| **Module** | **1** | **Machine Learning Foundations & Data Preparation** |
| --- | --- | --- |
| **Lesson** | **1** | **Introduction to Machine Learning** |
| **Important links** | |  |

**Module Table of Contents**

**Insert TOC here when finished**

| <H1> X.0 Introduction |  |
| --- | --- |

| **Learning time estimation: 2 mins** |
| --- |

**<Module banner>**

Use this graphic and adapt size if necessary

<Module banner end>

**<H2> Welcome to Lesson 1:**

**<H2> Learning Objectives:**

After studying the material covered in this lesson, you should be able to:

1. Define machine learning and distinguish it from traditional programming approaches
2. Differentiate between supervised, unsupervised, self-supervised, and reinforcement learning paradigms
3. Identify real-world African use cases that map to each paradigm
4. Set up students learning environment for the course

## 

| <H1> X.1 Opening Content Video |  |
| --- | --- |

| **Learning time estimation: 4 mins** |
| --- |

Let's get started. First, please watch the following short video lecture. Just click on the image to view it in a new tab.

**Video Script:**

Did you know analysts project that artificial intelligence could inject an extra $15.7 trillion into the world economy by 2030—more than India and China produce today, and about five-and-a-half times Africa’s entire GDP.

My name is Moise, and welcome to Machine Learning Essentials —the course where you won’t just learn machine-learning theory; you’ll build production-ready solutions that matter here in Africa

Across 15 lessons grouped into four learning modules, we’ll move from data wrangling and model fundamentals to deployment today’s cutting-edge Machine Learning.”  
  
Each lesson ends with hands-on labs and auto-graded checkpoints, so every new idea gets anchored in practice.  
  
By the final module, you’ll deploy a live model behind an API that your classmates can actually call.

Throughout the course we’ll show you how to choose—and mix—these paradigms to fit the data you actually have

| <H1> X.2 Reading 1: What is Machine Learning |  |
| --- | --- |

| **Learning time estimation: 15 mins** |
| --- |

**<H2> Reading 1: What is Machine Learning**

Machine learning is not a recent invention. As early as 1959, Arthur Samuel described it as the field of study that gives computers the ability to learn without being explicitly programmed. This idea—letting machines improve through experience—laid the groundwork for decades of progress.

In 1997, Tom Mitchell offered a more formal and widely adopted definition. He stated:

“A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks in T, as measured by P, improves with experience E.”

Mitchell’s wording gives us a clear four-part test for deciding whether something really *is* machine learning.

| **Mitchell Criterion** | **Diagnostic Question** | **Illustrative Example** |
| --- | --- | --- |
| **Task (T)** | What specific job should the program accomplish? | “Label each incoming e-mail as *spam* or *not-spam*.” |
| **Performance Measure (P)** | How will we quantify success on that job? | “Count how many emails it classifies correctly.” |
| **Experience (E)** | What past data or feedback will it learn from? | “A set of 50 000 emails already tagged spam or not-spam.” |
| **Improvement** | Does its performance metric rise as it sees more experience? | “Accuracy improves from 80 % to 92 % after retraining on new data.” |

**<H2> Your next activity**

Continue to take a quiz to help you reinforce what you learned.

For each scenario below, decide whether the system counts as **machine learning** under Mitchell’s criteria

| **Scenario** | **Description (as deployed today)** | **ML (y/n)** |
| --- | --- | --- |
| **A · Rule-Based Thermostat** | An industrial thermostat follows two fixed fuzzy rules (“IF temperature > X THEN cool”) set by an engineer. | No |
| **B · Reactive Obstacle Robot** | A mobile robot reads proximity sensors. If an obstacle is ahead it checks left; if that side is clear it turns right, | No |
| **C · Sender-List Spam Filter** | An e-mail gateway blocks any message whose sender appears in a static blacklist | No |
| **D · Traffic-Count GPS Router** | A car navigator queries roadside sensors for the current number of cars and directs you along the route with the lowest count. | No |
| **E. Personalized News Recommender** | Suggests articles based on what the user has read and liked in the past | Yes |
| **F.. Predictive Maintenance System** | Analyzes equipment sensor data and failure logs to predict when a machine is likely to break down | Yes |

