ML Algorithms for WSN

Sahand Khoshdel, 810196607

A Comprehensive Survey on ML algorithms used in Wireless Sensor Networks

This presentation is a summary of the main topics discussed in the paper:

“Machine Learning Algorithms for Wireless Sensor Networks: A survey” by

D. Praveen Kumar, T.Amgoth, C.Annavarapu

Published on Elsevier, 2019

Table of Contents

01. Introduction To WSN

Wireless Sensor Networks

Self-configured, Infrastructure-less wireless networks of sensors that monitor physical or environmental conditions and co-operatively pass their data through the network to a main location (sink) where the data can be observed and analyzed

What's actually done?

Sensing & Monitoring

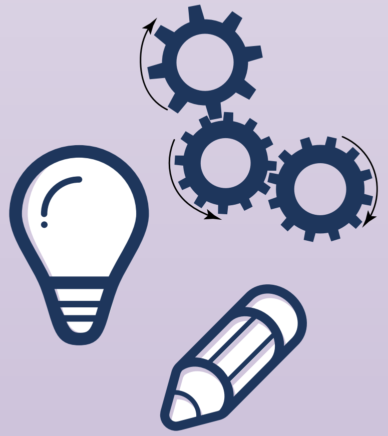
01

Sensors collect the signals from a source and convert them into electric signals and eventually a stream of bits

02

Routing & Data Transmission

Nodes act as routers, find the best route to deliver data to the sink node



Post Processing & Data Analysis

03

After monitored data is delivered to the sink node (base station), it is ready for further processing in order to extract useful information and to issue control signals

IoT Technology

In IoT, all the sensors directly send their information on the Internet.

Data is uploaded and downloaded from the network which each device is connected to

WSN Technology

WSN sensors have no direct connection to the internet. They act in a router-wise manner.

WSN's are more specifically defined than IoT networks. As almost all their components are categorized as sensors

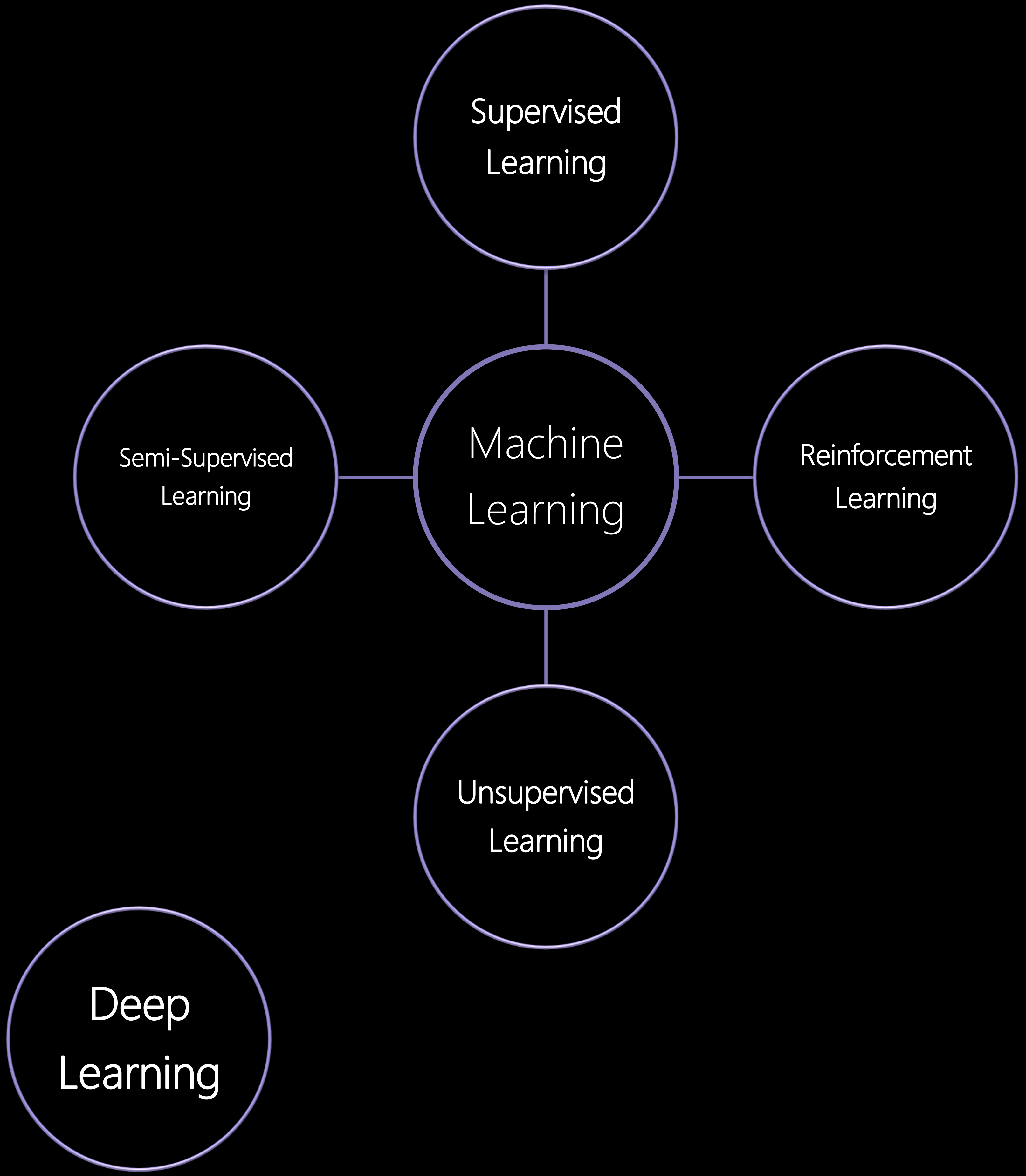
WSN's are often used within IoT systems, where a large collection of sensors can act as an IoT device

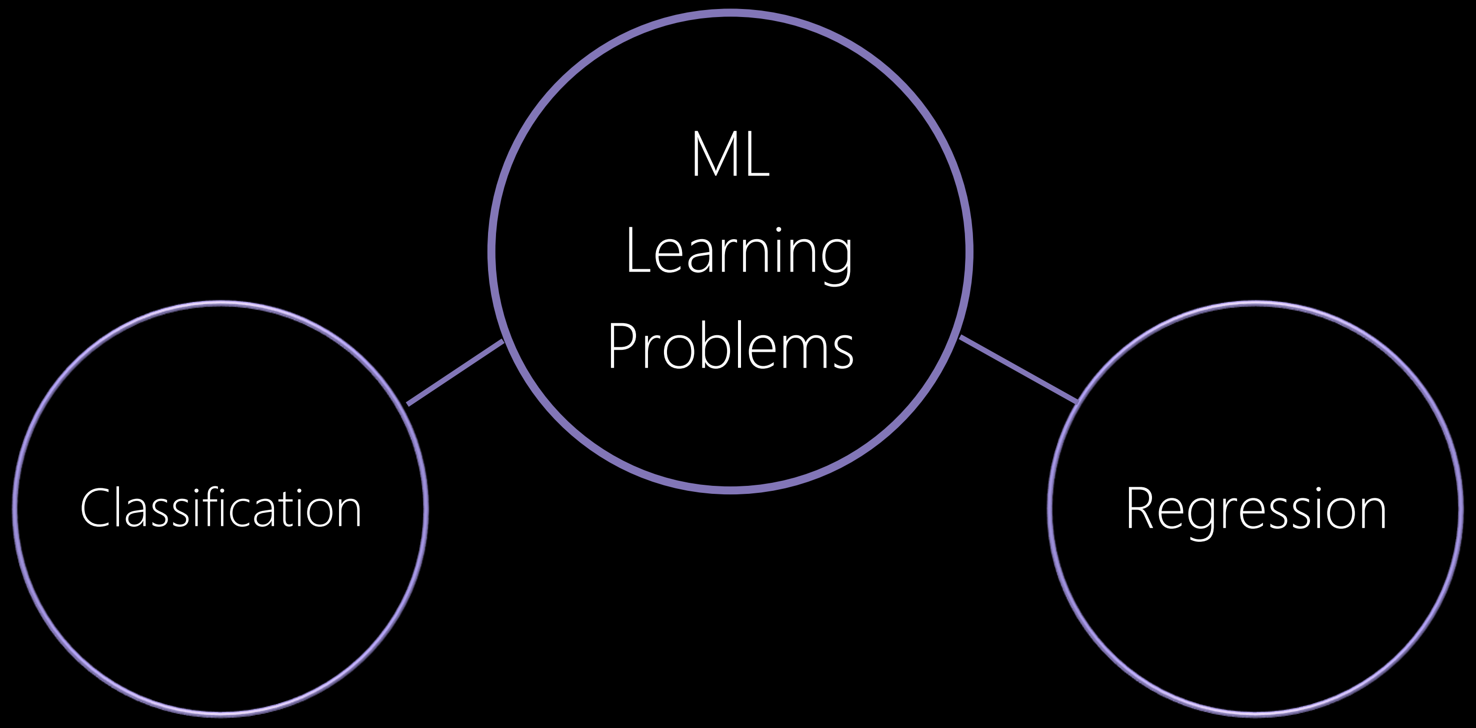
02. Overview on ML algorithms

The Art of Machine Learning

The process of self-improvement using data in computing algorithms in order to extract useful, hidden and complex information, predict unknown outcomes and learn how to react according to a certain data distribution

An overview on Machine Learning Approaches





Supervised Learning

Learning is done by observing training data labels, extracting high-level info about the data and trying to use it to predict labels for unseen test data.

Evaluation is done by calculating a cost function which the learning process aims to minimize.

