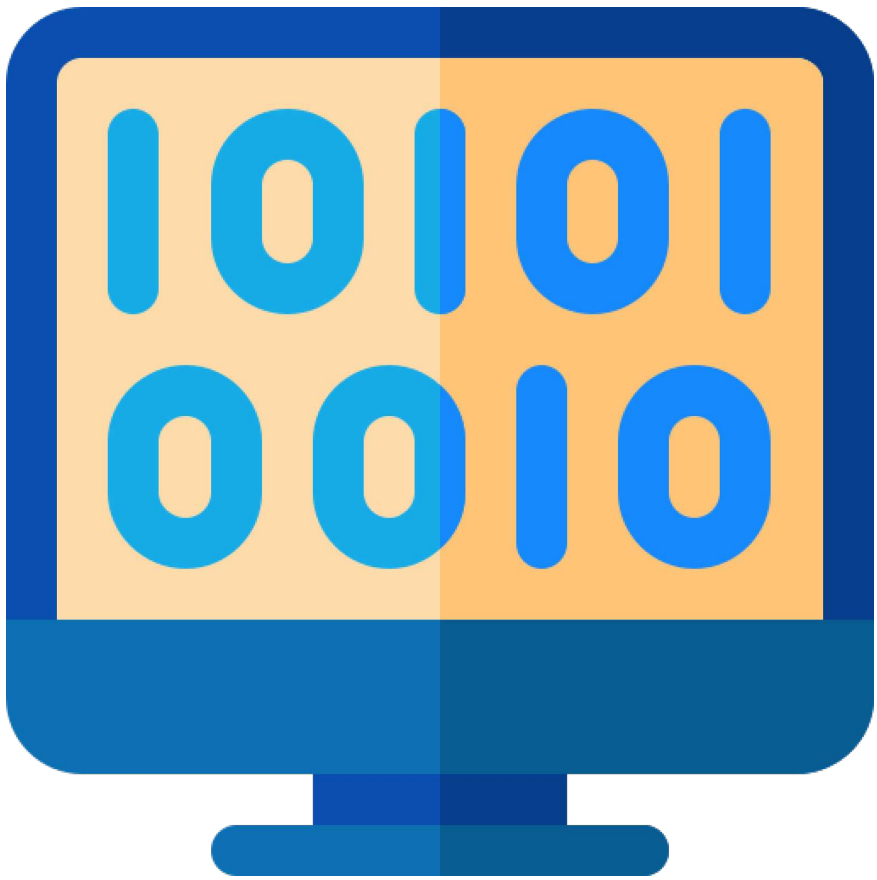
Binary Classification

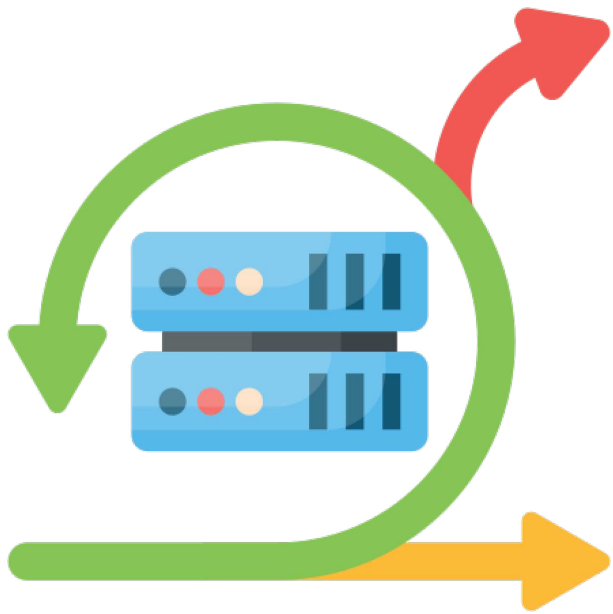


What is Binary Classification?

In machine learning, binary classification is a supervised learning algorithm that categorizes new observations into one of **two** outcomes usually represented as 0 or 1, true or false, positive or negative, etc.

For example, predicting whether a credit card transaction is fraud or not fraud, whether an email is a spam or not spam, and whether a customer will purchase a product or not, are all examples of binary classification problems.

How Binary Classification Works?



In binary classification, the algorithm is trained on a labeled dataset, where each data point is associated with a binary label.

