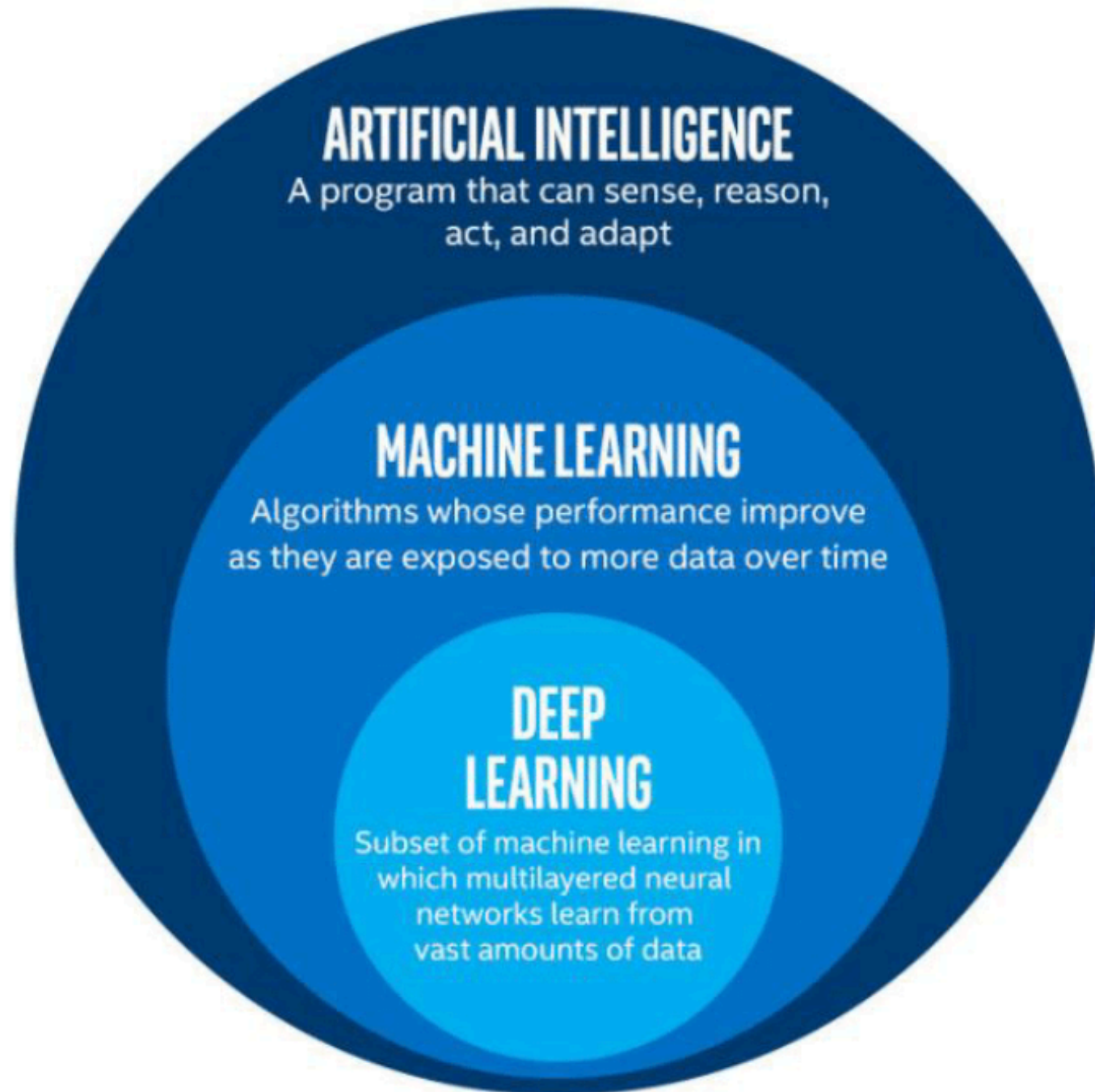


Machine learning Overview

https://github.com/as-budi/Embedded_AI.git

AI



1. Artificial Intelligence (AI)

- Bidang ilmu komputer yang berfokus pada pengembangan sistem yang mampu meniru kecerdasan manusia untuk melakukan tugas-tugas seperti pemecahan masalah, pengenalan pola, pengambilan keputusan, dan pemrosesan bahasa alami.

