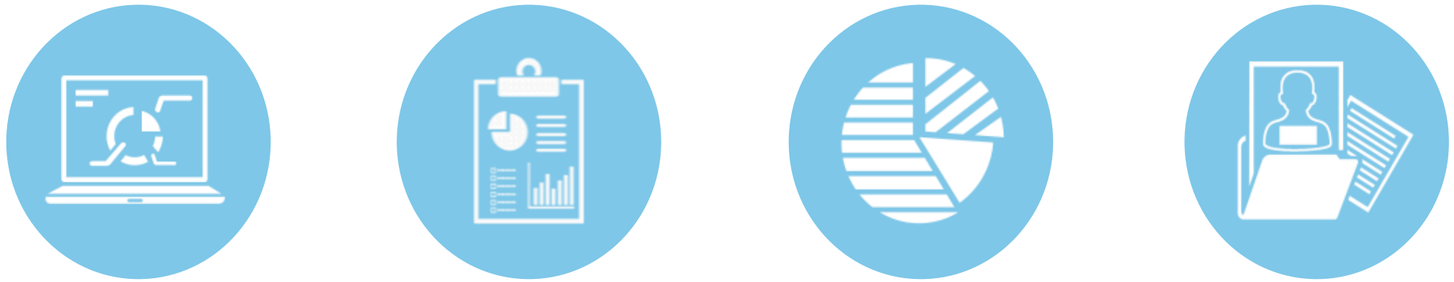
Data Science with R

Lesson 8— Classification



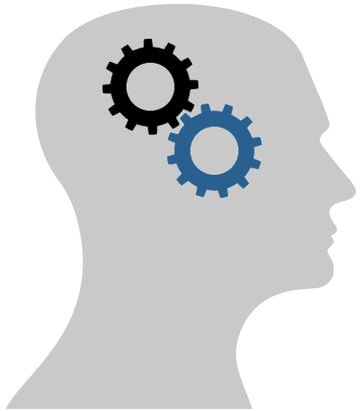
Learning Objectives

- Discuss classification and the types of classification algorithms
- Describe logistic regression
- Explain support vector machines
- Discuss K-Nearest Neighbors (KNN)
- Explain Naive Bayes classifier
- Describe decision tree and random forest classification
- Examine how to evaluate the classifier models

Classification

Topic 1— Classification and Its Types

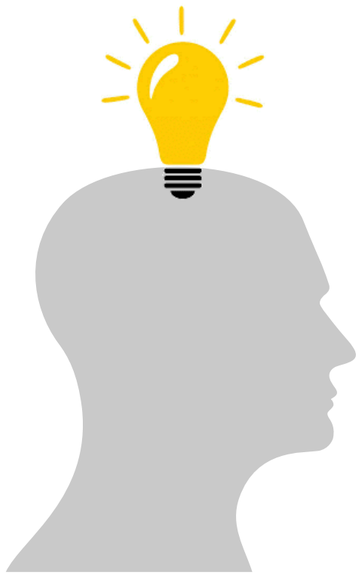
Bank Loans



A bank has to analyze if loans should be granted to all their customers. There are a lot of variables like age, employment status, income, etc. to be considered.

How will the bank arrive at a process?

Bank Loans



With the help of **decision tree classification algorithm**, the bank decides if a customer should be granted a loan or not.

What Is Classification?

It is a technique to determine the extent to which a data sample will or will not be a part of a category or type. Classification models predict categorical class labels.

Classification Process

The classification process includes the following techniques for prediction:

Model Construction

- It is done to describe a set of predetermined classes.
- Every sample belongs to a predefined class.
- The model is represented as decision trees, classification rules, or mathematical formulae.

Model Usage

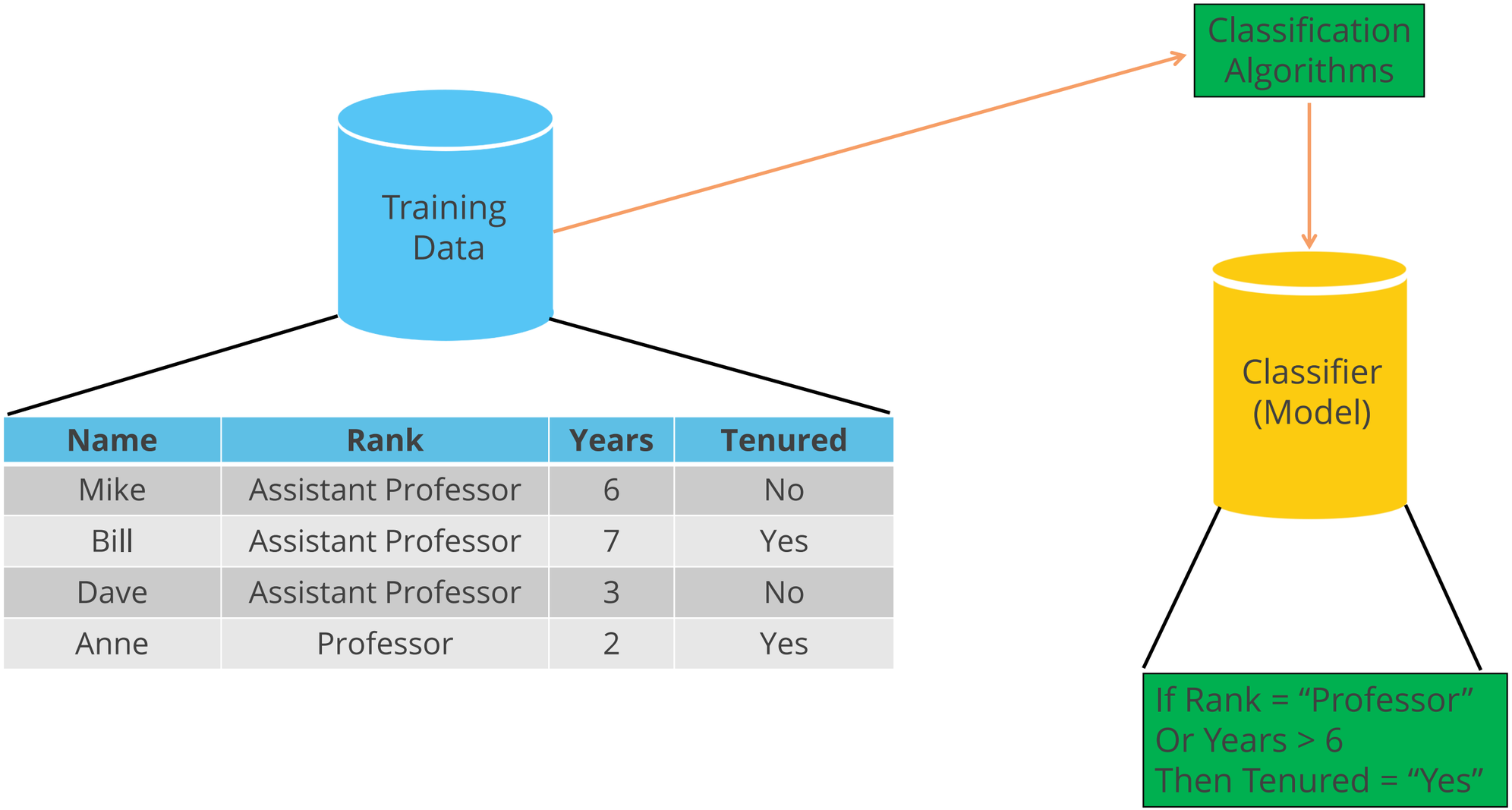
- It is done to classify unknown or future objects and to estimate the accuracy of a model.
- The accuracy rate is the percentage of the test set samples correctly classified by the model. In case of acceptable accuracy, the model is used for classifying data samples with unknown class labels.

Classification Process

Here's an example of model construction:

Model Construction

Model Usage



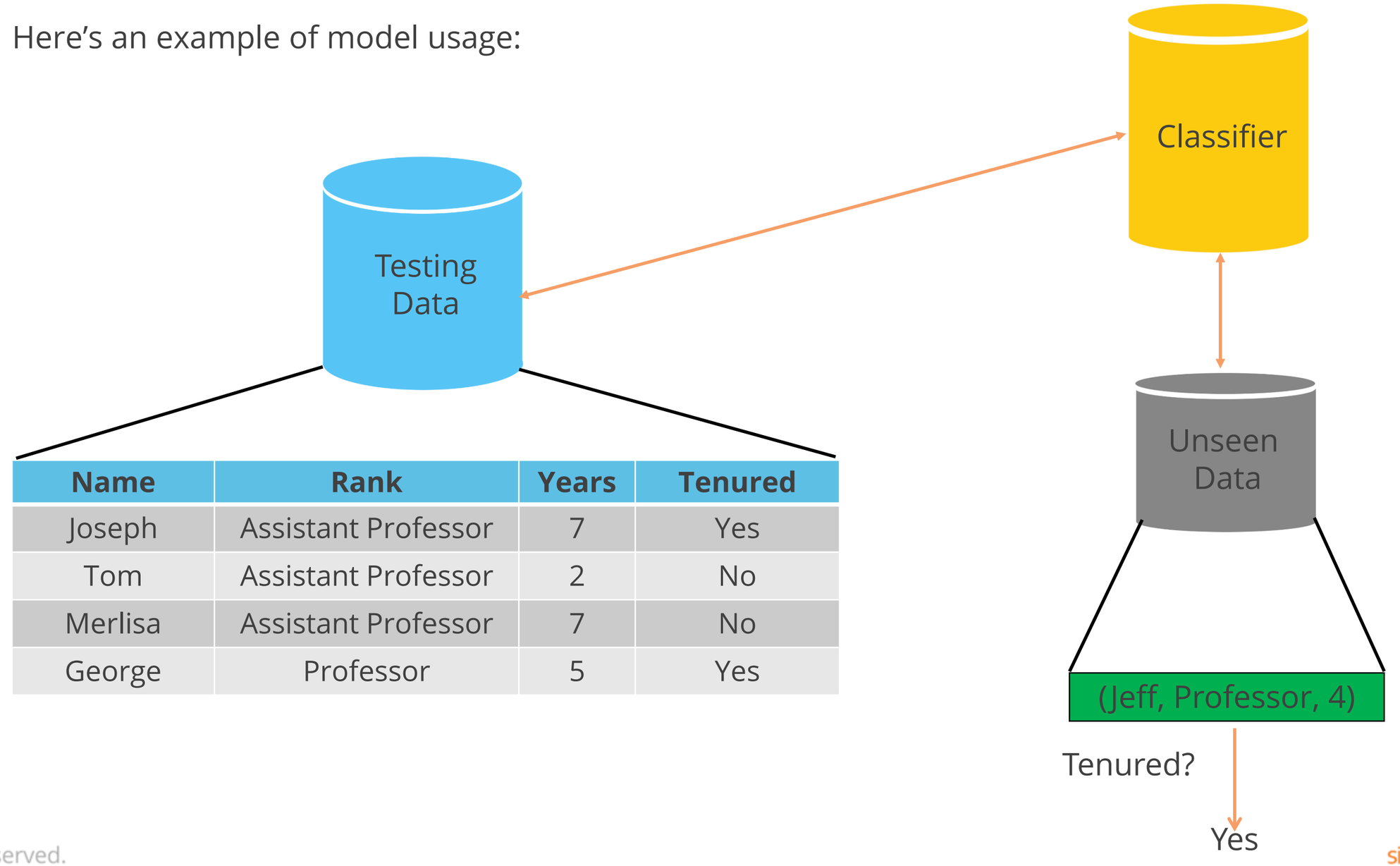
Name	Rank	Years	Tenured
Mike	Assistant Professor	6	No
Bill	Assistant Professor	7	Yes
Dave	Assistant Professor	3	No
Anne	Professor	2	Yes

