Big Data Hadoop and Spark Developer

DATA AND
ARTIFICIAL INTELLIGENCE

Machine Learning Using Spark ML

simplilearn

Learning Objectives

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

- List the advantages of analytics in spark and its types
- Explain machine learning and its types with their applications
- Explain the relationship between data science and machine learning
- Summarize the flow of supervised learning, unsupervised learning, semi-supervised learning, and reinforcement learning

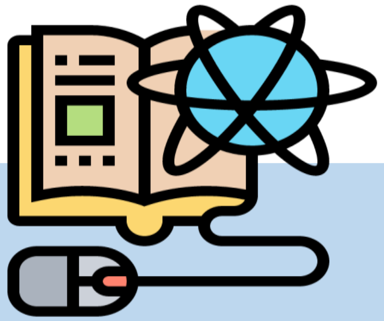
Learning Objectives

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

- Analyze the face detection use cases of machine learning
- List the various types of tools and algorithms provided by Spark ML
- Explain the mechanism of ML Pipeline
- List the various APIs offered by ML pipeline

Analytics in Spark

Apache Spark



- Apache Spark is an open-source unified analytics engine for large-scale data processing.
- It is a multi-language engine for executing data engineering, data science, and machine learning on single-node machines or clusters.

Analytics in Spark

There are various advantages of using spark for analytics. These are:

Types of Analytics

There are three types of analytics.

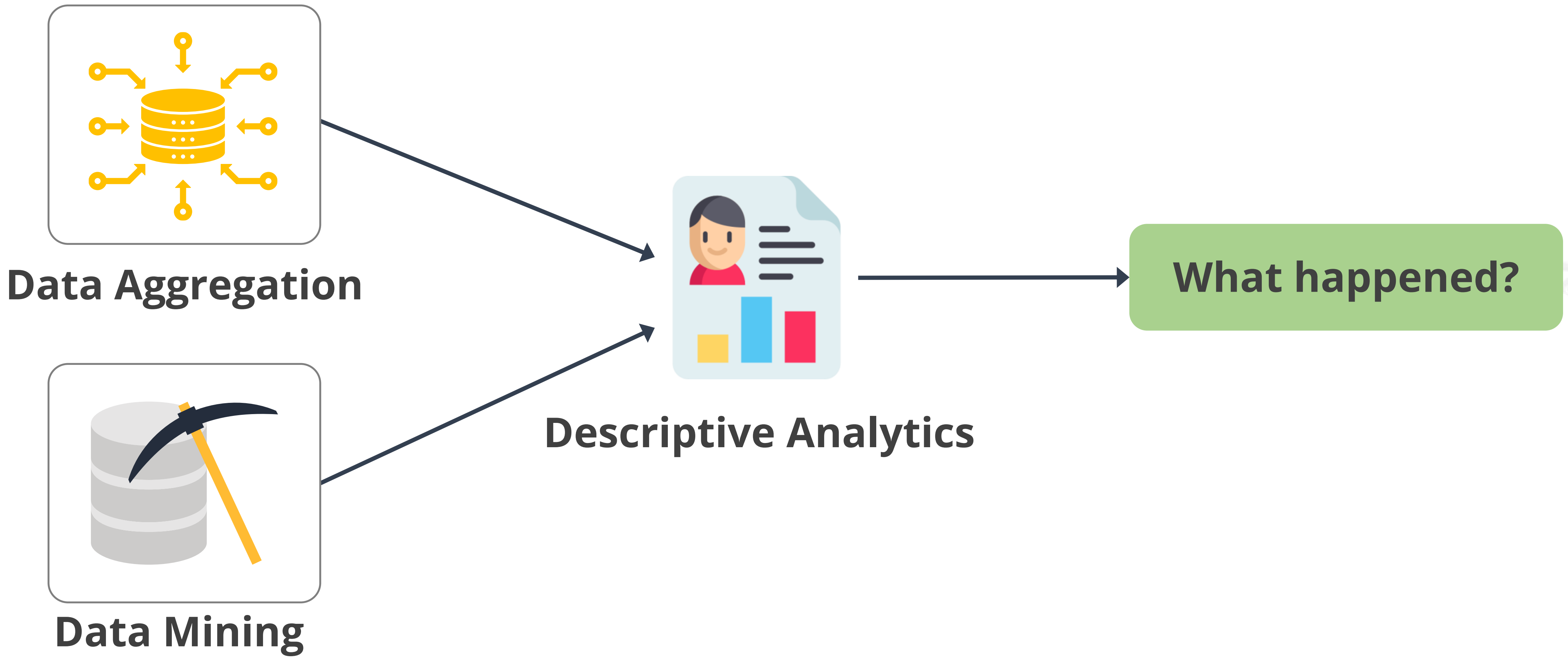
Descriptive Analytics

Predictive Analytics

Prescriptive Analytics

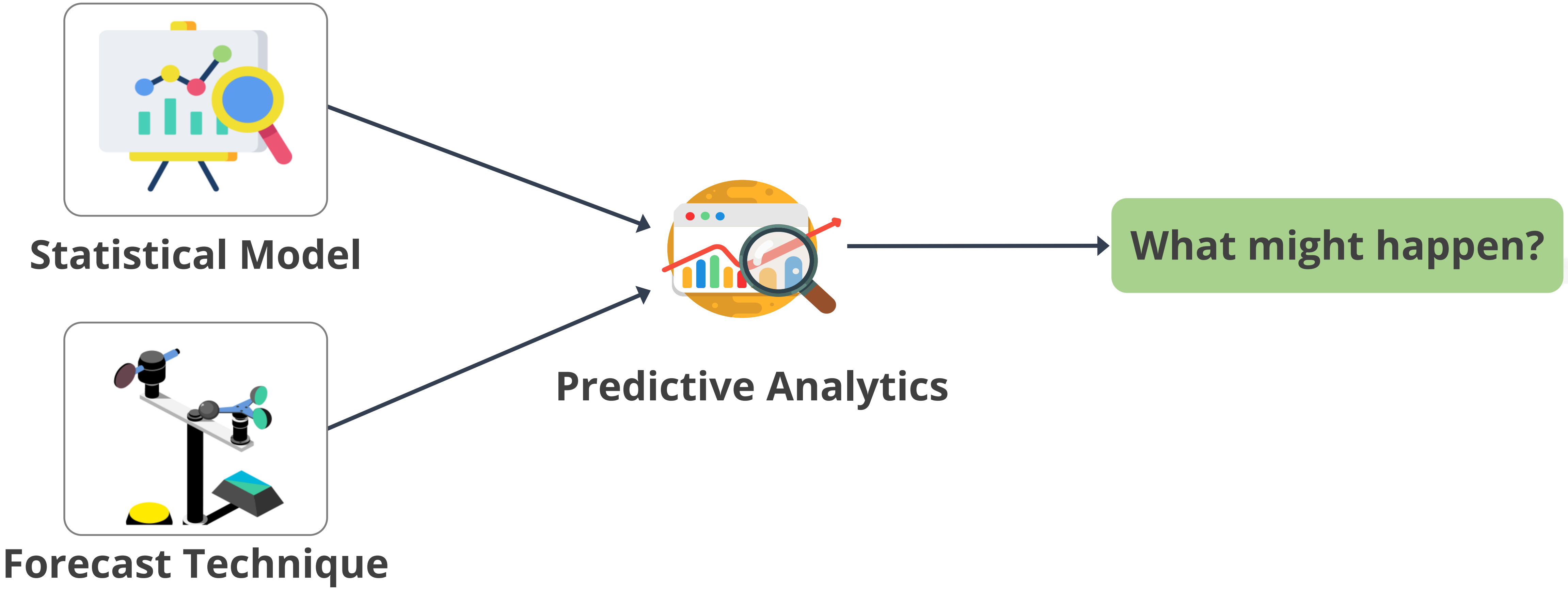
Descriptive Analytics

The type of analytics that describes the past and answers the question: “What happened?”



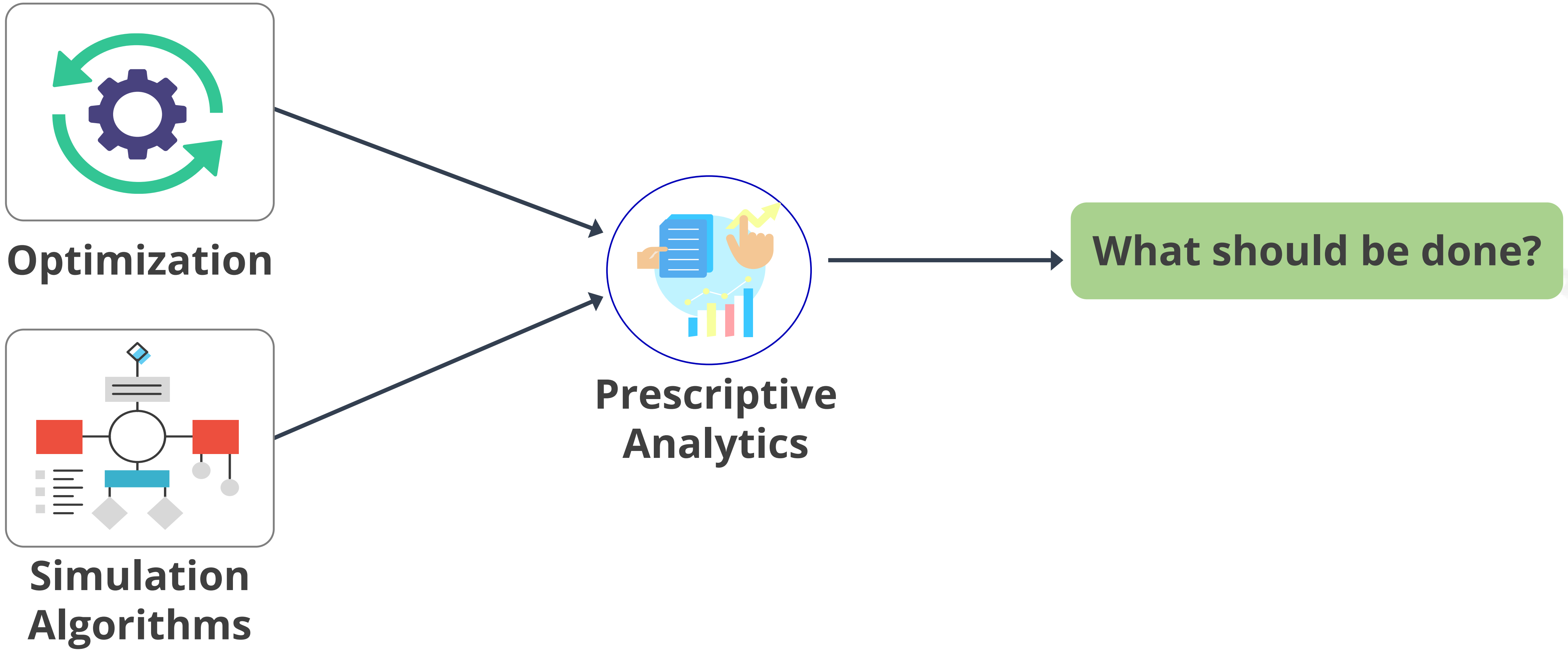
Predictive Analytics

The type of analytics that can predict the future and answers the question: “What might happen?”



Prescriptive Analytics

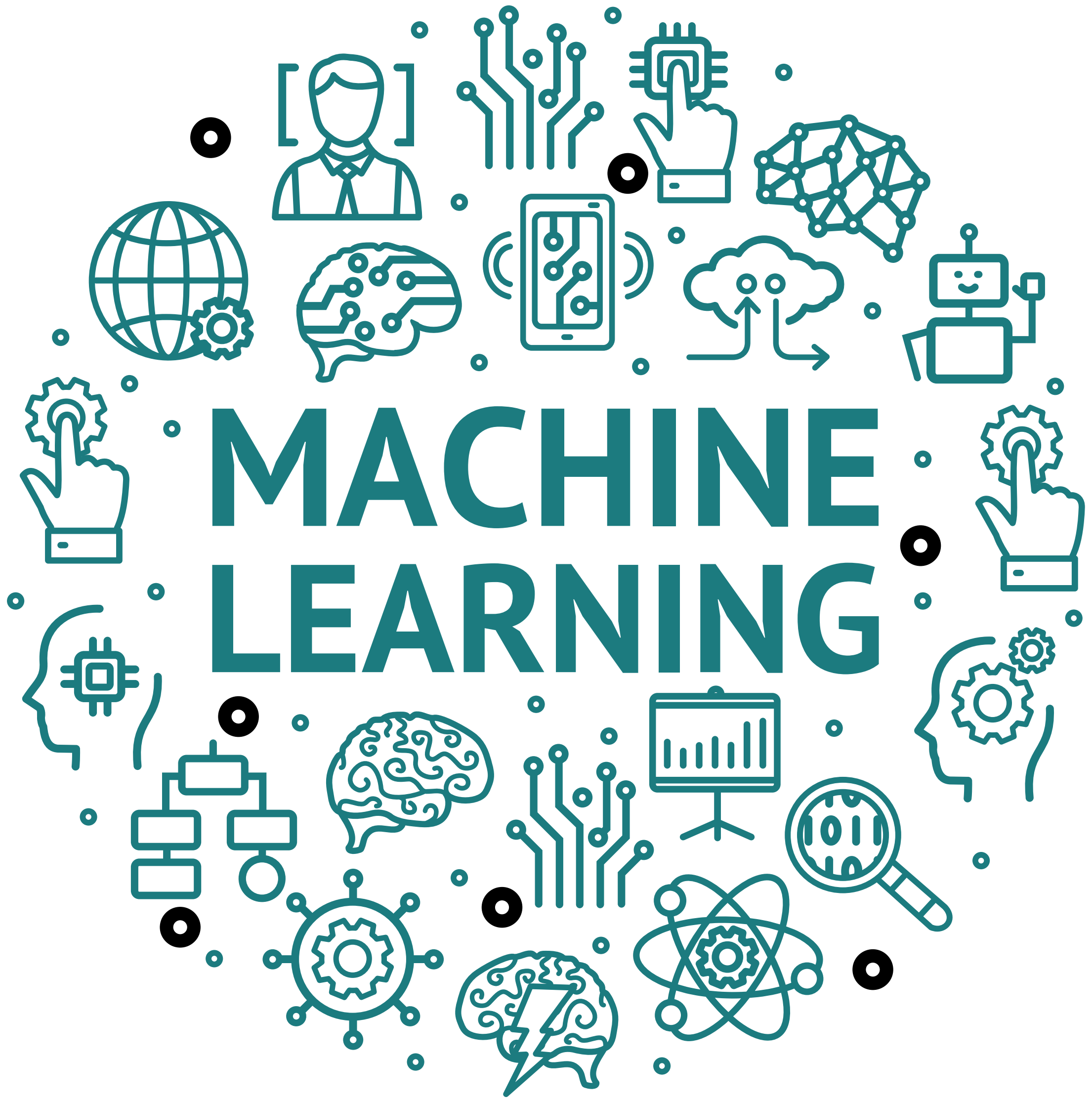
The type of analytics that advises users on possible outcomes and answers the question: "What should be done?"



DATA AND
ARTIFICIAL INTELLIGENCE

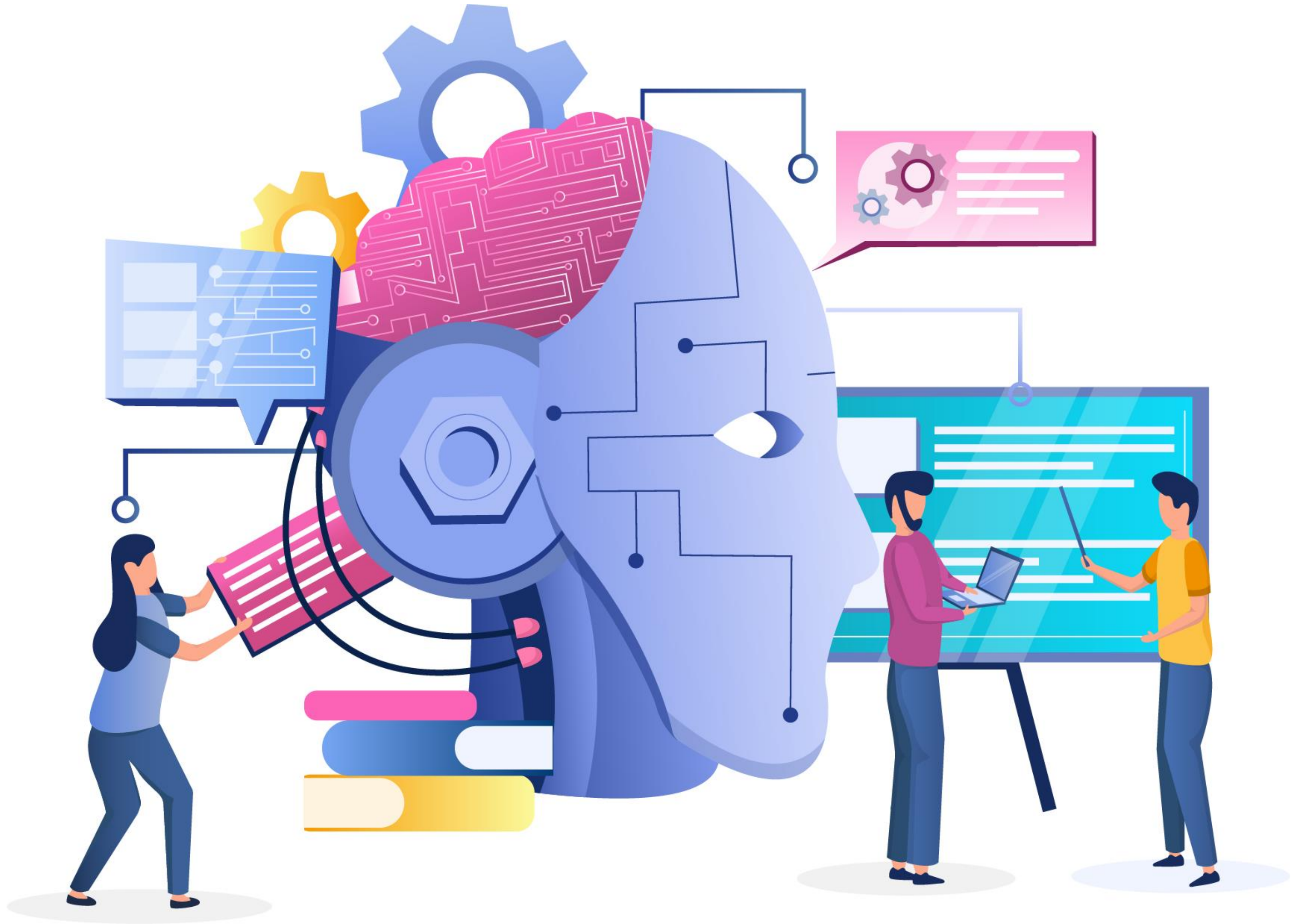
Introduction to Machine Learning

What Is Machine Learning?