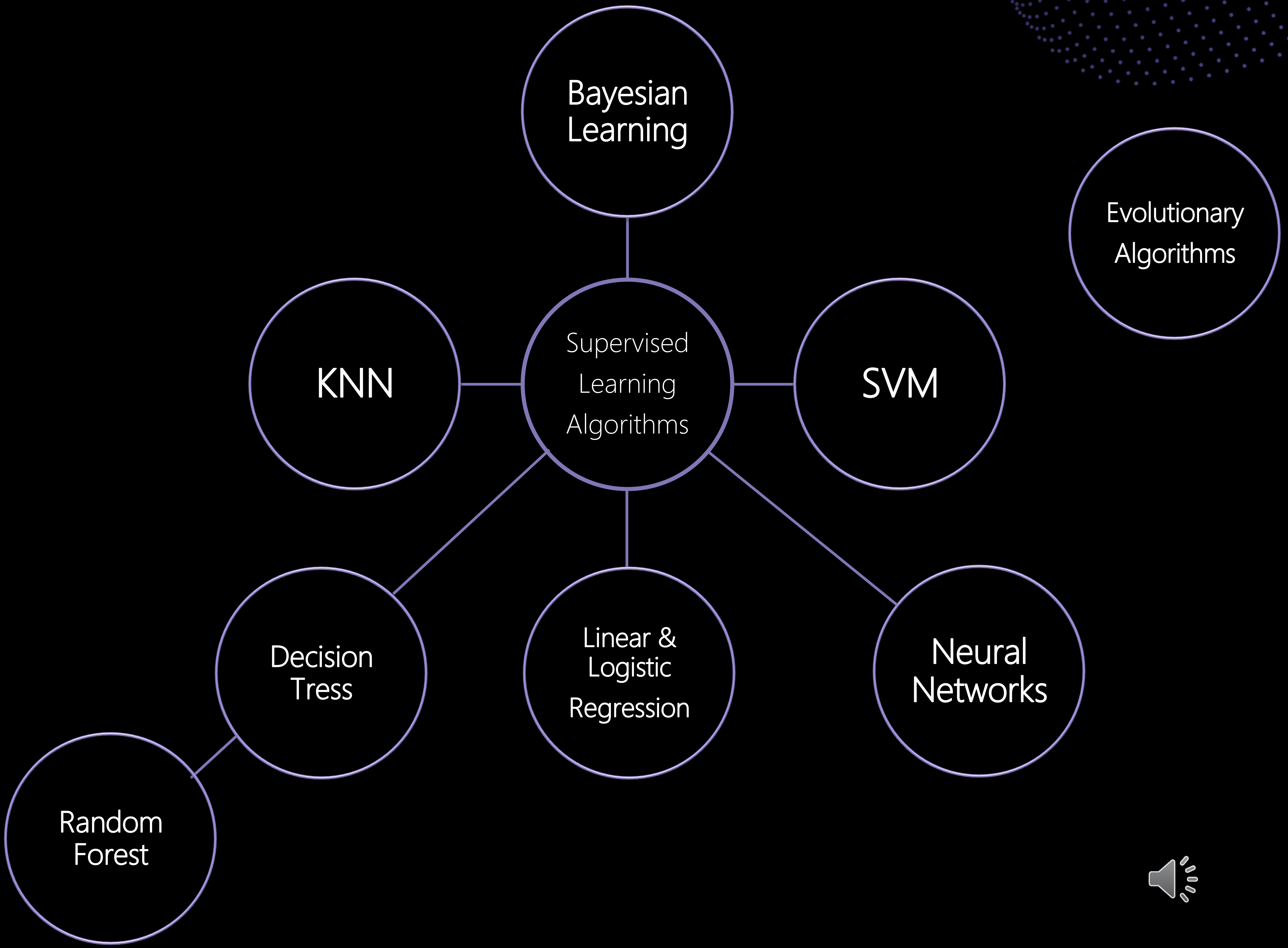
$$MSE = \frac{1}{n} \Sigma \underbrace{\left(y - \widehat{y} \right)^2}_{\text{The square of the difference between actual and predicted}}$$

Supervised Learning Algorithms

The major responsibilities of supervised learning algorithms is to generate a model that:

1. Represents relationships and dependency links between input features.

2. Forecast objective outputs.



Unsupervised Learning

The model tries to extract relationships according to similarities and differences among data, instead of using a labeled dataset

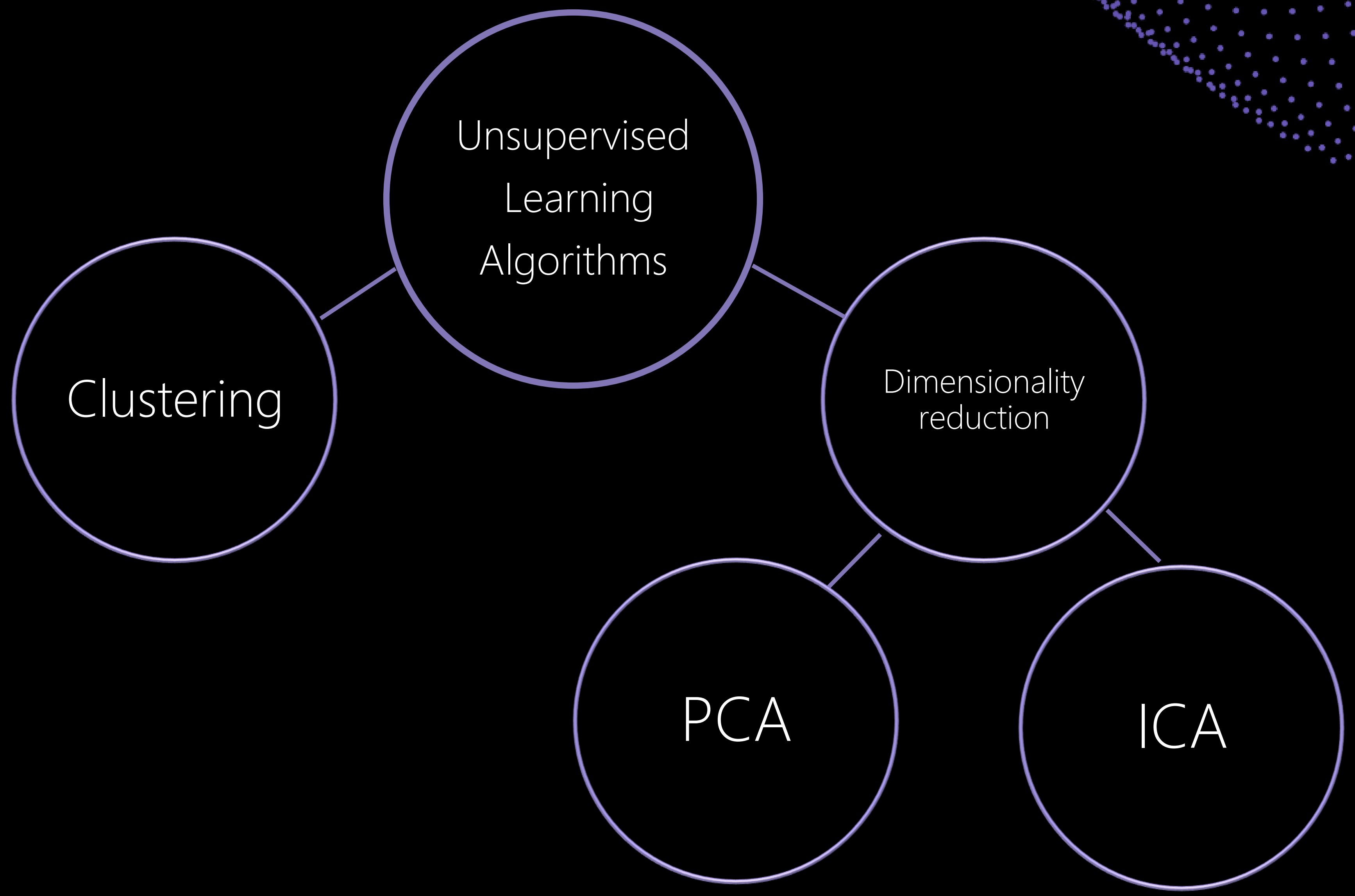
This may be necessary when no labels exist, when we don't have access to labels, when we cant rely on them, or when only similarities are meant to be found

Unsupervised Learning Algorithms

Clustering algorithms group data based on similarities and differences spotted among them.

Too much features make data analysis, interpretation and label prediction very complex.

Dimensionality reduction relies on expressing useful information in less dimensions, making it easier to process and interpret



Reinforcement Learning

Agent interacts with the environment by taking actions, receiving rewards and moving to the next state.

In each iteration, either state values (V) or action-state (Q) values are updated according to the observed reward, resulting in a policy update leading to agent’s better performance.

