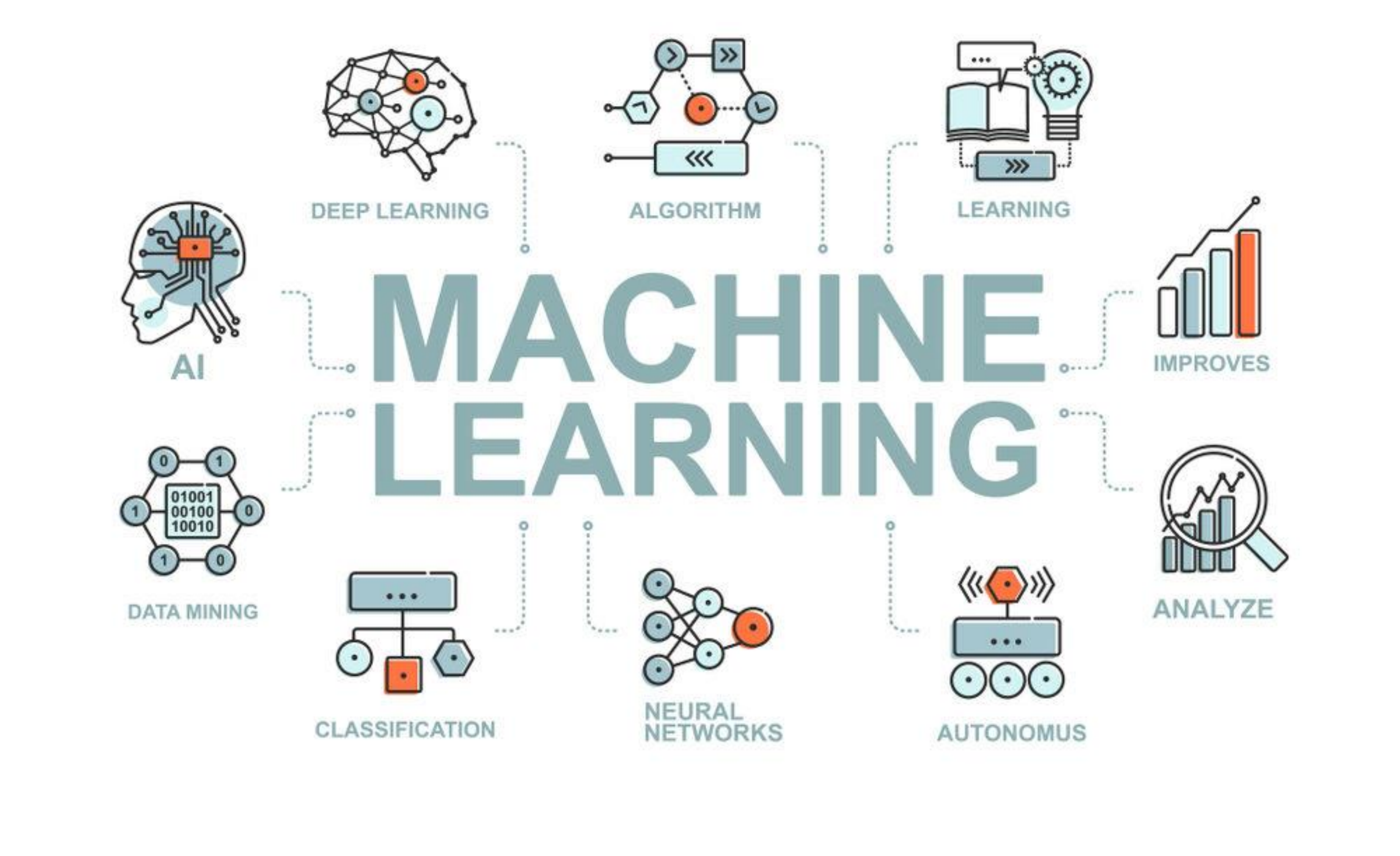
German Jordanian University
الجامعة الألمانية الأردنية

