06 September 2019

Ministry of Defence

Safety Assurance of Autonomous Systems: Progress and Challenges

Rob Ashmore, Dstl Fellow

UK Defence Science and Technology Laboratory (Dstl)

DATAIA, September 2019, DSTL/TR117544

Disclaimer

This presentation is an overview of UK MOD sponsored research and is released for informational purposes only. The contents of this presentation should not be interpreted as representing the views of the UK MOD, nor should it be assumed that they reflect any current or future UK MOD policy. The information contained in this presentation cannot supersede any statutory or contractual requirements or liabilities and is offered without prejudice or commitment.

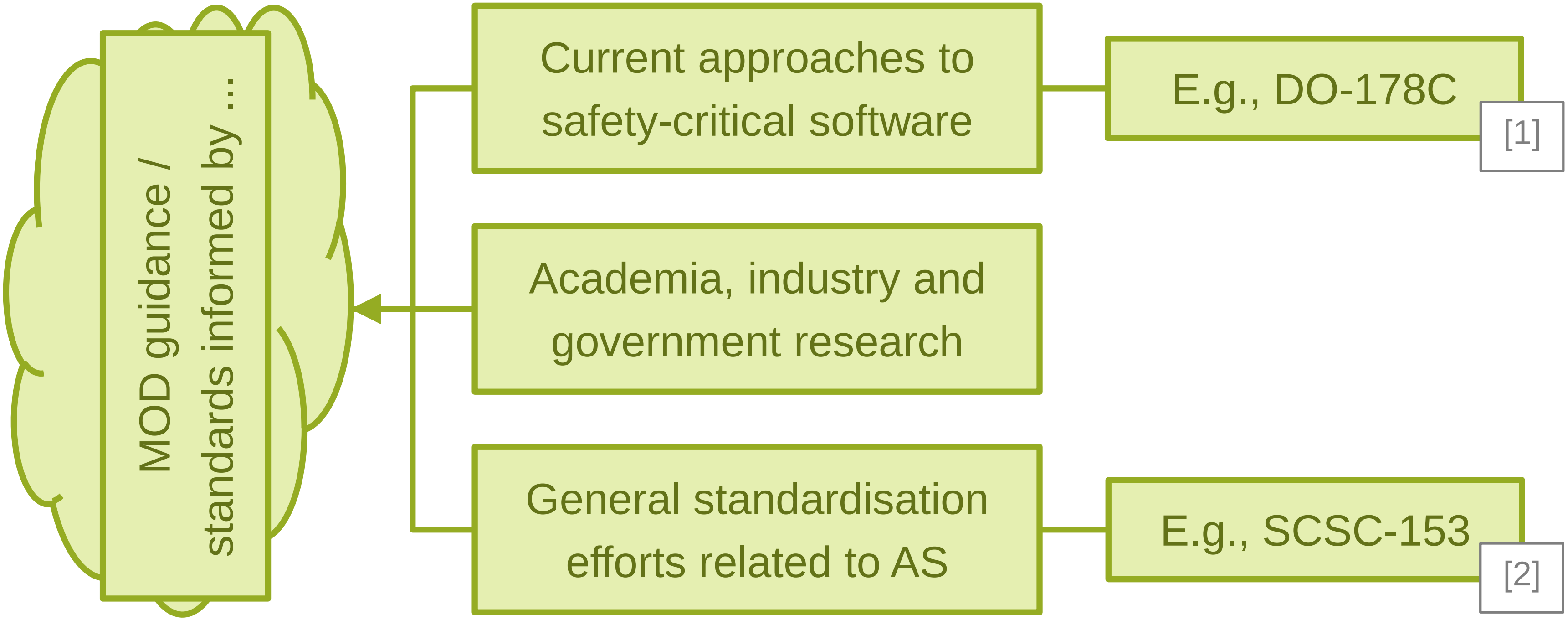
Intent

The assurance of Autonomous Systems so that they can be safely used with confidence

- **Assurance** is a logical, structured argument supported by evidence
- Supported by standards, which document RGP, as recognised by the community (developers, regulators, etc)
- Interested in a wide variety of **Autonomous Systems**, e.g., self-driving vehicles, image-based medical diagnostics
- Typically, but not always, based on AI implemented using ML based techniques; these are the main focus of this presentation

AI - Artificial Intelligence
ML - Machine Learning
RGP - Recognised Good Practice