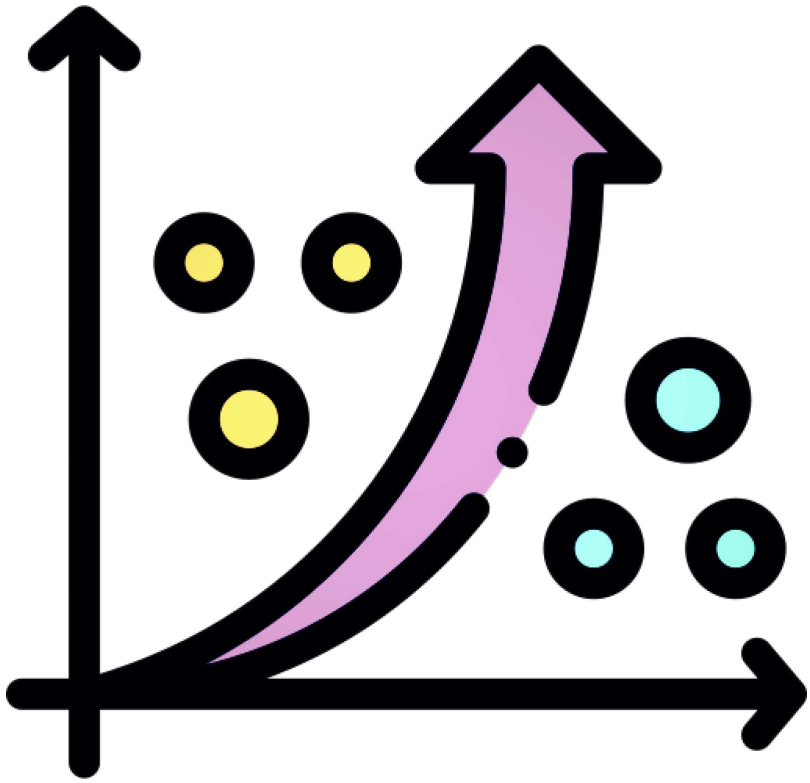
The algorithm then learns to map the input features to the corresponding binary label. Once trained, the algorithm can be used to predict the binary label for new, unseen data points.