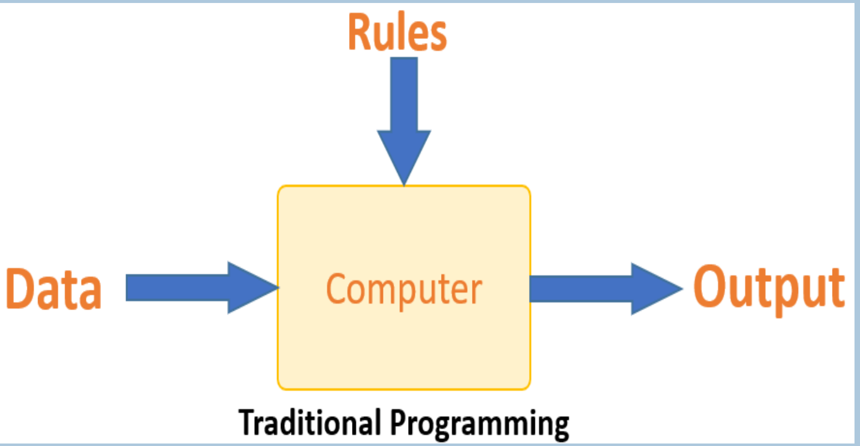
Machine Learning



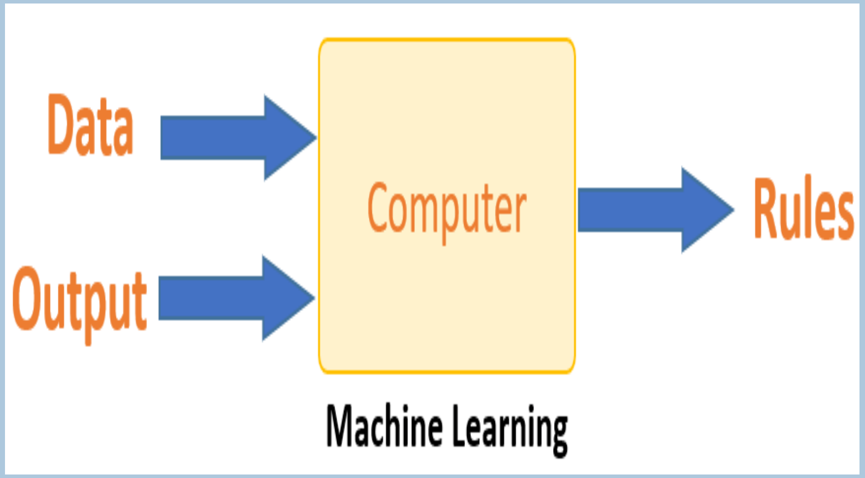
Machine Learning is a system of computer algorithms that can learn from example through self-improvement without being explicitly coded by a programmer. Machine learning is a part of artificial Intelligence which combines data with statistical tools to predict an output which can be used to make actionable insights.
The breakthrough comes with the idea that a machine can singularly learn from the data (i.e., example) to produce accurate results. Machine learning is closely related to data mining and Bayesian predictive modeling. The machine receives data as input and uses an algorithm to formulate answers.

Machine Learning vs. Traditional Programming

Traditional programming differs significantly from machine learning. In traditional programming, a programmer code all the rules in consultation with an expert in the industry for which software is being developed. Each rule is based on a logical foundation; the machine will execute an output following the logical statement. When the system grows complex, more rules need to be written. It can quickly become unsustainable to maintain.



Machine learning is supposed to overcome this issue. The machine learns how the input and output data are correlated and it writes a rule. The programmers do not need to write new rules each time there is new data. The algorithms adapt in response to new data and experiences to improve efficacy over time.



How does Machine Learning Work?

Machine learning is the brain where all the learning takes place. The way the machine learns is similar to the human being. Humans learn from experience. The more we know, the more easily we can predict. By analogy, when we face an unknown situation, the likelihood of success is lower than the known situation. Machines are trained the same. To make an accurate prediction, the machine sees an example. When we give the machine a similar example, it can figure out the outcome. However, like a human, if its feed a previously unseen example, the machine has difficulties to predict.