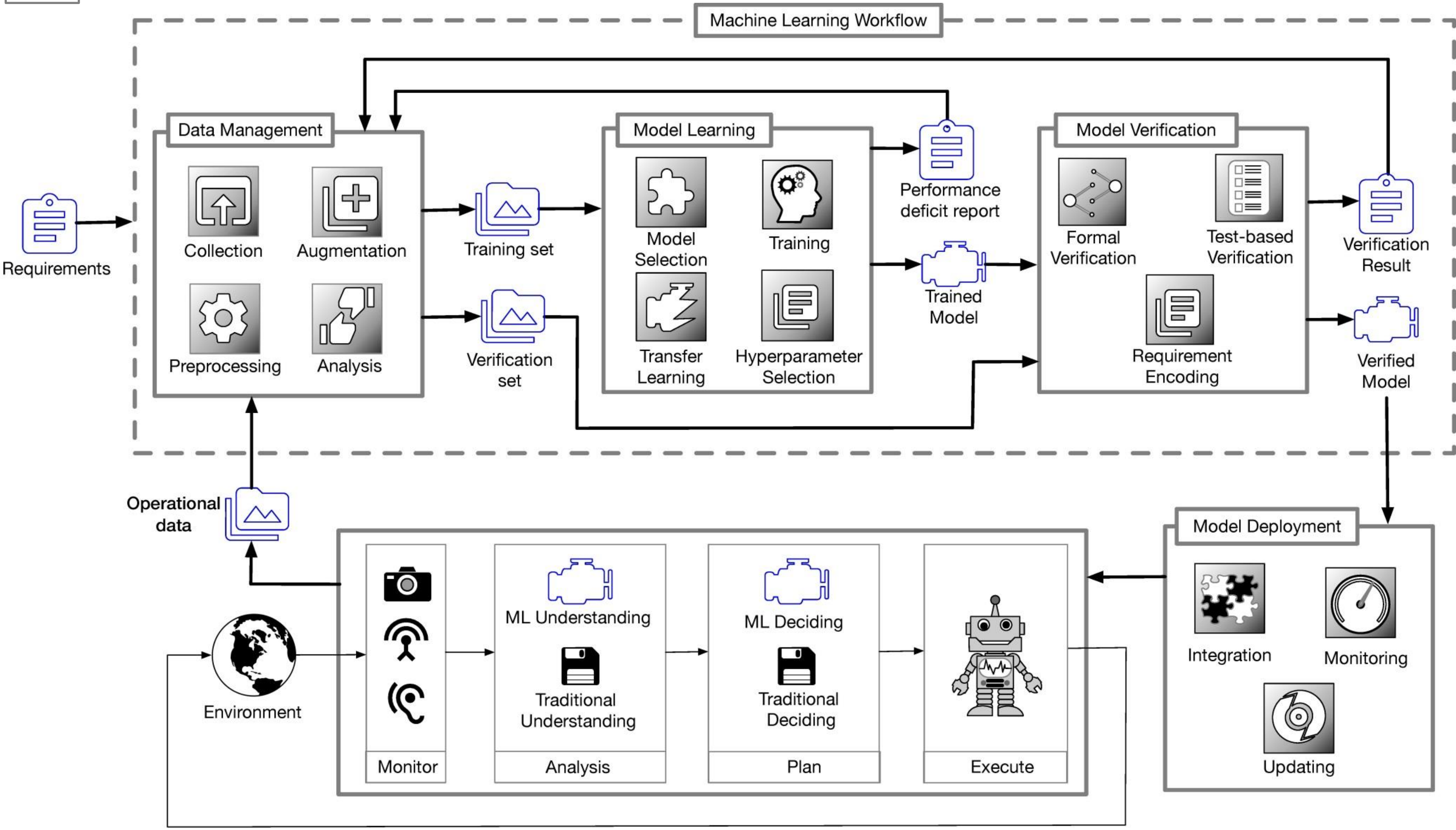
Approach



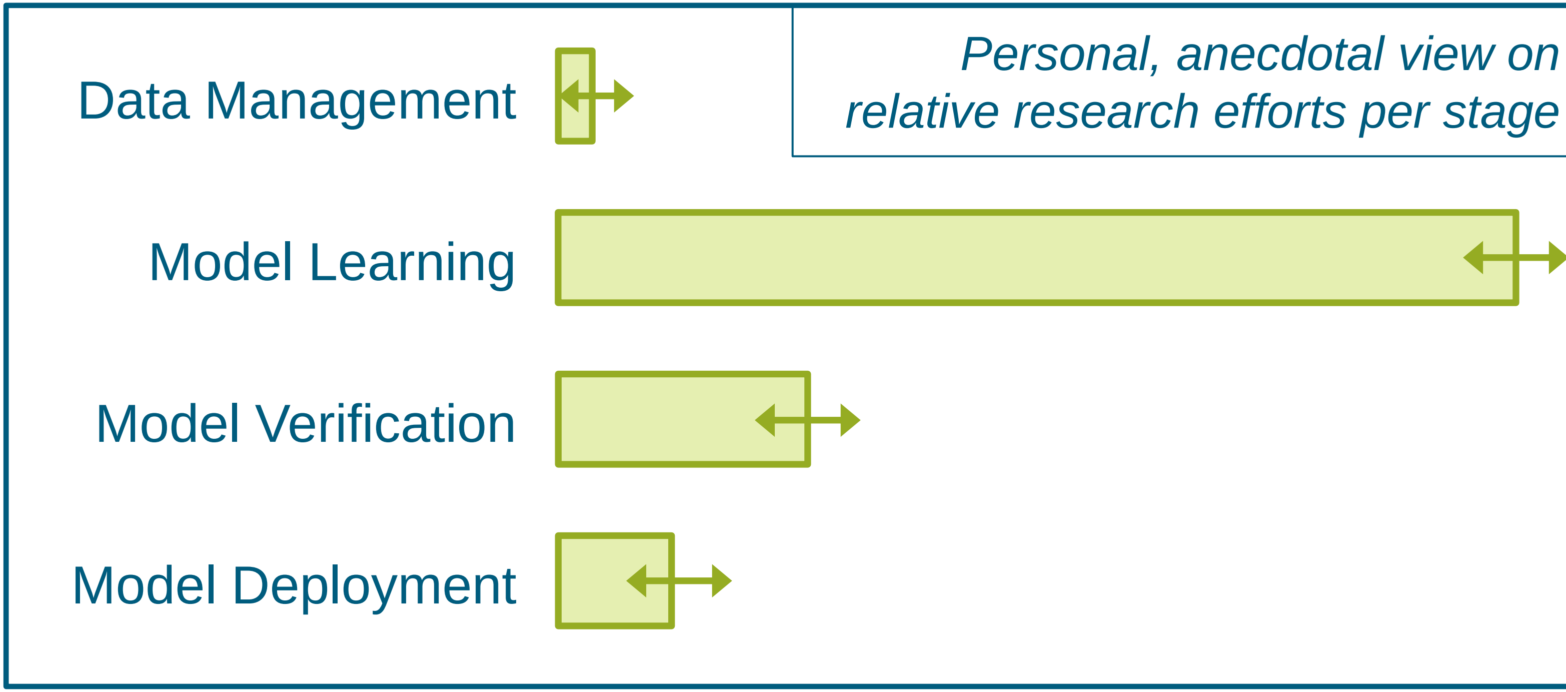
- Standards are tricky: have to be "accepted"; they should not lag too far behind technology; but they should not change too frequently or too dramatically