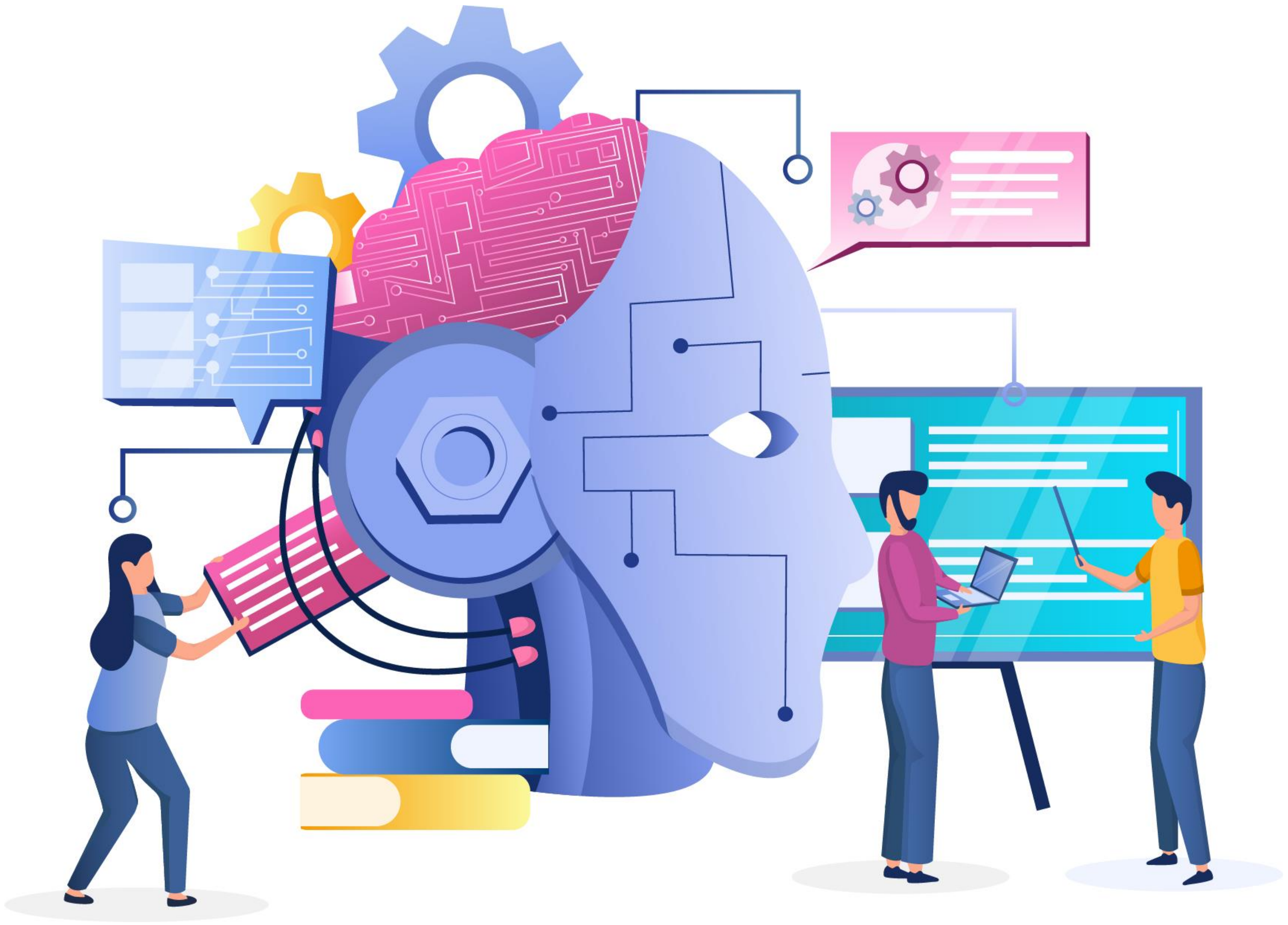
- Machine learning is a subset of artificial intelligence.
- ML allows software applications to become more accurate at predicting outcomes without being explicitly programmed.
- Machine learning algorithms use historical data as input to predict new output values.

Why Machine Learning?



- For many businesses, machine learning has become a crucial competitive differentiation.
- Machine learning is a fundamental aspect of the operations of leading companies such as Facebook and Uber.
- Machine learning is important because it allows businesses to see trends in customer behavior.

Why Machine Learning?

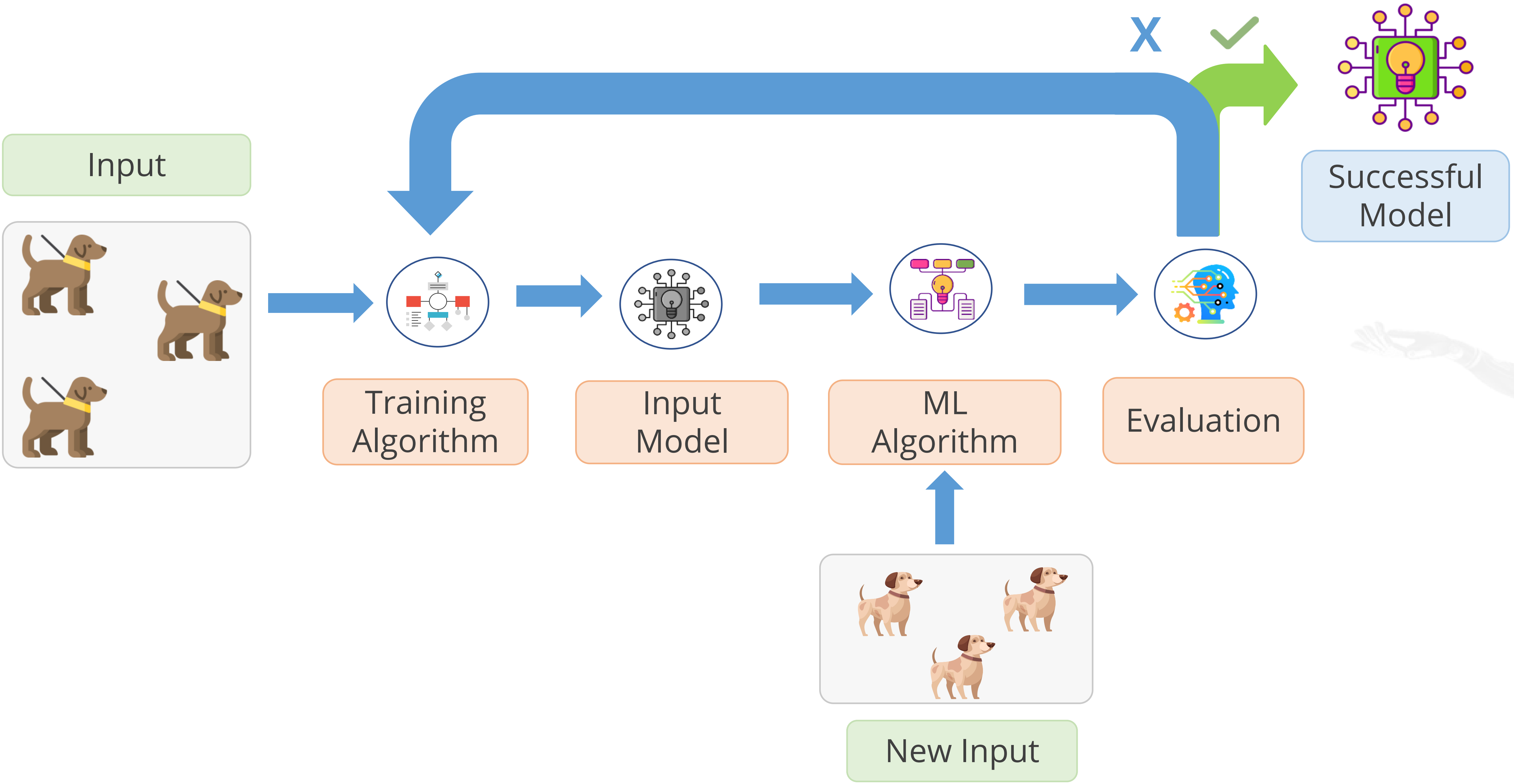


- It aids in the understanding of business operational patterns and the development of new products for businesses.

- Many businesses utilize machine learning in manufacturing to reduce cost, improve quality control, and streamline supply chains.

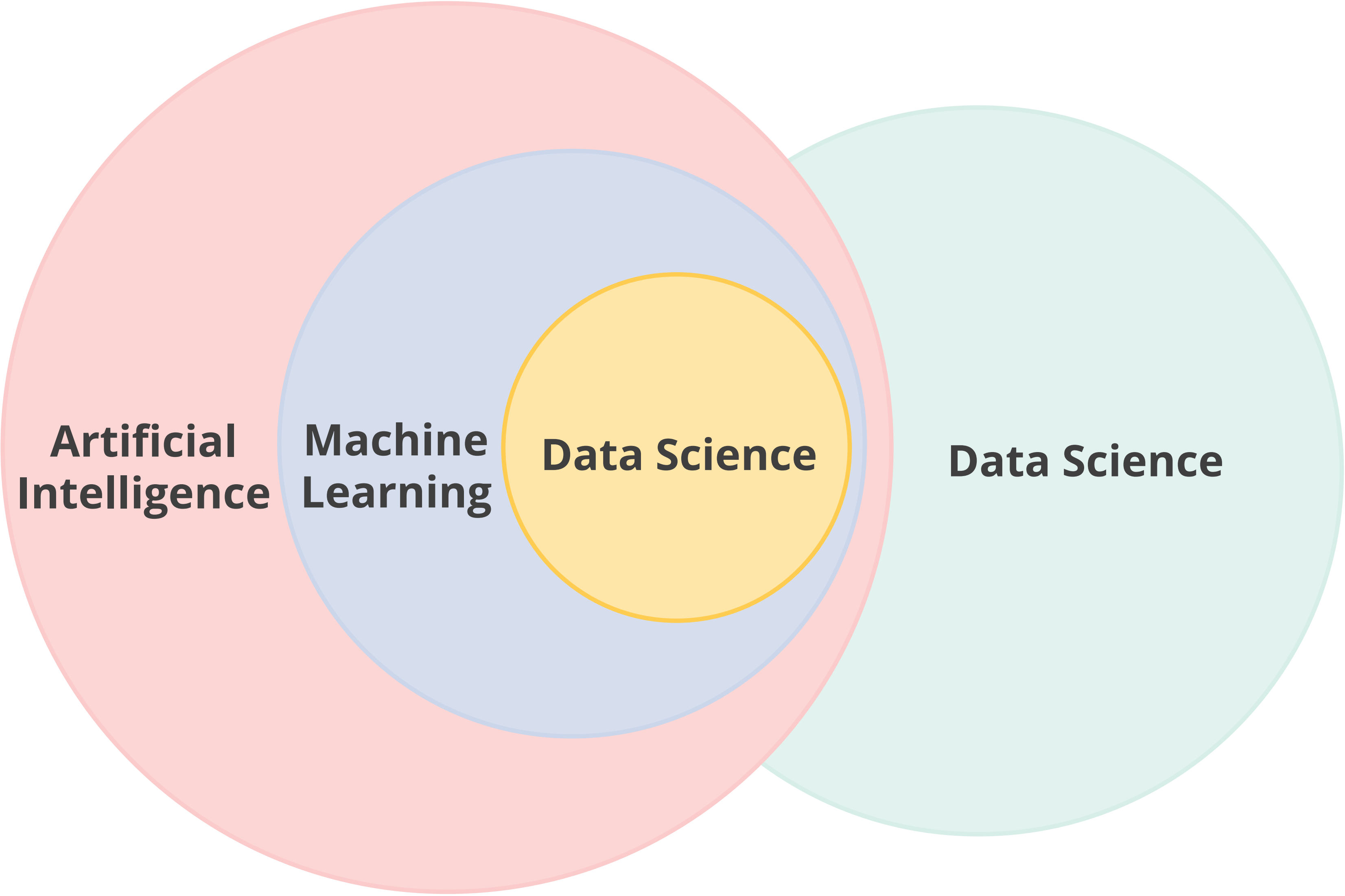
Machine Learning: Process Flow

The machine learning process has several stages which are depicted below:

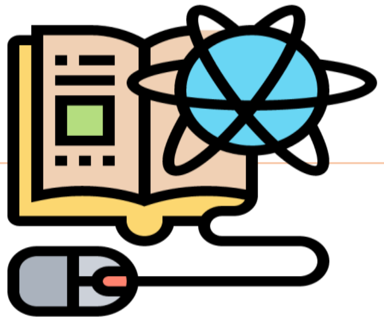


Relationship Between Data Science and Machine Learning

Data Science and Machine Learning go hand in hand.
Data Science aids in the evaluation of data for Machine Learning algorithms.



Large-Scale Machine Learning



Large-scale machine learning requires a vast amount of data with many training features or classes.

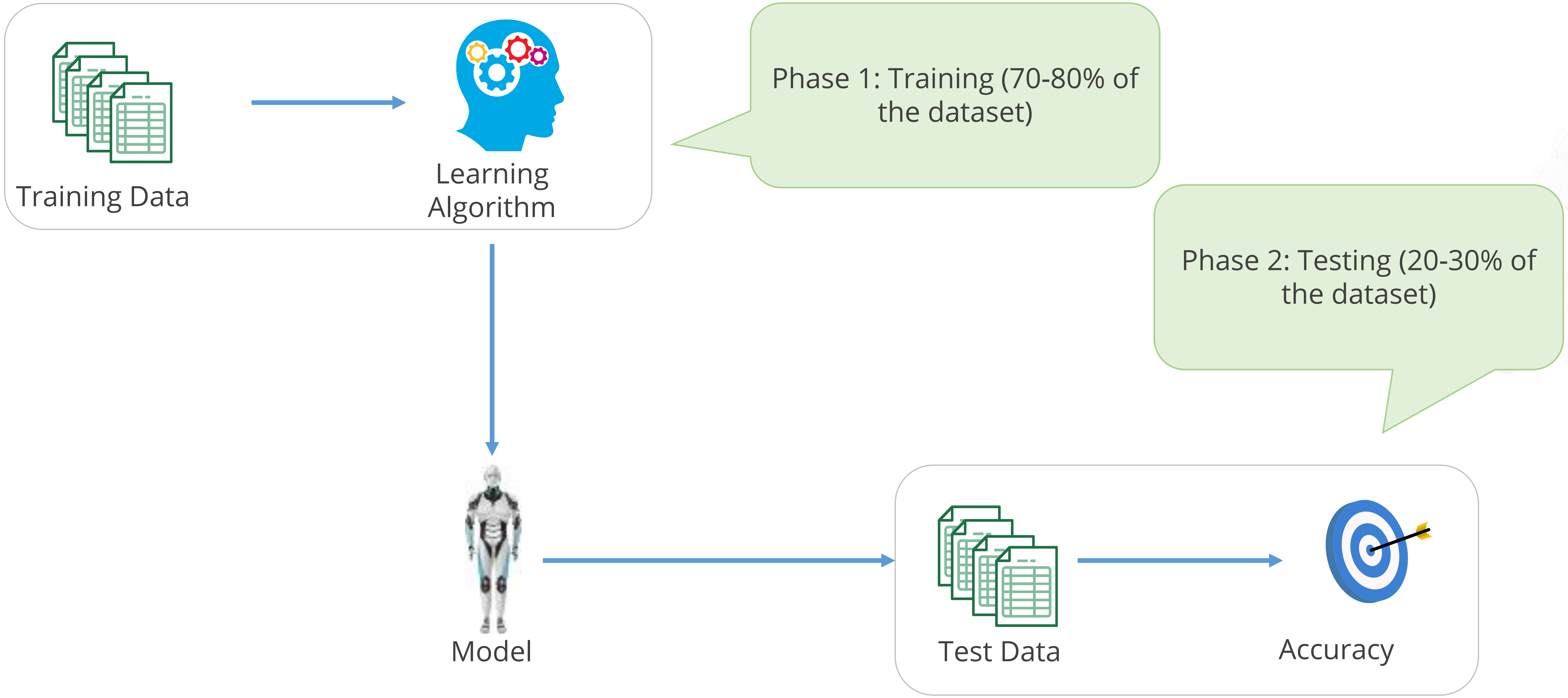
Large-Scale Machine Learning: Tools

There are various large-scale machine learning tools in the market, such as:

Machine Learning Implementation

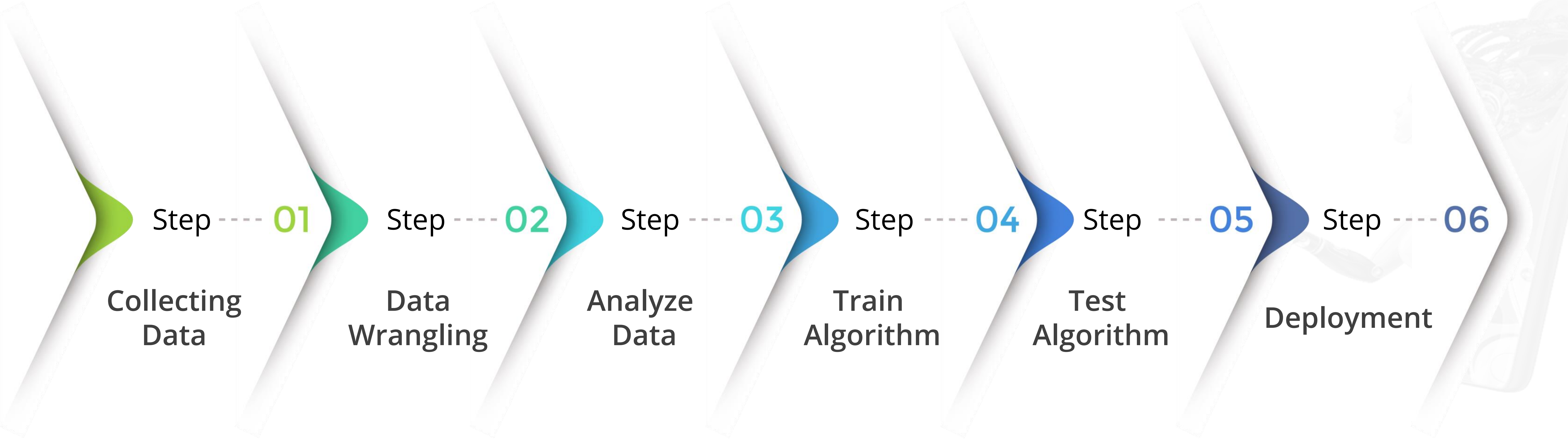
Phases of Machine Learning

Machine learning has several phases which are depicted below:



Phases of Machine Learning

The detailed steps involved in the machine learning phases are:

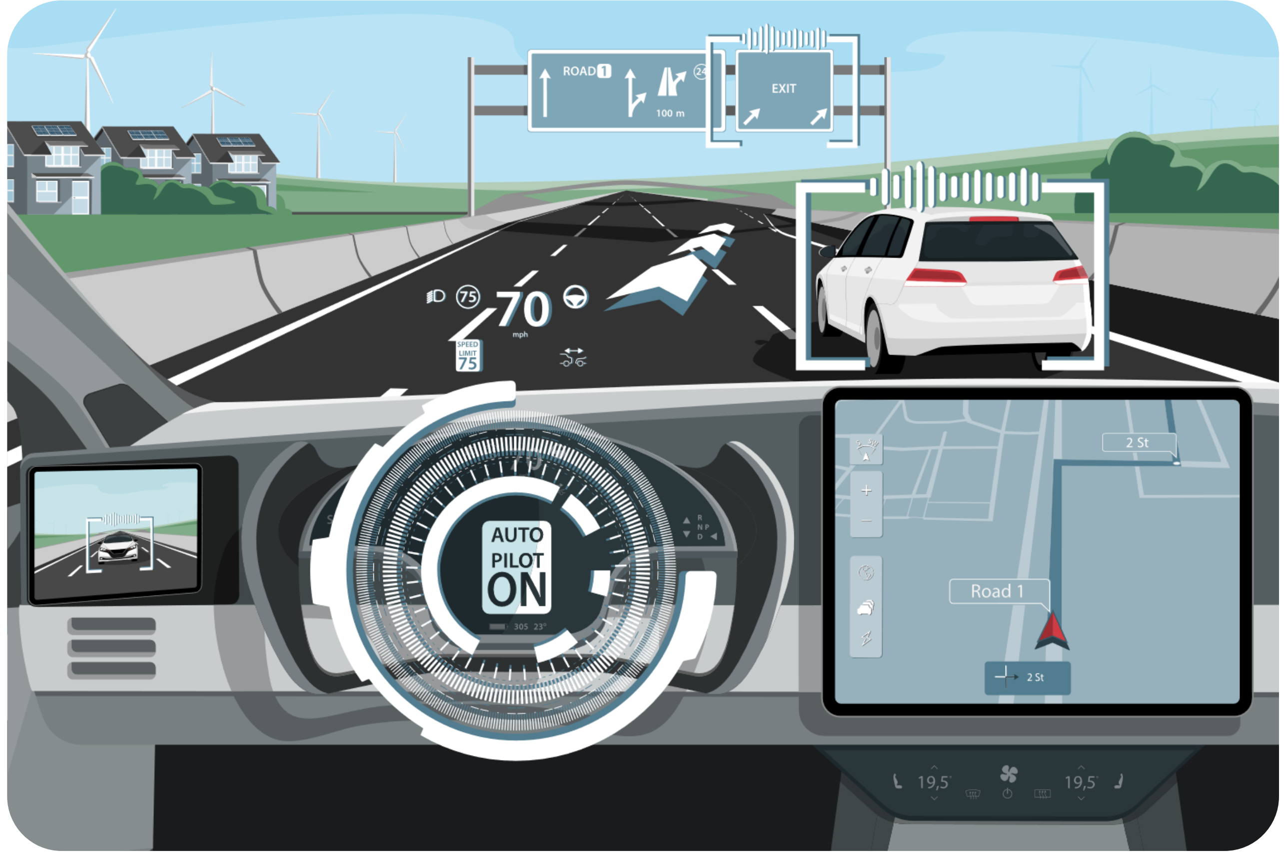


Applications of Machine Learning

ML Application: Fraud Detection

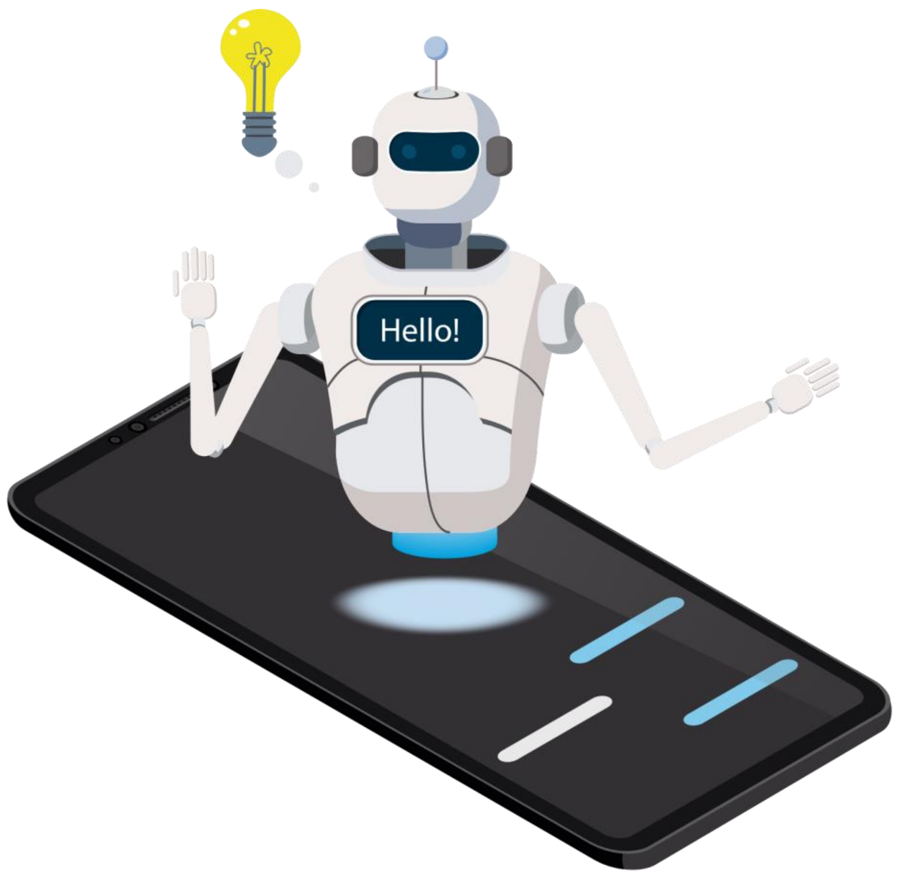
- Banking institutions use machine learning to detect fraud.
- It is valuable to the organization that processes credit card transactions.
- According to the company's standards, the machine learning algorithm is trained to detect transactions that appear to be fraudulent.

ML Application: Self-Driving Cars



- Machine learning algorithms are trained on real-life datasets to enable self-driving cars to make decisions.
- Self-driving cars utilize machine learning algorithms for the following tasks:
  - Identifying objects in the environment
  - Calculating the distance between the car in front
  - Determining the location of the pavement and traffic signals
  - Assessing the driver's state
  - Performing scene classification

ML Application: Smart Phones



- Machine Learning is also used in mobile applications to provide intelligent features.
- Some of the applications of machine learning in smartphone devices are:
  - The voice assistant that sets the alarm and finds the finest restaurants.
  - The basic use case of unlocking the phone using face recognition.

ML Application: Healthcare

- Women's mammography scans can also be reviewed for cancer prediction using computer-assisted diagnosis (CAD), a machine learning application.

- Machine learning aids in treatment planning and delivery, resulting in improved results, cheaper healthcare costs, and increased patient satisfaction.

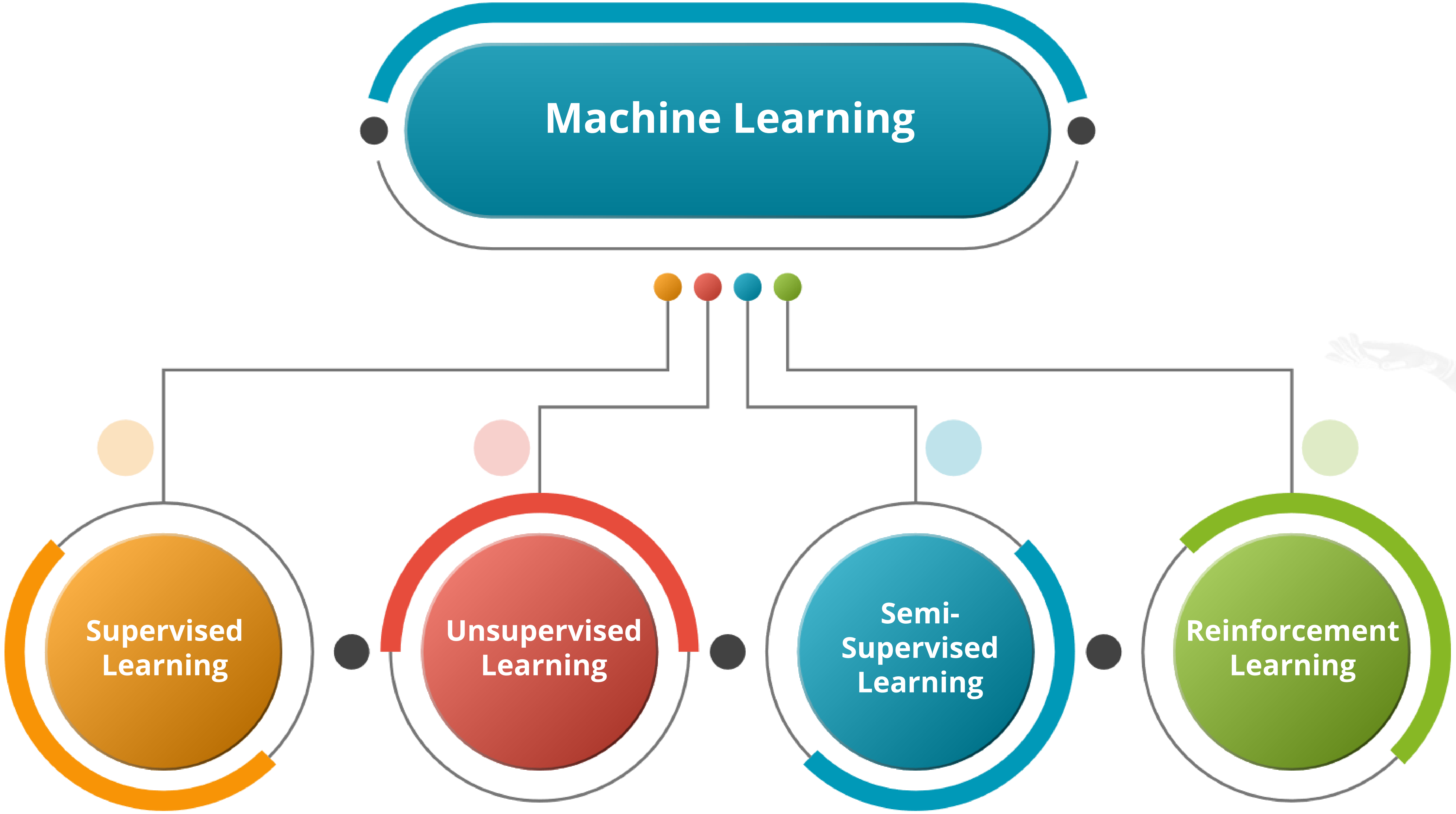
DATA AND ARTIFICIAL INTELLIGENCE

Machine Learning Types

simplilearn

Types of Machine Learning

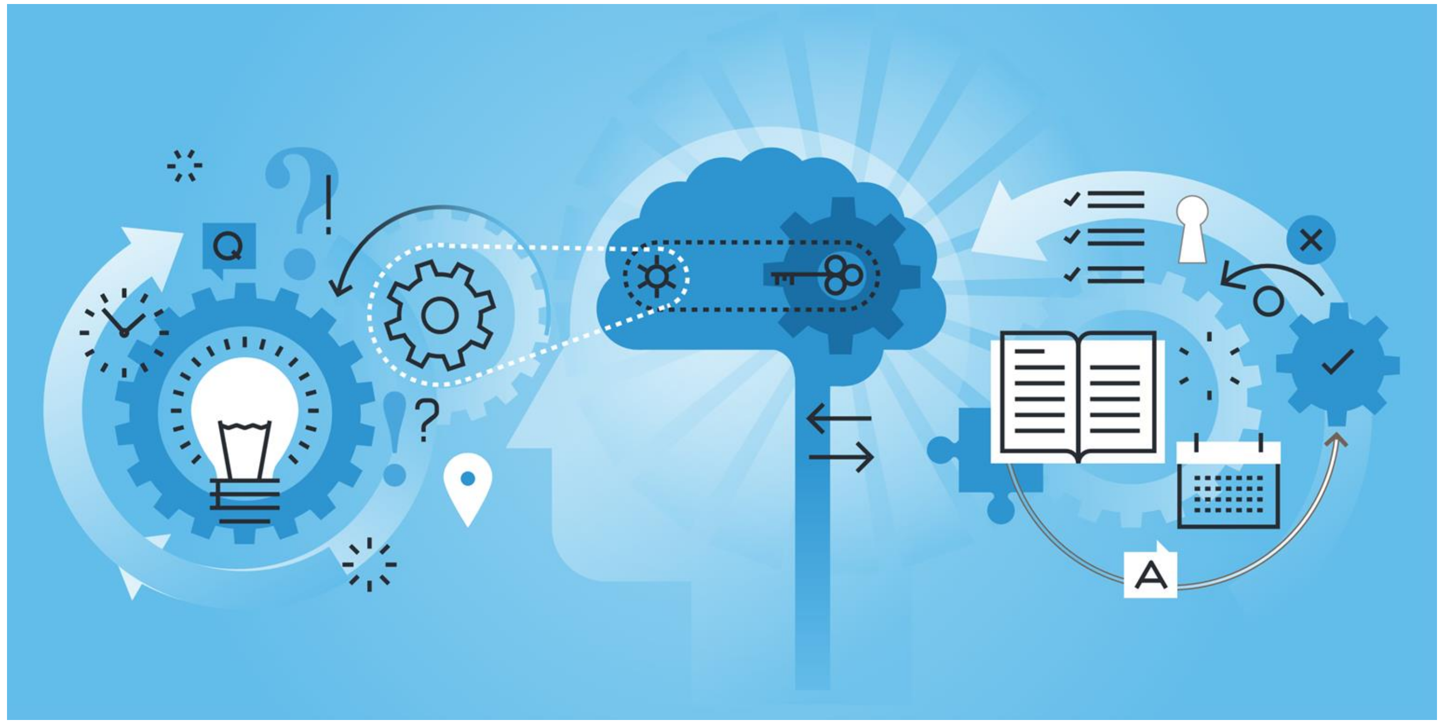
There are four types of machine learning categories.