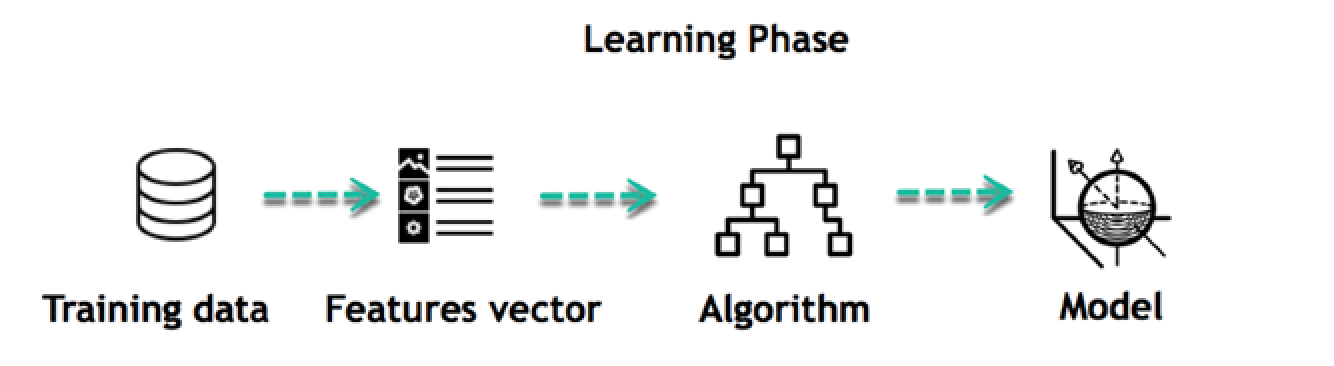
The core objective of machine learning is the **learning** and **inference**. First of all, the machine learns through the discovery of patterns. This discovery is made thanks to the **data**. One crucial part of the data scientist is to choose carefully which data to provide to the machine. The list of attributes used to solve a problem is called a **feature vector.** You can think of a feature vector as a subset of data that is used to tackle a problem.

The machine uses some fancy algorithms to simplify the reality and transform this discovery into a **model**. Therefore, the learning stage is used to describe the data and summarize it into a model.

Learning and Inferring

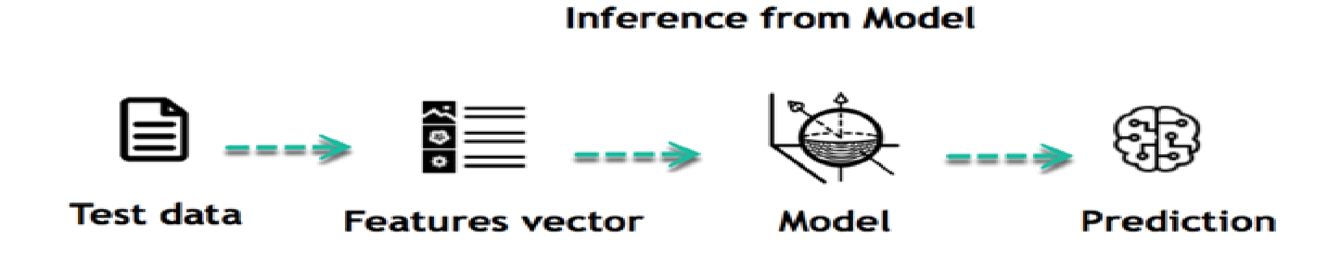
Learning

For instance, the machine is trying to understand the relationship between the wage of an individual and the likelihood to go to a fancy restaurant. It turns out the machine finds a positive relationship between wage and going to a high-end restaurant: This is the model



Inferring

When the model is built, it is possible to test how powerful it is on never-seen-before data. The new data are transformed into a features vector, go through the model and give a prediction. This is all the beautiful part of machine learning. There is no need to update the rules or train again the model. You can use the model previously trained to make inference on new data.