Classification Process

Here's an example of model usage:

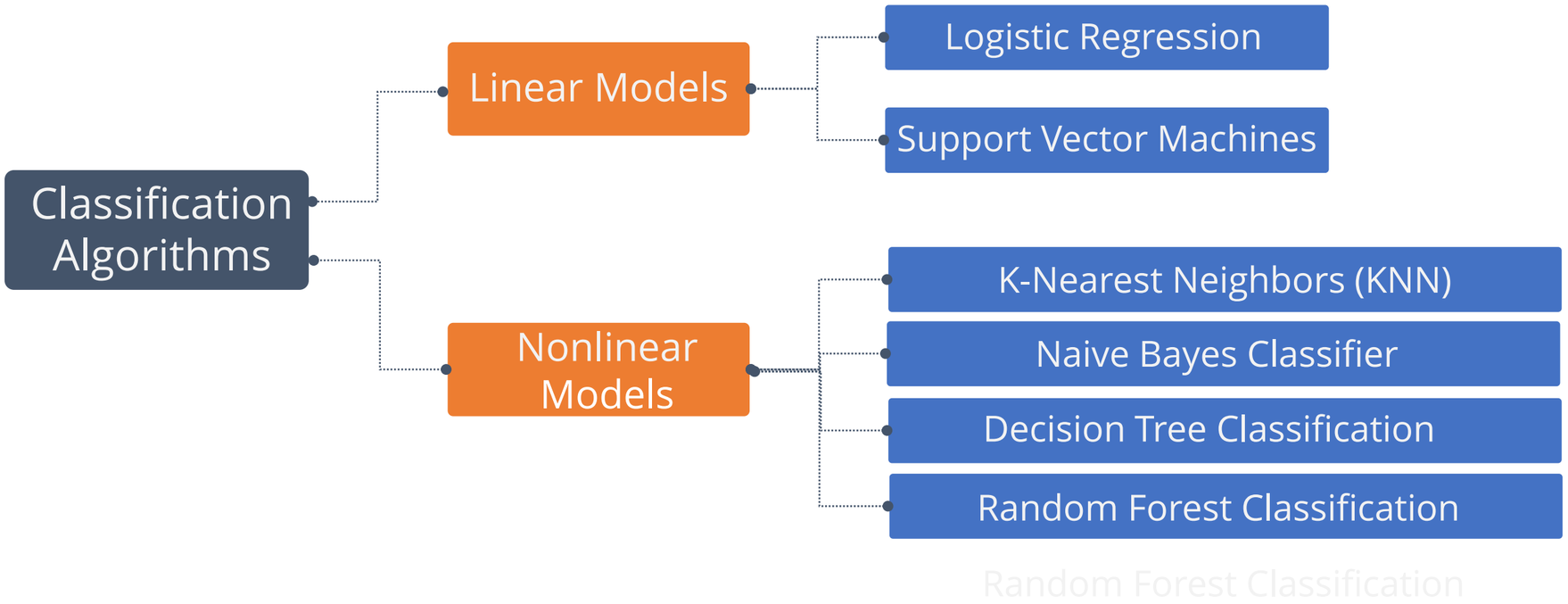
Model Construction

Model Usage



Name	Rank	Years	Tenured
Joseph	Assistant Professor	7	Yes
Tom	Assistant Professor	2	No
Merlisa	Assistant Professor	7	No
George	Professor	5	Yes

Types of Classification Algorithms

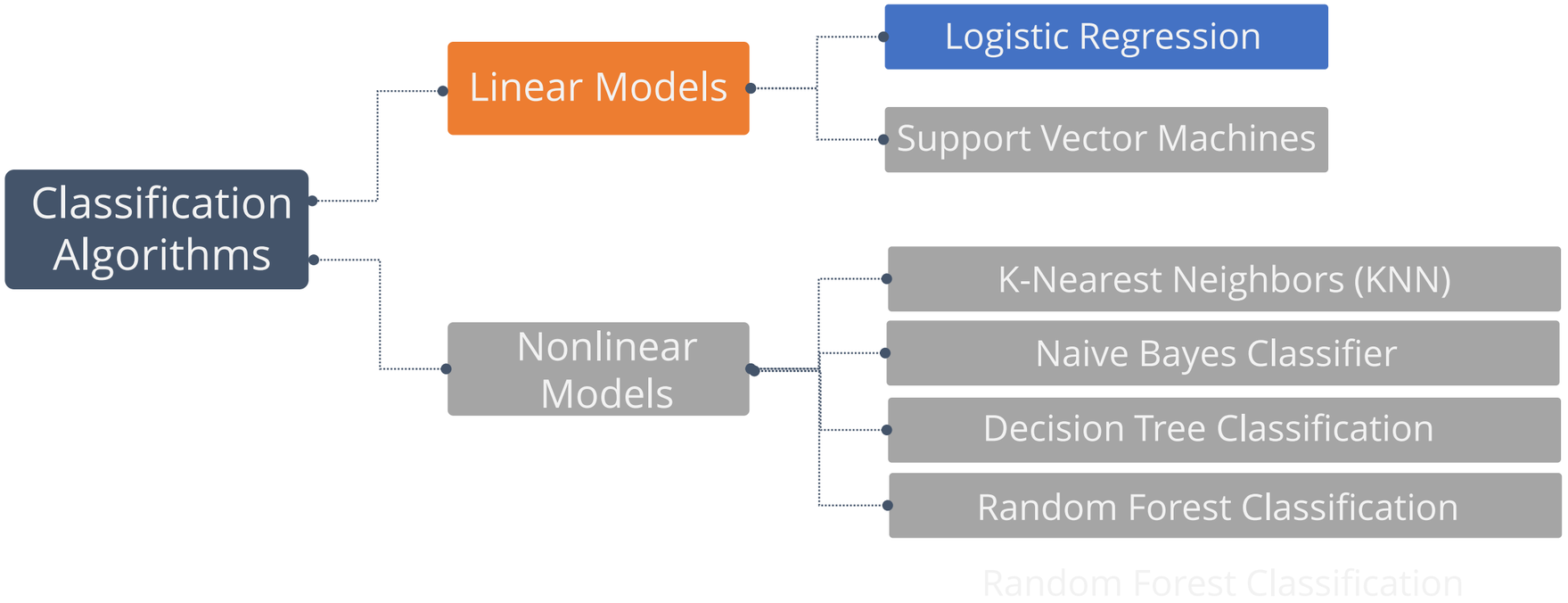


Random Forest Classification

Classification

Topic 2— Logistic Regression

Types of Classification Algorithms

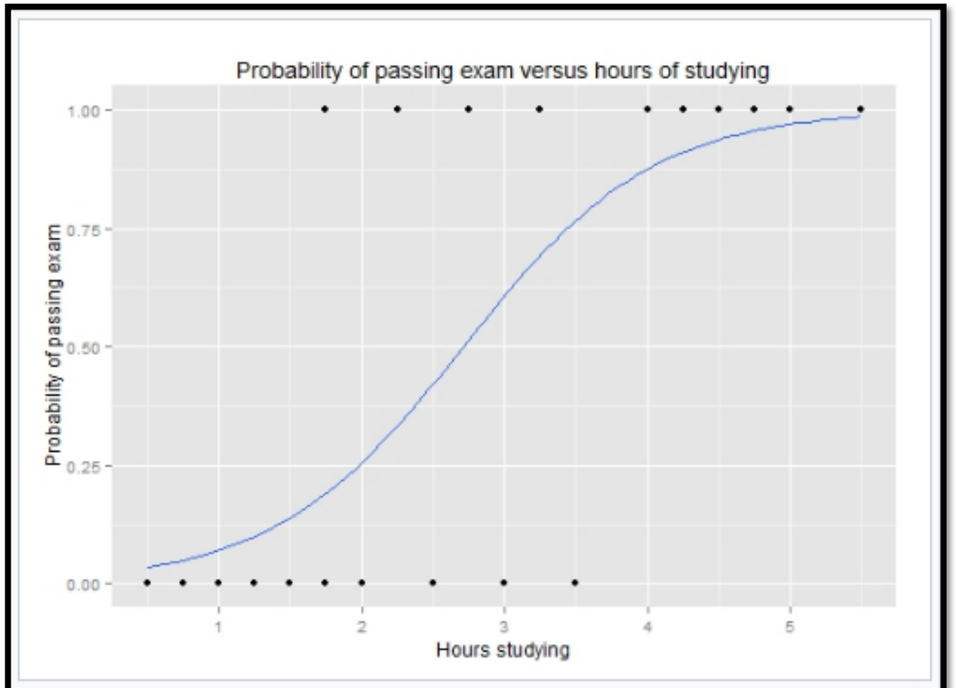


Random Forest Classification

Logistic Regression

- This method is widely used for binary classification problems. It can also be extended to multi-class classification problems.
- Here, the dependent variable is categorical: $y \epsilon \{0, 1\}$.
- A binary dependent variable can have only two values, like 0 or 1, win or lose, pass or fail, healthy or sick, etc.

Logistic Regression



Graph of a logistic regression curve showing probability of passing an exam versus hours studying

- In this case, you model the probability distribution of output y as 1 or 0. This is called as sigmoid probability (σ).
- If $\sigma(\theta Tx) > 0.5$, set $y = 1$, else set $y = 0$.
- Unlike Linear Regression, there is no closed form solution for finding optimal weights of Logistic Regression. Instead, you must solve this with maximum likelihood estimation (a probability model to detect maximum likelihood of something happening).
- It can be used to calculate the probability of a given outcome in a binary model, like probability of being classified as sick or passing an exam.