Supervised Learning

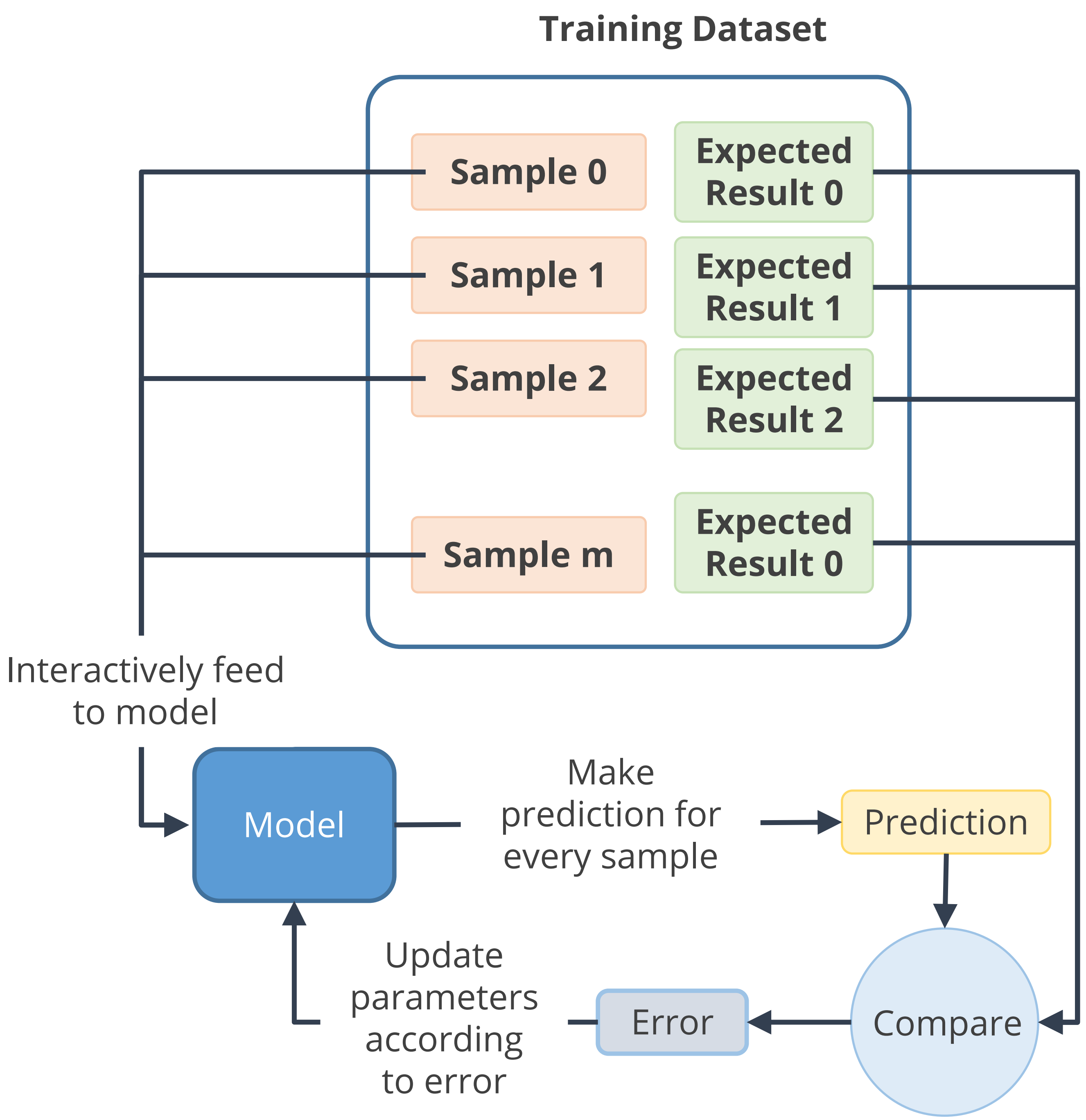
Supervised Learning

Supervised learning is used to train models using labeled training data. It provides the ability to predict the output of future or unseen data.



The goal of a supervised learning algorithm is to discover a mapping function that translates the input variable (x) to the output variable (y).

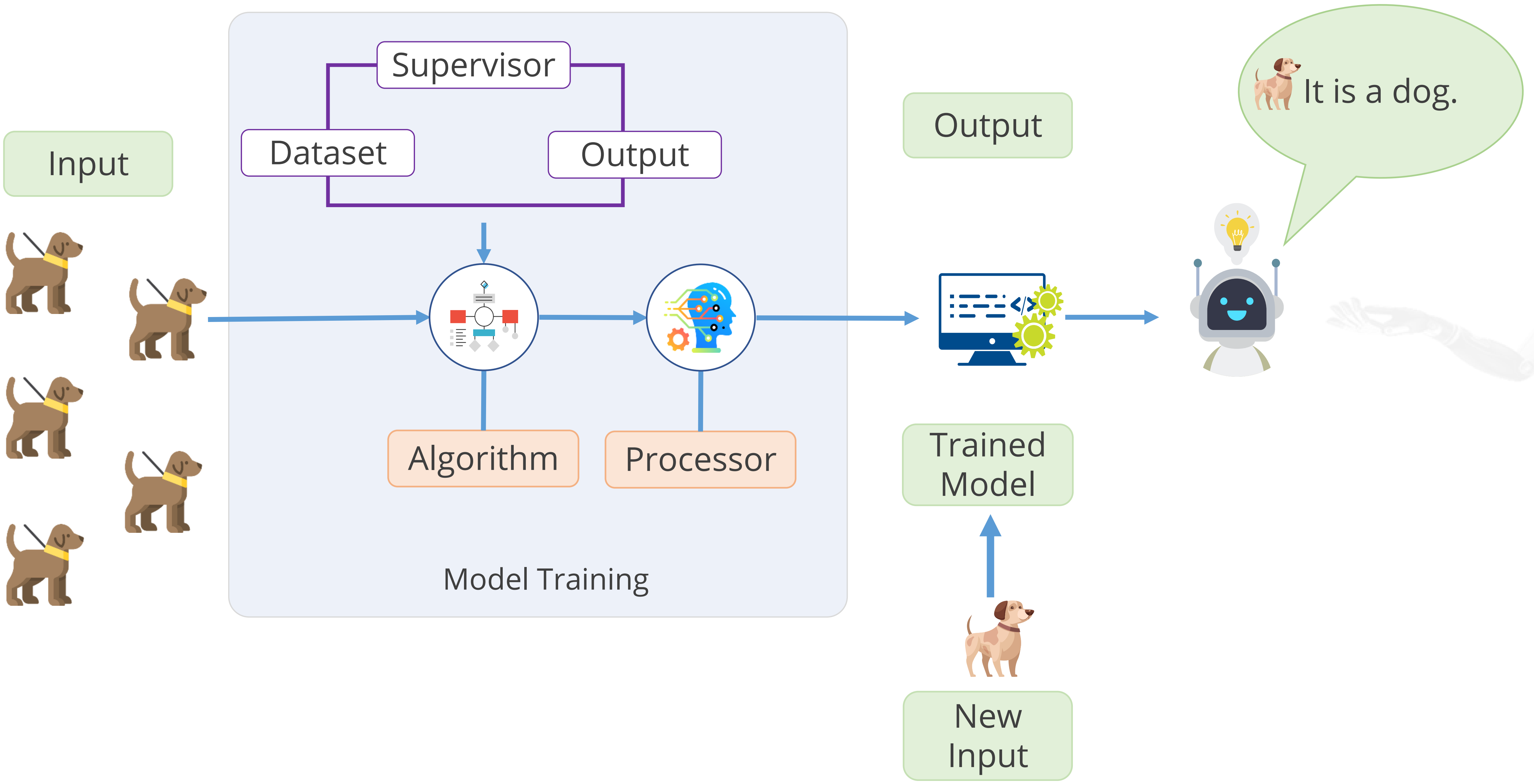
Supervised Learning: Process Flow



- The input-output pairs should make up the required dataset.
- Each pair consists of a data sample for prediction and a label for the expected outcome.
- The human supervisor is responsible for assigning labels to the data in the machine learning process.

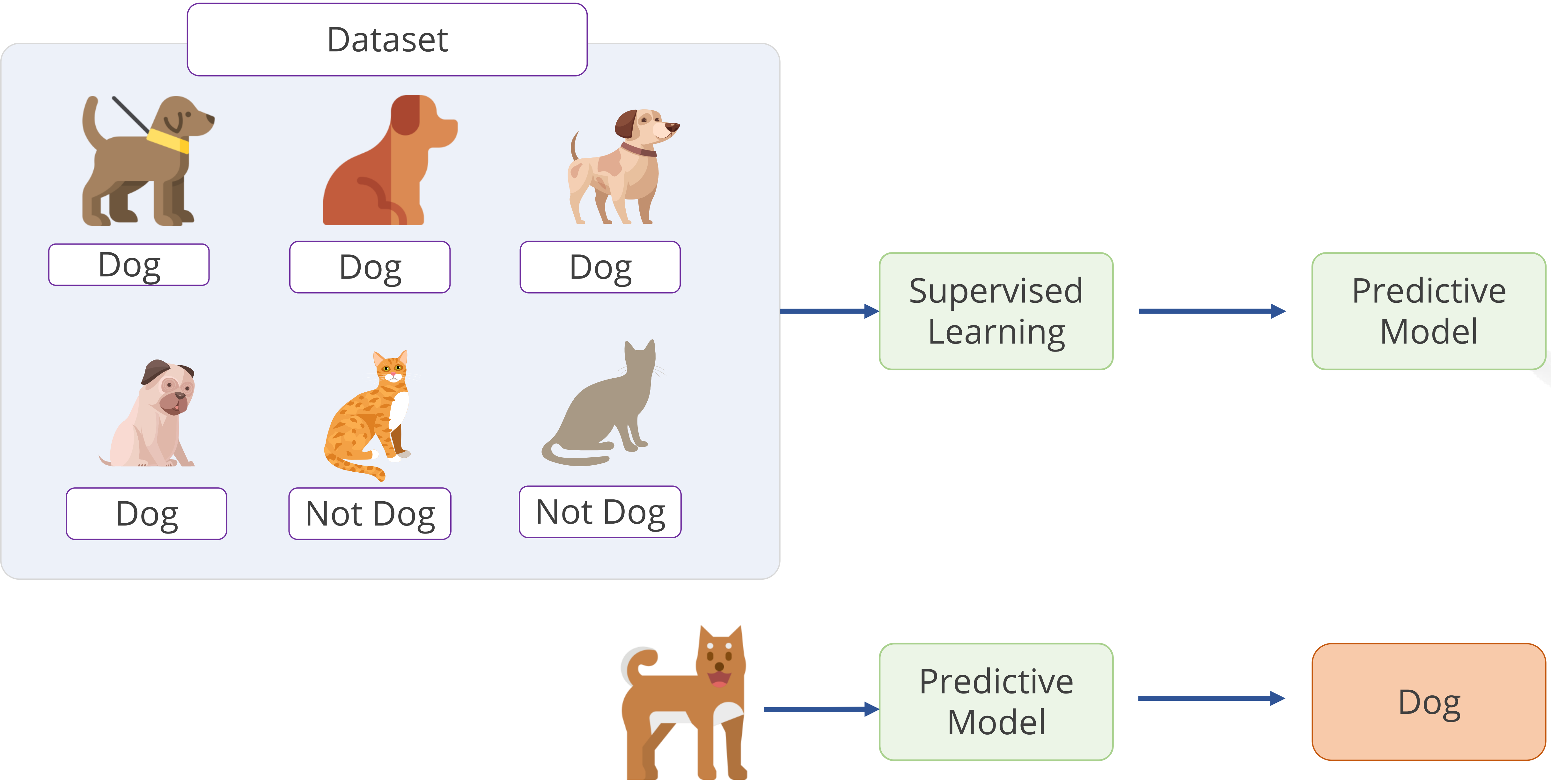
Supervised Learning: Process Flow

The supervised learning process has several stages which are depicted below:



Supervised Learning: Example

An example of a supervised learning process is depicted below:

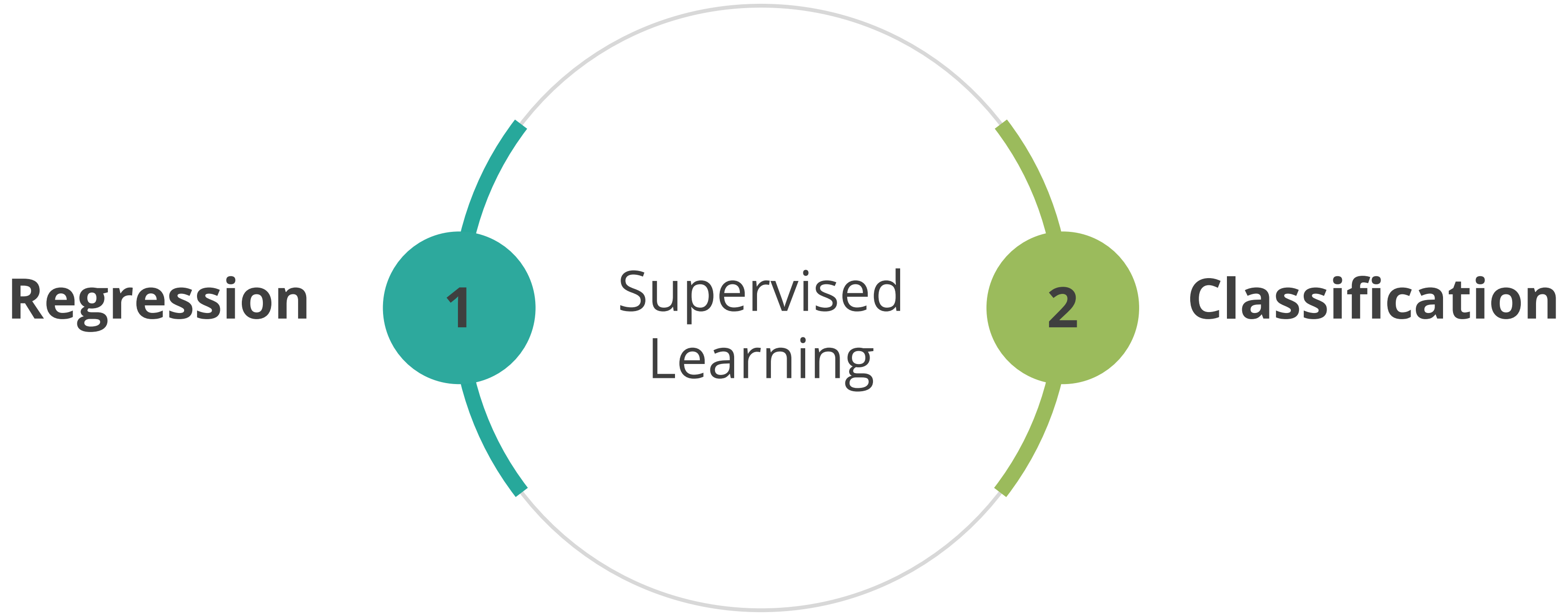


Supervised Learning: Example

Netflix uses **supervised learning** algorithms to recommend shows for the users based on the viewing history and ratings by similar classes of users.

Types of Supervised Learning

In supervised learning, an algorithm is selected based on the target variable.



Unsupervised Learning

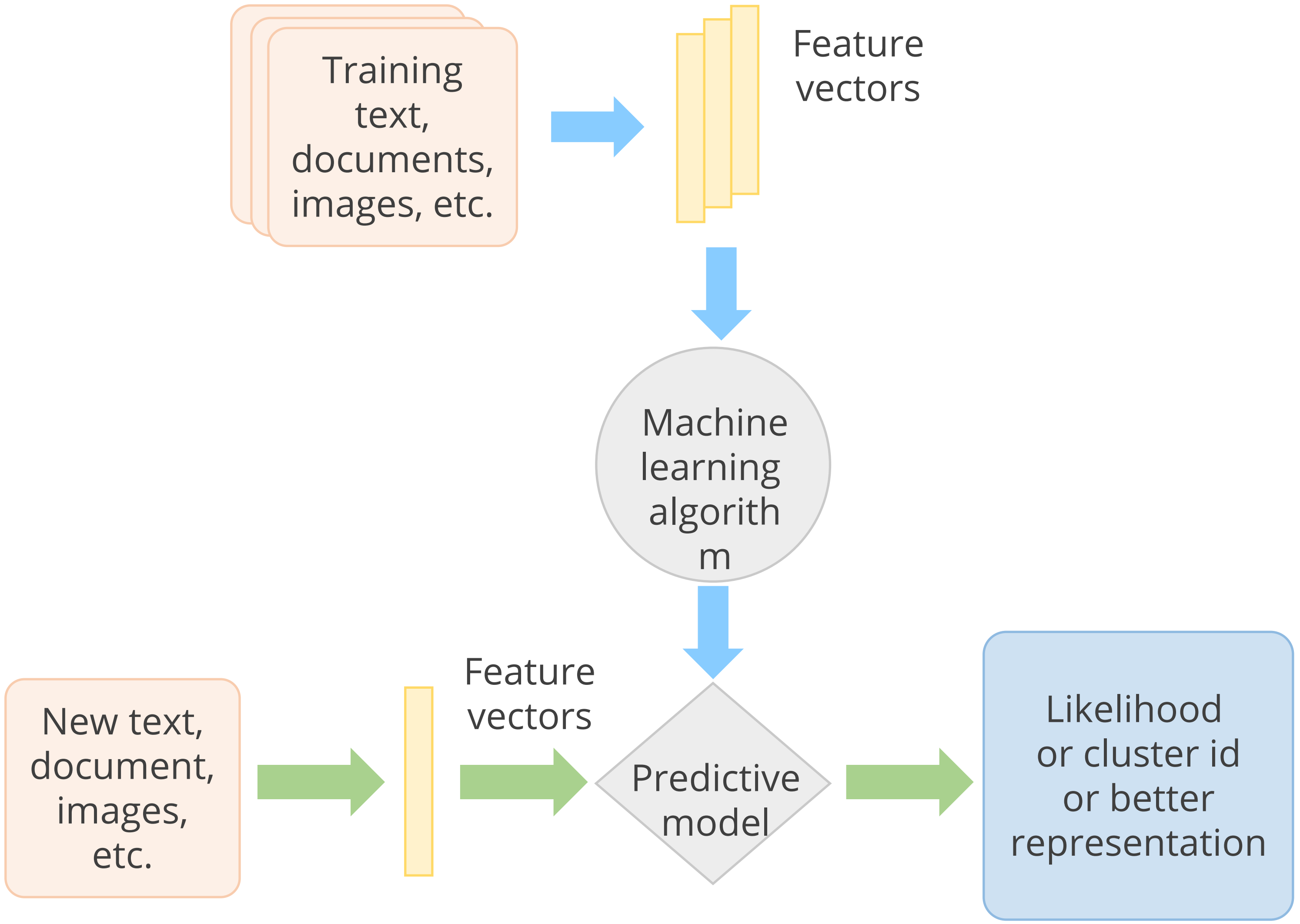
Unsupervised Learning

- Unsupervised learning allows the model to discover patterns and information that were previously undetected.
- It helps to find out all kinds of unknown patterns in data.
- It helps to find out the features required for categorization.