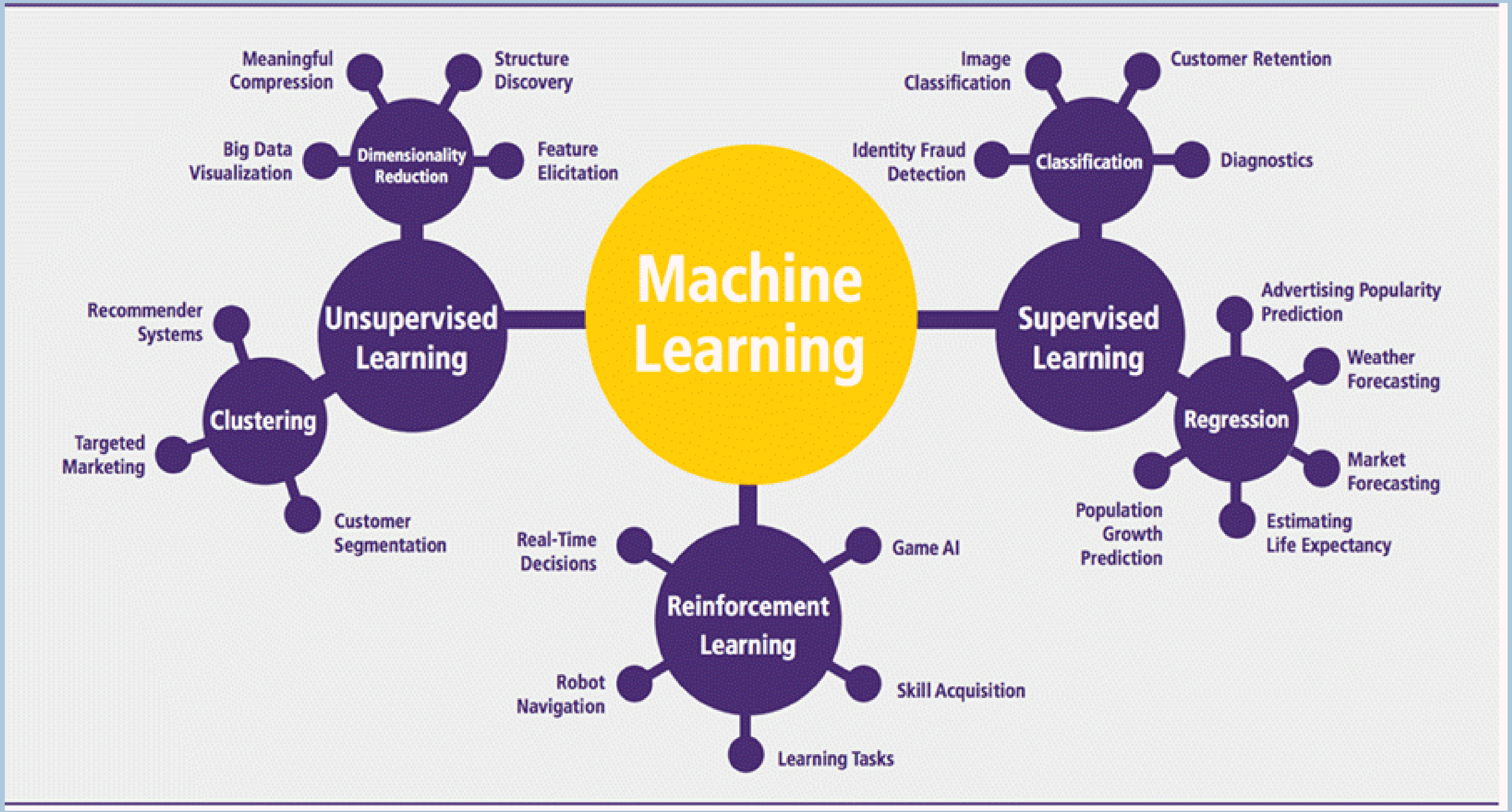


Machine Learning programs

The life of Machine Learning programs is straightforward and can be summarized in the following points:

1. Define a question
2. Collect data
3. Visualize data
4. Train algorithm
5. Test the Algorithm
6. Collect feedback
7. Refine the algorithm
8. Loop 4-7 until the results are satisfying
9. Use the model to make a prediction