Feedback for the quiz:

| **Scenario** | **Correct Answer** | **Feedback (if Correct)** | **Feedback (if Incorrect)** |
| --- | --- | --- | --- |
| **A. Rule-Based Thermostat** | **No** | Correct! This system uses fixed rules and does not involve any **learning**. Its behavior never changes based on experience. | Incorrect. This is not machine learning because the system does **not learn** from temperature patterns or past performance. It always follows the same rules. |
| **B. Reactive Obstacle Robot** | **No** | Correct! This robot follows hardcoded instructions and does not engage in any form of **learning** from past obstacles. | Incorrect. While it reacts to obstacles, it does not **learn** from past experiences. A learning system would improve its navigation over time. |
| **C. Sender-List Spam Filter** | **No** | Correct! This system blocks emails based on a static list. It doesn’t adapt or **learn** from new messages or user actions. | Incorrect. Machine learning involves adapting based on data—this system does **not learn** from feedback or new spam examples. |
| **D. Traffic-Count GPS Router** | **No** | Correct! Although it uses live data, the system doesn’t **learn** from past trips. It simply reacts to current traffic counts. | Incorrect. This system doesn’t involve **learning**. It doesn’t get better at predicting routes over time, which ML systems would. |
| **E. Personalized News Recommender** | **Yes** | Correct! This system uses past user behavior to **learn** what content to recommend. Over time, it gets better at predicting what you’ll like. | Incorrect. This is machine learning because the system **learns** from your reading habits and improves its recommendations based on that experience. |
| **F. Predictive Maintenance System** | **Yes** | Correct! It’s a clear case of machine learning—the system **learns** from historical data to better predict failures and reduce downtime. | Incorrect. This system qualifies as machine learning because it **learns** patterns from past sensor data and uses that knowledge to improve its predictions. |

For each system that are not examples of Machine Learning, imagine the extra data or feedback it would need—and the changes in behaviour you’d expect—to *qualify* as machine learning.

When you’re ready, compare your ideas with the example upgrades below.

| **Original System** | **Possible ML Upgrade** | **What Changes?** |
| --- | --- | --- |
| **Thermostat** Fixed temperature thresholds never change. | **Smart Thermostat** — logs hourly indoor/outdoor readings and energy use, reviews them each night, and adjusts tomorrow’s cooling schedule. | Learns from its own history; comfort stays stable while power use steadily drops. |
| **Obstacle Robot** Preset left/right turns based on proximity sensors. | **Learning Robot** — records sensor readings, chosen turns, and collisions on every trip, then tweaks its turning choices after each run. | Uses past encounters to cut collision rate over time. |
| **Spam Filter** Blocks senders on a static list. | **Self-Updating Filter** — stores every “Spam / Not spam” click, and uses the information to refine its decision rules. | Accuracy climbs as it digests new user feedback. |
| **Traffic GPS** Picks the route with the fewest cars in live sensor data. | **Getting-Smarter GPS** — after each drive it saves the route and actual travel times, then adjusts tomorrow’s travel-time estimates. | Arrival predictions tighten and average commute times shrink as more trips are logged. |

The table below summarizes the distinction between rule-based systems—which include both traditional programming and non-learning AI—and machine learning, which allows systems to learn and adapt from data.

| **Aspect** | **Rule-Based Systems(*Traditional Programming & Non-ML AI*)** | **Machine Learning (ML)** |
| --- | --- | --- |
| **Approach** | Uses fixed rules and logic defined by humans | Learns patterns and rules from data |
| **Adaptability** | Static — requires reprogramming to change | Dynamic — adapts and improves with experience |
| **Knowledge Source** | Encoded by programmers or domain experts | Derived from training data (examples) |
| **Examples** | Sorting algorithms, if-then logic, expert systems, rule-based chatbots | Spam filters, recommendation engines, self-driving cars |
| **Type of Intelligence** | Symbolic / logic-based (if any) | Statistical / data-driven |
| **Limitations** | Struggles with ambiguity or unseen scenarios | Requires lots of data, can be a black box |

## 

| <H1> X.3 Reading 2: Data at the Core of Machine Learning |  |
| --- | --- |

| **Learning time estimation: 10 mins** |
| --- |

**<H2> Reading 2: Data at the Core of Machine Learning**

Earlier, we defined machine learning as a way for computers to **learn from experience** and improve at a task without being programmed with hard rules.

