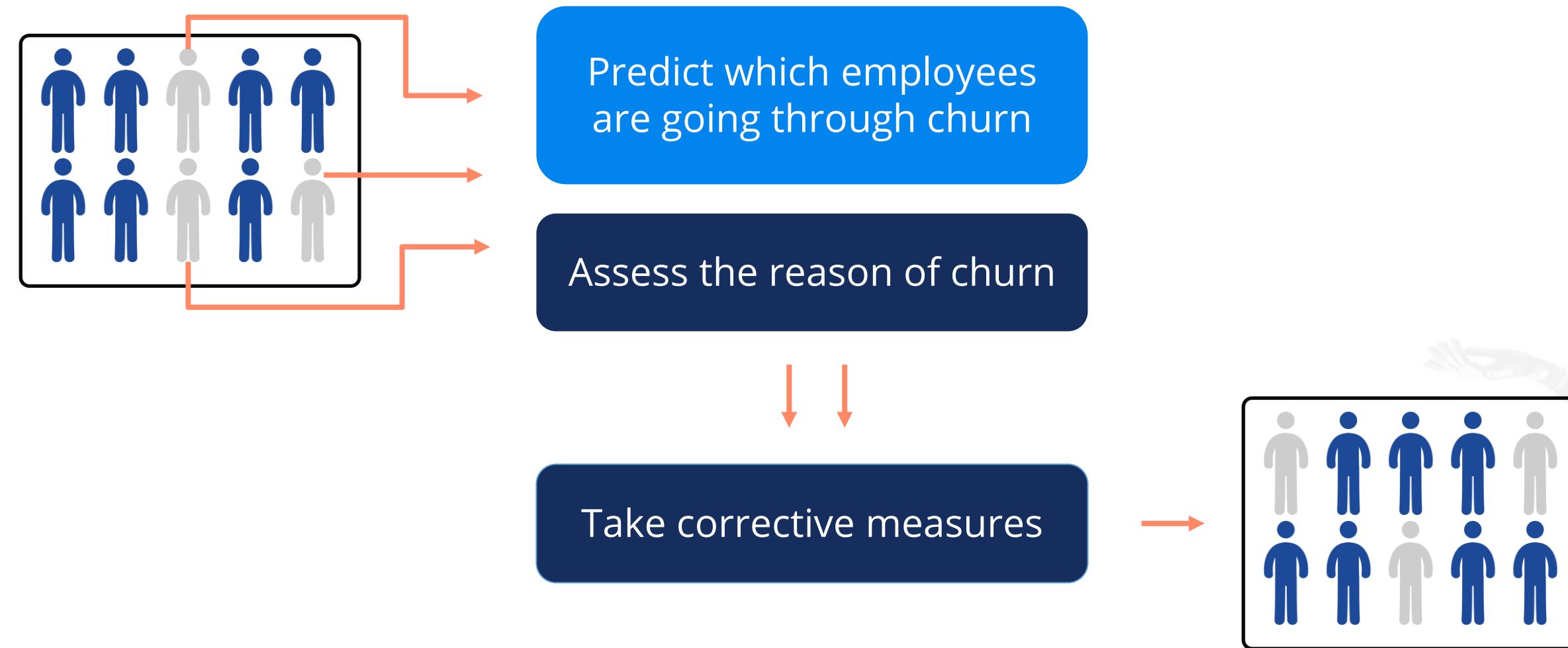
Introduction to Classification

Employee Attrition

Do you want to identify the employees who might leave and know the reason for attrition?

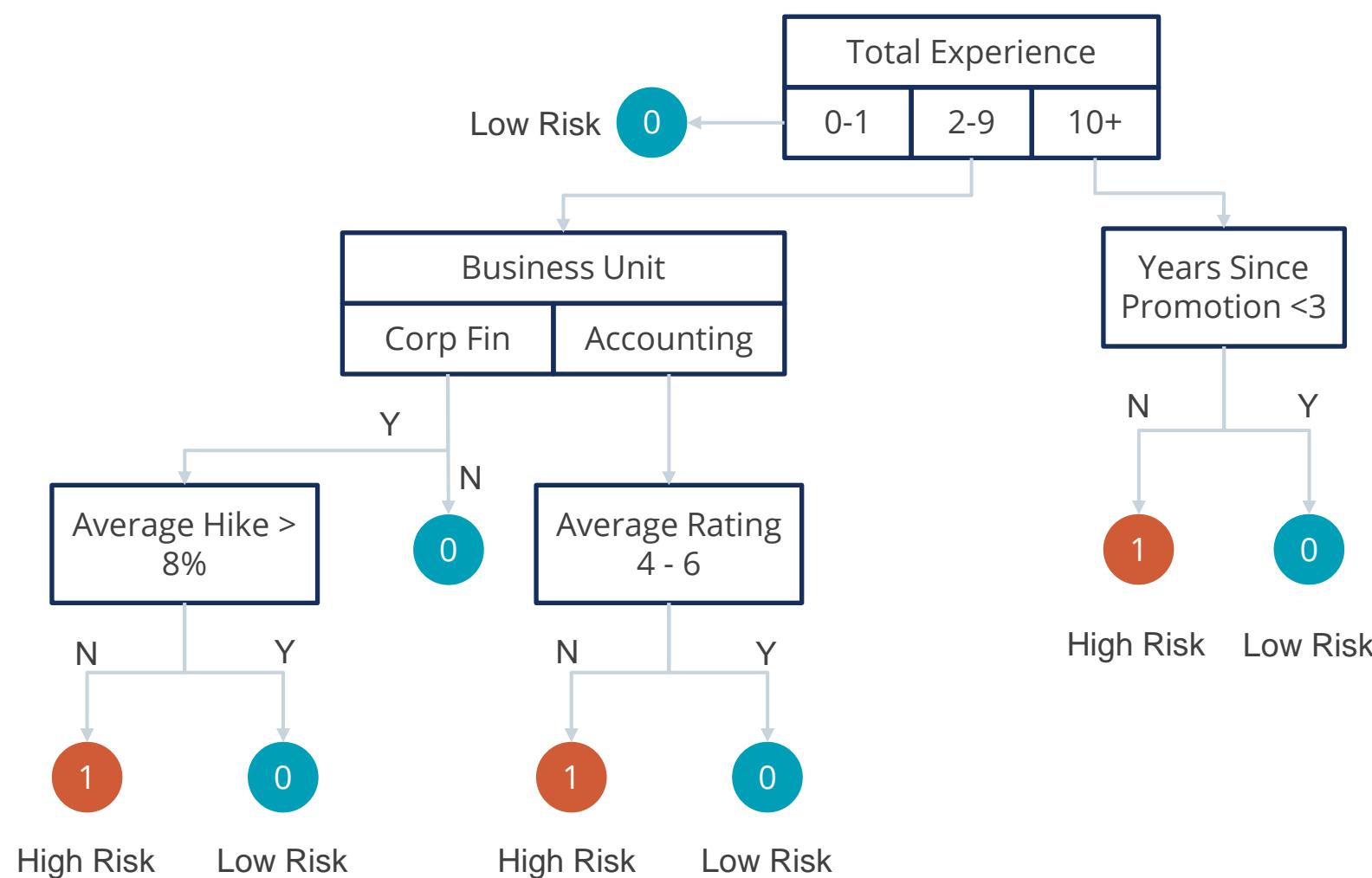


Employee Attrition



Attrition Analysis Using Decision Tree Classifier

Decision tree classification is used to understand the reasons of attrition and identify potential employees who have a high probability of leaving.



Classification

It is a technique that helps to decide the class, label, or category of any scenario. This technique helps to determine the extent to which the scenario belongs to a class.



This is achieved by creating a statistical model from the scenarios with known classes, labels, or categories.

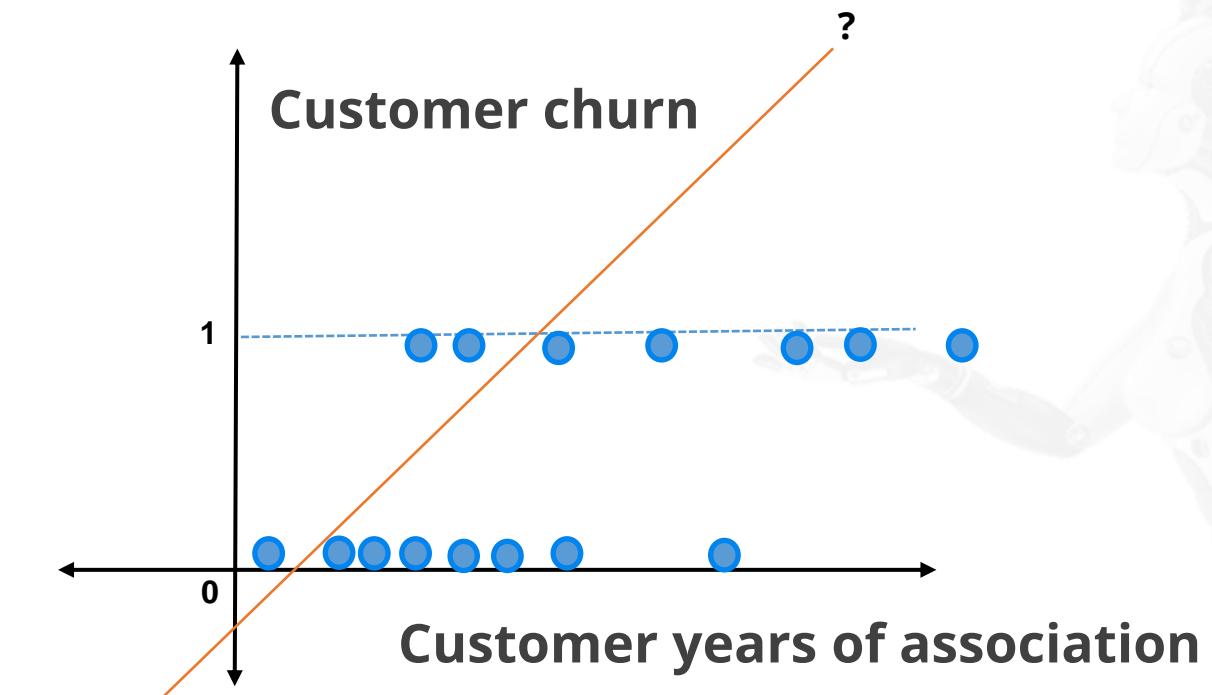
Linear Regression for Classification: Disadvantages

Linear regression does not address the classification problems because:

The response or dependent variable can only take the values as class labels, like $\{0,1\}$ and $\{1,2,3\}$.