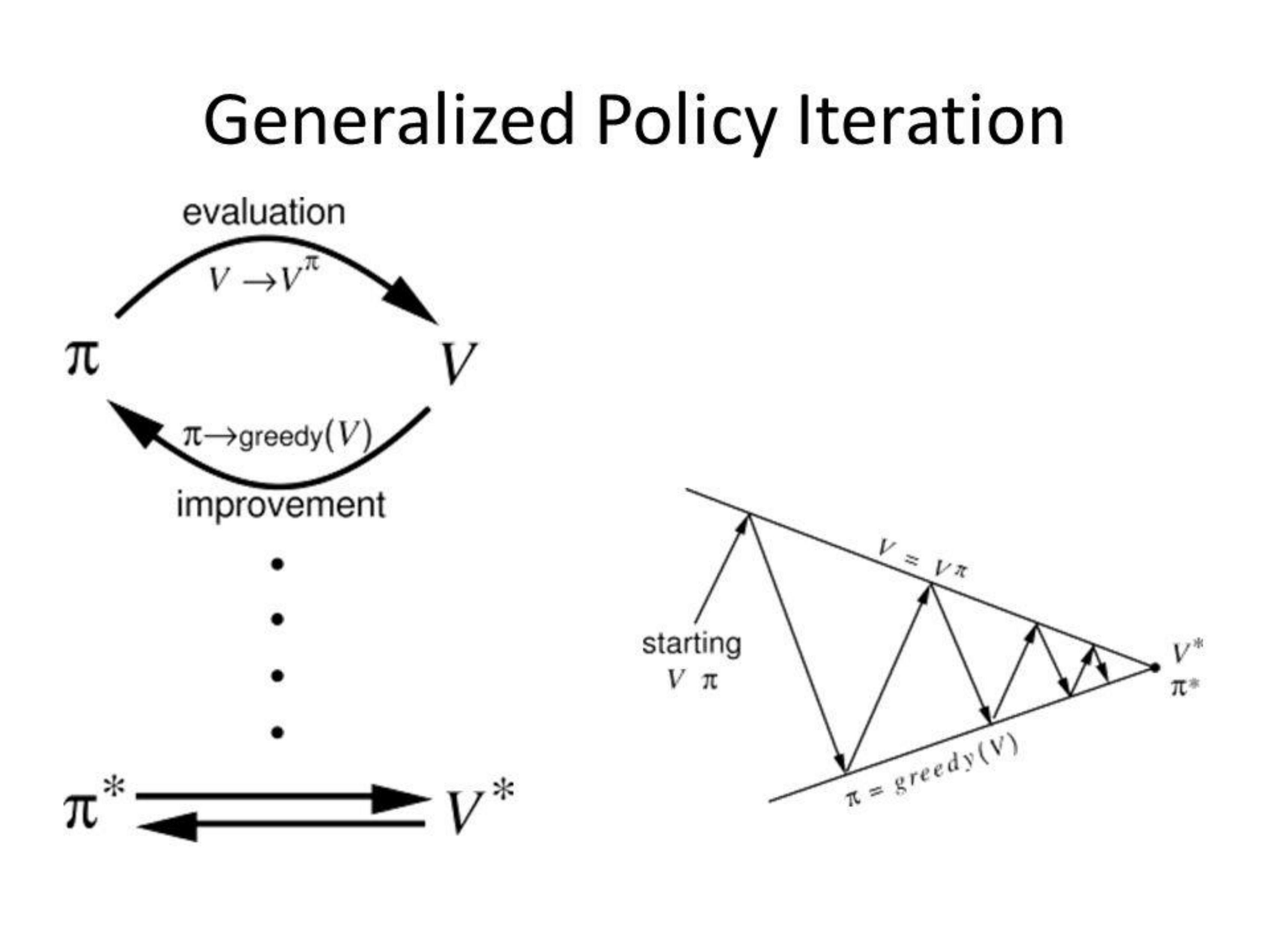
Fig. 5. Visualization of reinforcement learning.

Q- learning Algorithm:

$$Q(s_t, a_t) \leftarrow \underbrace{Q(s_t, a_t)}_{\text{old value}} + \underbrace{\alpha}_{\text{learning rate}} \cdot \left(\overbrace{\underbrace{r_{t+1}}_{\text{reward}} + \underbrace{\gamma}_{\text{discount factor}} \cdot \underbrace{\max_a Q(s_{t+1}, a)}_{\text{estimate of optimal future value}}}^{\text{learned value}} - \underbrace{Q(s_t, a_t)}_{\text{old value}} \right)$$

Reinforcement Learning

Many RL algorithms are based on value or policy iteration, the process of evaluation and improvement through a loop until values from a policy and optimal values converge to one another.



What are the main considerations in designing and maintaining a WSN?

How is ML solving these issues?

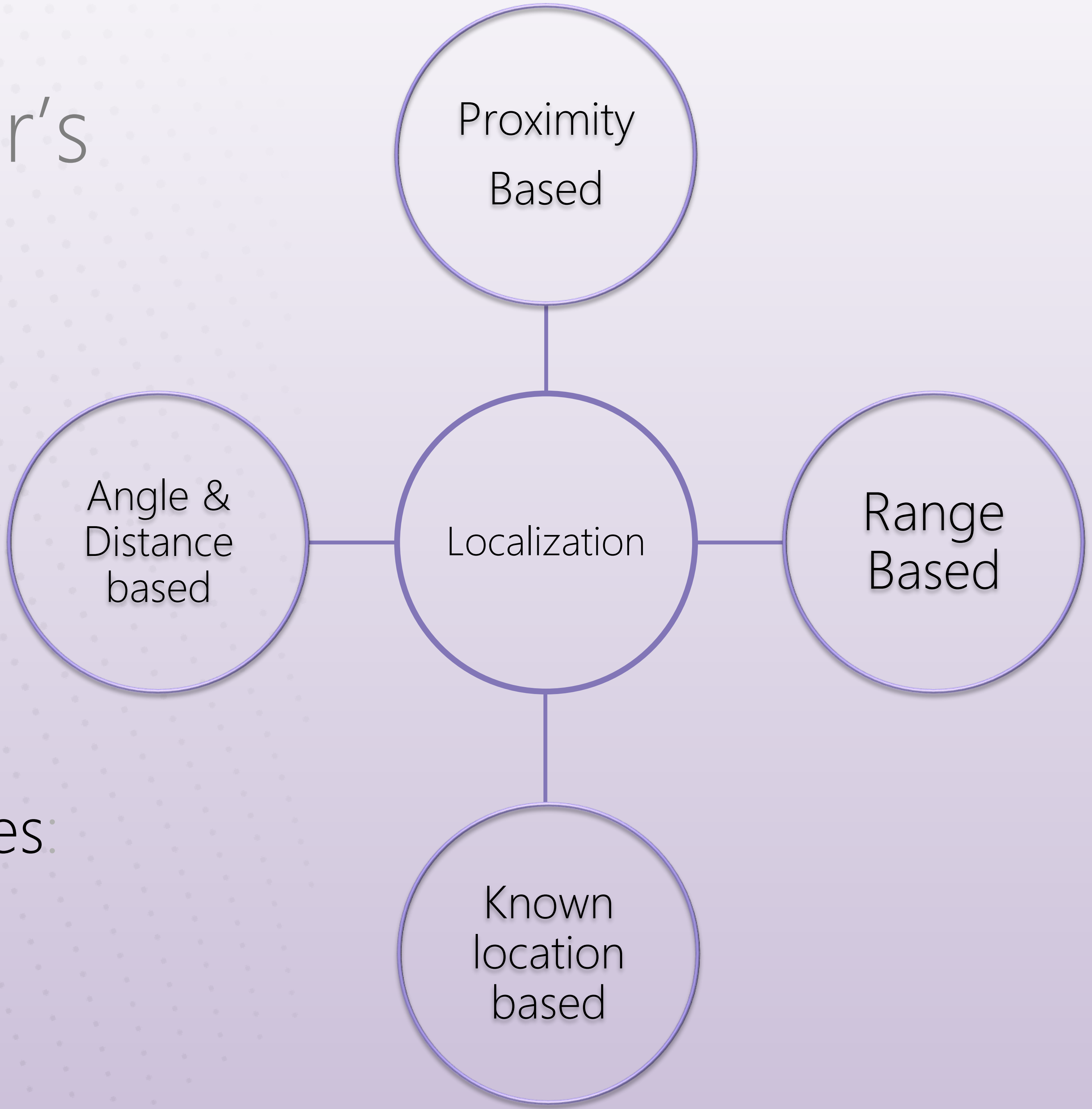
Appropriate algorithms?

03 & 04. WSN problems, ML solutions

1. Localization

The problem of finding each sensor's exact position.

ML becomes helpful in **reconfiguration** and increasing **location accuracy**



The main problem is when the **position of a sensor changes dynamically** due to external causes.

Solution to dynamic changes

Reconfiguration and Reprogramming

How does ML help Localization?

1. Classifying **anchor** nodes and **unknown** nodes in the network.
2. Creating clusters which can be trained separately to find coordinates rapidly.

3. Higher Accuracy is obtained as intelligent search in parameter space is done to optimize the system.

- **Clustering** nodes requires **RSSI** data for each node, which shows the received signal strength of each node.