But what does that look like in practice?

We can think of any machine learning system as having **three key components**:

1️. **Input** – the data you feed in  
2. **Algorithm** – the process that learns patterns from the data  
3. **Output** – the prediction or decision the system produces

[Represent as a diagram]: Machine Learning = Data (Input) + Algorithm(Model/Learning Process) → Prediction (Output)

The **Input** is where everything starts. It’s the *experience* the algorithm learns from.

Here, *data* means **lots of examples** of the thing we want the computer to learn about. An **example** is a single observation or case (like one row in a dataset). For instance:

* One email message
* One photo of a plant leaf
* One patient’s medical record

Each example usually has:

* **Features (Inputs):** The measurable properties that describe the example. For an email, these might be the words it contains. For a photo, the pixel values. For a patient record, age, symptoms
* **Label (Output):** The *answer* we want the model to learn to predict. Spam or not-spam? Diseased or healthy? Diagnosis present or absent?

By learning from many examples, the algorithm finds patterns that map **features** to **labels**, so it can make accurate predictions on new, unseen data.

We’ll dedicate an entire lesson later to preparing, cleaning, and organizing this data. For now, remember: your data is the starting point—and the kind of data you have shapes what kind of learning you can do.

## 

| <H1> X.4 Reading 3: The Four Learning Paradigms |  |
| --- | --- |

| **Learning time estimation: 15 mins** |
| --- |

**<H2> Reading 3: The Four Learning Paradigms**

Machine learning problems come in many shapes—but they often boil down to *what kind of data you have* and *what kind of learning signal you can use*. In the next section we will introduce 4 major learning paradigms:

### **Supervised Learning**

Supervised learning is the most widely used paradigm.

**Definition:**

The algorithm learns to map inputs (features) to outputs (labels) by seeing many examples of correct input–output pairs.

**Key characteristic:**

* You need a labeled dataset: each example has known features and the correct answer.

**Example:**

* **Crop Disease Diagnosis:** Farmers photograph diseased leaves. Each image is labeled with the correct disease. A model learns to map new images to the correct disease
* **Credit Risk Prediction:** Banks train models on historical loan data labeled as “paid” or “default” to predict if a new loan will be paid or not.

The ultimate goal for supervised learning is to automate accurate prediction on new, unseen examples.

**[A graphic here: [Frame 1 – 1 Observation]  
 “Here’s our very first example: an input value on the x-axis paired with its true output on the y-axis. With just this one point, our model draws an initial guess—a simple straight line through that example.”**

**[Frame 2 – 2 Observations]  
 “Now a second data point arrives. The line shifts to accommodate both observations, but it still misses the true trend. Each labeled example nudges the model closer to the underlying relationship.”**

**[Frame 3 – 5 Observations]  
 “With five examples in its notebook, the model refits the line again. It’s beginning to capture the positive correlation—higher x values tend to yield higher y values.”**

**[Frame 4 – 10 Observations]  
 “Ten points give a clearer picture. The line now balances more of the data’s ups and downs, reducing its errors on average.”**

**[Frame 5 – 25 Observations]  
 “Twenty-five pairs of (x, y) reveal subtler patterns. The model’s estimate improves noticeably, as the line hugs the scatter more tightly.”**

**[Frame 6 – 50 Observations]  
 “At fifty examples, outliers have less pull on the fit. The line keeps adjusting to minimize the overall mismatch between its predictions and the true outputs.”**

**[Frame 7 – 75 Observations]  
 “Seventy-five points give us a robust view of the trend. The model is now very close to the real underlying function.”**

**[Frame 8 – 100 Observations + Initial Line]  
 “Finally, with all one hundred examples, the line settles in its best position. Notice the dashed line from Frame 1—our rough first guess—versus the solid line learned from every data point. This side-by-side shows how supervised learning works: we feed in input–output pairs, measure how far off our predictions are, and update the model so its errors shrink as more labeled examples arrive.”]**

**Self-Supervised Learning**

Sometimes perceived as a subset of supervised learning, self-supervised is an innovative way to **create labels automatically from the data itself**.

**Definition:**

The algorithm sets up prediction tasks within the raw data, using part of the data to predict another part.

**Key characteristic:**

* No manual labeling required; labels are generated automatically.