A straight line will predict values between negative and positive infinity.

Assumptions of linear regression are not satisfied.



Therefore, it is not a good idea to try and create a direct relationship between the response variable (categorical) and predictors.

Logistic Regression

Logistic Regression

The logistic function relates the predictor variables to the probability of the event occurring.

$$P(y_i=1) = F(\beta_0 + \beta_1 X_i)$$

$$P(y_i=0) = 1 - P(y_i=1) = 1 - F(\beta_0 + \beta_1 X_i)$$

Where $F(\cdot)$ is the logistic function.

$F(\cdot)$ should return values between 0 and 1.

Example: If response variable (y) is Customer Churn, which is binary, $P(y_i=1)$ is the probability of a customer to churn.

Sigmoid Function

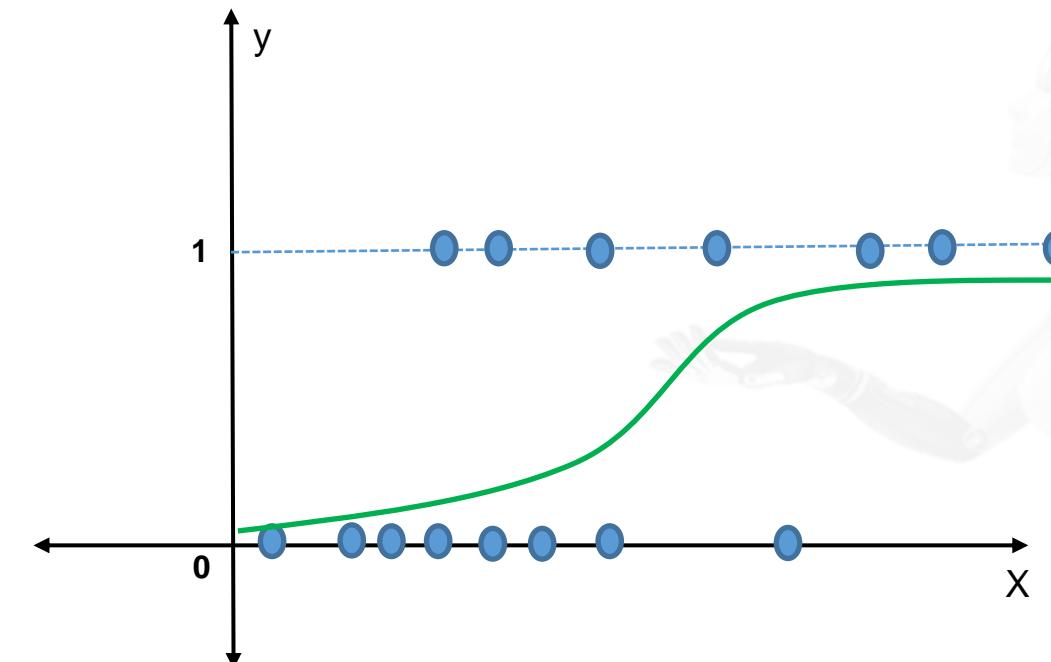
Sigmoid function is defined as $F(z) = e^z/(1+e^z)$

$F(z)$ will always return values between 0 and 1.

$$\text{Prob}(y_i=1) = F(\beta_0 + \beta_1 X_i)$$

$$P(y_i=1) = \frac{e^{\beta_0 + \beta_1 X_i}}{1+e^{\beta_0 + \beta_1 X_i}}$$

$$\ln\left(\frac{P(y_i=1)}{1-P(y_i=1)}\right) = \ln(\text{ODDS}) = \beta_0 + \beta_1 X_i$$



Maximum Likelihood Estimation

To establish a logistic regression model, the values of β_0 and β_1 need to be determined using the maximum likelihood techniques.

This helps to find the parameters that make the observed values most likely to have occurred, maximizing the probability of obtaining the samples at hand.

Probability of observing actual y for any record = $P(y_i) = (P(y_i=1))^{y_i} * (1 - P(y_i=1))^{(1-y_i)}$

We calculate β parameters for maximum value of

$$L(\beta) = \prod_{i=1}^n [(P(y_i=1))^{y_i} * (1 - P(y_i=1))^{(1-y_i)}]$$

Logistic Regression in R

Logistic Regression algorithm can be implemented in R using the `glm()` function.

A description of the error distribution. For Logistic Regression it is specified as “binomial”.

Symbolic description of the model to be fitted as $y \sim x$, where y is the dependent or target variable and x is the independent variable

Syntax:

```
glm(formula, family = gaussian, data)
```

Dataframe or list containing the variables in the model

To include more predictors in the formula for multiple regression, variable names can be separated by $+$, such as $y \sim x_1 + x_2 + x_3$.

Loan Default Prediction



Duration: 10 minutes

Problem Statement: Use the *germancredit.csv* file to build a model using logistic regression to predict loan default probability.

ASSISTED PRACTICE

Note: Please download the data set and solution document from the **Course Resources** section and follow the steps given in the document

K-Nearest Neighbors (KNN)

K-Nearest Neighbors (KNN)

KNN algorithm is a parametric technique that classifies new data points based on similarity that can be measured using the Euclidean distance.

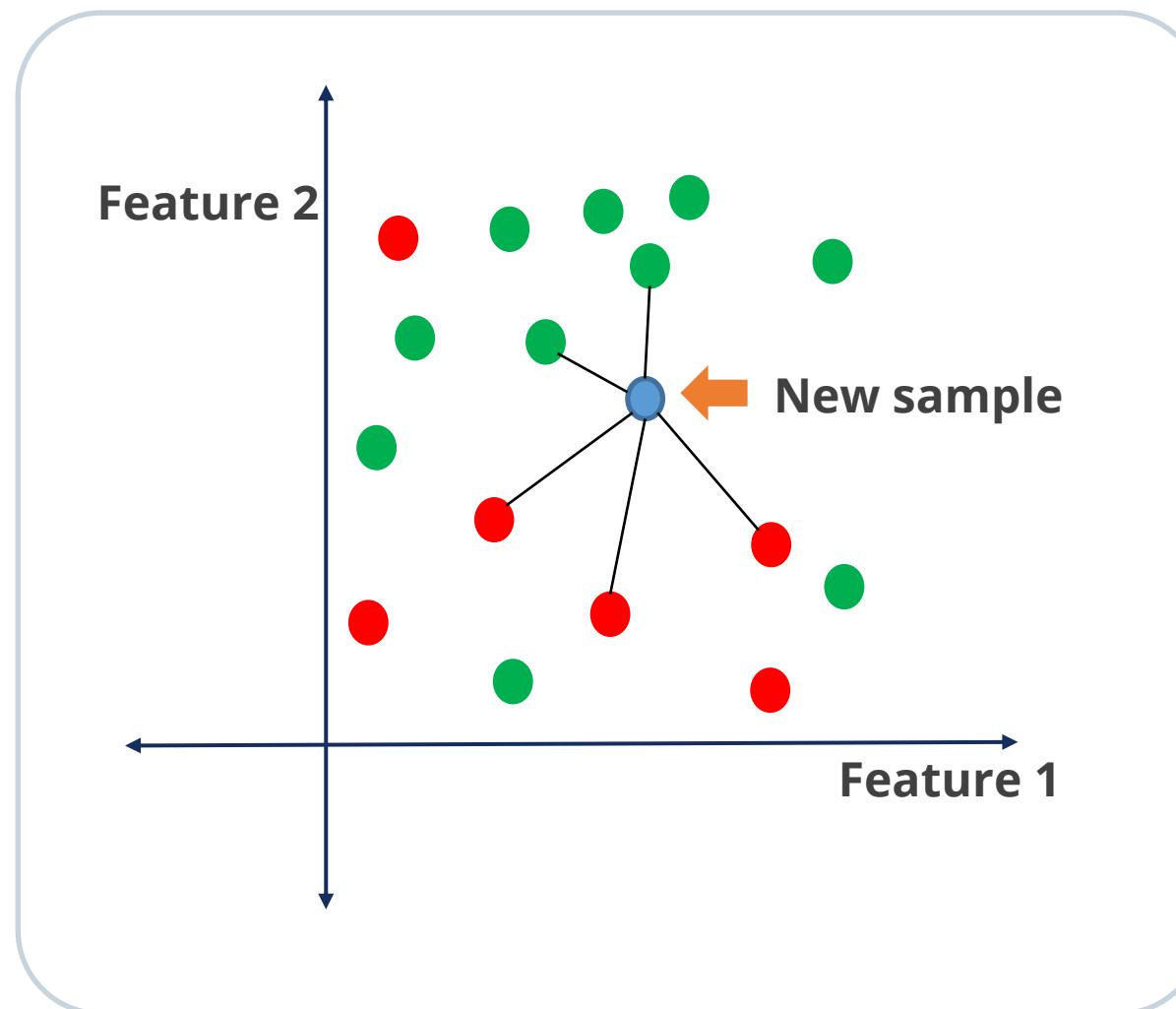


KNN algorithm can be used for regression as well as classification, but generally, it is used to solve classification problems.