AS - Autonomous Systems
SCSC - Safety-Critical Systems Club

Problem Structure



Problem Structure



We need to cover all four stages; but some appear to be much more "interesting" than others

Requirements [4]

[https://xkcd.com/1838/]

Traditional

Requirements

Traceable, hierarchical decomposition

Code

AI / ML

Requirements

Data

Training

Code

Requirements are difficult!

AI - Artificial Intelligence
ML - Machine Learning

Requirements

There is often (but not always) a difference between safety requirements of real-world interest and safety requirements considered in academic papers

Academic Papers

- There are no adversarial inputs in an L_p ball around a training sample

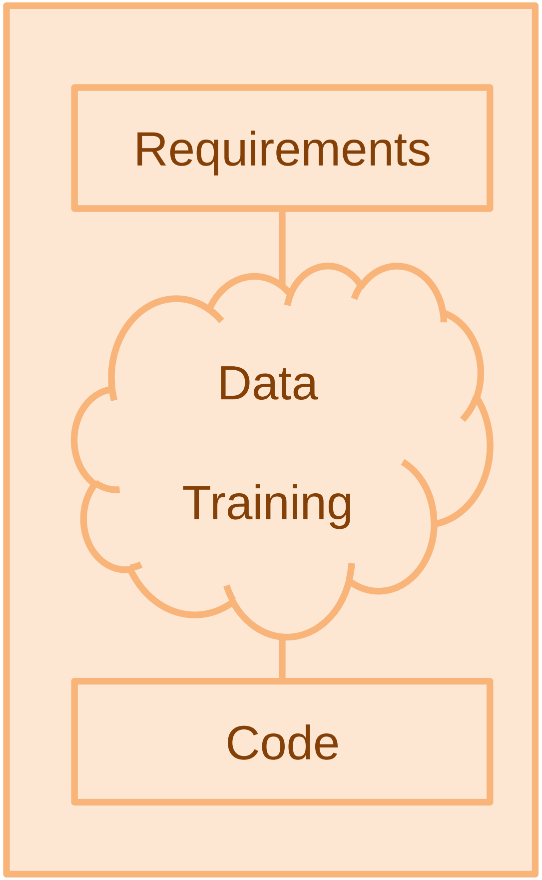
Real-World

- Class X will never be misclassified as Class Y
- There are no neighbouring inputs where one is Class X and the other is Class Y

Requirements [5]

In AI / ML system-level requirements are closely linked to training data

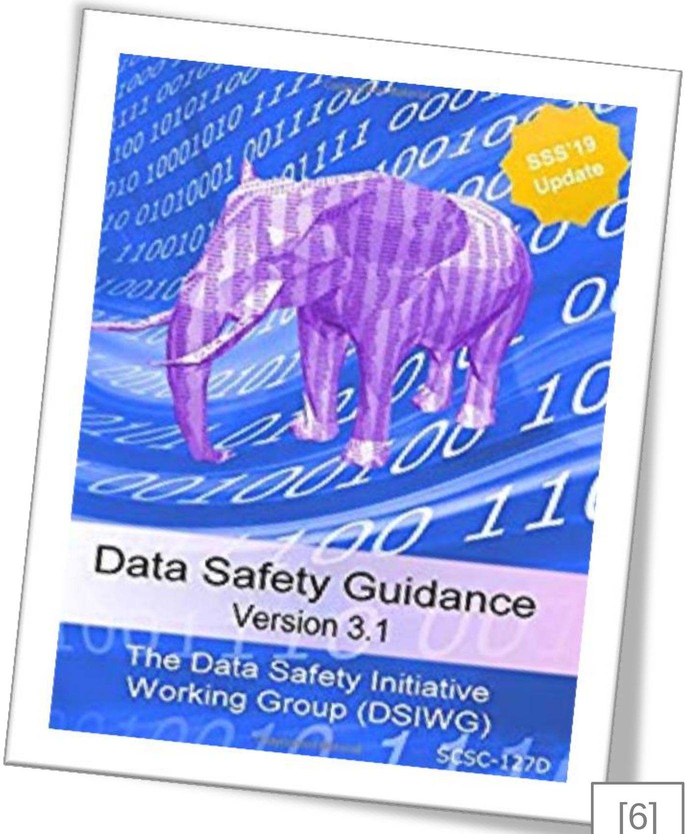
- D1. Relates to the intent of the HLR;
- D2. Does not contain bias;
- D3. Is sufficient;
- D4. Is syntactically and semantically correct;
- D5. Addresses normal and robustness behaviours;
- D6. Is self-consistent;
- D7. Conforms to standards;
- D8. Is compatible with target computer; and
- D9. Is verifiable.