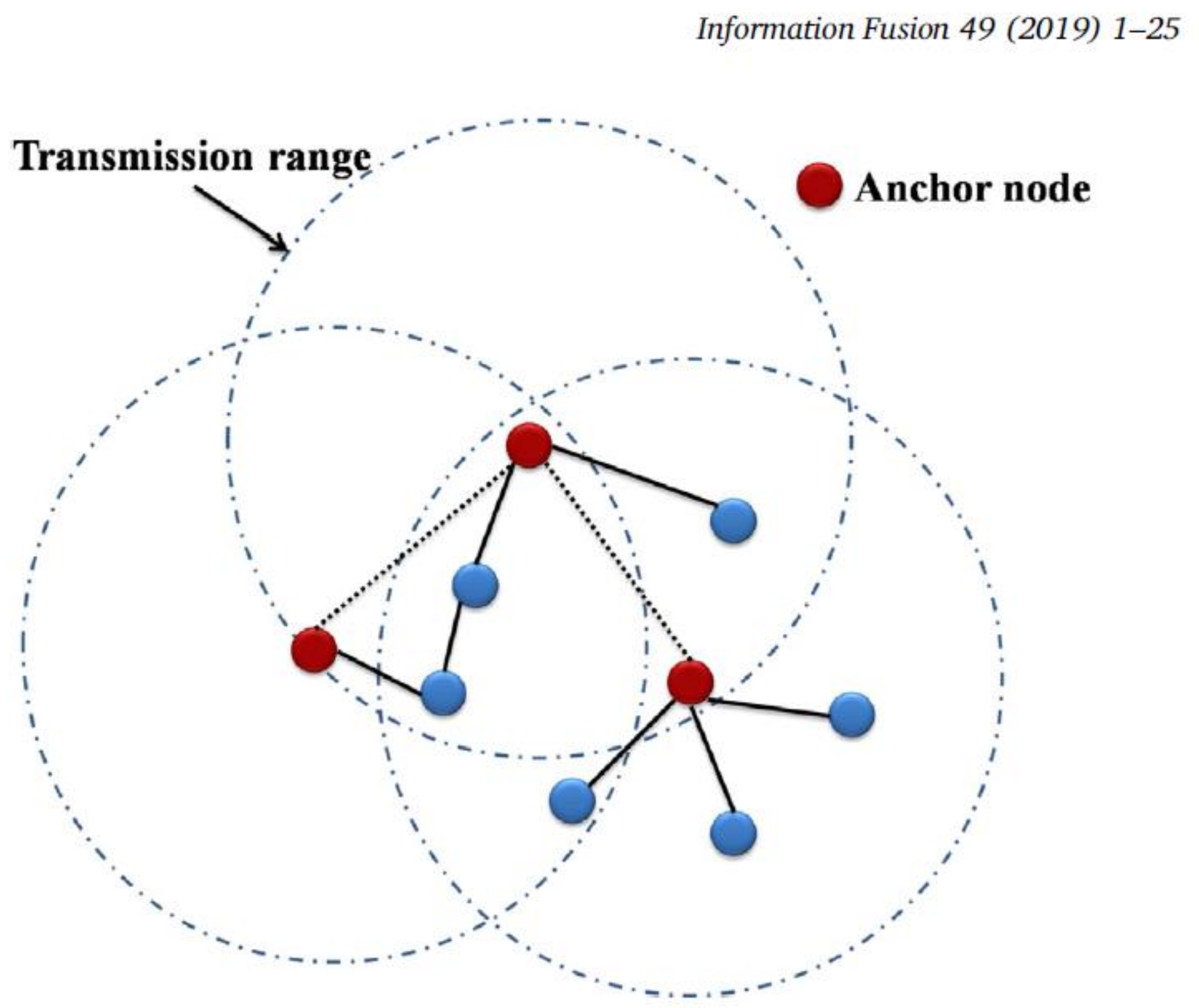


Fig. 7. Example of localization.

Anchor nodes are the nodes which already know their location via GPS, …

Other nodes locate themselves according anchor nodes

Lets see some examples

1. LPSONN:[10]

A new PSO-based neural network, where each anchor node calculates hop count with others and sends the info to the header. The head anchors are then trained using neural nets.(routing optimization)

A PSO(Particle swarm optimization) algorithm is used to find optimal number of hidden layers. [10]

Lets see some examples

2. Range-free localization via Energy efficient distance estimation (ANN based): [11]

- Robust to attenuation, interference, shadowing and fading
- High accuracy guaranteed

3. Improved PSO localization for mobile sensors (ANN based): [12]

- PSO used to optimize neuron numbers
- Higher accuracy than traditional methods

4. Vectored PSO + Fuzzy logic for non-uniform node deployment [13]

- Lower Complexity

Lets see some examples

5. Fuzzy Linguistic Model for indoor WSN's [14]

- Fuzzy logic used in two areas: 1- RSSI variation handling / 2- Fuzzy location indicator

6. Graph-based localization using CNN+SVM [15]

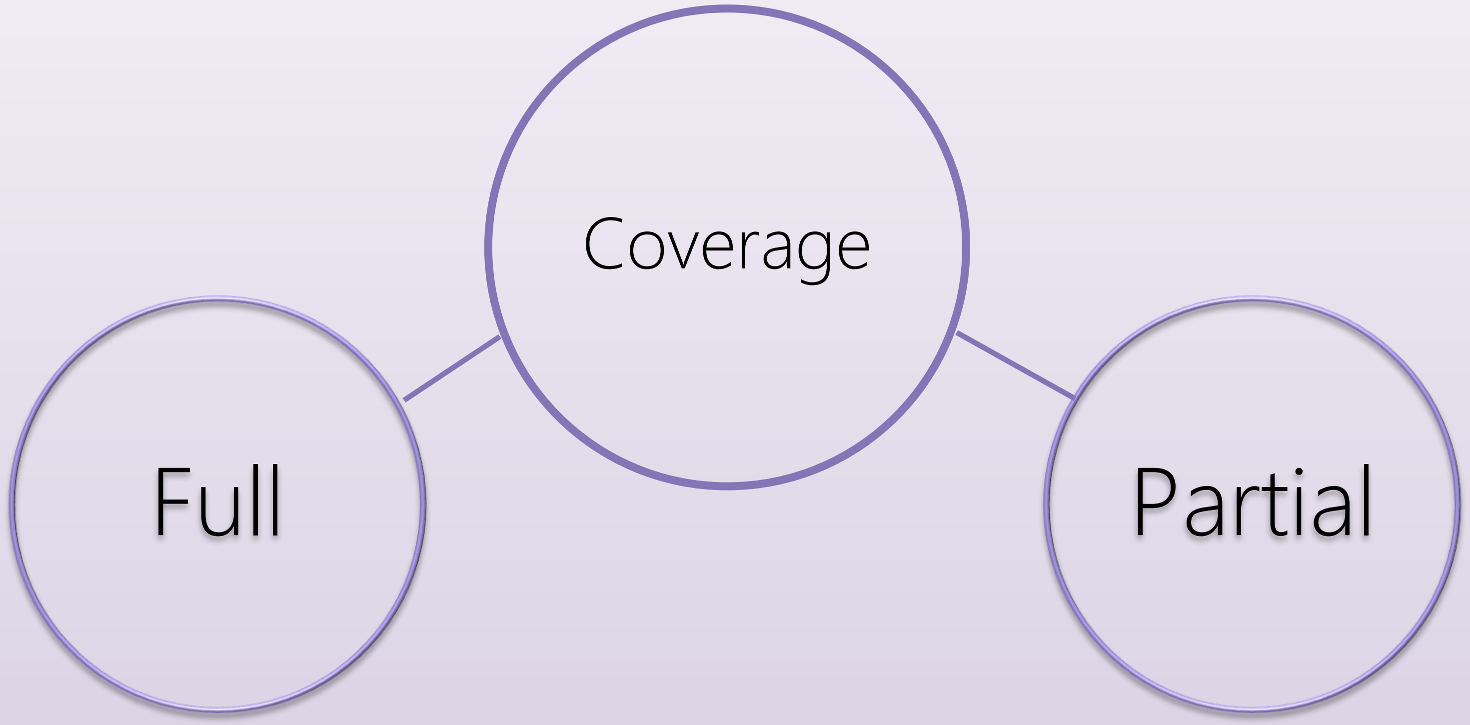
- Analyzing target signals with heterogeneous dual classifiers
- Identifies leakage location of the network

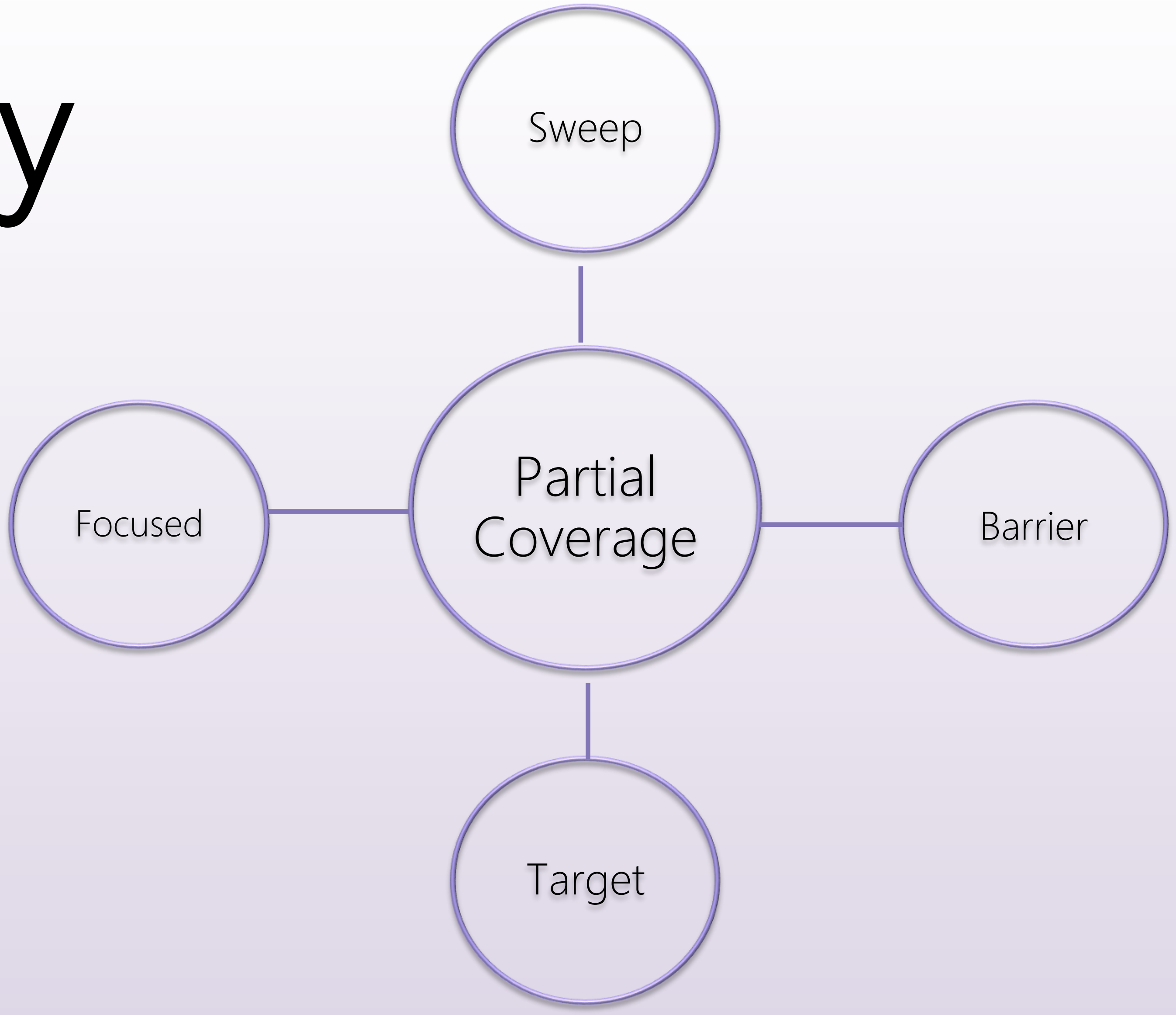
7. SVM for large scale WSN's [16]