Euclidean distance between 2 points $A(x_1, y_1)$ and $B(x_2, y_2)$

$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

KNN Methodology



Step 1: Select the value of k (number of neighbors)

Step 2: Take k neighbors based on the distance measurement

Step 3: Count the number of samples of each class in the neighborhood

Step 4: Assign the majority class to the new sample

Deciding the Value of K

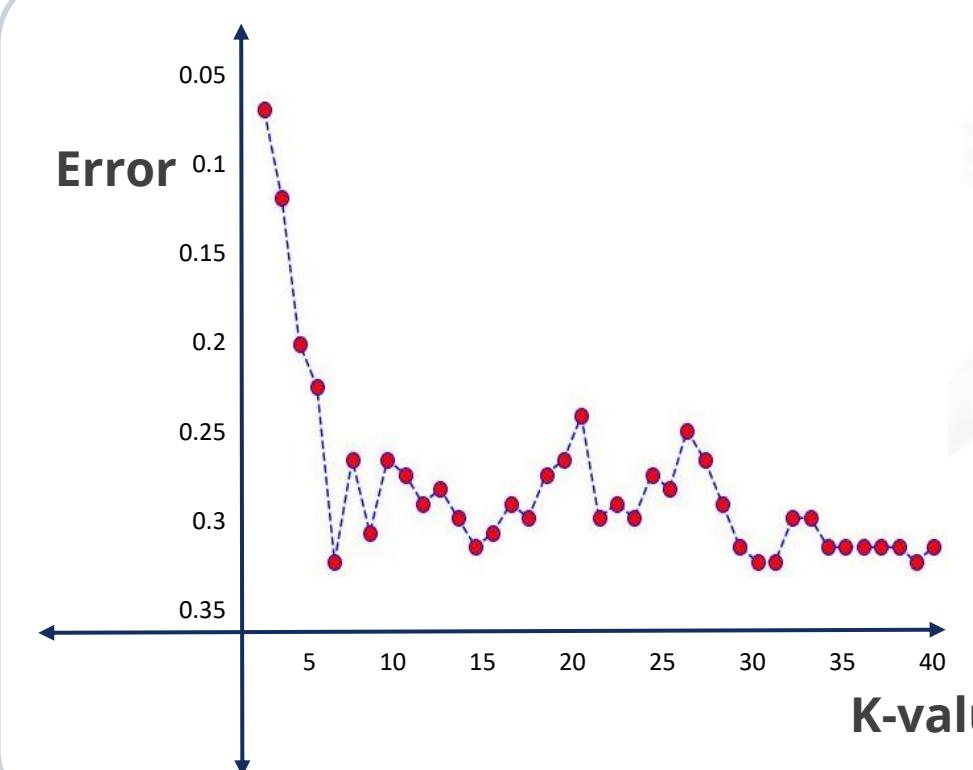
Value of K is decided based on performance metrics like error or accuracy.



K should not be taken as a small value to ensure generalization.



K should not be taken as a big value. This will create problems, especially in cases of unbalanced classes.



KNN in R

KNN algorithm can be implemented in R using `knn()` function from the `class` package.

Development and validation set

K values or number of neighbours

Syntax:

```
knn(train, test, cl, k = 1, l = 0, prob = FALSE, use.all = TRUE)
```

The vector of class labels from the
train data

The functions return a vector of predicted classes for the test data.

Cancer Prediction



Duration: 10 minutes

Problem Statement: Consider the **cancer.csv** data and use KNN to build and classify the data to identify benign and malignant cancers.

Note: Please download the data set and the solution document from the **Course Resources** section and follow the steps given in the document

ASSISTED PRACTICE

Decision Tree

Identifying Possible Loan Defaulter

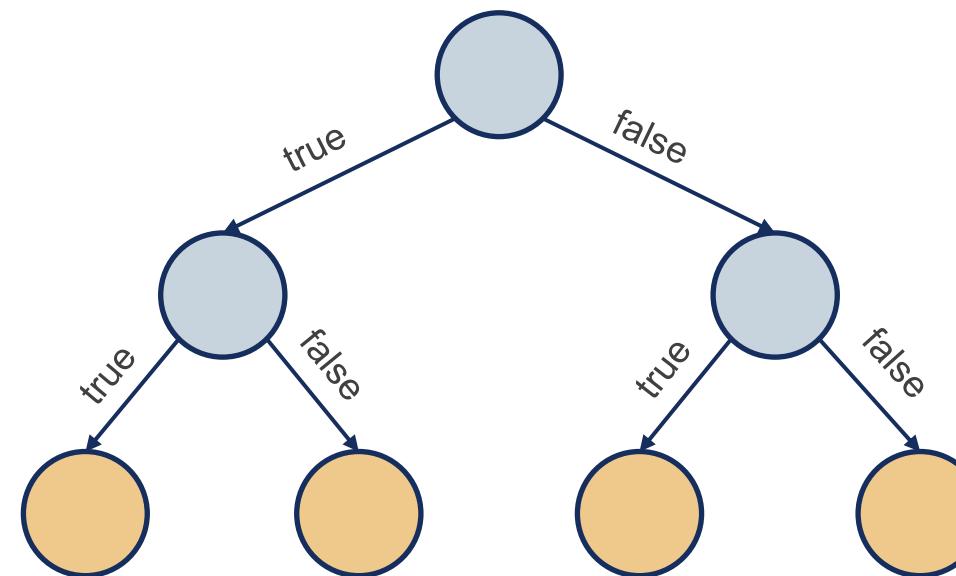
A loan applicant has applied for a loan of \$40K. They are earning \$1250 per month. If their credit score is 720, can they be a possible loan defaulter?



Identifying Possible Loan Defaulter: Solution

Step 1: Using customer data, create a classification model like a decision tree.

Decision tree is used to display an algorithm that contains conditional control statements. It is a representation of patterns identified from observations (samples) in the form of a tree.