Unsupervised Learning: Process Flow

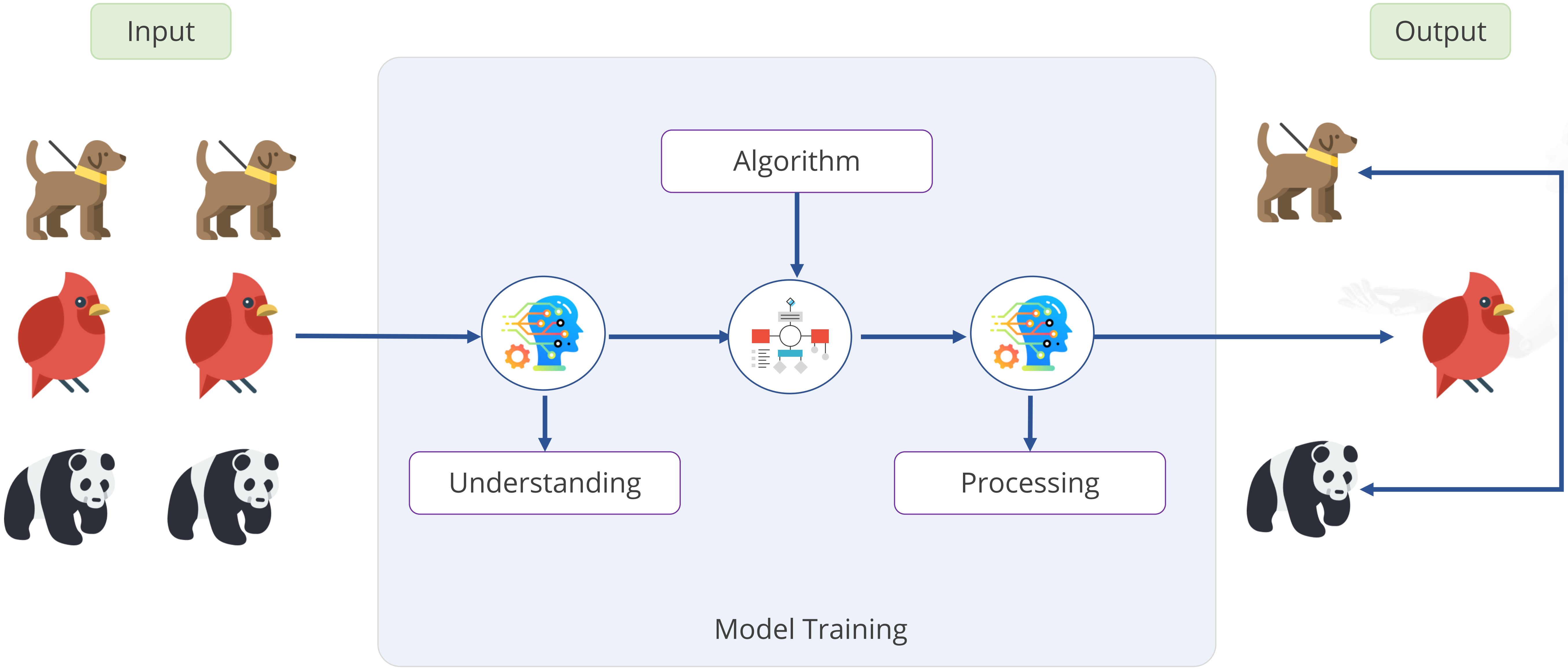
There are no labels on the data. The machine learning algorithm searches for the patterns it can detect.

Unsupervised Learning Model



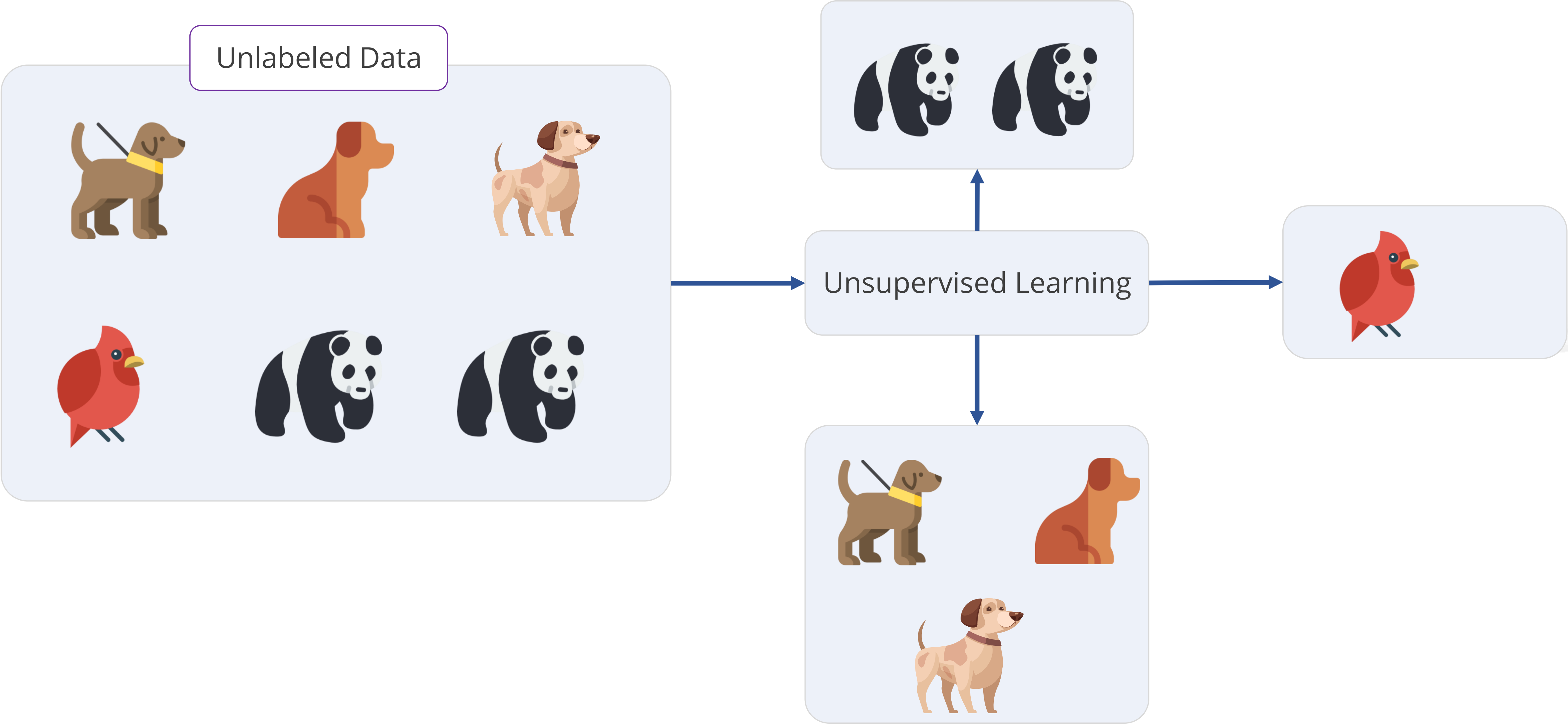
Unsupervised Learning: Process Flow

The unsupervised learning process has several stages which are depicted below:



Unsupervised Learning: Example

An example of unsupervised learning process is depicted below:



Types of Unsupervised Learning

Unsupervised Learning

- Clustering
- Association rule

Clustering

- It is a method of grouping the objects into a cluster.
- The grouping is done in such a way that objects with the most similarities remain in a group and objects with no or fewer similarities are placed in another group.

Association rule

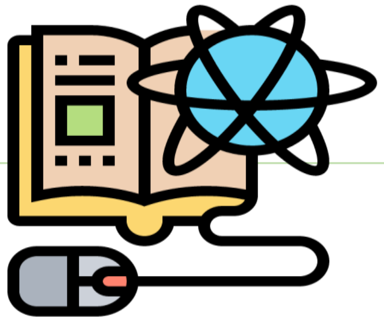
- It is used to find out the relationships between the variables in the large databases.
- For example, people that buy a new home are most likely to buy new furniture.

DATA AND
ARTIFICIAL INTELLIGENCE

Semi-Supervised Learning

simplilearn

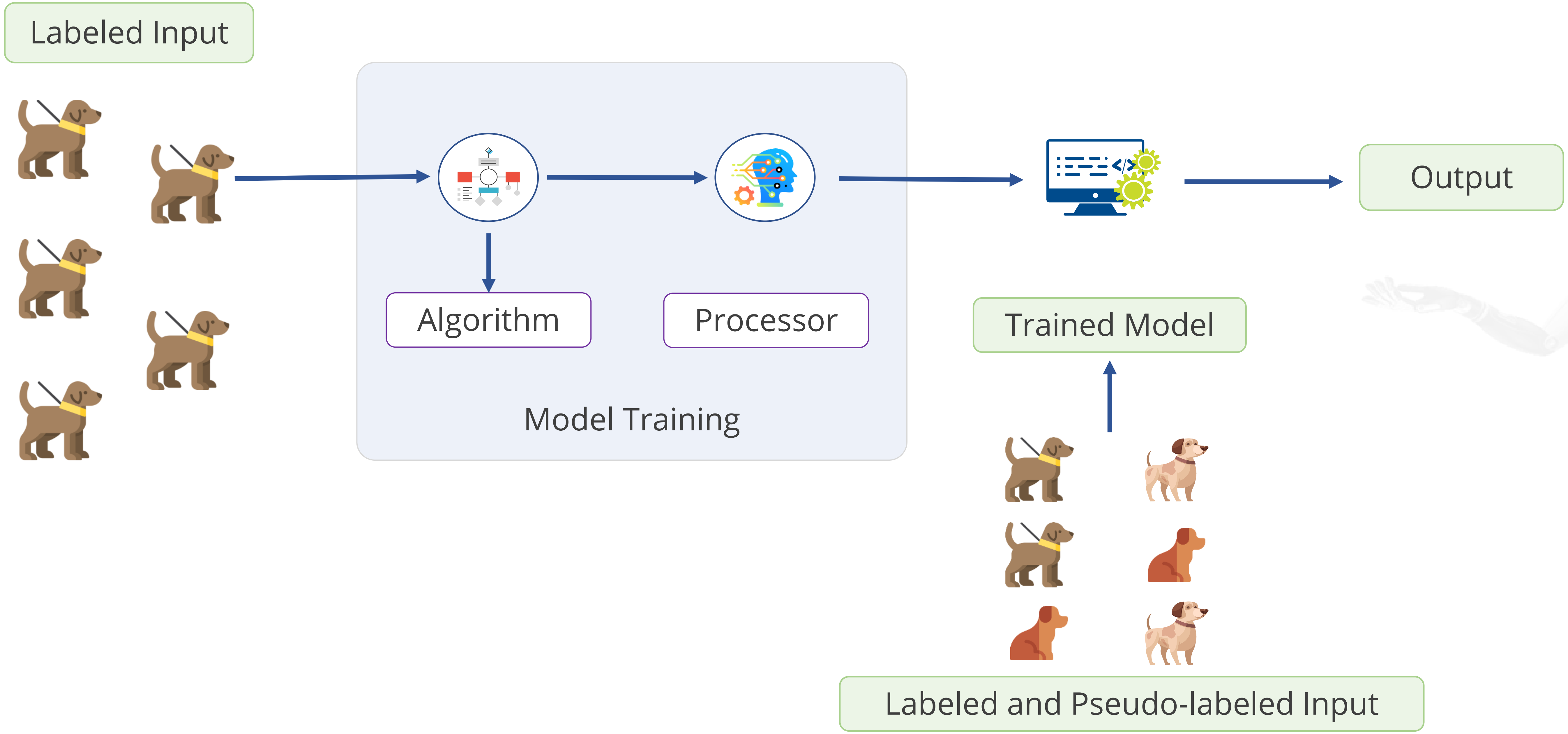
Semi-Supervised Learning



- Semi-supervised learning is a machine learning technique that falls between supervised and unsupervised learning. During the training stage, it utilizes a combination of labeled and unlabeled datasets.
- This sort of learning problem is difficult to solve since neither supervised nor unsupervised learning algorithms can effectively use a mixture of labeled and unlabeled data. As a result, specialized semi-supervised learning algorithms are required.

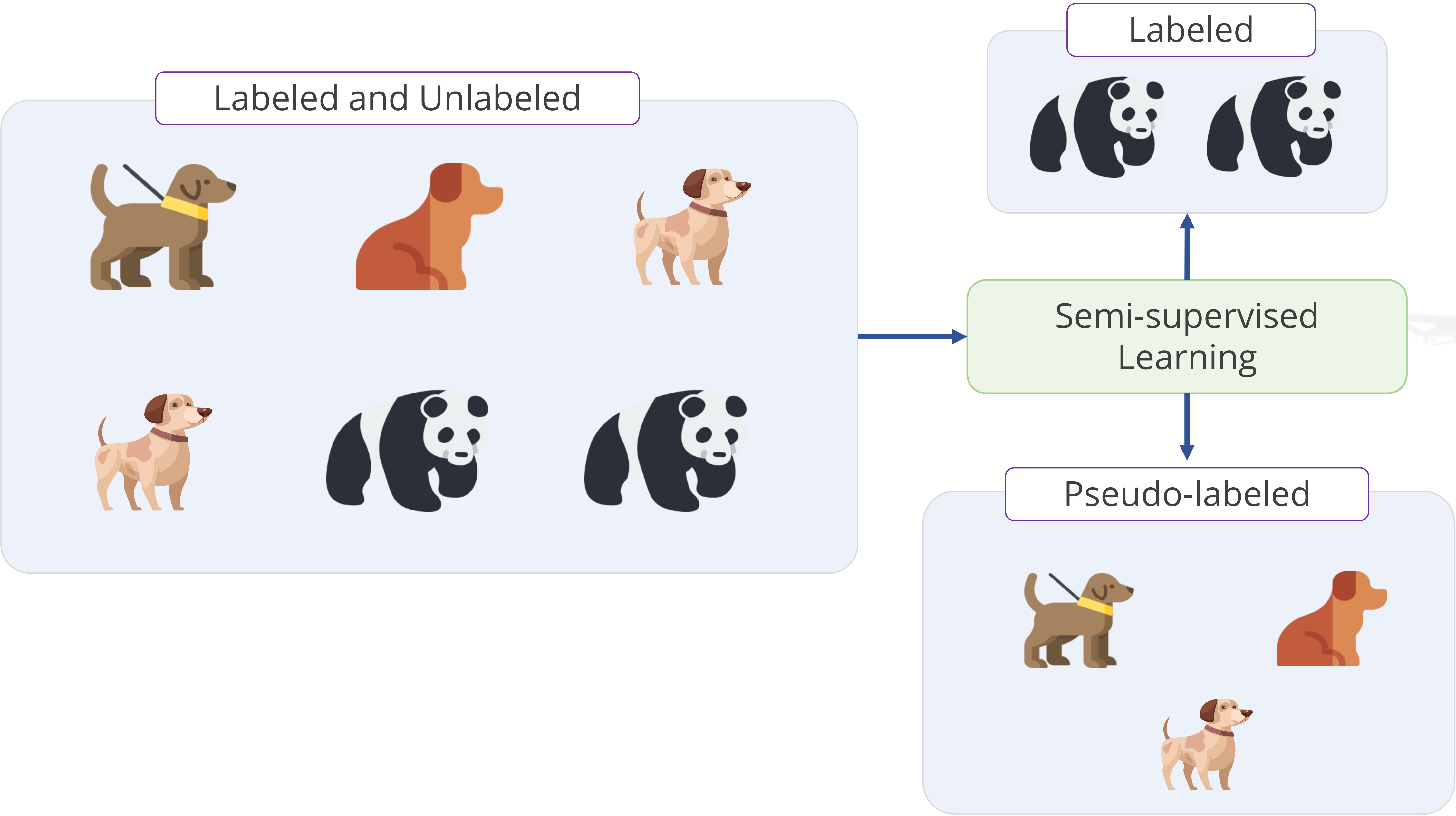
Semi-Supervised Learning: Process Flow

The semi-supervised learning process has several stages which are depicted below:



Semi-supervised Learning: Example

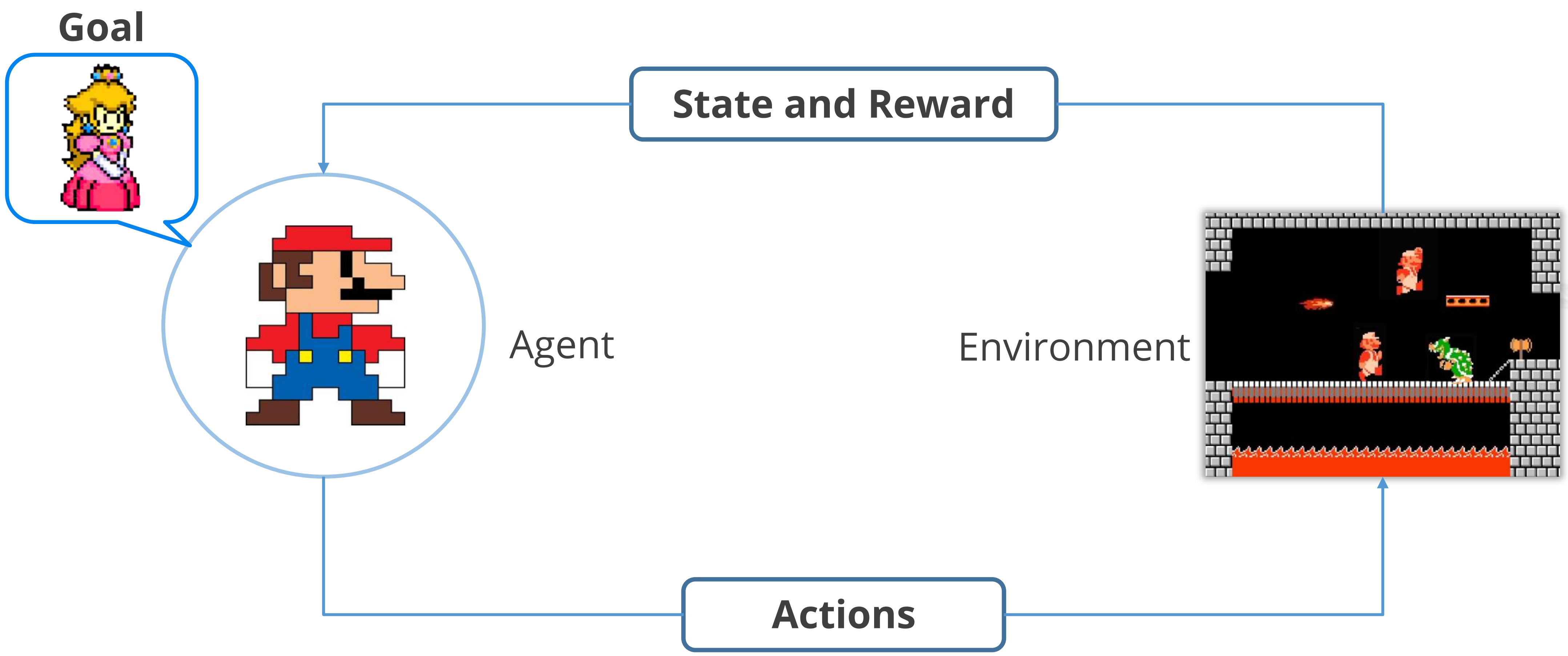
An example of semi-supervised learning process is depicted below:



Reinforcement Learning

Reinforcement Learning

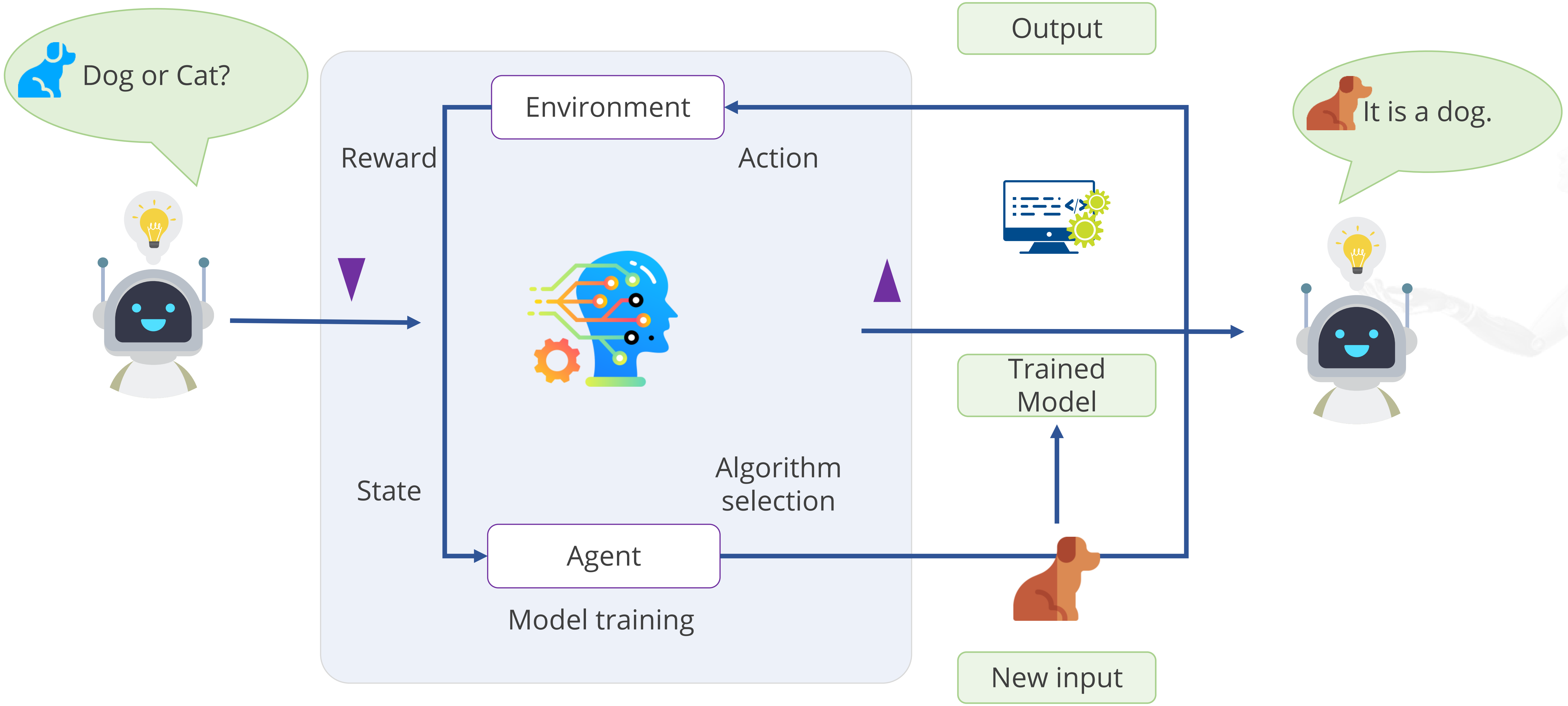
Reinforcement learning agents are goal-oriented. They learn by trial and error in an environment that provides rewards or penalties in response to the agents' outputs.



The aim is to find the best path that maximizes the likelihood of winning a reward.

Reinforcement Learning: Process Flow

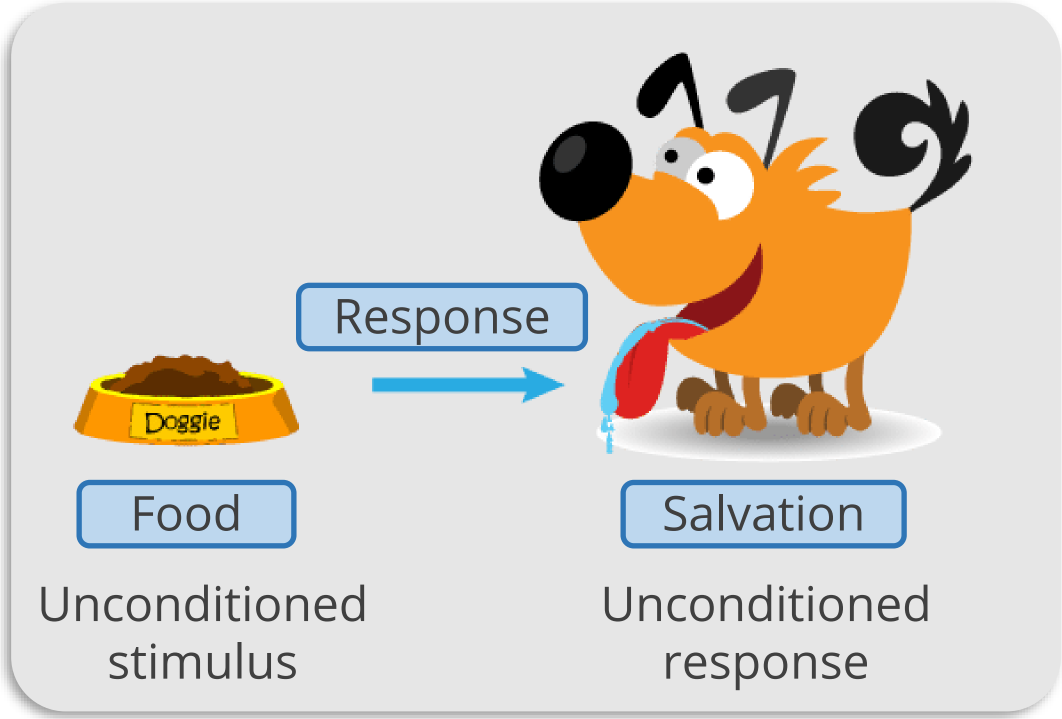
The reinforcement learning process has the following stages:



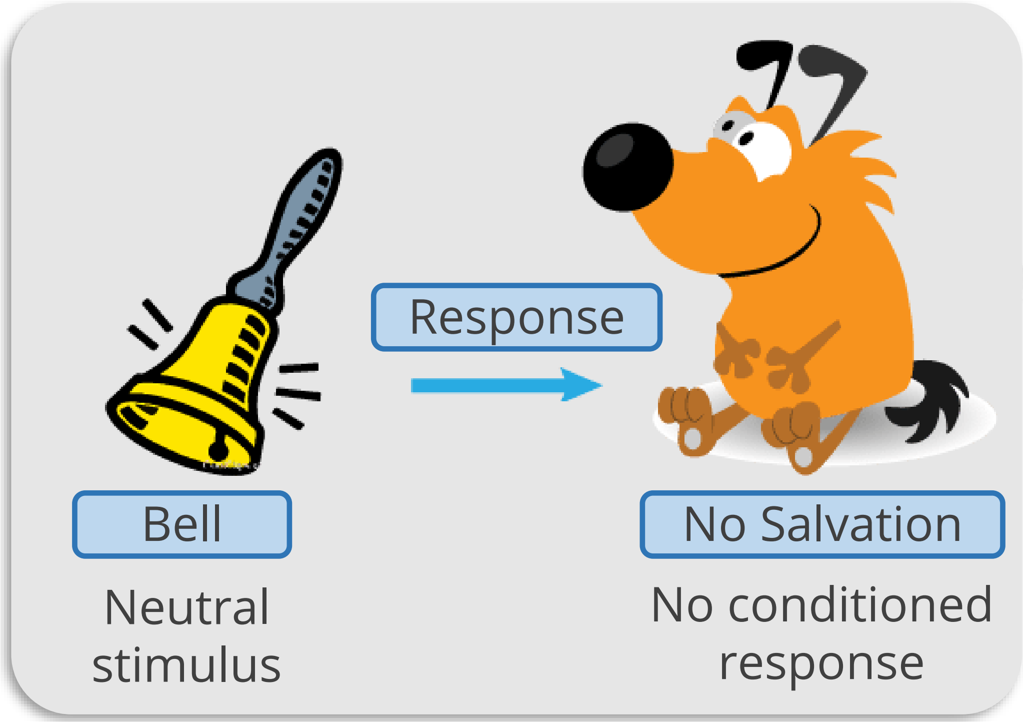
Reinforcement Learning: Example

An example of reinforcement learning process is depicted below:

1. Before conditioning



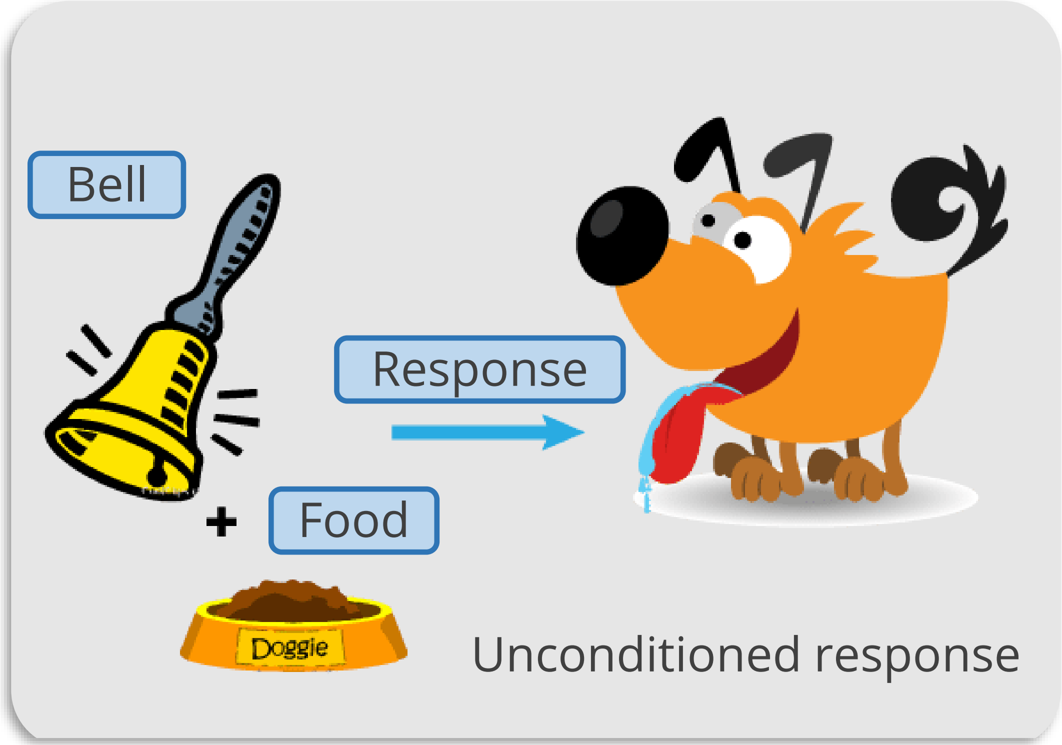
2. Before conditioning



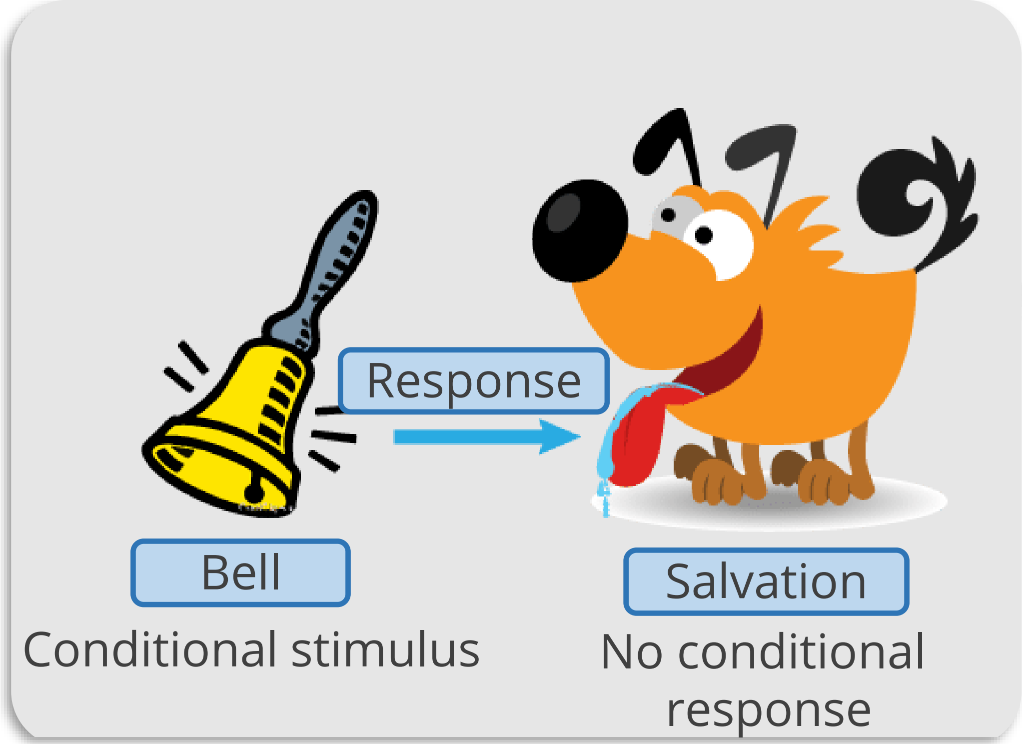
Reinforcement Learning: Example

An example of reinforcement learning process is depicted below:

3. During Conditioning



4. After Conditioning



Reinforcement Learning: Algorithms

A list of reinforcement machine learning algorithms includes:

- Monte Carlo
- Q-Learning
- State–Action–Reward–State–Action (SARSA)
- Deep Q Network (DQN)
- Q-Lambda
- SARSA-Lambda

DATA AND ARTIFICIAL INTELLIGENCE

Machine Learning Use Case: Face Detection

simplilearn

Face Detection with ML

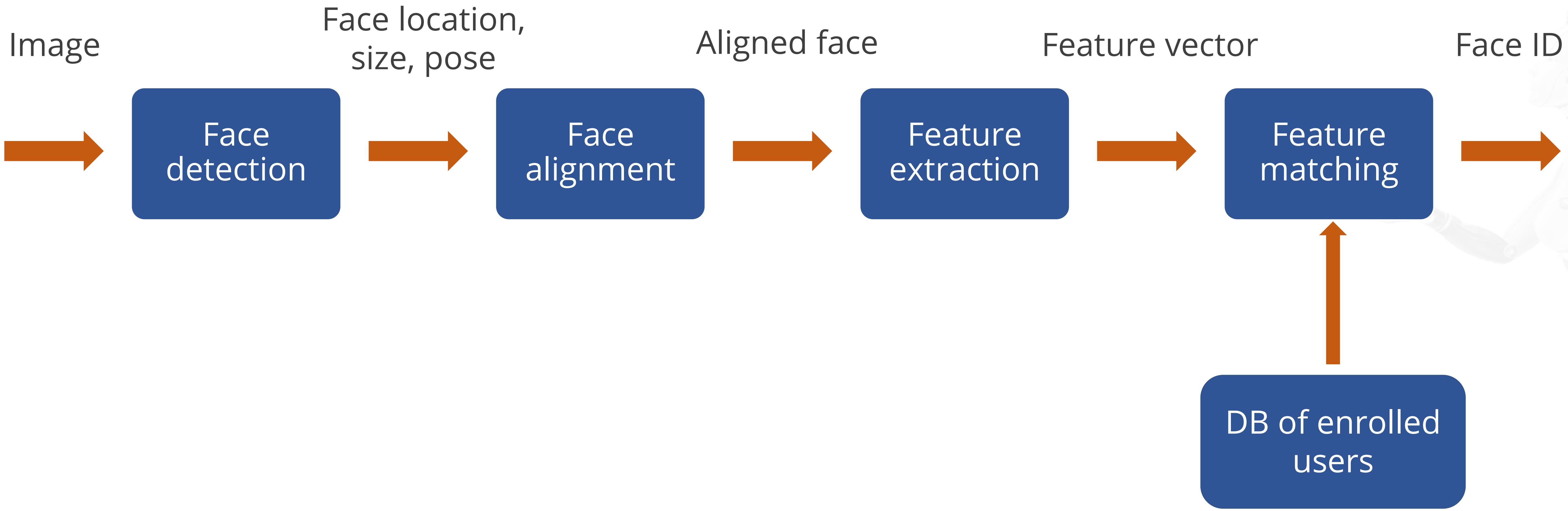
- Face detection is one of the most famous applications of supervised machine learning.
- The process of an algorithm learning from a training dataset is referred to as supervised learning since the procedure may be compared to a teacher supervising the learning process.