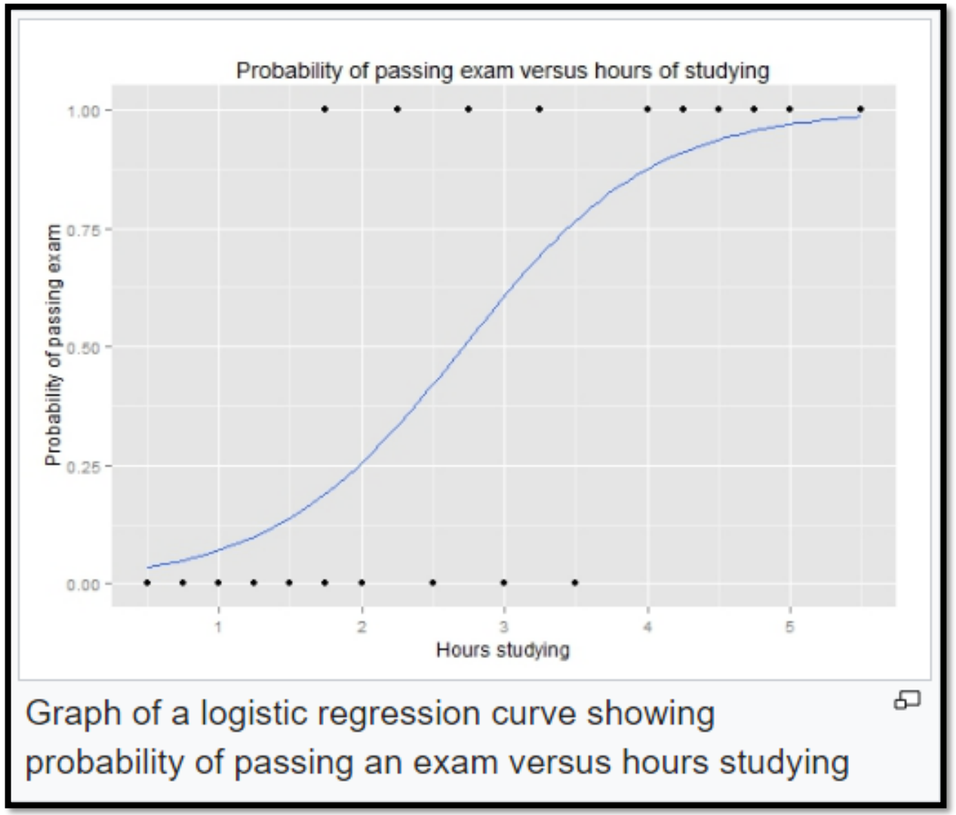
Logistic Regression

$$P(y = 1 \mid x; \theta) = h_\theta(x)$$
$$P(y = 0 \mid x; \theta) = 1 - h_\theta(x)$$

- The first equation shows the probability of output variable y being equal to 1 i.e. P(y = 1). It is equal to sigmoid (σ) of θTx.
- Note that θTx is the vector notation for θ1*x1 + θ2*x2 + θ3*x3 + …. + θn*xn.
- The second equation shows the probability of output variable y being equal to 0 i.e. P(y) = 0.
- The total of two probabilities is 1.

Logistic Regression

SIGMOID PROBABILITY



Graph of a logistic regression curve showing probability of passing an exam versus hours studying

- The probability in the logistic regression is often represented by the Sigmoid function (also called the logistic function or the S-curve):

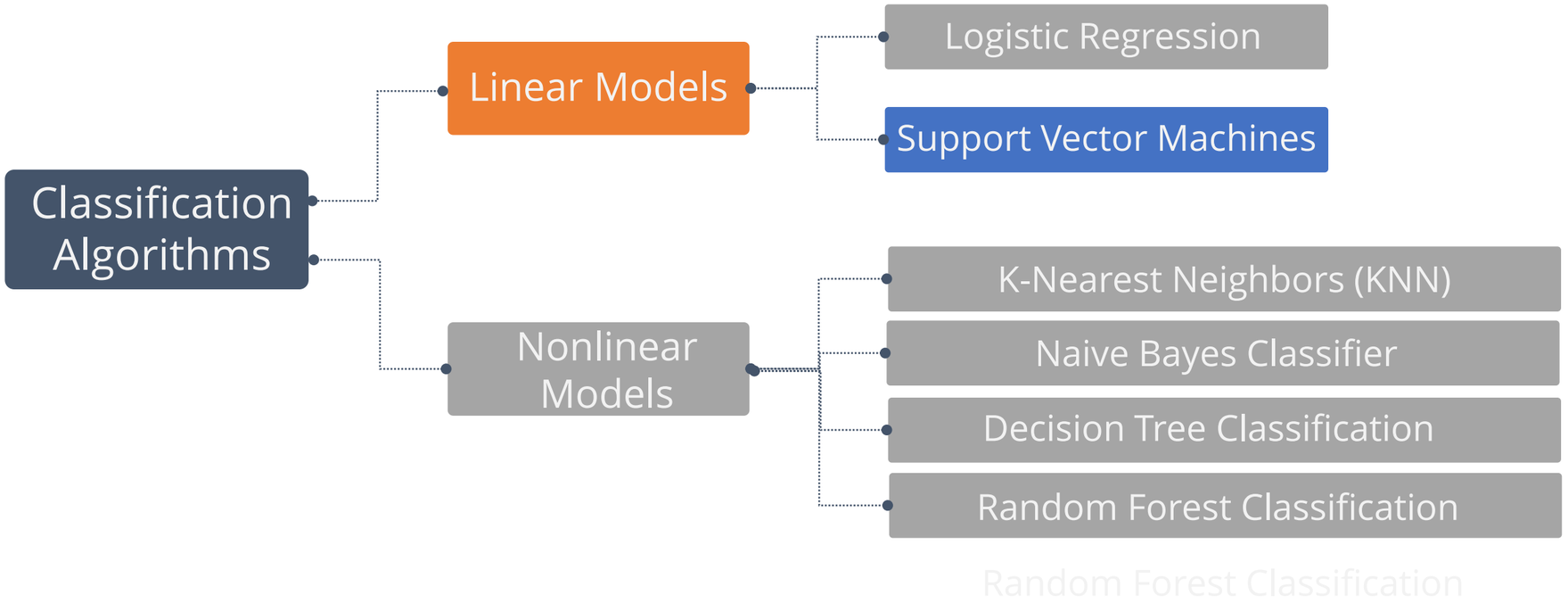
$$S(t) = \frac{1}{1 + e^{-t}}$$

- In this equation, t represents data values * number of hours studied and S(t) represents the probability of passing the exam.
- The points lying on the sigmoid function fits are either classified as positive or negative cases. A threshold is decided for classifying the cases.

Classification

Topic 3— Support Vector Machines

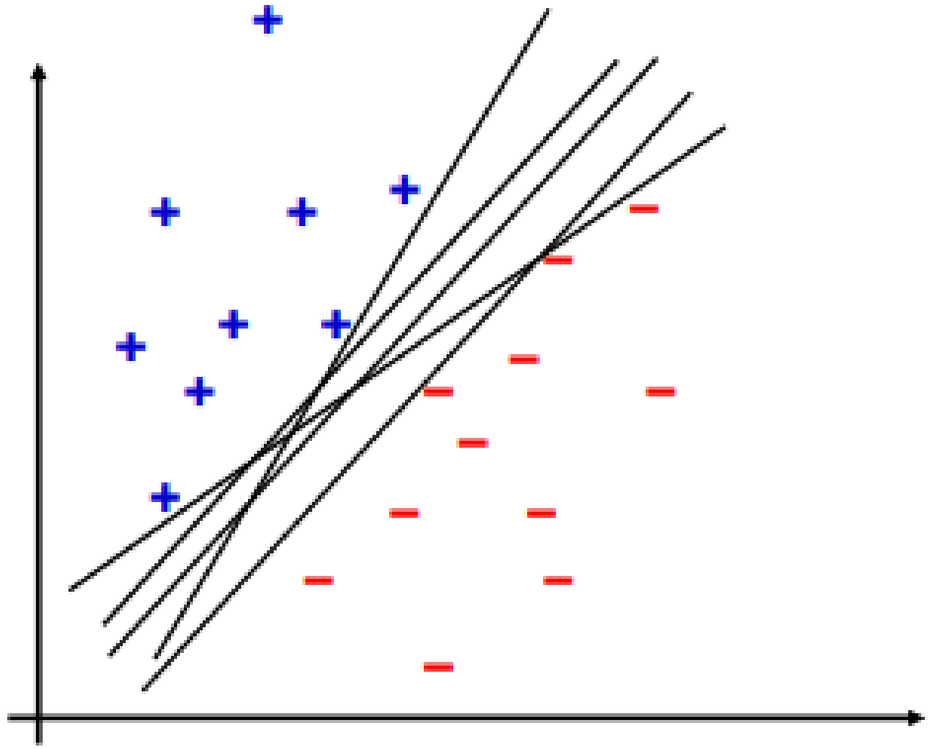
Types of Classification Algorithms



Random Forest Classification

Support Vector Machines

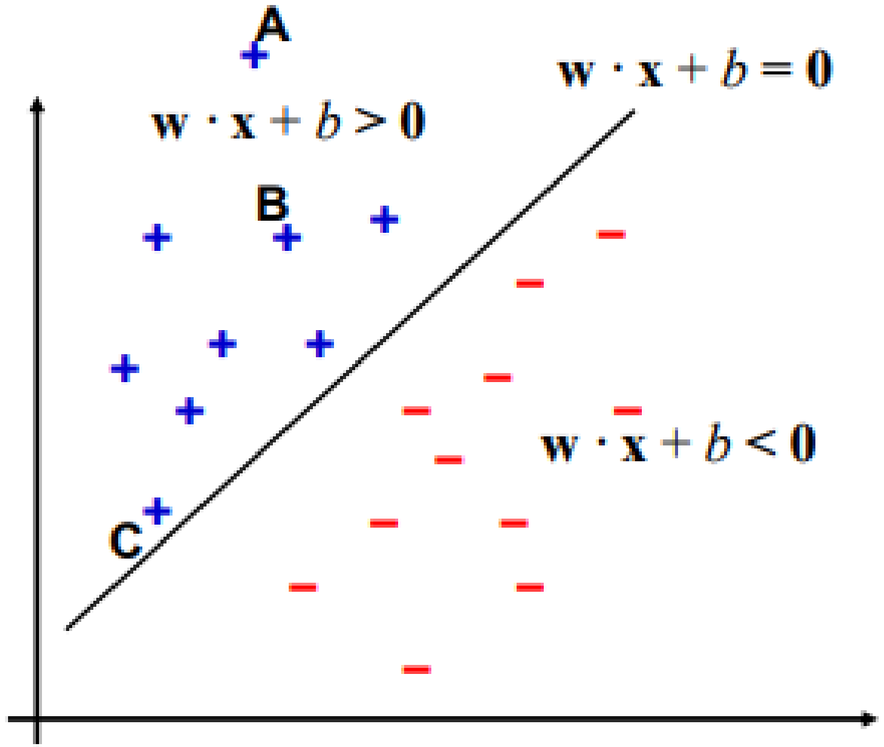
- Support Vector Machines (SVMs) are classification algorithms used to assign data to various classes.
- They involve detecting hyperplanes (decision boundary) which segregate data into classes.