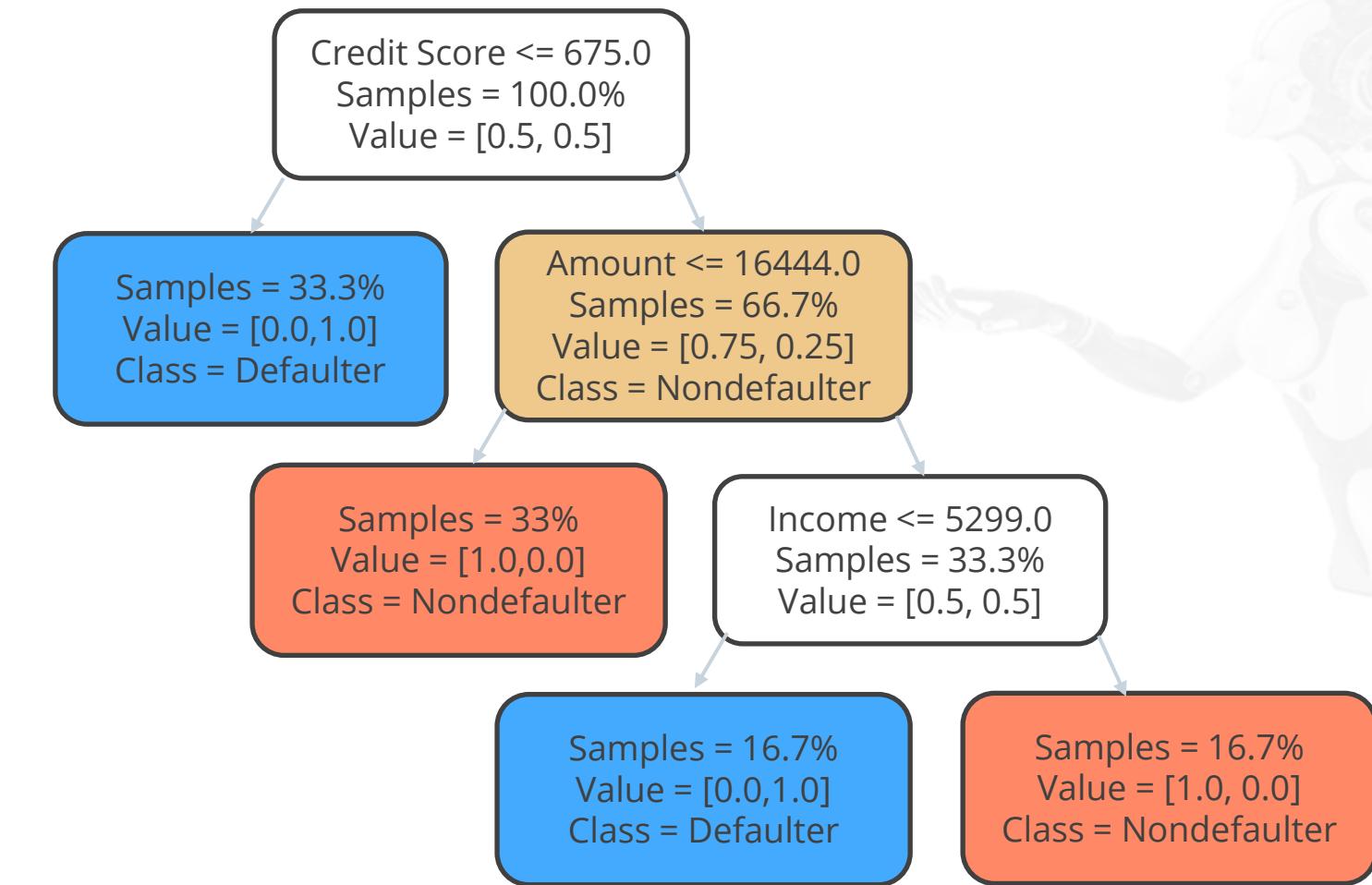
Identifying Possible Loan Defaulter

Step 1: Using customer data, create a classification model like a decision tree

Customer Data of a Bank

ID	Amount	Credit Score	Income
1	\$14000	750	\$1000
2	\$2000	720	\$1200
3	\$2000	730	\$1250
4	\$12000	700	\$5500
5	\$70000	570	\$5000
6	\$60000	850	\$12000
7	\$55000	600	\$1190
8	\$30000	750	\$5398
9	\$30000	620	\$15000
10	\$18888	750	\$5200
11	\$18000	650	\$1380
12	\$25000	700	\$1280

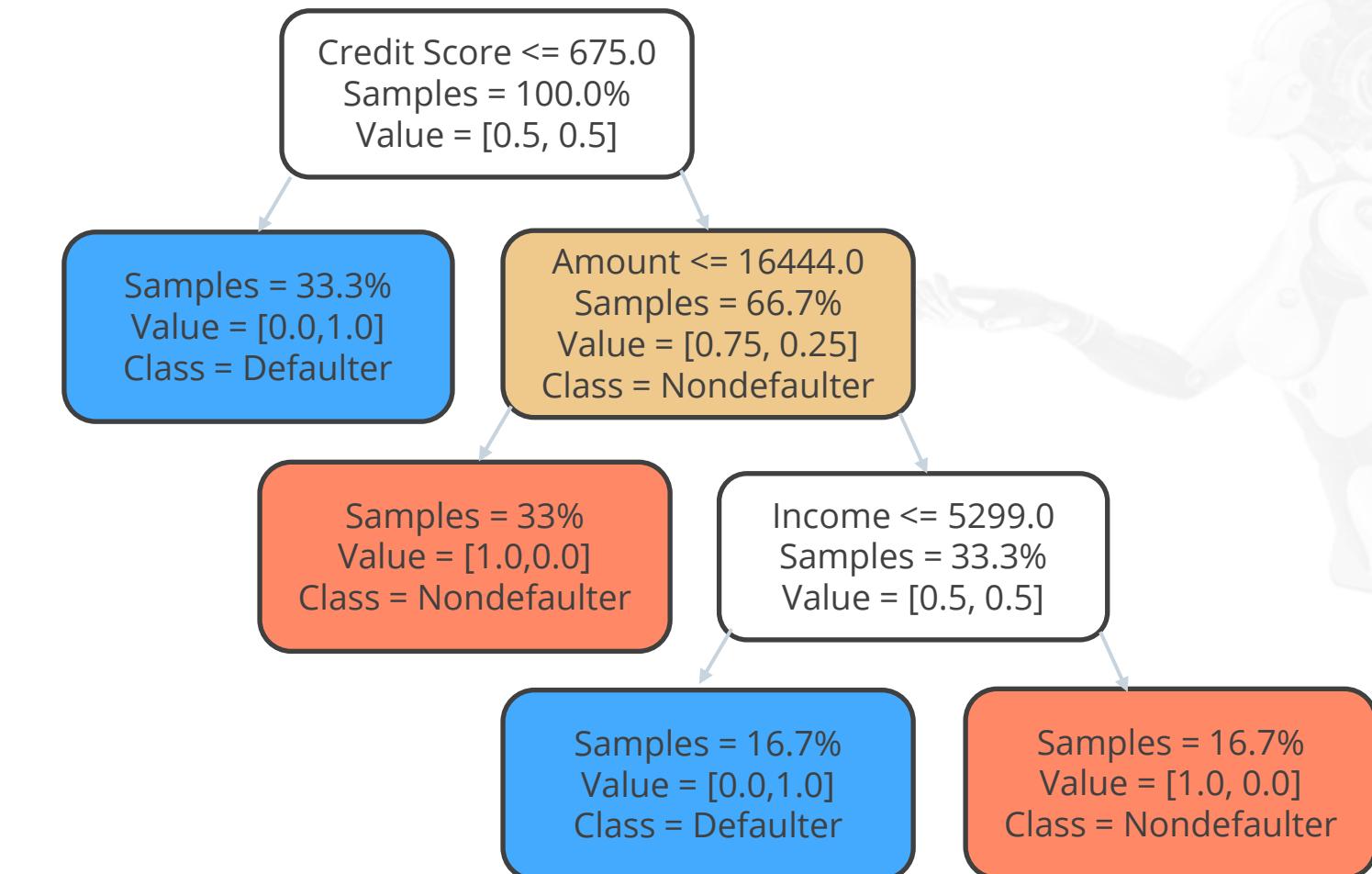
Decision Tree for the data



Identifying Possible Loan Defaulter

Step 2: Check for the relevant pattern or rule to decide the probable class

From the customer data and using the decision tree, you can derive that if the amount is greater than 16444 and income is greater than 5299, then it is highly probable that the customer may turn out to be a defaulter.



Popular Techniques

Popular techniques for developing a Decision Tree Classifier are:

Chi-Square
Automatic
Interaction
Detector
(CHAID)

Classification and
Regression Trees
(CART)

C4.5

Splitting Criterion

CART uses Gini index as a measure of impurity.

$$\text{Gini at a node} = \text{GINI} = 1 - \sum p_j^2$$

$$\begin{aligned} N &= 15 \\ C_1 &= 10 \\ C_2 &= 5 \end{aligned}$$

$$\text{Gini} = 1 - ((10/15)^2 + (5/15)^2) = 0.44$$

Higher the purity, lower the Gini.

$$\begin{aligned} N &= 15 \\ C_1 &= 15 \\ C_2 &= 0 \end{aligned}$$

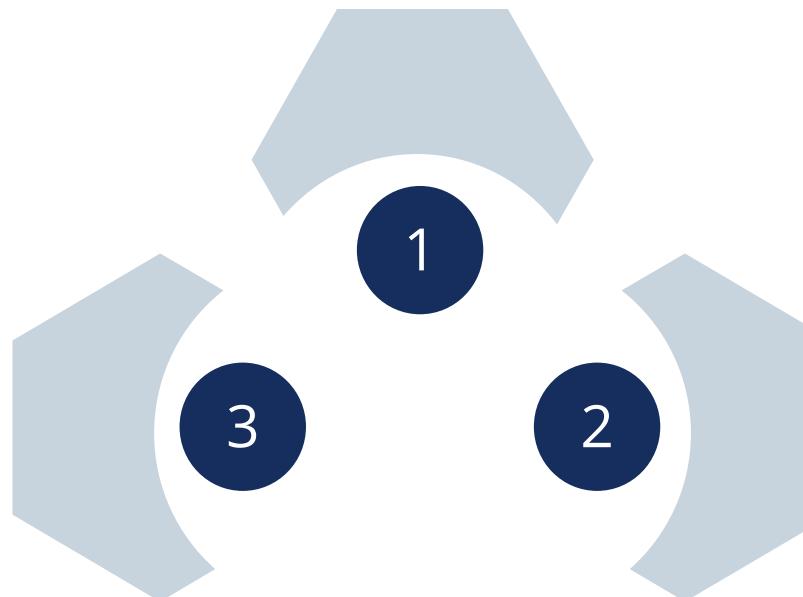
$$\text{Gini} = 1 - ((15/15)^2 + (0/15)^2) = 0$$

Here, C_1 is class 1, C_2 is class 2, and N is the number of samples at the node.

Advantages and Disadvantage

Resistant to outliers

Intuitive and easy to explain



Easy handling of both categorical and numerical data



Decision trees are prone to overfitting.

Decision Trees in R

CART model can be implemented in R using the rpart() function of the rpart package.

Symbolic description of the model to be fitted as $y \sim x$, where y is the dependent or target variable and x is the independent variable

Dataframe containing the variables specified in formula

```
rpart(formula, data, control )
```

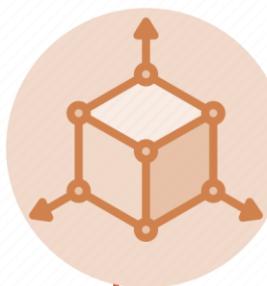
List of options that control details of the rpart algorithm

Random Forest

Bootstrap Aggregation

Bootstrap aggregation or bagging is a class of ensemble learning where the noisy but approximately unbiased models are averaged, which reduces the variance.

