

# Systems AI: Machine Learning Technology Readiness Levels

**Alexander Lavin**

Founder & Chief Technologist, Pasteur Labs & ISI

Lead & Mentor, FDL AI Lab



Immunai | Jan 22, 2022



# Alexander Lavin

- Probabilistic ML + engineering science
  - Probabilistic Programming, GP + BO
  - Probabilistic Numerics and Computation
  - *Simulation Intelligence*
- Now:
  - Pasteur Labs & ISI → [simulation.science](https://simulation.science)
  - NASA AI @ [frontierdevelopmentlab.org](https://frontierdevelopmentlab.org)
- Previously:
  - Vicarious AI, Numenta, Blue Origin ...
  - Cornell, Carnegie Mellon, Duke

👉 [lavin.io](https://lavin.io)

🐦 @AlexLavin\_C137



# Outline

1. Setting the scene
2. Systems Engineering and AI
3. MLTRL
4. Examples
5. Takeaways

## TECHNOLOGY READINESS LEVELS FOR MACHINE LEARNING SYSTEMS

**Alexander Lavin\***  
Pasteur Labs

**Ciarán M. Gilligan-Lee**  
Spotify

**Alessya Visnjic**  
WhyLabs

**Siddha Ganju**  
Nvidia

**Dava Newman**  
MIT

**Atılım Güneş Baydin**  
University of Oxford

**Sujoy Ganguly**  
Unity AI

**Danny Lange**  
Unity AI

**Amit Sharma**  
Microsoft Research

**Stephan Zheng**  
Salesforce Research

**Eric P. Xing**  
Petuum

**Adam Gibson**  
Konduit

**James Parr**  
NASA Frontier Development Lab

**Chris Mattmann**  
NASA Jet Propulsion Lab

**Yarin Gal**  
Alan Turing Institute

[arxiv.org/abs/2101.03989](https://arxiv.org/abs/2101.03989)

*Artificial Intelligence:* systems to enable rational decision making under uncertainties.

*Artificial Intelligence:* **systems** to enable rational decision making under uncertainties.

Holistic perspective: complex, dynamic combinations of data + software + hardware + humans

*Artificial Intelligence: systems to enable rational decision making under uncertainties.*

*Decisions / Causality: If I take this action today, what will be the outcome tomorrow?*

*Artificial Intelligence:* systems to enable rational decision making under **uncertainties**.

Principled uncertainty reasoning is essential for useful AI.

# Outline

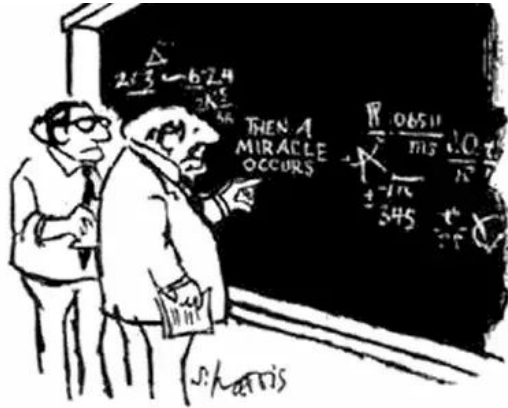
1. Setting the scene
2. **Systems Engineering and AI**
3. MLTRL
4. Examples
5. Takeaways



SW req's and spec

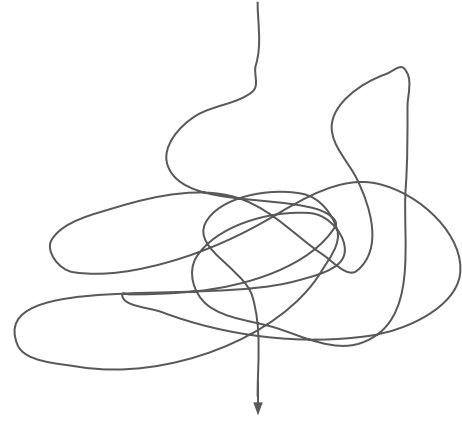


shipped SW product



"I THINK YOU SHOULD BE MORE EXPLICIT  
HERE IN STEP TWO."

ML development



deployed ML system

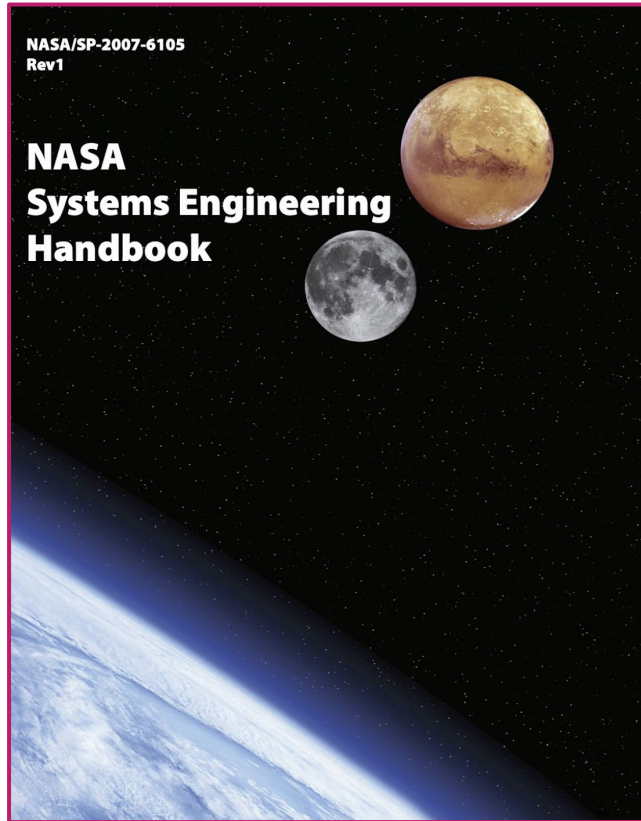


While the building blocks are in place, the principles for putting these blocks together are not, and so the blocks are currently being put together in ad-hoc ways...

