SUSTAINMENT MANAGEMENT SYSTEM (SMS) SUMMIT 2024

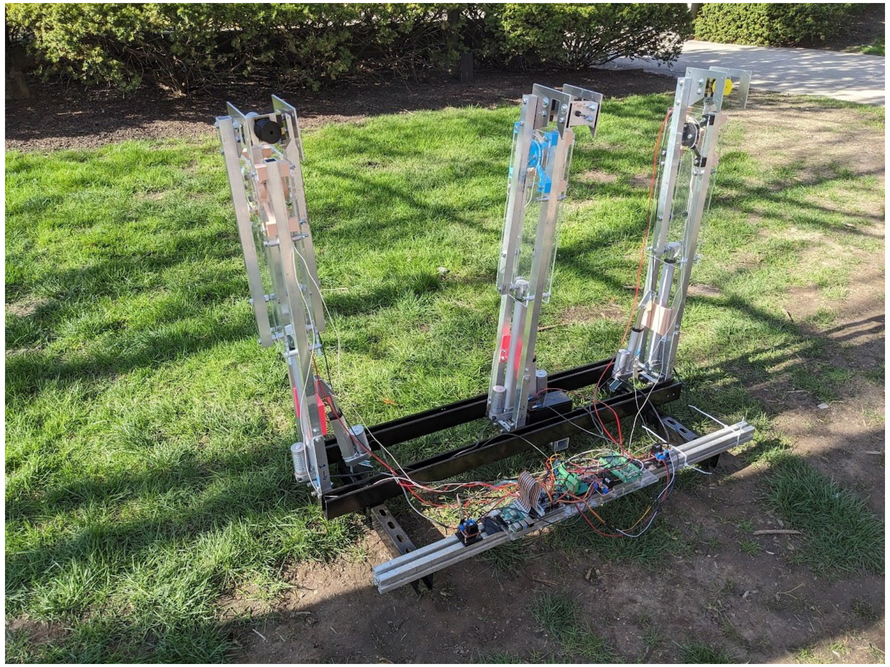
Power Line Inspections: Moving Forwards, and Improvement within the Department of Defense

U.S. ARMY

US Army Corps of Engineers®

ERDC
ENGINEER RESEARCH & DEVELOPMENT CENTER

WHY POWER LINE INSPECTIONS?



https://engineering.purdue.edu/fsae/wordpress/purdue-college-of-engineering/;

WHY ARE POWER LINE INSPECTIONS IMPORTANT?

ERDC

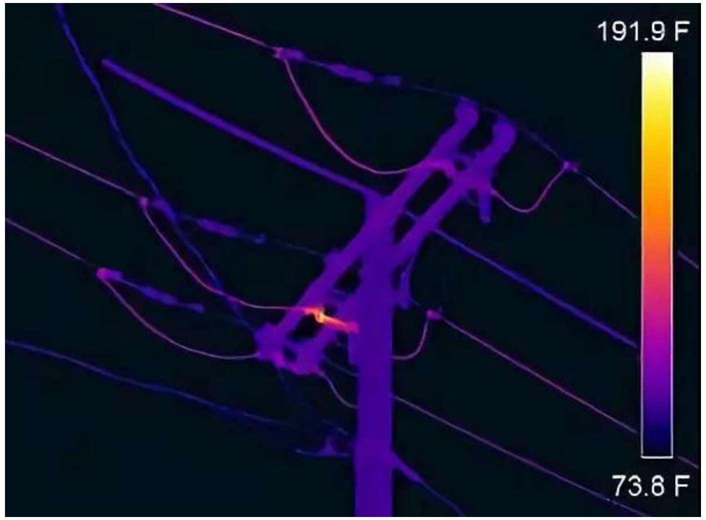
- Increasing demand
- Aging grid
- Disasters are devastating
- Lots of money involved

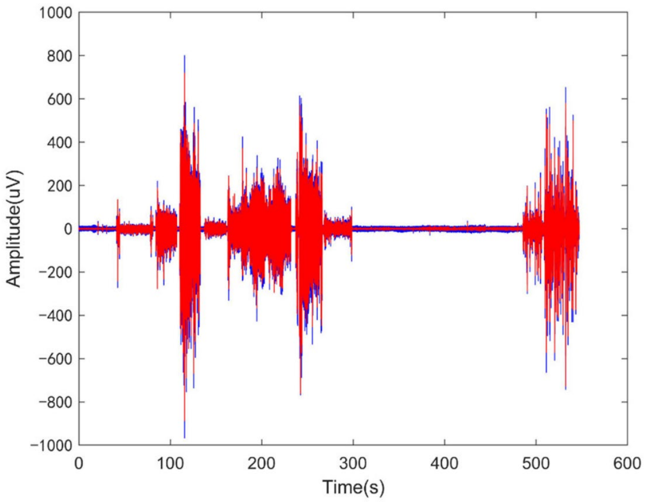
https://abcnews.go.com/US/people-missing-devastating-colorado-wildfire/story?id=82034519; https://www.cummins.com/news/2019/08/28/10-us-states-longest-power-outages

WHAT INFO DO POWER INSPECTIONS GATHER?

ERDC

- Trees or foreign objects
- Temperature
- Excess slack
- Damaged components
- Frayed cables
- Internal cracks
- Corrosion
- Radio interference

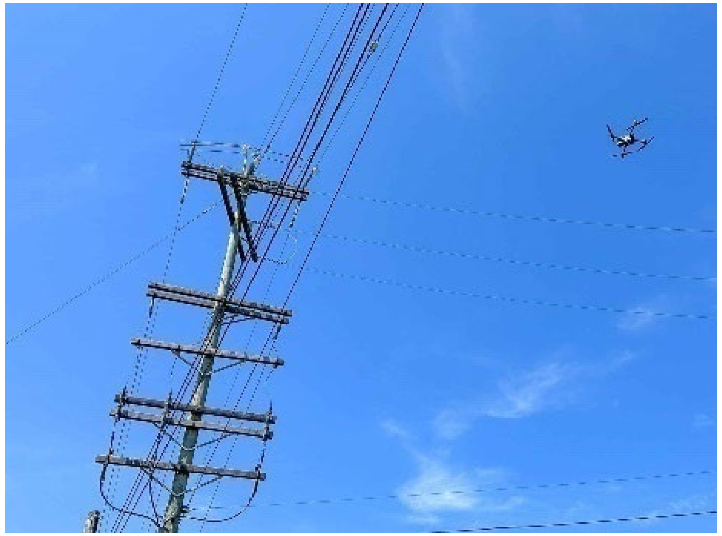




https://hort.ifas.ufl.edu/treesandpowerlines/goodandpoor.shtml; https://www.uavfordrone.com/powerline-inspection-by-drone-uav-thermal-imaging-camera/; https://www.oklahoman.com/story/weather/2014/09/18/update-winds-tipped-power-poles-in-north-oklahoma-city-as-storms-roll-through-thursday-morning/60798459007/; https://www.mdpi.com/1424-8220/23/11/5182; https://nj.pseg.com/safetyandreliability/safetytips/balloonkitesafety;

WHAT CURRENT POWER INSPECTIONS LOOK LIKE

ERDC



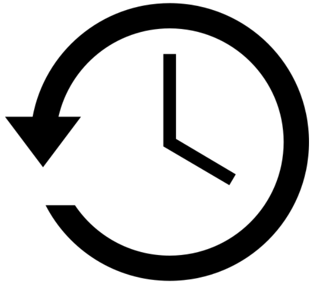
https://www.tdworld.com/electric-utility-operations/article/20966170/take-a-look-inside-a-linemans-bucket-truck; https://www.linkedin.com/pulse/power-lines-inspection-methods-aispeco; https://www.meridianhelicopters.com.au/portfolio-item/powerline-inspections/; https://thedronelifenj.com/power-line-inspections-guide/;

WHERE CURRENT POWER INSPECTIONS FAIL

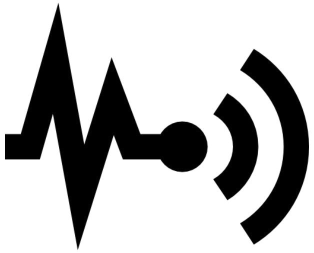
1. Slow
2. Tedious
3. Expensive
4. Can miss defects
5. Can be disruptive
6. Can’t go everywhere

ROBOTS

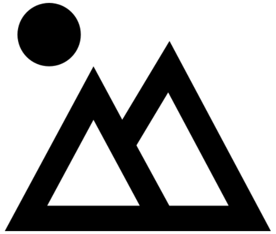
Less expensive



Continuous Inspection



Can Use Multiple Sensors



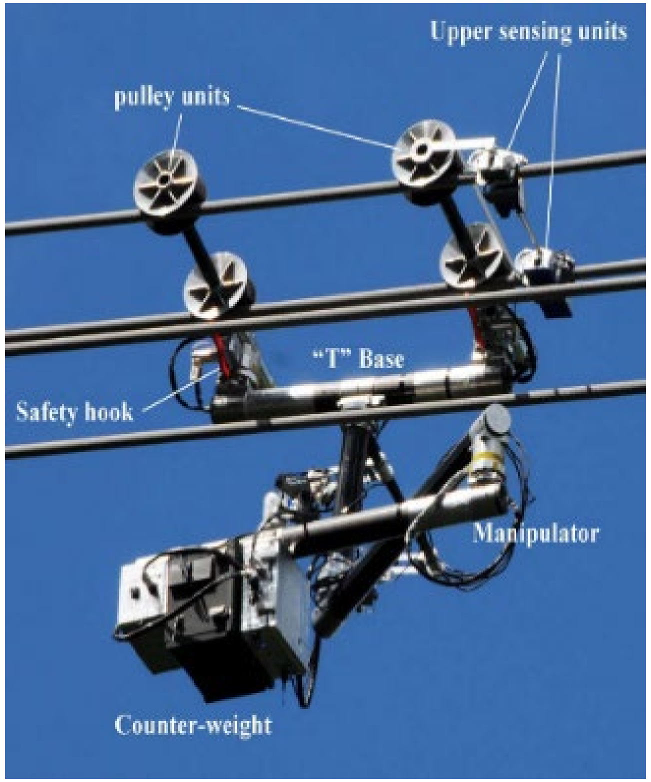
Can access hard-to-reach areas

https://icon-icons.com/;

CRAWLING ROBOTS

- Longer battery life
- Higher quality data

- Slower
- Not as mobile
- High voltages



https://ieeexplore.ieee.org/document/5624434; https://ieeexplore.ieee.org/document/7030054; https://ieeexplore.ieee.org/document/8794397;

FLYING ROBOTS

- Fast
- Mobile

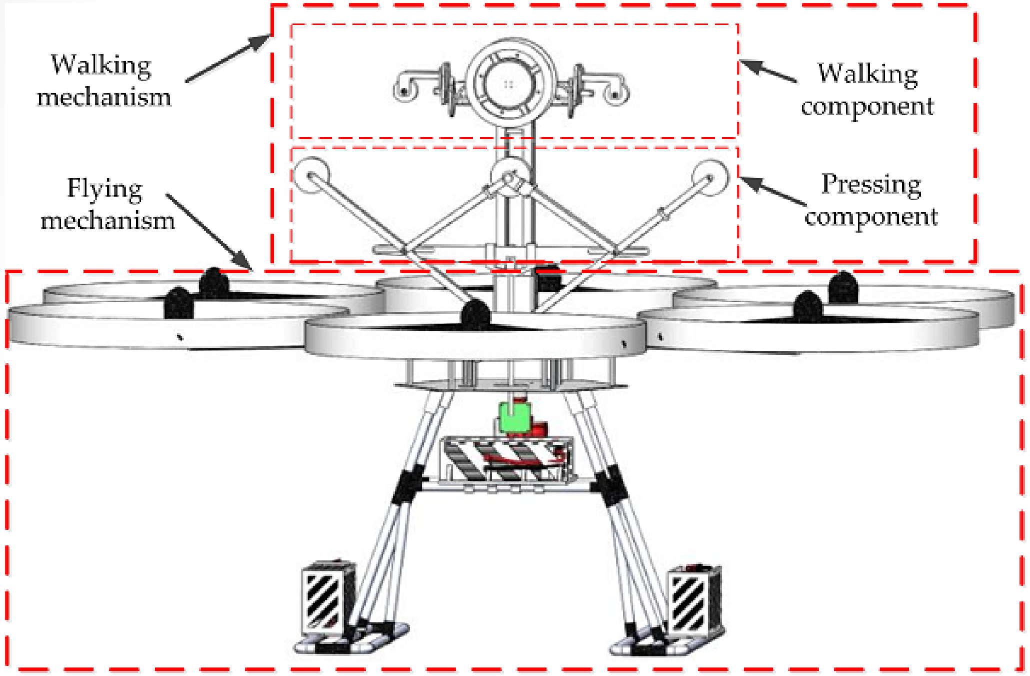
- Poor operating time
- Difficult to automatize
- Limited data acquisition

https://www.flir.com/products/skyranger-r70/?vertical=uas&segment=uis; https://ieeexplore.ieee.org/document/5624430;

HYBRID ROBOTS

- Best of both worlds

- More complex
- Balance of battery life vs. data acquisition

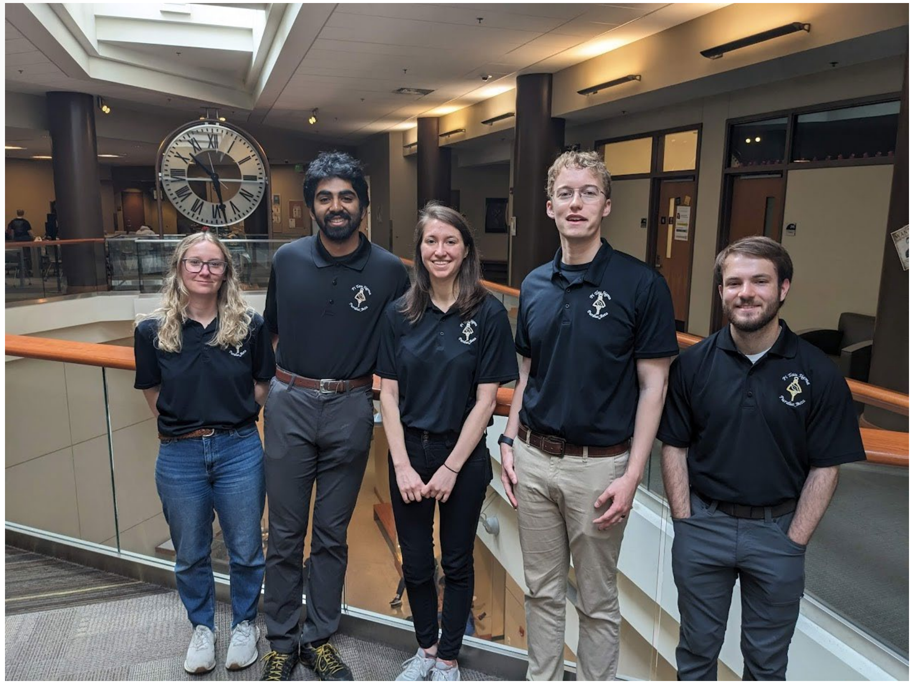


https://ms.copernicus.org/articles/13/257/2022/; https://dl.acm.org/doi/abs/10.1145/3483845.3483850;

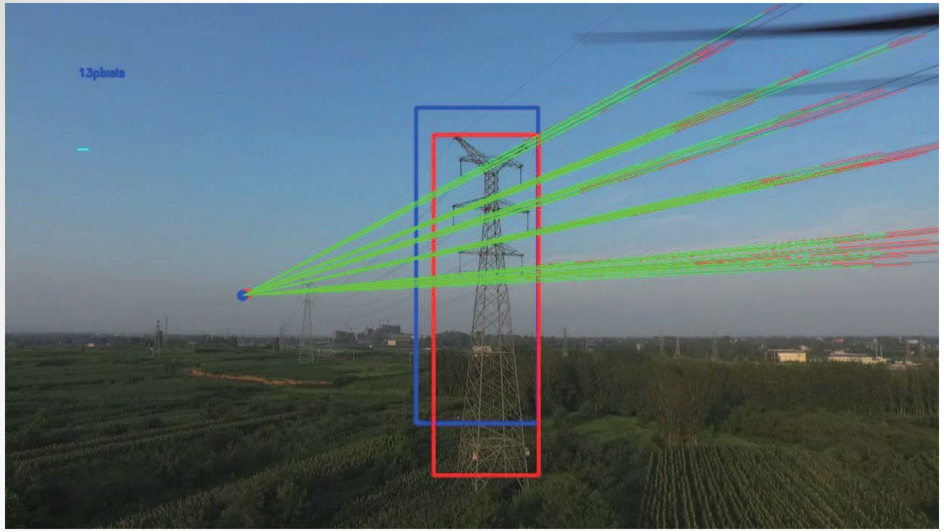
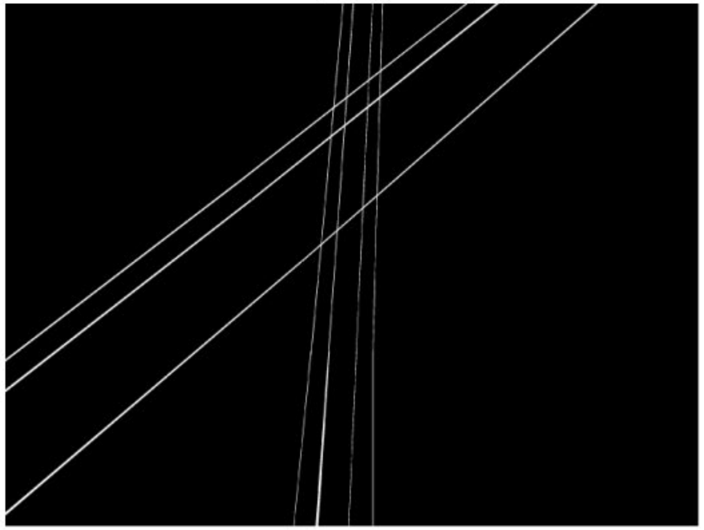
OUR ATTEMPT

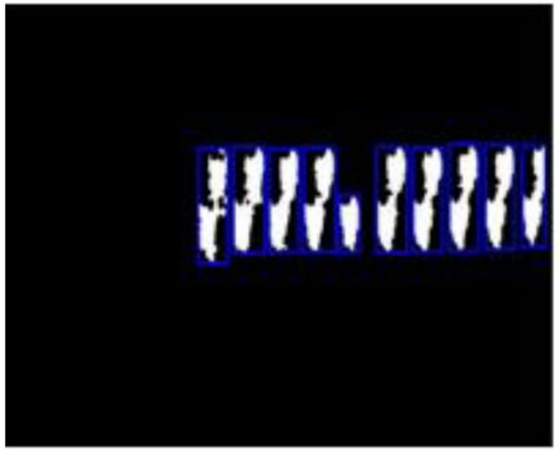
ERDC

- ~4-month time frame
- $1000 budget
- Not autonomous
- No sensing capabilities
- Possible with more resources and time



OBJECT RECOGNITION



https://ieeexplore.ieee.org/document/8377502; https://www.mdpi.com/1996-1073/12/3/543; https://ieeexplore.ieee.org/document/8822756;

OTHER CHALLENGES

ERDC

Battery Life

Obstacle Detection and Navigation

Environmental Effects

https://icon-icons.com/;

SUMMARY

ERDC

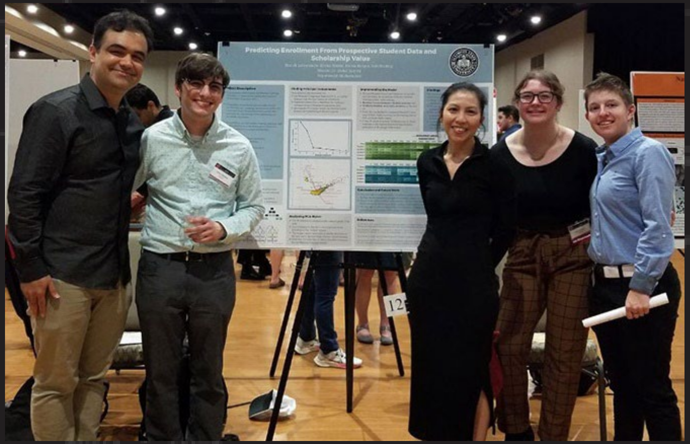
Power Lines are Valuable

Deserve Better Inspections

Can Improve Drastically with Resources

INTRODUCTION

- Robert Skudnig
- ISU Graduate '24
- Pure and Applied Mathematics
- Psychology
- Master's projected '25



SUSTAINMENT MANAGEMENT SYSTEM TCX

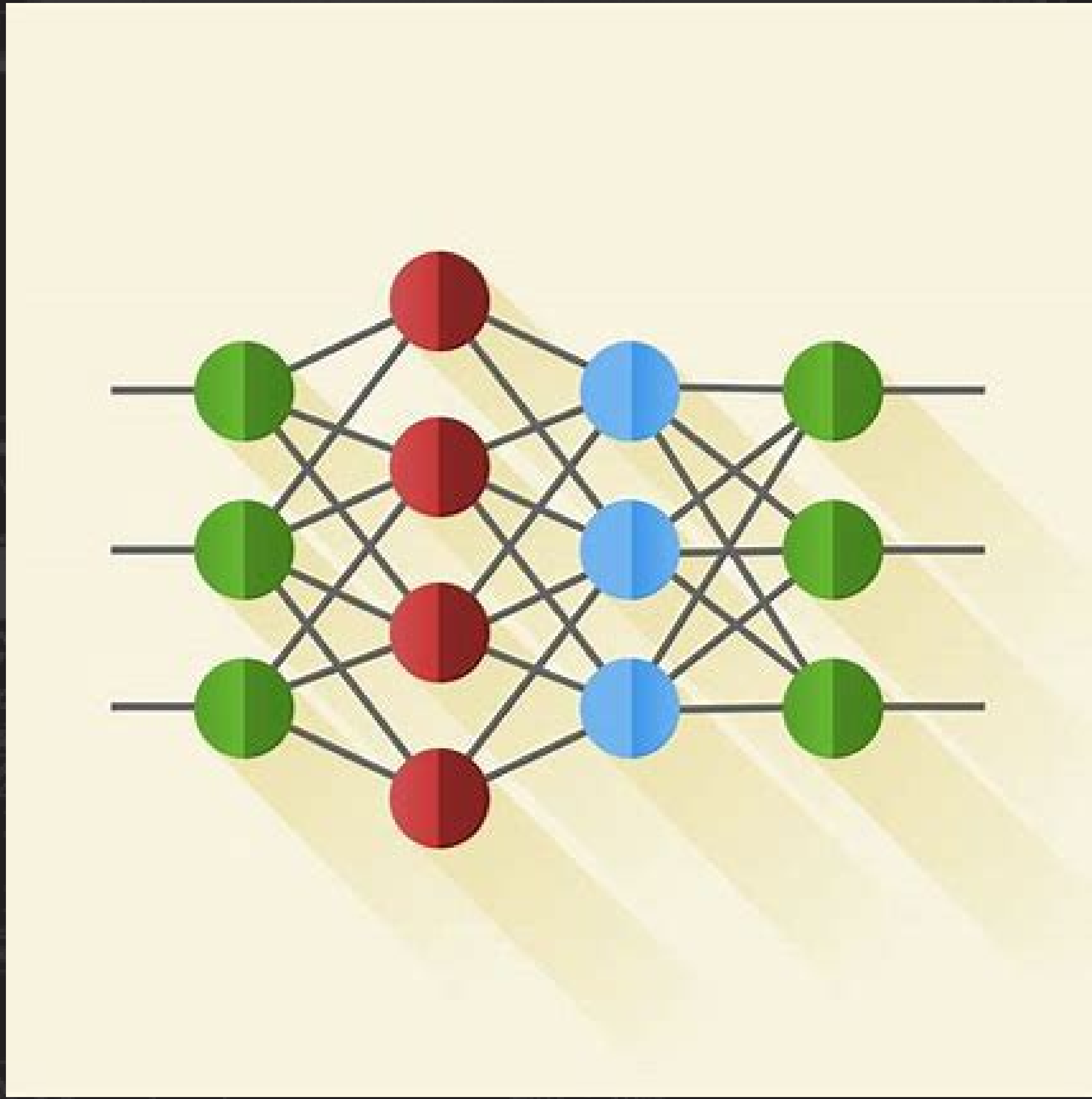
U.S. ARMY

US Army Corps of Engineers®

ERDC

NEURAL NETWORKS

Overview of a common Machine Learning concept

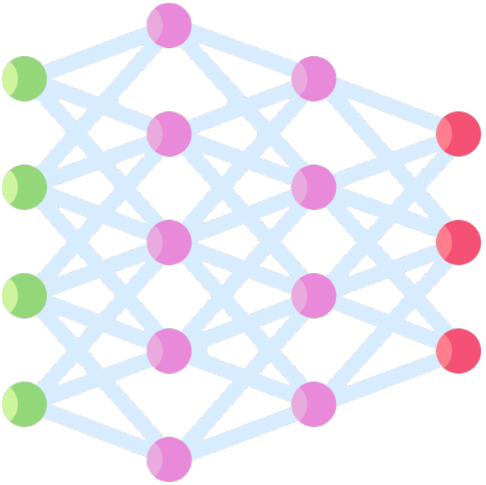


U.S. ARMY

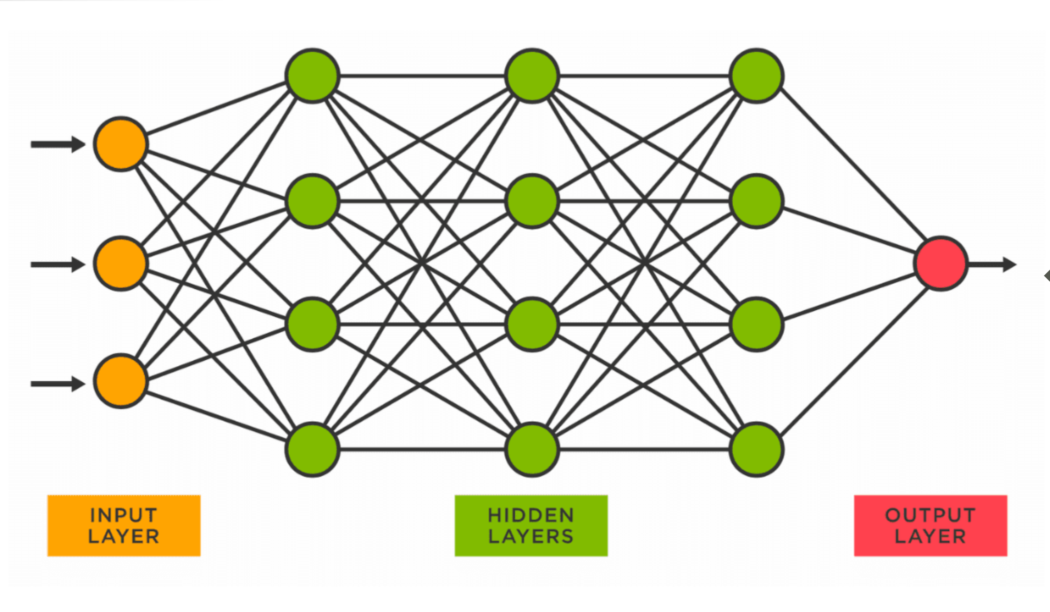
US Army Corps of Engineers®

ERDC

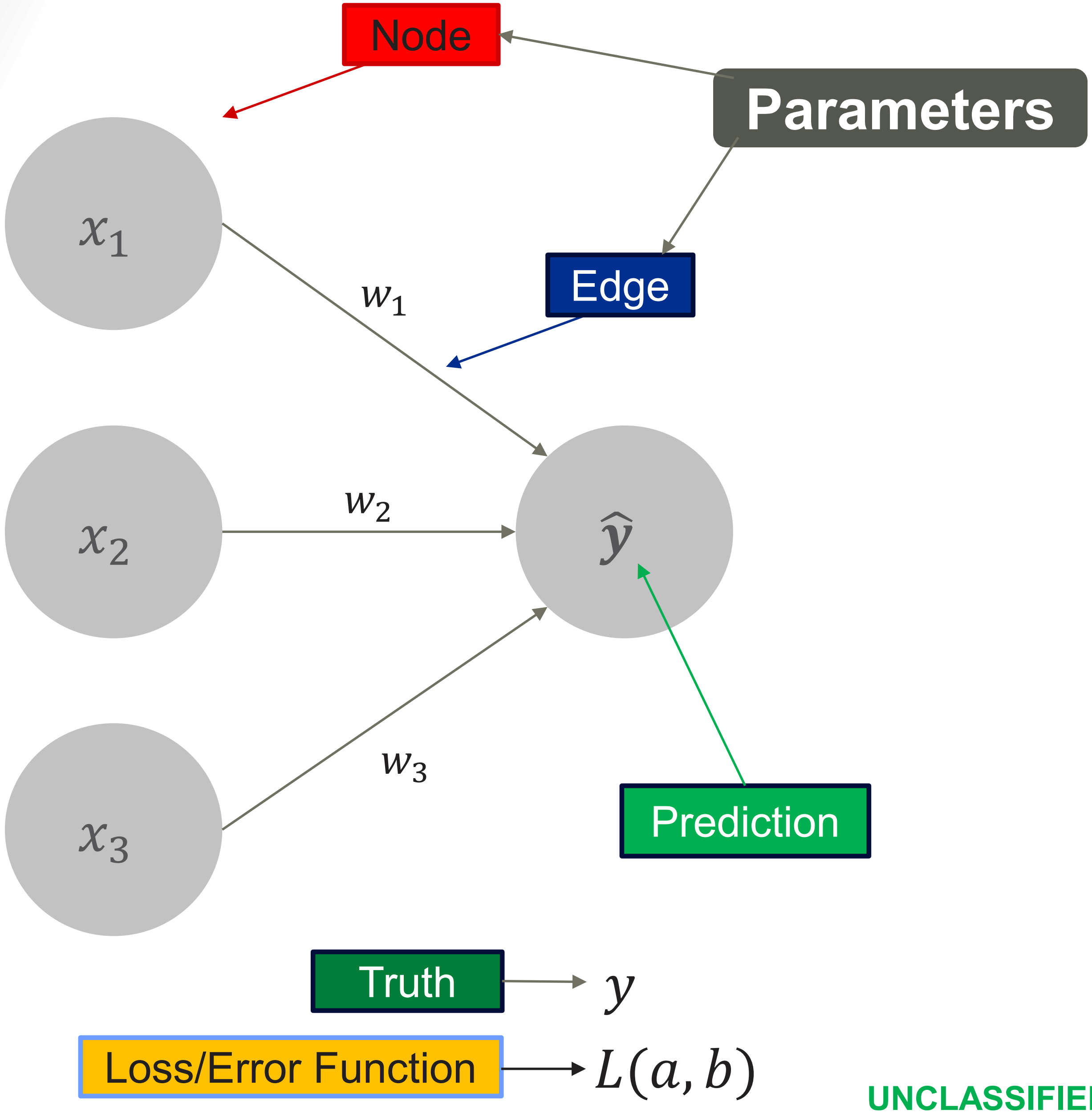
WHY SHOULD WE CARE?