Trees are ideal candidates for bagging as they can capture complex interactions.

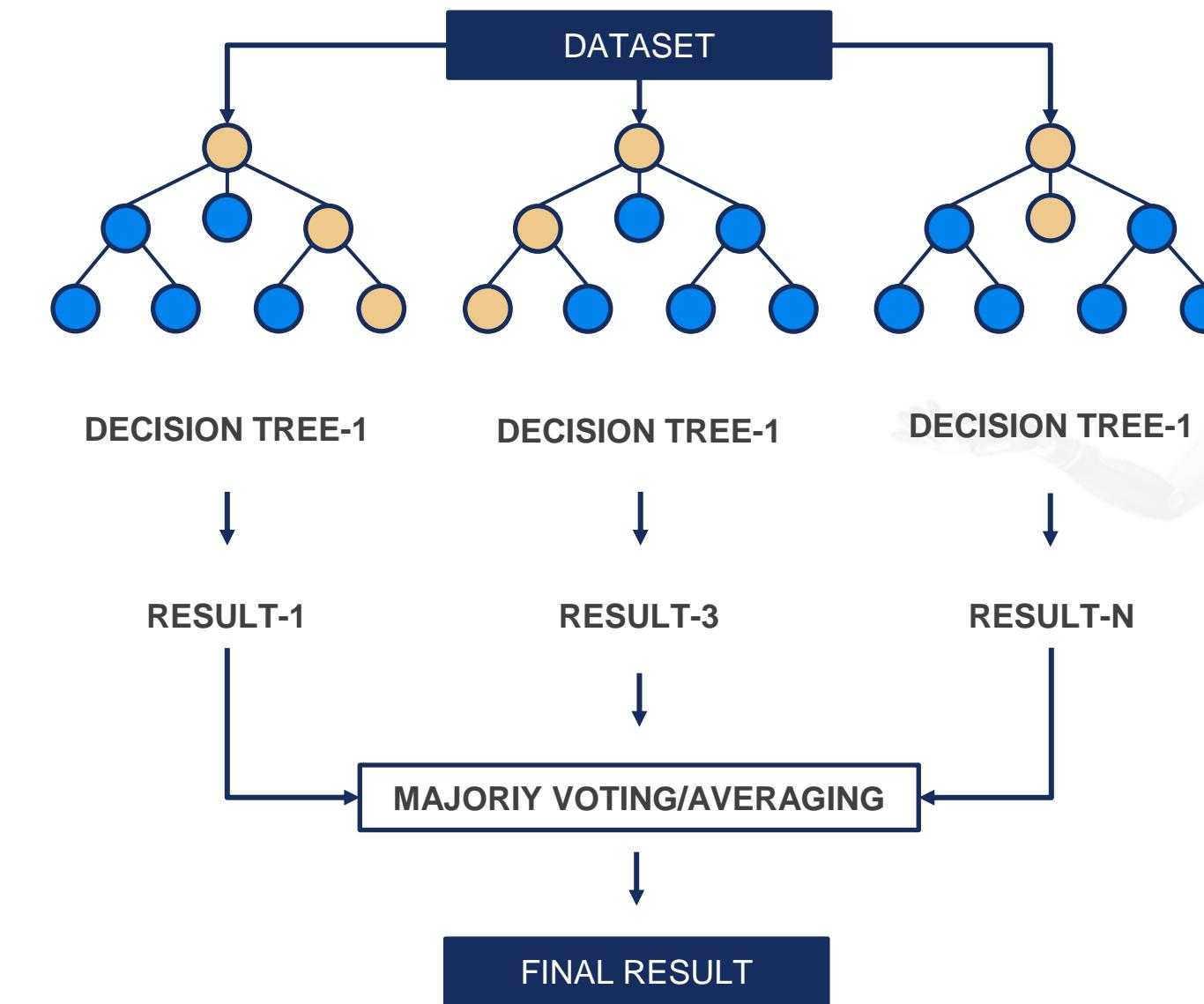


Ensemble refers to the use of multiple models to achieve better performance than could be expected by any of the constituent models.

Random Forest

Random forest is a bagging technique that combines multiple decision trees for building models.

It helps to overcome the issue of overfitting. This helps to achieve great accuracy in the performance of the model.



Steps in Random Forest

Step 1: N bootstrapped samples are created. Here, random sampling with replacement takes place.

Customer Data

ID	Amount	Credit Score	Income	Defaulter
1	\$14000	750	\$1000	0
2	\$2000	720	\$1200	0
3	\$2000	730	\$1250	0
4	\$12000	700	\$5500	0
5	\$70000	570	\$5000	1
6	\$60000	850	\$12000	0
7	\$55000	600	\$1190	1
8	\$30000	750	\$5398	0
9	\$30000	620	\$15000	1
10	\$18888	750	\$5200	1
11	\$18000	650	\$1380	1
12	\$25000	700	\$1280	1

n = 12

Bootstrapped Data: Sample 1

ID	Amount	Credit Score	Income	Defaulter
1	\$14000	750	\$1000	0
2	\$2000	720	\$1200	0
3	\$2000	730	\$1250	0
4	\$2000	720	\$1200	0
5	\$70000	570	\$5000	1
6	\$60000	850	\$12000	0
7	\$55000	600	\$1190	1
8	\$30000	750	\$5398	0
9	\$30000	620	\$15000	1
10	\$30000	750	\$5398	0
11	\$18000	650	\$1380	1
12	\$60000	850	\$12000	0

n = 12

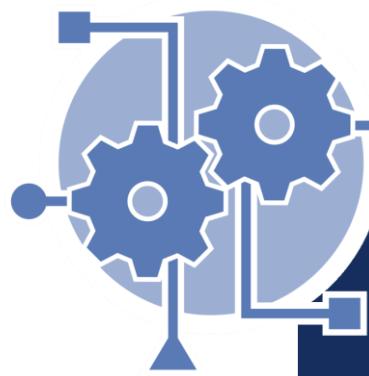
Steps in Random Forest

Step 2: Create N decision trees for N bootstrapped samples. In a decision tree, only a random subset of m variables out of total of p variables are used at each node to decide the splitting of each node.

Step 3: Each record is classified based on N decision trees. The final class of each record is decided based on the majority vote.

OOB Samples

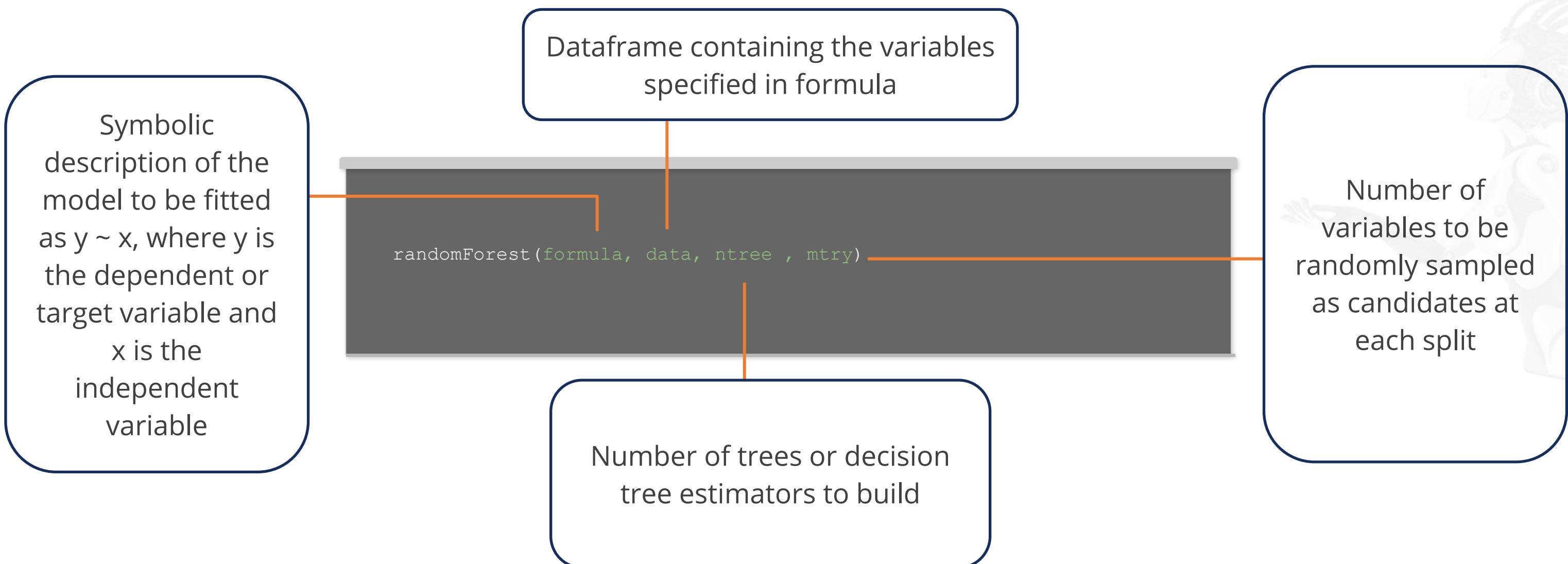
Records left out in the k^{th} bootstrapped sample (bag) are Out of Bag samples.