Common Binary Classification Models

DATARANCH.org

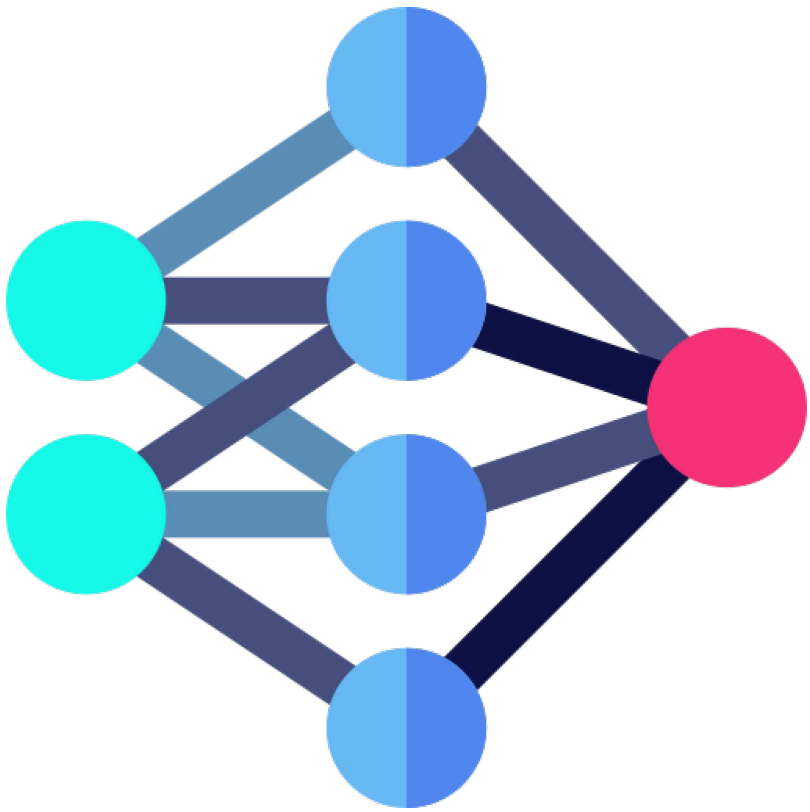
VISUALIZE | ANALYZE | CAPITALIZE

Logistic Regression



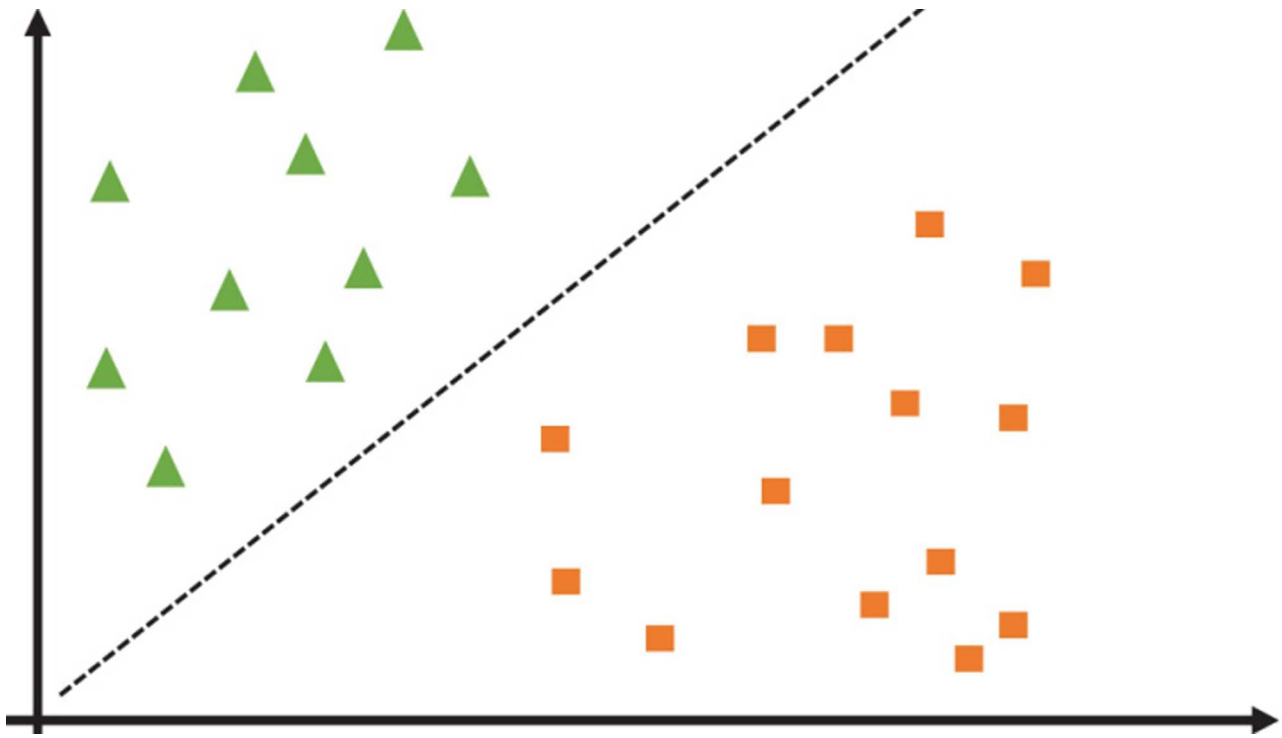
It is used for binary classification problems, where the output variable is categorical with two possible values. For example, predicting whether a customer will buy a product or not based on their demographics and purchase history.

Neural Networks



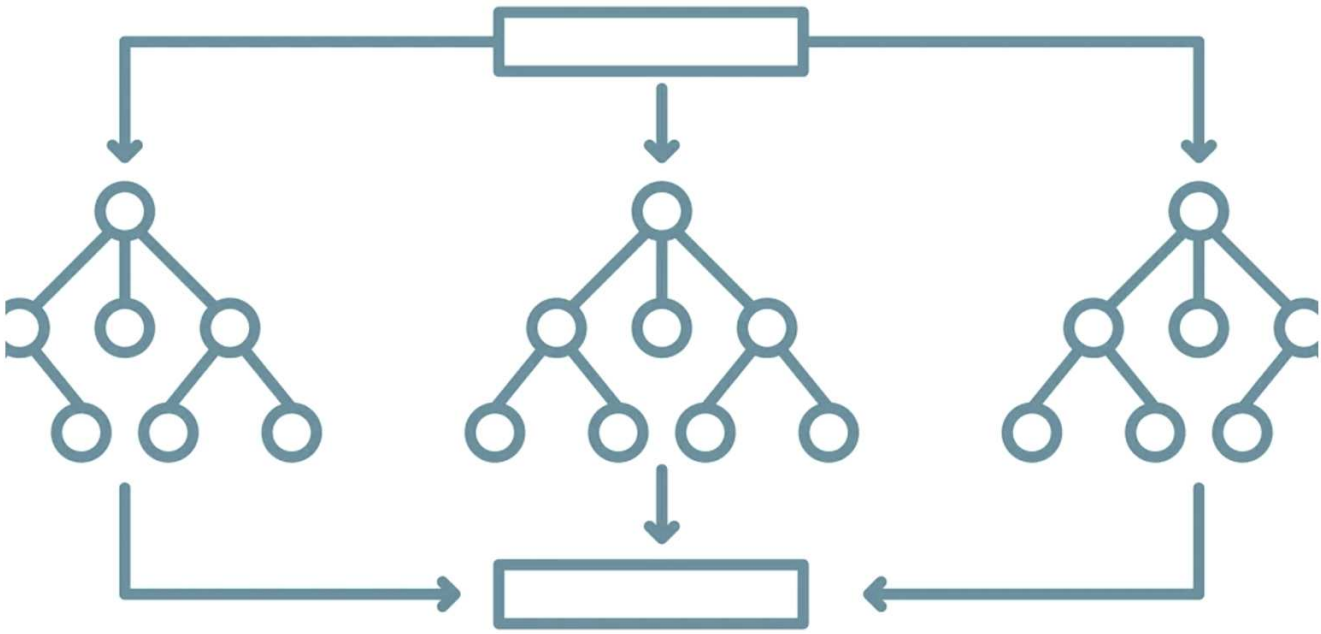
This algorithm is designed to cluster raw input, recognize patterns, or interpret sensory data. Despite their multiple advantages, neural networks require significant computational resources. It can get complicated to fit a neural network when there are thousands of observations.