AI - Artificial Intelligence
HLR - High-Level Requirement
ML - Machine Learning

Data Management

- Data Safety is an issue in ***all*** safety-related systems, as demonstrated by a number of historical accidents and incidents
- Consequently, system safety needs to consider software, hardware and data as first-class citizens
- The close link between requirements and training data means this is even more important for AI / ML approaches



[6]

Data Management

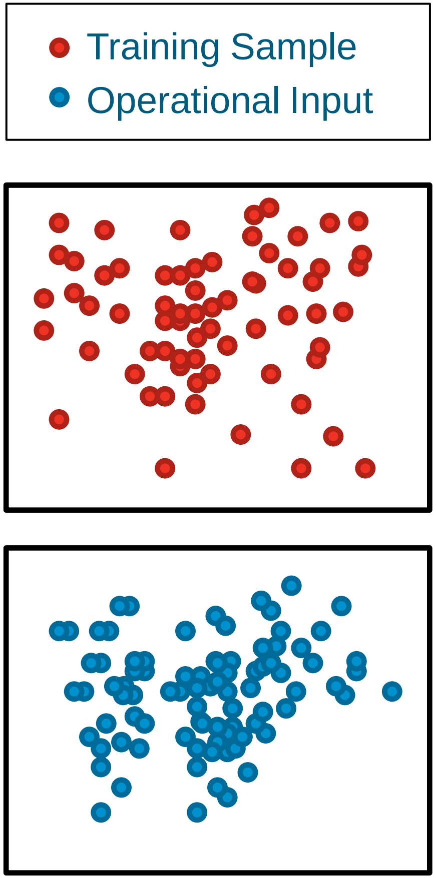
Space	Domain	Description (e.g., for Facial Recognition)
I	Input	Input parameters of software implementation (e.g., 256 x 256 x UINT8)
O	Operational	Expected inputs when used in intended operational domain (e.g., images of faces)
F	Failure	Inputs associated with failures elsewhere in the system (e.g., black pixels)
A	Adversarial	Inputs associated with deliberate attacks by an adversary

[3]

When thinking about completeness, it is helpful to consider four, related, domains; each needs to be covered appropriately

Data Management [7], [8]

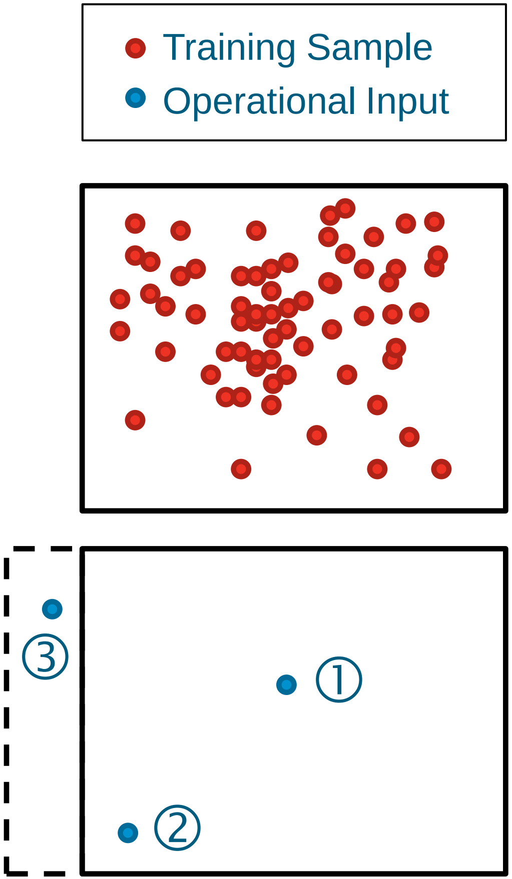
- We need to monitor inputs seen during operational use and compare them with the training data
- ***Distribution shift*** compares distributions; so we need some (often lots of) operational inputs before we can use a statistical analysis to make a decision
- A distribution shift indicates that we should not expect to achieve the same level of performance as we observed during the development process
- Comparatively, there is a lot of work on distribution shift; but important questions remain, e.g., ***when is a shift significant*** (MNIST 6s)?



Ministry of Defence

Data Management [9], [10]

- We need to monitor inputs seen during operational use and compare them with the training data
- Determining whether an operational input is within the support of the training data is an ***input-by-input*** decision
- To decide this we may need to know: bounds of training data; a distance metric; and whether there are any large holes in the training data
- Comparatively, there seems to be little work on this question: how would you answer it for the three points shown to the right?



Model Learning

- Oversimplifying things, model learning is about optimisation
- Choice of hyper-parameters, including model structure and training options, affect what can be learnt and how fast
- We need to detect and protect against "typical" errors, e.g., overfitting

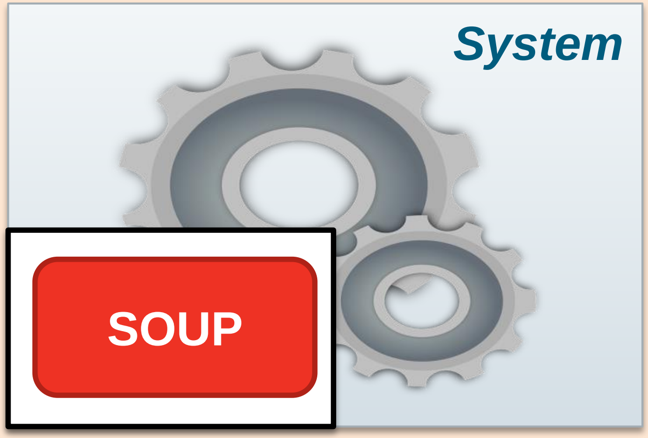
Outcome Judgements		*Model Prediction*	
		Healthy	*Disease*
Actual	*Healthy*	OK	Bad
	Disease	Very Bad	OK
Training Samples		9900	100