For each observation z_i , construct its Random Forest predictor by averaging only those trees corresponding to bootstrap samples in which z_i did not appear.

Random Forest in R

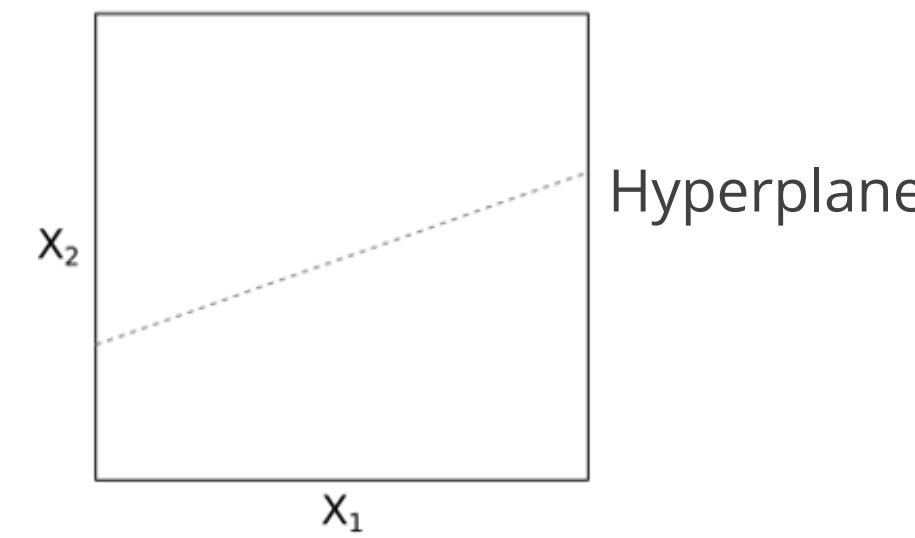
Random forest model can be implemented in R using randomForest() function of the randomForest package.



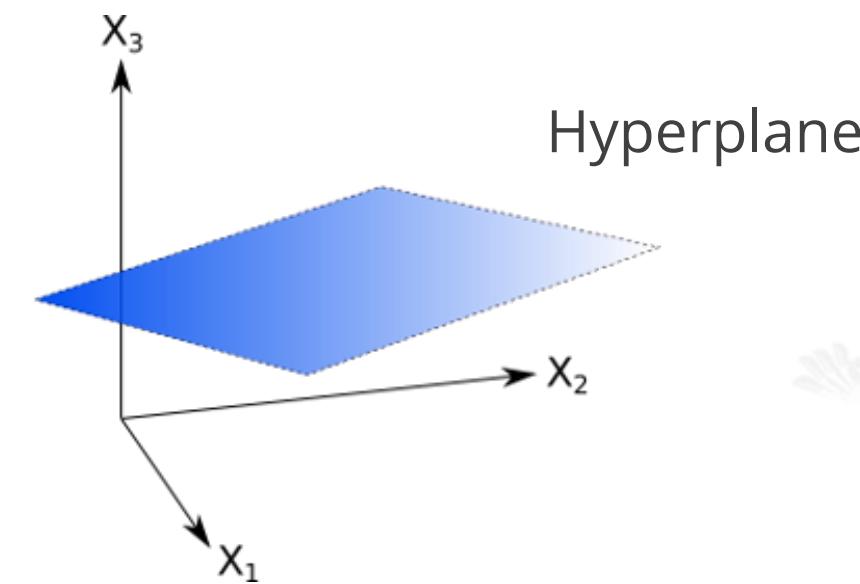
Support Vector Machine

Hyperplane

In an n-dimensional space, a hyperplane is a flat subspace of dimension $n - 1$.



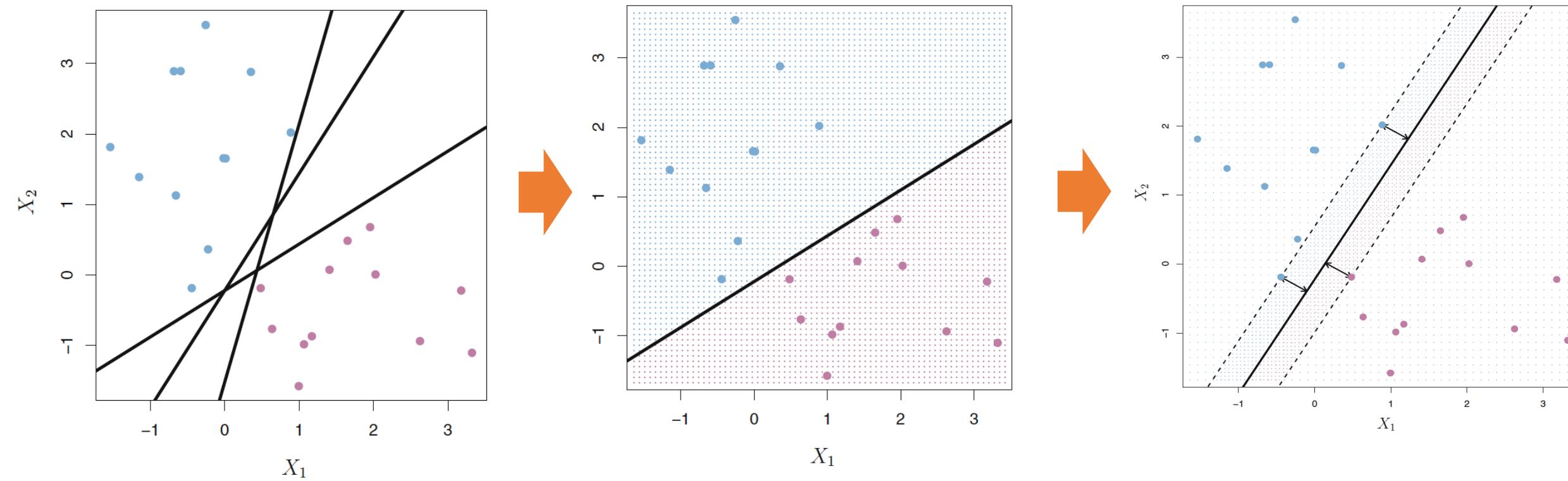
In two dimensions, a hyperplane is a line.



In three dimensions, a hyperplane is a flat plane.

Maximal Margin Hyperplane

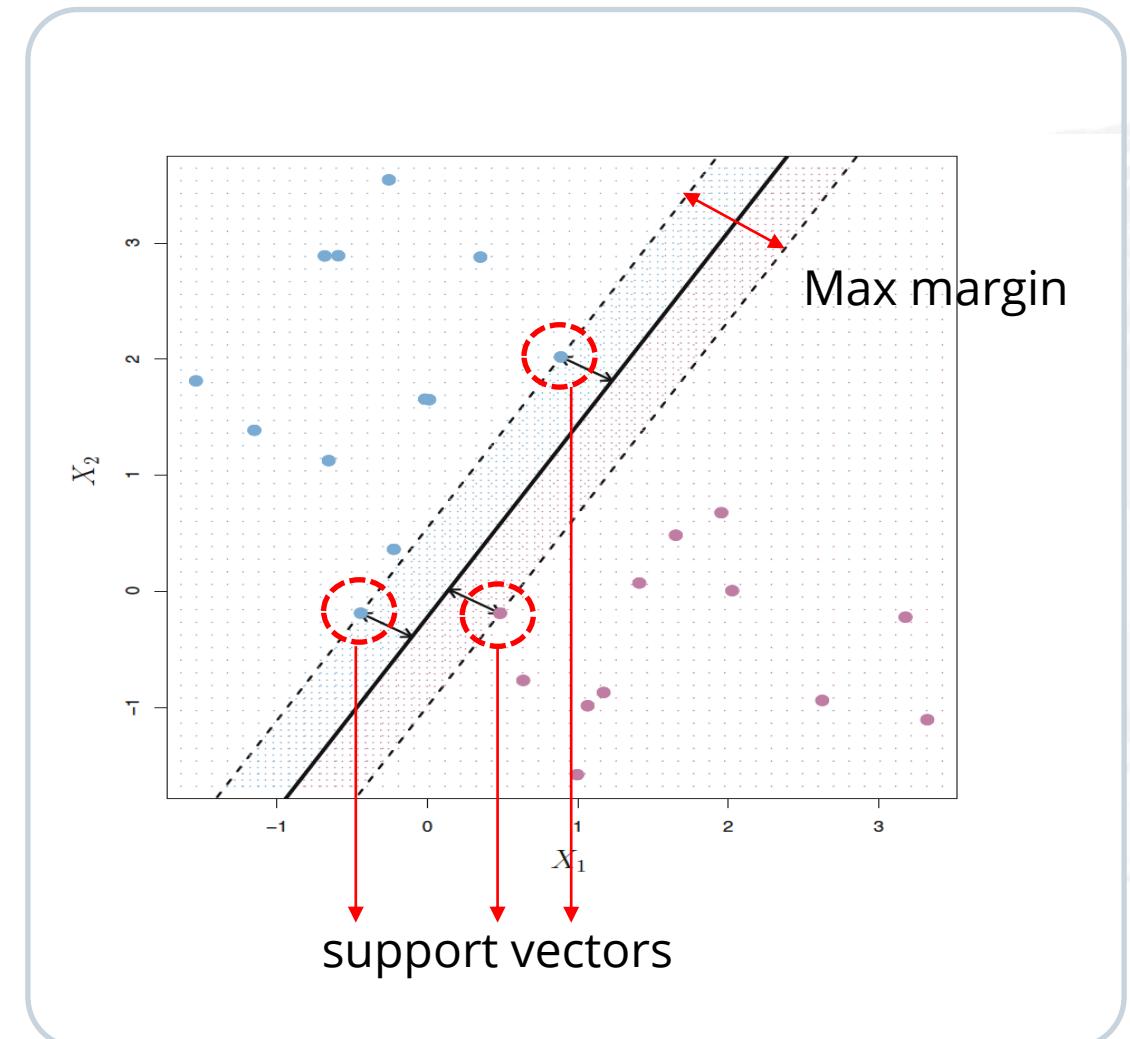
The quest of finding the best separating hyperplane:



Separating a hyperplane that is farthest from the training observations seems a natural choice. Once the best separating hyperplane is obtained, classification can be done for any new sample depending on which side of the hyperplane the new sample falls on.