- Solving large training overhead issue with sample reduction by fuzzy c-means

2. Coverage & Connectivity

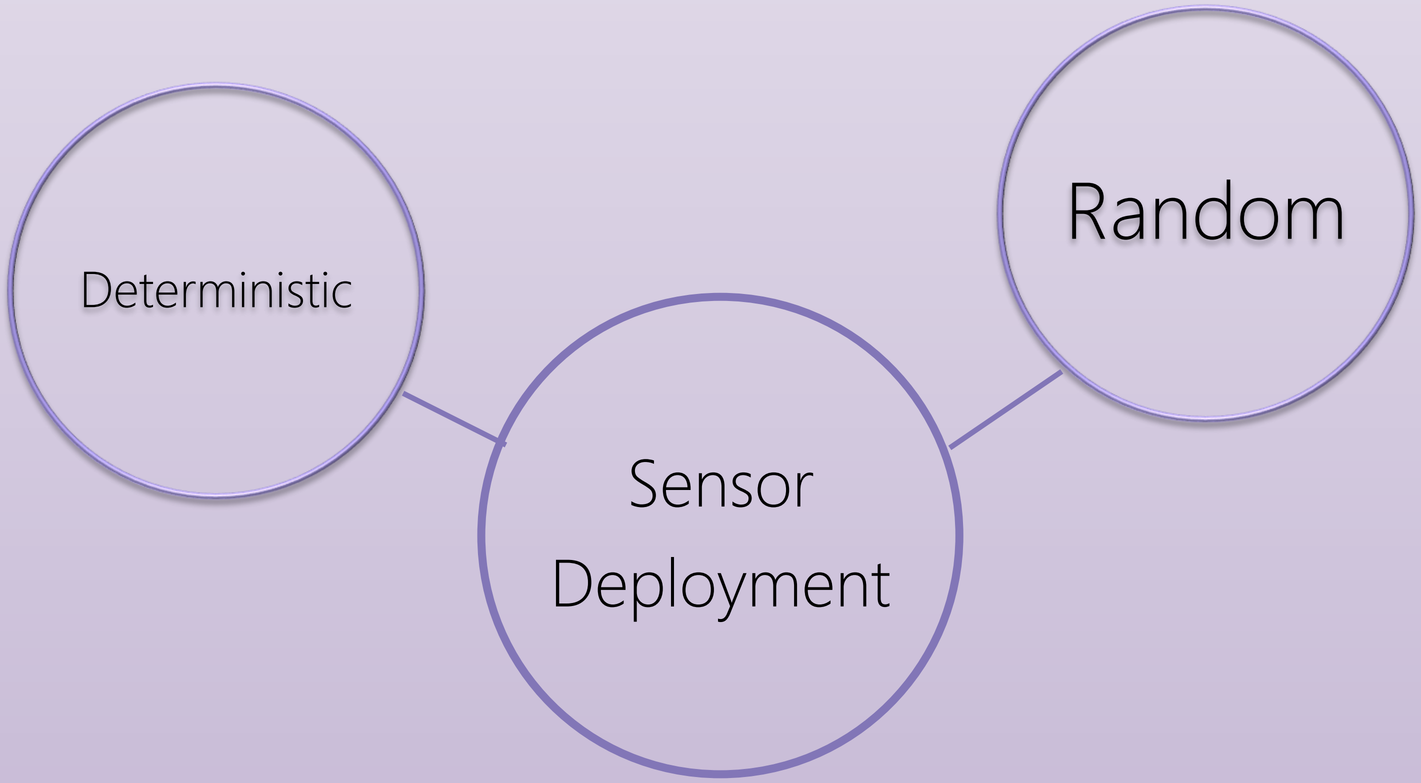
Coverage: Sensor monitoring efficiency





Connectivity: No isolated sensors!

Every sensor in the network should directly send their data to the sink node through relay nodes



Coverage holes (uncovered areas) are shown in white

Node A, B are disconnected as no nodes are in their range to sense their signal (Low RSSI)

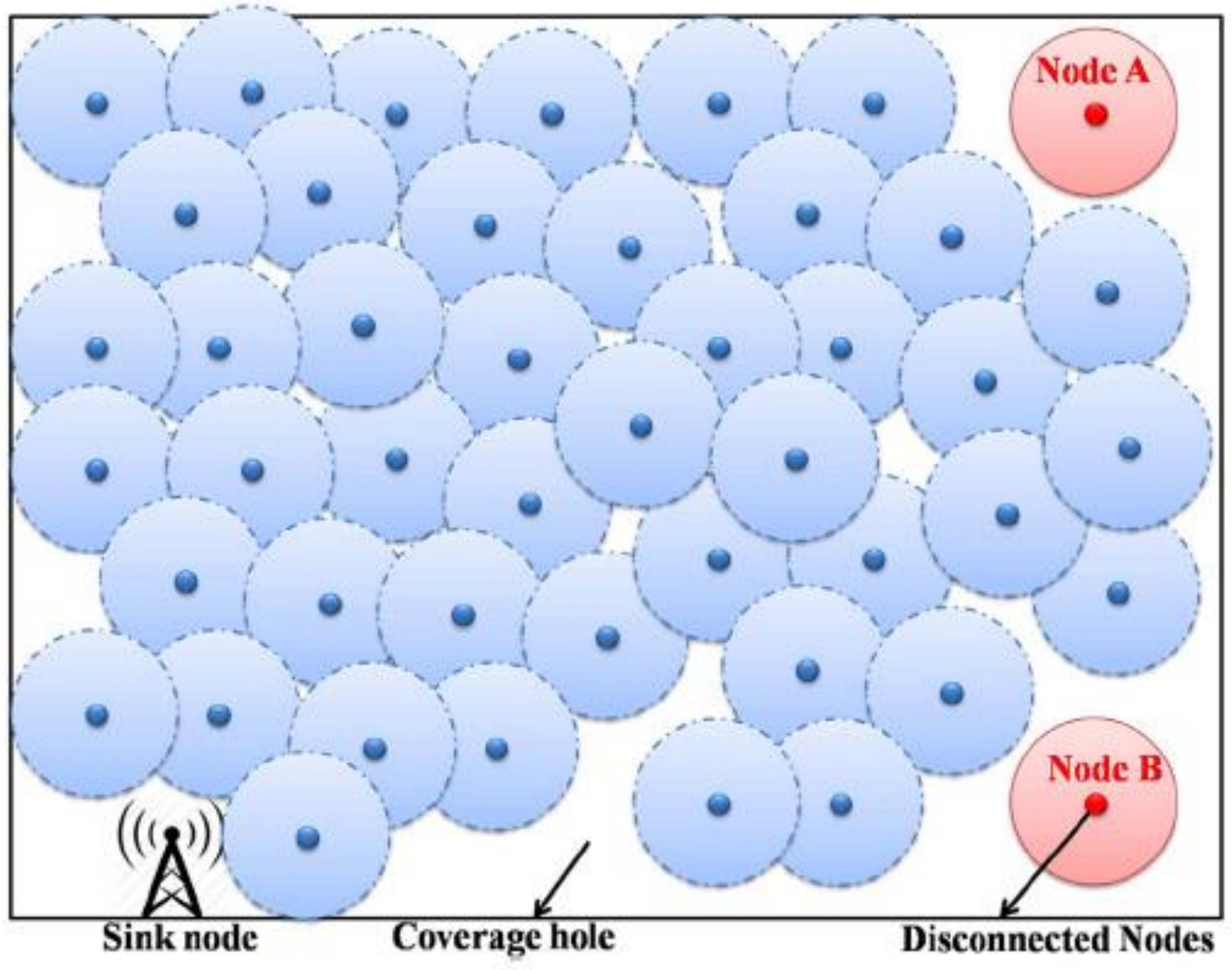


Fig. 8. Example of coverage and connectivity.

How does ML help Coverage & Connectivity?

