Testing Self-Learning Systems

(for autonomous cars)

Stuart Reid PhD, FBCS

(stureid.test@gmail.com / www.stureid.info)

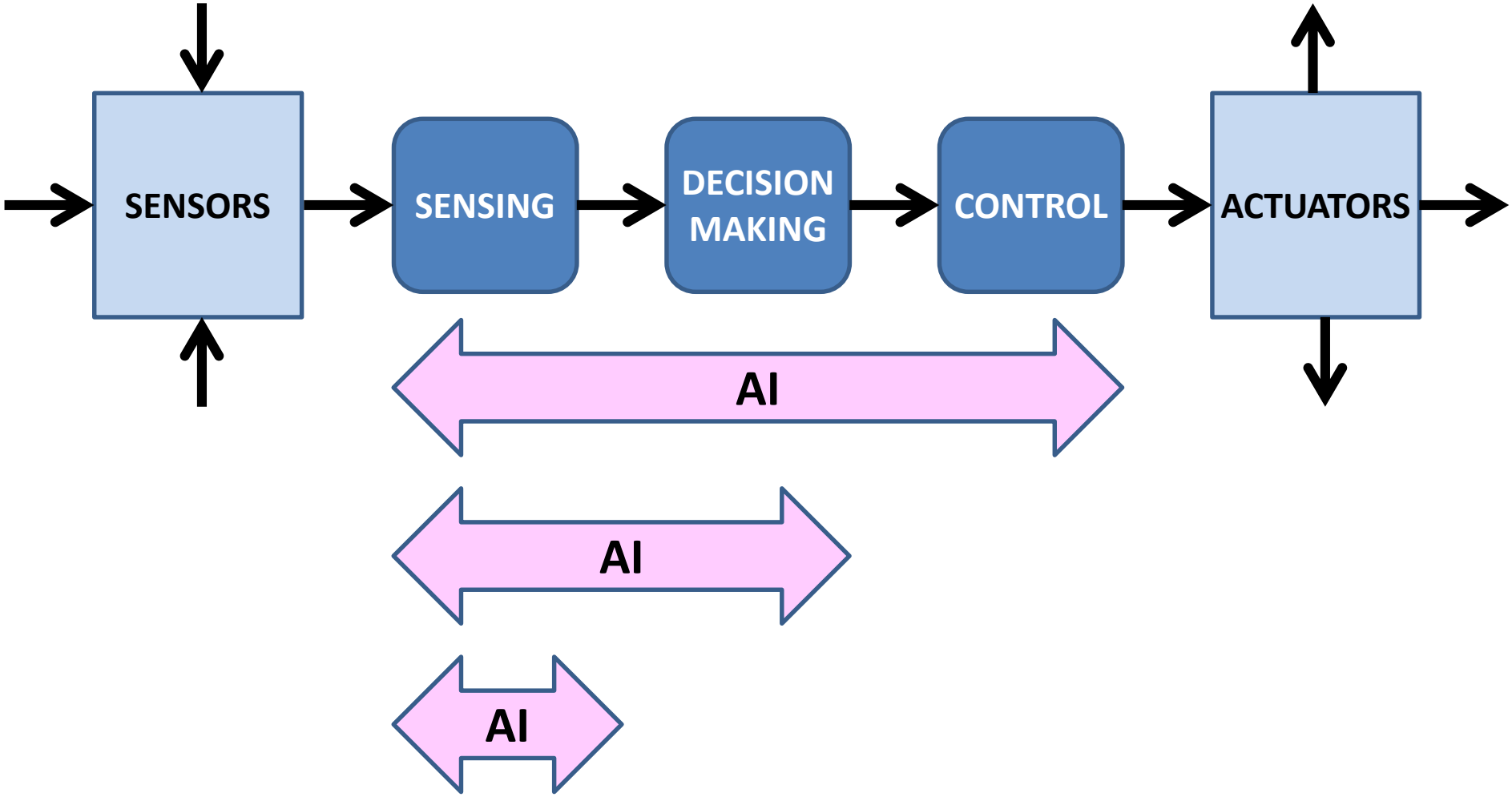
Is this a good training set?