Unfortunately, we are not very good at anticipating what the next emerging serious flaw will be. [What we're missing is an engineering discipline with principles of analysis and design.](#)"

*Michael Jordan, Prof. UC Berkeley*

*Systems Engineering* describes the principled processes and organizational frameworks that enable the cohesion and synergy of complex, interdependent subsystems.



## Technology Readiness Levels

Technology Readiness Levels (TRL): a systems engineering protocol for deep tech and scientific endeavors at scale, ideal for integrating many interdependent components and cross-functional teams of people, ensuring robust, safe systems.

## Machine Learning Technology Readiness Levels (MLTRL)

*Industry proven, deep-tech systems engineering, but lean for efficient AI & ML research, development, productization, and deployment.*

Organizational mechanisms, empowering inter-team collaboration and principled processes; defines a lingua franca for all stakeholders.

Ensures reliable, robust, responsible AI technologies.

# Outline

1. Setting the scene
2. Systems Engineering and AI
3. **MLTRL**
  - a. Overall process
  - b. Anatomy of a level
  - c. Key components
4. Example
5. Takeaways

# AI & ML Readiness Levels

## **TRL 0. First Principles**

A stage for greenfield research.

## **TRL 1. Goal-oriented Research**

Moving from basic principles to practical use.

## **TRL 2. Proof of Principle (PoP) Development**

Active R&D is initiated.

## **TRL 3. Systems Development**

Software integration.

## **TRL 4. Proof of Concept (PoC) Development**

Demonstration in a real scenario.

## **TRL 5. Machine Learning “Capability”**

The R&D to product handoff.

## **TRL 6. Application development**

Robustification of ML modules, specifically towards one or more use-cases

## **TRL 7. Integrations**

ML infrastructure, product platform, data pipes, security protocols

## **TRL 8. Flight-ready**

The end of system development.

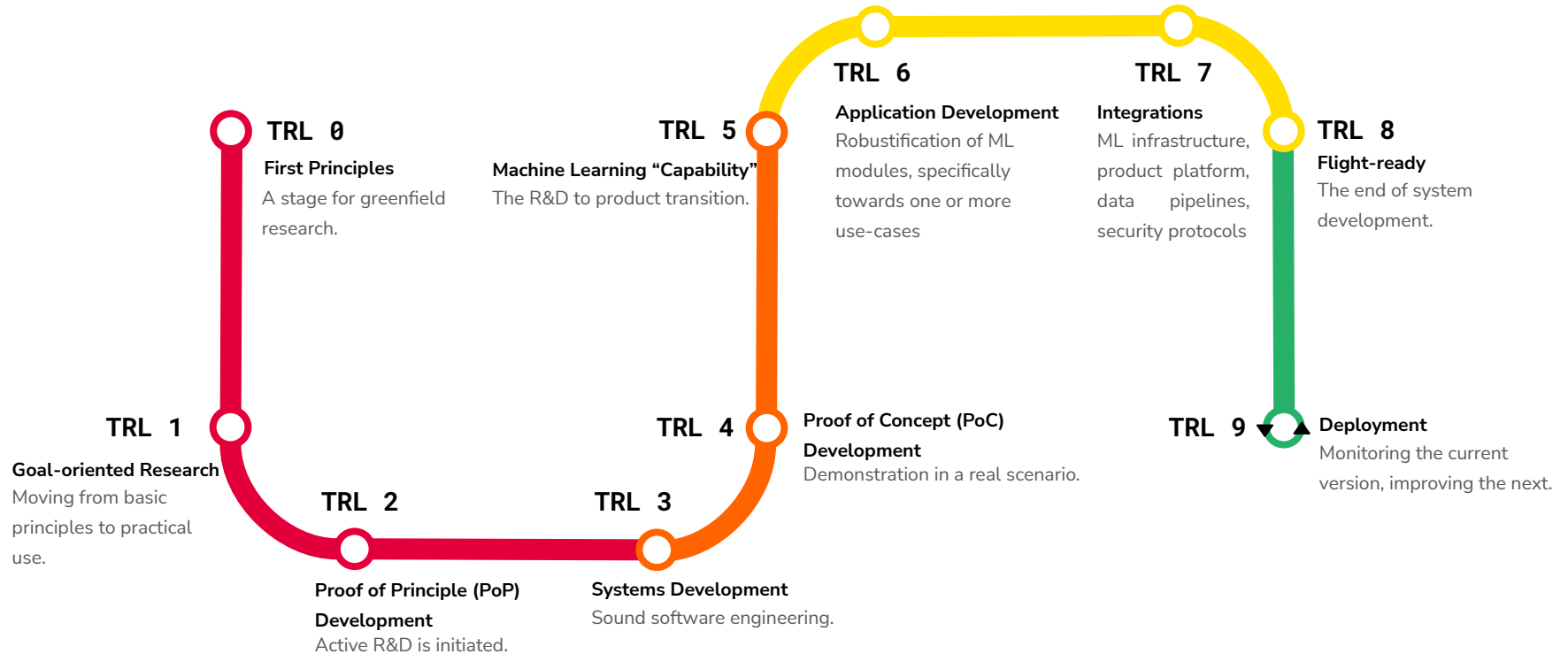
## **TRL 9. Deployment**

Monitoring the current version, improving the next.