2. Machine Learning (ML)

- AI yang belajar dari data dan memperbaiki performanya berdasarkan pengalaman tanpa diprogram secara eksplisit.
- Contoh: Algoritma klasifikasi, regresi, clustering.

3. Deep Learning (DL)

- Subset dari ML yang menggunakan jaringan saraf tiruan (Artificial Neural Networks) dengan banyak lapisan (deep networks).
- Contoh: CNN untuk pengenalan gambar, RNN/LSTM untuk pemrosesan bahasa alami.

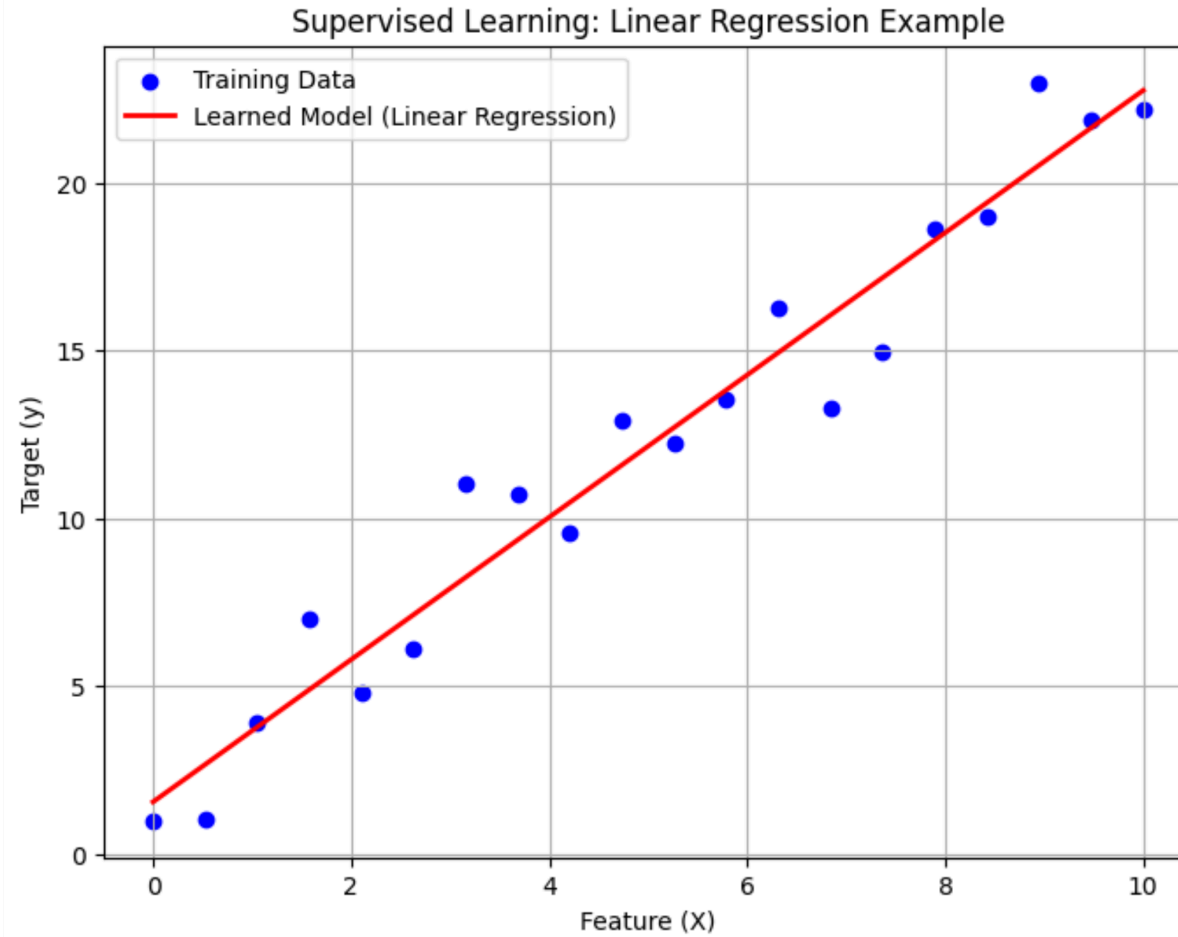
Tipe Pembelajaran dalam Machine Learning