**Example:**

* **African Language Models:** Training a model to predict the next word in Swahili or Yoruba text. The model learns grammar, semantics, and context without human-annotated labels.
* **Satellite Imagery Pre-training:** Masking parts of an image and training the model to fill in the missing region, learning general features useful for land-use classification later.

The ultimate goal for self-supervised learning is to learn rich representations from large amounts of raw data.

**[GRAPHIC SUGGESTION:** Animation of Image with missing section → model predicts missing section/ Or text with blanked-out words → model fills in.]

### **Unsupervised Learning**

Unsupervised learning is about discovering hidden patterns in **unlabeled data**.

**Definition:**

The algorithm analyzes data with no explicit labels, grouping or structuring it in meaningful ways.

**Key characteristic:**

* No labels provided. The model must find structure on its own.

**Example:**

* **Customer Segmentation:** A mobile-money provider analyzes transaction histories to cluster users into distinct groups for personalized marketing.
* **Urban Planning:** Analyzing GPS movement patterns in a city to identify natural traffic clusters and plan transport routes.

The ultimate goal for unsupervised learning is to reveal structure or relationships in the data that weren’t obvious.

**[GRAPHIC:** Data points on a scatter plot grouped into colored clusters automatically by algorithm.]

**Reinforcement Learning**

Reinforcement learning is about **learning through interaction** with an environment.

**Definition:**

An agent takes actions in an environment, gets feedback as rewards or penalties, and learns a policy to maximize long-term reward.

**Key characteristic:**

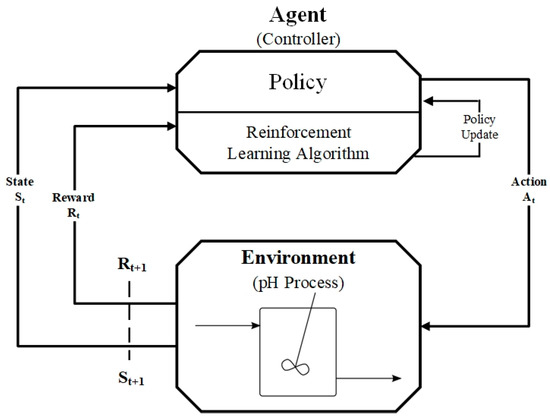
* No fixed dataset; learning happens through trial and error.

**African-context example:**

* **Traffic Signal Optimization:** A system adjusts traffic light timings to minimize congestion in Nairobi or Lagos, learning from live traffic feedback.
* **Drone Delivery:** A delivery drone learns efficient routes while avoiding obstacles in variable urban environments.

The ultimate goal for reinforcement learning is to develop strategies that perform well over time in dynamic environments.

**[GRAPHIC SUGGESTION:** Agent → Action → Environment → Reward loop diagram. And a animation of agent (robot learning to walk iteratively)



**COMPARISON TABLE**

| **Paradigm** | **Data Requirement** | **Learning Approach** | **African Example** |
| --- | --- | --- | --- |
| Supervised | Labeled examples | Learn mapping from features to labels | Diagnosing crop diseases from labeled photos |
| Unsupervised | Unlabeled data | Find structure or groups in the data | Segmenting mobile-money users by spending behavior |
| Self-Supervised | Unlabeled data (auto-labeled) | Predict part of data from other parts | Training language models for local African languages |
| Reinforcement | Interaction with environment | Learn by trial and error to maximize reward | Optimizing traffic light control in urban African cities |

*Throughout this course, you’ll see these paradigms applied to real-world challenges in Africa. You’ll learn not only what they are, but* ***how to choose and apply them effectively to the data you have.***

**<H2> Your next activity**

For each scenario, review the *Goal* and *Data Available* columns and select the single machine-learning paradigm that best fits: **Supervised, Unsupervised, Self-Supervised, or Reinforcement Learning**.