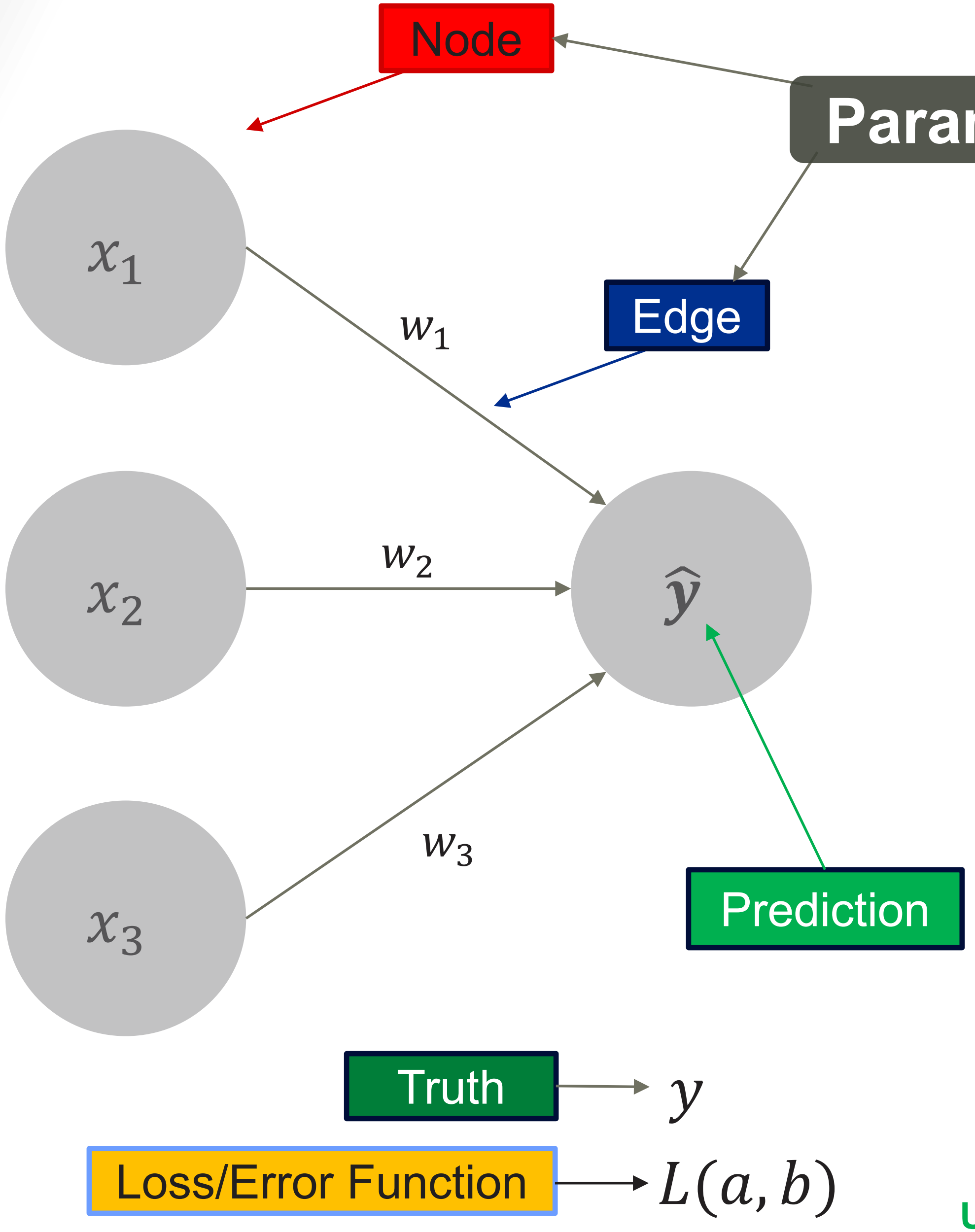
WHAT ARE NEURAL NETWORKS?



HOW DO THEY LEARN?



HOW DO THEY LEARN?

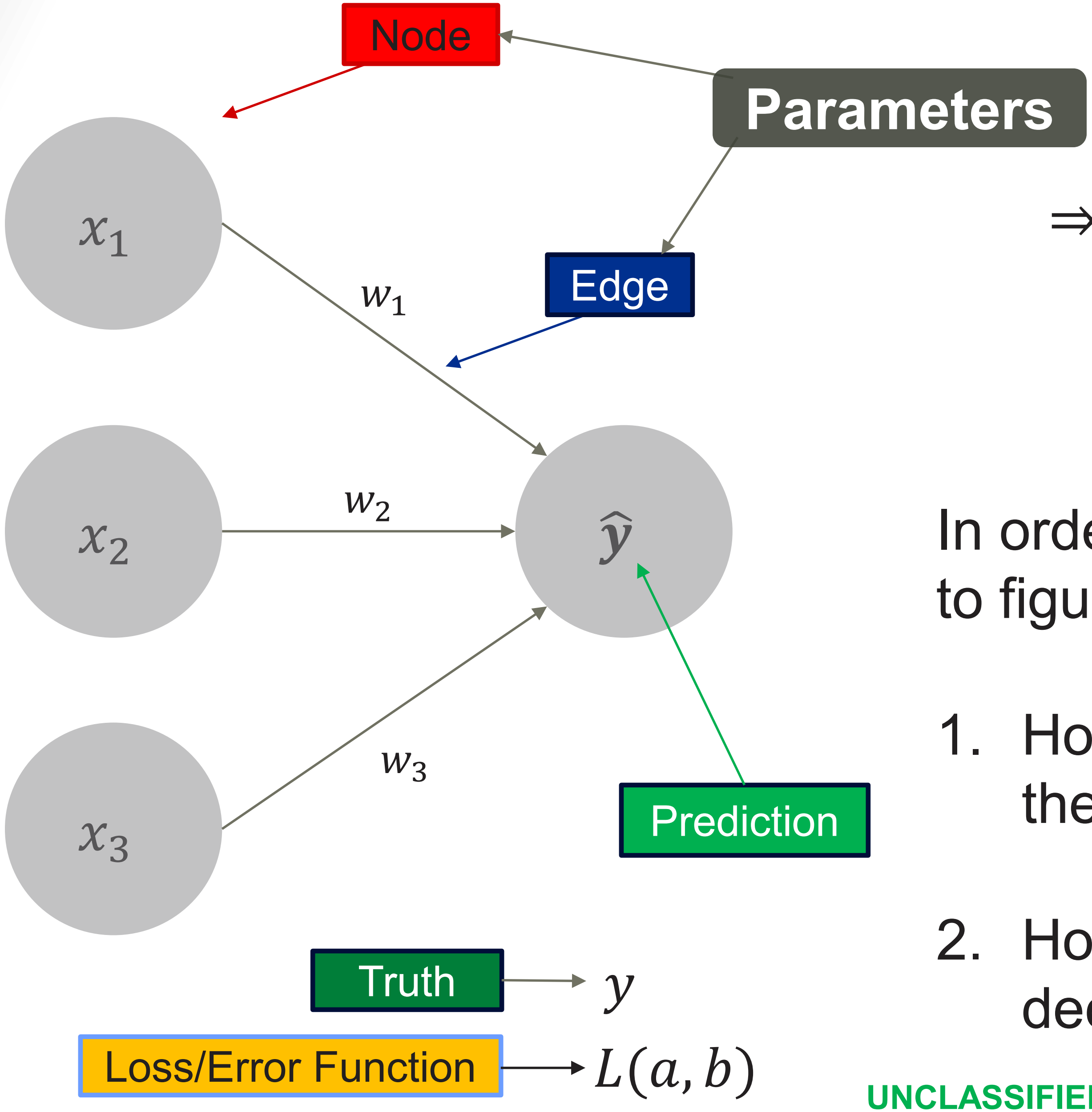


$$\hat{y} = x_1 w_1 + x_2 w_2 + x_3 w_3$$

$$\Rightarrow L(y, \hat{y}) = L(y, x_1 w_1 + x_2 w_2 + x_3 w_3)$$

$$\underset{*}{\Rightarrow} L = f(x_1, w_1, x_2, w_2, x_3, w_3)$$

HOW DO THEY LEARN?



$$\hat{y} = x_1 w_1 + x_2 w_2 + x_3 w_3$$

$$\Rightarrow L(y, \hat{y}) = L(y, x_1 w_1 + x_2 w_2 + x_3 w_3)$$

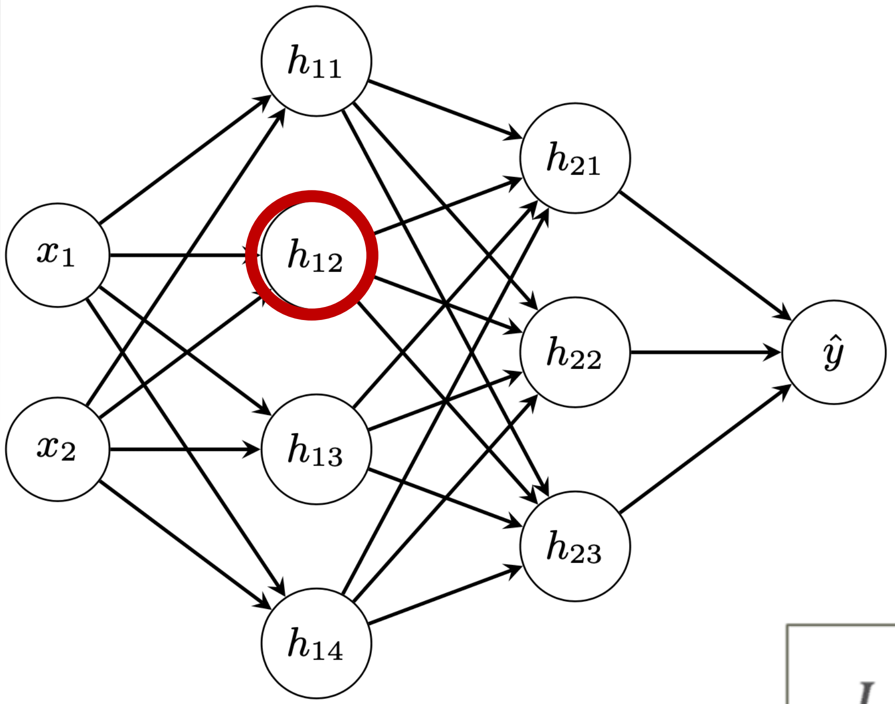
$$\underset{*}{\Rightarrow} L = f(x_1, w_1, x_2, w_2, x_3, w_3)$$

In order to make the model “learn,” we need to figure out two things:

1. How do the parameters generally impact the loss?
2. How do we change the parameters to decrease the loss?

HOW DO THE PARAMETERS IMPACT THE LOSS?

ERDC

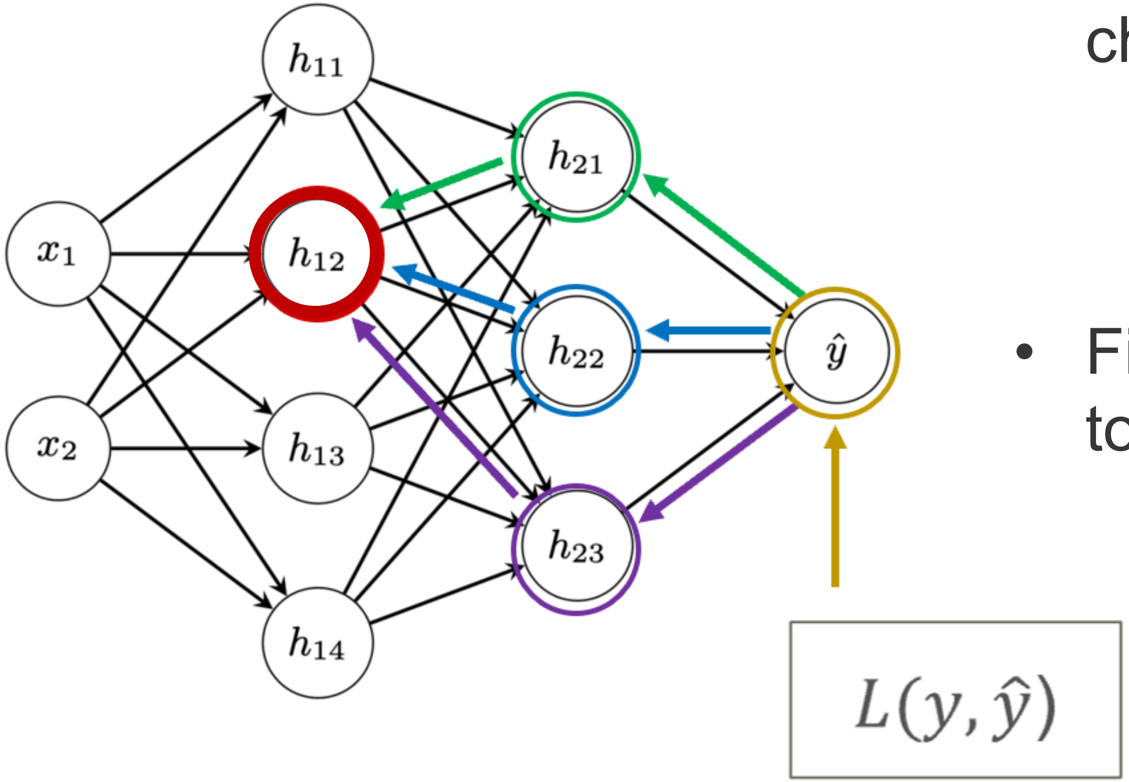


$L(y, \hat{y})$

- Example: How does the loss (L) change as we change node h_{12}?

$$\frac{\partial L}{\partial h_{12}} = ?$$

HOW DO THE PARAMETERS IMPACT THE LOSS?

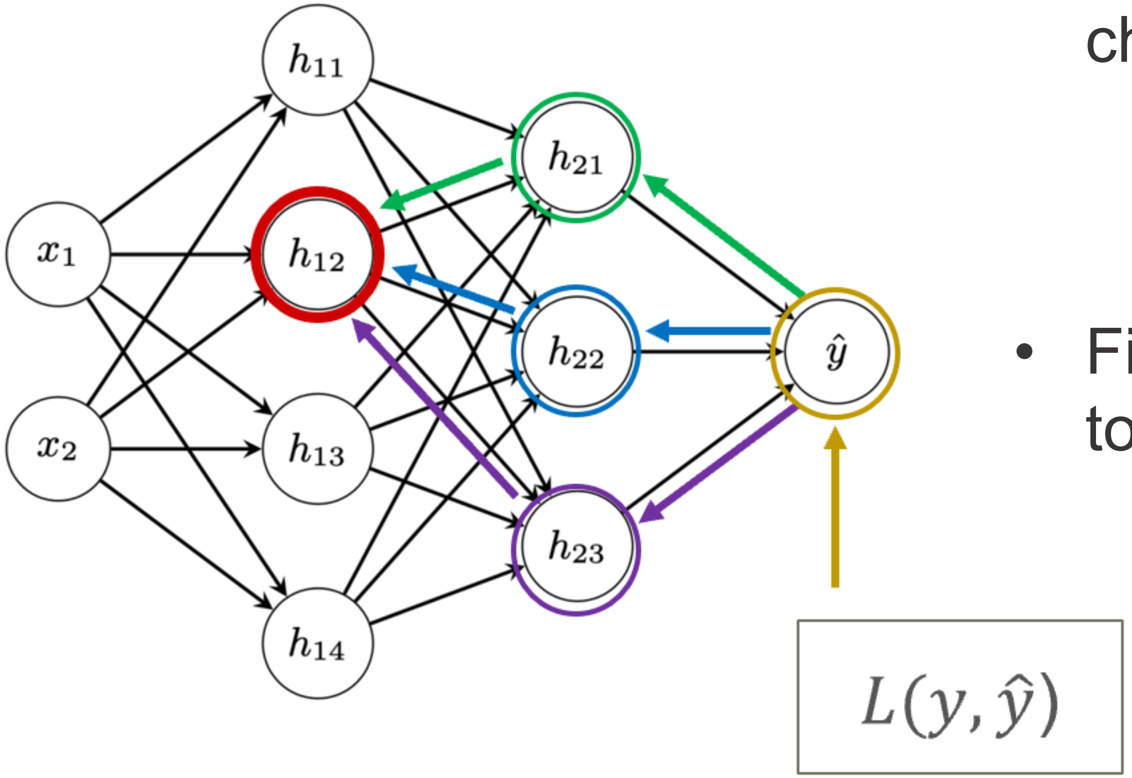


- Example: How does the loss (L) change as we change node h_{12}?

$$\frac{\partial L}{\partial h_{12}} = ?$$

- First we must determine every possible path back to the parameter in question

HOW DO THE PARAMETERS IMPACT THE LOSS?



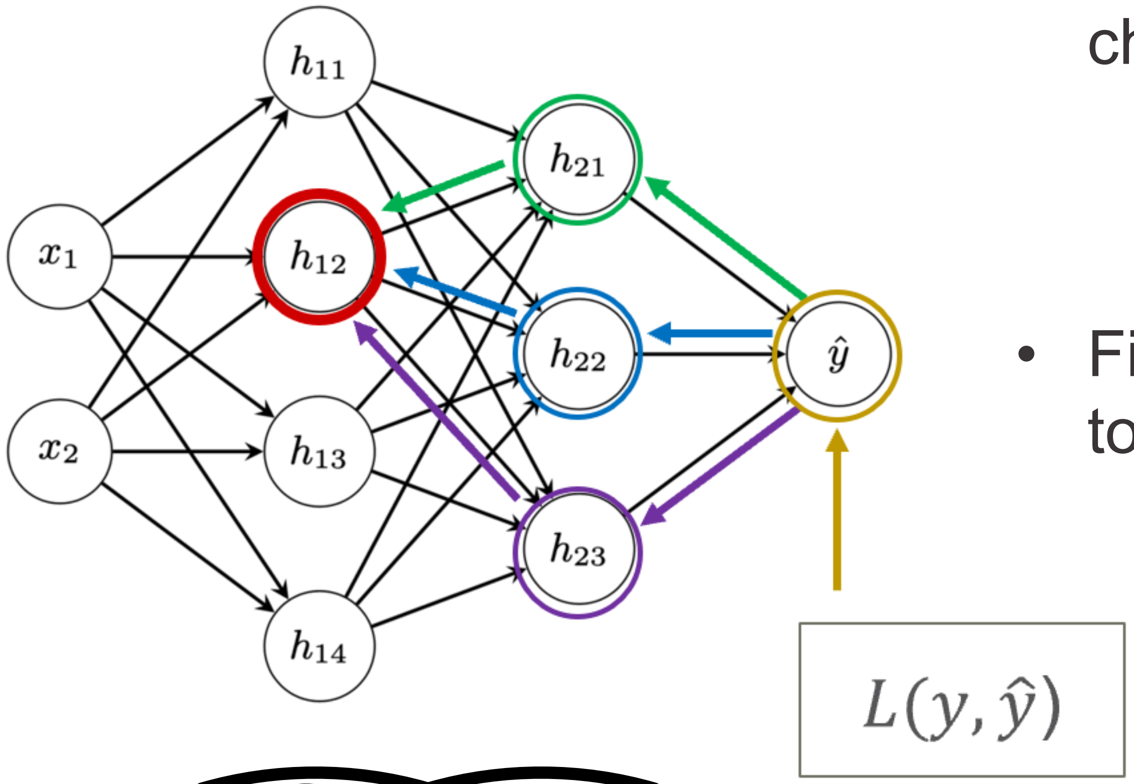
- Example: How does the loss (L) change as we change node h_{12}?

$$\frac{\partial L}{\partial h_{12}} = ?$$

- First we must determine every possible path back to the parameter in question

$$= \frac{\partial L}{\partial \hat{y}} \frac{\partial \hat{y}}{\partial h_{21}} \frac{\partial h_{21}}{\partial h_{12}} + \frac{\partial L}{\partial \hat{y}} \frac{\partial \hat{y}}{\partial h_{22}} \frac{\partial h_{22}}{\partial h_{12}} + \frac{\partial L}{\partial \hat{y}} \frac{\partial \hat{y}}{\partial h_{23}} \frac{\partial h_{23}}{\partial h_{12}}$$

HOW DO THE PARAMETERS IMPACT THE LOSS?

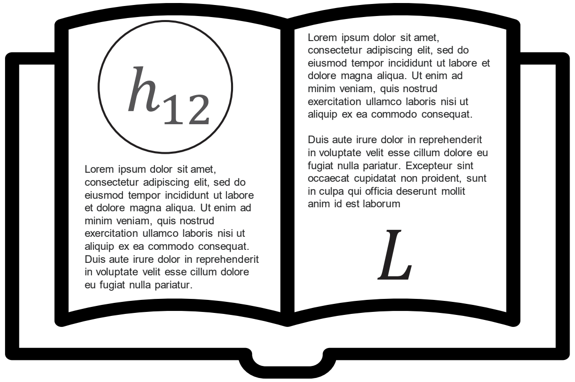


- Example: How does the loss (L) change as we change node h_{12}?

$$\frac{\partial L}{\partial h_{12}} = ?$$

- First we must determine every possible path back to the parameter in question

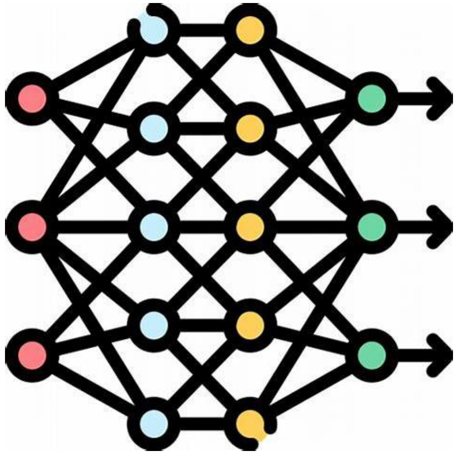
$$= \frac{\partial L}{\partial \hat{y}} \frac{\partial \hat{y}}{\partial h_{21}} \frac{\partial h_{21}}{\partial h_{12}} + \frac{\partial L}{\partial \hat{y}} \frac{\partial \hat{y}}{\partial h_{22}} \frac{\partial h_{22}}{\partial h_{12}} + \frac{\partial L}{\partial \hat{y}} \frac{\partial \hat{y}}{\partial h_{23}} \frac{\partial h_{23}}{\partial h_{12}}$$



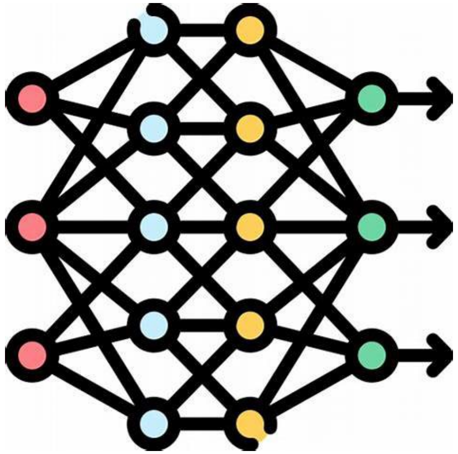
- This is called **backward propagation**

HOW DO WE DECREASE THE LOSS?

- Impossible to find the **absolute best** parameter set

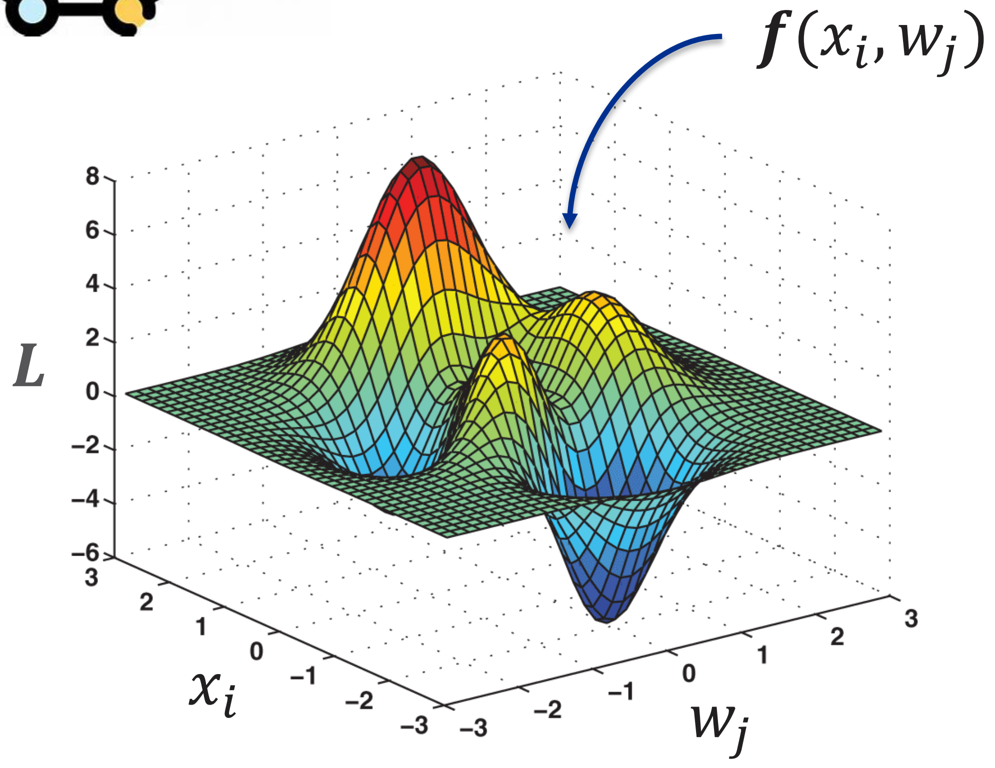


HOW DO WE DECREASE THE LOSS?