Support Vector Machines

CHOOSING HYPERPLANE

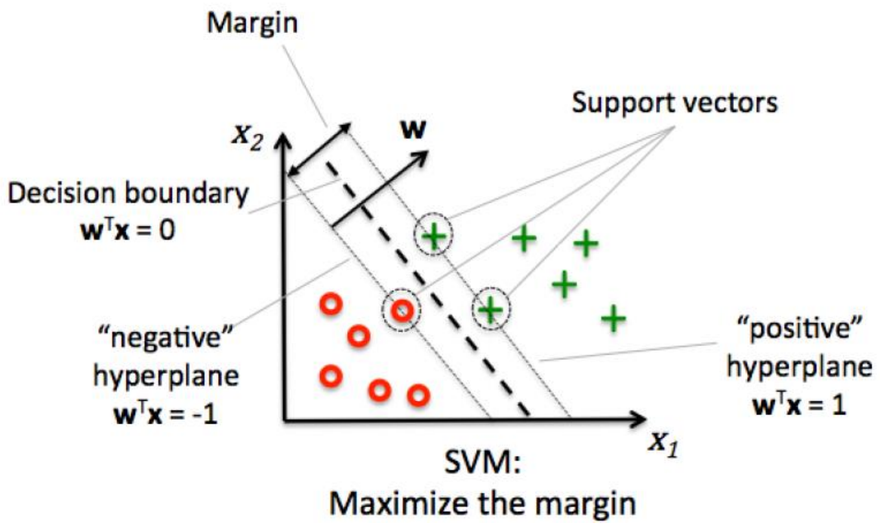
- The hyperplane chosen should be such that all the points in the data set are far away from it. This helps in performing the classification easily.
- For example, in the given graph it is easy to classify points A and B as they are reasonably far away from the hyperplane.
- But one cannot classify C confidently as the point is very close to the hyperplane.



Support Vector Machines

MARGIN

- Once ideal hyperplanes are discovered, new data points can be easily classified.
- The objective of optimization is to maximize the distance between the data points (support vectors) and the hyperplane. This distance is called the margin.



Source: "Python Machine Learning" by Sebastian Raschka

Functional Margin

- Functional margin of a point (x_i, y_i) is:

$$y_i (w^T x_i + b)$$

Where w is a weight vector and b is bias

- Functional margin indicates whether a particular point is properly classified or not. The result would be positive for properly classified points and negative otherwise.

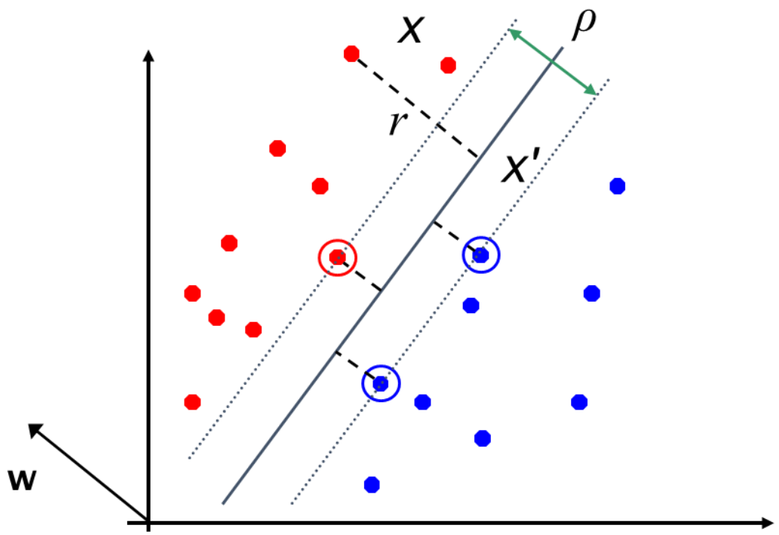
- Assume that the entire data is, at least, at distance 1 from the hyperplane. Then for a training set $\{(x_i, y_i)\}$:

$$w^T x_i + b \geq 1 \quad \text{if } y_i = 1$$
$$w^T x_i + b \leq -1 \quad \text{if } y_i = -1$$

Geometric Margin

A geometric margin is the Euclidean distance between a certain data point x to the hyperplane. Geometric margin not only indicates if the point is properly classified or not, but also calculates the magnitude of the distance in term of units of |w|.

Calculation of Margin



- Distance from a point to the separator is denoted by r.
- A unit vector in this direction is w/|w|. Therefore, the dotted line in the diagram is rw/|w|.
- Assuming the point closest to the hyperplane as x',

$$x' = x - yrw/|w| \quad \text{---------- (1)}$$

and x' satisfies $w^T x' + b = 0$ ---------- (2)

Using (2) in (1), $w^T(x - yrw/|w|) + b = 0$

Therefore:

$$r = y(w^T x + b)/|w|$$

Calculation of Margin

- For the ease of solving large data sets, choose the functional margin of all data points as at least 1. So for all the data points,

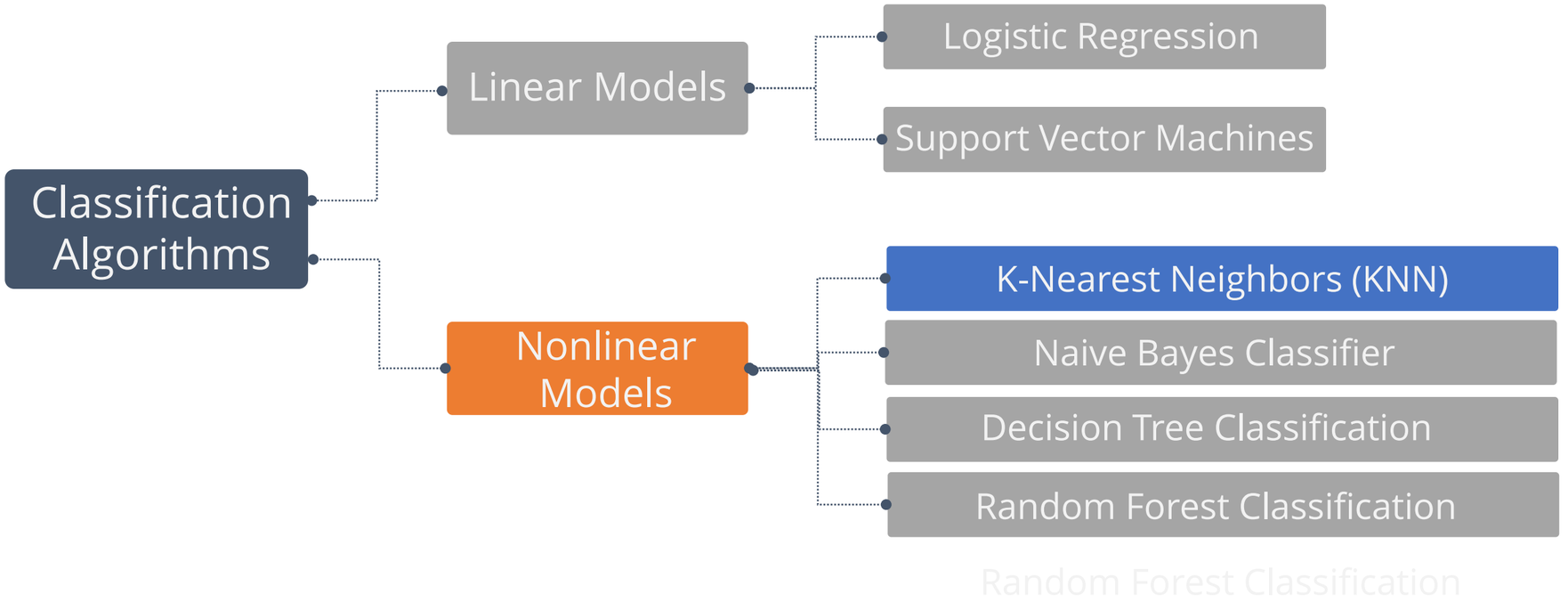
$$y_i(w^T x_i + b) \geq 1$$

- Since each data point's distance from the hyperplane is $r = y(w^T x + b)/|w|$, the geometric margin is $\rho = 2/|w|$ where ρ is the margin.

Classification

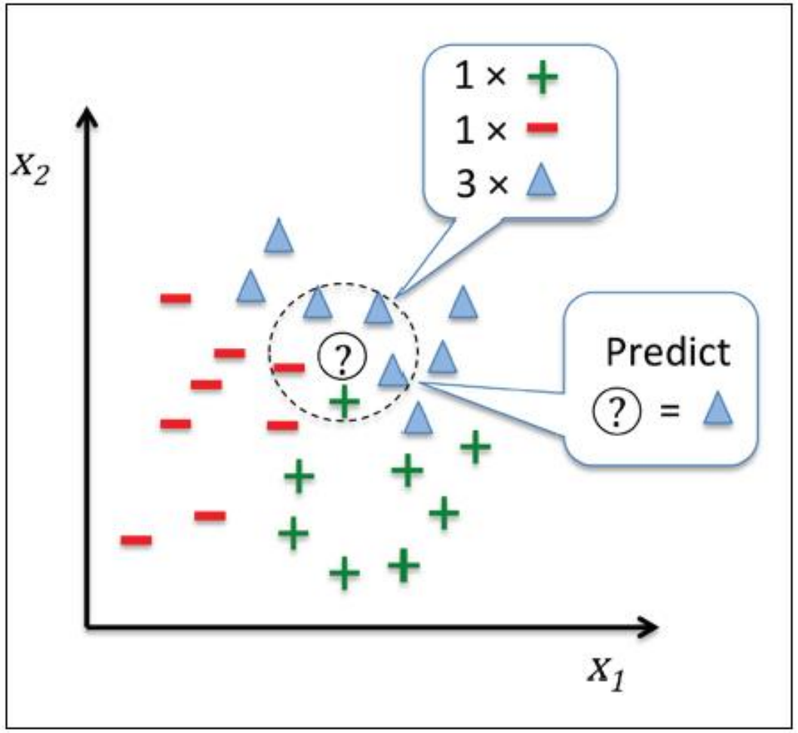
Topic 4— K-Nearest Neighbors (KNN)

Types of Classification Algorithms



K-Nearest Neighbors (KNN)

- K-nearest neighbors is an algorithm that classifies data points by a majority vote of its k neighbors.
- It is used to assign a data point to clusters based on similarity measurement.
- A new input point is classified in the category such that it has the most number of neighbors from that category.
- For example: Marking an email as spam or ham.



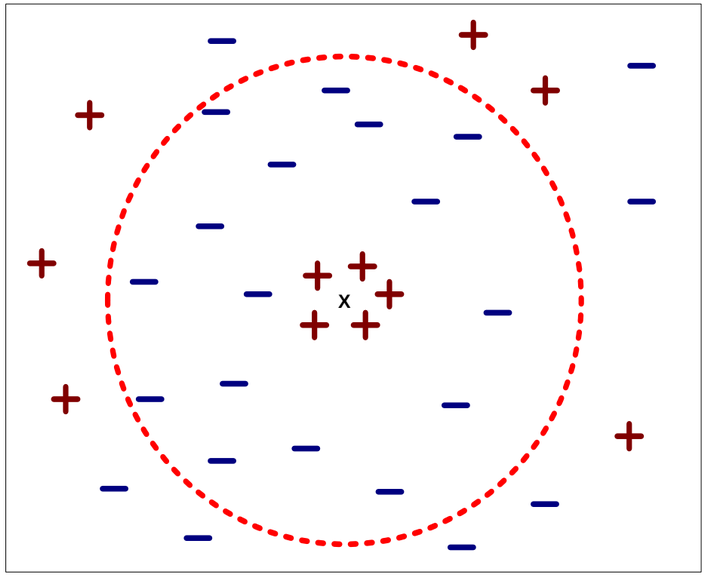
K-Nearest Neighbors (KNN)

STEPS TO CALCULATE THE ALGORITHM

- Calculate the distance of the unknown data points with other training data points i.e. choose the number of k and a distance metric.
- Identify k-nearest neighbors.
- Use category of nearest neighbors to find the category of the new data points based on majority vote.