## AI & ML Readiness Levels

A *technology readiness level (TRL)* represents the maturity of a model or algorithm, data pipes, software module, or composition thereof.

# Research ↔ Development ↔ Productization ↔ Deployment





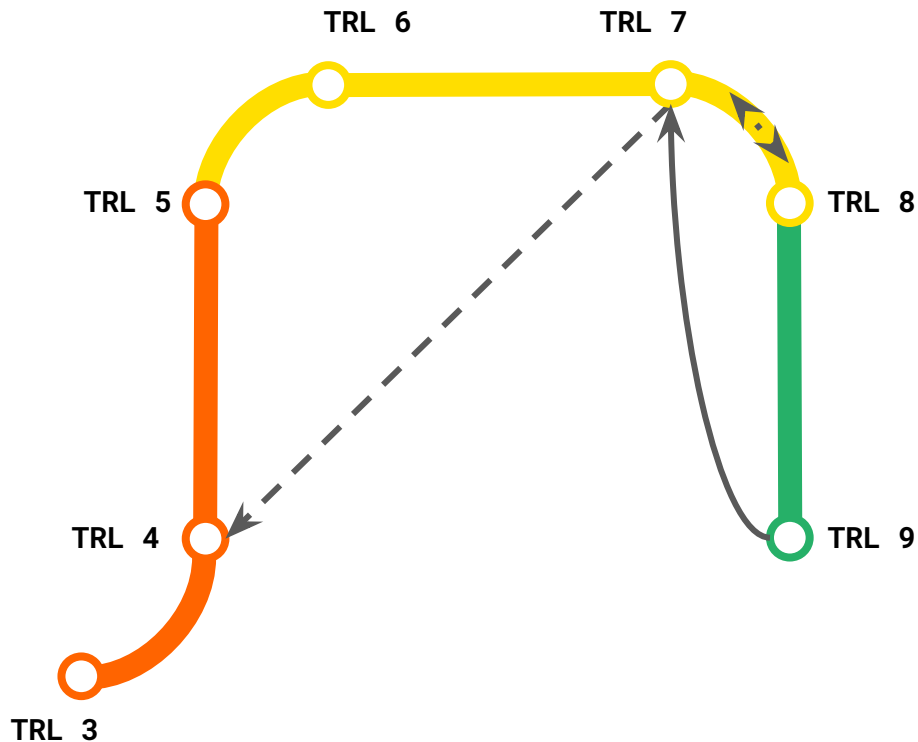
## Most ML projects...

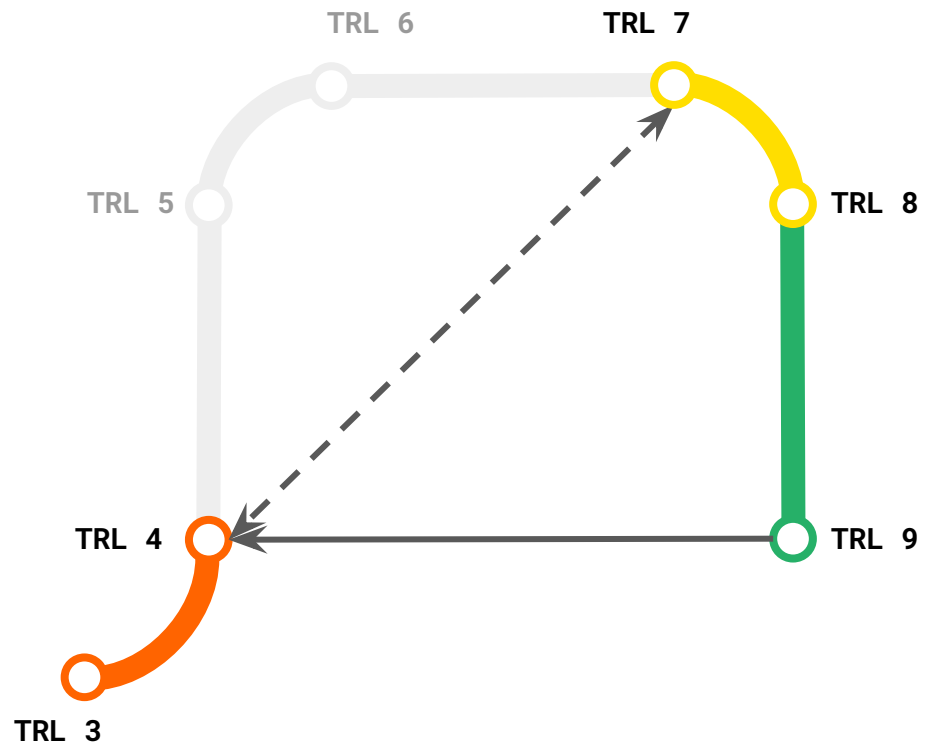
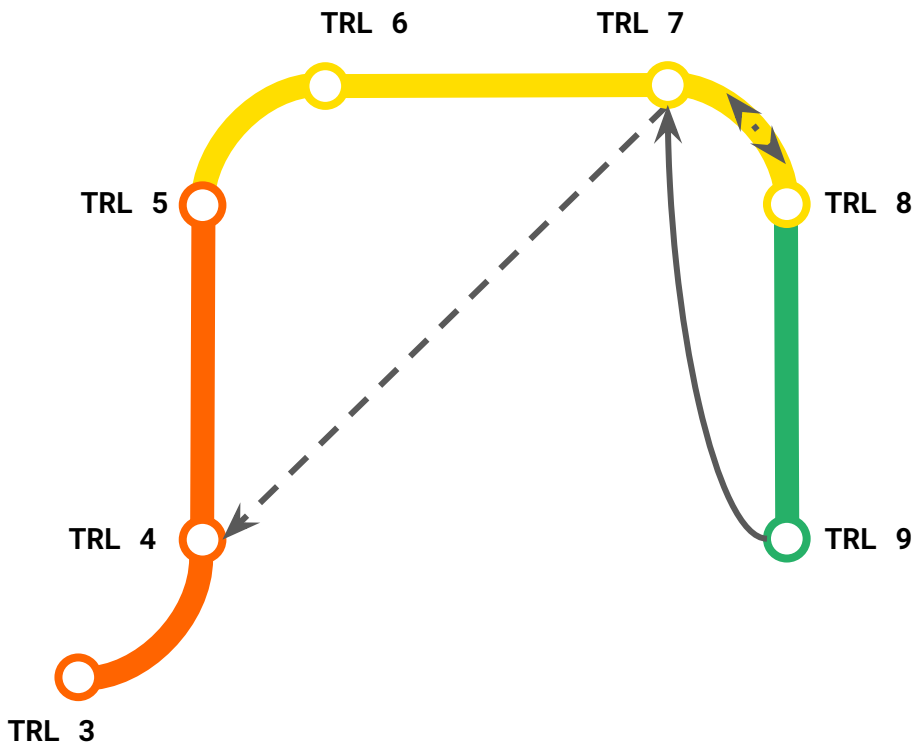
Project starts at TRL 3 or 4, developing with off-the-shelf models that are flight-proven.