Support Vector Machines



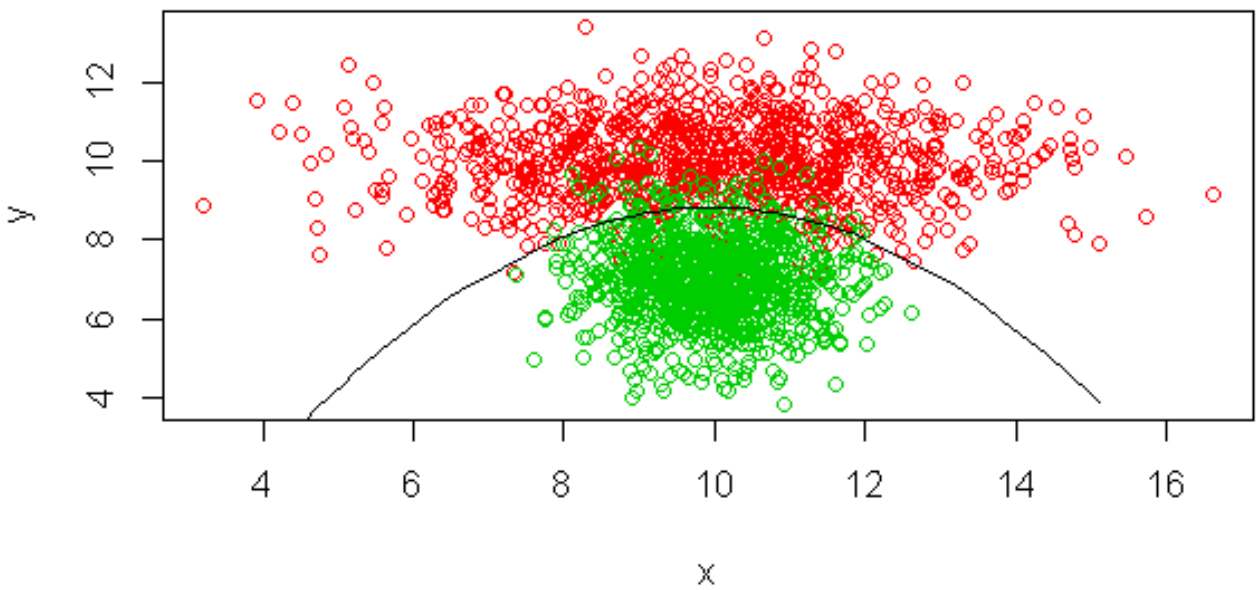
A support vector machine is typically used for classification problems by constructing a hyperplane where the distance between two classes of data points is at its maximum. This hyperplane is known as the decision boundary, separating the classes of data points (e.g., oranges vs. apples) on either side of the plane.

Random Forest



Random forest is another flexible supervised machine learning algorithm used for both classification and regression purposes. It is an ensemble learning algorithm that combines multiple decision trees to improve accuracy and reduce overfitting.

Naive Bayes



Naive Bayes assumes that the features (input variables) are conditionally independent of each other given the class label. This is a "naive" assumption because in reality, features may be correlated with each other. The three main types of Naive Bayes algorithms: Gaussian Naive Bayes, Multinomial Naive Bayes and Bernoulli Naive Bayes

Evaluating a Binary Classification Model

DATARANCH.org

VISUALIZE | ANALYZE | CAPITALIZE

Key Concepts

For example, in a medical diagnosis scenario, **True Positive (TP)** is when the patient is diseased and the model predicts "diseased"

False Positive (FP) or Type 1 Error is when the patient is healthy but the model predicts "diseased"

True Negative (TN) is when the patient is healthy and the model predicts "healthy"

False Negative (FN) or Type 2 Error is when the patient is diseased and the model predicts "healthy"