- Impossible to find the **absolute best** parameter set
- Let $\boldsymbol{L} = \boldsymbol{f}(\boldsymbol{\theta})$ where θ is the set of ALL parameters:

$$\theta = \{x_1, \dots, x_n, w_1, \dots, w_m\} = \begin{bmatrix} x_1 \\ \vdots \\ x_n \\ w_1 \\ \vdots \\ w_m \end{bmatrix}$$

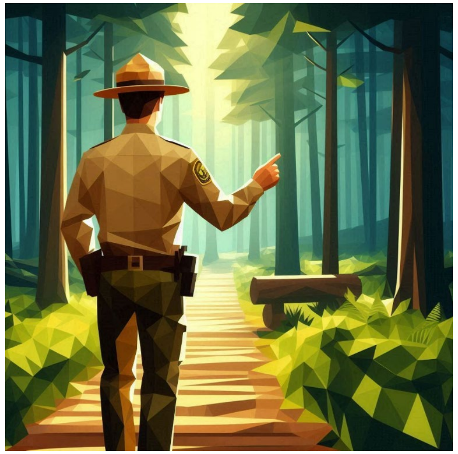


HOW DO WE DECREASE THE LOSS?

Let $\boldsymbol{\theta_0}$ denote the INITIAL set of model parameters. This is usually randomized.

In order to find a *better* set, we can perform the following operation on our loss function:

$$\overrightarrow{grad_0} = \nabla \boldsymbol{L} = \nabla \boldsymbol{f}(\boldsymbol{\theta_0})$$



To obtain our next set of *improved* parameters θ_1 :

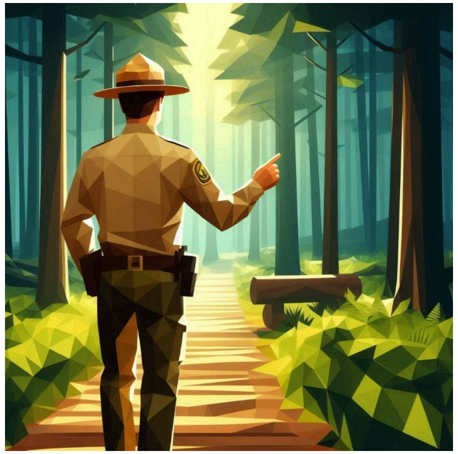
$$\theta_1 = \theta_0 - \overrightarrow{grad_0}$$

HOW DO WE DECREASE THE LOSS?

Let $\boldsymbol{\theta_0}$ denote the INITIAL set of model parameters. This is usually randomized.

In order to find a *better* set, we can perform the following operation on our loss function:

$$\overrightarrow{grad_0} = \nabla \boldsymbol{L} = \nabla \boldsymbol{f}(\boldsymbol{\theta_0})$$



To obtain our next set of *improved* parameters θ_1 :

$$\theta_1 = \theta_0 - \overrightarrow{grad_0}$$

This process repeats until the user is satisfied with the results:

$$\overrightarrow{grad_i} = \nabla L = \nabla f(\theta_i)$$

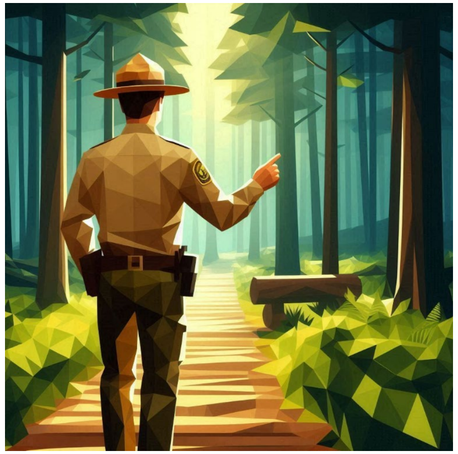
$$\theta_{i+1} = \theta_i - \overrightarrow{grad_i}$$

HOW DO WE DECREASE THE LOSS?

Let $\boldsymbol{\theta_0}$ denote the INITIAL set of model parameters. This is usually randomized.

In order to find a *better* set, we can perform the following operation on our loss function:

$$\overrightarrow{grad_0} = \nabla \boldsymbol{L} = \nabla \boldsymbol{f}(\boldsymbol{\theta_0})$$



To obtain our next set of *improved* parameters θ_1 :

$$\theta_1 = \theta_0 - \overrightarrow{grad_0}$$

This process repeats until the user is satisfied with the results:

$$\overrightarrow{grad_i} = \nabla L = \nabla f(\theta_i)$$

$$\theta_{i+1} = \theta_i - \overrightarrow{grad_i}$$

$$\begin{bmatrix} x_1^{i+1} \\ \vdots \\ x_n^{i+1} \\ w_1^{i+1} \\ \vdots \\ w_m^{i+1} \end{bmatrix} = \begin{bmatrix} x_1^{i} \\ \vdots \\ x_n^{i} \\ w_1^{i} \\ \vdots \\ w_m^{i} \end{bmatrix} - \begin{bmatrix} \partial x_1^{i} \\ \vdots \\ \partial x_n^{i} \\ \partial w_1^{i} \\ \vdots \\ \partial w_m^{i} \end{bmatrix}$$

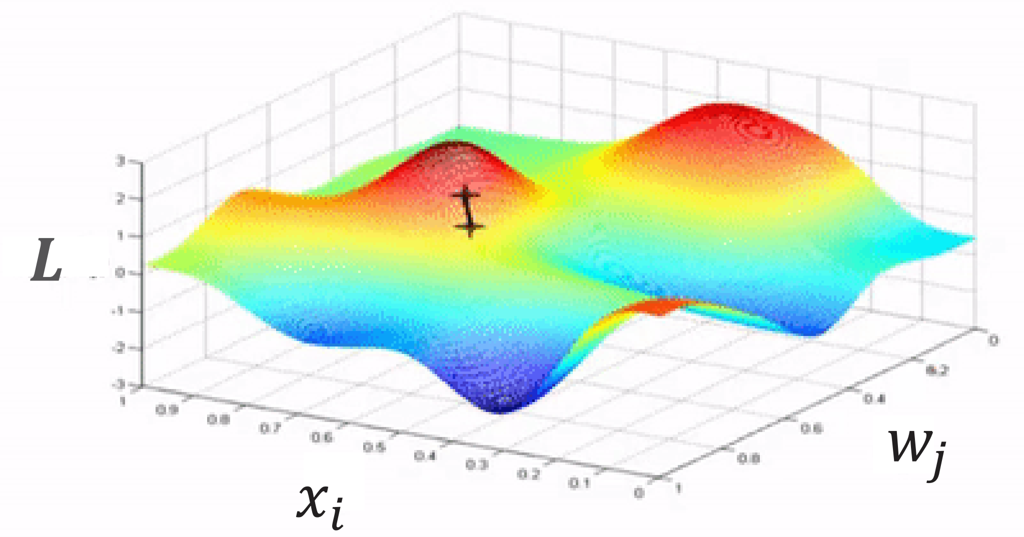
This is known as **gradient descent**

FOGGY MOUNTAIN ANALOGY

ERDC

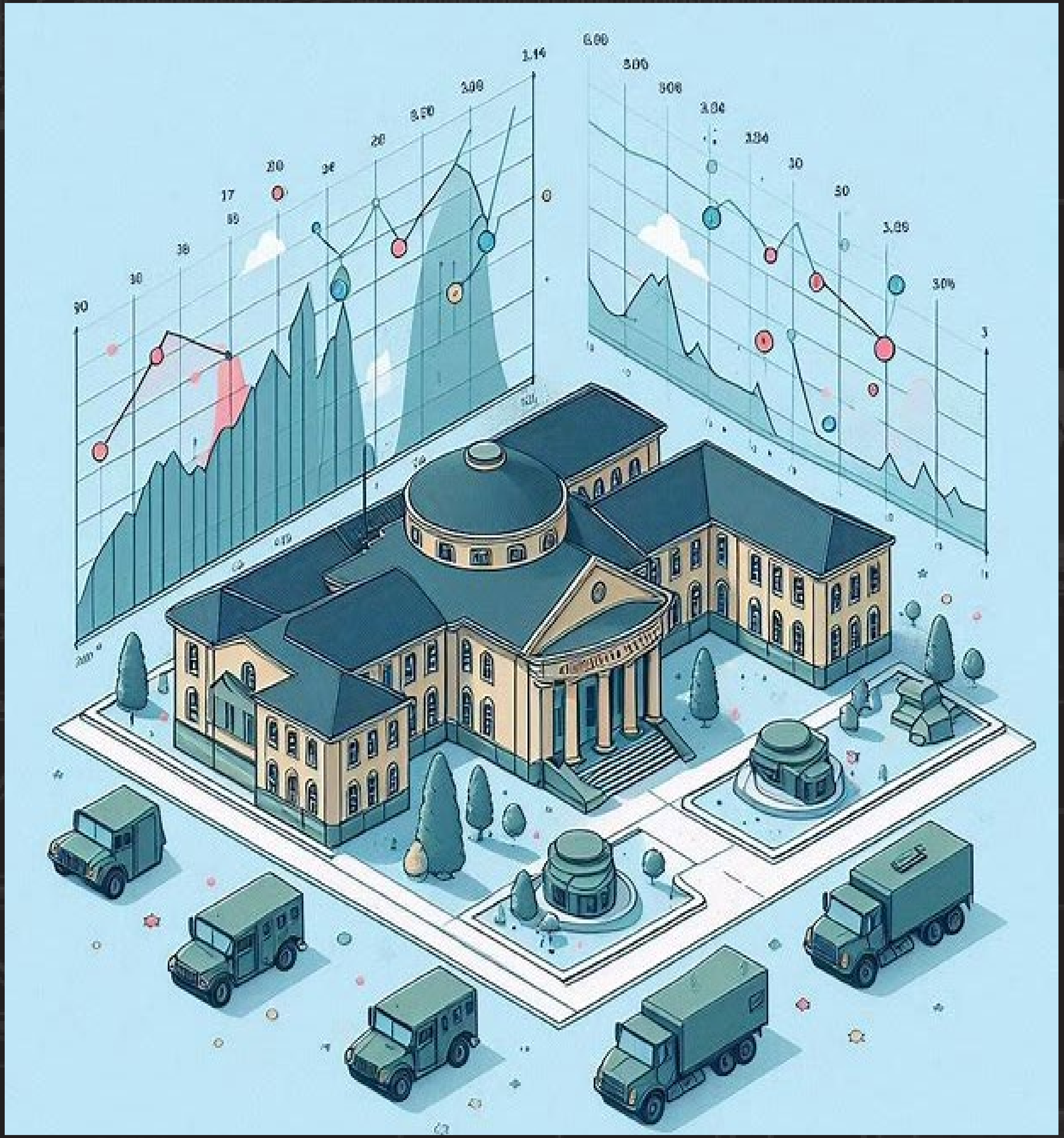
- Imagine a squad of soldiers on a foggy mountain returning to their outpost at the mountain's base.
- They don't know the exact route to the bottom, but they can tell which direction declines where they are standing.
- They continue down for as long as they can to try and reach the outpost.

- **Connection**:
  - Latitude-Longitude are the model **parameters**
  - Altitude is the **error/loss** at that coordinate
  - Slope is the **gradient** that is guiding them
    - There are other important concepts in the training process.**



MISSION DEPENDENCY INDEX PREDICTION

Application of Neural Networks



U.S. ARMY

US Army Corps of Engineers®

ERDC

MISSION DEPENDENCY INDEX (MDI)

MDI Overview

U.S. ARMY

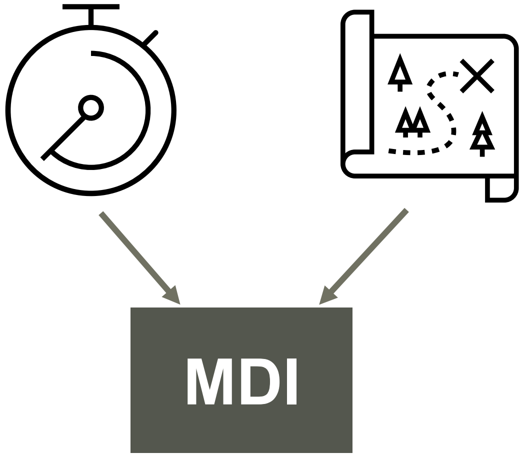
US Army Corps of Engineers®

ERDC

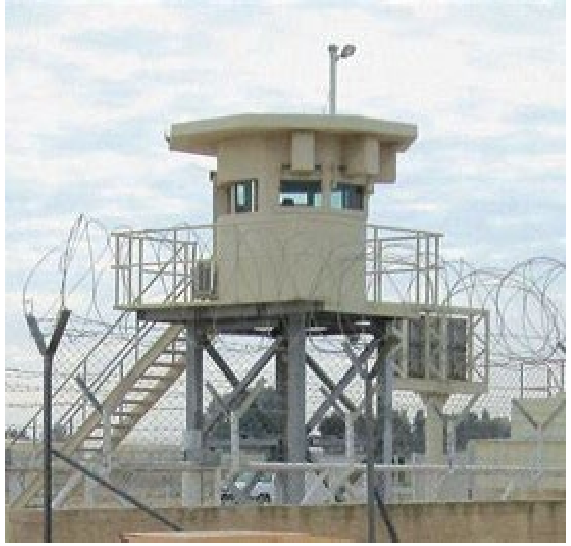
WHAT IS MISSION DEPENDENCY INDEX?

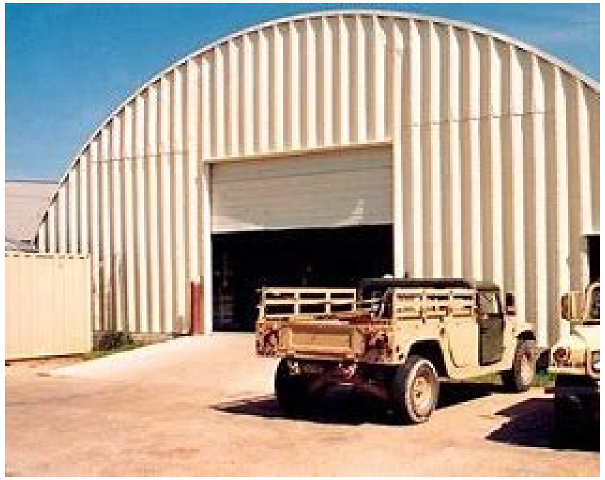
- **Mission Dependency Index** (MDI)
- Usefulness to SMS
- Summarizes:
  - **Interruptability**
  - **Replicability**
- Matrix:

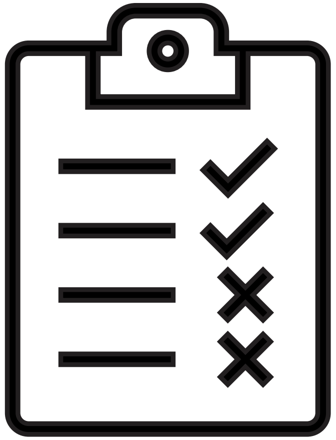
MISSION DEPENDENCY INDEX					
MDI		Question 1 INTERRUPTABILITY How fast would the mission be impacted if the asset's operations were interrupted?			
		IMMEDIATE < 15 minutes	BRIEF < 24 hours	SHORT < 7days	PROLONGED > 7 days
Question 2 REPLICABILITY How difficult would it be to relocate the asset's mission capabilities?	IMPOSSIBLE	100	88	76	64
	EXTREMELY DIFFICULT	92	80	68	56
	DIFFICULT	84	72	60	48
	POSSIBLE	76	64	52	40



THE MDI SURVEY







THE SETUP

The Neural Network Setup for MDI Prediction

U.S. ARMY

US Army Corps of Engineers®

ERDC

TRAINING SETUP

Input Data

- Real Property Asset information
- PRV
- Size
- Facility Built Date
- Unit present (if any)
  - Type, Name, Hierarchy
- CATCODE
  - Name
- Misc.

Task:

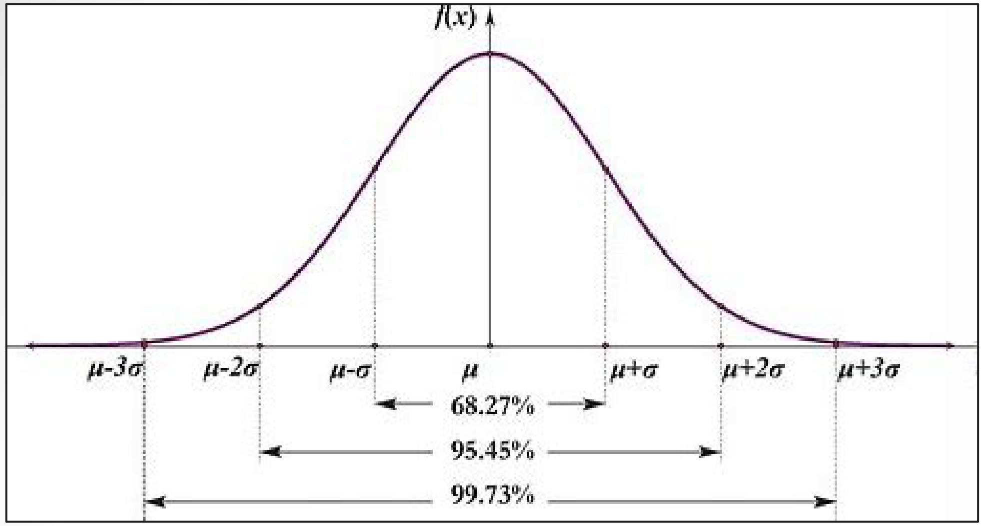
Predict Interruptability and Replicability responses

** MDI was inferred from these

MISSION DEPENDENCY INDEX					
MDI		Question 1 INTERRUPTABILITY How fast would the mission be impacted if the asset's operations were interrupted?			
		IMMEDIATE < 15 minutes	BRIEF < 24 hours	SHORT < 7days	PROLONGED > 7 days
Question 2 REPLICABILITY How difficult would it be to relocate the asset's mission capabilities?	IMPOSSIBLE	100	88	76	64
	EXTREMELY DIFFICULT	92	80	68	56
	DIFFICULT	84	72	60	48
	POSSIBLE	76	64	52	40

DATA TREATMENT

- Most data was either **standardized** or **one-hot-encoded**



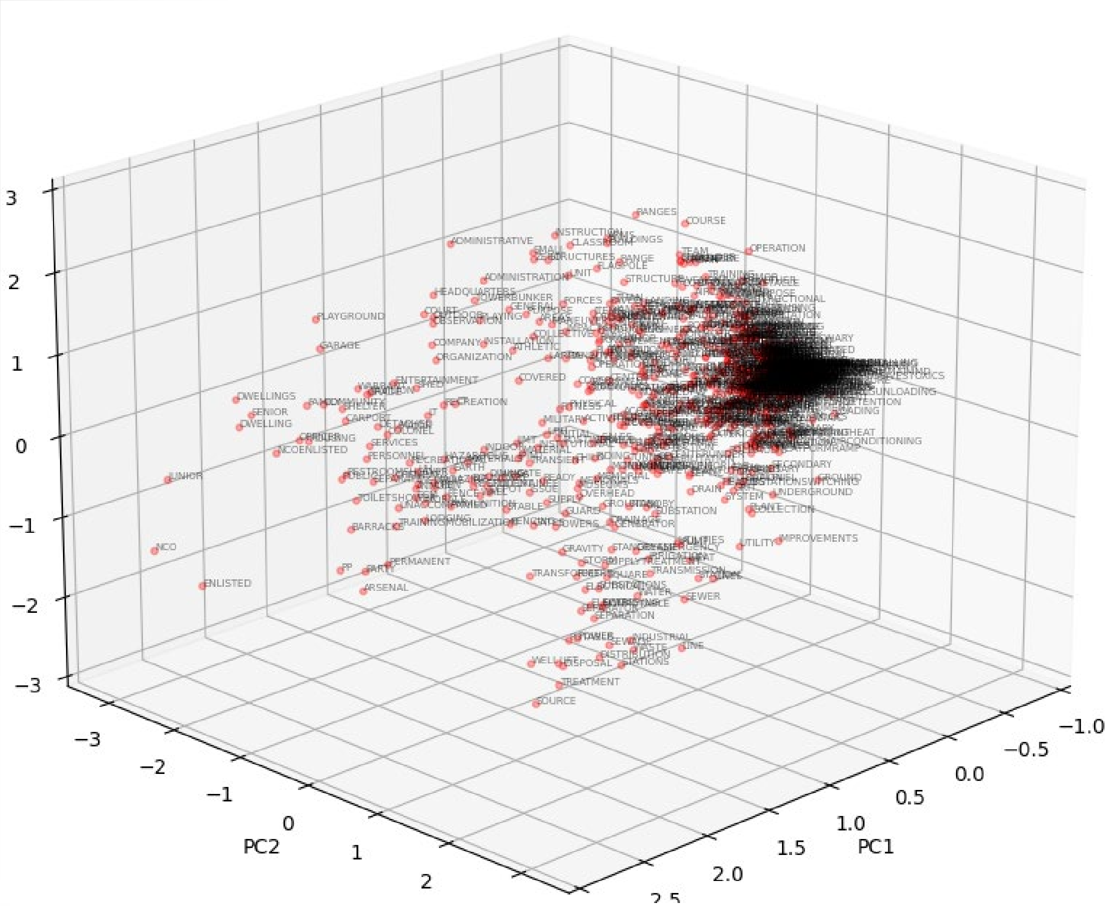
Color
Red
Red
Yellow
Green
Yellow

Red	Yellow	Green
1	0	0
1	0	0
0	1	0
0	0	1
0	1	0

- Outputs were also numerically represented
- Text data about units can’t be fed directly into these models
- There are too many CATCODES to one hot encode
- Need to apply another method

DATA TREATMENT

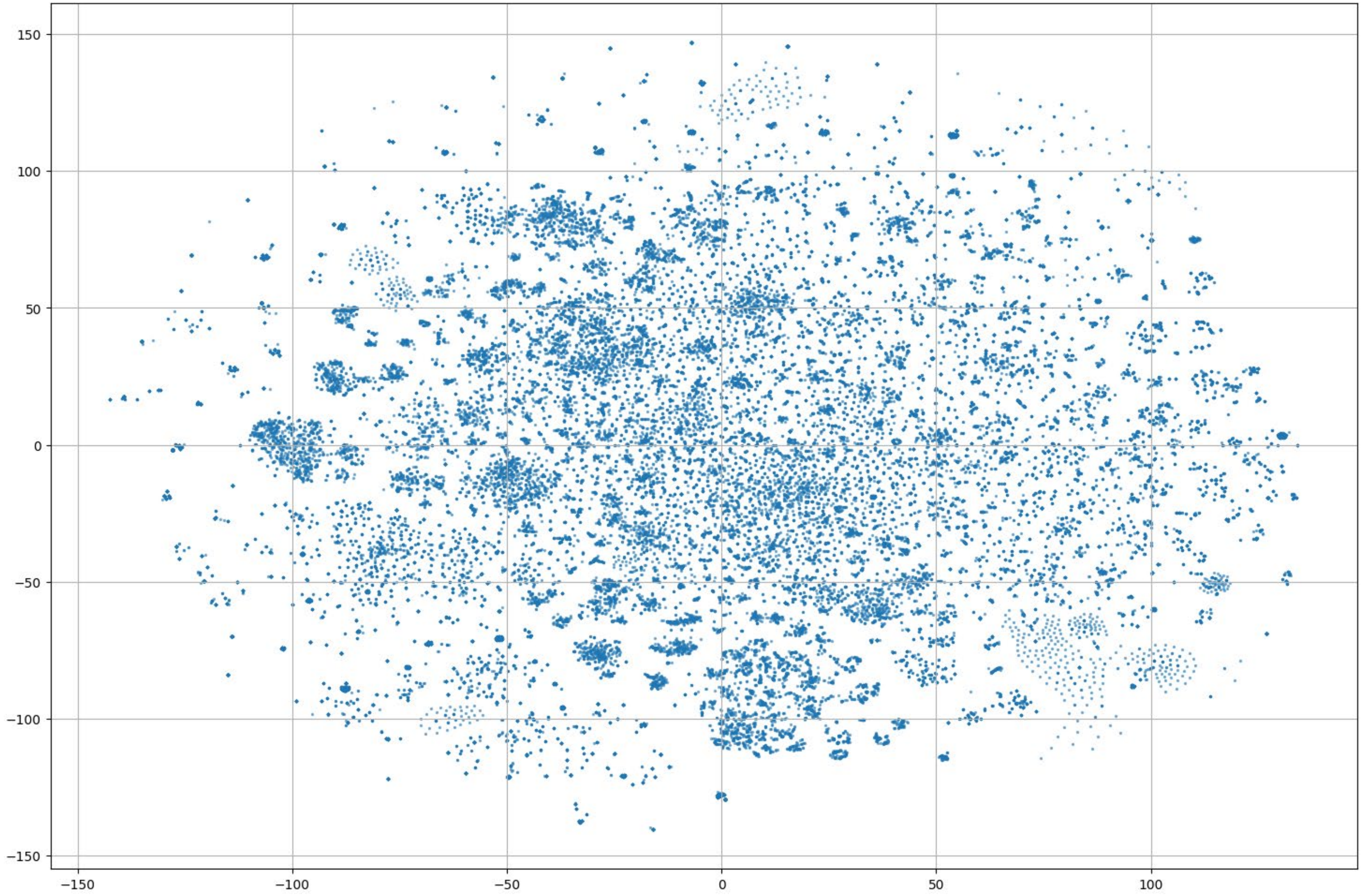
- Word embedding models:



DATA TREATMENT

ERDC

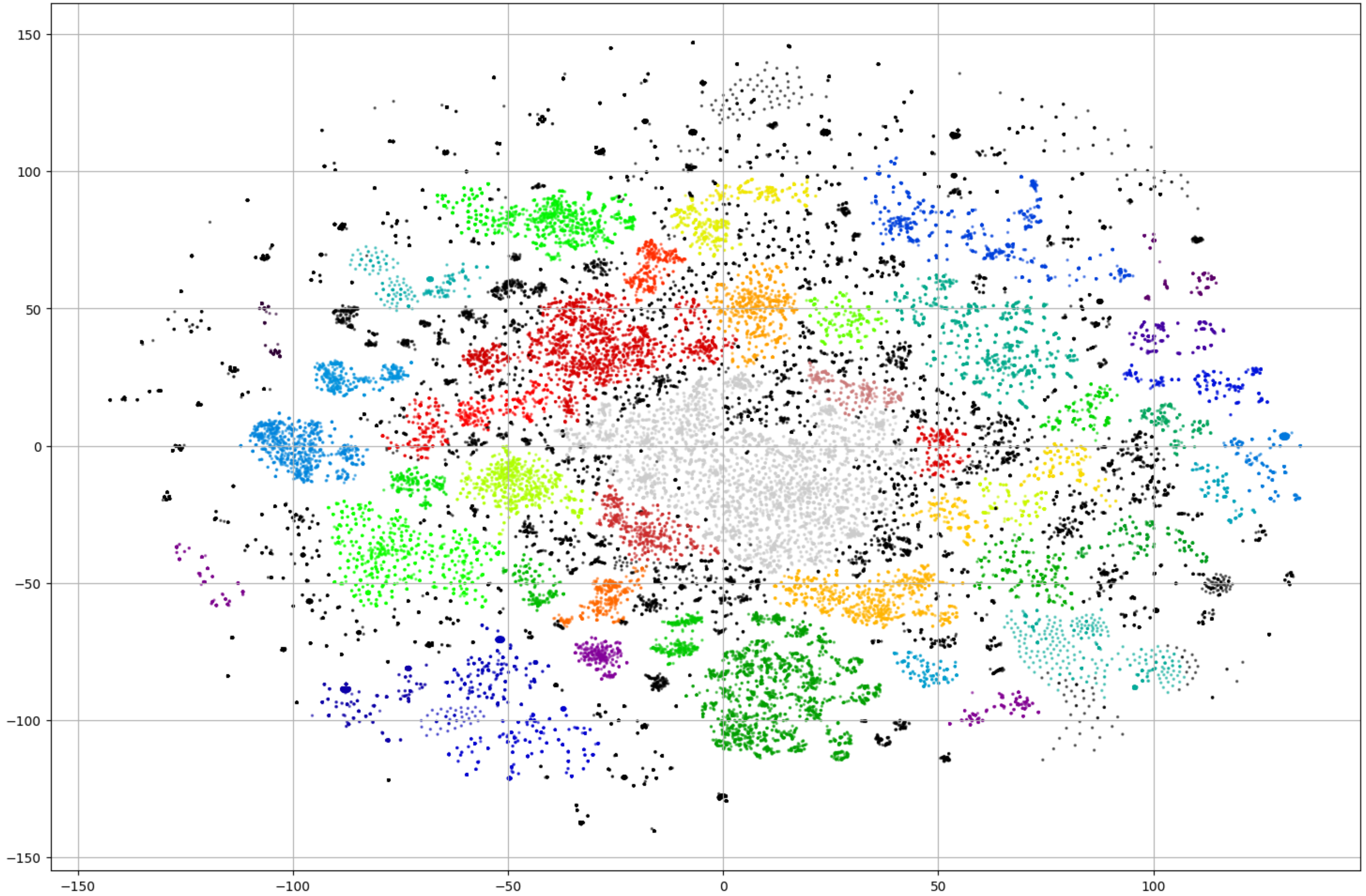
- Bidirectional Encoder Representations from Transformers (BERT)
- Application to Unit Name data:



DATA TREATMENT

ERDC

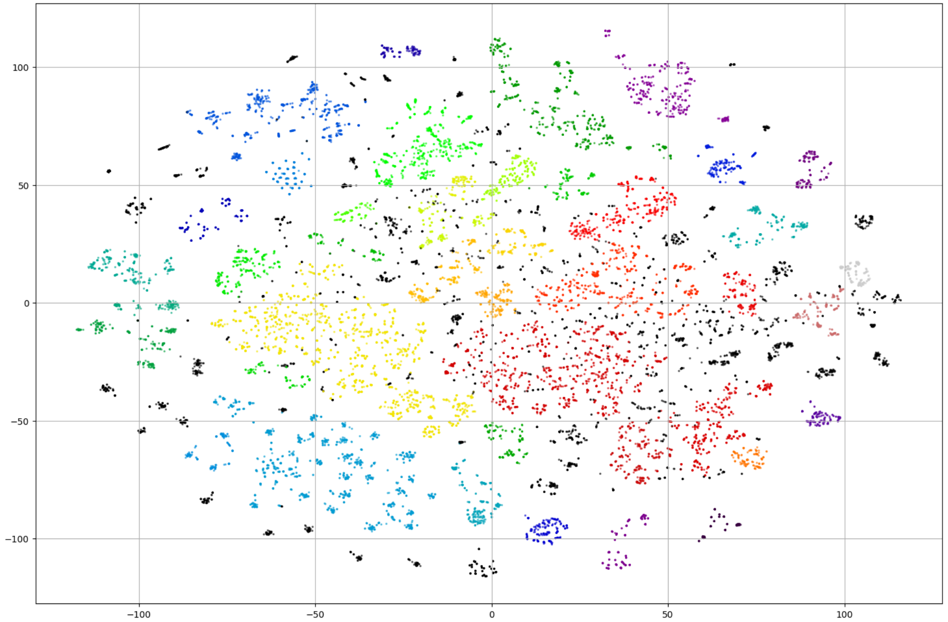
- Hierarchical Density-Based Spatial Clustering of Applications with Noise (HDBSCAN)
- Application to Unit Name data:



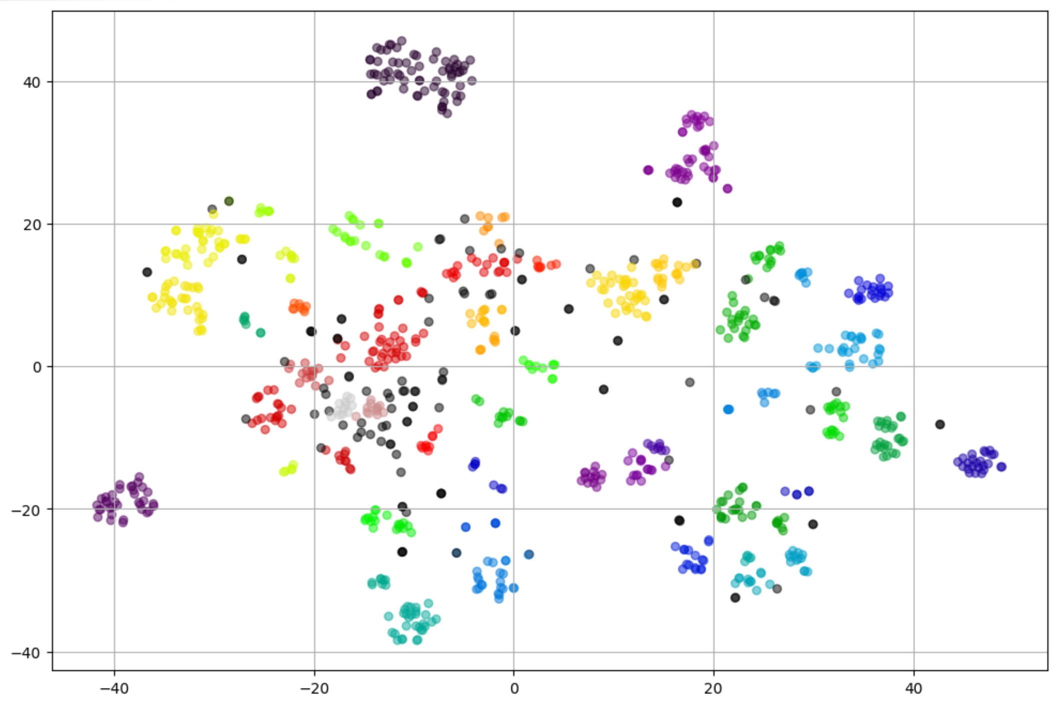
DATA TREATMENT

- All embedded-clustered results:

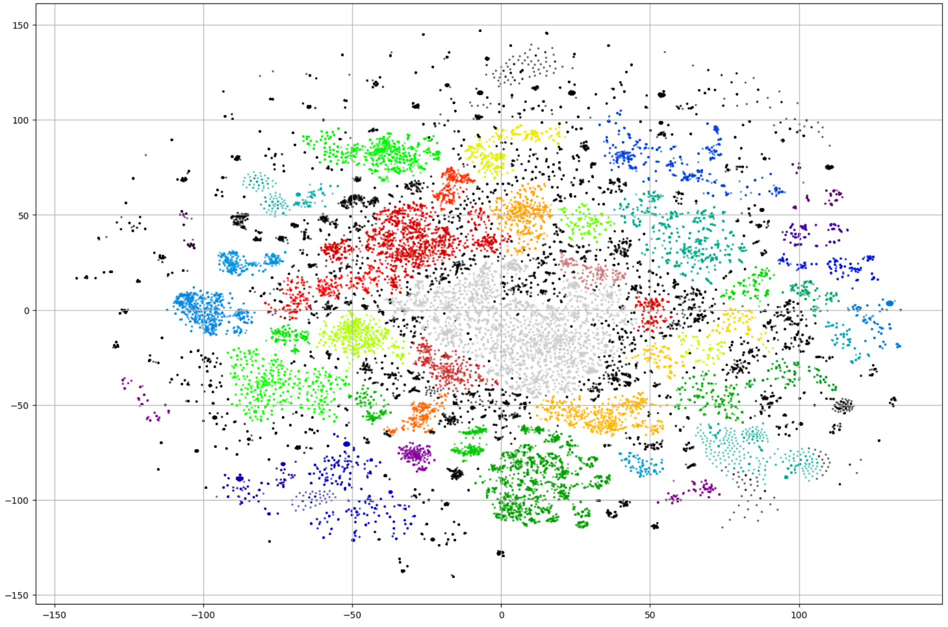
UIC (Hierarchy):



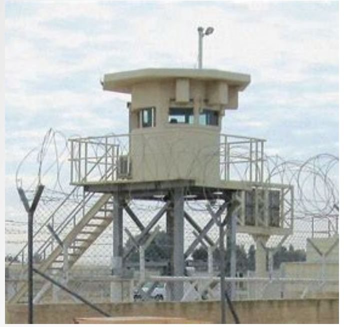
CATCODEs:

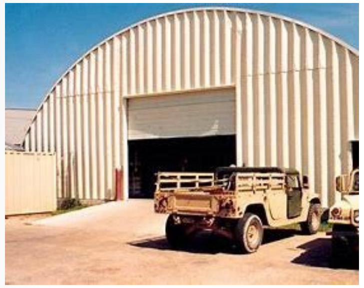


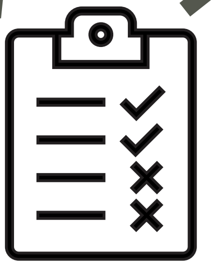
UIC (Name):

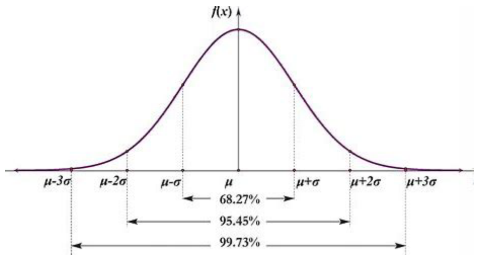


MODEL STRUCTURE

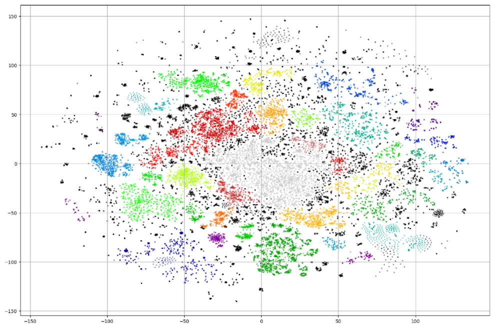


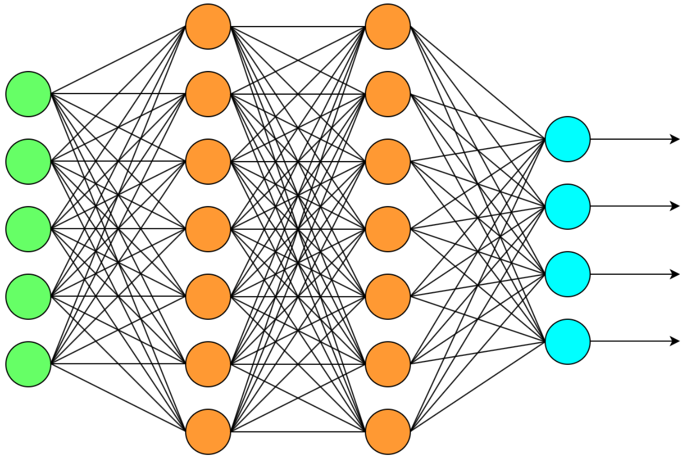




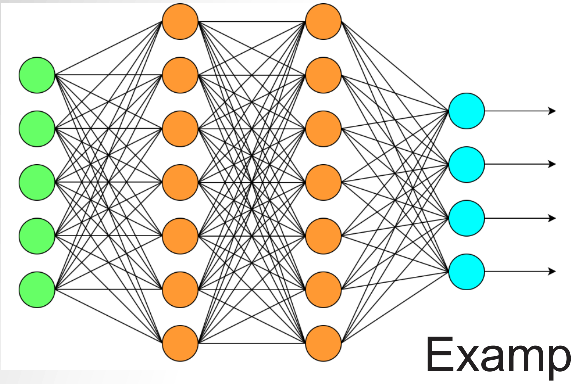


Red	Yellow	Green
1	0	0
1	0	0
0	1	0
0	0	1
0	1	0



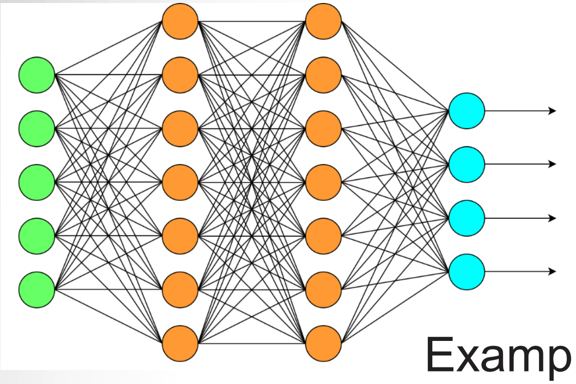


MODEL STRUCTURE



	Interruptability				Replicability			
	Prolonged	Short	Brief	Immediate	Possible	Difficult	Extremely Difficult	Impossible
	p_0	p_1	p_2	p_3	q_0	q_1	q_2	q_3
Example:	0.03	0.12	0.56	0.29	0.22	0.42	0.23	0.13

MODEL STRUCTURE



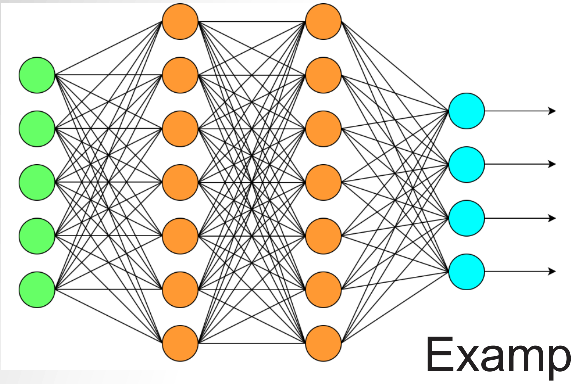
Example:

Interruptability				Replicability			
Prolonged	Short	Brief	Immediate	Possible	Difficult	Extremely Difficult	Impossible
p_0	p_1	p_2	p_3	q_0	q_1	q_2	q_3
0.03	0.12	**0.56**	0.29	0.22	**0.42**	0.23	0.13

(Brief, Difficult)

MODEL STRUCTURE

ERDC



Example:

Interruptability				Replicability			
Prolonged	Short	Brief	Immediate	Possible	Difficult	Extremely Difficult	Impossible
p_0	p_1	p_2	p_3	q_0	q_1	q_2	q_3
0.03	0.12	**0.56**	0.29	0.22	**0.42**	0.23	0.13

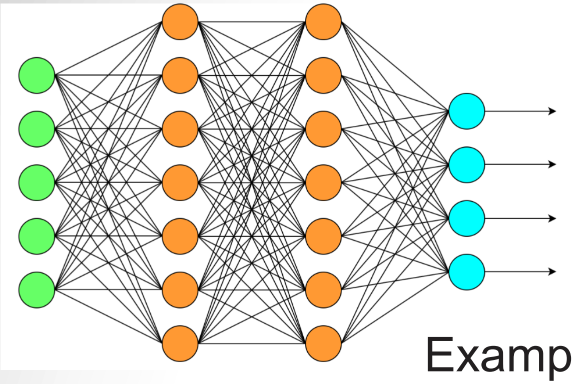
(Brief, Difficult) $\Rightarrow$

MISSION DEPENDENCY INDEX

MDI	Question 1 INTERRUPTABILITY How fast would the mission be impacted if the asset's operations were interrupted?			
Question 2 REPLICABILITY How difficult would it be to relocate the asset's mission capabilities?	IMMEDIATE < 15 minutes	BRIEF < 24 hours	SHORT < 7days	PROLONGED > 7 days
IMPOSSIBLE	100	88	76	64
EXTREMELY DIFFICULT	92	80	68	56
DIFFICULT	84	72	60	48
POSSIBLE	76	64	52	40

MODEL STRUCTURE

ERDC



Example:

Interruptability			
Prolonged	Short	Brief	Immediate
p_0	p_1	p_2	p_3
0.03	0.12	**0.56**	0.29

Replicability			
Possible	Difficult	Extremely Difficult	Impossible
q_0	q_1	q_2	q_3
0.22	**0.42**	0.23	0.13

(Brief, Difficult) ⇒

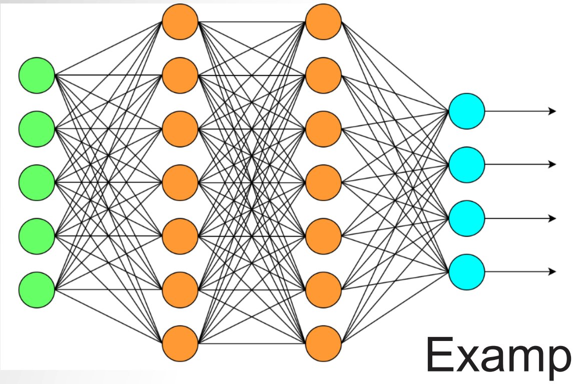
MISSION DEPENDENCY INDEX

MDI	Question 1 INTERRUPTABILITY How fast would the mission be impacted if the asset's operations were interrupted?			
Question 2 REPLICABILITY How difficult would it be to relocate the asset's mission capabilities?	IMMEDIATE < 15 minutes	BRIEF < 24 hours	SHORT < 7days	PROLONGED > 7 days
IMPOSSIBLE	100	88	76	64
EXTREMELY DIFFICULT	92	80	68	56
DIFFICULT	84	72	60	48
POSSIBLE	76	64	52	40

⇒ **72**

MODEL STRUCTURE

ERDC



Example:

Interruptability				Replicability			
Prolonged	Short	Brief	Immediate	Possible	Difficult	Extremely Difficult	Impossible
p_0	p_1	p_2	p_3	q_0	q_1	q_2	q_3
0.03	0.12	**0.56**	0.29	0.22	**0.42**	0.23	0.13

(Brief, Difficult) ⇒

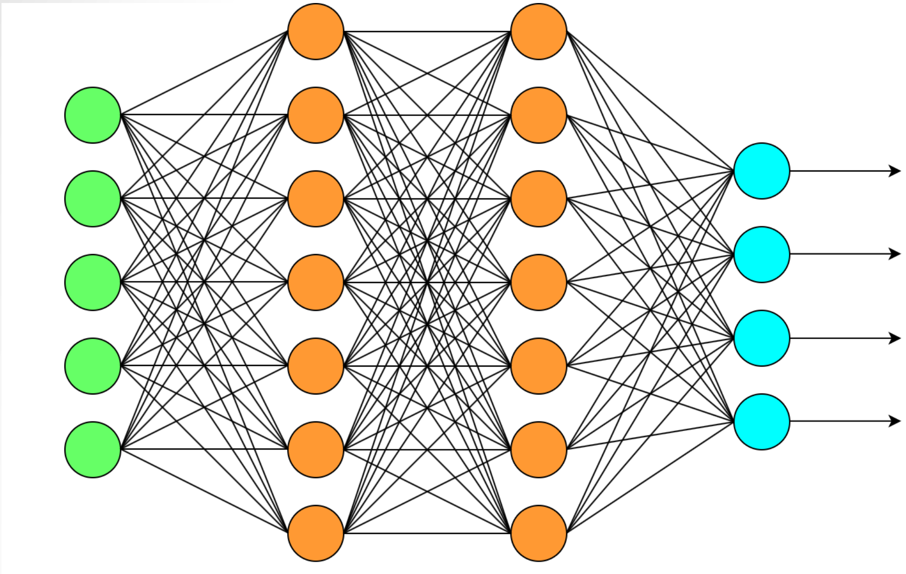
MISSION DEPENDENCY INDEX

MDI		Question 1 INTERRUPTABILITY How fast would the mission be impacted if the asset's operations were interrupted?			
		IMMEDIATE < 15 minutes	BRIEF < 24 hours	SHORT < 7days	PROLONGED > 7 days
Question 2 REPLICABILITY How difficult would it be to relocate the asset's mission capabilities?	IMPOSSIBLE	100	88	76	64
	EXTREMELY DIFFICULT	92	80	68	56
	DIFFICULT	84	72	60	48
	POSSIBLE	76	64	52	40

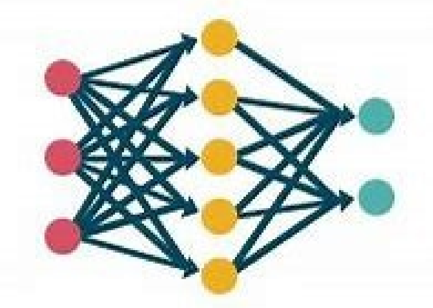
⇒ **72**

Unless (Prolonged, Possible)…

MODEL STRUCTURE



IF (Prolonged, Possible)…



MDI $\in [0,40]$

ELSE…

MISSION DEPENDENCY INDEX

MDI		Question 1 INTERRUPTABILITY How fast would the mission be impacted if the asset's operations were interrupted?			
		IMMEDIATE < 15 minutes	BRIEF < 24 hours	SHORT < 7days	PROLONGED > 7 days
Question 2 REPLICABILITY How difficult would it be to relocate the asset's mission capabilities?	IMPOSSIBLE	100	88	76	64
	EXTREMELY DIFFICULT	92	80	68	56
	DIFFICULT	84	72	60	48
	POSSIBLE	76	64	52	40

THE RESULTS

The Neural Network Performance

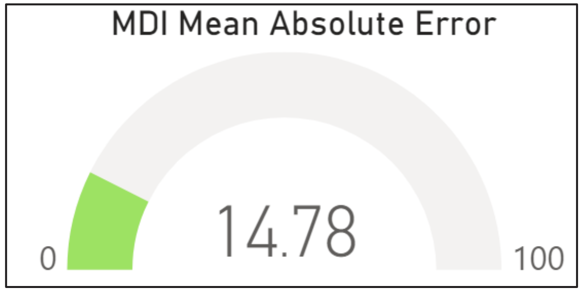
U.S. ARMY

US Army Corps of Engineers®

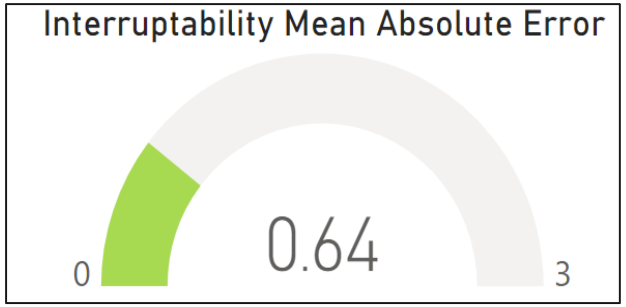
ERDC

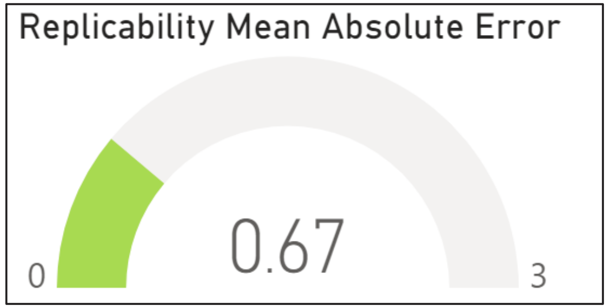
PERFORMANCE AT A GLANCE

ERDC



(Median: 8)





Class	Response
0	Prolonged
1	Short
2	Brief
3	Immediate

Predicted

Truth

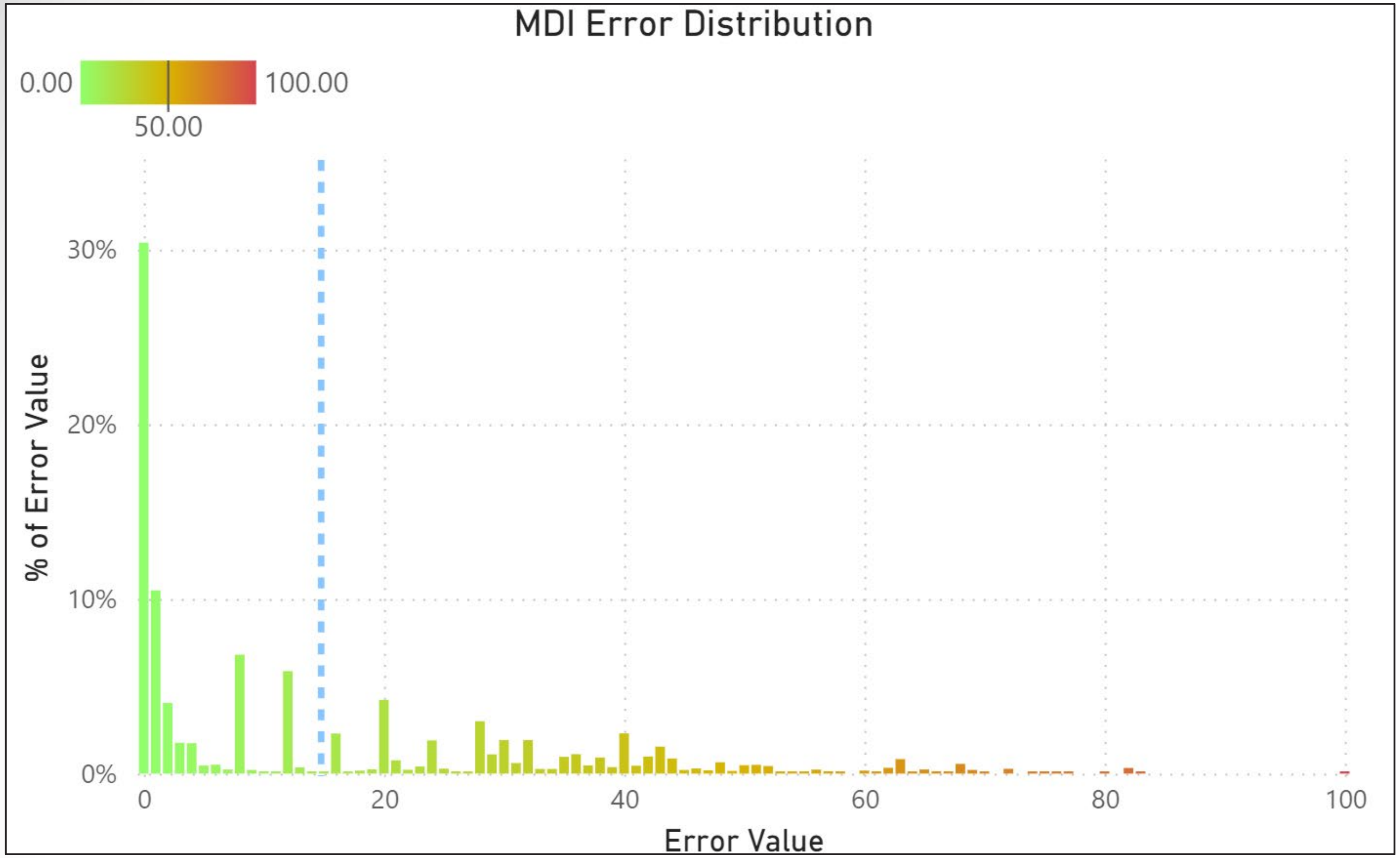
Interruptability	0	1	2	3	**Total**
0	3414	1010	454	453	**5331**
1	400	1374	692	282	**2748**
2	222	603	905	320	**2050**
3	25	129	131	226	**511**
Total	**4061**	**3116**	**2182**	**1281**	**10640**