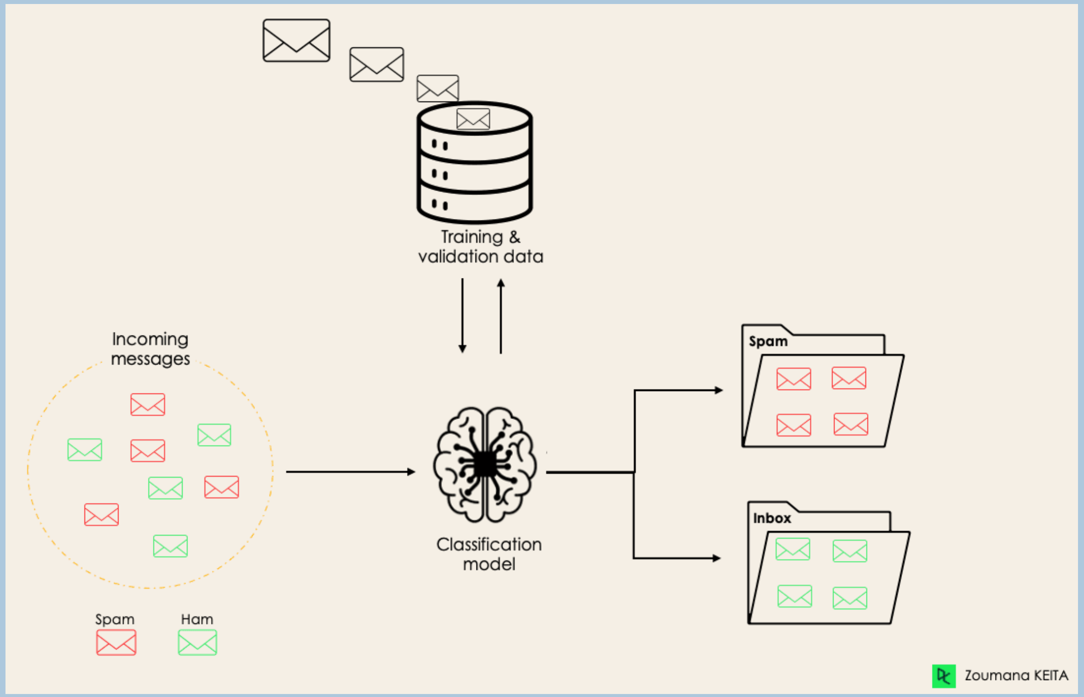
Machine Learning Algorithms and Where they are Used?



Classification

Classification is a supervised machine learning method where the model tries to predict the correct label of a given input data. In classification, the model is fully trained using the training data, and then it is evaluated on test data before being used to perform prediction on new unseen data.

For instance, an algorithm can learn to predict whether a given email is spam or ham (no spam), as illustrated below.



Learners in Classification Problems

Lazy Learners Vs. Eager Learners

There are two types of learners in machine learning classification: lazy and eager learners.

Eager learners are machine learning algorithms that first build a model from the training dataset before making any prediction on future datasets. They spend more time during the training process because of their eagerness to have a better generalization during the training from learning the weights, but they require less time to make predictions.

Most machine learning algorithms are eager learners, and below are some examples:

- Logistic Regression.
- Support Vector Machine.
- Decision Trees.
- Artificial Neural Networks.