Impact of False Negatives and False Positives

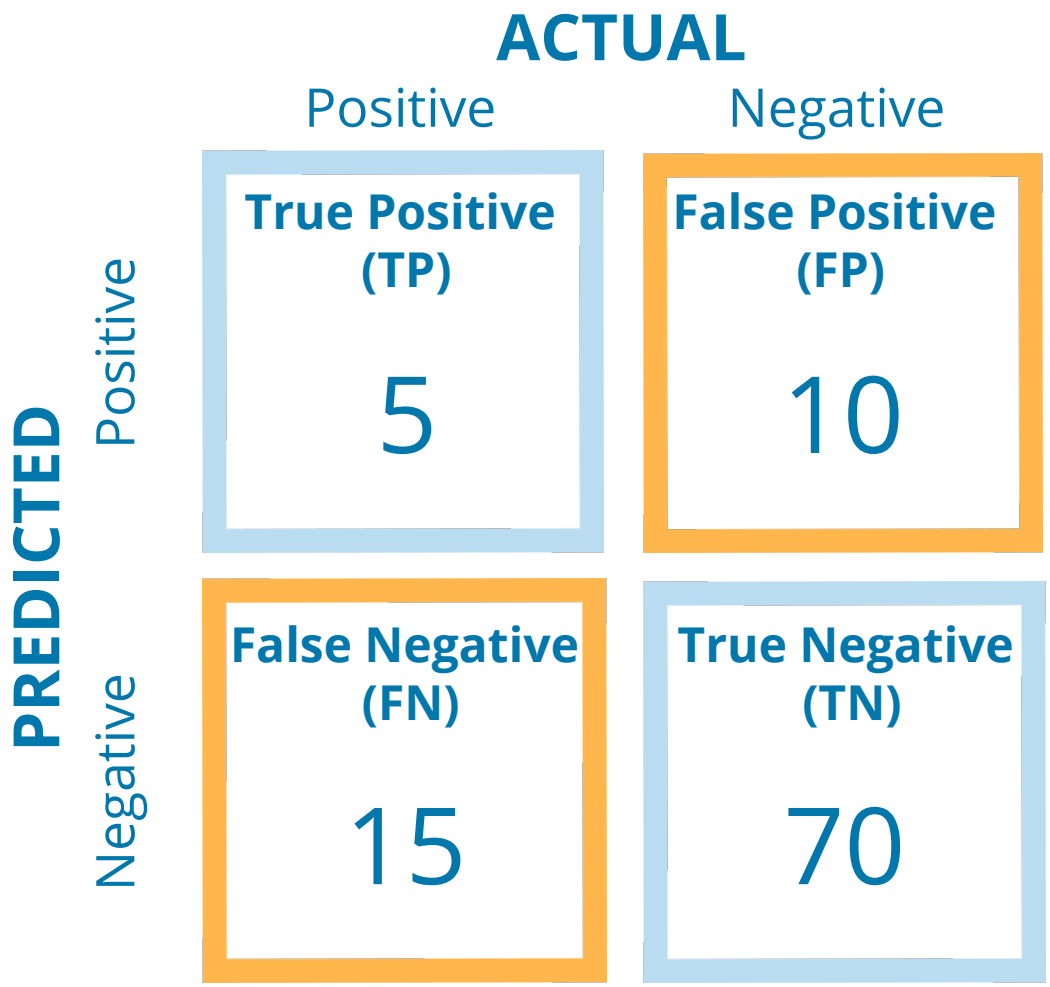
False negatives and false positives can have different impacts depending on the specific problem and context of the classification model.

In a medical diagnosis scenario, a **false negative** can result in a patient not receiving the necessary treatment for a disease, leading to a worsened health condition.

In airport security screening, a **false positive** result for a potential threat can result in unnecessary delays and inconvenience for the passengers.

Confusion Matrix

PREDICTED	ACTUAL: Positive	ACTUAL: Negative
Positive	**True Positive (TP)** Correctly predicts a diseased patient as diseased	**False Positive (FP)** Incorrectly predicts a healthy patient as diseased
Negative	**False Negative (FN)** Incorrectly predicts a diseased patient as healthy	**True Negative (TN)** Correctly predicts a healthy patient as healthy



$$\textbf{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} = 0.75$$

$$\textbf{Recall} = \frac{TP}{TP + FN} = 0.25$$

$$\textbf{Precision} = \frac{TP}{TP + FP} = 0.33$$

$$\textbf{F1 Score} = 2 \times \frac{Precision * Recall}{Precision + Recall} = 0.28$$

DATARANCH.org
VISUALIZE | ANALYZE | CAPITALIZE

Accuracy

$$\textbf{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

$$\textbf{Accuracy} = \frac{\text{Number of correct anwsers}}{\text{Total number of anwsers}}$$

When we want to analyze the performance of a binary classifier, the most common and accessible metric is the accuracy. It tells us how many times our model has correctly classified an item in our dataset with respect to the total.

it is **not recommended** to use accuracy as an evaluation metric when we are working with an unbalanced dataset.

Recall or Sensitivity

$$\textbf{Recall} = \frac{TP}{TP + FN}$$

Recall is also called sensitivity because as recall increases, our model becomes less and less accurate and also classifies negative classes as positive.

E.g. In the case of tumor detection, we want our model to have high recall, as we want to be sure that every single example considered positive by the model is subjected to human inspection. We don't want a malignant tumor to go unnoticed, and we will gladly accept false positives.