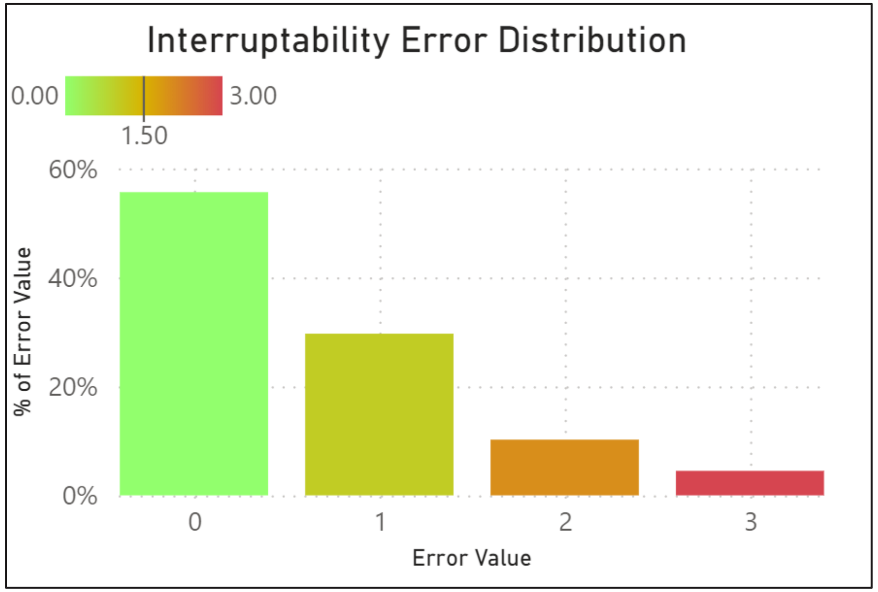
Replicability	0	1	2	3	**Total**
0	3267	1419	711	310	**5707**
1	400	1209	706	367	**2682**
2	115	507	850	302	**1774**
3	39	123	105	210	**477**
Total	**3821**	**3258**	**2372**	**1189**	**10640**

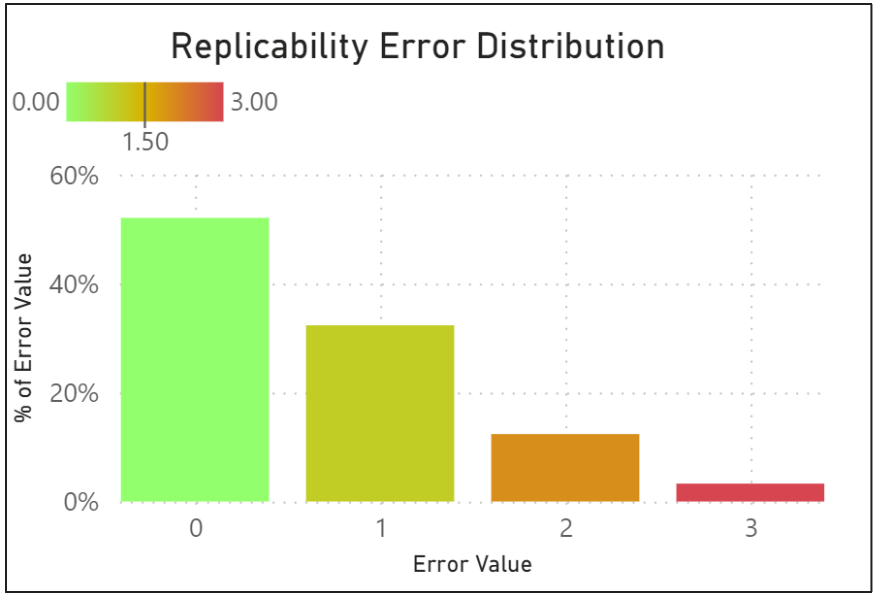
Class	Response
0	Possible
1	Difficult
2	Extreme Difficult
3	Impossible

PERFORMANCE AT A GLANCE

ERDC







CALCULATING CONFIDENCE

ERDC

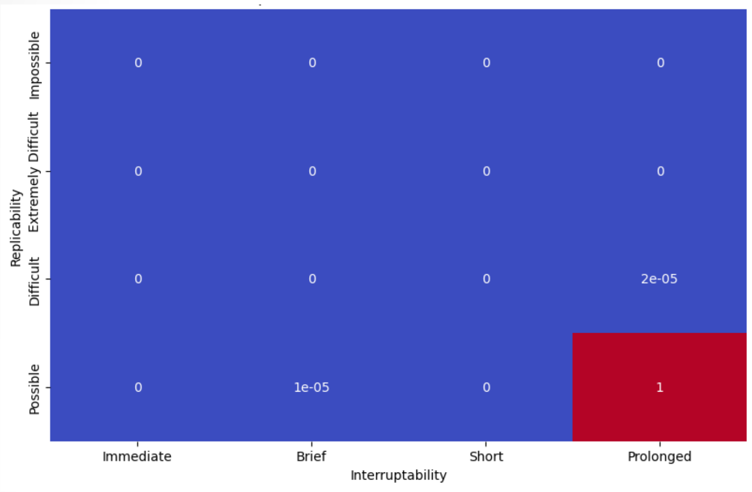
- The model outputs likelihoods
- Use these to get MDI likelihood
- Likelihood translates to confidence
- Confidence heatmap examples:

$$M = \begin{vmatrix} p_3q_3 & p_2q_3 & p_1q_3 & p_0q_3 \\ p_3q_2 & p_2q_2 & p_1q_2 & p_0q_2 \\ p_3q_1 & p_2q_1 & p_1q_1 & p_0q_1 \\ p_3q_0 & p_2q_0 & p_1q_0 & p_0q_0 \end{vmatrix}$$

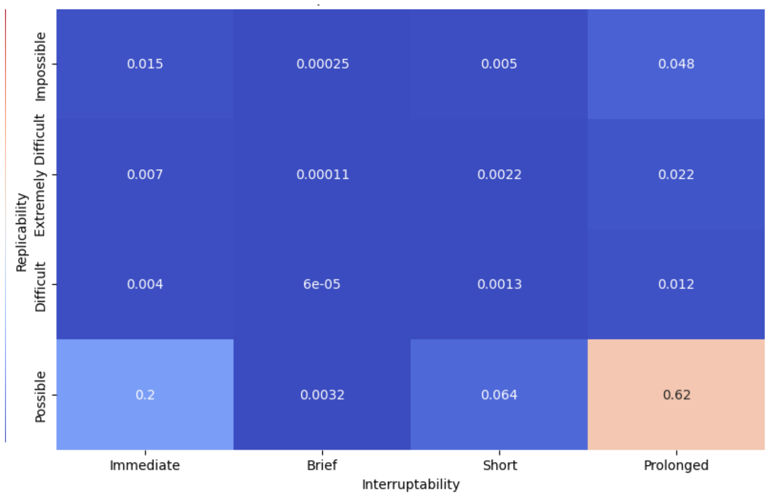
MISSION DEPENDENCY INDEX

MDI		Question 1 INTERRUPTABILITY How fast would the mission be impacted if the asset's operations were interrupted?			
		IMMEDIATE < 15 minutes	BRIEF < 24 hours	SHORT < 7days	PROLONGED > 7 days
Question 2 REPLICABILITY How difficult would it be to relocate the asset's mission capabilities?	IMPOSSIBLE	100	88	76	64
	EXTREMELY DIFFICULT	92	80	68	56
	DIFFICULT	84	72	60	48
	POSSIBLE	76	64	52	40

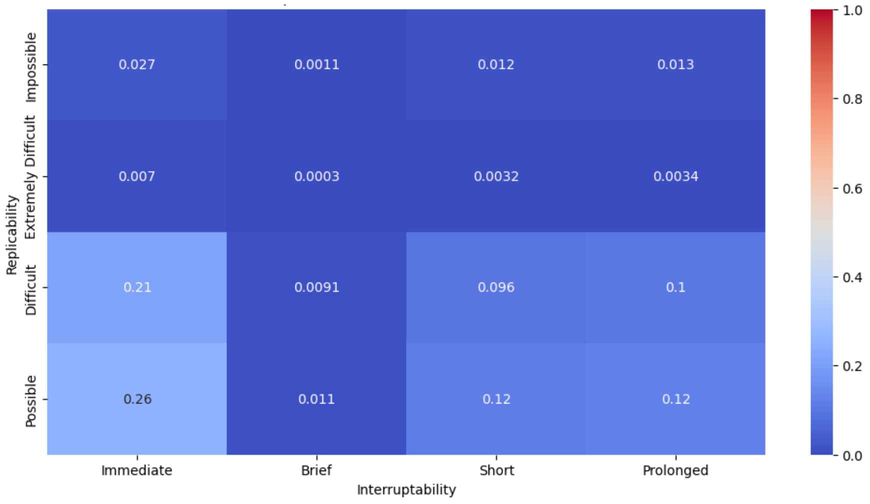
High



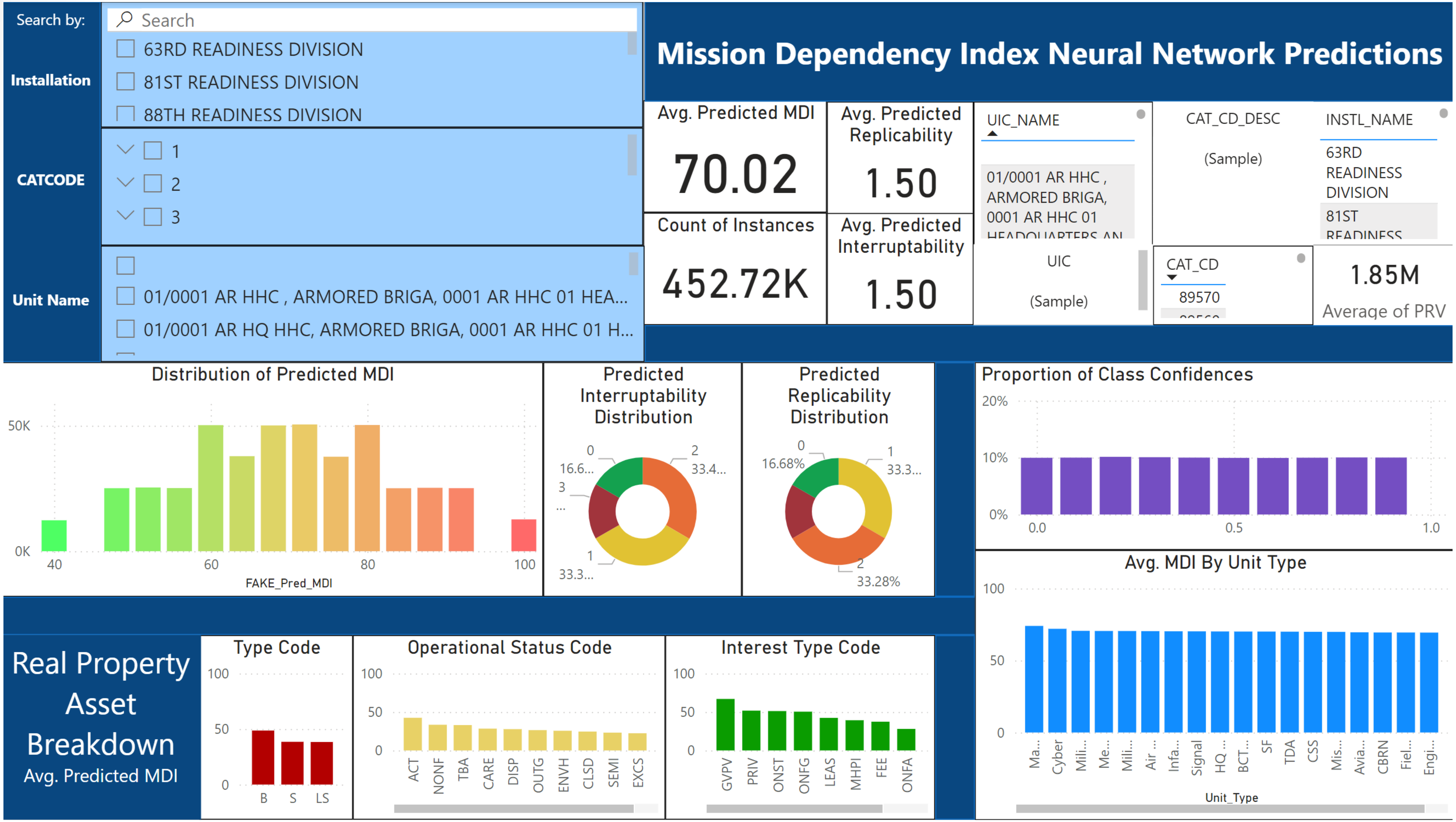
Moderate



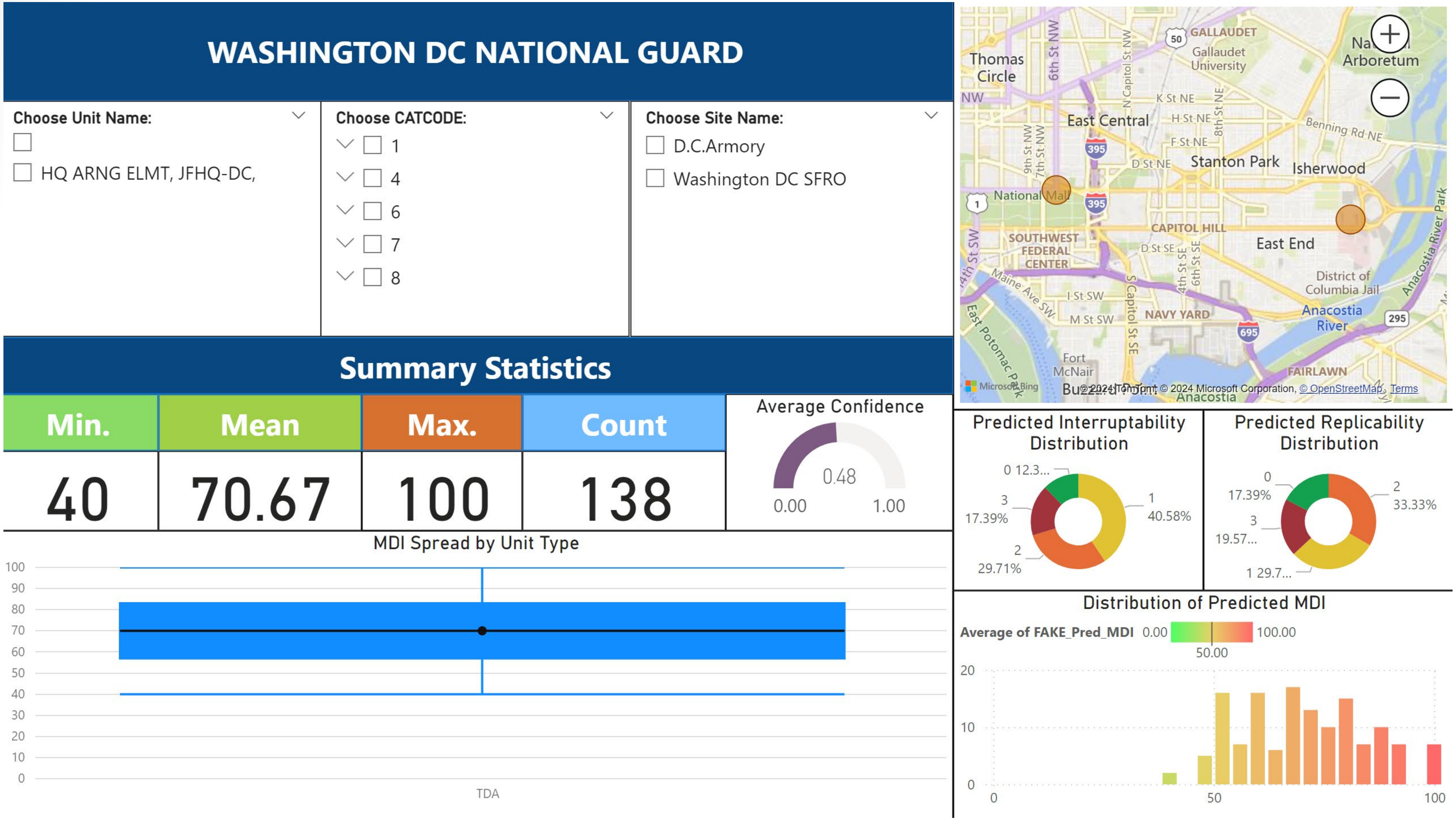
Low



POWER BI DASHBOARD



POWER BI DASHBOARD



THANK YOU!

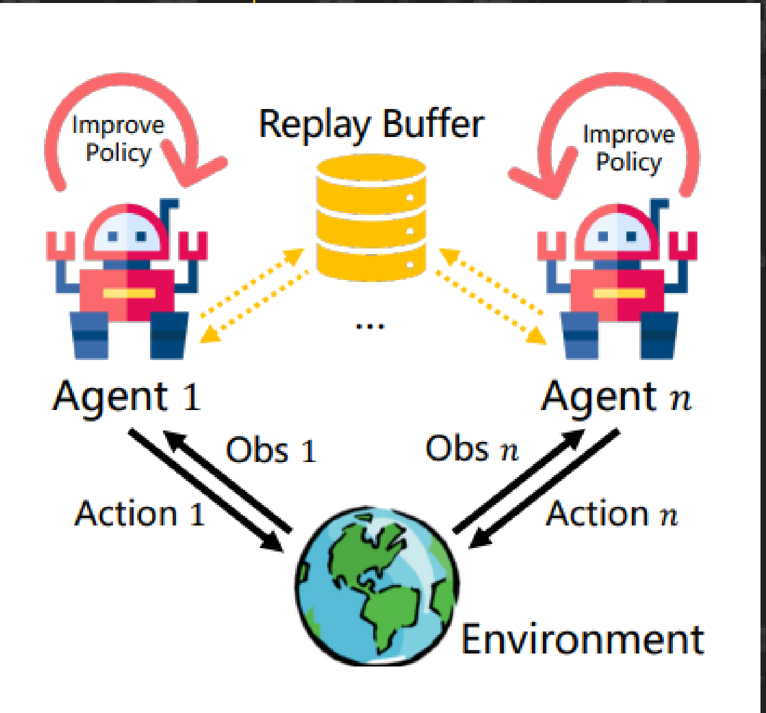
U.S. ARMY

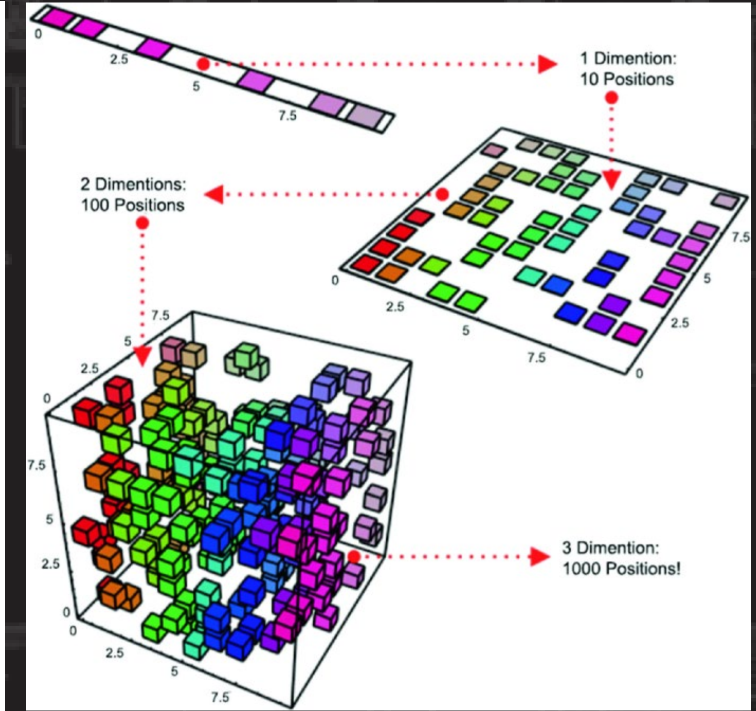
US Army Corps of Engineers®

ERDC

OPTIMIZING MAINTENANCE WITH SIMULATIONS

Joseph Wittrock
SMS-TCX





SUSTAINMENT MANAGEMENT TCX SYSTEM

U.S. ARMY

US Army Corps of Engineers®

ERDC
ENGINEER RESEARCH & DEVELOPMENT CENTER

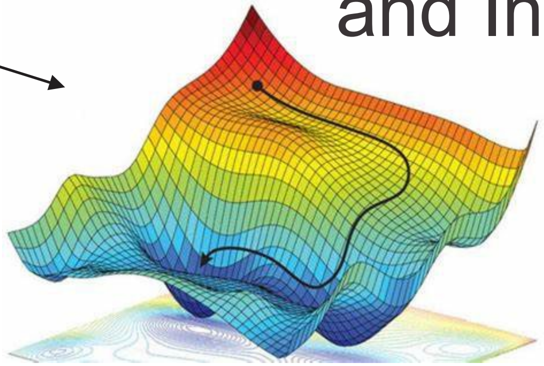
OVERVIEW

THE CYCLE OF OPTIMIZATION

Models and Simulations

Accumulated Data

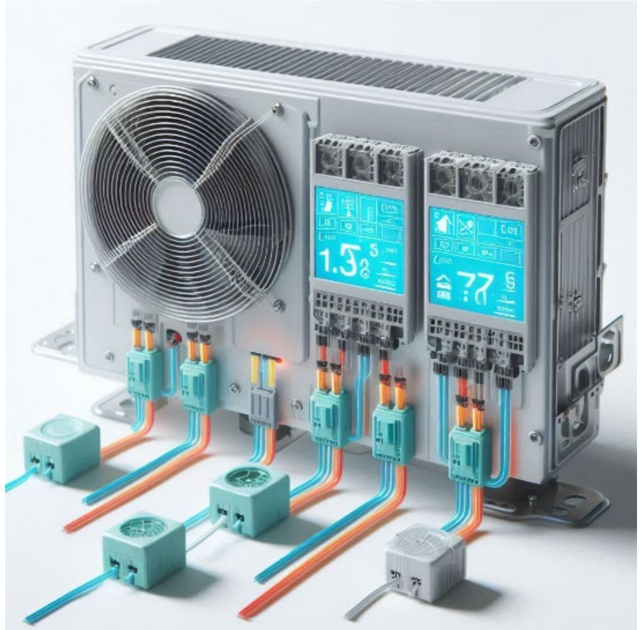
Optimization and Insights

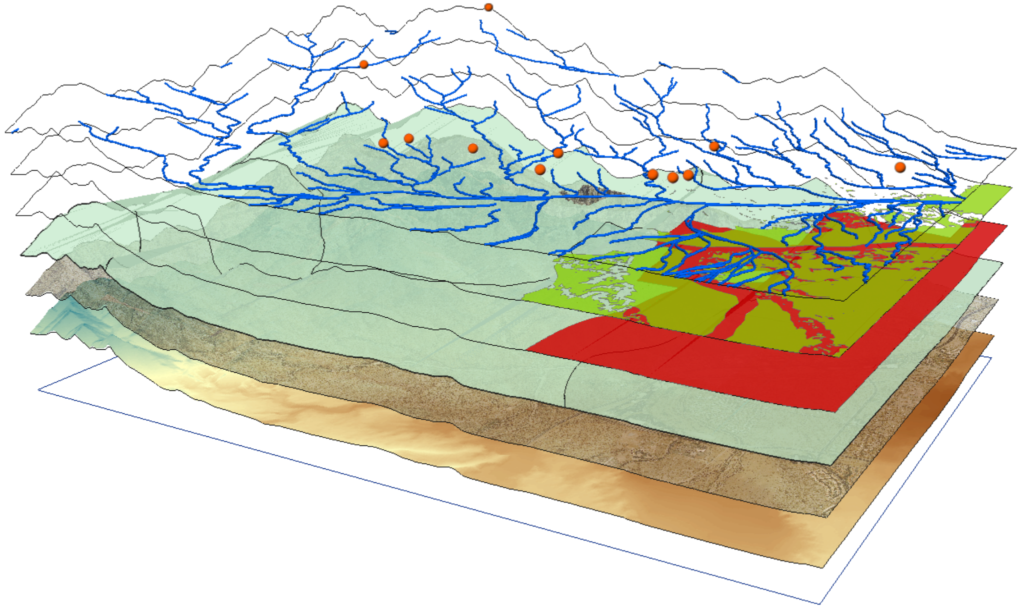


Real World Application

TYPES OF MAINTENANCE DATA

- Inspection Data
- Sensor Data
  - Thermal Sensors
  - Vibration Sensors
  - Occupancy Sensors
  - ...
- Geospatial
- Form Text
- Maintenance History





WAYS DATA IS USED TO IMPROVE SUSTAINMENT

ERDC

- Inspections of components help us **determine RUL**
- Sensors on components help detect and **predict Failures**
- Geographic data paired with component data help understanding of how different **climates effect** component degradation

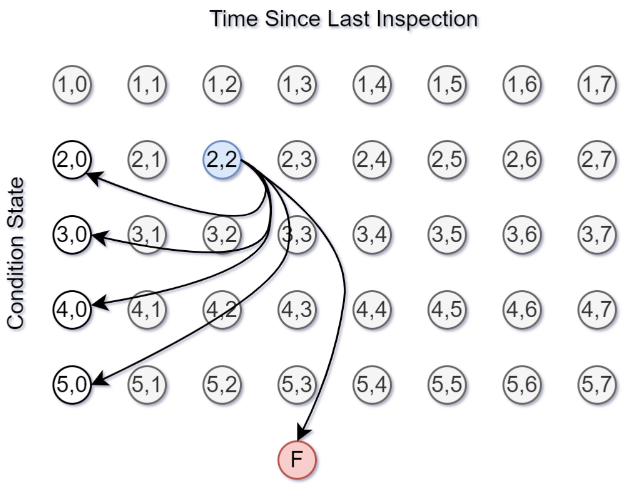
TYPES OF CONDITION MODELS

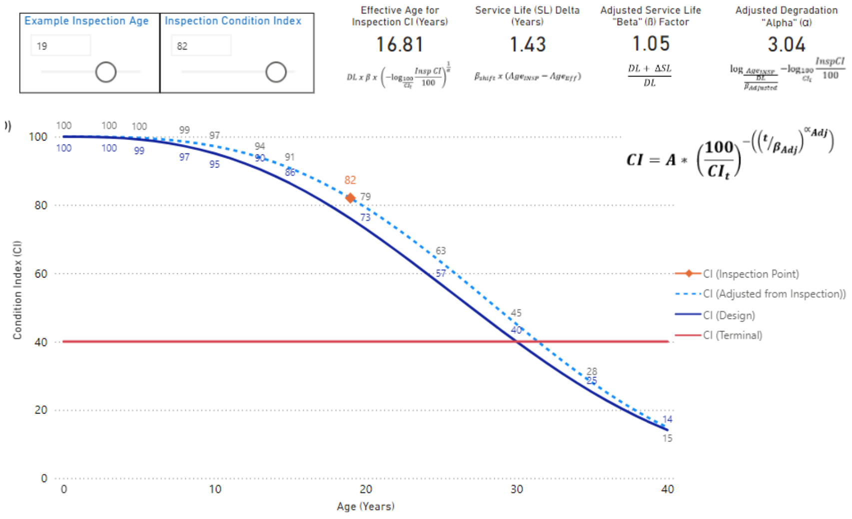
Discrete Models

- Discrete Timesteps
- Discrete Condition Steps
- Generally Fast Computations
- Probabilistic by nature

Continuous Models

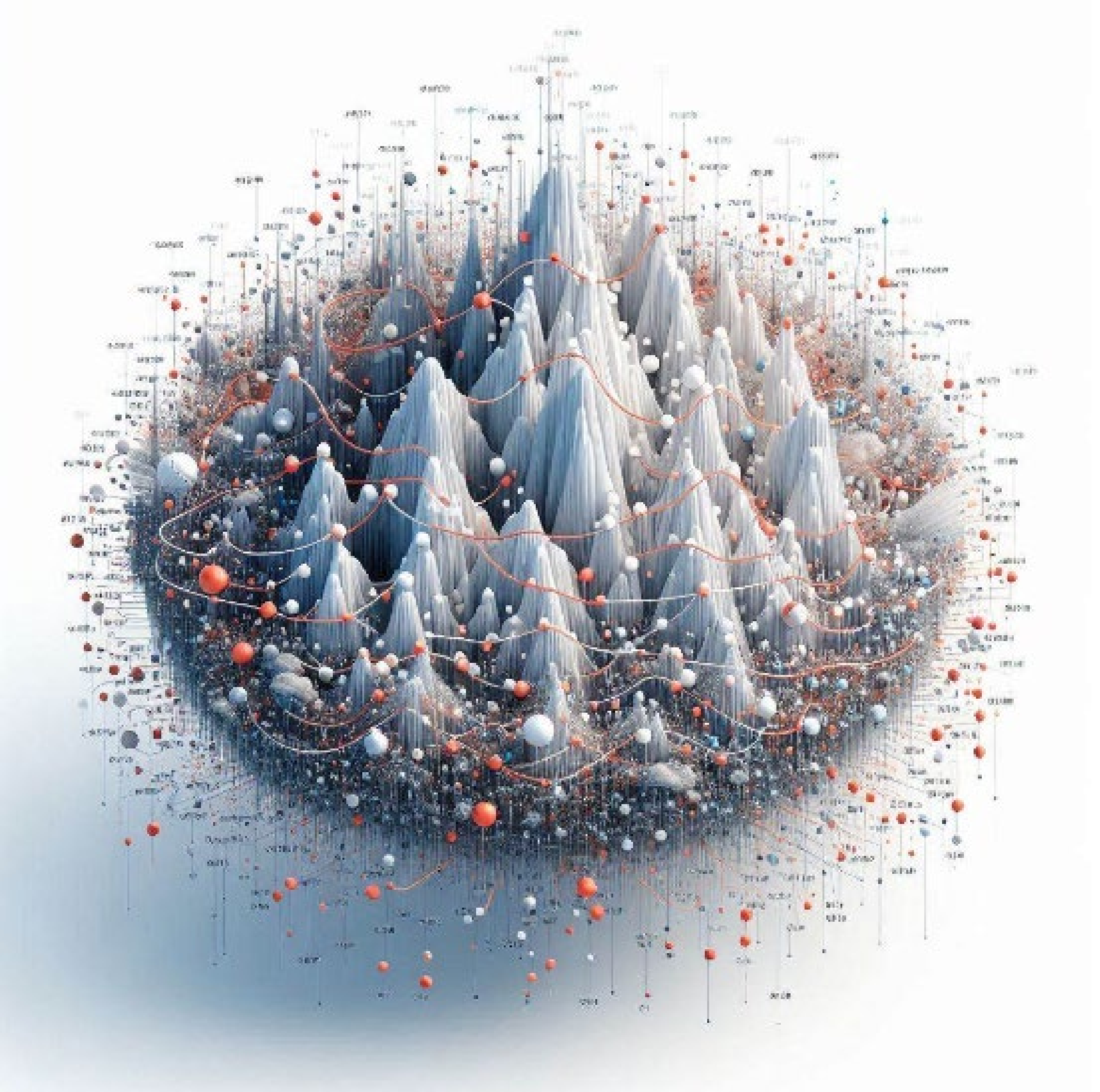
- Continuous Time
- Continuous Condition
- Slopes! Calculus!