Post deployment lifecycle is focused on,

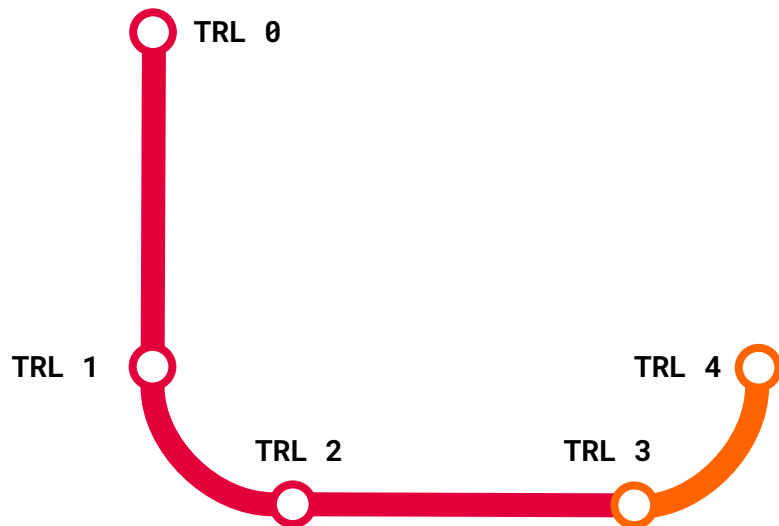
- monitoring
- developing new ML features (dashed line)
- incremental version improvements (solid line)

Frequently iteration between TRL 7 ("integrations") and 8 ("flight-ready")  
-- more tests on use-case specific critical scenarios and data-slices.





# Academic AI lives in TRL sub-5



Only universities and select industry research labs do fundamental R&D.

Projects are typically means to an end:

- PhD studies
- Publications and maybe tech demos

**In reality there's a non-trivial lab-to-practice gap!**

- AI tech transfer is a challenge (10x more than SW)
- Datasets and software from R&D are narrowly focused

# Anatomy of a MLTRL stage - Process

## Formal level reviews

- Present the tech developments and their validations, make key decisions on path(s) forward (or backward), debrief the process.
- Inclusion of **stakeholders and domain experts** ← key for interdisciplinary projects
- Stage-specific criteria and reviewers

# Anatomy of a MLTRL stage - People

## Working group, owners, stakeholders

Roles **evolve over lifetime**, e.g.:

- Stakeholders (ie reviewers) in R&D are largely AI peers, but in TRL 7 consist of QA engineers and PMs
- PMs take ownership after TRL 5, but R&D maintains key touch points

# Anatomy of a MLTRL stage - Execution

## Software engineering

Bringing agile workflows and best-practices *safely* to ML.

### Defined code-quality paradigms:

1. **RESEARCH**: Quick and dirty, moving fast through iterations of experiments. Hacky code is okay, and full test coverage is actually discouraged, as long as the overall codebase is organized and maintainable.
2. **PROTO**: Step up in robustness and cleanliness. This needs to be well-designed, well-architected for dataflow and interfaces, generally covered by unit and integration tests, meet team style standards, and sufficiently-documented.
3. **PRODUCT**: This code will be deployed to users and thus needs to follow precise spec, have comprehensive test coverage, well-defined APIs, etc.