Precision or Specificity

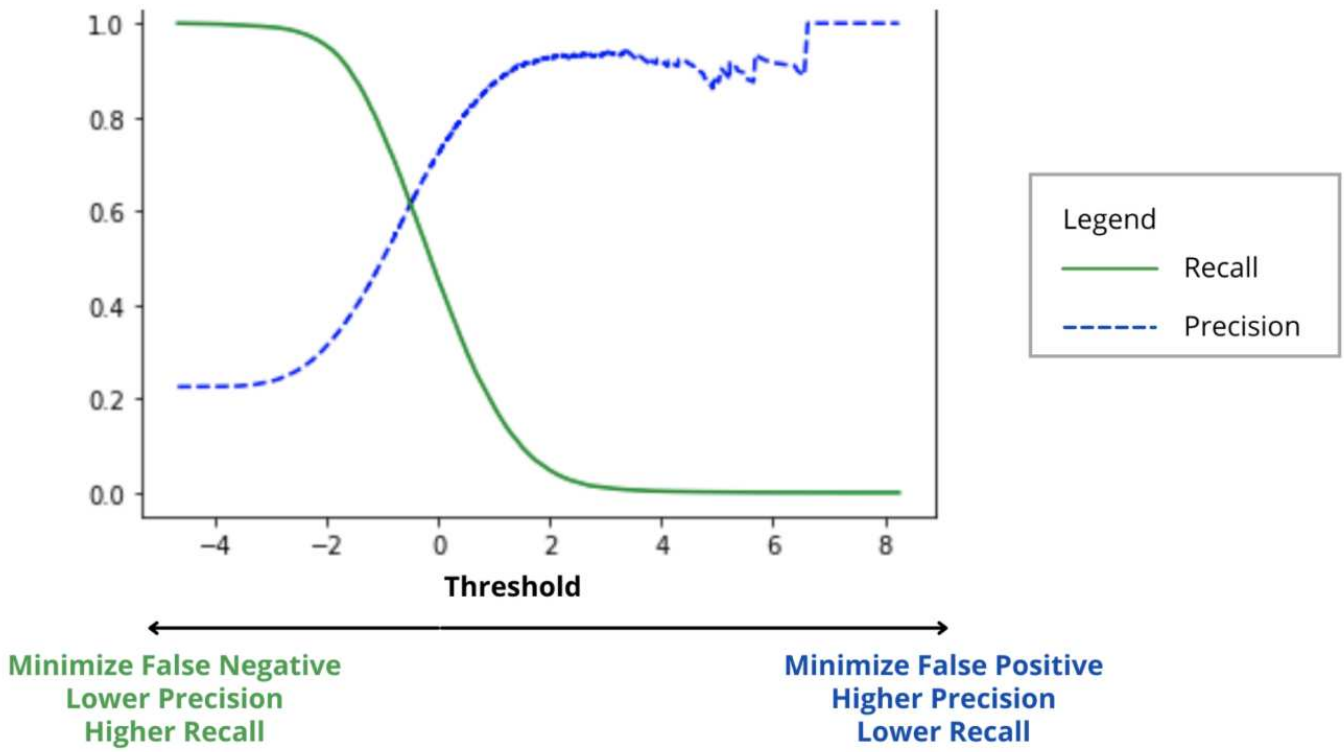
$$\textbf{Precision} = \frac{TP}{TP + FP}$$

Precision is just the accuracy calculated only for positive classes. It is also called specificity since it defines how sensitive an instrument is when there is the signal to be recognized. In fact, the metric tells us how often we are correct when we classify a class as positive.

A high precision model is conservative: it doesn't always recognize the class correctly, but when it does, we can be assured that its answer is correct.

A high recall model is liberal: it recognizes a class much more often, but in doing so it tends to include a lot of noise as well (false positives).

Precision / Recall Trade-off



Both precision and recall range from 0 to 1. As a general rule of thumb, the closer to 1, the better the model is. Unfortunately, you can't have the best of both worlds because increasing precision would cause recall to drop and vice versa.