1. Minimizing number of nodes for covering an area

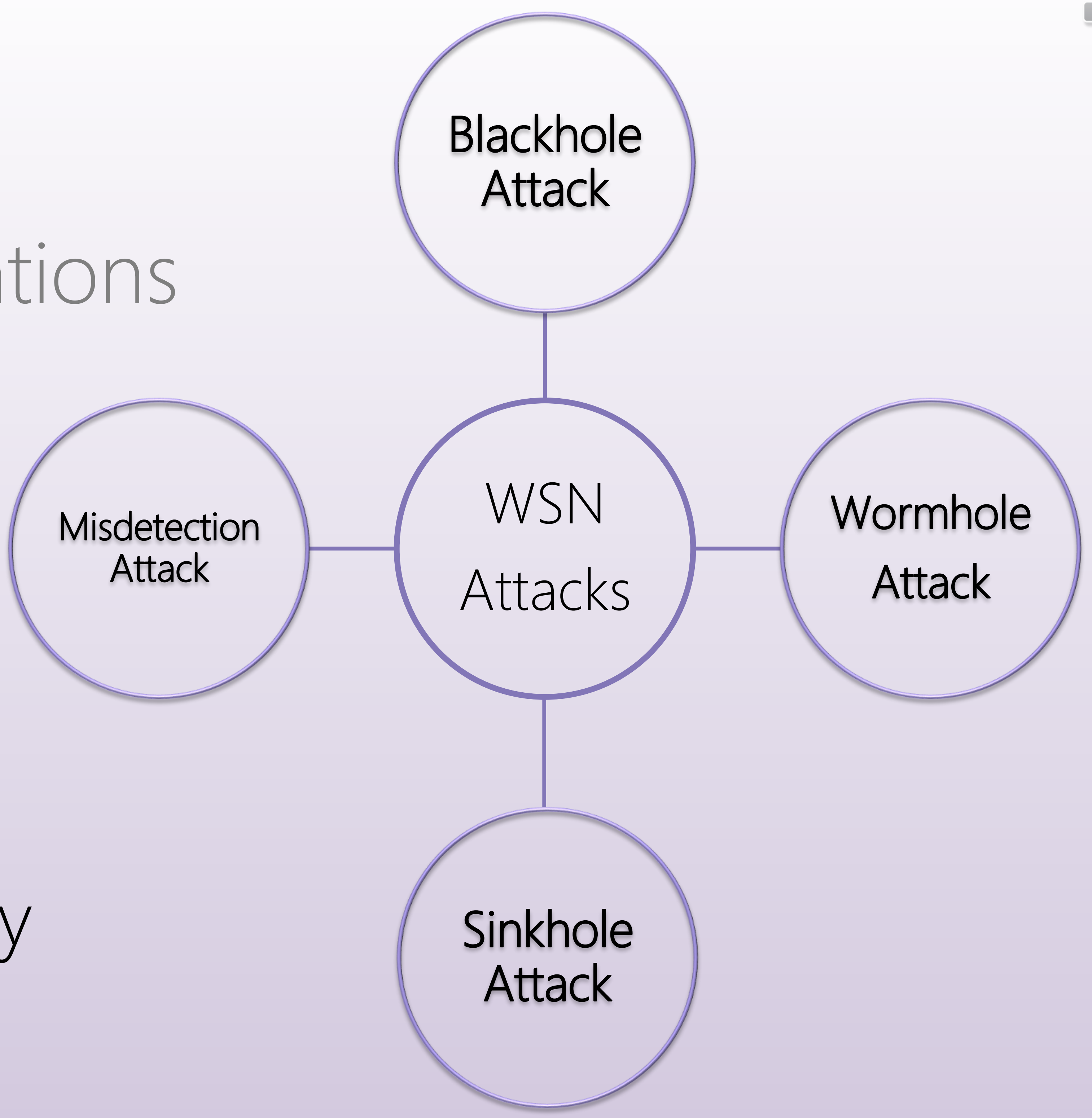
2. Obtaining desired locations for #1

3. Changing routes dynamically based on the connected nodes.

3. Anomaly Detection

Anomaly: Rare and significant observations

These misbehaviors mainly happen in two fields:
1. Measuring sensor data
2. Traffic-related attributes



While data transmission there is a possibility of data loss because of abnormal attacks!

We want fast networks in many applications!

Anomaly detection leads to overhead minimization! Higher pure data rate!

Blackhole Attack

In a blackhole attack a node receives the data instead of forwarding it towards the base station

Misdirection Attack

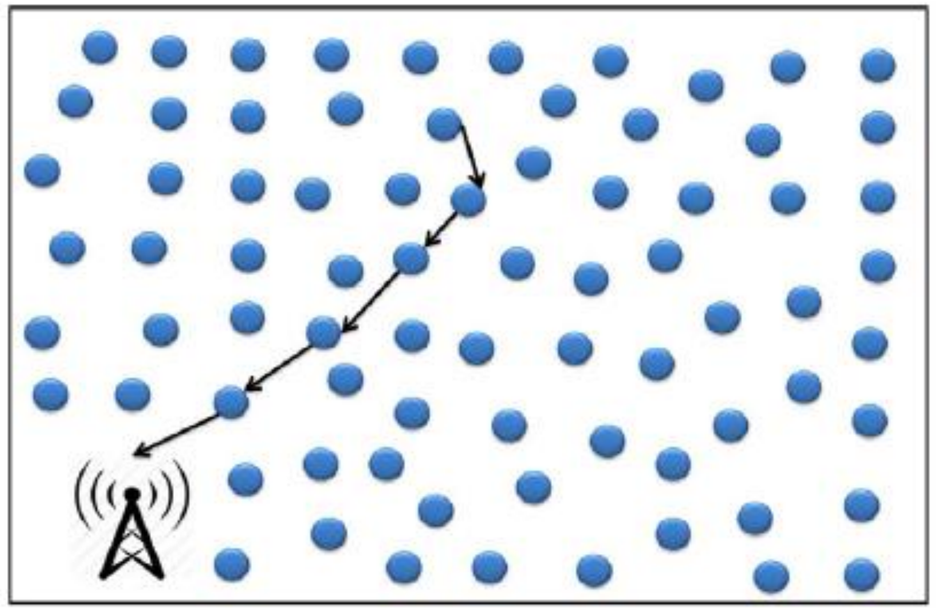
The attacker routes the packet through distant nodes rather than it's neighbors, resulting in a longer route and a throughput decrease

Wormhole attack

Some attacker nodes create a tunnel route to manipulate packets in their journey towards the base station

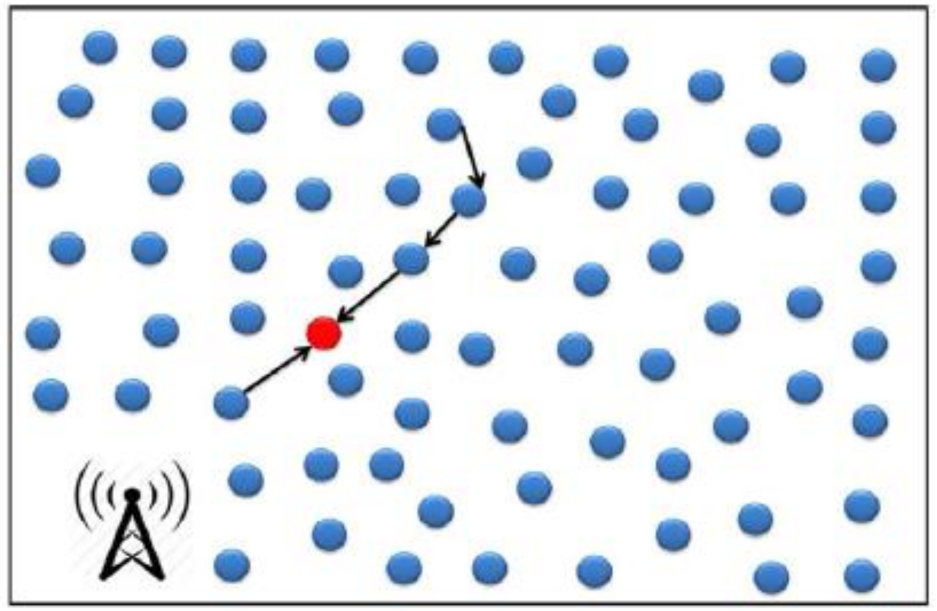
Sinkhole attack

The attacker node advertises an optimal route (via itself to the base station) to its neighbors, causing data loss due to congestion and expiration



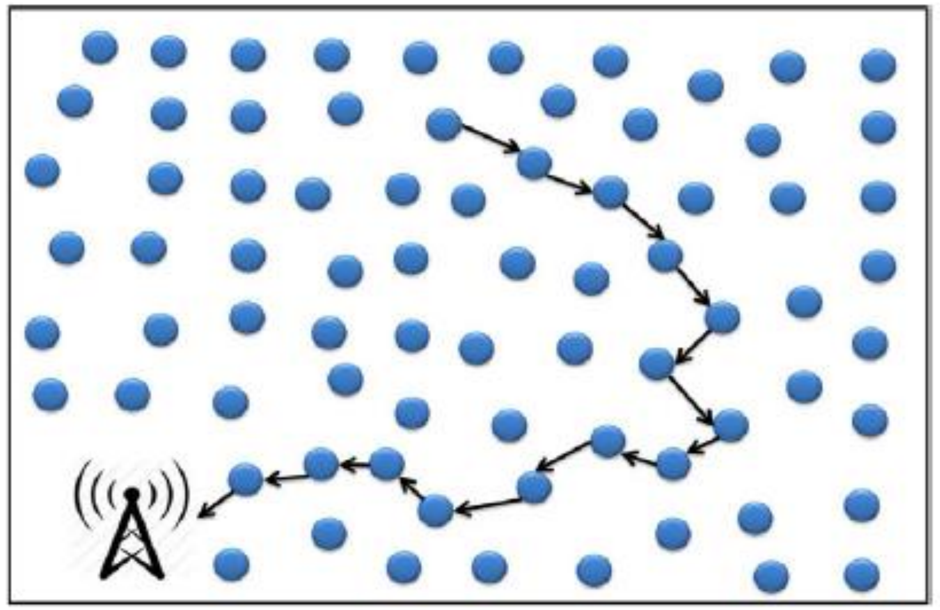
(a)

Normal Flow



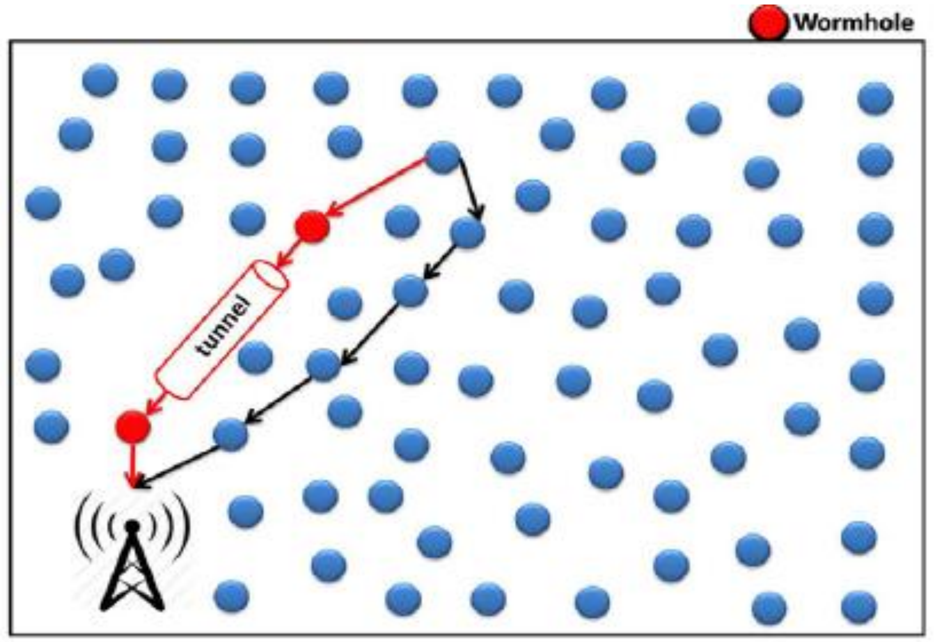
(b)