# Key Components & Deliverables

## Formal requirements and V&V

- R&D and product versions (overlap at TRL 5)
- Req: a singular documented physical or functional need that a particular design, product, or process aims to satisfy.
- Verification: *Are we building the product right?*
- Validation: *Are we building the right product?*
- Need these docs for the gated reviews

## Risk matrices & mitigation strategies

$\text{risk} = p(\text{failure}) \times \text{value}$

- risk score for each [tech and product requirement](#)
- Explicit risk mitigation steps for sim-to-real transfer

## Process metrics and optimization

- OKRs and KPIs can be defined on TRL scale
- Identify operational bottlenecks
- Strategically minimize [ML tech debt](#)

## MLTRL “Cards”

### Non-linear, non-monotonic paths

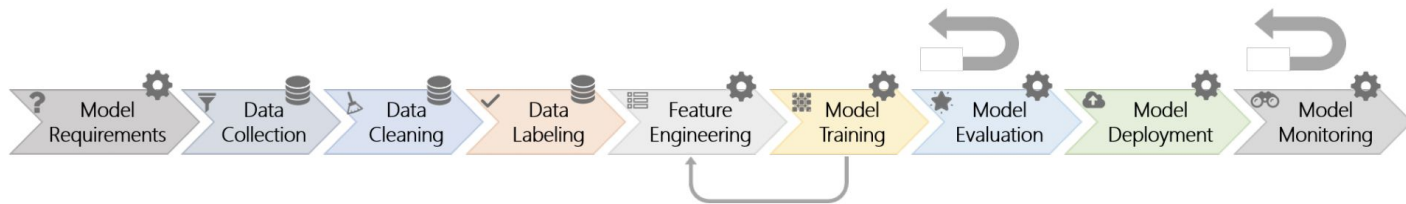
### Ethics prioritization and transparency

Components not in this presentation:

- Full ethics checklist
- Data readiness
- Specific distinctions from SWE
- ML testing suites and rubric

## Workflows require non-linear, non-monotonic paths

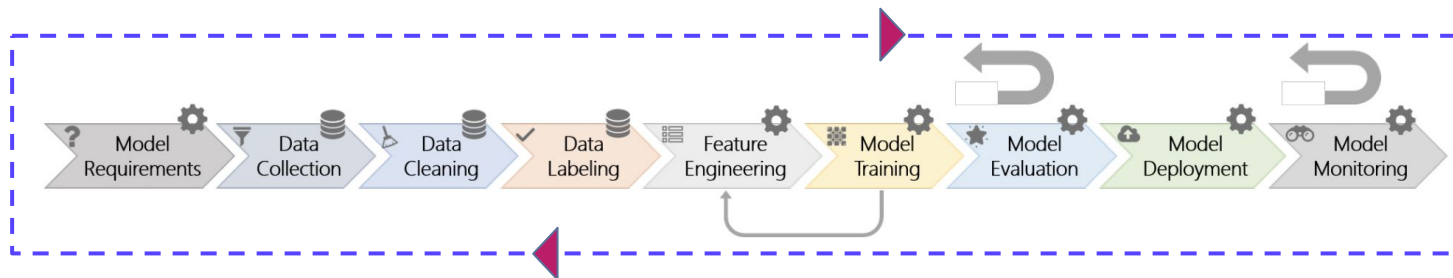
Current best-practice workflow is one straight path with minimal feedback:





## Workflows require non-linear, non-monotonic paths

Iterate with subsequent levels as *full system* matures!



Why? **Evolving** people, requirements, validations, datasets, objectives.

# ML Reporting

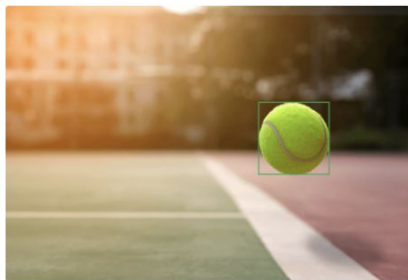
## Google model card (object recognition)

### Object Detection

The model analyzed in this card detects one or more physical objects within an image, from apparel and animals to tools and vehicles, and returns a box around each object, as well as a label and description for each object.

On this page, you can learn more about how the model performs on different classes of objects, and what kinds of images you should expect the model to perform well or poorly on.

### MODEL DESCRIPTION

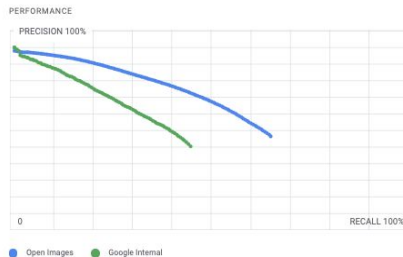


**Input:** Photo(s) or video(s)

**Output:** The model can detect 550+ different object classes. For each object detected in a photo or video, the model outputs:

- Object bounding box coordinates
- Knowledge graph ID ("MID")
- Label description
- Confidence score

**Model architecture:** Single shot detector model with a Resnet 101 backbone and a feature pyramid network feature map.



Performance evaluated for specific object classes recognized by the model (e.g. shirt, muffin), and for categories of objects (e.g. apparel, food).

Two performance metrics are reported:

- Average Precision (AP)
- Recall at 60% Precision

Performance evaluated on two datasets distinct from the training set:

- Open Images Validation set, which contains ~40k images and 600 object classes, of which the model can recognize 518.
- An internal Google dataset of ~5,000 images of consumer products, containing 210 object classes, all of which model can recognize.