| **Goal – build a model / system to…** | **Data Available** | **Answer** |
| --- | --- | --- |
| Identify the disease type in cassava-leaf photos | 10 000 RGB images, each already tagged with one of six diseases | Supervised |
| Segment mobile-money subscribers into clusters for personalized offers | 3 million anonymized transaction histories with no labels | Unsupervised |
| Suggest the next word while users type in Hausa | 200 MB corpus of raw Hausa sentences | Self-Supervised |
| Enable a drone to land safely on uneven terrain | Physics-based simulator streaming state feedback after each landing attempt | Reinforcement |
| Detect previously unmapped informal settlements in satellite imagery | 10 000 high-resolution satellite images without labels | Unsupervised |
| Predict hidden pixels in Landsat tiles to learn useful image features | 100 GB of unlabeled Landsat imagery | Self-Supervised |
| Forecast whether a micro-loan will be repaid or default | 50 000 historical loan records, each marked “repaid” or “default” | Supervised |
| Discover effective kicking force and angle for a soccer robot during matches | Real-time sensor streams from the robot in play | Reinforcement |
| Flag Swahili SMS messages as spam or legitimate | 12 000 SMS texts labeled spam / not-spam | Supervised |
| Reduce queue length at busy intersections in Accra through signal control | Live traffic-sensor streams and access to a city-traffic simulator; no historical dataset | Reinforcement |
| Spot unusual vibration patterns that may signal rural water-pump failure | Six months of multivariate sensor time-series with no failure labels | Unsupervised |
| Predict the next reply in customer-service chat logs | 500 000 anonymized multi-turn chats (unlabeled) | Self-Supervised |
| Adjust ride-hailing fares in real time to balance rider demand and driver supply | Marketplace simulator streaming supply-and-demand metrics after each price change; no historical dataset | Reinforcement |
| Cluster retail product photos by visual similarity | 40 000 product images with no category labels | Unsupervised |
| Estimate next-day rainfall totals at weather stations | 20 years of daily weather-station records, each with measured rainfall | Supervised |
| Plan efficient, obstacle-free cleaning paths for an office robot | On-board lidar and bump-sensor feedback plus task-completion signals; no pre-collected dataset | Reinforcement Learning |

## 

| <H1> X.5 Lab: Exploring Machine Learning Paradigms with African Data |  |
| --- | --- |

| **Learning time estimation: 20 mins** |
| --- |

**Objective**: Experience all four ML paradigms using African-contextualized datasets

**Activity Structure**:

* Setup (3 minutes):
  + Environmental set up
* Supervised Learning Exercise (5 minutes):
  + Load a pre-prepared dataset
  + Choose the input and the output
  + Choose the learning type
  + Modify parameters and observe prediction changes
  + Train the model to learn the best parameter
* Unsupervised Learning Exercise (4 minutes):
  + Same as above with discrete classes
* Self-Supervised Learning Exercise (4 minutes):
  + Masked word prediction
* Reinforcement Learning Exercise (4 minutes):
  + Learn the policy

## 

| <H1> X.6 Did I get it? Time for a quiz to make sure. |  |
| --- | --- |

| **Learning time estimation: xxx mins** |
| --- |

**Q1:**   
According to Tom Mitchell's definition, which of the following is NOT one of the required components for a system to qualify as machine learning?

1. Task (T) - What specific job should the program accomplish?
2. Performance Measure (P) - How will we quantify success on that job?
3. Experience (E) - What past data or feedback will it learn from?
4. Algorithm (A) - What specific computational method will be used?

**Correct Answer: D**

**Feedback for Correct Answer:** Excellent! Mitchell's definition focuses on three key components: Task, Performance measure, and Experience. The specific algorithm used is not part of his formal definition - what matters is that the system improves its performance on a task through experience.

**Feedback for Wrong Answers:** Not quite. Mitchell's definition requires three components: Task (T), Performance measure (P), and Experience (E). The improvement in performance through experience is what makes it machine learning, regardless of the specific algorithm used.

**Q2:**

A mobile-money provider in Kenya wants to group their users into different categories based on spending patterns to offer personalized services. They have transaction data but no predetermined categories. Which learning paradigm would be most appropriate?

1. Supervised Learning
2. Unsupervised Learning
3. Self-Supervised Learning
4. Reinforcement Learning

**Correct Answer: B**

**Feedback for Correct Answer:** Correct! This is a classic unsupervised learning problem. Since there are no predetermined categories (no labels), the algorithm needs to discover hidden patterns and group users based on their spending behavior. Unsupervised learning excels at finding structure in unlabeled data.