- Loss function is important; "always healthy" looks very good for this data

Model Learning

- Assurance argument needs to cover all aspects, not just those directly controlled by the development team
- ***Open-source frameworks*** are important; we cannot sandbox these and carefully control inputs and outputs
- ***Pre-trained models***, are also important; likewise, so are ***pre-prepared data sets***

[11]



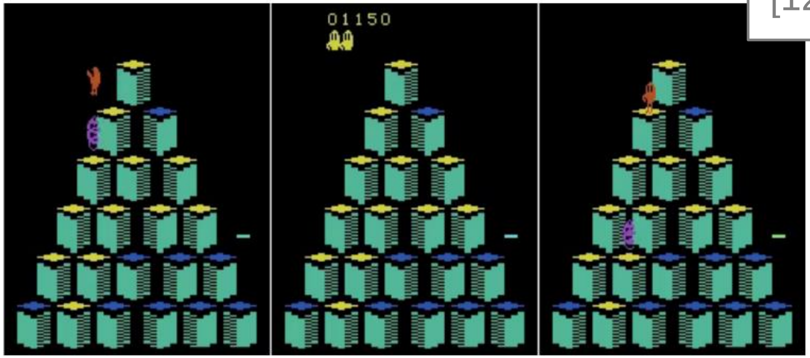
https://pixabay.com/vectors/cogwheel-gear-gearwheel-cog-145804/

- Checking for mistakes is one thing, looking for deliberate hostile acts is another

SOUP - Software Of Uncertain Provenance

Model Learning

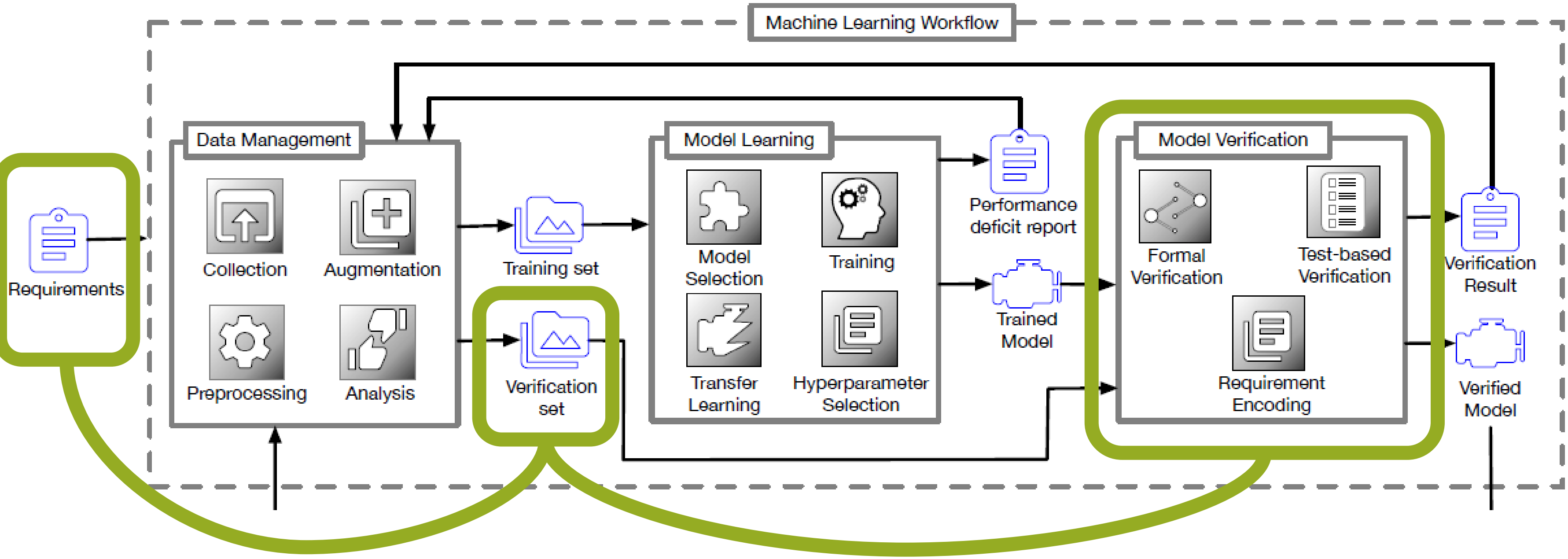
[12]



- Reinforcement Learning often makes use of simulation
- This is also applicable for other types of ML, e.g., to generate synthetic data (Data Management) or to estimate model performance (Model Verification)
- In these stages, simulation replaces things that might be too costly, or too dangerous, to conduct in the real world

- ***Demonstrating that the simulation is a suitable representation*** is a significant challenge
- Many examples of "reward hacking" where training exploits loopholes in the simulation