Maximum Margin Classifier

Classifier	Description
Margin	Smallest distance of training examples from the hyperplane is known as the margin.
Maximal margin hyperplane	The maximal margin hyperplane is the separating hyperplane for which the margin is largest. In other words, it has the farthest minimum distance to the training observations.
Support vectors	Support vectors in n-dimensional space support the maximal margin hyperplane. If these points are moved, then the maximal margin hyperplane would move as well.



Support Vector Machines

In the Support Vector Machine algorithm, a hyperplane in an n-dimensional space is identified which distinctly classifies the data points.



The Support Vector Machine is a generalization of Maximal Margin Classifier.

SVM: Setting Up the Discrimination Boundaries

A hyperplane can be defined by $\vec{\beta} \cdot \vec{X} + \beta_0 = 0$, where $\vec{\beta}$ is perpendicular (normal) to the hyperplane.

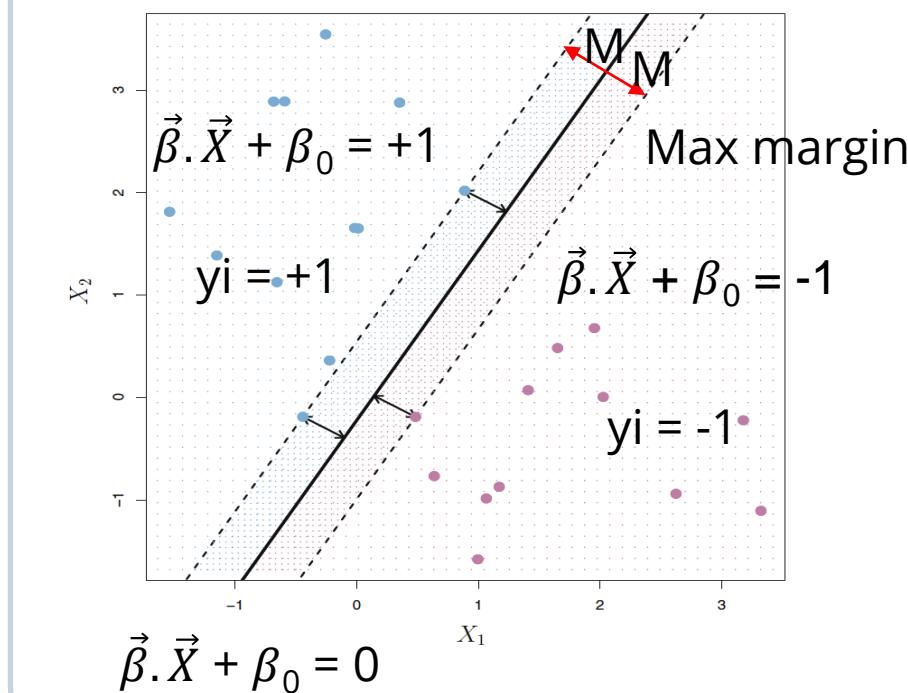
The classification rule would be:

$\vec{\beta} \cdot \vec{X} + \beta_0 \geq +1$ for known positive samples

$\vec{\beta} \cdot \vec{X} + \beta_0 \leq -1$ for known negative samples

For mathematical simplicity: $y_i (\vec{\beta} \cdot \vec{X} + \beta_0) \geq +1$

For the points falling on broken lines on either side of the hyperplane: $y_i (\vec{\beta} \cdot \vec{X} + \beta_0) - 1 = 0$



SVM: Maximizing the Margins

Margin $2M$ can be calculated as:

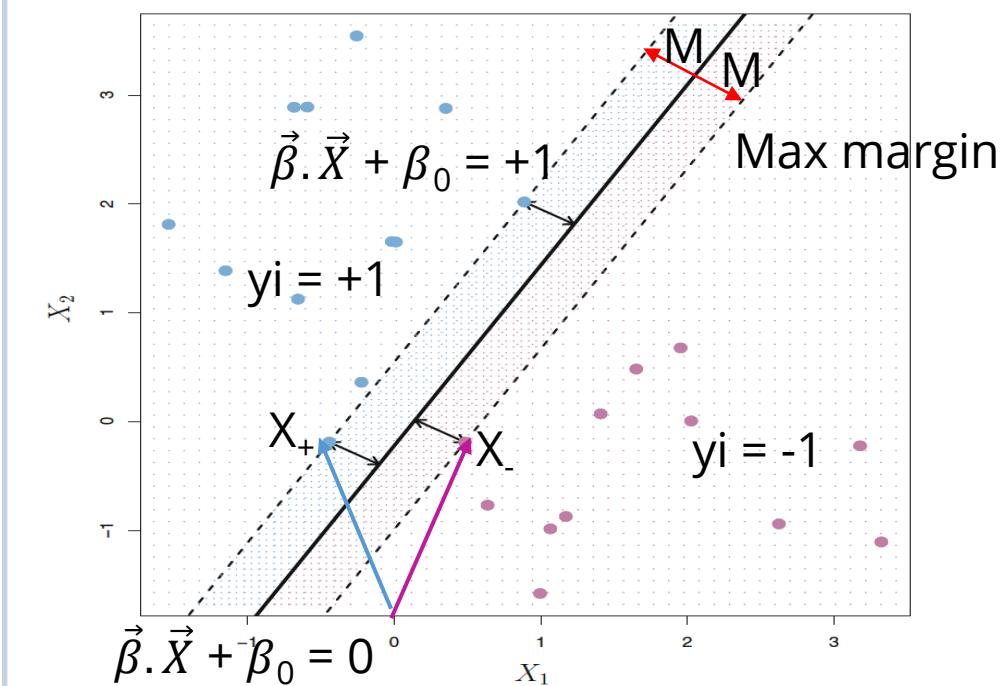
Since $\vec{\beta}$ is normal to the hyperplane, the width of the slab or margin can be calculated as:

$$(X_+ - X_-) \cdot \vec{\beta} / |\vec{\beta}| = 1 / |\vec{\beta}| [(1 - \beta_0) - (-1 - \beta_0)] = 2 / |\vec{\beta}|$$

In order to find the best $\vec{\beta}$ and β_0 ,

$$\text{Min } [2 / |\vec{\beta}|] \text{ subject to } y_i (\vec{\beta} \cdot \vec{X} + \beta_0) \geq +1$$

Separable classes



SVM in R

In R, SVM model can be implemented using the `svm()` function of the `e1071` package.

Symbolic description of the model to be fitted as $y \sim x$, where y is the dependent or target variable and x is the independent variable

```
svm(formula, data )
```

Dataframe containing the variables specified in the formula

Naïve Bayes Classification

Bayes' Theorem

Bayes' Theorem describes the probability of an event based on prior knowledge of conditions that might be related to the event.

$$P(Class_j | x) = \frac{P(x|Class_j) * P(Class_j)}{\sum_{j=1}^c P(x|Class_j) * P(Class_j)}$$

$P(Class_j | x)$ - Posterior probability of class

$P(x | Class_j)$ - Likelihood probability of predictor given class

$P(Class_j)$ - Class prior probability

$P(x)$ - Predictor prior probability

Bayes' Theorem: Simplified

Bayes' theorem can be simplified using probability theory:

$$P(\text{Class}_j | x) = \frac{P(x|\text{Class}_j) * P(\text{Class}_j)}{\sum_{j=1}^c P(x|\text{Class}_j) * P(\text{Class}_j)}$$

where c is the number of classes

Using the assumption of independence, the numerator for predictors (x_1, x_2 , etc.) can be expanded:

$$P(x|\text{Class}_j) * P(\text{Class}_j) = [P(x_1|\text{Class}_j) * P(x_2|\text{Class}_j) * P(x_3|\text{Class}_j) * P(x_4|\text{Class}_j) ...] * P(\text{Class}_j)$$

Note

A denominator is effectively a constant for a given data. In practice, the numerator is compared.

Bayes' Theorem: Example

Given the previous patients a doctor has seen, should he believe that the patient with the following symptoms has flu?

Chills	Runny nose	Headache	Fever	Flu
Y	N	Mild	Y	N
Y	Y	No	N	Y
Y	N	Strong	Y	Y
N	Y	Mild	Y	Y
N	N	No	N	N
N	Y	Strong	Y	Y
N	Y	Strong	N	N
Y	Y	Mild	Y	Y

Chills	Runny nose	Headache	Fever	Flu
Y	N	Mild	Y	?