1. Supervised Learning (Pembelajaran Terawasi)

- Model dilatih menggunakan data berlabel.
- Contoh: Klasifikasi objek dalam gambar (contoh: mengidentifikasi apakah gambar mengandung mobil atau tidak).
- Algoritma: Decision Tree, Support Vector Machine (SVM), Neural Network.

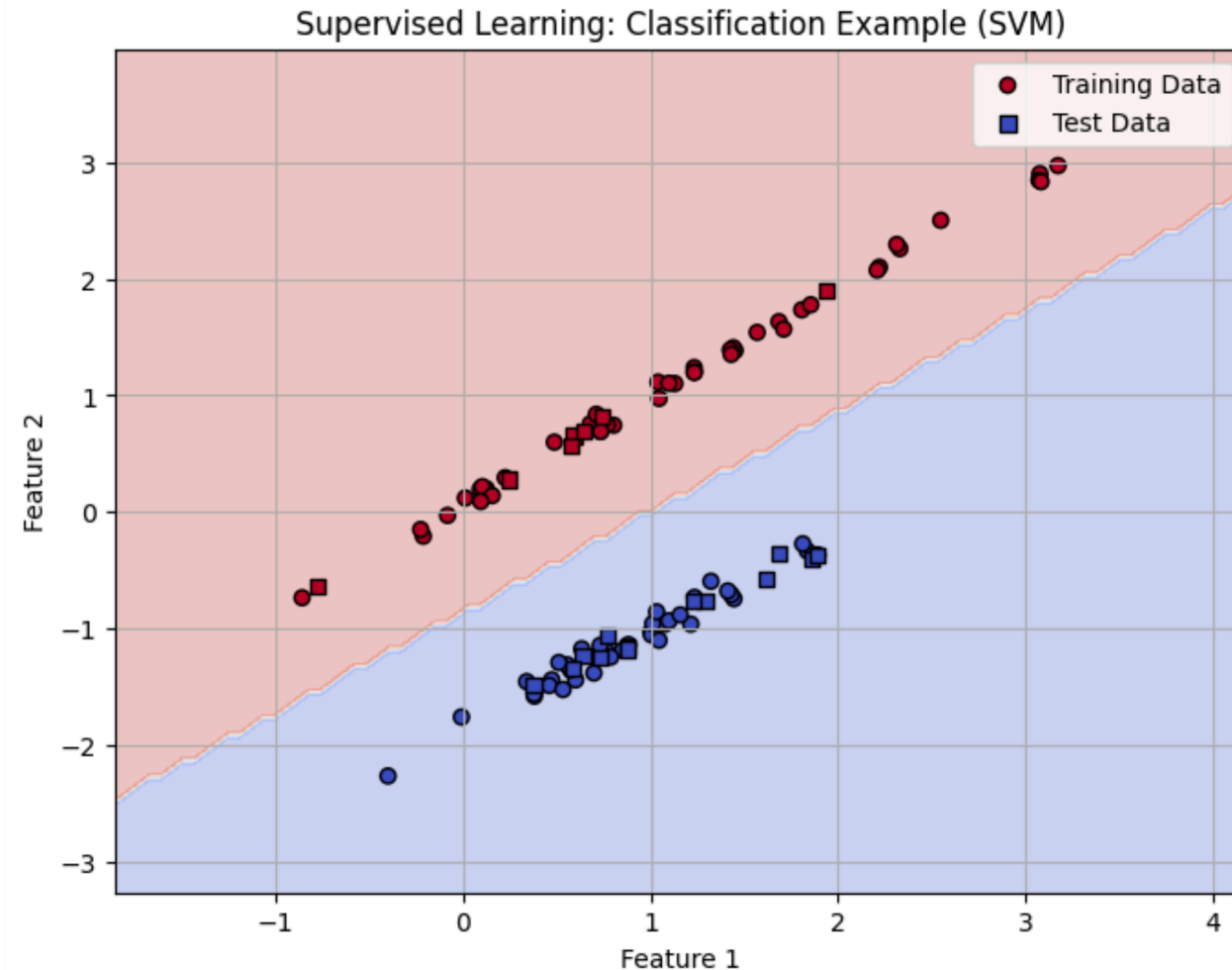
Contoh supervised learning (Linear regression)