Choosing the Value of k

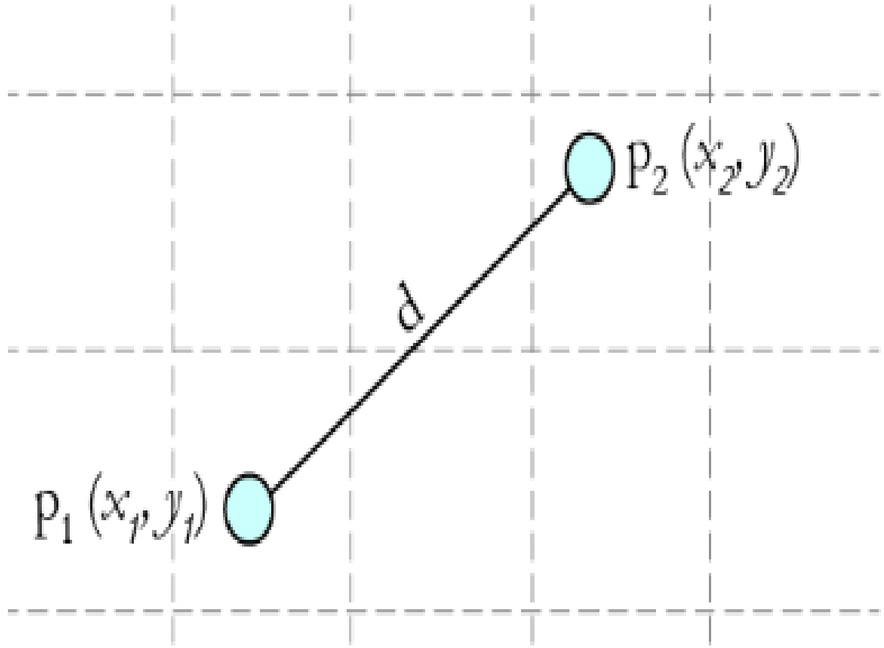
- When choosing the value of k, keep the following points in mind:
  - If its value is too small, neighborhood is sensitive to noise points
  - If its value is too large, neighborhood may include points from other classes



Computing Distance and Determining Class

- For the Nearest Neighbor Classifiers, the distance between two points is expressed in the form of Euclidean distance, which is calculated by:

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



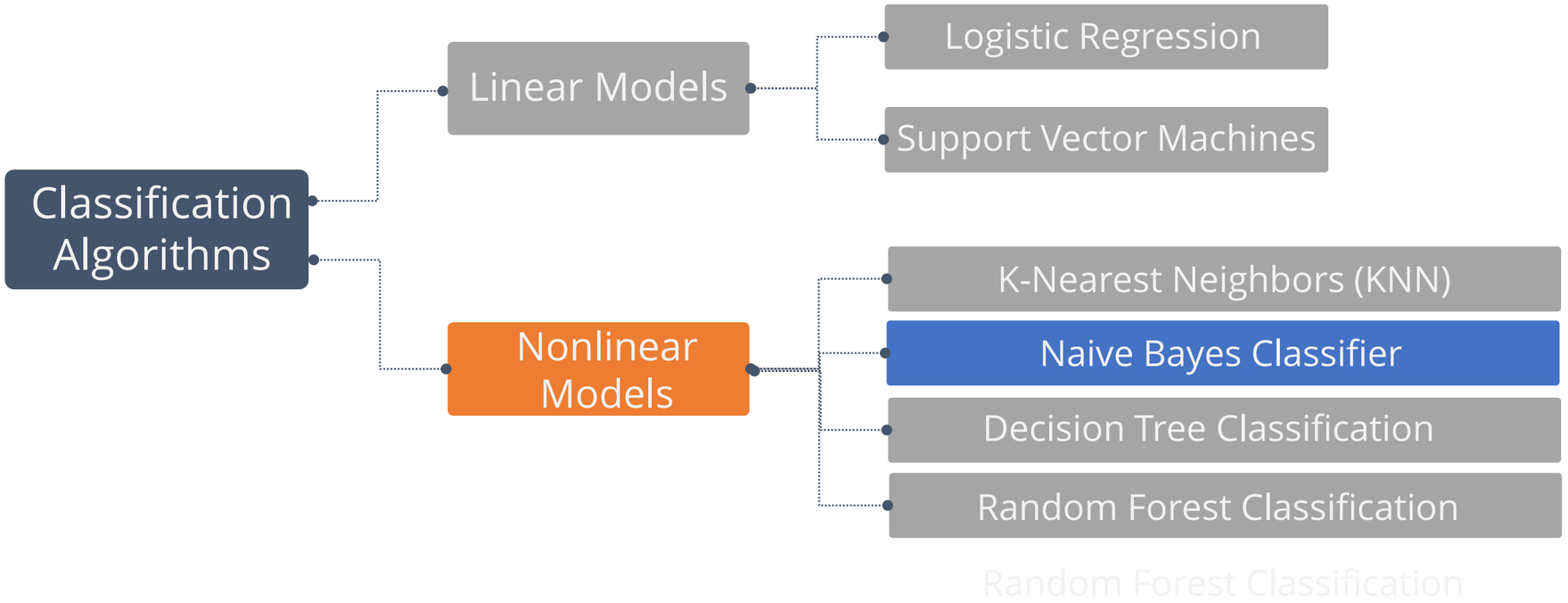
- You can determine the class from the nearest neighbor list by:
  - Taking the majority number of votes of class labels among the k-nearest neighbors
  - Weighing the vote according to the distance

    Weight factor, $w = 1/d^2$

Classification

Topic 5— Naive Bayes Classifier

Types of Classification Algorithms



Random Forest Classification

Naive Bayes Classifier

- This is a probabilistic model which assumes conditional independence between features.
- Given a set of features, Naive Bayes classifier is used to predict a class using probability.

Features of Naive Bayes Classifier

Probabilistic Learning	Determines explicit probabilities for hypothesis
Incremental	Allows each training example to incrementally increase or decrease the probability that a hypothesis is correct
Standard	Provides a standard of optimal decision making to measure other methods
Probabilistic Prediction	Predicts various hypotheses that are weighted by their probabilities

Bayesian Theorem

Assume:

X = Data sample with unknown class label

H = A hypothesis that X belongs to class C

- For classification, you need to determine:
  - P(H|X): Probability that the hypothesis holds, given the observed data sample X
  - P(H): Prior probability of hypothesis H
  - P(X): Probability that the sample data is observed
  - P(X|H): Probability of observing the sample X, given that the hypothesis holds

Bayesian Theorem

- According to Bayes model, the conditional probability P(Y|X) can be calculated as:

$$P(Y|X) = P(X|Y)P(Y) / P(X)$$

- This means you have to estimate a very large number of P(X|Y) probabilities for a relatively small vector space X.
- For example, for a Boolean Y and 30 possible Boolean attributes in the X vector, you will have to estimate 3 billion probabilities P(X|Y).
- To make it practical, a Naive Bayes classifier is used, which assumes conditional independence of P(X) to each other, with a given value of Y.
- This reduces the number of probability estimates to 2*30=60 in the above example.

Detecting Spam

Problem statement

Study

Outcome

Consider a labeled SMS database having 5574 messages. It has messages as given below:

spam	FreeMsg Hey there darling it's been 3 week's now and no word back! I'd like some fun you up for it still? Tb ok! XxX std chgs to send, Â£1.50 to rcv
ham	Even my brother is not like to speak with me. They treat me like untouchable.
ham	As per your request 'Melle (Oru Minnaminunginte Nurungu Vettam)' has been set as your callertune for all Callers. Press *9 to copy your friends Callertune.
spam	WINNER!! As a valued network customer you have been selected to receivea Â£900 prize reward! To claim call 09061701461. Claim code KL341. Valid 12 hours only.

Each message is marked as spam or ham in the data set.

Detecting Spam

Problem statement

Study

Outcome

Let’s train a model with Naive Bayes algorithm to detect spam from ham.

Detecting Spam

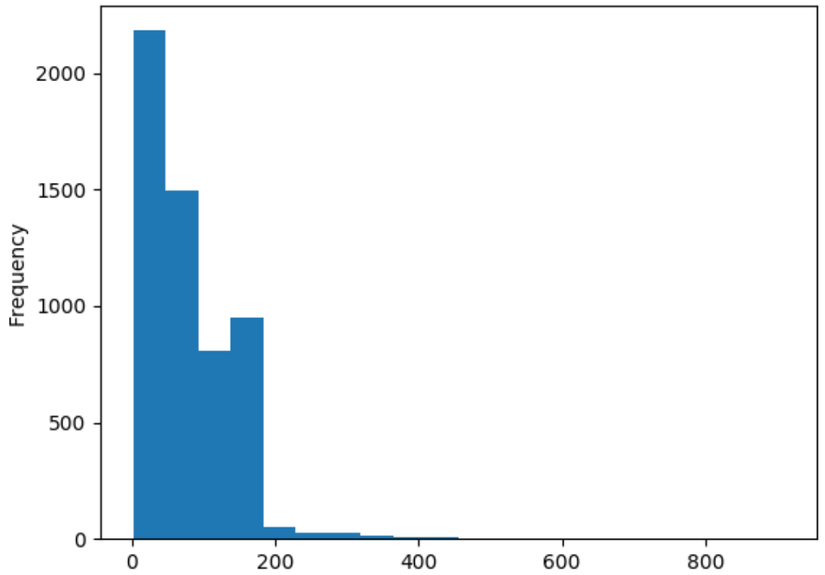
Problem statement

Study

Outcome

The chart shows number of messages on the y axis and SMS word length on the x axis. It indicates frequency of SMS with certain word lengths.