Face Detection with ML

- The learning algorithm receives a set of inputs along with the corresponding correct outputs, and it learns by comparing its actual output to the correct output in order to find errors.
- It then modifies the model accordingly using classification, regression, prediction, and gradient boosting techniques.
- Supervised learning uses patterns to predict the label values on additional unlabeled data.

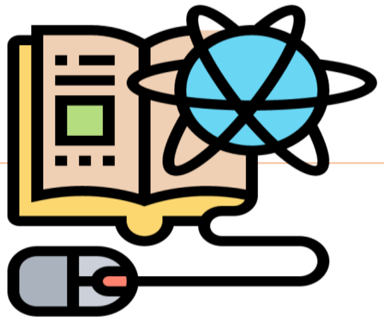
Face Detection with ML: Process Flow

The ML model for face detection has the following stages:



Introduction to Spark ML

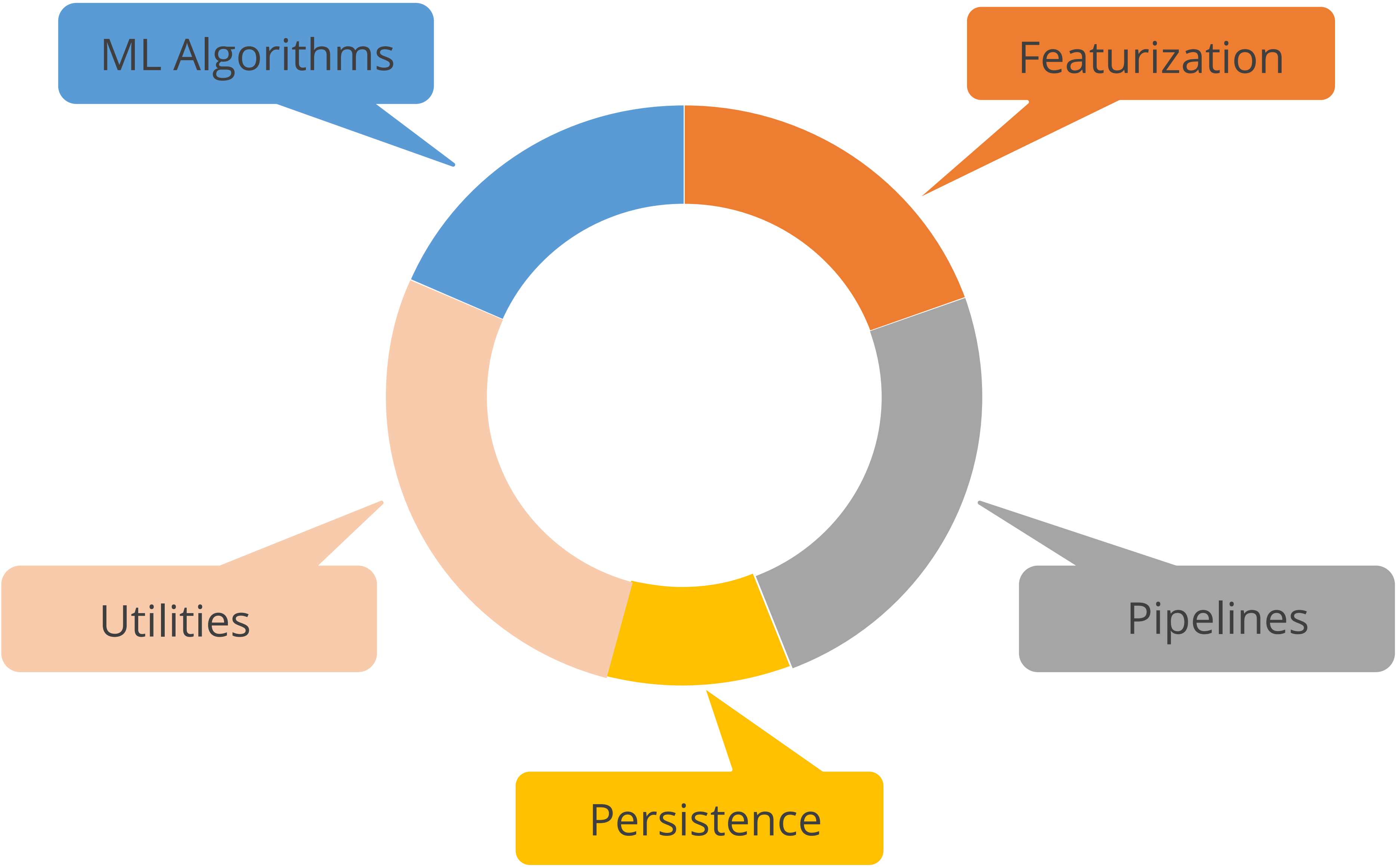
Spark ML



MLlib is a scalable machine learning library for Spark consisting of common learning algorithms, tools, and optimization primitives.

Spark ML: Tools

SparkML comprises the following five major tools:

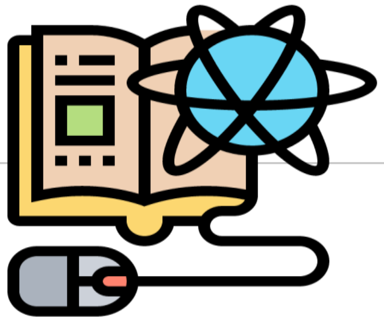


Spark ML: Algorithms

The **MLlib** library of SparkML consists of various ML algorithms and utilities.

ML Pipeline

ML Pipeline

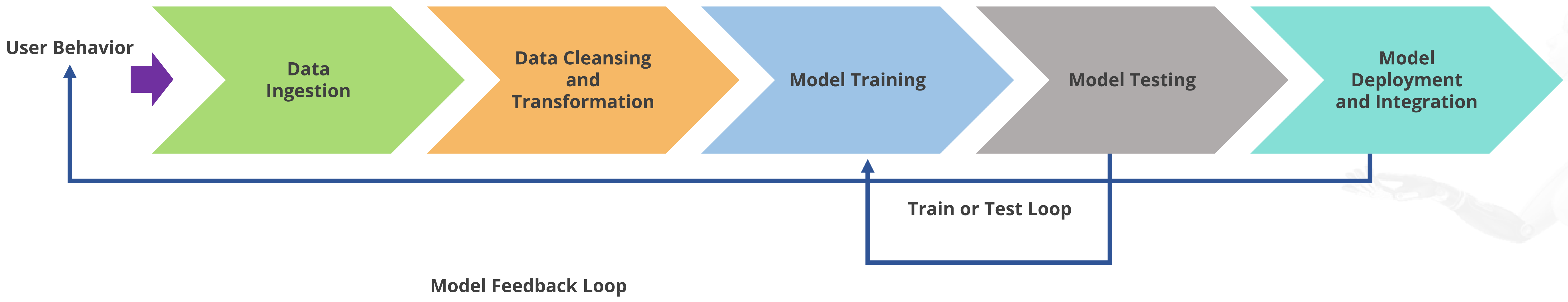


- A machine learning pipeline is used to assist in the automation of machine learning workflows. They work by enabling a sequence of data to be transformed and correlated in a model that can be tested and evaluated to achieve a positive or negative outcome.

- Machine learning (ML) pipelines are made up of a series of sequential steps that handle everything from data extraction and preprocessing to model training and deployment.

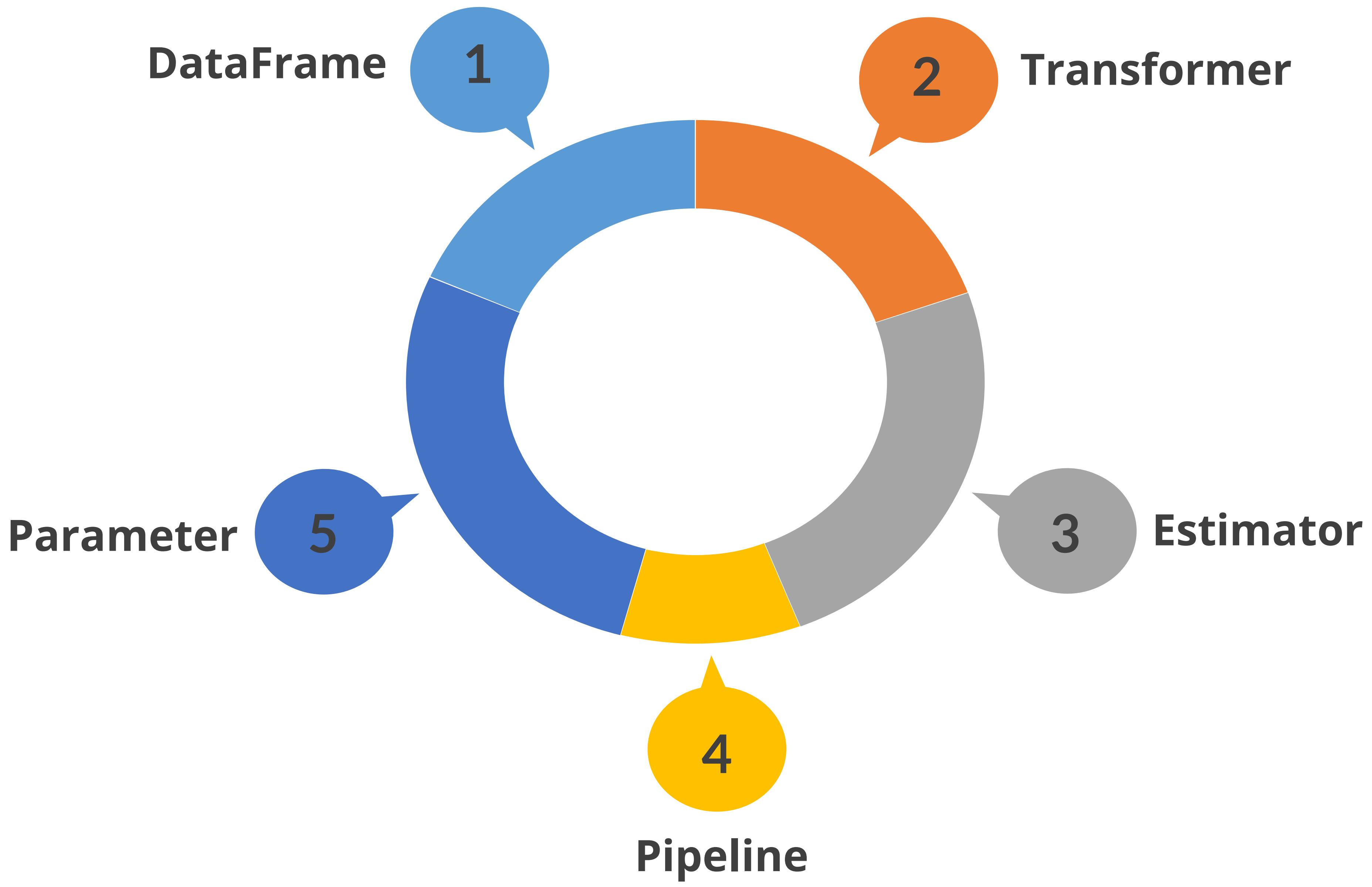
ML Pipeline: Process Flow

Machine learning (ML) pipelines consist of several steps to train a model.



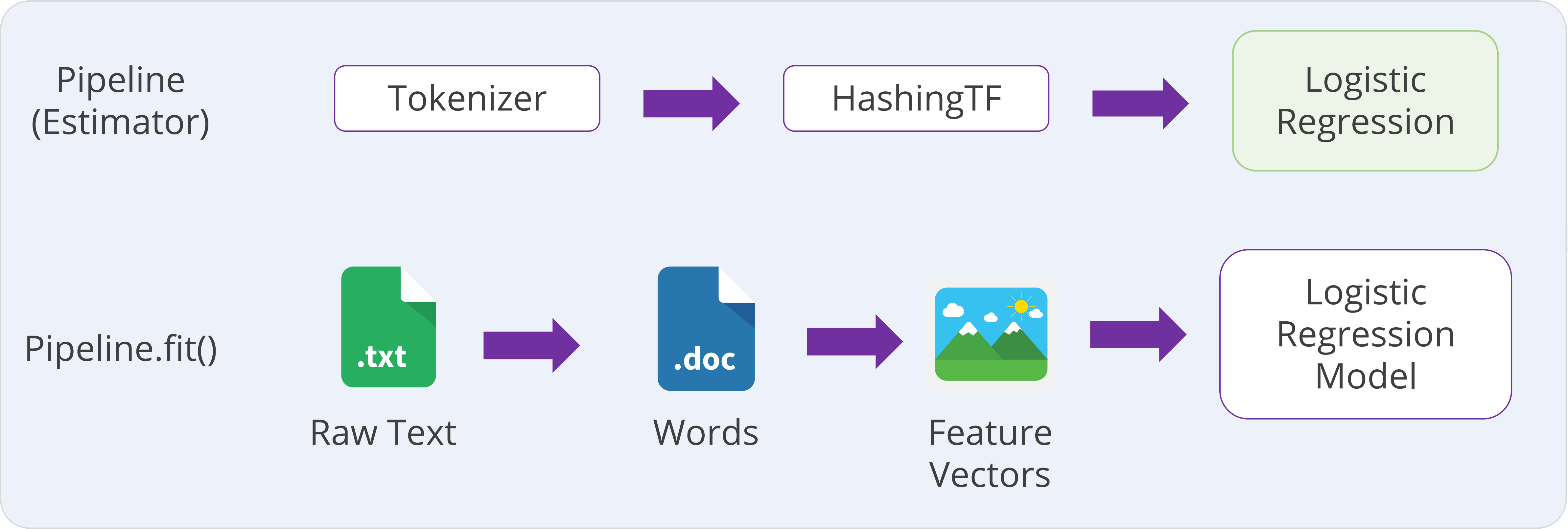
ML Pipeline: APIs

ML Pipelines offer a consistent collection of high-level APIs that assist users in creating and tuning practical machine learning pipelines.



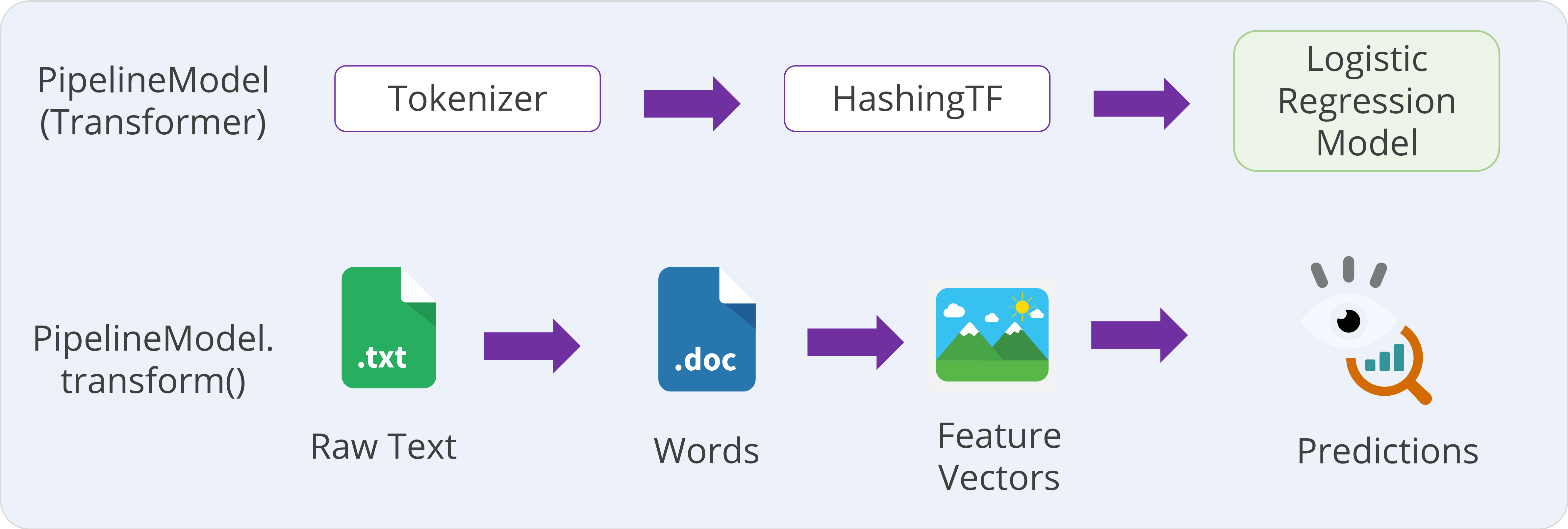
ML Pipeline: Workflow

A pipeline is viewed as a sequence of stages, each of which is either a transformer or an estimator.



ML Pipeline: Workflow

A pipeline is viewed as a sequence of stages, each of which is either a transformer or an estimator.



Machine Learning Examples

ML Example: Linear Regression

The format for the customer data is shown below:

label	Avg Session Length	Time on App	Time on Website	Length of Membership
587.951054	34.49726773	12.65565115	39.57766802	4.082620633
392.2049334	31.92627203	11.10946073	37.26895887	2.664034182
487.5475049	33.00091476	11.33027806	37.11059744	4.104543202
581.852344	34.30555663	13.71751367	36.72128268	3.120178783
599.406092	33.33067252	12.79518855	37.5366533	4.446308318
637.1024479	33.87103788	12.02692534	34.47687763	5.493507201
521.5721748	32.0215955	11.36634831	36.68377615	4.685017247
549.9041461	32.73914294	12.35195897	37.37335886	4.434273435
570.200409	33.9877729	13.38623528	37.53449734	3.273433578

Input: Customer Data

ML Example: Linear Regression

This example shows how to train a linear regression model.