SIMULATIONS

- Weibull Projections
  - "Looking Ahead" with age based models
- Monte Carlo
  - Run stochastic simulations and generate data



MODEL GENERATION

US Army Corps of Engineers®

U.S. ARMY

DISCRETE MODELS

Class	Upper	Lower
1	100	99
2	99	92
3	92	86
4	86	76
5	76	65
6	65	55
7	55	40
8	40	10
9	10	0

Condition State

1

2

3

4

5

F

DISCRETE MODELS FROM INSPECTION DATA

ERDC

Inspection Data Format:

- **Last Inspected Condition**
- **New Inspected Condition**
- **Time Between Inspections**

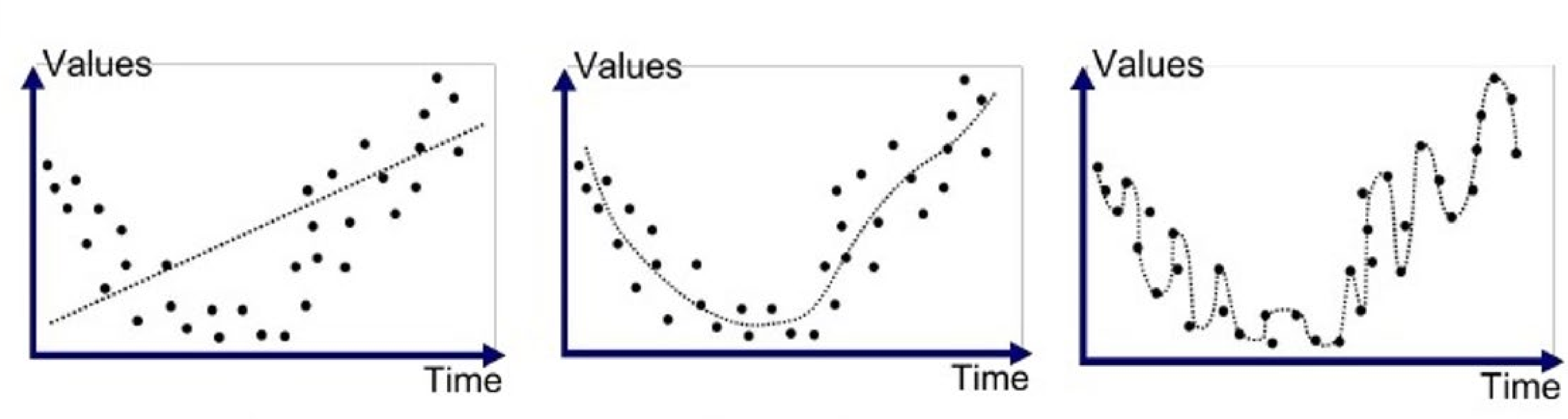
Inspection Data:

```
1 timesteps:
[[1030  803   89   22    0    5    0    3    0    5]
 [   0  473  160   26    0   17    0   26    0    5]
 [   0    0   83   22    0   14    0    7    0    2]
 [   0    0    0   25    0   15    0   10    0    2]
 [   0    0    0    0    0    0    0    0    0    0]
 [   0    0    0    0    0   12    0    5    0    1]
 [   0    0    0    0    0    0    0    0    0    0]
 [   0    0    0    0    0    0    0    8    0    3]
 [   0    0    0    0    0    0    0    0    0    0]
 [   0    0    0    0    0    0    0    0    0    3]]
```

Generated Model:

```
[[0.5263 0.4103 0.0455 0.0112 0.     0.0026 0.     0.0015 0.     0.0026]
 [0.     0.669  0.2263 0.0368 0.     0.024  0.     0.0368 0.     0.0071]
 [0.     0.     0.6484 0.1719 0.     0.1094 0.     0.0547 0.     0.0156]
 [0.     0.     0.     0.4808 0.     0.2885 0.     0.1923 0.     0.0385]
 [0.     0.     0.     0.     0.     0.     0.     0.     0.     0.    ]
 [0.     0.     0.     0.     0.     0.6667 0.     0.2778 0.     0.0556]
 [0.     0.     0.     0.     0.     0.     0.     0.     0.     0.    ]
 [0.     0.     0.     0.     0.     0.     0.     0.7273 0.     0.2727]
 [0.     0.     0.     0.     0.     0.     0.     0.     0.     0.    ]
 [0.     0.     0.     0.     0.     0.     0.     0.     0.     1.    ]]
```

HOW DO WE KNOW IF A MODEL WORKS?

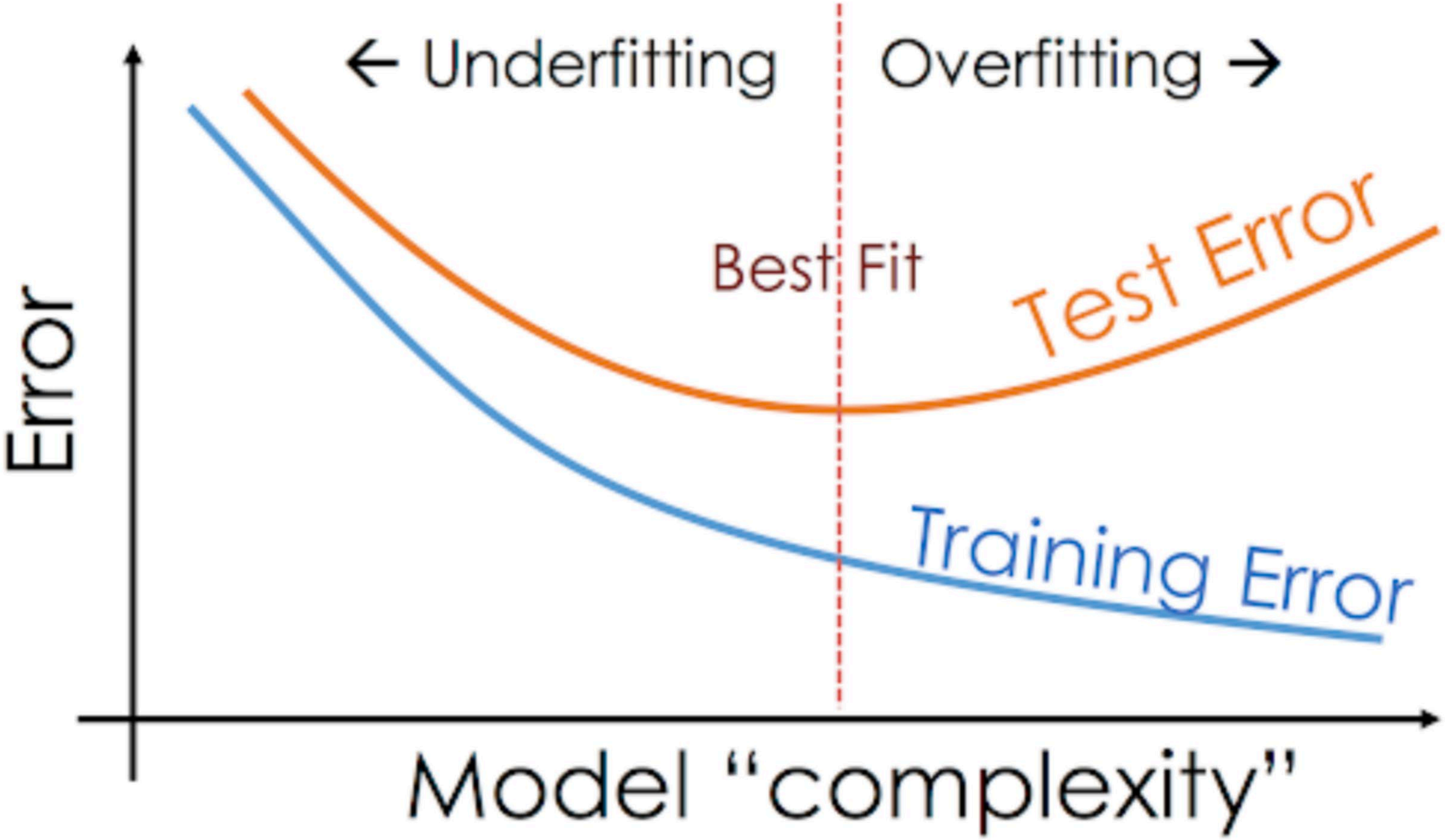


Underfitted

Good Fit/Robust

Overfitted

HOW DO WE KNOW IF A MODEL WORKS?



SIMULATIONS

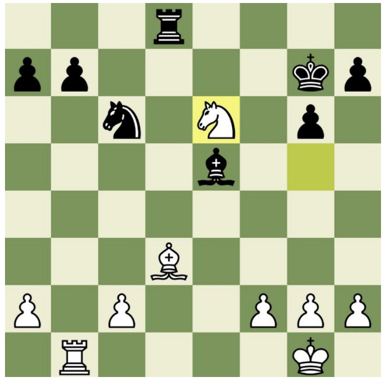
US Army Corps of Engineers®

U.S. ARMY

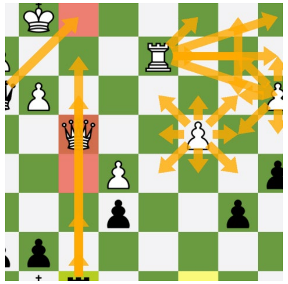
MARKOV DECISION PROCESS

- A standardized framework for sequential decision modeling
- States: A ‘Snapshot’ of the environment

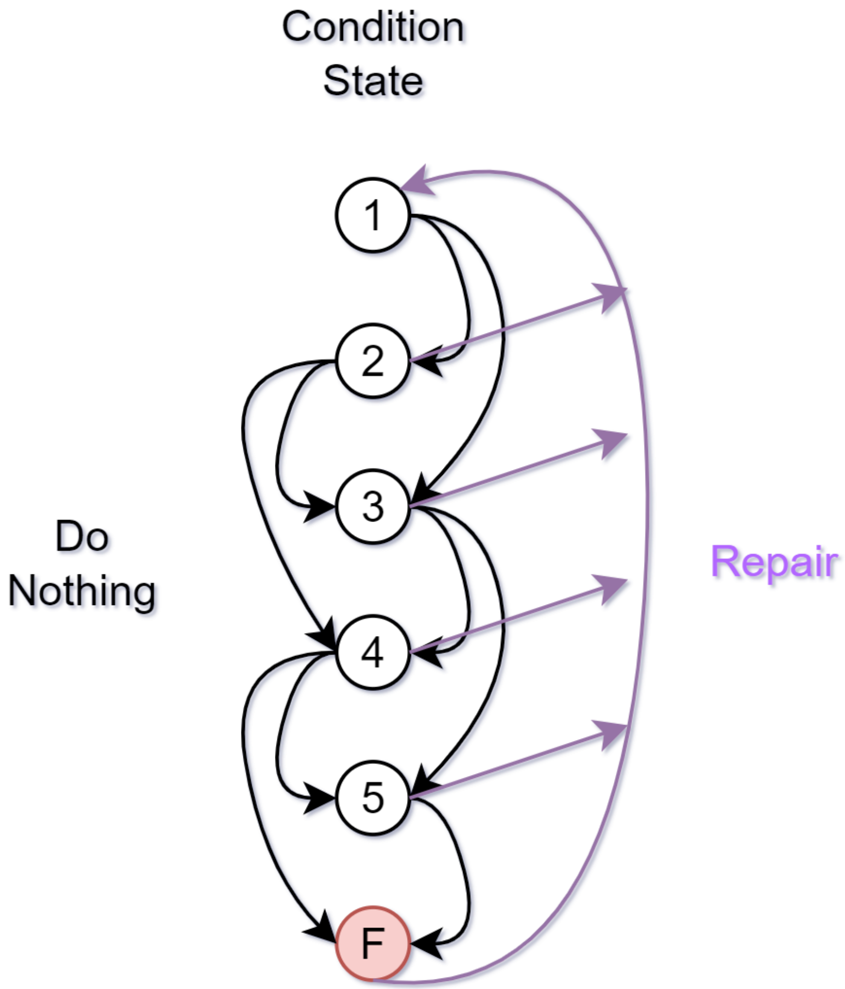




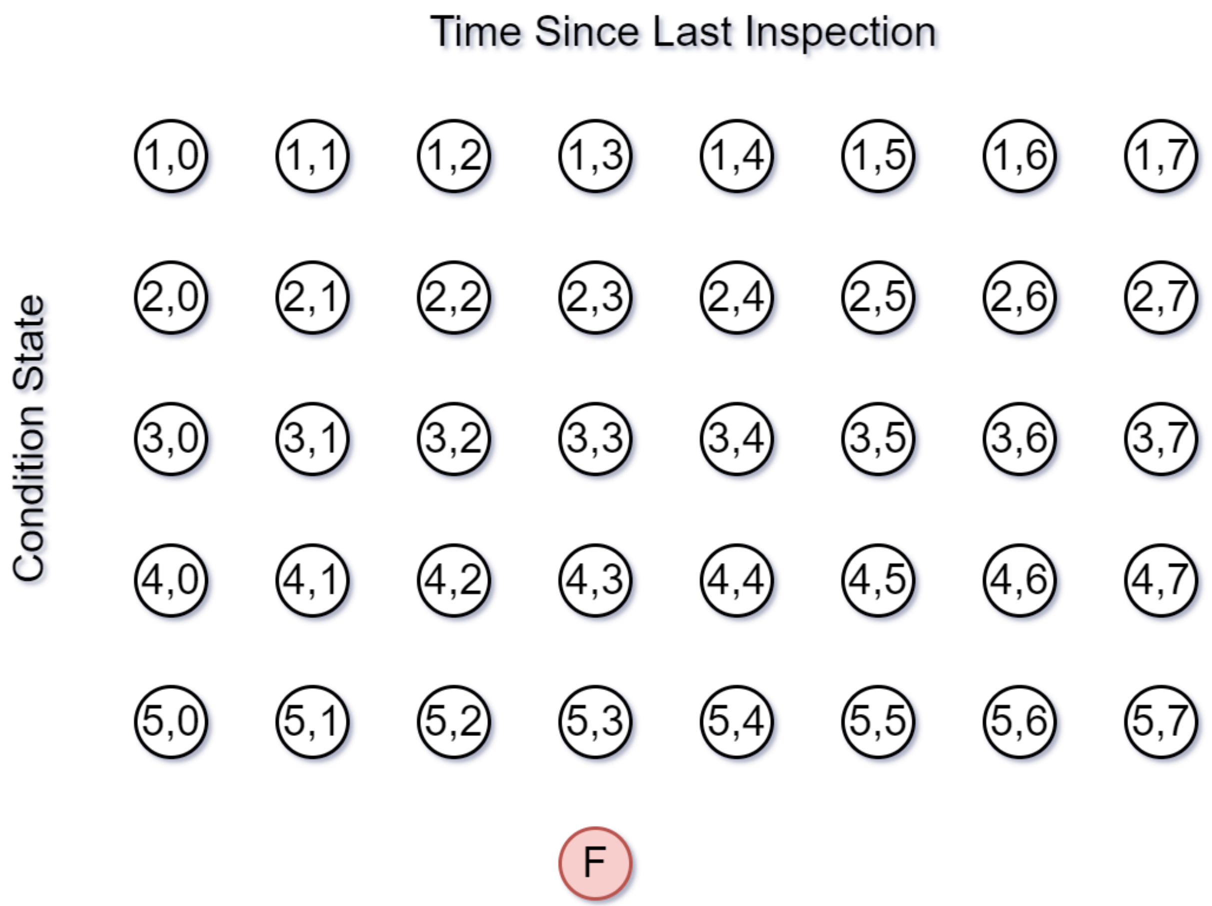
- Actions: Ways to interact with the environment given a state



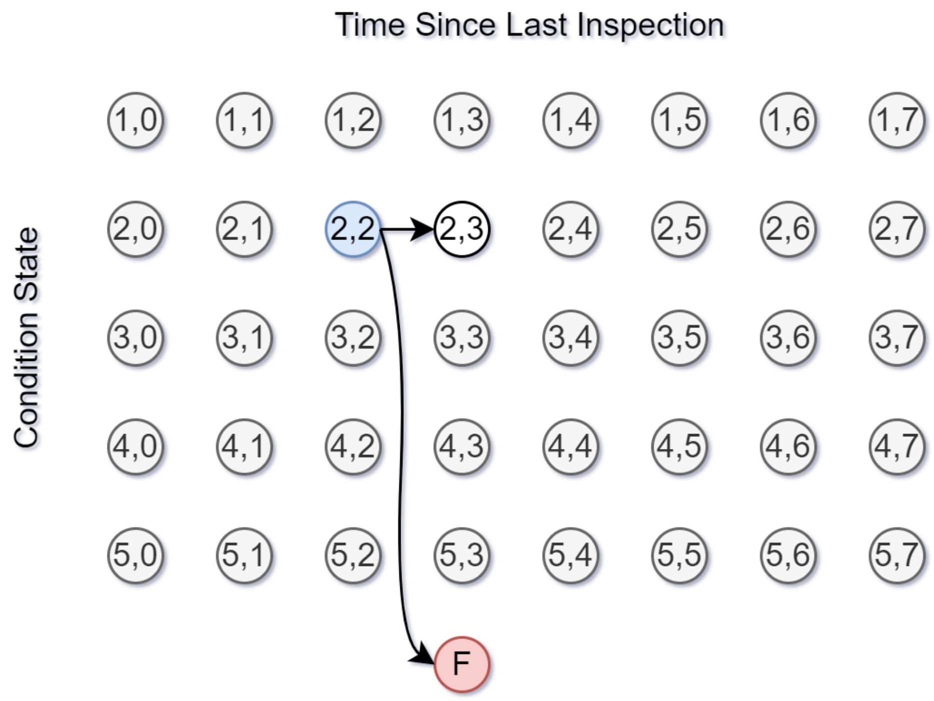

MAINTENANCE MDP EXAMPLE

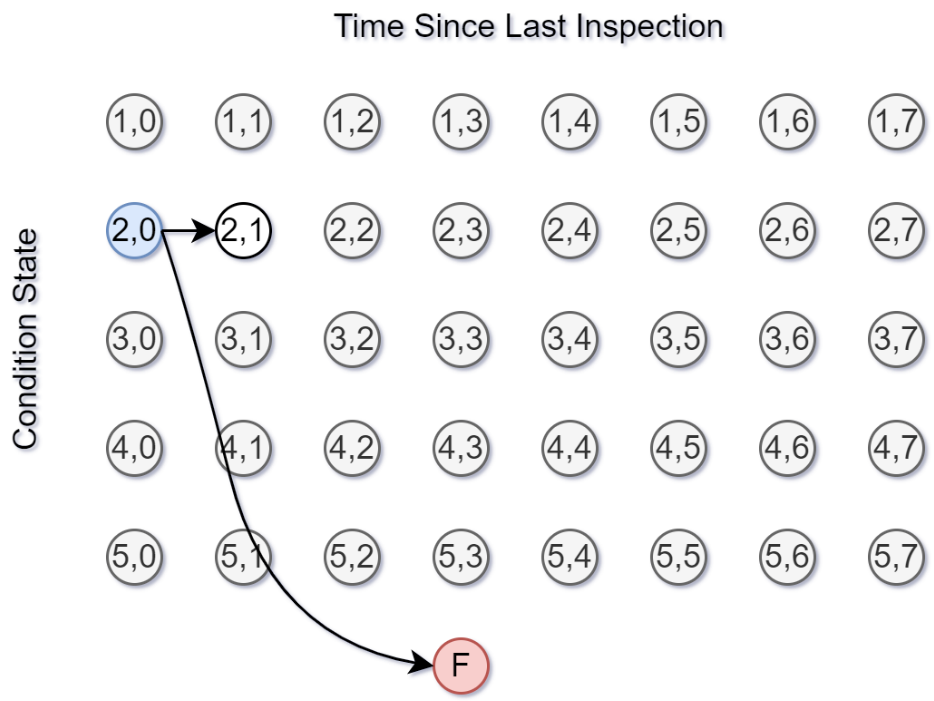


MDP WITH UNCERTAIN CONDITIONS

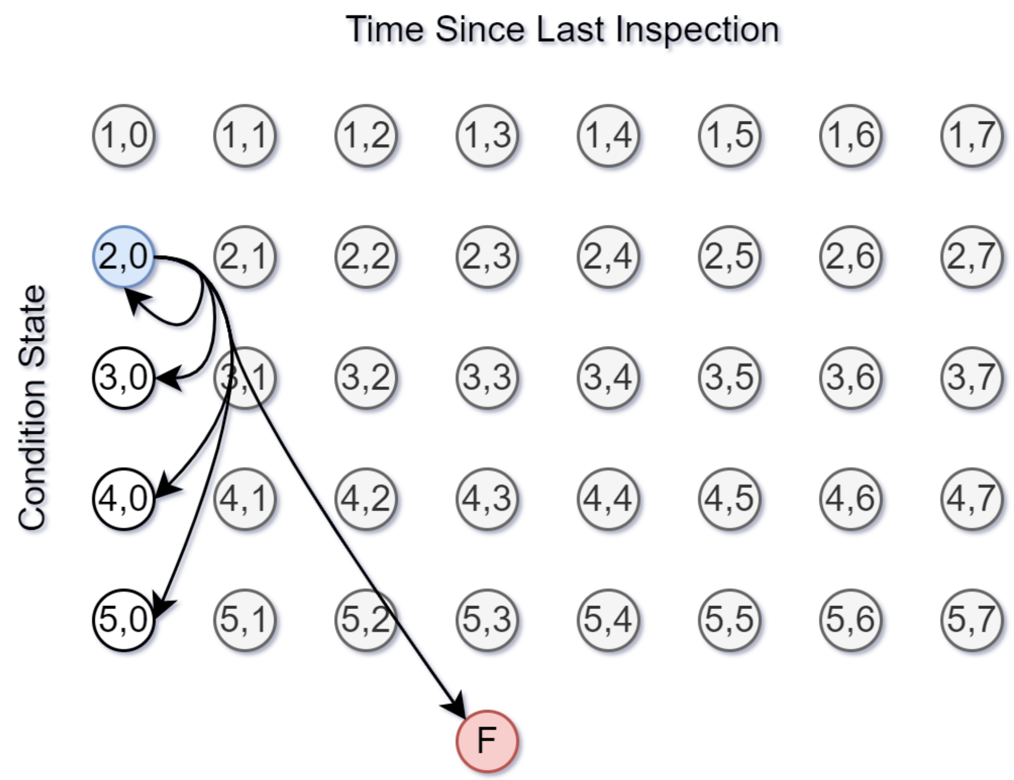


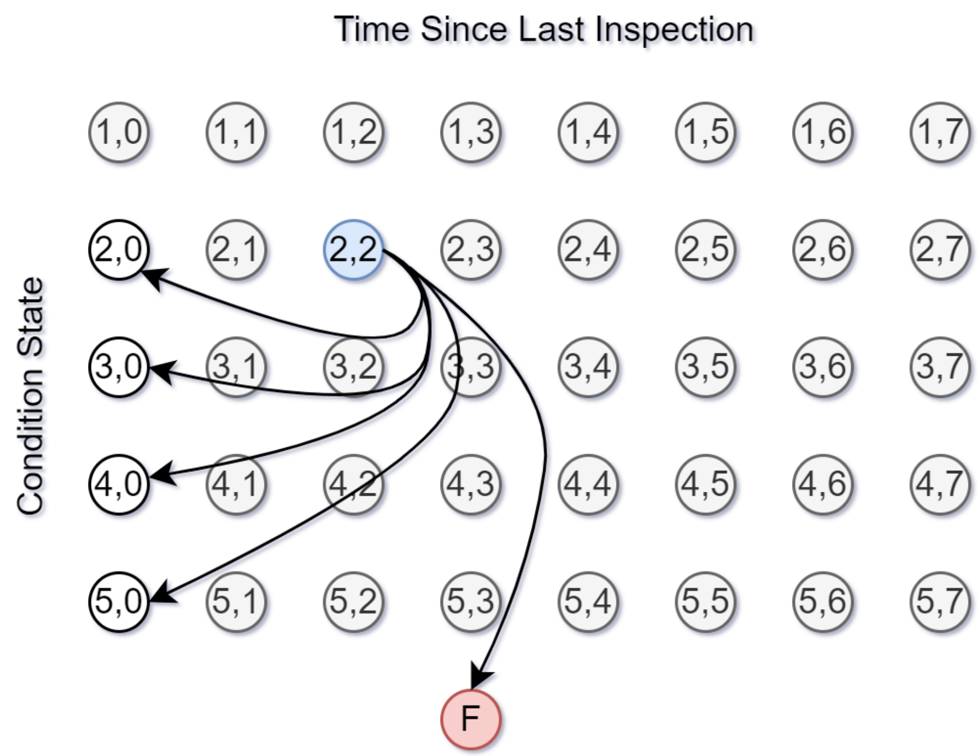
DO NOTHING ACTION





INSPECTION ACTION





POLICIES AND REWARD

Policies: Behavior – What actions are taken in each state.

$$\pi(s) \in A$$

Reward: A Measure of Goal

$$R_t$$

+ : 

- : 

Return: Sum of future reward

$$G_t = \Sigma_{i=t}^{T} R_i$$

VALUES

- Values describe the **Expected Return**, given a policy.