Example: 100 or 200 word SMS



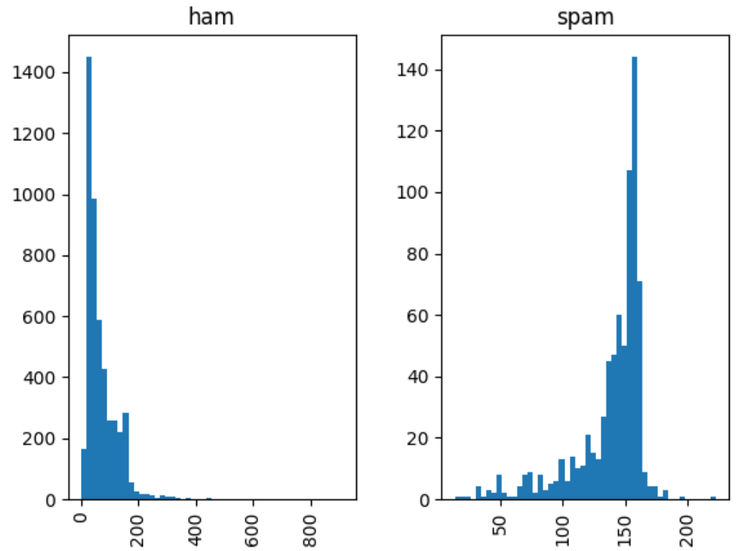
Detecting Spam

Problem statement

Study

Outcome

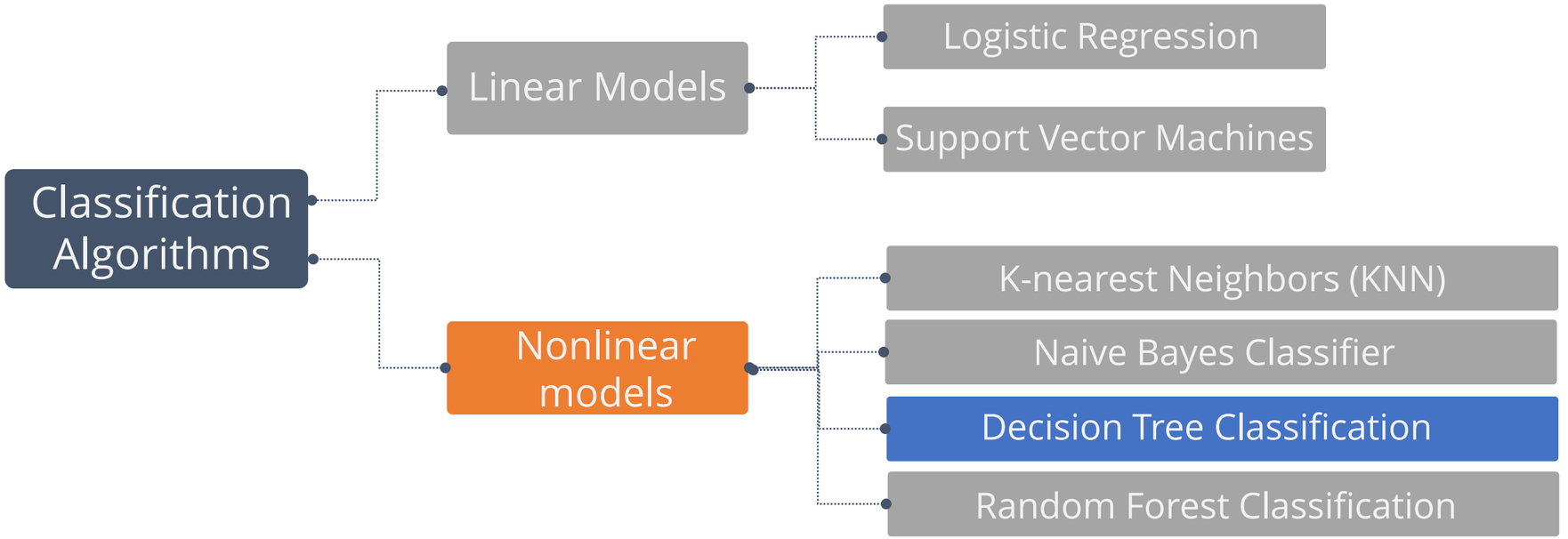
The chart shows same data for ham texts and spam texts. Clearly spam SMS typically has smaller lengths (compared to ham SMS) as is evident from the charts.



Classification

Topic 6 — Decision Tree Classification

Types of Classification Algorithms



Decision Tree Classification

- A decision tree is a graph that makes use of branching method to demonstrate every possible outcome of a decision.
- In classification, the data is segregated based on a series of questions.

Advantages of Decision Tree

- Has a faster learning speed than other classification methods
- Can be converted to easy and simple classification rules
- Can use SQL queries
- Has a high classification accuracy

Basic Algorithm for a Decision Tree

- A tree is constructed in a top-down manner and includes the following steps:

Place all training examples at the root

Categorize the attributes

Partition examples recursively based on the selected attributes

Select test attributes on the basis of a heuristic or statistical measure

Conditions to stop partitioning:

- For a node, all samples belong to the same class.
- No attributes are left for further partitioning.
- No samples are left for classification.

Decision to Buy Computer

Problem statement

Study

Outcome

Consider the given “Buy Computer” dataset. The attributes need to be categorized.

Age	Income	Student	Credit Rating	Buys Computer
≤ 30	High	No	Fair	No
≤ 30	High	No	Excellent	No
31…40	High	No	Fair	Yes
>40	Medium	No	Fair	Yes
>40	Low	Yes	Fair	Yes
>40	Low	Yes	Excellent	No
31…40	Low	Yes	Excellent	Yes
≤ 30	Medium	No	Fair	No
≤ 30	Low	Yes	Fair	Yes
>40	Medium	Yes	Fair	Yes
≤ 30	Medium	Yes	Excellent	Yes
31…40	Medium	No	Excellent	Yes
31…40	High	Yes	Fair	Yes
>40	Medium	No	Excellent	No

Decision to Buy Computer

Problem statement

Study

Outcome

Let us categorize the attributes using a decision tree algorithm until no samples are left for classification.

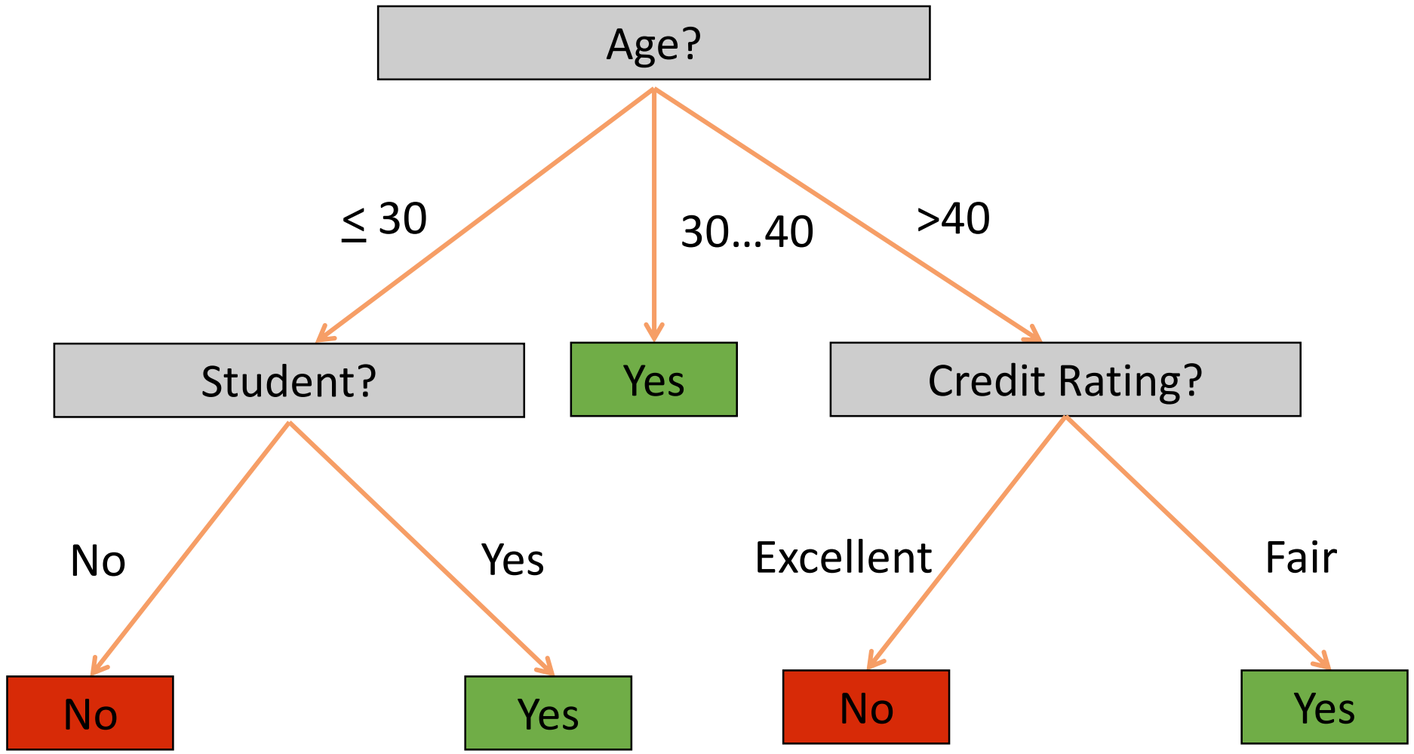
Decision to Buy Computer

Problem statement

Study

Outcome

As an output of the dataset, the following decision tree can be created:



Classification Rules of Trees

- In these rules:
  - The statements are represented as IF-THEN rules
  - There is, at least, one rule for every path from the root to a leaf in a tree
  - A conjunction is formed for every attribute-value pair along a path in a tree
  - The class prediction is held by the leaf node in a tree

Let's apply these rules on the "Buy Computer" dataset:

IF Age = "<=30" AND Student = "No" THEN buys_computer = "No"
IF Age = "<=30" AND Student = "Yes" THEN buys_computer = "Yes"
IF Age = "31…40" THEN buys_computer = "Yes"
IF Age = ">40" AND Credit Rating = "Excellent" THEN buys_computer = "Yes"
IF Age = "<=30" AND Credit Rating = "Fair" THEN buys_computer = "No"

Overfitting in Classification

- Sometimes, a tree may overfit the training data which can lead to issues, such as:
  - Too many branches
  - Less accurate and unseen samples

How to avoid overfitting?

There are two approaches:

- **Prepruning**: Stop the construction of a tree early. If the goodness measure is falling below a threshold, do not split the node.
- **Postpruning**: In case selecting an appropriate threshold is difficult, remove branches from a fully-developed tree by getting a progressively pruned trees' sequence.

Tips to Find the Final Tree Size

Tip 1	Separate training (2/3) and testing (1/3) sets
Tip 2	Apply cross-validation
Tip 3	Use a statistical test (for example, chi-square) to determine whether pruning or expanding a node can improve the distribution

Information Gain

- Entropy is a measure of impurity, and information gain is the reduction that occurs in entropy as one traverses down the tree.
- You need to select an attribute with the highest information gain, which is defined as:

$$\text{Gain}(A) = I(s_1, s_2, ..., s_m) - E(A)$$

Where,

S contains si samples of class Ci for i = {1, ..., m}

$$I(s_1, s_2, ..., s_m) = -\sum_{i-1}^{m} \frac{si}{s} \log_2 \frac{si}{s}$$

Entropy of attribute A with values $\{a_1, a_2, ..., a_v\}$:

$$E(A) = \sum_{j=1}^{v} \frac{s_{1j} + ... + s_{mj}}{s} I(s_{1j}, ..., s_{mj})$$

Information Gain

EXAMPLE

Now, let's consider the "Buy Computer" dataset, and calculate Gain(A).