Find:

$$P(\text{Class}_1 | x) = P(\text{flu=Y} | x) \text{ and}$$

$$P(\text{Class}_2 | x) = P(\text{flu=N} | x)$$

to solve this problem.

Calculations

Calculate likelihood of x in the given class:

$$(x|Class_j) = P(x_1|Class_j) * P(x_2|Class_j) * P(x_3|Class_j) * P(x_4|Class_j)$$

$$P(x | \text{flu=Y}) =$$

$$P(\text{chills} = Y | \text{flu=Y}) * P(\text{runny nose} = N | \text{flu=Y}) * P(\text{headache} = \text{mild} | \text{flu=Y}) * \\ P(\text{fever} = Y | \text{flu=Y})$$

$$= \frac{3}{5} * \frac{1}{5} * \frac{2}{5} * \frac{4}{5} \Rightarrow P(x | \text{flu=Y}) = .0384$$

Similarly $\Rightarrow P(x | \text{flu=N}) = .0246$

Calculations

Using the data, these can be found:

$$P(\text{flu}=Y) = 5/8 = 0.625 \text{ and } P(\text{flu}=N) = 3/8 = 0.38$$

Hence,

- Numerator of $P(\text{flu}=Y | x) = .0384 * .625 = .024$
- Numerator of $P(\text{flu}=N | x) = .024 * .38 = .009$ using Bayes' theorem
- Since denominator remains constant $P(\text{flu}=Y | x) > P(\text{flu}=N | x)$, the doctor should believe that the patient with the given symptoms has the flu.

Naïve Bayes in R

Naïve Bayes can be implemented in R using the `naiveBayes()` function of the `e1071` package.

Symbolic description of the model to be fitted as $y \sim x$, where y is the dependent or target variable and x is the independent variable

```
naiveBayes(formula, data)
```

Dataframe containing the variables specified in the formula

Model Evaluation

Evaluating Classification Models

A confusion matrix is used to evaluate the performance of a classification model on a set of test datapoints for which the actual values are known.

ID	Actual class	Predicted class	Result
1	0	0	TN
2	0	0	TN
3	1	0	FN
4	0	0	TN
5	1	1	TP
6	1	0	FN
7	0	1	FP
8	1	0	FN
9	1	1	TP
10	1	1	TP

1 is positive class

		Actual Values	
		Positive	Negative
Predicted Values	Positive	True Positive	False Positive
	Negative	False Negative	True Negative

Important Metrics

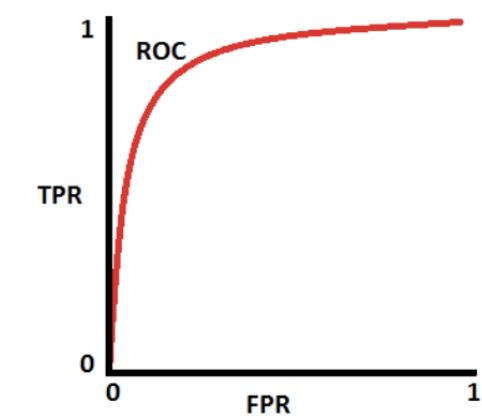
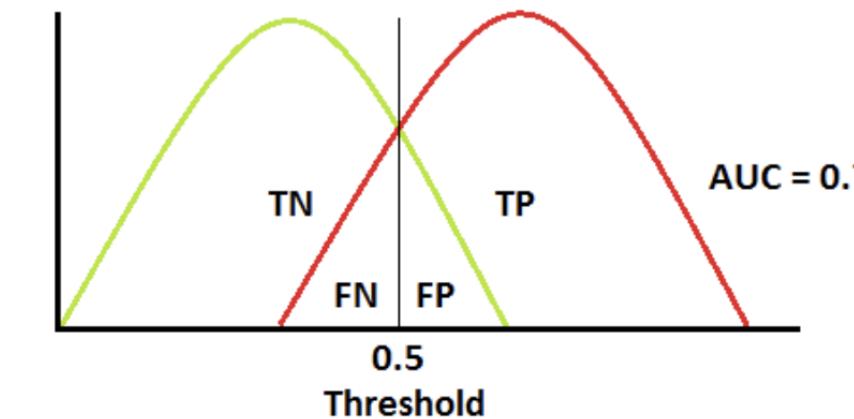
Accuracy = $\frac{(TP+TN)}{(P+N)}$	Positive (P)	Negative (N)	
Positive (PP) (Predicted Positive)	True Positive	False Positive	Precision = $\frac{TP}{PP}$
Negative (PN) (Predicted Negative)	False Negative	True Negative	False omission rate $= \frac{FN}{PN}$
	Sensitivity/rec all $= \frac{TP}{P}$	False positive rate $= \frac{FP}{N}$	

Receiver Operating Characteristic (ROC) Curve

ROC curve is a performance measurement metric for classification problems.

ROC curve is obtained by plotting a graph between TPR and FPR at different thresholds for predicted probabilities.

AUC is the area under ROC curve that helps to measure the distinguishing power of the model. The greater the AUC, the better the model is at predicting 0 classes as 0 and 1 classes as 1.



Key Takeaways

- Classification is a technique to determine the extent to which a data sample will be a part of a category or type.
- The classification process uses two techniques for prediction: model construction and model usage.
- Different classification techniques include logistic regression, support vector machine, K-nearest neighbors, Naive Bayes classifier, decision tree, and random forest classification.
- Bias and variance are the two types of major errors in a predictive model.
- Validation methods such as K-fold cross-validation can be used to decrease overfitting in a model.





Knowledge Check

Knowledge Check

1

Which one of the following statements is true for a Decision Tree?

- A. Decision tree is only suitable for the classification problem statement.
- B. In a decision tree, the Gini of a node decreases as we go down the decision tree.
- C. In a decision tree, Gini determines purity.
- D. Decision tree can be used only for numeric values and continuous attributes.

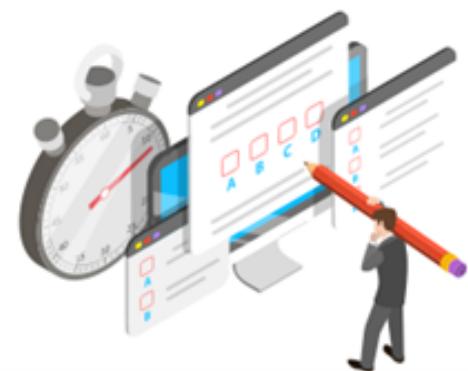


**Knowledge
Check**

1

Which one of the following statements is true for a Decision Tree?

- A. Decision tree is only suitable for the classification problem statement.
- B. In a decision tree, the Gini of a node decreases as we go down the decision tree.
- C. In a decision tree, Gini determines purity.
- D. Decision tree can be used only for numeric values and continuous attributes.



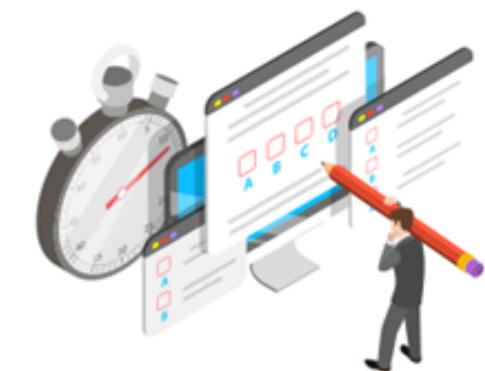
The correct answer is **B**

The Gini of a node decreases as we go down the decision tree. Gini determines the impurity of a node.

**Knowledge
Check**
2

Which of the following is a nonprobabilistic classification model?

- A. Logistic regression
- B. K-nearest neighbors
- C. Naïve Bayes algorithm
- D. Decision tree classifier



**Knowledge
Check
2**

Which of the following is a nonprobabilistic classification model?

- A. Logistic regression
- B. K-nearest neighbors
- C. Naïve Bayes algorithm
- D. Decision tree classifier



The correct answer is **B**

K-nearest neighbors is a nonprobabilistic model. The other three predict the class probabilities.

**Knowledge
Check**

3

ROC curve is plotted between _____ (Select all that apply)

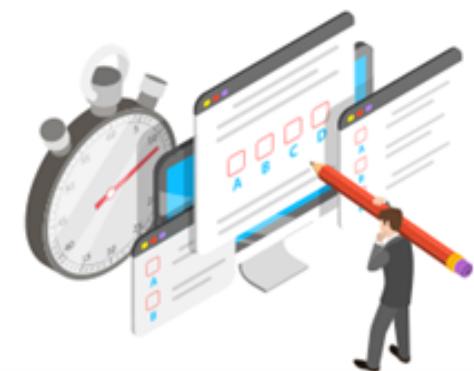
- A. Sensitivity and specificity
- B. True positive rate and false positive rate
- C. True positive rate and false negative rate
- D. Sensitivity and 1- specificity



**Knowledge
Check
3**

ROC curve is plotted between _____ (Select all that apply).

- A. Sensitivity and specificity
- B. True positive rate and false positive rate
- C. True positive rate and false negative rate
- D. Sensitivity and 1- specificity



The correct answer is **B and D**

ROC curve is plotted between true positive rate, that is, sensitivity and false positive rate, that is, (1 - specificity).