- **State Value** is the Expected Return if in a State

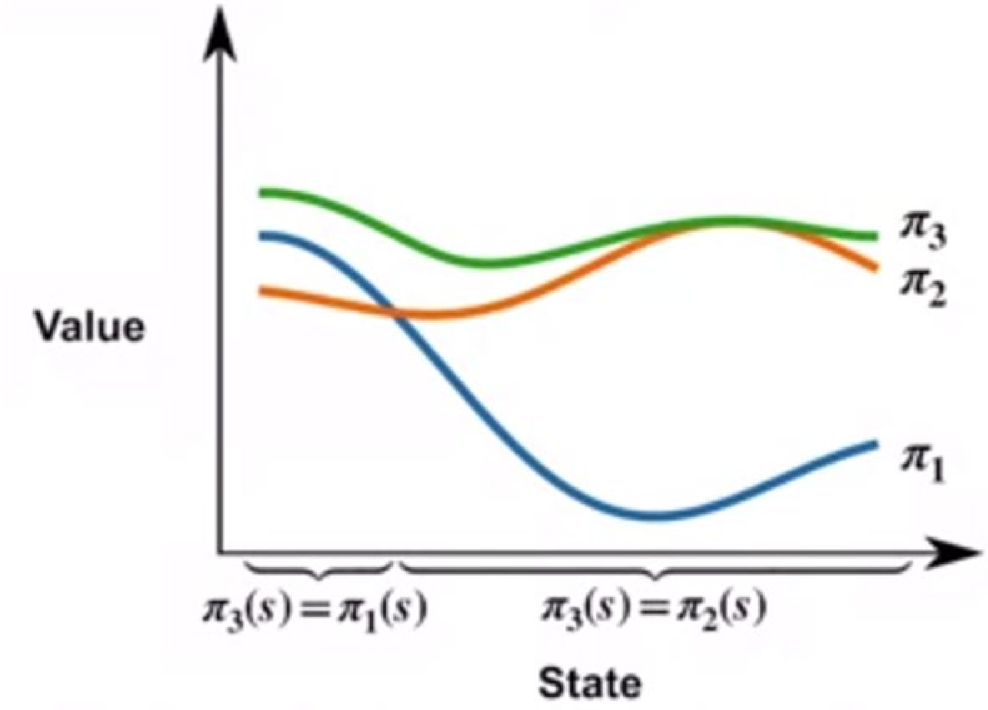
$$V_\pi(s) = E_\pi[G_t | S_t = s]$$

- **Action Value** is the Expected Return if an Action is taken in a State

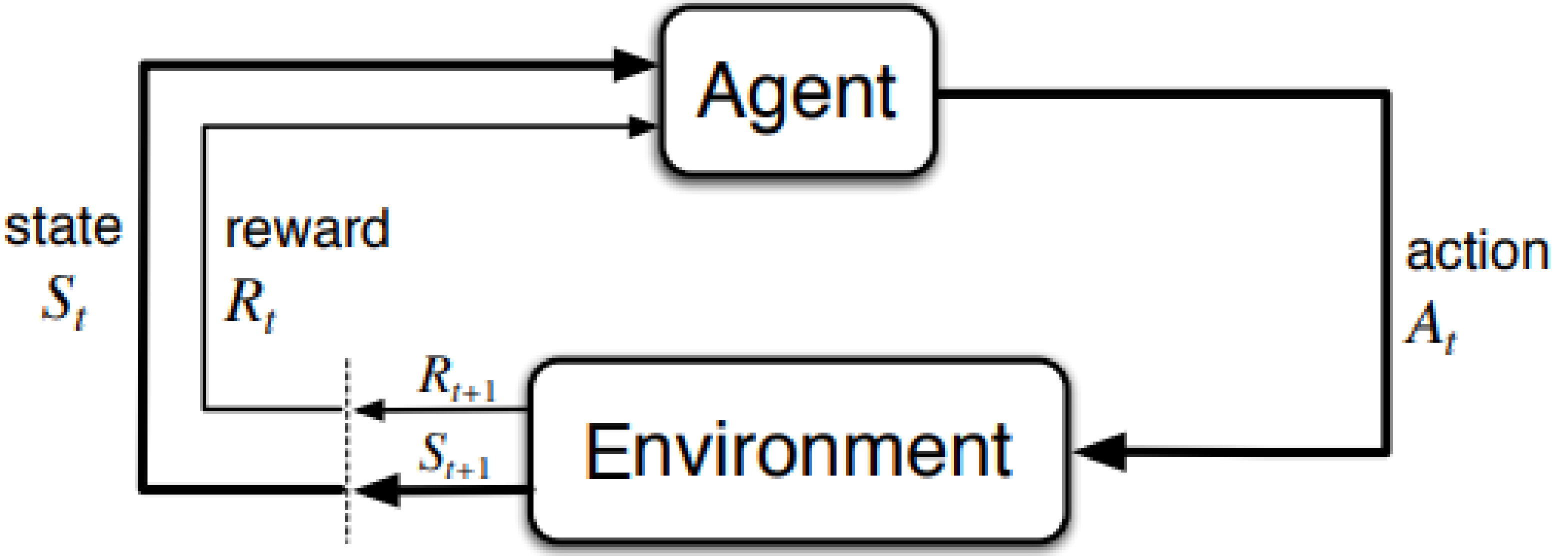
$$Q_\pi(s, a) = E_\pi[G_t | S_t = s, A_t = a]$$

OPTIMAL POLICY

- The Optimal Policy is a behavior which takes actions that maximize the expected return.
- The Optimal Policy gives the largest Value of a State or Behavior



REINFORCEMENT LEARNING

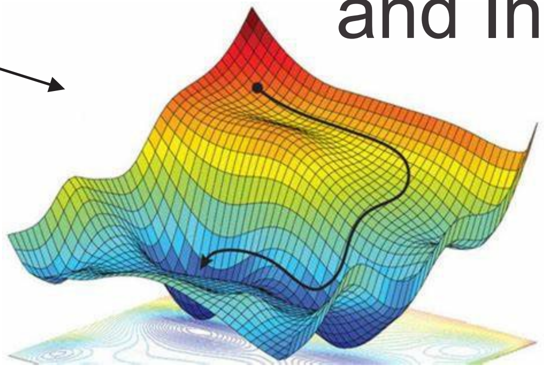


THE CYCLE OF OPTIMIZATION

Accumulated Data

Models and Simulations

Optimization and Insights



Real World Application

OPTIMIZATION

US Army Corps of Engineers®

U.S. ARMY

POLICY IMPROVEMENT

We want to find the actions that have a better value:

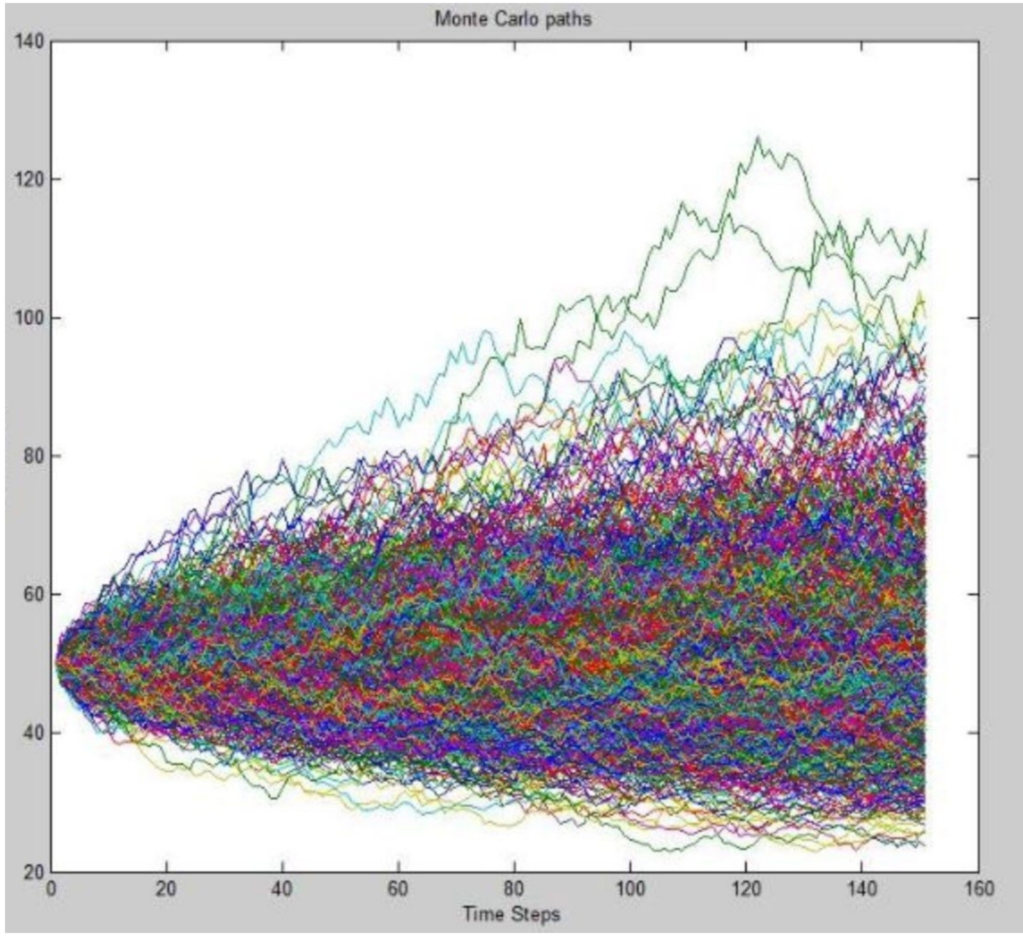
$$Q_\pi(s, a) > V_\pi(s)$$

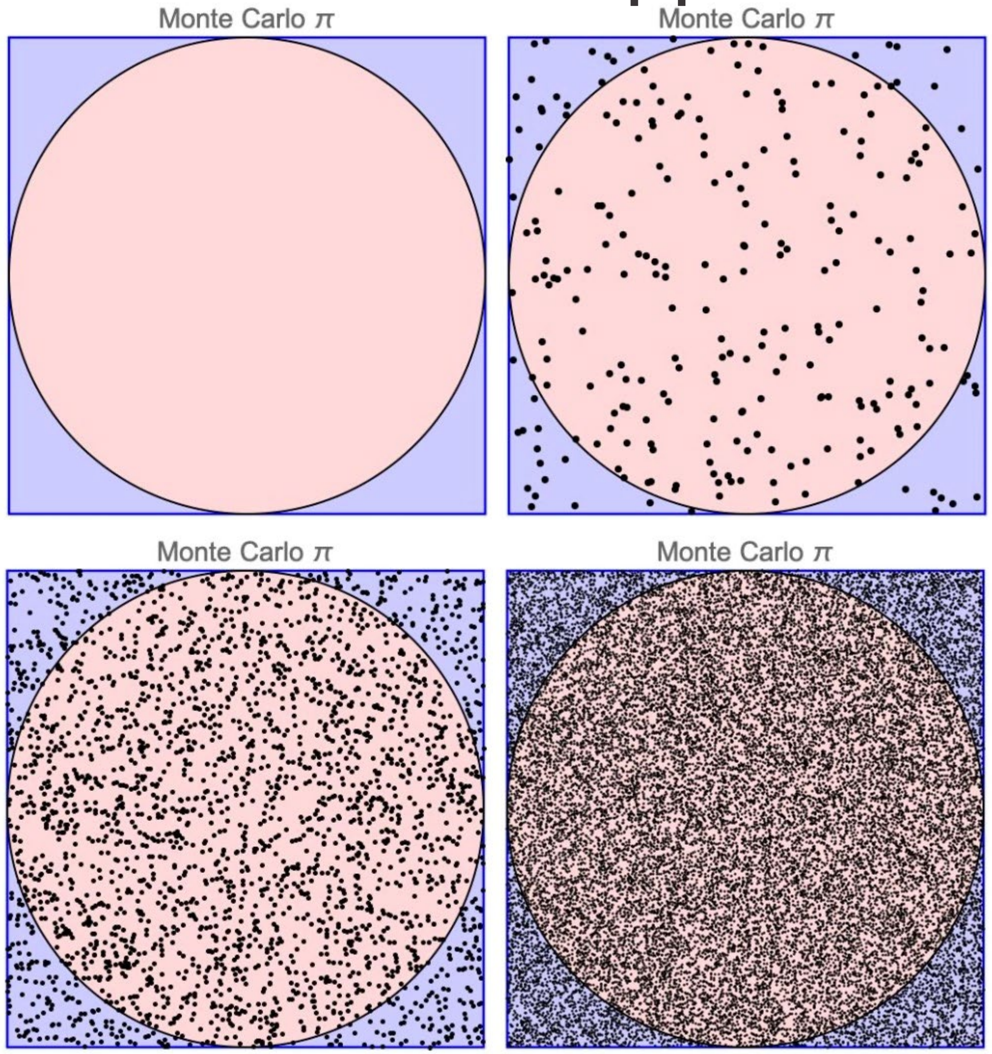
We improve our policy by taking actions with largest value

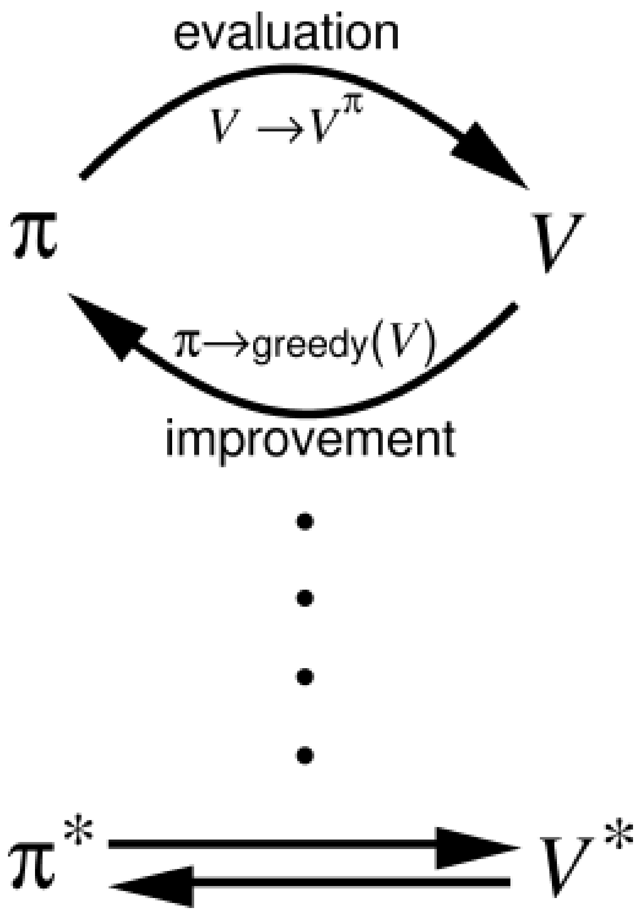
$$\pi'(s) := argmax[Q_\pi(s, a)]$$

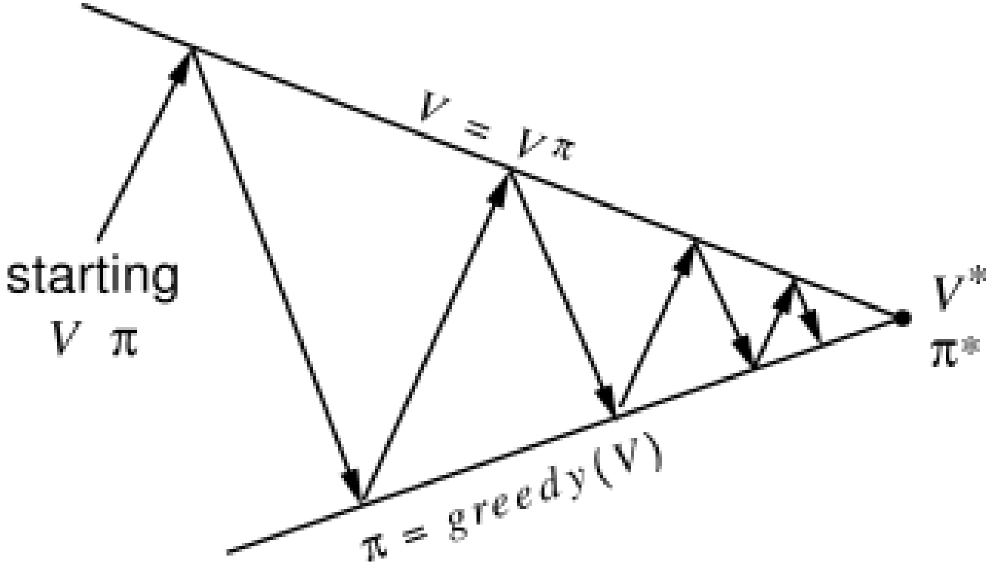
POLICY EVALUATION: MONTE CARLO METHODS

- **Monte Carlo**: Simulate it a bunch and see what happens

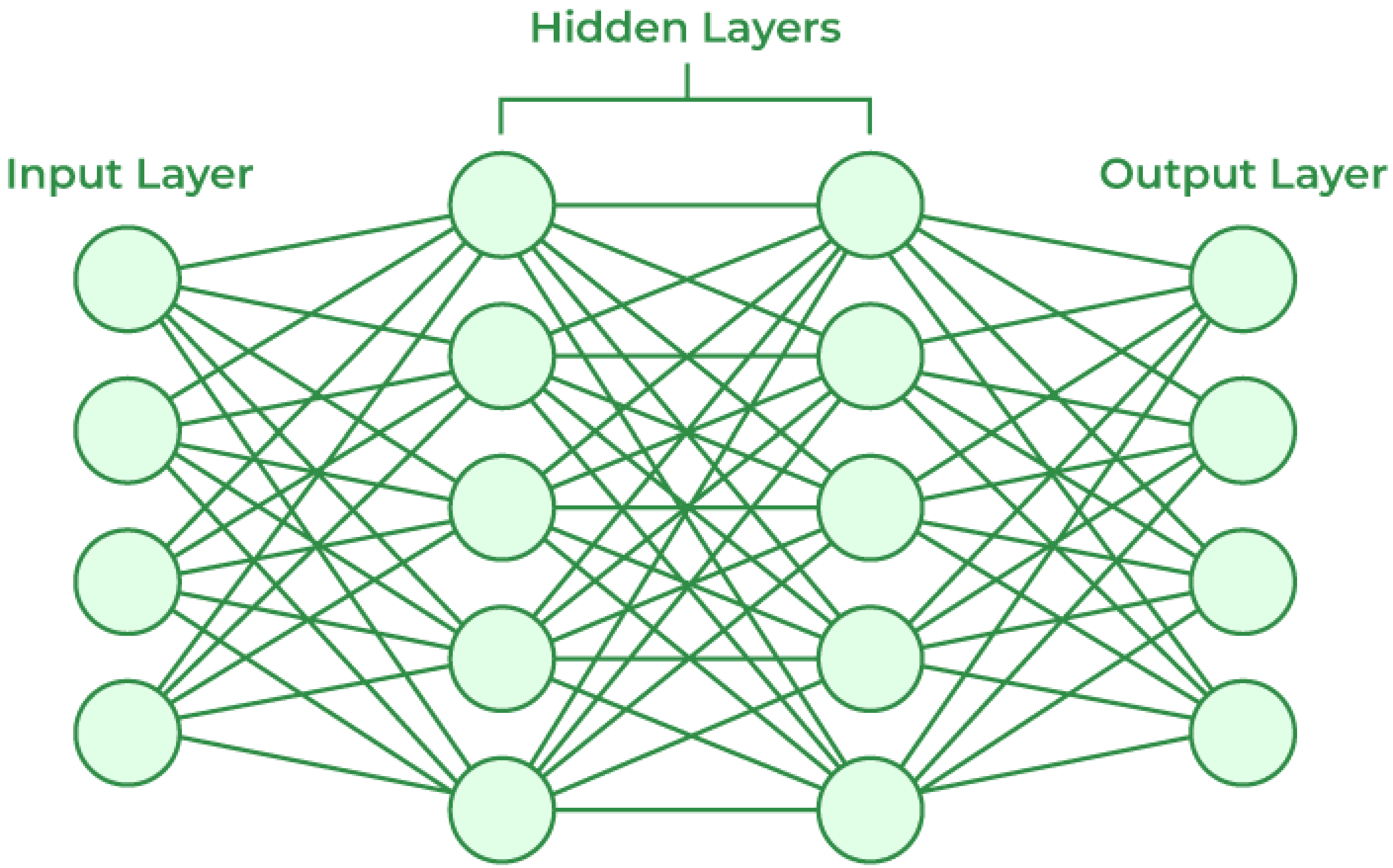





evaluation
$V \rightarrow V^{\pi}$
π
V
$\pi \rightarrow \text{greedy}(V)$
improvement
π^*
V^*

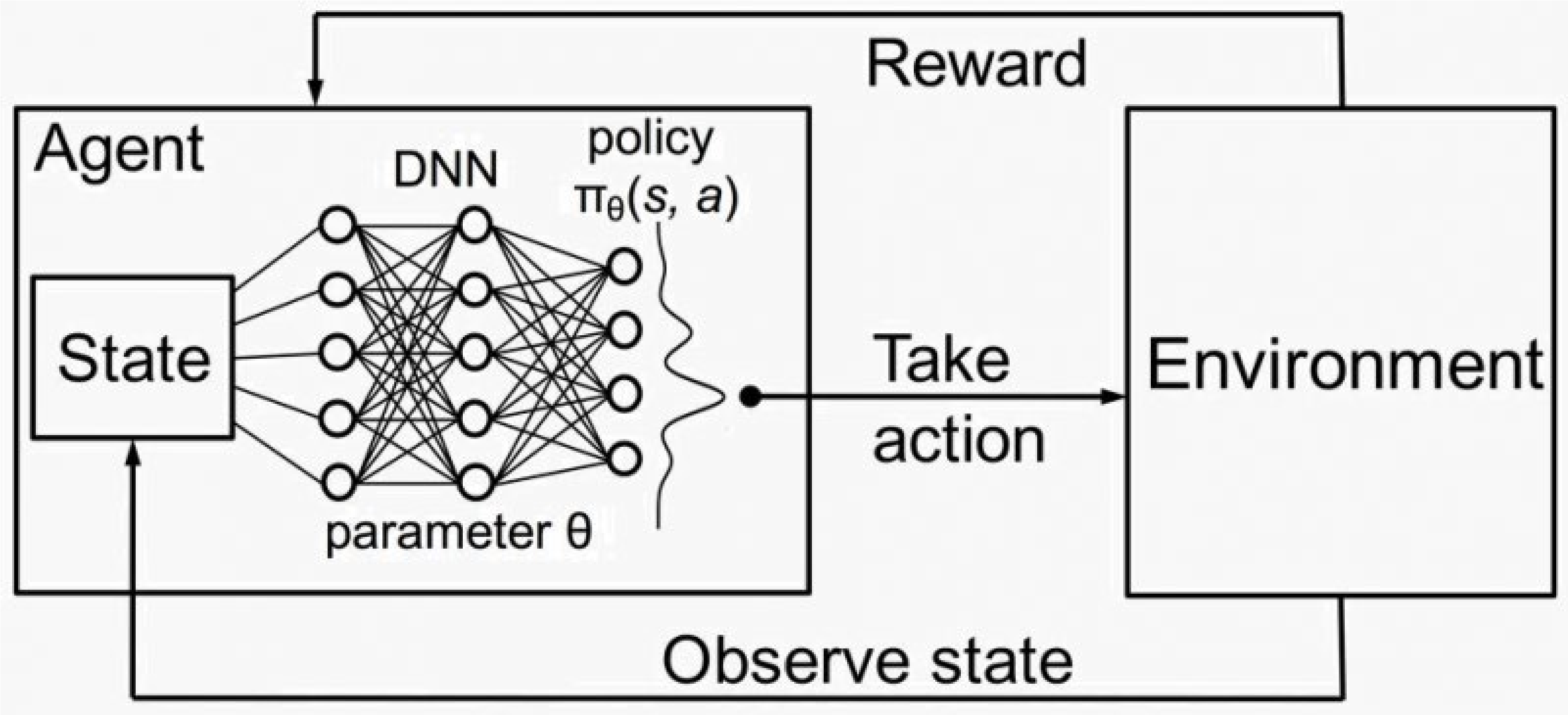

$V = V^{\pi}$
starting
V π
V^*
π^*
$\pi = \text{greedy}(V)$

NEURAL NETWORKS

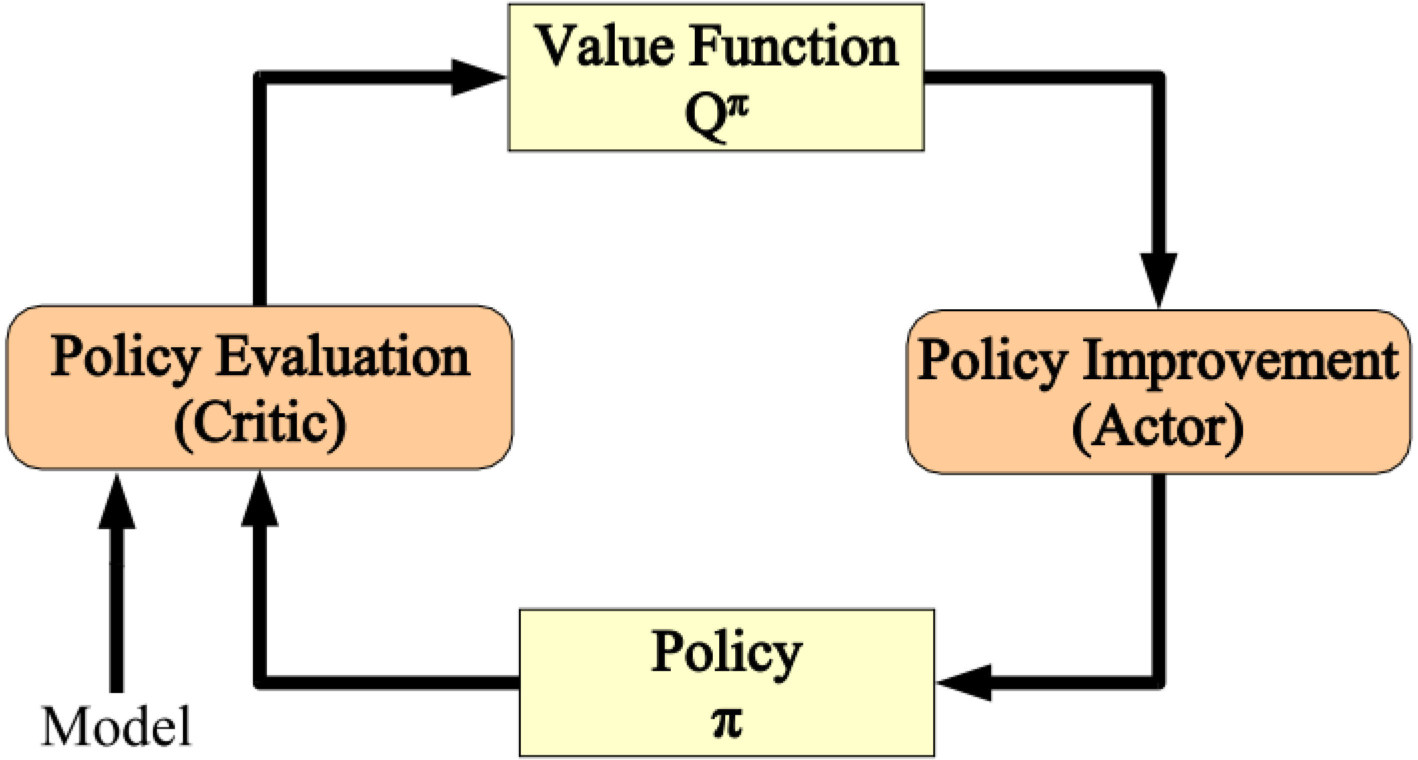


DEEP Q REINFORCEMENT LEARNING

- Neural Network used to approximate Q values.