X	y
0	0.99343
0.526315789	1.03926
1.052631579	3.92696
1.578947368	6.99343
2.105263158	4.79485
2.631578947	6.11067
3.157894737	11.0532
3.684210526	10.7454
4.210526316	9.58737
4.736842105	12.9272
5.263157895	12.2311
5.789473684	13.5422
6.315789474	16.2734
6.842105263	13.2787
7.368421053	14.9712
7.894736842	18.6123
8.421052632	19.027
8.947368421	22.9969
9.473684211	21.8682
10	22.1754



Contoh supervised learning (classification)

0	1	y
1.228312	-0.75718	0
0.698409	-1.3803	0
2.548817	2.502258	1
0.573579	-1.35298	0
0.5859	-1.33746	0
0.633562	-1.17279	0
0.556563	-1.31038	0
0.124372	0.198967	1
1.322721	-0.5934	0
1.425869	1.39003	1
0.639441	-1.23395	0
1.234995	-0.73169	0
0.739028	0.821807	1
2.221406	2.100366	1
3.084021	2.833249	1
1.001831	-1.02647	0
1.033074	-0.85546	0
0.103207	0.217912	1

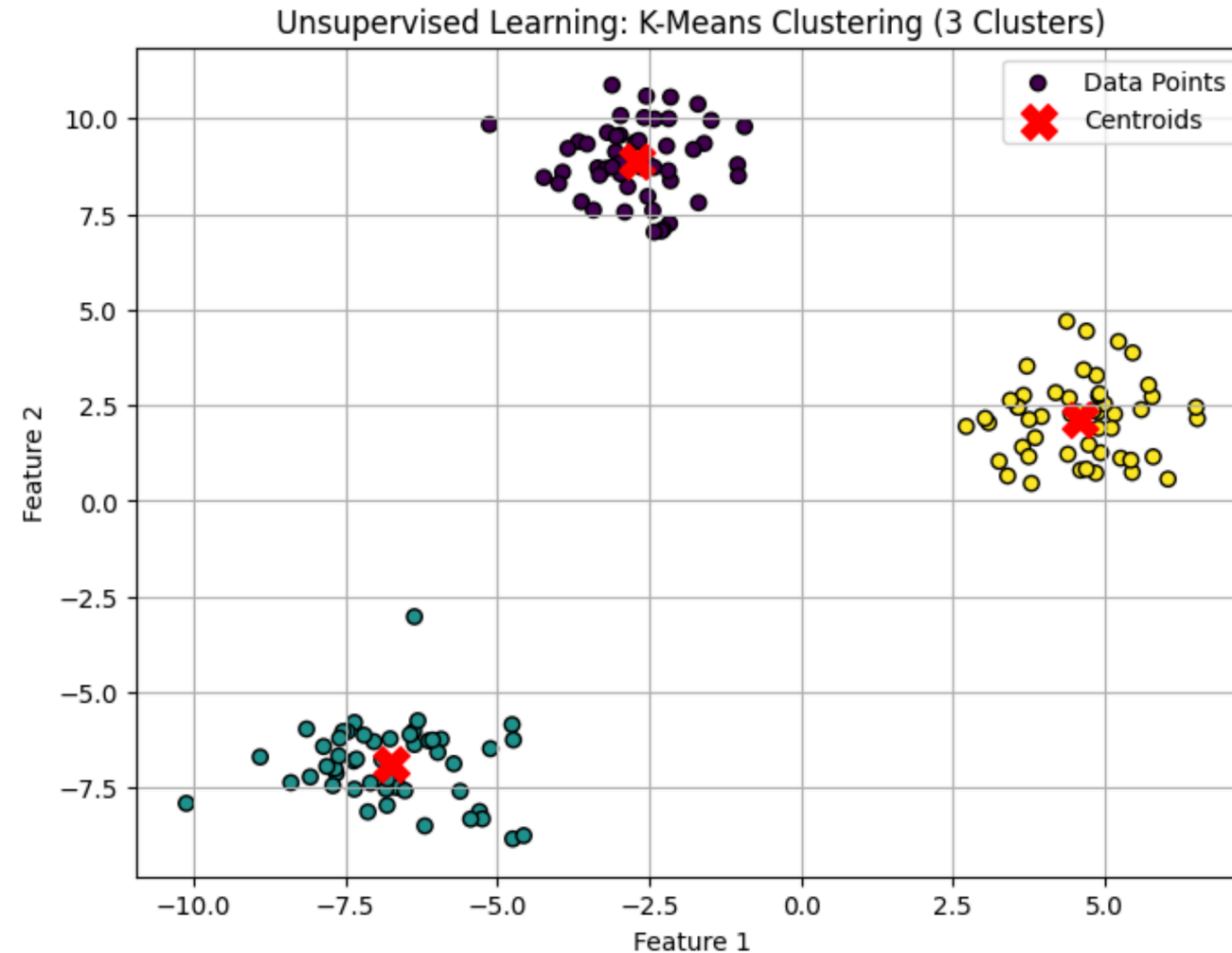


2. Unsupervised Learning (Pembelajaran Tak Terawasi)

- Model mencari pola dalam data tanpa label.
- Contoh: Clustering data sensor untuk mendeteksi pola anomali dalam perangkat IoT.
- Algoritma: K-Means, DBSCAN, Autoencoder.

Contoh unsupervised learning (clustering)

Feature 1	Feature 2
-2.545024	10.578930
-3.348415	8.705074
-6.598635	-7.502809
6.526065	2.147747
-6.364592	-6.366324
5.265546	1.116012
-3.186120	9.625962
-2.978672	9.556846
-7.351559	-5.791159
3.793085	0.458322
-4.757471	-5.847644
-7.132195	-8.127893
4.605167	0.804492
-6.766110	-6.217979
3.665197	2.760254
-3.924568	8.593641
-6.680567	-7.480326
-2.504084	8.779699
4.890372	2.319618