Too high-level

## Scientific product card (medical assay)

Useful details

### Quanterix<sup>®</sup>

The Science of Precision Health

### Simoa<sup>™</sup> TDP-43 Kit

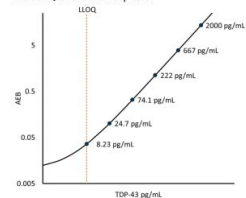
SR-X<sup>™</sup> Data Sheet

Item 103293

### Description

The TAR DNA binding protein of 43 kDa (TDP-43 or TARDBP) is a highly conserved and ubiquitously expressed nuclear protein with roles in transcription and splicing regulation. It is also the major component of ubiquitin-positive cytoplasmic inclusions found in the brains of patients with frontotemporal lobar degeneration (FTLD) and amyotrophic lateral sclerosis (ALS). In addition, TDP-43-containing aggregates are found in a significant number of patients with Alzheimer's Disease (AD) and other neuromuscular disorders. The majority of TDP-43 protein found in cytoplasmic inclusions is truncated, and it has been shown that the C-terminal domain is intrinsically prone to aggregation. Mutations in the C-terminal region of the TDP-43 gene have been associated with both ALS and FTLD, and are thought to facilitate ubiquitination and phosphorylation of the TDP-43 protein, leading to the formation of pathological inclusions and eventual neurodegeneration. Analysis of TDP-43 levels in plasma may allow the indexing of TDP-43 pathology within the brain to aid in the diagnosis of different forms of dementia and distinguish between TDP-43 proteinopathy and tauopathy. The Simoa TDP-43 assay has been developed with a full-length protein calibrator and antibodies against AA 203 – 209 and the C-terminal region. It is expected to detect both full-length and pathological, truncated forms of the protein.

**Calibration Curve:** Calibrator concentrations and Lower Limit of Quantification depicted.



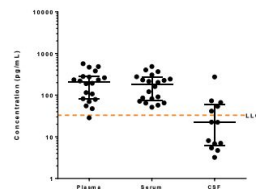
**Lower Limit of Quantification (LLOQ):** Triplicate measurements of serially diluted calibrator were read back on the calibration curve over 6 runs each for 1 reagent lot across 2 instruments (6 runs total).

**Limit of Detection (LOD):** Calculated as 2.5 standard deviations from the mean of background signal read back on each calibration curve over 6 runs each for 1 reagent lot across 2 instruments (6 runs total).

Analytical LLOQ	8.23 pg/mL pooled CV 11% mean recovery 112%
LOD	0.780 pg/mL range 0.019-1.59 pg/mL
Dynamic range (serum and plasma)	0 - 8000 pg/mL
Diluted Sample volume*	100 µL per measurement
Tests per kit	96

\*See Kit Instruction for details

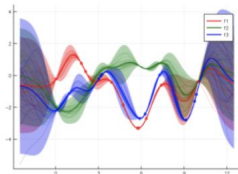
**Endogenous Sample Reading:** Healthy donor matched EDTA plasma (n=20), and serum (n=20) were measured. 13 CSF samples were measured. Bars depict median with interquartile range. Orange line represents functional LLOQ.



closer! [huggingface.co/docs](https://huggingface.co/docs)

# TRL Cards

Tool for communicating AI/ML technology readiness across all internal stakeholders.

TECHNOLOGY NAME		Solar Array Optimization v1.0	
TRL		7 <link to previous cards>	
R&D OWNER / REVIEWER		A. Lavin / G. Renard	
PROD OWNER / REVIEWER		S. Wozniak / S. Jobs	
COMPONENT CODES		1.1, 4.2, 4.3	
TL;DR	Applying our multivariate BayesOpt (MVBO) algorithm to the problem of solar panel configuration optimization, specifically towards client SolarUS.		
Data considerations	Two datasets have been used to train and validate the system: 1. Pilot dataset provided by SolarUS 2. Simulated datasets (which we derived from SolarUS data, w/ Gaussian noise); explores add'l geographic regions and climates		
Ethics	The datasets do not represent any biases. The algorithms have a very low carbon footprint. Augustus Ethics Checklist has been completed.		
Model / alg details		MVBO runs iterative optimization over several surrogate GP models f1...n, each representing an independently modulated portion of the array field.   <i>Example with n=3 (radians and hours on the x and y axes)</i>	
Metrics, results		MVBO algorithm converges to solution on opt. benchmark problems in ~1.0s on 4-core CPU. Full quantitative reports: <link to experiments wiki >	
Caveats, known edge cases, recommendations		For the solar array problem we require multi-objective optimization: maximize energy-gain objective while minimizing panel-movement, accomplished via Pareto front optimization. This was stable on 98.8% of simulated scenarios (full range of solar exposures).	
Key assumptions		We model solar radiance w/ simple Gaussian noise, and assume near-perfect actuation of solar panels.	
Intended use		Optimize up to 5 continuous or discrete parameters of a given device, and a system of up to 40 devices.	

The maturity of each model or algorithm is tracked via TRL cards. This is a card subset that reflects an example BO algorithm at TRL 7.

# TRL Cards

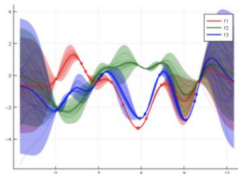
Tool for communicating AI/ML technology readiness across all internal stakeholders.

Enables inter-team and cross-functional communication.

Standardized “report cards” for TRL4ML stage reviews.