Model Verification

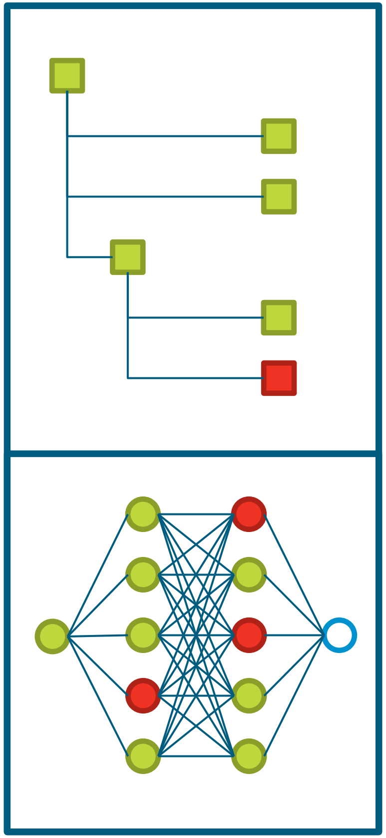


Requirements are encoded in the data set; part of this bypasses the development team (Model Learning) and goes straight to an independent verification team (Model Verification)

Model Verification

- Coverage is an important consideration; it shows (roughly) how much of the software's potential behaviour has been exposed during verification
- Traditional software testing supplements coverage of requirements with notions like statement, branch and MC/DC coverage [1]
- Equivalent notions are being suggested, especially for DNNs, but there is little empirical evidence that are meaningful and some suggestions they are not [13]

Good coverage measures, with theoretical and empirical justification, are not available



DNN - Deep Neural Network
MC/DC - Modified Condition / Decision Coverage

Model Verification

[14]

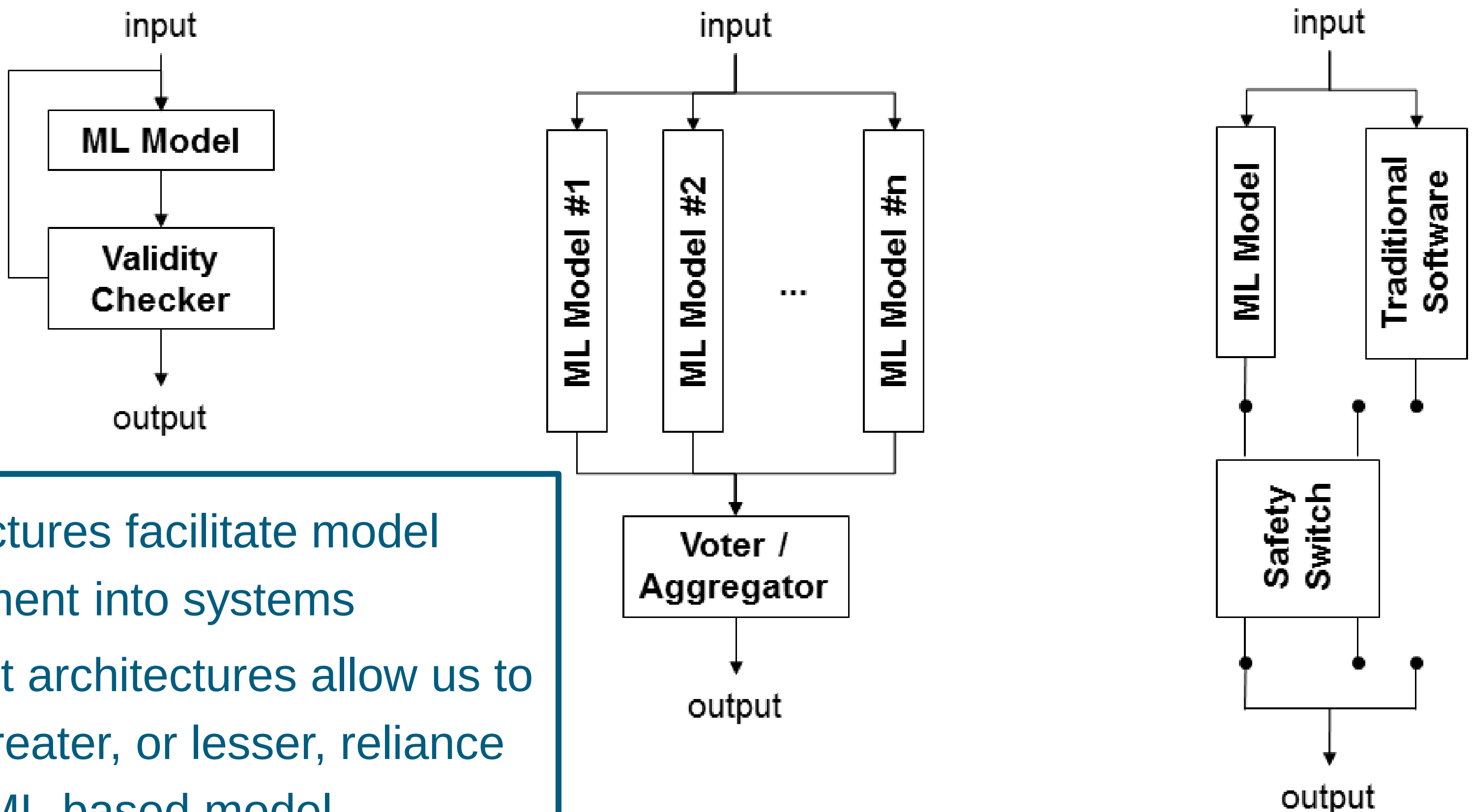
Local Explainability

- About how the model responds to a single input
- Lots of good progress in this area; e.g., we can build a simple, explainable-by-design model around the input

Global Explainability

- About how the model responds to classes of input, or the entire input domain
- Cannot be achieved by repeated local explainability
- Could restrict ourselves to explainable-by-design models
- But, generally speaking, ***this is an open challenge***