Example

```
#Importing the required libraries
from pyspark.ml.feature import VectorAssembler
from pyspark.ml.regression import LinearRegression
from pyspark.sql import SparkSession
from pyspark.sql import functions as F

#Create a spark session
spark = SparkSession \
    .builder \
    .appName("LinReg Example") \
    .getOrCreate()

#Load the input data
ecommDF = spark.read.option("header", "true") \
    .option("inferSchema", "true") \
    .csv("../data-files/customers-ml/EcommerceCustomers.csv")
```

ML Example: Linear Regression

Example (Continued/…)

```
ecommDF1 = ecommDF.select(F.col("Yearly Amount Spent").alias("label"), F.col("Avg Session
Length"), F.col("Time on App"), F.col("Time on Website"), F.col("Length of Membership"))

#Create a list with the names of the input columns
inputCols = ["Avg Session Length", "Time on App", "Time on Website", "Length of Membership"]

#Create a vector assembler
assembler = VectorAssembler().setInputCols(inputCols).setOutputCol("features")

#Transform the input dataframe using VectorAssembler
ecommDF2 = assembler.transform(ecommDF1).select(F.col("label"), F.col("features"))
ecommDF2.show(10, False)
```

ML Example: Linear Regression

Output:

```
+-----------+---------------------------------------------------+
|label      |features                                           |
+-----------+---------------------------------------------------+
|587.951054 |[34.49726773,12.65565115,39.57766802,4.082620633]|
|392.2049334|[31.92627203,11.10946073,37.26895887,2.664034182]|
|487.5475049|[33.00091476,11.33027806,37.11059744,4.104543202]|
|581.852344 |[34.30555663,13.71751367,36.72128268,3.120178783]|
|599.406092 |[33.33067252,12.79518855,37.5366533,4.446308318] |
|637.1024479|[33.87103788,12.02692534,34.47687763,5.493507201]|
|521.5721748|[32.0215955,11.36634831,36.68377615,4.685017247] |
|549.9041461|[32.73914294,12.35195897,37.37335886,4.434273435]|
|570.200409 |[33.9877729,13.38623528,37.53449734,3.273433578] |
|427.1993849|[31.93654862,11.81412829,37.14516822,3.202806072]|
+-----------+---------------------------------------------------+
only showing top 10 rows
```

ML Example: Linear Regression

Example (Continued/…)

```
#Create an object of LinearRegression
lr = LinearRegression()

#Provide the training data
lrModel = lr.fit(ecommDF2)

print("\n\nCoefficients:" + str(lrModel.coefficients) + "\n\nIntercept:" +
str(lrModel.intercept))
```

Output:

```
Coefficients:[25.734271083497525,38.70915381360397,0.43673882831278l9,61.57732374979357]

Intercept:-1051.5942549969473
```

ML Example: Linear Regression

Example (Continued/...)

```
#Print the model training summary
trainingSummary = lrModel.summary
print("\n\nModel Summary:\n")
print("numIterations: " + str(trainingSummary.totalIterations) + "\n")
print("objectiveHistory: " + str(trainingSummary.objectiveHistory) + "\n")
trainingSummary.residuals.show(5)
print("\n\n")
```

Output:

```
Model Summary:

numIterations: 0

objectiveHistory: [0.0]
```

ML Example: Linear Regression

Output:

```
+--------------------+
|           residuals|
+--------------------+
|  -6.788234207763367|
|  11.841128372827995|
|-17.652627154231084|
|  11.454889368172758|
|   7.783382546060011|
+--------------------+
only showing top 5 rows
```

ML Example: Linear Regression

Example (Continued/...)

```
#Model Evaluation
print("\n\nModel Summary:\n")
print("RMSE: " + str(trainingSummary.rootMeanSquaredError)+"\n")
print("MSE: " + str(trainingSummary.meanSquaredError)+"\n")
print("r2: " + str(trainingSummary.meanSquaredError)+"\n\n")
```

Output:

```
Model Summary:

RMSE: 9.923256786178925

MSE: 98.47102524444608

r2: 98.47102524444608
```

ML Example: Logistic Regression

This example shows how to train binomial and multinomial logistic regression models for binary classification with elastic net regularization. Here, the elasticNetParam corresponds to α and the regParam corresponds to λ.

Example

```
#Importing the required libraries
from pyspark.ml.classification import LogisticRegression
from pyspark.sql import SparkSession

#Create a spark session
spark = SparkSession \
    .builder \
    .appName("LogReg Example") \
    .getOrCreate()

#Loading the training data
training = spark.read.format("libsvm") \
    .load("../data-files/logistic-regression/sample_libsvm_data.txt")
```

ML Example: Logistic Regression

Example (Continued/...)

```
#Create an object of the logistic regression model
lr = LogisticRegression(maxIter=10, regParam=0.3, elasticNetParam=0.8)

#Fit the model
lrModel = lr.fit(training)

#Print the coefficients and intercept for logistic regression
print("\n\nCoefficients: " + str(lrModel.coefficients) + "\n")
print("Intercept: " + str(lrModel.intercept) + "\n\n")
```

ML Example: Logistic Regression

Output:

```
Coefficients: (692,[244,263,272,300,301,328,350,351,378,379,405,406,407,428,4
33,434,455,456,461,462,483,484,489,490,496,511,512,517,539,540,568],[-7.35398
3524188241e-05,-9.102738505589566e-05,-0.00019467430546904 23,-0.000203006424 7
3486603,-3.147618331486458e-05,-6.842977602660821e-05,1.5883626898236275e-05,
1.4023497091368928e-05,0.0003543204752496838,0.0001144327289817 1099,0.0001001
6712383666487,0.0006014109303795511,0.0002840248179122765,-0.000115410847365 0
8905,0.0003859968863129 06,0.0006350195574241097,-0.0001150641238457573 3,-0.00
015271865864986 89,0.000280493380899 4214,0.000607011747119 1665,-0.000200845966
3247435,-0.0001421075579290 1347,0.0002739010341160883,0.000277304562449 6811,-
9.838027027269408e-05,-0.00038085224435175833,-0.00025315198008554285,0.00027
74771477075434,-0.0002443619763919 1286,-0.0015394744687597679,-0.000230733284
11330604])

Intercept: 0.22456315961250245
```

ML Example: Logistic Regression

Example (Continued/…)

```
#Using multinomial family for binary classification
mlr = LogisticRegression(maxIter=10, regParam=0.3, elasticNetParam=0.8,
family="multinomial")

#Fit the model
mlrModel = mlr.fit(training)

#Print the coefficients and intercepts for logistic regression with multinomial family
print("\n\nMultinomial coefficients: " + str(mlrModel.coefficientMatrix) + "\n")
print("Multinomial intercepts: " + str(mlrModel.interceptVector) + "\n\n" )
```

ML Example: Logistic Regression

Output:

```
Multinomial coefficients: 2 X 692 CSRMatrix
(0,244) 0.0
(0,263) 0.0001
(0,272) 0.0001
(0,300) 0.0001
(0,350) -0.0
(0,351) -0.0
(0,378) -0.0
(0,379) -0.0
(0,405) -0.0
(0,406) -0.0006
(0,407) -0.0001
(0,428) 0.0001
(0,433) -0.0
(0,434) -0.0007
(0,455) 0.0001
(0,456) 0.0001
..
..

Multinomial intercepts: [-0.12065879445860596,0.12065879445860596]
```

ML Example: K-Means Clustering

The format for the vehicle data is shown below:

```
dt,lat,lon,base
04-01-2014 00:11,40.769,-73.9549,B02512
04-01-2014 00:17,40.7267,-74.0345,B02512
04-01-2014 00:21,40.7316,-73.9873,B02512
04-01-2014 00:28,40.7588,-73.9776,B02512
04-01-2014 00:33,40.7594,-73.9722,B02512
04-01-2014 00:33,40.7383,-74.0403,B02512
04-01-2014 00:39,40.7223,-73.9887,B02512
04-01-2014 00:45,40.762,-73.979,B02512
04-01-2014 00:55,40.7524,-73.996,B02512
04-01-2014 01:01,40.7575,-73.9846,B02512
```

Input: vehicle_data.csv

ML Example: K-Means Clustering

This example shows how to perform classification using the kmeans clustering model.

Example

```
#Importing the required libraries
from pyspark.sql import SparkSession
from pyspark.ml.feature import VectorAssembler
from pyspark.ml.clustering import KMeans
from pyspark.sql.types import StructType, StructField, DoubleType, StringType, TimestampType

#Create a spark session and load the data
spark = SparkSession \
    .builder \
    .appName("K-Means Example") \
    .getOrCreate()
```

ML Example: K-Means Clustering

Example (Continued/...)

```
#Create a custom schema to read the data from the CSV
schema = StructType([
    StructField("dt", TimestampType(), True),
    StructField("lat", DoubleType(), True),
    StructField("lon", DoubleType(), True),
    StructField("base", StringType(), True)
])

#Read the input CSV data
tripDF = spark.read \
    .option("header", True) \
    .schema(schema) \
    .csv("../data-files/transport/vehicle-data.csv")
```

ML Example: K-Means Clustering

Example (Continued/...)

```
#Create feature columns
featureCols = ["lat", "lon"]

#Create a vector assembler
# VectorAssembler() is a transformer that combines a given
# list of columns into a single vector column
# It is useful for combining raw features and features
# generated by different feature transformers into a single
# feature vector, in order to train ML models like
# logistic regression and decision trees.

#Create a vector assembler
assembler = VectorAssembler() \
    .setInputCols(featureCols) \
    .setOutputCol("features")
```

ML Example: K-Means Clustering

Example (Continued/...)

```
#Transform the input dataframe using VectorAssembler
train_df = assembler.transform(tripDF)
train_df.show(10, False)
```

Output:

```
+----+-------+--------+------+------------------+
|dt  |lat    |lon     |base  |features          |
+----+-------+--------+------+------------------+
|null|40.769 |-73.9549|B02512|[40.769,-73.9549] |
|null|40.7267|-74.0345|B02512|[40.7267,-74.0345]|
|null|40.7316|-73.9873|B02512|[40.7316,-73.9873]|
|null|40.7588|-73.9776|B02512|[40.7588,-73.9776]|
|null|40.7594|-73.9722|B02512|[40.7594,-73.9722]|
|null|40.7383|-74.0403|B02512|[40.7383,-74.0403]|
|null|40.7223|-73.9887|B02512|[40.7223,-73.9887]|
|null|40.762 |-73.979 |B02512|[40.762,-73.979]  |
|null|40.7524|-73.996 |B02512|[40.7524,-73.996] |
|null|40.7575|-73.9846|B02512|[40.7575,-73.9846]|
+----+-------+--------+------+------------------+
only showing top 10 rows
```

ML Example: K-Means Clustering

Example (Continued/...)

```
#Train the KMeans algorithm
kmeans = KMeans(k=8, initMode='k-means||', featuresCol='features', predictionCol='cluster', maxIter=10)

#Provide the training data to learning algorithm
kmModel = kmeans.fit(train_df)

#Print the KMeans Cluster Centers
print("\n\nKMeans Cluster Centers: ")
for center in kmModel.clusterCenters():
    print(center)
print("\n\n")
```

ML Example: K-Means Clustering

Output:

```
KMeans Cluster Centers:
[ 40.74866129 -73.98904373]
[ 40.77810344 -73.87209008]
[ 41.00389204 -73.73744751]
[ 40.65618264 -73.78139413]
[ 40.70061918 -74.20172325]
[ 40.70953786 -73.98922528]
[ 40.77697551 -73.96268636]
[ 40.86556678 -73.41761694]
```

Assisted Practice: Data Exploration

Duration: 10 mins

Problem Scenario: Perform a data exploration and a descriptive analysis of the US companies' dataset

Objective: In this demonstration, you will explore different commands to perform data exploration and descriptive analysis in PySpark.

Dataset Name: "Fortune 500 Companies US.csv"

Tasks to Perform:

Step 1: Create a directory named "ML" and upload the dataset into it

Step 2: Create a Spark Session, and then create a DataFrame from a CSV file to load data

Step 3: Print the loaded data and schema of DataFrame

Step 4: Perform descriptive analysis on data using describe command

Note: The solution to this assisted practice is provided under the course resources section.

Key Takeaways

- Apache Spark is an open-source unified analytics engine for large-scale data processing such as data engineering, data science, and machine learning on single-node machines or clusters.
- Analytics can be categorized as descriptive, predictive, and prescriptive analytics.
- Machine learning is a subset of artificial intelligence that uses historical data as input to predict new output values.
- Fraud detection, self-driving cars, smartphones, healthcare, and face detection are some of the trending applications of machine learning.

Key Takeaways

- In supervised learning, an algorithm is selected based on the target variable.
- Unsupervised learning looks for previously undetected patterns.
- Semi-supervised learning is a machine learning technique that falls between supervised and unsupervised learning and utilizes a combination of labeled and unlabeled datasets.
- ML Pipeline operates by enabling a sequence of data to be transformed and correlated in a model that can be tested and evaluated to achieve a positive or negative outcome in order to assist in the automation of machine learning workflows.

Knowledge Check

Knowledge Check 1

Which of the following skills are required to become a data scientist?

A. Knowledge of Python and R

B. Ability to work with unstructured data

C. Experience in SQL

D. All of the above

Knowledge Check
1

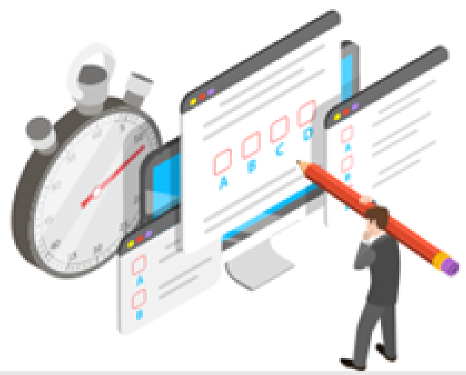
Which of the following skills are required to become a data scientist?

A. Knowledge of Python and R

B. Ability to work with unstructured data

C. Experience in SQL

D. All of the above



The correct answer is **D**

Knowledge of Python and R, the ability to work with unstructured data, and experience in SQL are required to become a data scientist.

Knowledge Check

2

Which of the following types of analytics describes the past and answers the question, “What happened?”?

A. Descriptive analytics

B. Predictive analytics

C. Prescriptive analytics

D. None of the above

Knowledge Check

2

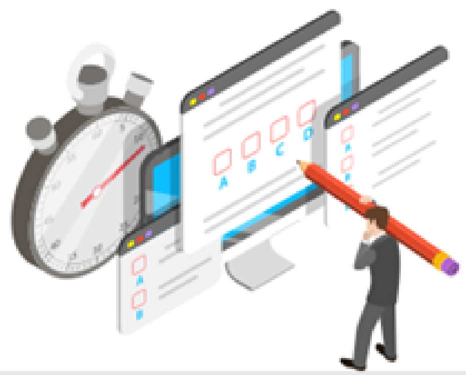
Which of the following types of analytics describes the past and answers the question, "What happened?"?

A. Descriptive analytics

B. Predictive analytics

C. Prescriptive analytics

D. None of the above



The correct answer is **A**

Descriptive analytics describes the past and answers the question, "What happened?".

Knowledge Check

3

Which machine learning algorithm develops a self-sustained system based on the interaction between the environment and the learning agent?

A. Supervised learning

B. Unsupervised learning

C. Reinforcement learning

D. None of the above

Knowledge Check

3

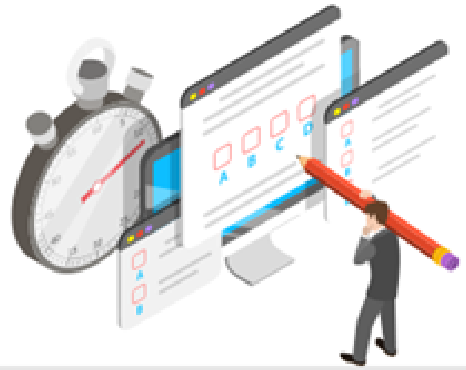
Which machine learning algorithm develops a self-sustained system based on the interaction between the environment and the learning agent?

A. Supervised learning

B. Unsupervised learning

C. Reinforcement learning

D. None of the above



The correct answer is **C**

Reinforcement learning algorithm develops a self-sustained system based on the interaction between the environment and the learning agent.

Knowledge Check 4

Which MLlib algorithm is a statistical process for estimating the relationships among variables?

A. Classification

B. Regression

C. Clustering

D. Optimization

Knowledge Check

4

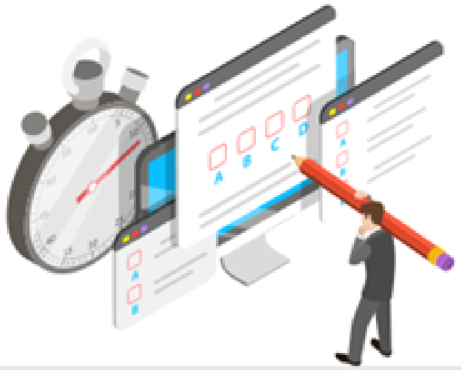
Which MLlib algorithm is a statistical process for estimating the relationships among variables?

A. Classification

B. Regression

C. Clustering

D. Optimization



The correct answer is **B**

Regression algorithm is a statistical process for estimating the relationships among variables.

simplilearn

Lesson-End Project: Linear Regression with Real-world Dataset

Problem Scenario:
Adam is working in an E-commerce company where customers can order products either from a mobile application or website. The company wants to know whether to focus its efforts on its mobile applications or website. Adam builds a model to analyze different features of customers in the real-world dataset and predict sales. He decided to use simple Linear Regression to perform this task.

Objective: The objective is to understand Spark Linear Regression with a real-world customer dataset.

Dataset Name: customers_ml

Lesson-End Project: Linear Regression with Real-world Dataset

Tasks to Perform:

1. Download the dataset “customers_ml“ folder from the course resource section
2. Create a directory in HUE named “data_files” and upload the dataset
3. Open the PySpark shell in “the Webconsole”
4. Import the required packages
5. Read the folder from the HDFS and display the 10 records of the dataset
6. Divide the input features and the label of the dataset to perform the Linear Regression
7. Initialize the Linear Regression model
8. Print the coefficients and intercept of the model with residuals
9. Print the root mean squared value, mean squared value, and r2 value

DATA AND
ARTIFICIAL INTELLIGENCE

Thank You

simplilearn