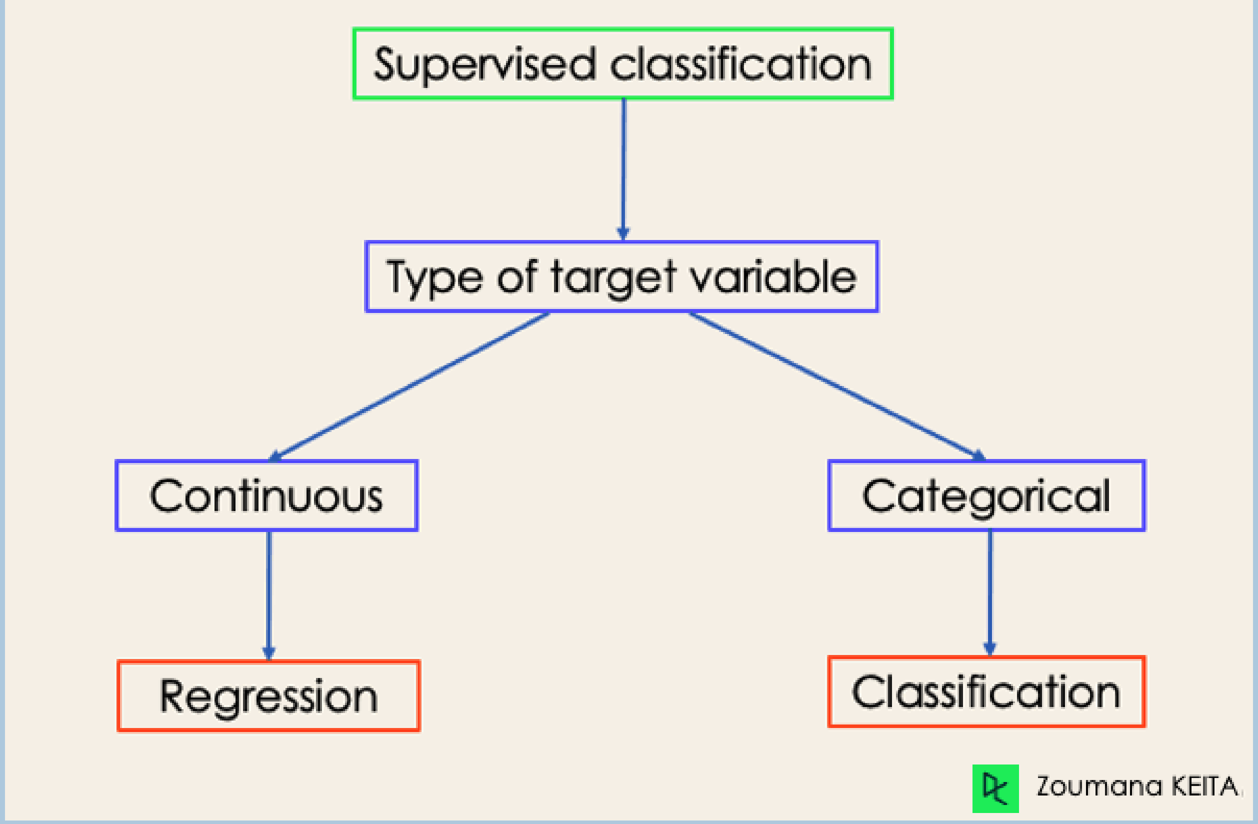
Lazy learners or instance-based learners, on the other hand, do not create any model immediately from the training data, and this is where the lazy aspect comes from. They just memorize the training data, and each time there is a need to make a prediction, they search for the nearest neighbor from the whole training data, which makes them very slow during prediction. Some examples of this kind are:

- K-Nearest Neighbor.

Classification and Regression

- The prediction task is a ***classification*** when the target variable is discrete. An application is the identification of the underlying sentiment of a piece of text.