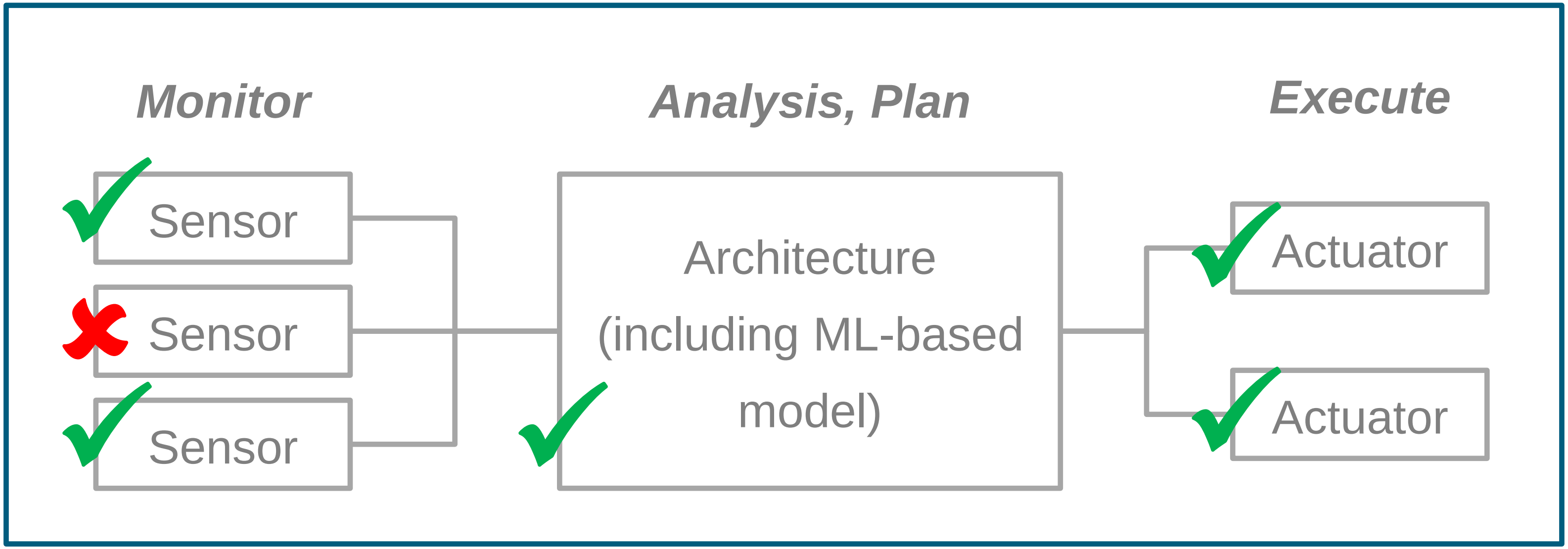
Model Deployment



- Architectures facilitate model deployment into systems
- Different architectures allow us to place greater, or lesser, reliance on the ML-based model

ML - Machine Learning

Model Deployment



- We need to ***monitor sub-system health***, e.g., of things that provide inputs to the model
- And, also health of the model itself

- We need to think about how we ***update the model***, e.g., when is a safe time? how do we handle failed updates?

(Multiple) Model Deployment [15]

Suppose you are responsible for a world-wide collection of data centres

Would you run each data centre at exactly the same software version level?

(Multiple) Model Deployment [15]

Suppose you are responsible for a world-wide collection of data centres

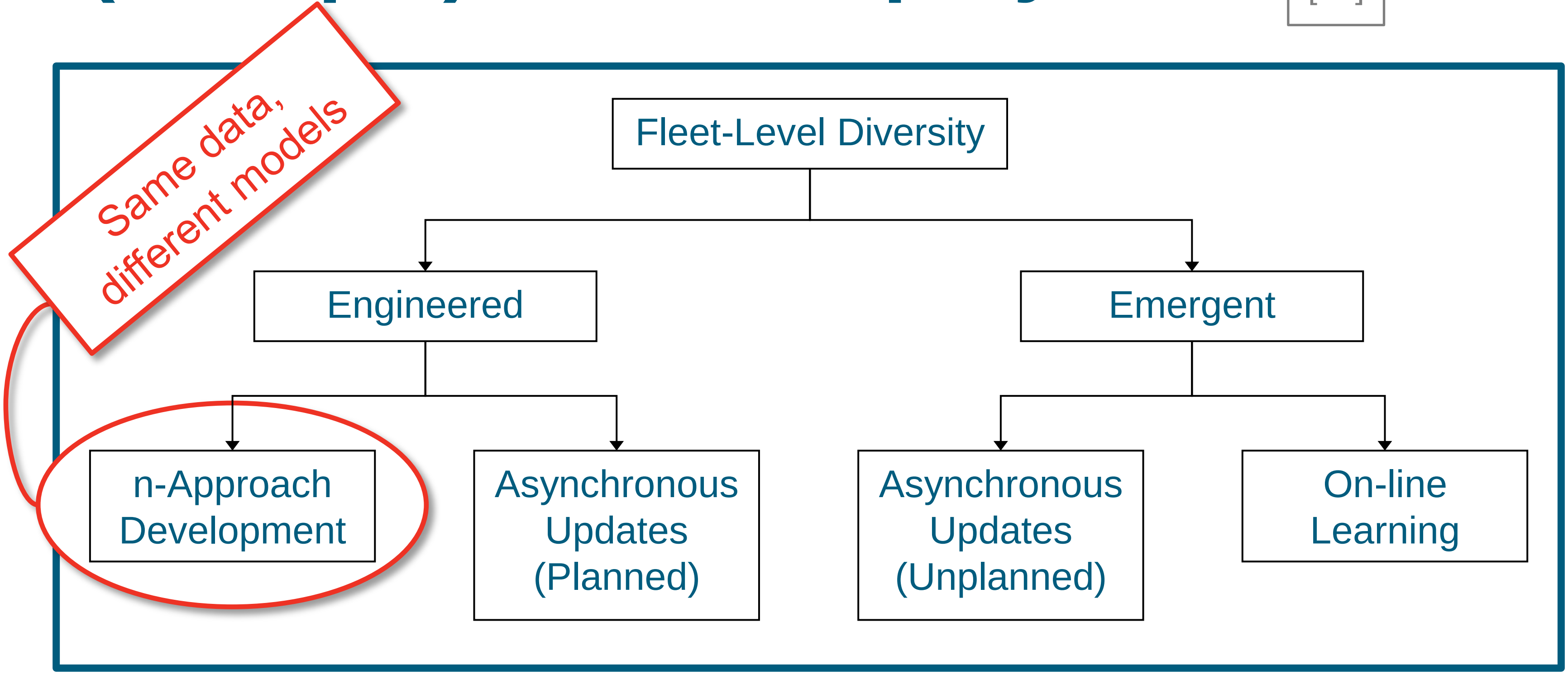
Would you run each data centre at exactly the same software version level?