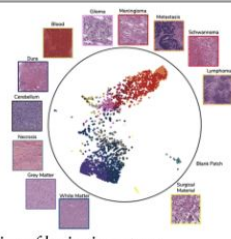
Lower-level and more process-oriented than other “ML cards” -- e.g. Google (Mitchell et al. ‘19) and Hugging Face.

Promotes ethics to first-class citizen.

TECHNOLOGY NAME		Solar Array Optimization v1.0		Model / alg details	MVBO runs iterative optimization over several surrogate GP models f1...n, each representing an independently modulated portion of the array field. 
TRL		7 <link to previous cards>			
R&D OWNER / REVIEWER		A. Lavin / G. Renard			
PROD OWNER / REVIEWER		S. Wozniak / S. Jobs			
COMPONENT CODES		1.1, 4.2, 4.3			
TL;DR	Applying our multivariate BayesOpt (MVBO) algorithm to the problem of solar panel configuration optimization, specifically towards client SolarUS.			Metrics, results	MVBO algorithm converges to solution on opt. benchmark problems in ~1.0s on 4-core CPU. Full quantitative reports: < link to experiments wiki >
Data considerations	Two datasets have been used to train and validate the system: 1. Pilot dataset provided by SolarUS 2. Simulated datasets (which we derived from SolarUS data, w/ Gaussian noise); explores add'l geographic regions and climates				
Ethics	The datasets do not represent any biases. The algorithms have a very low carbon footprint. Augustus Ethics Checklist has been completed.				
Caveats, known edge cases, recommendations		For the solar array problem we require multi-objective optimization: maximize energy-gain objective while minimizing panel-movement, accomplished via Pareto front optimization. This was stable on 98.8% of simulated scenarios (full range of solar exposures).			
Key assumptions		We model solar radiance w/ simple Gaussian noise, and assume near-perfect actuation of solar panels.			
Intended use		Optimize up to 5 continuous or discrete parameters of a given device, and a system of up to 40 devices.			

The maturity of each model or algorithm is tracked via TRL cards. This is a card subset that reflects an example BO algorithm at TRL 7.

## TRL Cards

TECHNOLOGY NAME		Neuropathology Copilot v1.0		Model / alg details	<p>The SP-VAE model runs unsupervised on neurological whole-slide images (WSI), producing a latent manifold that represents a hierarchical organization of tissue types. An medical expert identifies several data points to inspect.</p> <p><i>Example visualization of the latent organization of brain tissue types.</i></p> 
TRL		4 <link to previous cards>			
R&D OWNER / REVIEWER		A. Lavin / G. Renard			
PROD OWNER / REVIEWER		S. Wozniak / S. Jobs			
COMPONENT CODES		1.1, 4.2, 4.3			
TL;DR	Analyze WSI of brain tissue in 3 main steps: (1) unsupervised CV model produces Poincare manifold viz (Naud & Lavin ‘20), (2) domain expert selects data points, (3) U-Net classifier				
Data considerations	3 datasets have been used to train and validate the system: <ul style="list-style-type: none"><li>1. Open dataset (Naud &amp; Lavin ‘20)</li><li>2. Pilot dataset provided by BioLab, v1.0</li><li>3. Simulated datasets (w/ structured domain randomization), v2.3</li></ul>				
Ethics	Note the demographics info on specific Dataset Cards. Datasets anonymized, pipeline runs w/o metadata. The Latent Sciences Ethics Checklist has been completed.				
					</

A subset of a level 4 TRL card used in a medical AI project. Notice the ethics section, which refers the team to a company-specific ethics checklist. TRL4ML operationalizes ethics (not just in AI, also the relevant fields/domains of use) by requiring a formal checklist, and making ethics explicit in gated reviews and deliverables.

# TRL Cards @ Immunai

What would be most useful / practical / frictionless for you?

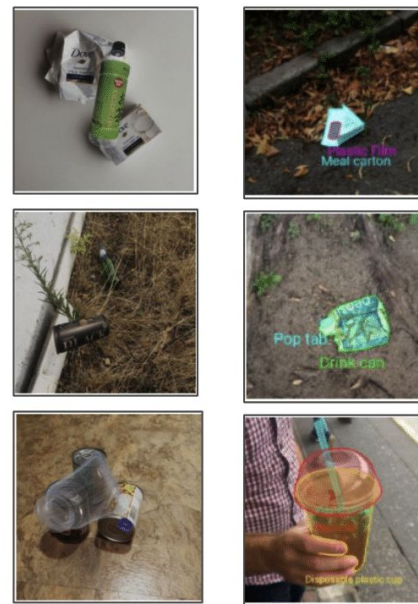
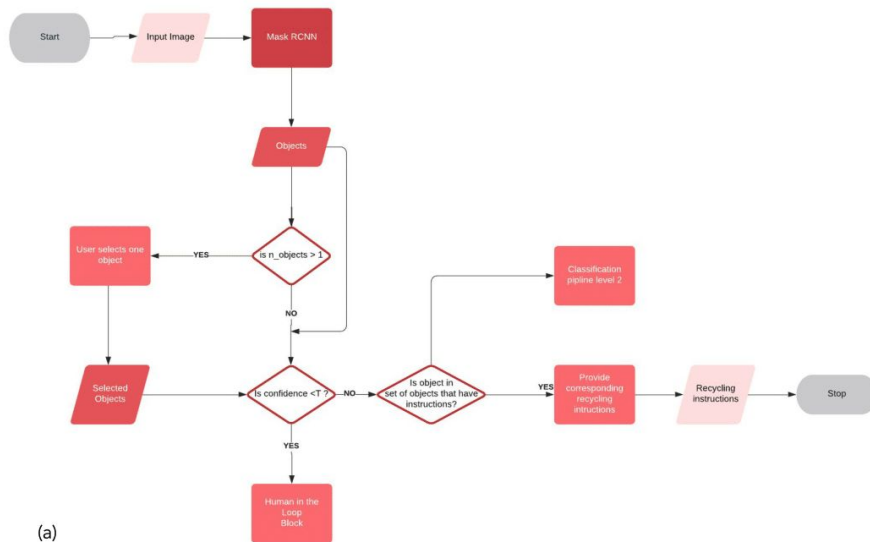
- What format? (e.g., generic template that's implemented in internal wikis, markdown on github, ...)
- If programmatic, how and why?
- Where would it fall short?
- How to build community best-practice / standards?
- Relative to TRL Cards, what is unique about AI in medicine/biotech that are critical to consider?