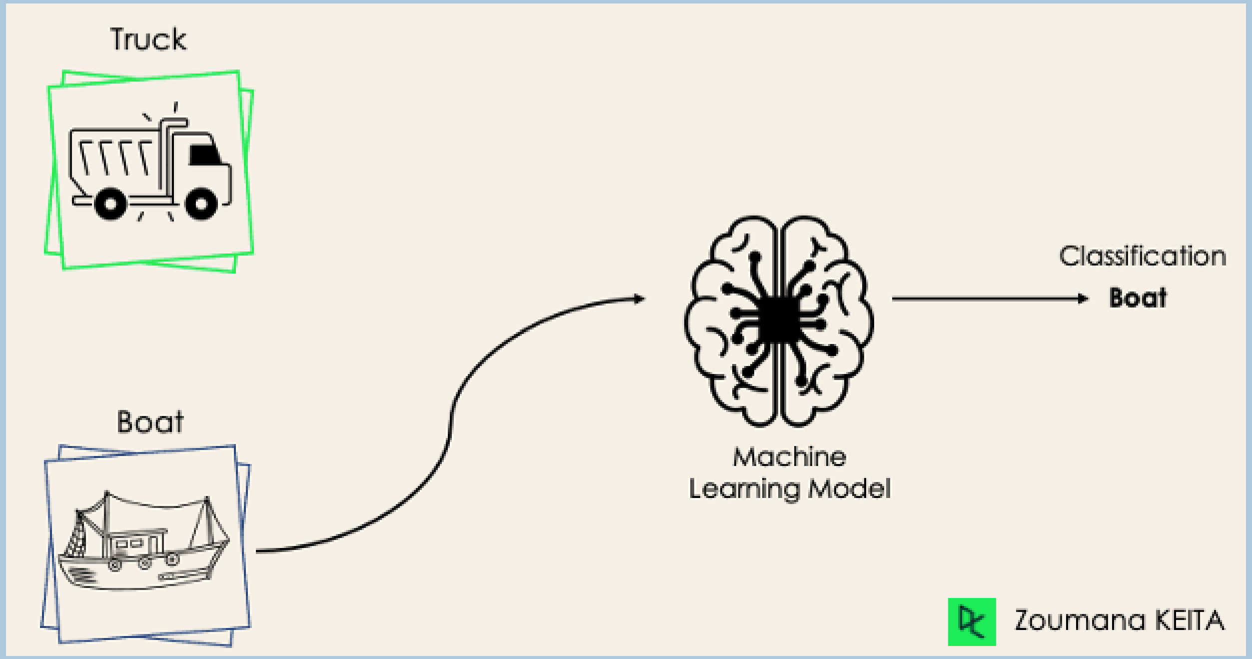
- The prediction task is a ***regression*** when the target variable is continuous. An example can be the prediction of the salary of a person given their education degree, previous work experience, geographical location, and level of seniority.



Types of Classification Tasks

Binary Classification

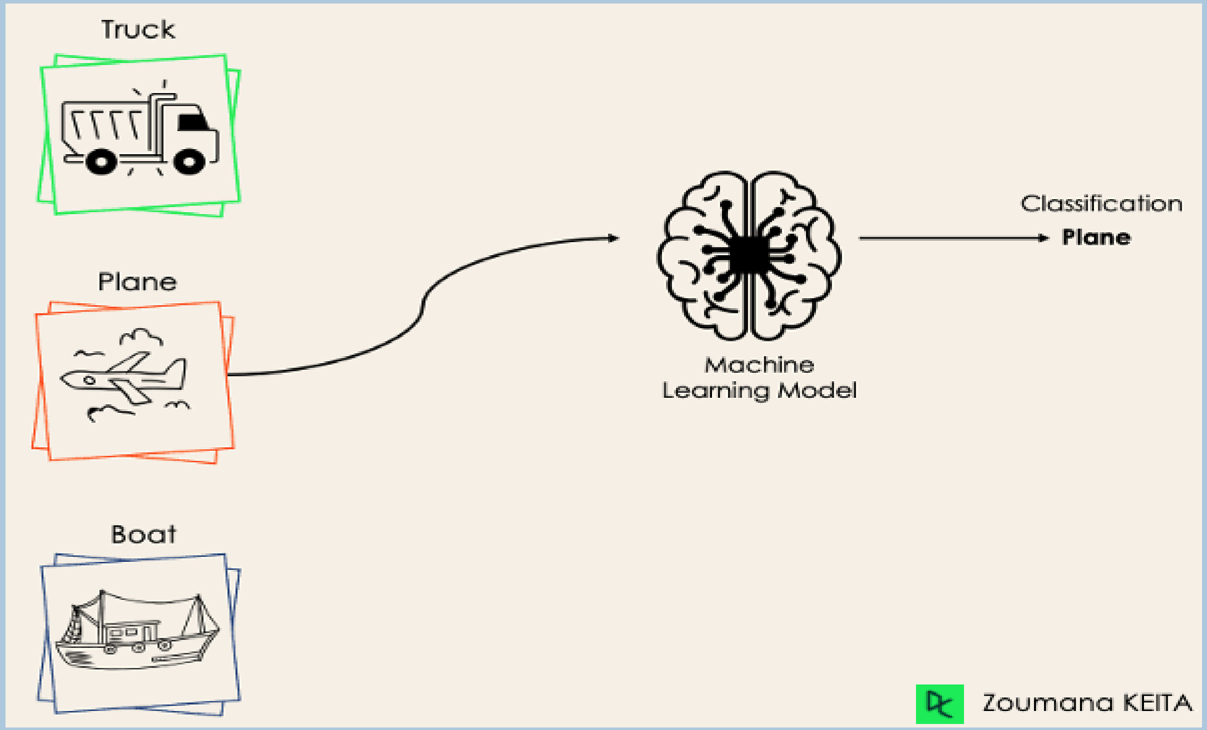
In a binary classification task, the goal is to classify the input data into two mutually exclusive categories. The training data in such a situation is labeled in a binary format: true and false; positive and negative; O and 1; spam and not spam, etc. depending on the problem being tackled. For instance, we might want to detect whether a given image is a truck or a boat.