ACTOR CRITIC MODELS



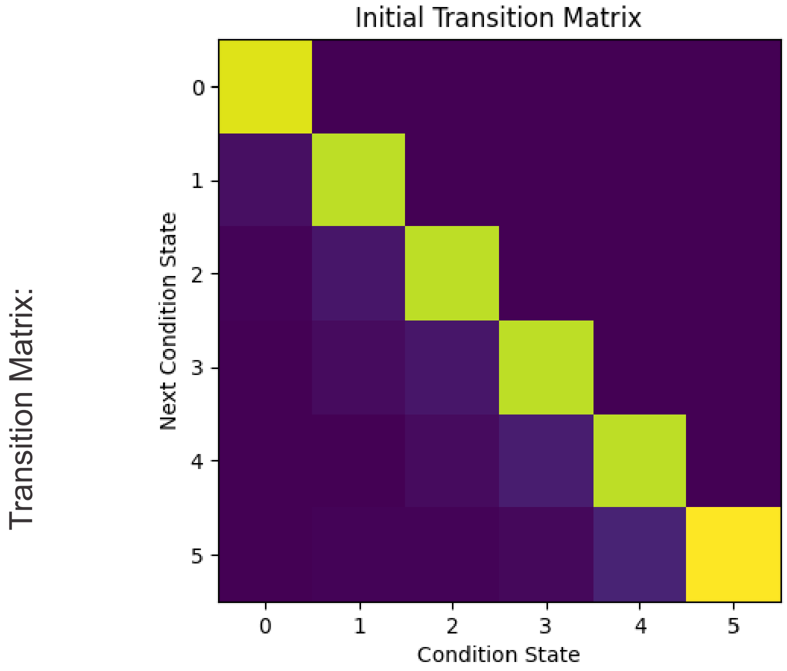
SINGLE COMPONENT EXAMPLE

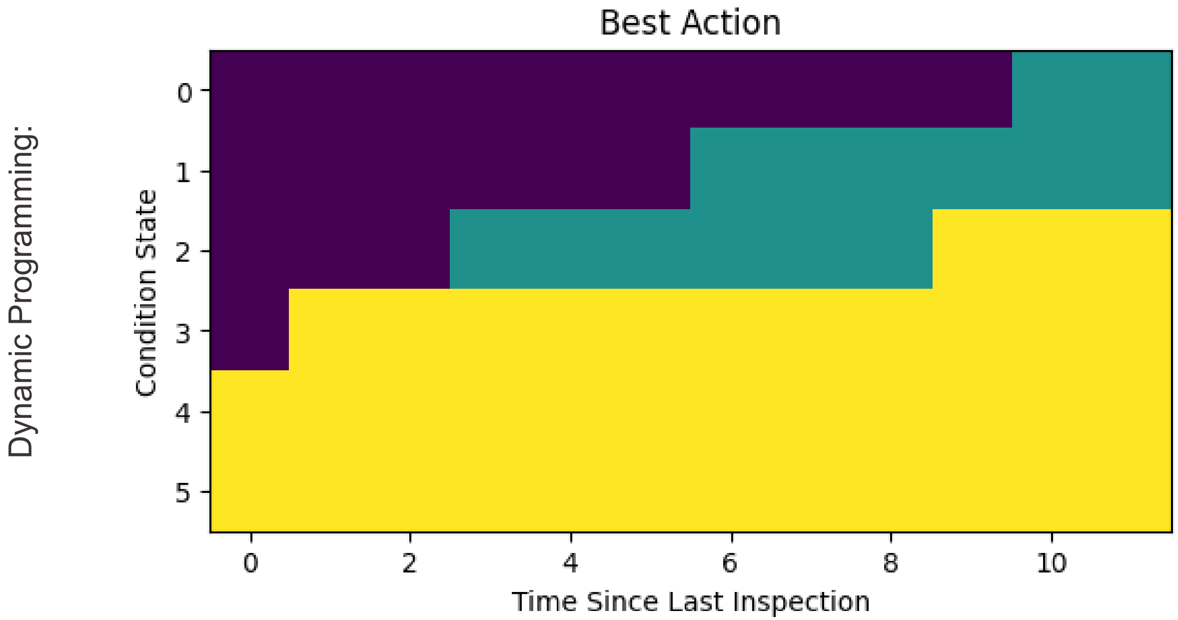
Costs:

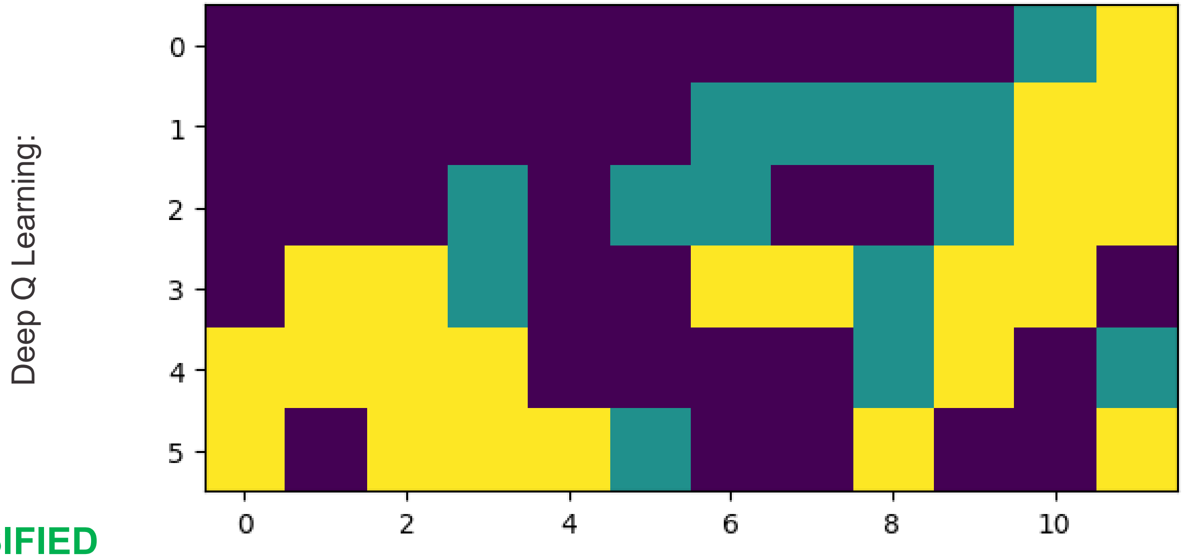
40 – Failure

10 – Replacement

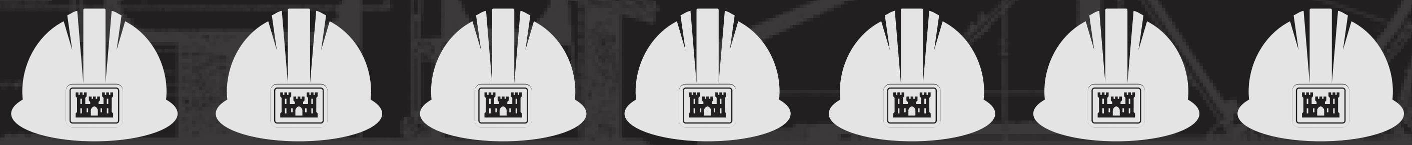
0.5 – Inspection







MUTLI-AGENT



U.S. ARMY

US Army Corps of Engineers®

MULTI AGENT SYSTEM EXAMPLES

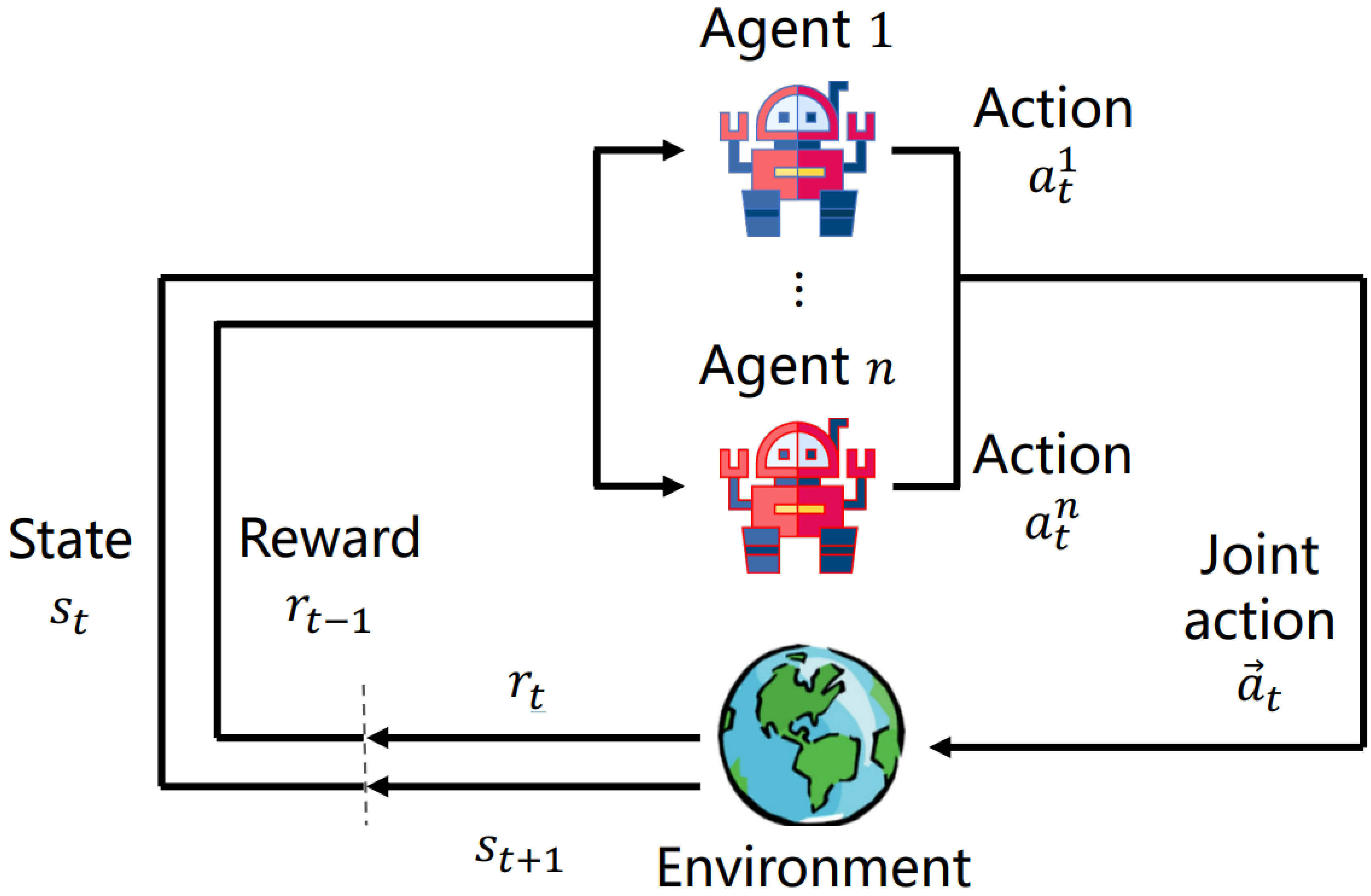
COOPERATION VS COMPETITION

STOCHASTIC GAMES

- The new problem is Non-Stationary with respect to an individual agent.

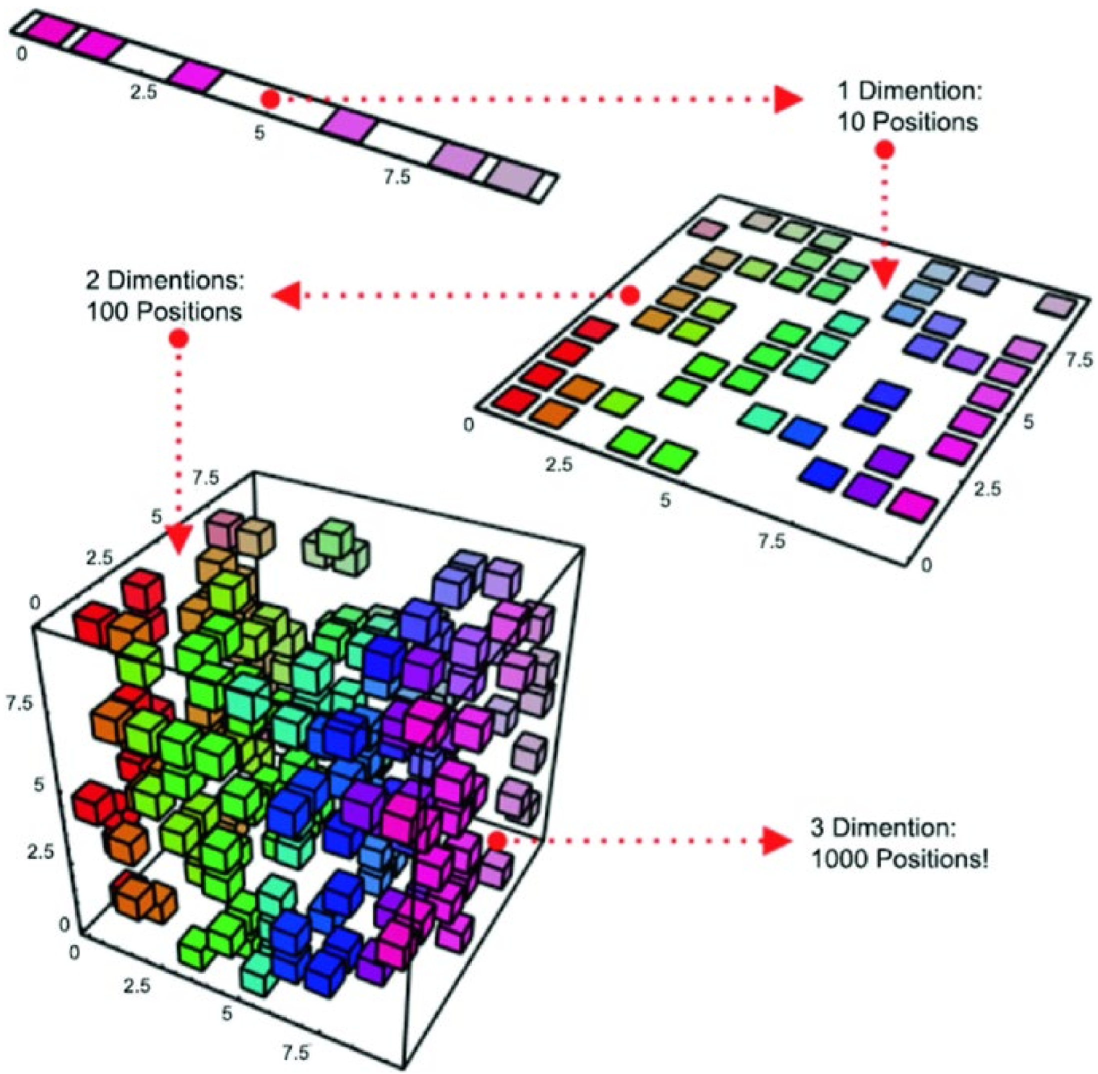
- Each agent must learn the best response to other agents' policies

- $Q_i(s, a_i | \pi^{-i})$

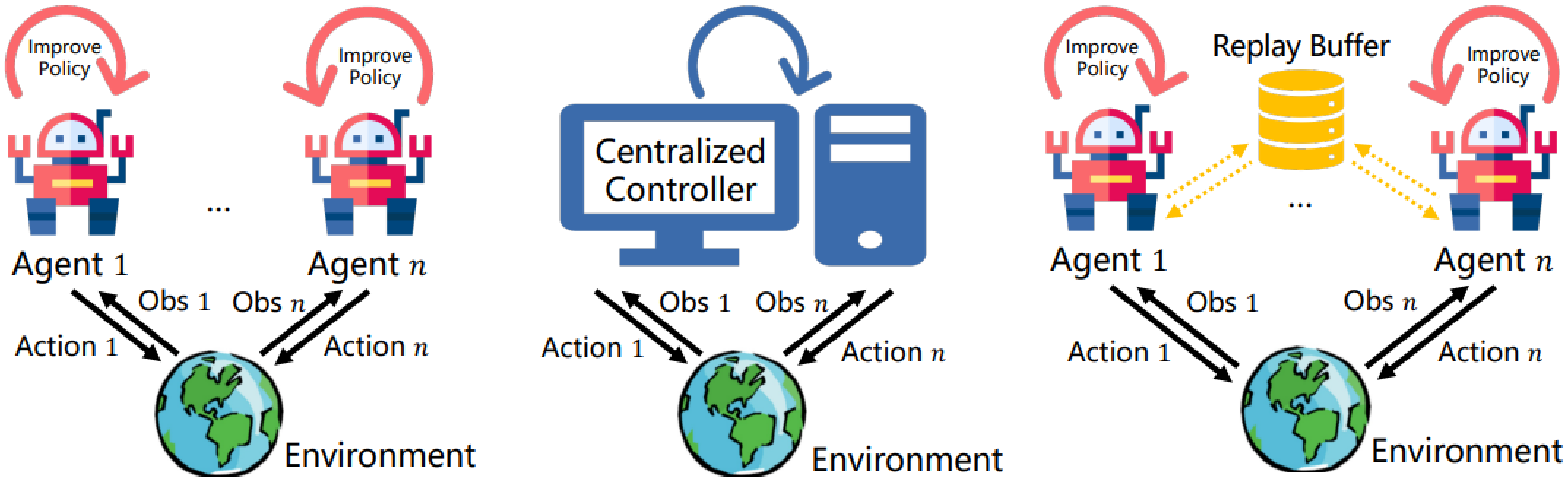

Agent 1
Action
a_t^1
⋮
Agent n
Action
a_t^n
State
s_t
Reward
r_{t-1}
r_t
Joint
action
$\vec{a}_t$
s_{t+1}
Environment

THE CURSE OF DIMENSIONALITY

- If each of n agents can take one of K actions, then there are K^n possibilities for each state.
- If each value is a byte of data, with **15 components** we would need more than a **terabyte** of data to store all **action values** for a single state.



CENTRALIZED VS DECENTRALIZED

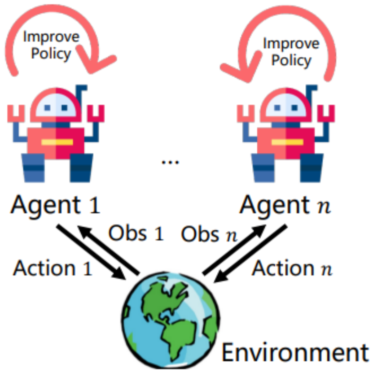


(a) Decentralized Training Decentralized Execution

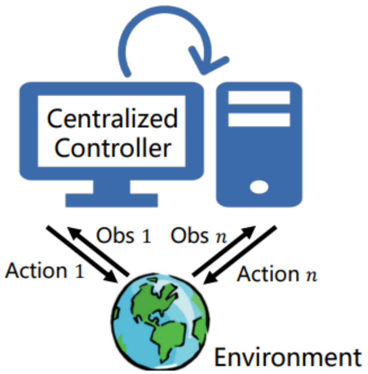
(b) Centralized Training Centralized Execution

(c) Centralized Training Decentralized Execution

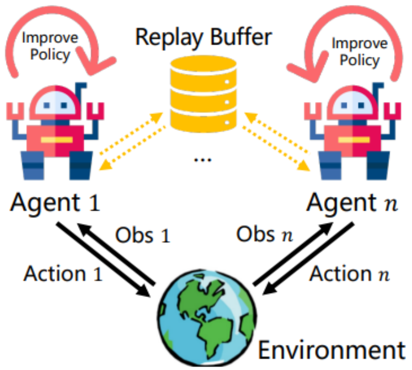
HIVE MINDS



(a) Decentralized Training
Decentralized Execution



(b) Centralized Training
Centralized Execution



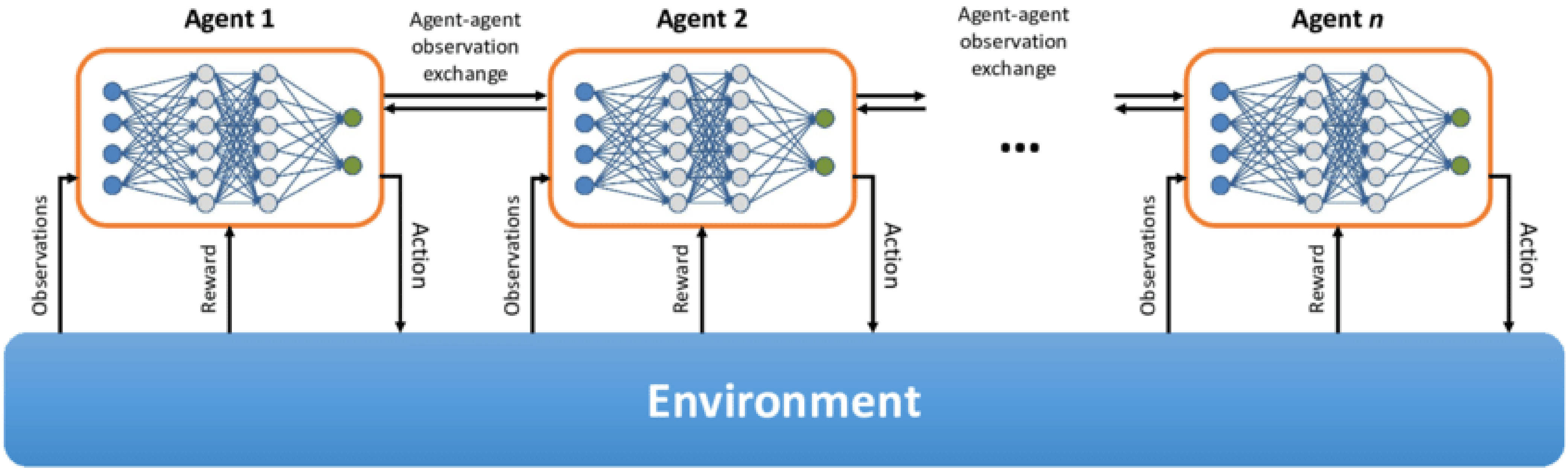
(c) Centralized Training
Decentralized Execution

MAJOR MARL CHALLENGES

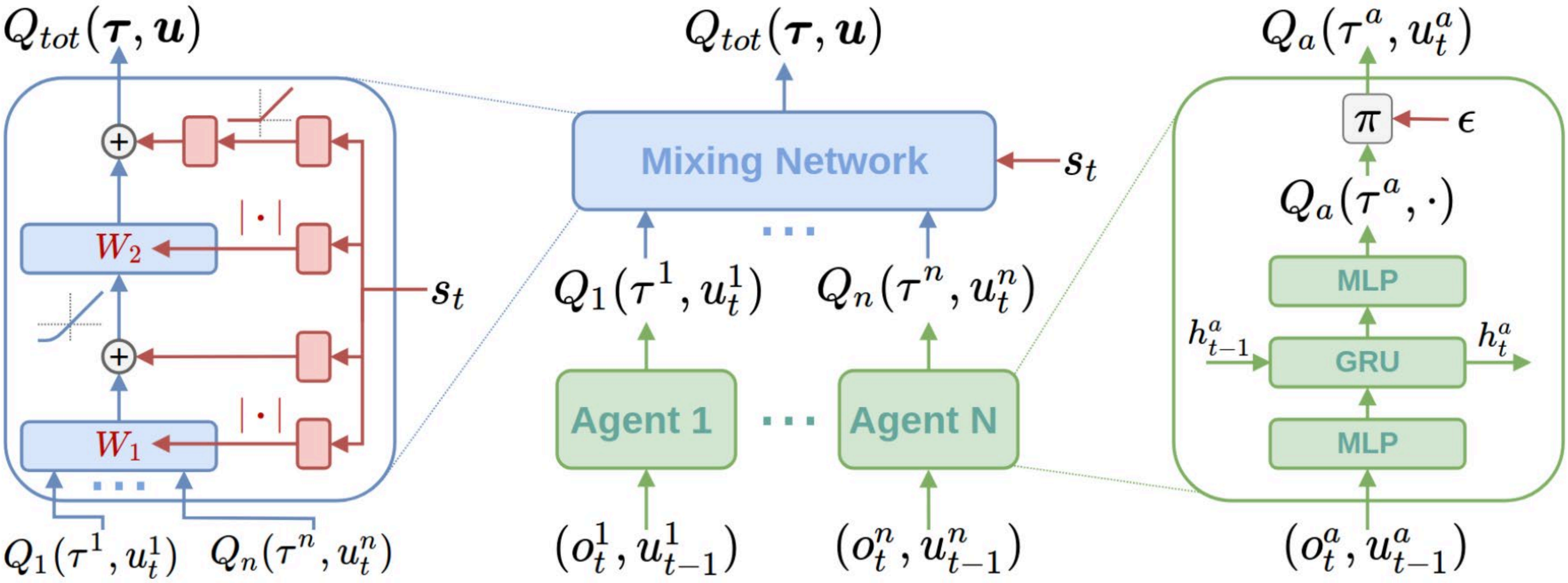
ERDC

- Credit Assignment
- Non-Stationary
- Observation & Privacy
- Cooperation Exploration

MADQRL


Agent 1
Agent-agent observation exchange
Agent 2
Agent-agent observation exchange
Agent n
...
Observations
Reward
Action
Observations
Reward
Action
Observations
Reward
Action
Environment

QMIX


$Q_{tot}(\boldsymbol{\tau}, \boldsymbol{u})$
W_2
$|\cdot|$
s_t
W_1
$|\cdot|$
$Q_1(\tau^1, u_t^1)$ $Q_n(\tau^n, u_t^n)$
$Q_{tot}(\boldsymbol{\tau}, \boldsymbol{u})$
Mixing Network
s_t
$Q_1(\tau^1, u_t^1)$ $Q_n(\tau^n, u_t^n)$
Agent 1
Agent N
(o_t^1, u_{t-1}^1) (o_t^n, u_{t-1}^n)
$Q_a(\tau^a, u_t^a)$
π
ϵ
$Q_a(\tau^a, \cdot)$
MLP
h_{t-1}^a
GRU
h_t^a
MLP
(o_t^a, u_{t-1}^a)

EXAMPLE

4 Agents learning when to best Inspect/Repair to minimize total maintenance cost.
Only 1 repair is allowed per timestep.