Blackhole Attack



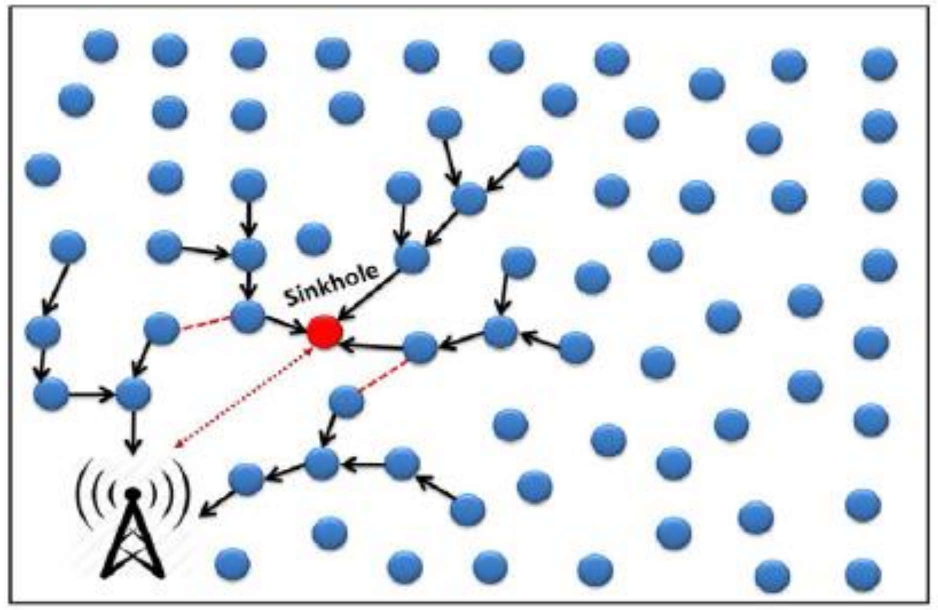
(c)

Misdirection Attack



(d)

Wormhole Attack



(e)

Sinkhole Attack

How does ML help Anomaly detection?

1. Self-learning threshold algorithm

2. Sleep-scheduling approaches

3. Detecting data inconsistencies, to prevent sinkhole attacks (such as DoS)

4- Suitable for non-stationary environment.

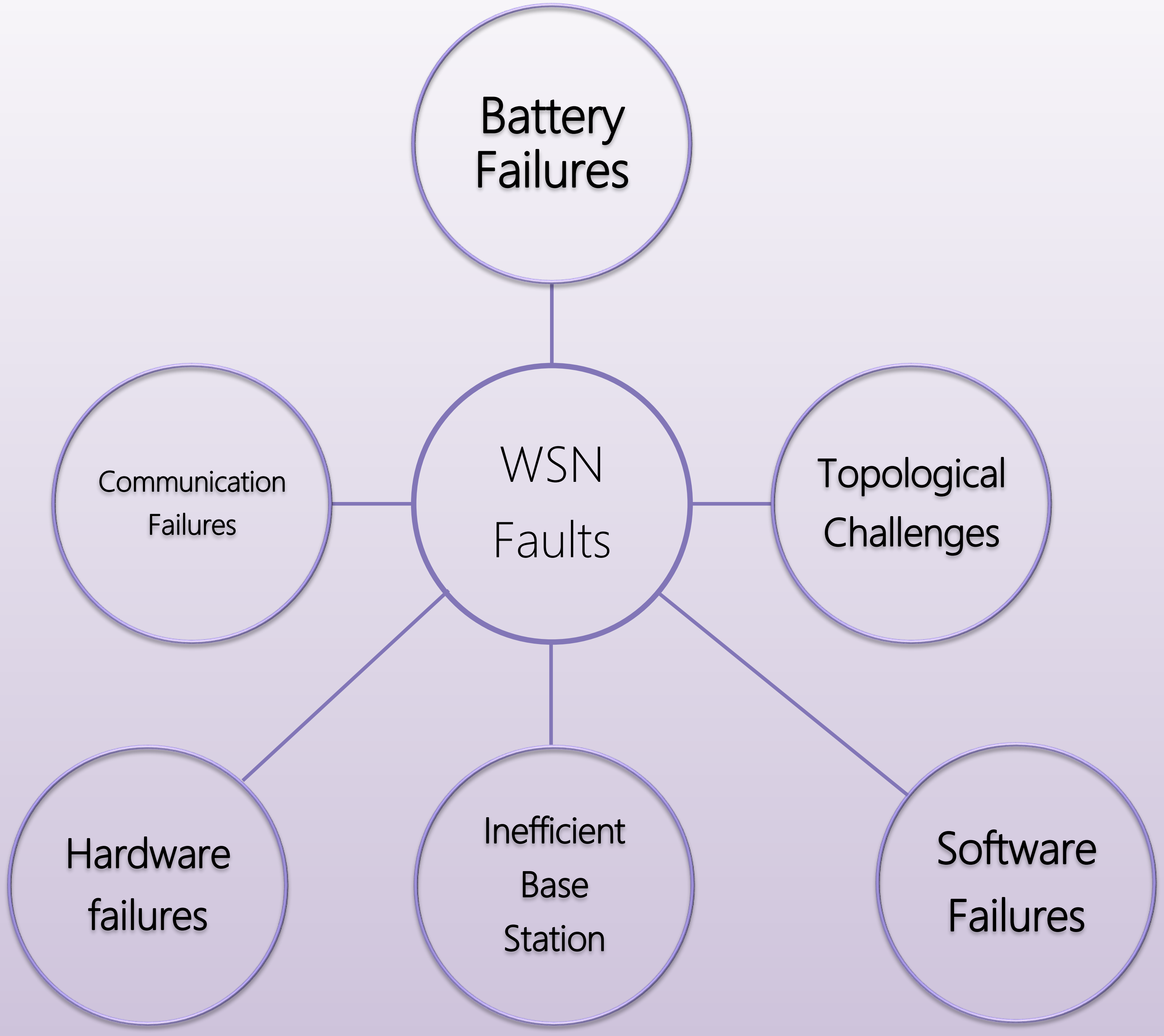
5- Using historical information in dynamic online anomaly detection

4. Fault detection

WSN's are often implemented in harsh and hostile conditions …

Challenges in fault detection:

1. Resource limitation
2. Environment variability
3. Deployment changes
4. Detection accuracy



How does ML help fault detection?

1. Fault categorization

2. Improving accuracy relative to traditional methods

3. Brisk detection

5. Routing

Routing is one of the most important and challenging issues in WSNs due to many limitations:

1. Power limitation
2. Transmission BW
3. Memory capacity
4. Processing capacity

Goals:

1. Reduce Energy Consumption
2. Increase Network Lifetime

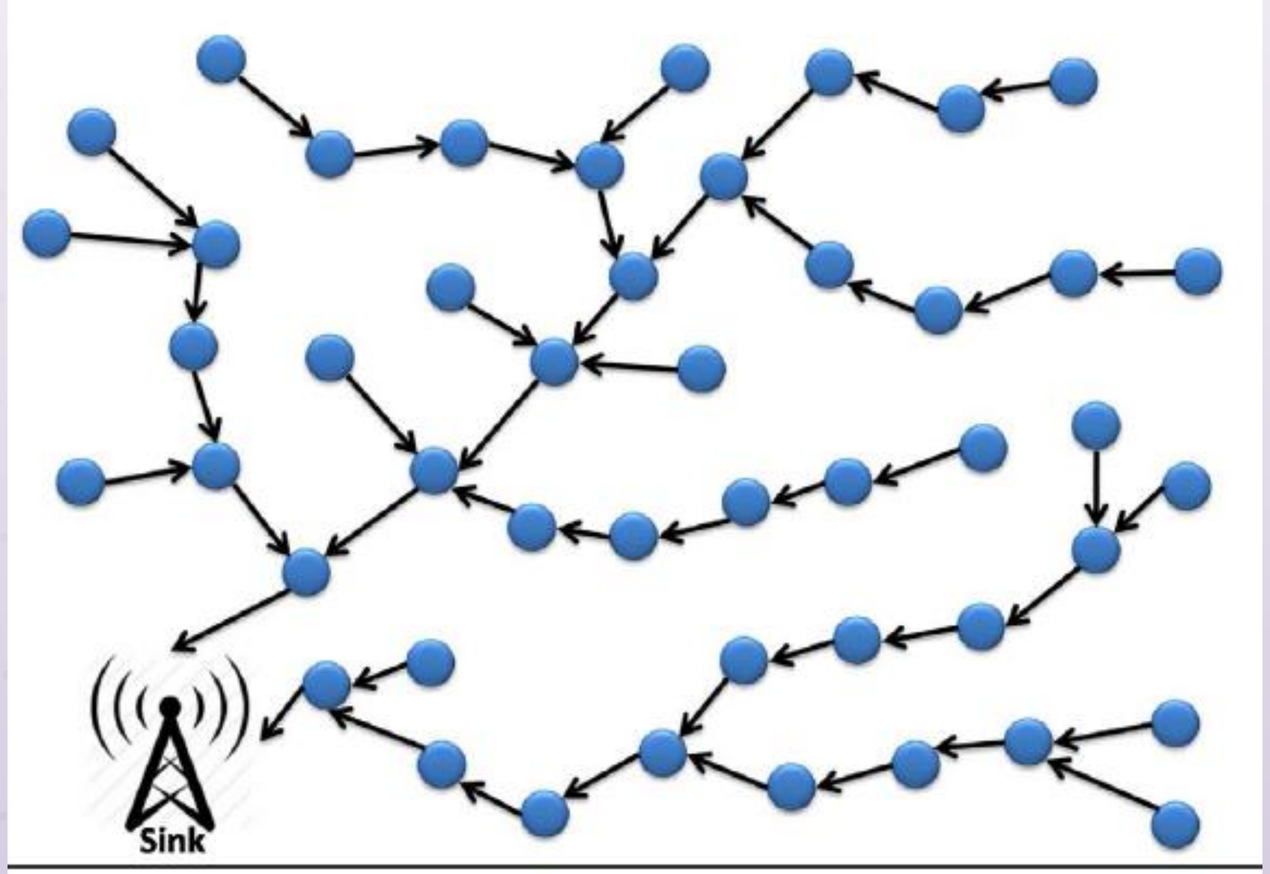


Fig. 10. Example of routing in WSNs.