F1 Score

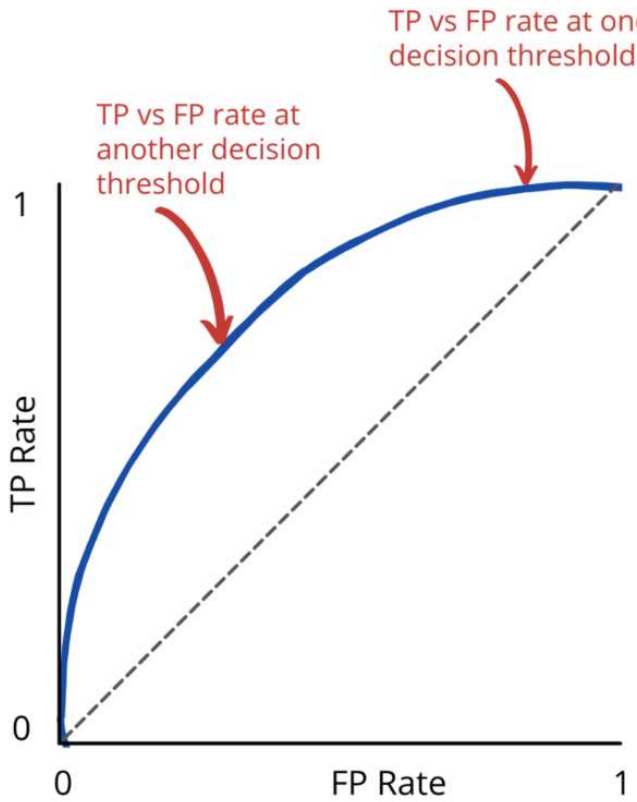
$$\textbf{F1 Score} = 2 \times \frac{\text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}}$$

F1 score combines precision and recall into one metric.

This is the harmonic mean of precision and recall, and is probably the most used metric for evaluating binary classification models.

If our F1 score increases, it means that our model has increased performance for accuracy, recall or both.

ROC Curve



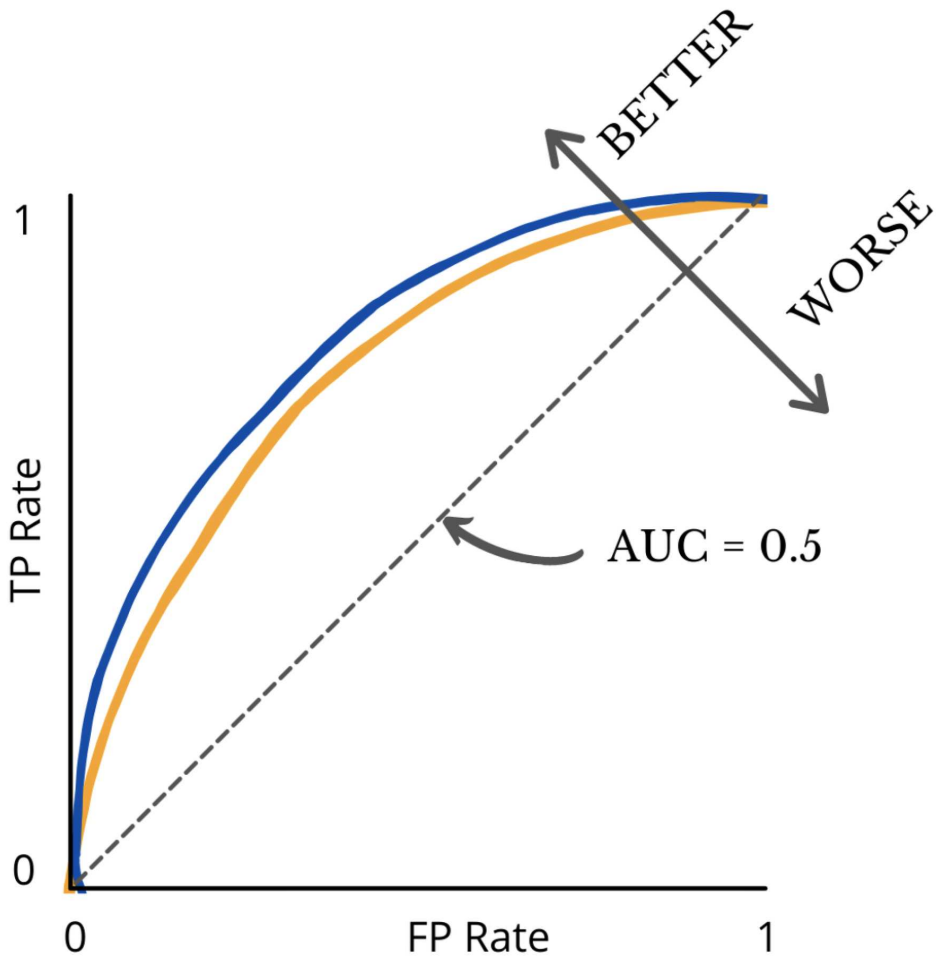
$$\text{TP Rate} = \frac{\text{TP}}{\text{TP + FN}}$$

$$\text{FP Rate} = \frac{\text{FP}}{\text{FP + TN}}$$

Lowering the decision threshold classifies more items as positive, thus increasing both FP and TP. Therefore, TP Rate and FP Rate increases.

A Receiver Operating Characteristic (ROC) curve is a plot of the True Positive Rate (TPR) against the False Positive Rate (FPR) for different classification thresholds. Generally, the closer the ROC curve is to the upper left corner, the better performance the model has.