# Outline

1. Setting the scene
2. Systems Engineering and AI
3. MLTRL
- 4. Examples**
5. Takeaways

# Computer vision app with real and synthetic data

Recycling classification pipeline





## Systems AI challenges for this example

## Mitigation mechanisms in MLTRL

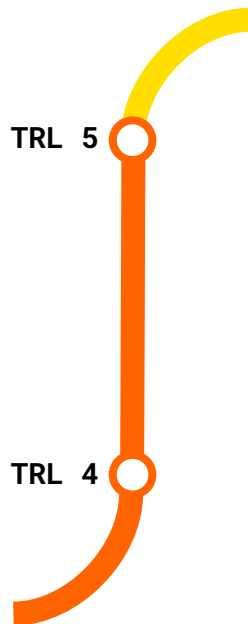
Multiple and disparate data sources	Prescribed data versioning and monitoring tests
Hidden performance degradation	Focused stress tests
Transitioning sim-to-real gap	Evolving from tech V&V to product V&V
Complex interacting systems: simulation engine, human feedback, etc.	Tests of course, but also multi-functional working groups
ML pipelines typically grow out of glue code	Code-caliber checkpoints, explicit infra stages

## Example “productization handoff”

Working group evolves to become more cross-functional as the tech matures.

TRL 4 to 5, we’re transitioning the model or algorithm from an isolated solution to a module of a larger application.

Graduation from level 5 is difficult, signifying the dedication of resources to push the ML technology through productization.



Example: Level 5 review of tech V&V and new product requirements identified the need for a different **data-oriented architecture**... Done in the subsequent level, where MLTRL already prescribes a software refactoring for productization.

Note: TRL 2-4 is driven by formal tech req’s and V&V. TRL 5 transitions to product-driven req’s and V&V.

# Up to snuff? MLTRL with NASA & ESA

## Emphasis on AI reliability

Generate local satellite images of future climate scenarios (right).

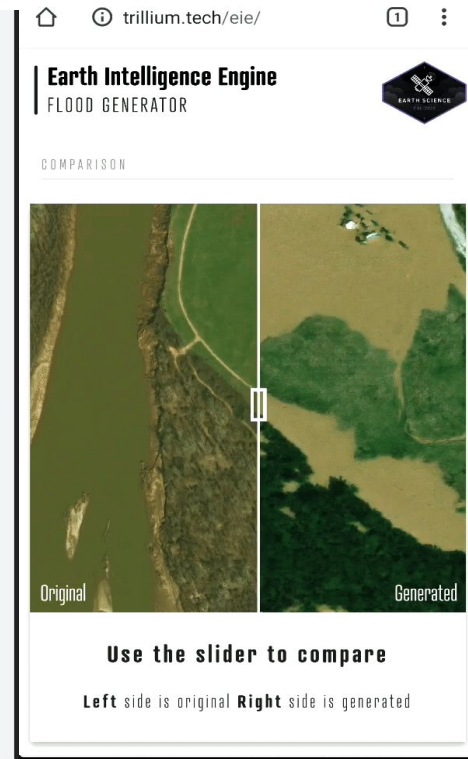
- Stakeholders from NASA, Google, MIT, NOAA, Portugal gov't.
- *Needs to be usable for our decision-makers*
- *How can we know results are reliable?*

PhiSat-1 is first European satellite with onboard AI.

- *Needs to integrate with several variations of hardware and sensors*
- *Needs full tech-transfer to ESA*

Explore the flood-simulation app at [trillium.tech/eie](https://trillium.tech/eie)

And more examples in Ganju et al (2020). *Learnings from the AI Accelerators for NASA and ESA*: [arxiv.org/abs/2011.04776](https://arxiv.org/abs/2011.04776)



# Outline

1. Setting the scene
2. Systems Engineering and AI
3. MLTRL
4. Example
5. **Takeaways**

## Take-home messages

ML != SW

Systems approach to AI is much-needed.

- Beyond one-off models, AI tech is built for **complex, dynamic systems (data + software + hardware + humans)**
- Industry lacks principled processes for robust, reliable, responsible AI/ML
- Interdisciplinary projects are exceptionally challenging

MLTRL is an industry-proven systems engineering framework, designed for efficient yet robust, reliable, and responsible AI/ML research, productization, and deployment.

MLTRL uniquely provides a holistic perspective, **lingua franca**, stakeholder-alignment, metrics and deliverables.

# Thank You!

Please reach out:

[lavin@simulation.science](mailto:lavin@simulation.science)

[@AlexLavin\\_C137](#)