Age	Income	Student	Credit Rating	Buys Computer
≤ 30	High	No	Fair	No
≤ 30	High	No	Excellent	No
31…40	High	No	Fair	Yes
>40	Medium	No	Fair	Yes
>40	Low	Yes	Fair	Yes
>40	Low	Yes	Excellent	No
31…40	Low	Yes	Excellent	Yes
≤ 30	Medium	No	Fair	No
≤ 30	Low	Yes	Fair	Yes
>40	Medium	Yes	Fair	Yes
≤ 30	Medium	Yes	Excellent	Yes
31…40	Medium	No	Excellent	Yes
31…40	High	Yes	Fair	Yes
>40	Medium	No	Excellent	No

Assume: Class P: Buy Computer = "Yes"

Class N: Buy Computer = "No"

Information Gain

EXAMPLE

The attributes of the “Buy Computer” table can be categorized as:

Age	p_i	n_i	$I(p_i, n_i)$
≤ 30	2	3	0.971
>40	3	2	0.971
31…40	4	0	0

From the dataset, for age≤30, class p_i=2, class n_i=3.

$$I(p_i,n_i) = -\left(\frac{2}{5}log_2\frac{2}{5} + \frac{3}{5}log_2\frac{3}{5}\right)$$

$$= 0.971$$

Similarly, $I(p, n) = I(9, 5) = 0.940$

Information Gain

EXAMPLE

The entropy to identify the age will be calculated as follows:

$$E(age) = \frac{5}{14} I(2,3) + \frac{4}{14} I(4,0) + \frac{5}{14} I(3,2) = 0.694$$

Therefore, Gain(Age) will be calculated as follows:

$$Gain(age) = I(p,n) - E(age) = 0.246$$

Similarly for the other attributes,

$$Gain(income) = 0.029$$

$$Gain(student) = 0.151$$

$$Gain(credit\_rating) = 0.048$$

Information Gain for Continuous-Value Attributes

- A continuous-value attribute is one which takes numeric values.
- Assume that A is a continuous-valued attribute. To calculate its Information Gain, you must determine the best midpoint for A by:
  - Sorting the values of A in an increasing order.
  - Selecting the midpoint between each pair of adjacent values.
  - Calculating entropy of each value.

Information Gain for Continuous-Value Attributes

EXAMPLE

Using the “Buy Computer” dataset, let us sort the data in increasing order.

Age	Buy Computer
18	Yes
18	No
25	Yes
28	Yes
28	No
34	No
45	No

For middle point in first 2 numbers:

Mid point = (18+25)/2 = 21

Information Gain for Continuous-Value Attributes

EXAMPLE

Information gain, Infoage<21(D):

= 2/7(I(1,1)) + 5/7(I(2,3))

= 2/7 (-1/2(log2(1/2)) – 1/2(log2(1/2)) + 5/7(-2/5(log2(2/5))-3/5(log2(3/5)))

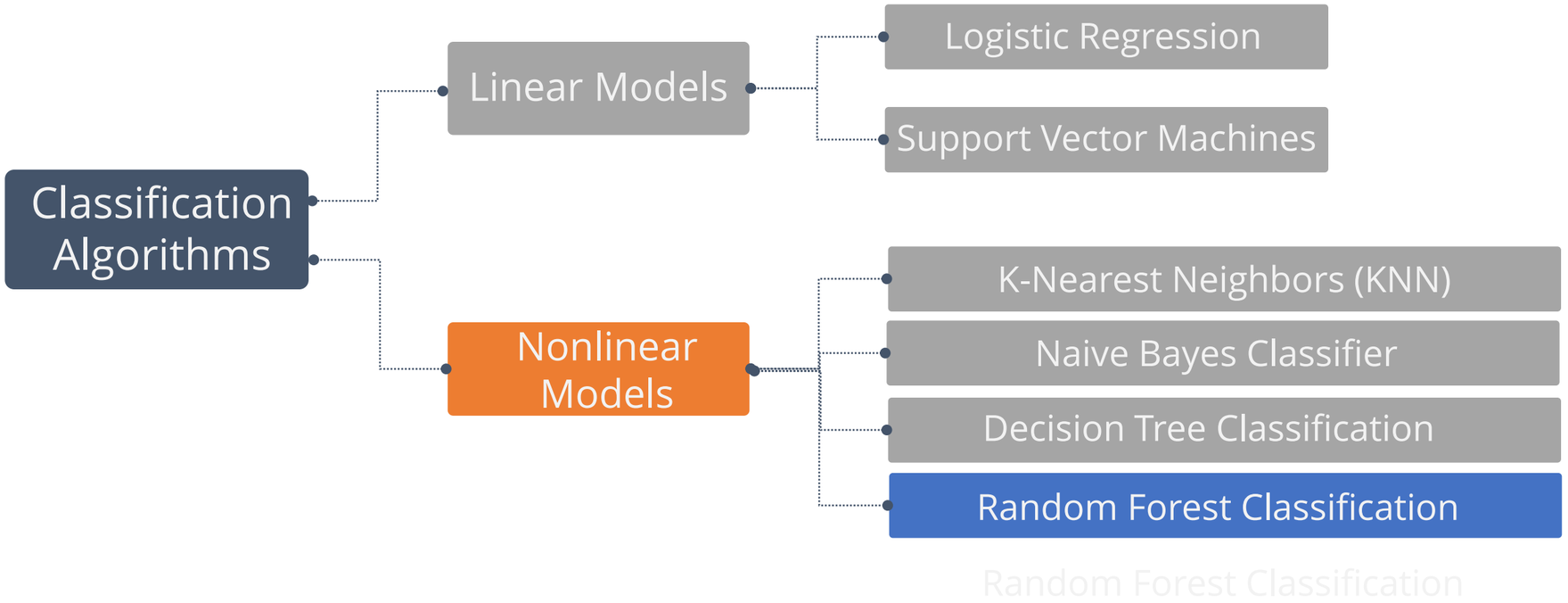
=.98

Age	Buy Computer
18	Yes
18	No
25	Yes
28	Yes
28	No
34	No
45	No

Classification

Topic 7 — Random Forest Classification

Types of Classification Algorithms



Random Forest Classification

- A random forest can be considered an ensemble of decision trees. It builds and combines multiple decision trees to get a more accurate prediction.
- Each of the decision tree models used is weak when employed on its own, but it becomes stable when put together.

They are called random because they operate by choosing predictors randomly at the time of training the model.
They are called forests because they take the output of multiple decision trees to make a decision.

Random Forest Algorithm

STEPS TO FOLLOW

- Draw a random bootstrap sample of size n (randomly choose n samples from the training set).
- Grow a decision tree from the bootstrap sample. At each node, randomly select d features.
- Split the node using the feature that provides best split according to objective function, for instance by maximizing the information gain.
- Repeat the steps 1 to 2 k times (k is the number of trees you want to create, using a subset of samples).
- Aggregate the prediction by each tree for a new data point to assign the class label by majority vote (pick the group selected by most number of trees and assign new data point to that group).

Classification

Topic 8 — Evaluating Classifier Models

Evaluating Classifier Models

- Evaluating a model is important to know the accuracy and performance of a model.
- To evaluate a model, different metrics are used.
  - Confusion Matrix
  - Gain and Lift Chart
  - Kolmogorov Smirnov Chart
  - AUC – ROC curve
  - Gini Coefficient
  - Concordant – Discordant Ratio
  - Root Mean Squared Error

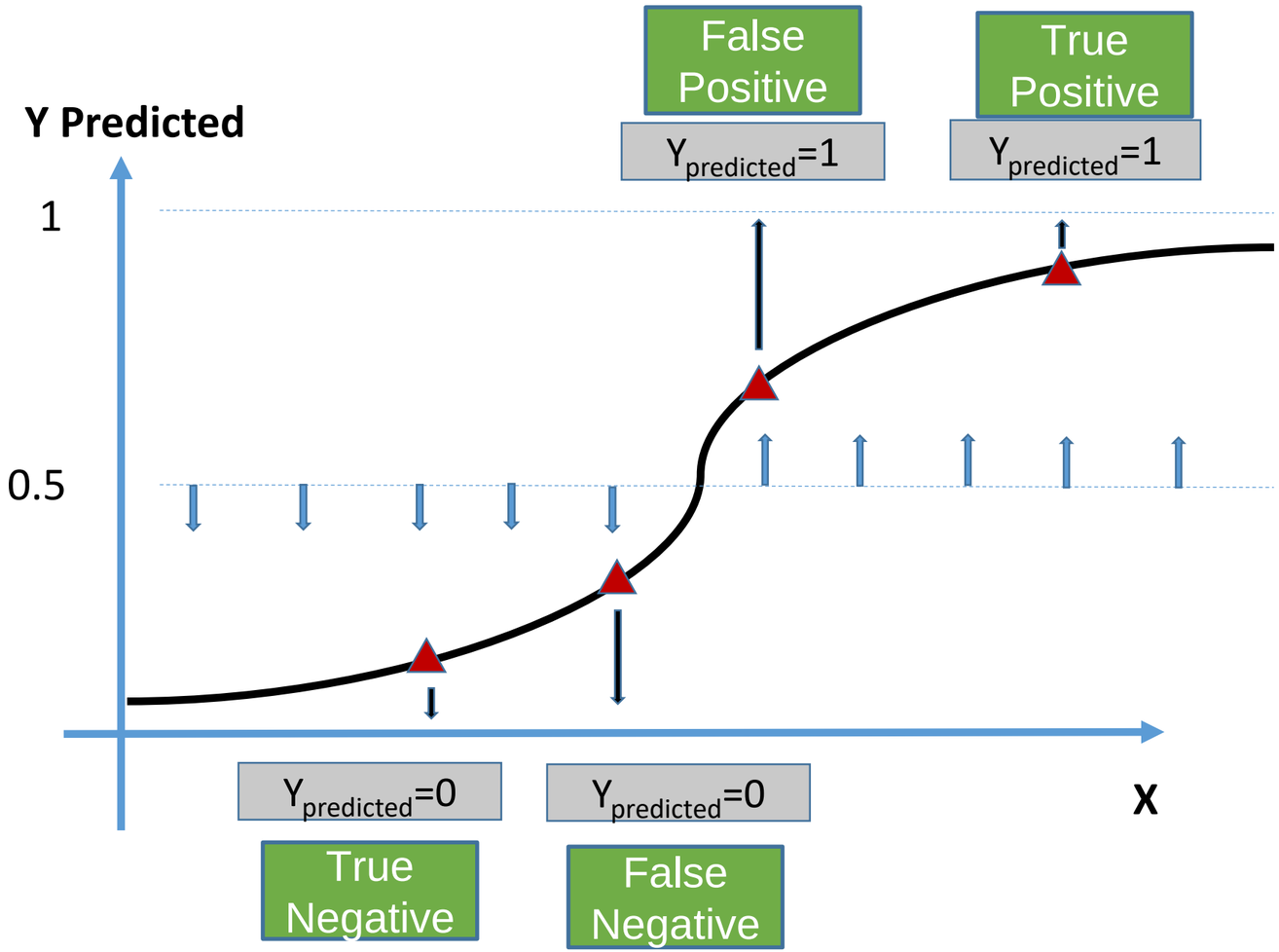
In this course the focus will be on confusion matrix and AUC-ROC curve.

Confusion Matrix

A confusion matrix examines all possible outcomes of prediction: true positive, true negative, false positive and false negative.

Confusion Matrix

FALSE POSITIVE AND FALSE NEGATIVE



- False Positives are like false alarms. They are called Type I error. They occur when a negative occurrence is wrongly classified as positive.

- False Negatives are also called Type II error. They occur when a positive occurrence is wrongly classified as negative.