Types of Classification Tasks

Multi-Class Classification

The multi-class classification, on the other hand, has at least two mutually exclusive class labels, where the goal is to predict to which class a given input example belongs to. In the following case, the model correctly classified the image to be a plane.



Metrics to Evaluate Machine Learning Classification Algorithms

We have an idea of the different types of classification models, it is crucial to choose the right evaluation metrics for those models. The most commonly used metrics: accuracy, precision, recall, F1 score, and area under the ROC (Receiver Operating Characteristic) curve and AUC (Area Under the Curve).

A bit of context
Imagine you are a healthcare startup, and want an AI assistant able to predict whether a given patient has a heart disease or not based on its health record. This is a binary classification problem where the model will predict

- 1, True or Yes if the patient has heart disease
- 0, False or No otherwise

1 Confusion matrix

A 2X2 matrix that nicely summarizes the number of correct predictions of the model. It also helps in computing different other performance metrics.

Predicti / Reality	Yes	No
Yes	True Positives (TP)	False Negatives (FN)
No	False Positives(FP)	True Negatives (TN)

Type I Error

Type II Error

Type I & II Errors can be used interchangeably when referring to False Positives and False negatives respectively

2 Accuracy

We get <u>accuracy</u> by answering this question: **"out of the predictions made by the model, what percentage is correct?"**

$$Accuracy = \frac{TP + TN}{Total\ number\ observation}$$

Metrics to Evaluate Machine Learning Classification Algorithms

3 Precision

We get *precision* by answering this question: **"out of all the YES predictions, how many of them were correct?"**

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

4 Recall / Sensitivity

It aims to answer this question: **"how good was the model at predicting real Yes events?"**, which can be considered as the flip of the precision.

$$Sensitivity = \frac{TP}{TP + FN}$$

5 Recall / Specificity

It aims to answer this question: **"how good was the model at predicting real No events?"**.

$$Specificity = \frac{TN}{TN + FP}$$

6 F1 Score

Sometimes used when dealing with imbalanced data set, meaning that there are more of one class/label than there are of the other. It corresponds to the harmonic mean of the precision and recall.

$$F1\ Score = 2 \cdot \frac{Precision \times Recall}{Precision + Recall}$$

الجامعة الألمانية الأردنية
German Jordanian University

Metrics to Evaluate Machine Learning Classification Algorithms

7 AUC – ROC Curve

AUC– ROC generates probability values instead of binary 0/1 values. It should be used when your data set is roughly balanced.

Using ROC for imbalanced data sets lead to incorrect interpretation.

ROC curves provide good overview of trade-off between the TP rate and FP rate for binary classifier using different probability thresholds.

- A value below 0.5 indicates a poor classifier
- A value of 0.5 means random classifier
- Value over 0.7 corresponds to a good classifier
- 0.8 indicates a strong classifier
- We have 1 when the classifier perfectly predicts everything.

Strategies to choose the right metric

Choose accuracy
- The cost of FP and FN are roughly equal.
- The benefit of TP and TN are roughly equal.

Choose Precision
- The cost of FP is much higher than a FN.
- The benefit of a TP is much higher than a TN.

Choose recall
- The cost of FN is much higher than a FP.
- The cost of a TN is much higher than a TP.

ROC AUC & Precision – Recall curves
- Use ROC when the dealing with balanced data sets.
- Use precision-recall for imbalanced data sets.