ROC AUC



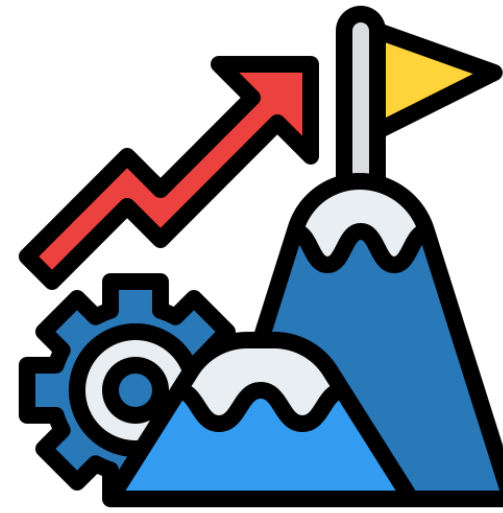
The area under the ROC curve (AUC) is a single scalar value that measures the overall performance of the model. The AUC ranges from 0 to 1, with a higher value indicating better performance. An AUC of 0.5 indicates a random guess, while an AUC of 1.0 indicates perfect classification.

Challenges in Binary Classification Models

DATARANCH.org

VISUALIZE | ANALYZE | CAPITALIZE

Challenges

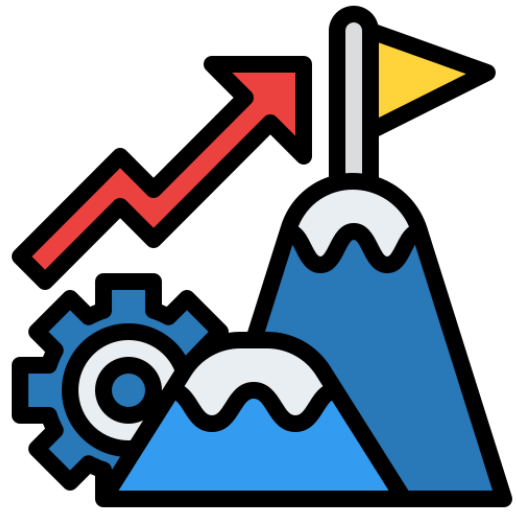


Imbalanced Classes: In many real-world scenarios, the positive and negative classes are not equally represented in the dataset. When one class has significantly more instances than the other, it can lead to biased models that have poor predictive performance on the minority class.

Overfitting: Overfitting occurs when the model is too complex and fits the noise in the training data instead of the underlying patterns. This can result in poor generalization performance and reduced predictive accuracy on new data.

Label Noise: In some cases, the labels in the training data may be noisy or incorrect, which can adversely affect the model's performance.

Challenges



Feature Selection: The performance of binary classification models can depend heavily on the quality and relevance of the input features. Feature selection can be challenging when dealing with high-dimensional data.

Model Interpretability: Binary classification models can be highly complex and difficult to interpret, especially when using non-linear or deep learning models. Interpretability is important in many applications, such as healthcare, where the model's predictions must be explained to clinicians and patients.

Scalability: Binary classification models can require significant computational resources and memory, especially when dealing with large datasets or complex models.

Follow **#DataRanch** on LinkedIn for more...

DATARANCH.org
VISUALIZE | ANALYZE | CAPITALIZE

Follow **#DataRanch** on LinkedIn for more...

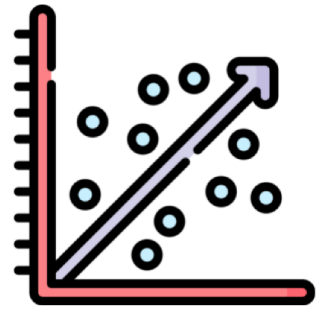
What is Supervised Learning?

DATARANCH.org

What is Unsupervised Learning?

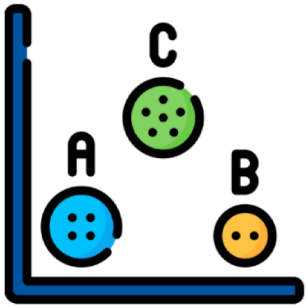
DATARANCH.org

Regression Analysis



DATARANCH.org

Clustering

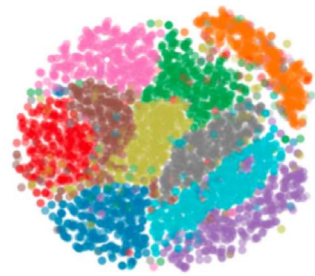


DATARANCH.org

Principal Component Analysis

DATARANCH.org

t-Distributed Stochastic Neighbour Embedding (t-SNE)



DATARANCH.org

DATARANCH.org
VISUALIZE | ANALYZE | CAPITALIZE

DATARANCH.org

VISUALIZE | ANALYZE | CAPITALIZE

info@dataranch.org

linkedin.com/company/dataranch