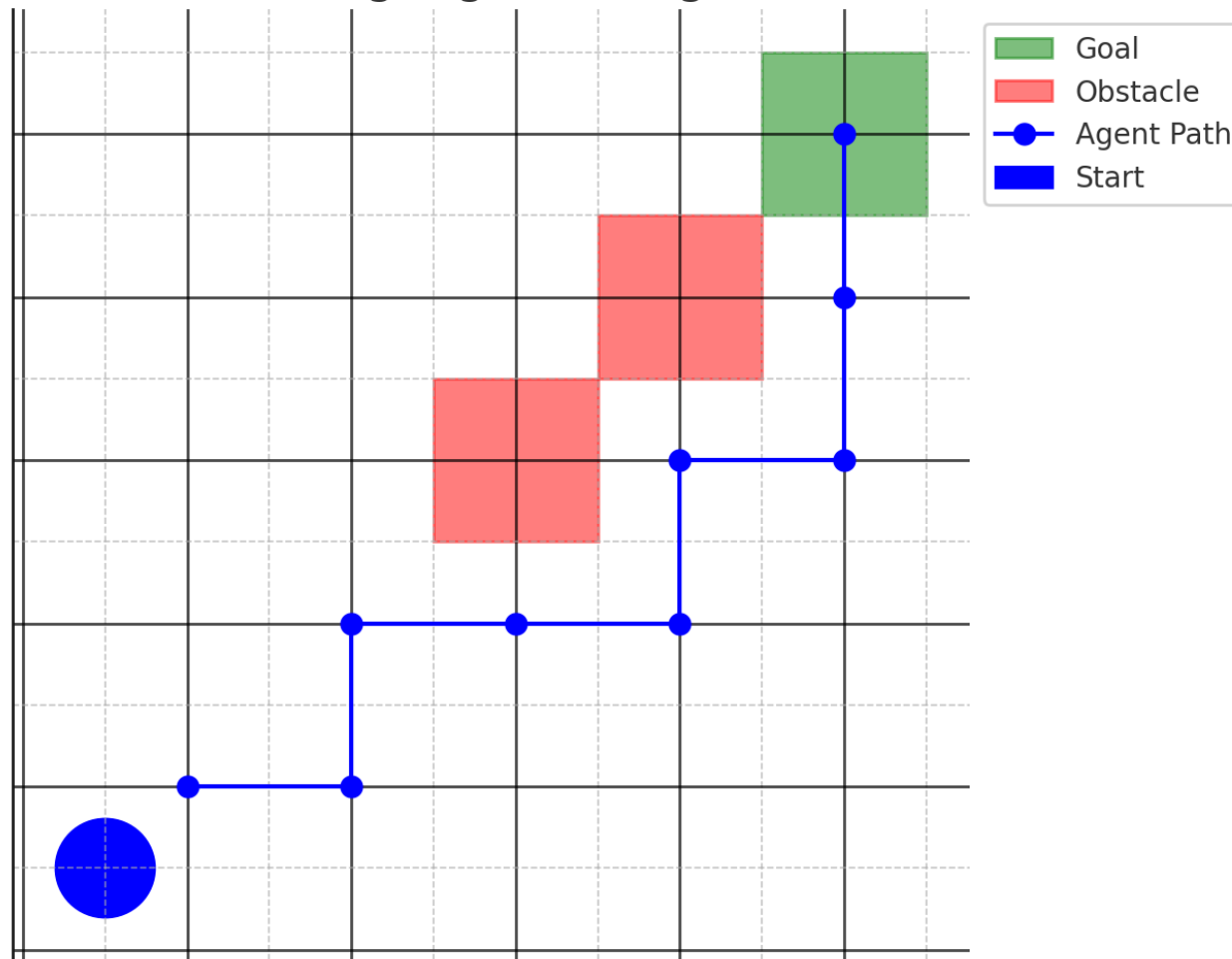


3. Reinforcement Learning (Pembelajaran Penguatan)

- Model belajar melalui trial and error untuk memaksimalkan reward.
- Contoh: Pickup and carrying robot
- Algoritma: Q-Learning, Deep Q-Network (DQN).

Contoh reinforcement learning

Reinforcement Learning: Agent Navigation in Grid World



Use Cases dalam Embedded AI

- **Classification (Klasifikasi)**

- Contoh: Identifikasi suara manusia dalam perangkat smart home.
- Model yang digunakan: CNN (Convolutional Neural Network) untuk pengenalan pola dalam gambar atau suara.

- **Clustering (Pengelompokan)**

- Contoh: Deteksi anomali pada sistem pemantauan kondisi mesin menggunakan data sensor.
- Model yang digunakan: K-Means atau DBSCAN untuk mendeteksi pola yang tidak biasa.

- **Prediction (Prediksi)**

- Contoh: Perkiraan waktu kedatangan kendaraan berdasarkan data GPS dan sensor lingkungan.
- Model yang digunakan: LSTM (Long Short-Term Memory) atau XGBoost untuk analisis data deret waktu.

Tantangan Machine Learning pada Perangkat Embedded

1. Keterbatasan Daya Komputasi

- Perangkat embedded memiliki CPU/GPU yang jauh lebih lemah dibandingkan server atau cloud.
- Solusi: Gunakan model lightweight seperti TinyML atau quantized neural networks.

2. Kapasitas Penyimpanan Terbatas

- Model ML seringkali berukuran besar dan tidak dapat disimpan sepenuhnya dalam memori perangkat.
- Solusi: Gunakan model kompresi seperti TensorFlow Lite, pruning, dan quantization.

3. Efisiensi Energi

- Perangkat embedded sering berjalan dengan baterai, sehingga konsumsi daya harus dioptimalkan.
- Solusi: Optimasi inferensi dengan edge computing dan pemrosesan berbasis event-driven.

4. Kecepatan dan Latensi

- Model ML harus mampu memberikan hasil cepat dengan latensi minimal.
- Solusi: Gunakan algoritma yang efisien secara komputasi, seperti decision tree atau model shallow neural network.