Confusion Matrix

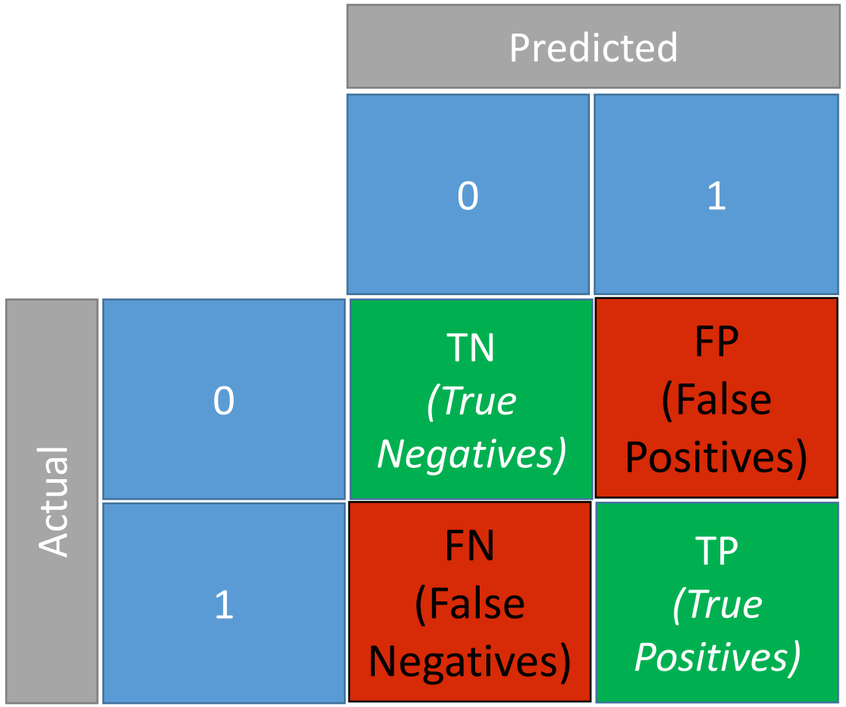
PARAMETERS

The parameters calculated from a confusion matrix are:

- Accuracy rate: The proportion of the total number of predictions that were right
- Precision/Positive Predicted Value: The proportion of positive cases that were correctly identified
- Negative Predictive Value: The proportion of negative cases that were correctly identified
- Recall/Sensitivity/True Positive Rate: The proportion of actual positive cases which are correctly identified
- Specificity/ True Negative Rate: The proportion of actual negative cases which are correctly identified

Confusion Matrix

PARAMETERS



Accuracy Rate = *(TP + TN) /(TP+TN+FP+FN)*

Error Rate = *(FP + FN) /(TP+TN+FP+FN)*

Precision/ Positive Predicted Value = *(TP) /(TP+FP)*

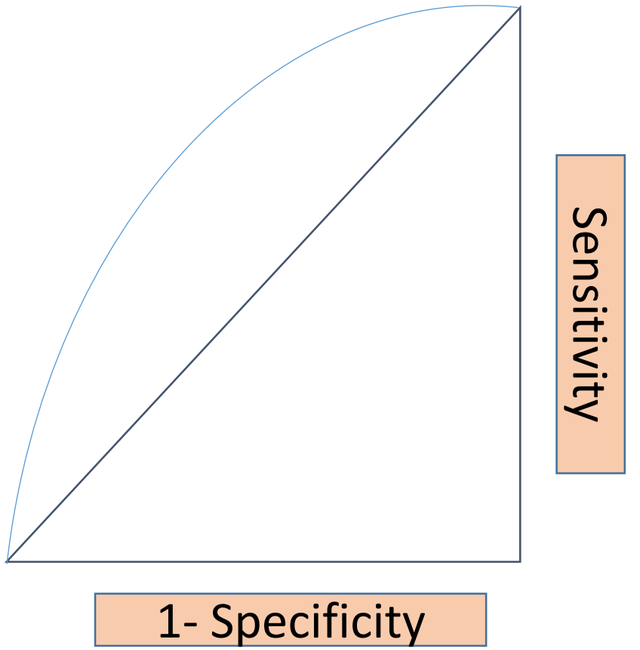
Recall/Sensitivity/True Positive Rate = *(TP) /(TP+FN)*

Specificity/ True Negative Rate= *(TN) /(TN+FP)*

AUC – ROC Curve

The ROC (Receiver Operating Characteristic curve) is the plot between True Positive Rate (Sensitivity) and the False Positive Rate (1- Specificity) for a classifier.

AUC: AUC or the Area Under the Curve is a measure of classifier's performance. A random classifier has an AUC of 0.5, whereas a perfect classifier has an AUC of 1.

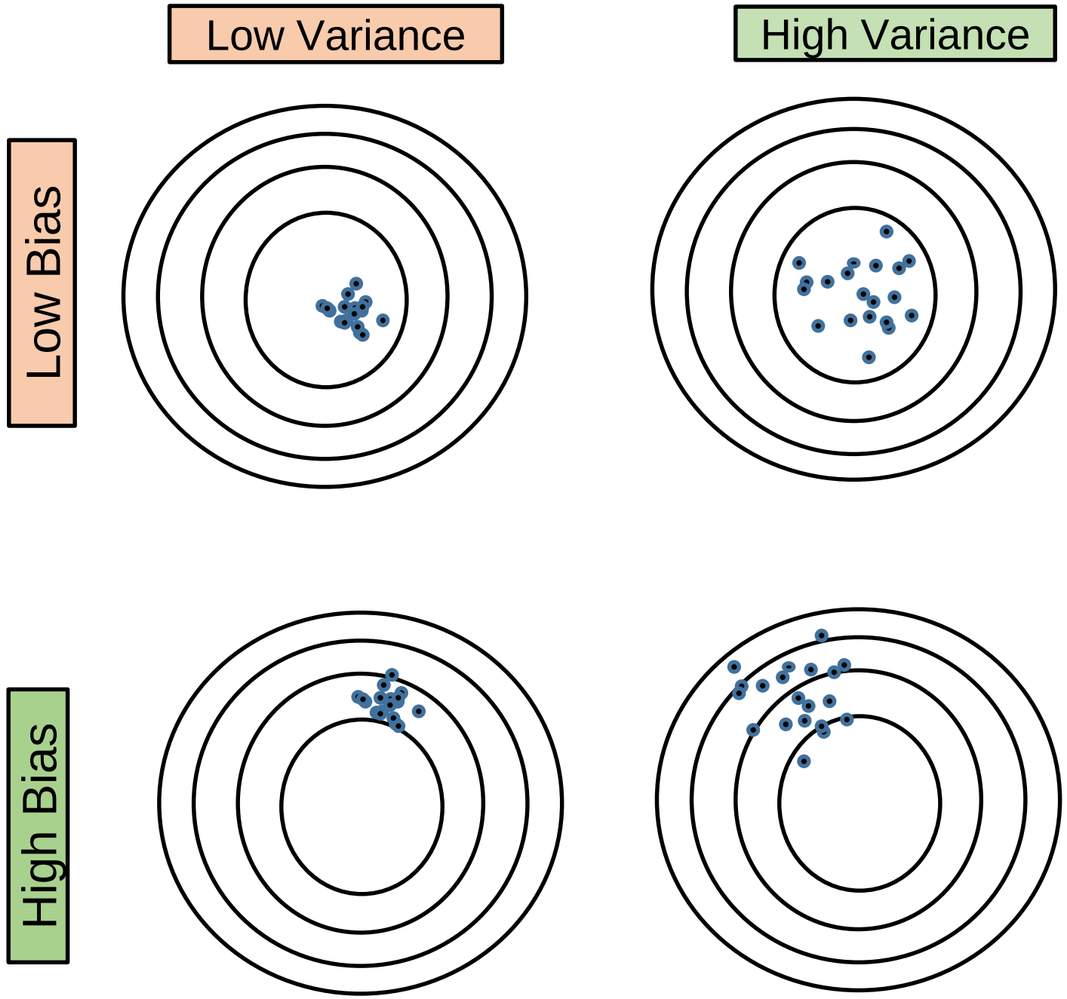


Bias Variance Trade-off

- Bias variance trade-off determines the model's ability to keep bias and variance to the minimum.
- Bias is a measure of error on how much the predicated values differ from the actual value.
- Variance indicates an algorithm's sensitivity to small changes in the training dataset.
- The error in a predictive model can be summarized as a summation of bias, variance, and irreducible error.
- Irreducible error, also known as noise, cannot be reduced by any algorithm.

Bias Variance Trade-off

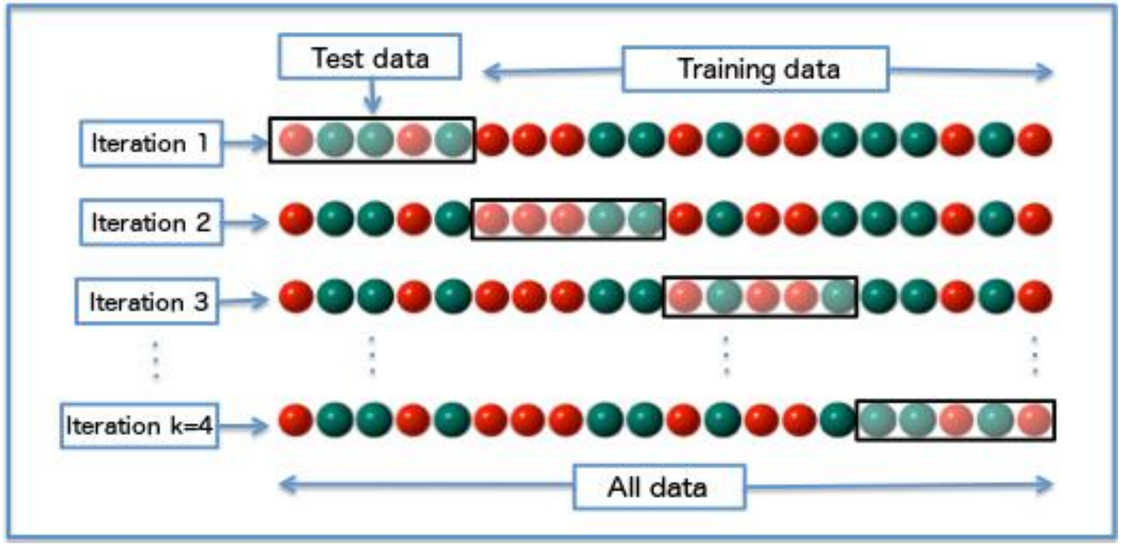
The goal of any classification algorithm is to achieve low bias and low variance.



K-Fold Cross Validation

ALGORITHM

- Original sample data is split into k random samples of equal sizes each.
- One out of k number of samples is selected as Test data while other k-1 samples are combined together into Training data. The model is built on k-1 folds and tested on k^{th} fold.
- Repeat the process for each of the k^{th} fold. The test data is rotated each time until all k number of samples have been allotted to test data at least once.
- The average error across folds is called the performance of the model.



Key Takeaways

- Classification is a technique to determine the extent to which a data sample will or will not 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 machines, 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.