Generally nodes **near** to the **base station consume more power** because they act as a relay to other nodes too.

How does ML help routing?

1. Multivariate optimization with smart search to find the best path according to several factors.
2. Adopts to dynamic environment without reprogramming
3. Decreasing overhead, increasing throughput

Smart Routing

- Sending messages to neighbor nodes on a random fashion causes energy consumption, data rate and congestion problems.
- Cluster nodes into efficient neighborhoods and select head nodes for each cluster
- Forward data to your head node which is responsible of sending it to the base station

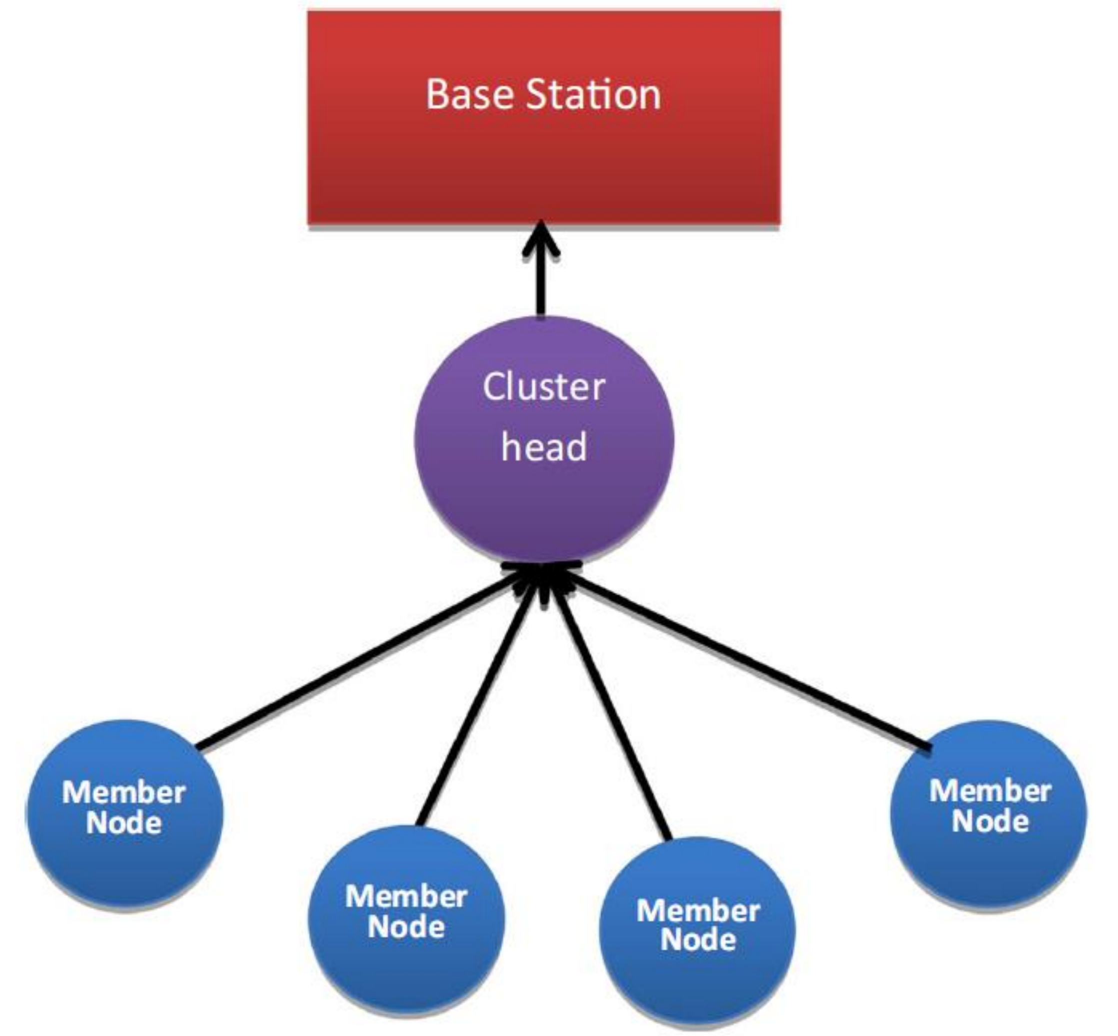


Fig. 1 A model of a clustered network

Clustering Nodes

- To implement smart routing algorithms are usually separately developed to cluster and then choose head nodes for clusters

Training data should be gathered and stored in a structured format to be ready for post processing

Table 1 Sample of the matrix containing data (a part of matrix)

	Node1	Node2	Node3		Node100
X-coordinate	95.7	37.8	14.1	…	46.27
Y-coordinate	48.5	81.1	42.1	…	50.64
The remaining energy level	1.60	1.06	1.83	…	1.95
Distance from base station	45.74	33.43	36.65	…	3.78
Local distance	5817.10	4630.13	4704.98	…	3703.19
Class	−1	−1	−1	…	1

6. Congestion Control

Congestion happens when a node handles more data than it’s capacity

It can happen for several reasons

1. Node buffer overflow
2. Many-to-One data transmission scheme
3. Packet Collision
4. Dynamic Time Variations

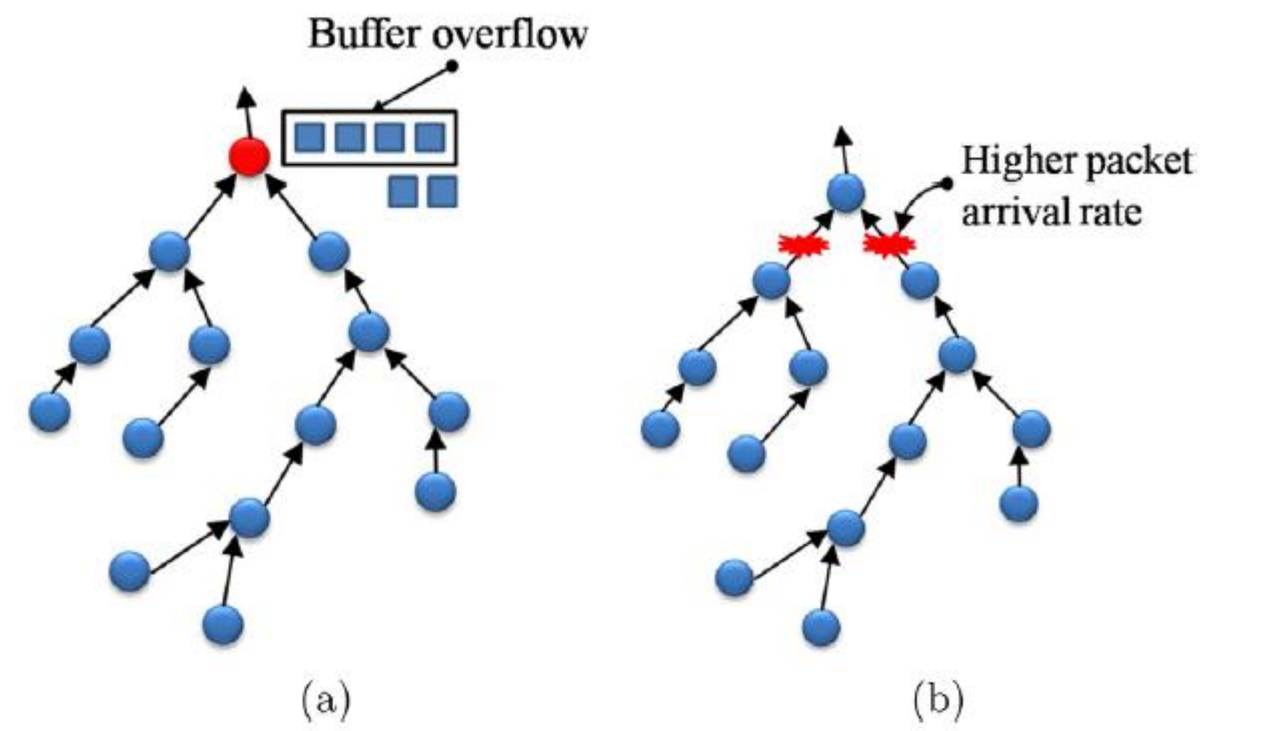


Fig. 12. Congestion in WSNs (a) Node level congestion (b) Link level congestion [213,214].

How does ML help congestion control?

1. Accurate traffic estimation, resulting in better routing and less congestion

2. Dynamic optimization of transmission ranges

Thanks for your attention!