Suppose you are responsible for a multiple-engine aircraft

Would you run each engine at exactly the same software version level?

Difference between these cases informs fleet-level diversity considerations

(Multiple) Model Deployment [15]



Fleet-level diversity may be engineered, or it may emerge; regardless it needs to be monitored and controlled appropriately

Closing Thoughts

- Assurance is a necessary enabler for practical use of Autonomous Systems that exploit AI developed using ML techniques
- This should be based on a structured argument, informed by RGP and supported by evidence
- There is lots of good work, but this is heavily focused on limited parts of the problem
- Areas that would benefit from greater consideration include: Requirements; Data; Frameworks; Simulation; Coverage; Global Explainability; Multiple Deployments

AI - Artificial Intelligence
ML - Machine Learning
RGP - Recognised Good Practice

References

[1] RTCA. Software Considerations in Airborne Systems and Equipment Certification. DO-178C, 2011.

[2] SCSC. Safety Assurance Objectives for Autonomous Systems. SCSC-153, 2019. Available from: https://www.amazon.co.uk/Safety-Assurance-Objectives-Autonomous-Systems/dp/1790421225 (hard copy); and https://scsc.uk/scsc-153 (soft copy).

[3] Ashmore R, Calinescu R, Paterson C. Assuring the Machine Learning Lifecycle: Desiderata, Methods, and Challenges. https://arxiv.org/abs/1905.04223, 2019.

[4] Ashmore R, Lennon E. Progress Towards the Assurance of Non-Traditional Software. In Developments in System Safety Engineering, Proceedings of the Twenty-fifth Safety-Critical Systems Symposium, Safety-Critical Systems Club, 2017. ISBN 978-1540796288.

[5] Banks A, Ashmore R. Requirements Assurance in Machine Learning (ML) Applications. In SafeAI 2019.

[6] SCSC. Data Safety Guidance, Version 3.1. SCSC-127D, 2019. Available from: https://www.amazon.co.uk/Safety-Guidance-Initiative-Working-Group/dp/1793375763 (hard copy); and https://scsc.uk/scsc-127D (soft copy).

[7] Moreno-Torres JG, Raeder T, Alaiz-RodríGuez R, Chawla NV, Herrera F. A unifying view on dataset shift in classification. Pattern Recognition, 45(1), pp.521-530, 2012.

[8] Rabanser S, Günnemann S, and Lipton ZC. Failing Loudly: An Empirical Study of Methods for Detecting Dataset Shift. https://arxiv.org/abs/1810.11953, 2018.

[9] Ashmore R, Hill M. “Boxing Clever”: Practical Techniques for Gaining Insights into Training Data and Monitoring Distribution Shift. In International Conference on Computer Safety, Reliability, and Security (pp. 393-405). Springer, Cham, 2018.

[10] Lemley J, Jagodzinski F, Andonie R, June. Big holes in big data: A Monte Carlo algorithm for detecting large hyper-rectangles in high dimensional data. In 2016 IEEE 40th Annual Computer Software and Applications Conference (COMPSAC) (Vol. 1, pp. 563-571), 2016.

[11] Gu T, Liu K, Dolan-Gavitt B, Garg S. BadNets: Evaluating Backdooring Attacks on Deep Neural Networks. IEEE Access, 7, pp.47230-47244, 2019.

[12] Chrabaszcz P, Loshchilov I, Hutter F. Back to basics: Benchmarking canonical evolution strategies for playing Atari. https://arxiv.org/abs/1802.08842, 2018.

[13] Li Z, Ma X, Xu C, Cao C.. Structural coverage criteria for neural networks could be misleading. In Proceedings of the 41st International Conference on Software Engineering: New Ideas and Emerging Results (pp. 89-92). IEEE Press, 2019.

[14] Ribeiro MT, Singh S, Guestrin C. Why should i trust you?: Explaining the predictions of any classifier. In Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining (pp. 1135-1144). ACM, 2016.

[15] Ashmore R, Madahar B. Rethinking Diversity in the Context of Autonomous Systems. In Engineering Safe Autonomy, 27th Safety-Critical Systems Symposium. 175–192, 2019.

06 September 2019

Ministry of Defence