**Feedback for Wrong Answers:** This scenario involves finding hidden patterns in data without predetermined categories, which is the hallmark of unsupervised learning. The provider wants to discover natural groupings in spending behavior, not predict specific labels or learn through trial and error.

**Q3:**

A system is being developed to predict the next word in Swahili text messages to help with autocomplete features. The model learns by reading millions of Swahili sentences and trying to predict missing words. Which learning paradigm does this represent?

1. Supervised Learning
2. Unsupervised Learning
3. Self-Supervised Learning
4. Reinforcement Learning

**Correct Answer: C**

**Feedback for Correct Answer:** Perfect! This is self-supervised learning. The system creates its own labels by using part of the text to predict other parts (predicting missing or next words). No human annotation is needed - the labels are generated automatically from the structure of the data itself.

**Feedback for Wrong Answers:** This is self-supervised learning because the system generates its own training labels from the data structure. It uses part of the text (context) to predict other parts (missing words) without requiring human-annotated labels or external feedback.

**Q4:**

Which of the following systems would currently qualify as machine learning according to Mitchell's criteria?

1. A thermostat that follows fixed rules: "IF temperature > 25°C THEN turn on AC"
2. A robot that always turns left when it detects an obstacle ahead
3. An email filter that learns from user "spam/not spam" clicks to improve its accuracy over time
4. A GPS that always chooses the route with the current lowest car count from traffic sensors

**Correct Answer: C**

**Feedback for Correct Answer:** Exactly right! The email filter meets all of Mitchell's criteria: it has a clear task (classify emails), a performance measure (classification accuracy), experience to learn from (user feedback), and it improves over time. The other systems follow fixed rules without learning from experience.

**Feedback for Wrong Answers:** The email filter is the only system that learns from experience and improves over time. The other options follow predetermined rules or logic without adapting based on past experience, so they don't meet Mitchell's criteria for machine learning.

**Q5:**

A research team wants to develop a system that learns to optimize traffic light timing in Lagos to reduce congestion. The system will try different timing patterns, observe traffic flow results, and adjust its strategy based on whether congestion improves or worsens. Which learning paradigm fits this scenario?

1. Supervised Learning
2. Unsupervised Learning
3. Self-Supervised Learning
4. Reinforcement Learning

**Correct Answer: D**

**Feedback for Correct Answer:** Excellent! This is reinforcement learning. The system acts as an agent that takes actions (adjusting traffic light timing), receives feedback from the environment (traffic flow results), and learns to maximize rewards (reduced congestion) through trial and error over time.

**Feedback for Wrong Answers:** This scenario describes reinforcement learning, where an agent (traffic system) learns through interaction with an environment (traffic), taking actions (timing adjustments) and receiving rewards/penalties (congestion changes) to improve its strategy over time.

## 

| <H1> X.7 Lesson Summary |  |
| --- | --- |

| **Learning time estimation: 1 min** |
| --- |

Let's summarize the main points in this lesson.

1. Machine Learning enables computers to learn from experience and improve at tasks without being explicitly programmed, following Mitchell's criteria of Task (T), Performance measure (P), and Experience (E).
2. Every Machine Learning system has three essential parts:
   1. **Input** (data/features)
   2. **Algorithm** (learning process)
   3. **Output** (predictions/decisions)
3. The four learning paradigms include:
   1. **Supervised Learning:** Learns from labeled examples to map inputs to outputs (e.g., crop disease diagnosis from labeled photos)
   2. **Unsupervised Learning:** Discovers hidden patterns in unlabeled data (e.g., mobile money user segmentation)
   3. **Self-Supervised Learning:** Creates its own labels from data structure to learn representations (e.g., African language models predicting missing words)
   4. **Reinforcement Learning**: Learns through trial-and-error interaction with environment (e.g., optimizing traffic lights in African cities)
4. The type and quality of data you have determines which learning paradigm you can use and what kind of problems you can solve
5. Each paradigm has relevant applications across African contexts - from healthcare and agriculture to finance and urban planning
6. Success depends on matching the right learning approach to your specific data situation and problem type

## 

| <H1> X.8 Further Resources |  |
| --- | --- |

**<H2> Recommended Reading**

Mitchell, Tom M. (1997). Machine Learning. New York: McGraw-Hill.

**<H2> References**

Here are some of the references cited to support the main points in what we covered in this lecture.