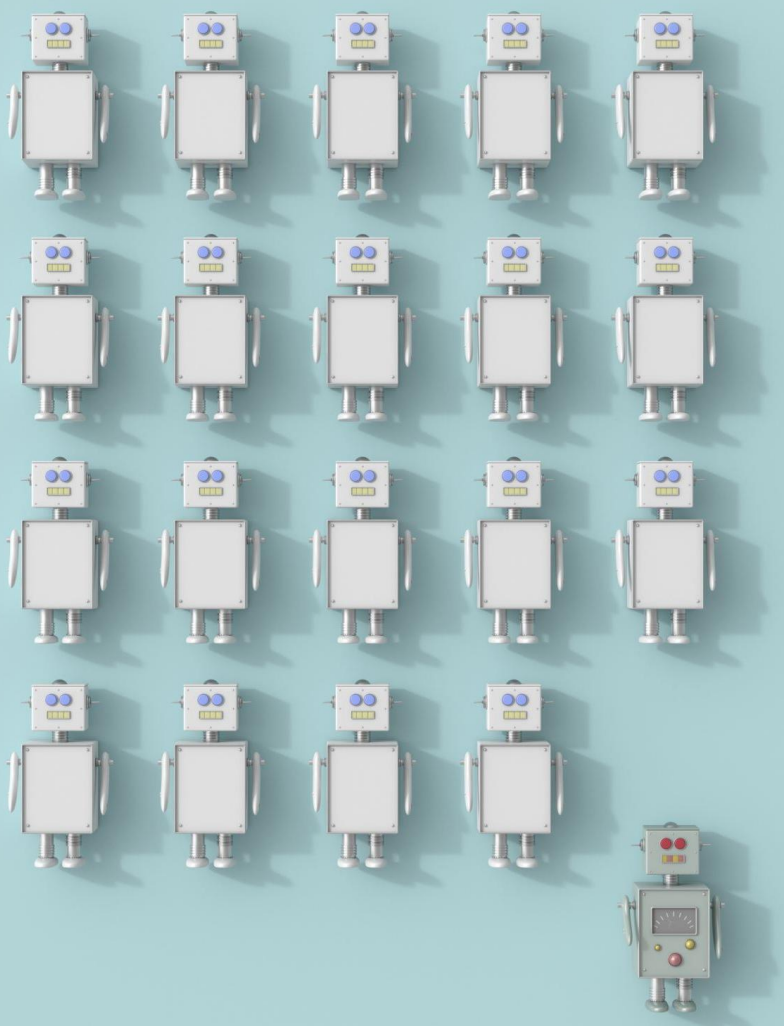
GROUP 3

- Judith Otieno
- Kiprono Langat
- Norman Mwapea
- Pauline Kariuki
- Alvin Kipleting
- Wesley Owino

Ice Breaker

- If you had to select 3 electronic devices to use for the rest of your life, which ones will you choose?



Concepts Covered

- **Random Forest**
- **XGBoost**
- **Random Forest vs. XGBoost**
  (Key Differences When Using Decision Trees

Random Forest Overview

What is a Random Forest?

- An **ensemble learning method** that combines multiple decision trees to make more accurate and stable predictions.
- Used for both **classification** (categorical labels) and **regression** (continuous values).
- Helps reduce **overfitting** by averaging across diverse trees rather than relying on one.

Key Concepts:

- **High variance** in single decision trees is reduced by combining many.
- Final prediction is made by **majority vote** (classification) or **average** (regression).
- Implements the **“wisdom of the crowd”**—many models are better than one.

How It Works – Bootstrapping & Bagging

Bootstrapping (Random Sampling with Replacement):

- Each tree is trained on a **random sample** (with replacement) from the dataset.
- Some data points may appear multiple times; others may be excluded (called **out-of-bag** samples).
- OOB samples are used to internally estimate model performance.

Bagging (Bootstrap Aggregating):

1. **Resample:** Generate multiple bootstrap datasets.
2. **Train:** Build an independent tree for each dataset.
3. **Aggregate:** Combine predictions from all trees:
    - **Majority vote** for classification
    - **Average** for regression

Example (Classification):

Trees vote: *Spam, Spam, Not Spam, Spam, Not Spam* → Final Prediction: **Spam**

XGBoost - Overview

What is XGBoost?

- **Extreme Gradient Boosting** – powerful, fast, and accurate ML algorithm.
- Widely used in **classification, regression**, and data science competitions.

How It Works (Boosting):

- Builds **trees sequentially** – each new tree fixes the previous tree's errors.
- Final prediction = **sum of all tree outputs** → strong model from weak learners.
- Analogy: **Tutors helping a student** improve by focusing on past mistakes.

Boosting vs. Bagging:

Aspect	Bagging (Random Forest)	Boosting (XGBoost)
Training	Parallel, independent	Sequential, dependent
Goal	Reduce variance	Reduce bias
Analogy	Independent doctors	Chain of tutors
Final Output	Vote/Average	Weighted sum

Why XGBoost Works So Well

Key Components:

- **Objective = Loss + Regularization**
  - *Loss:* Measures prediction error
  - *Regularization:* Prevents overfitting → simpler models

Innovations:

1. 2nd-order derivatives (Hessians):

Improve learning speed & precision

Analogy: Gradient = direction, Hessian = slope steepness

2. Built-in Regularization:

Keeps models simple and generalizable

3. Handles Missing Data:

Automatically manages gaps without preprocessing

4. Fast & Scalable:

Parallelized tree construction and optimized memory use

Random Forest vs. XGBoost (Key Differences When Using Decision Trees)

Feature	Random Forest	XGBoost
Method	Bagging	Boosting
Tree Building	Parallel, independent	Sequential, dependent
Data Sampling	Bootstrapped subsets	Full dataset, weighted
Combines Outputs	Majority vote / average	Weighted sum of predictions
Goal	Reduce variance	Reduce bias & error