Scope of the Talk

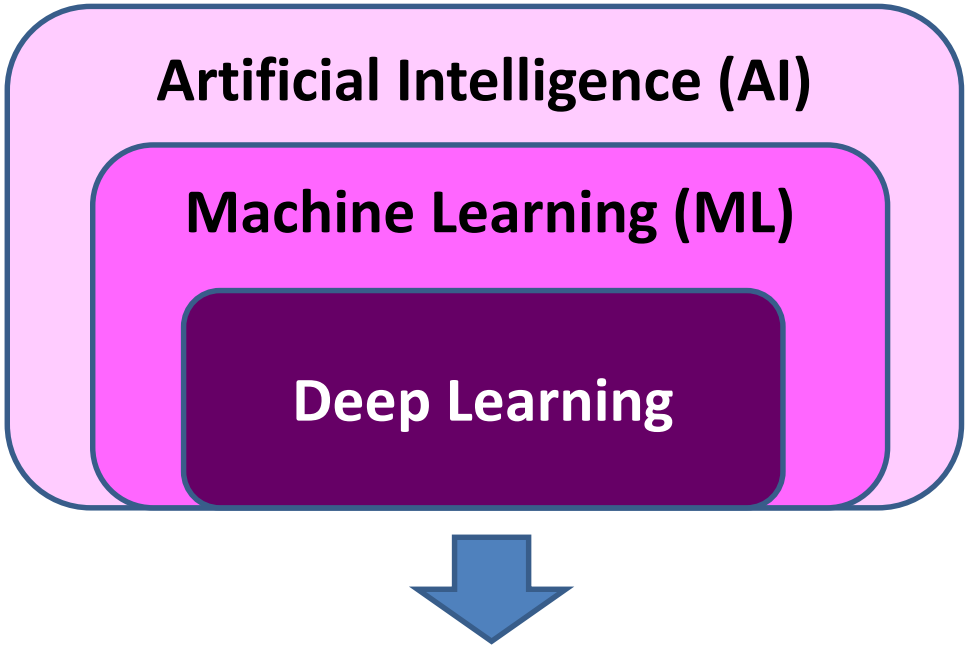
- Self-Learning Systems & Autonomous Cars
- Machine Learning Challenges & Test Opportunities
- Black Box Testing of Neural Networks
- White Box Testing of Neural Networks
- The Necessity of Virtual Test Environments
- Conclusions

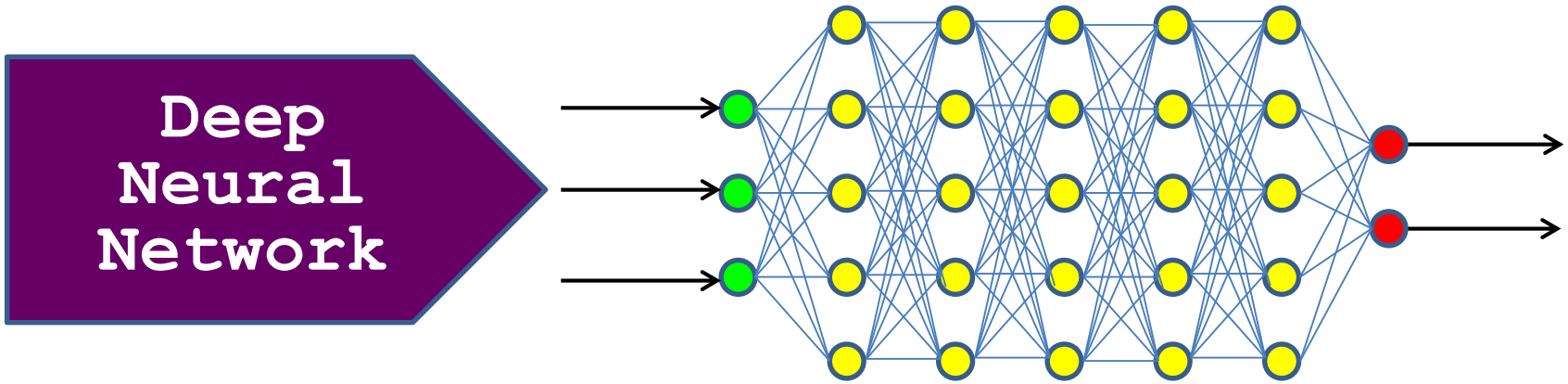
Self-Learning Systems & Autonomous Cars

Basic Autonomous Car Framework

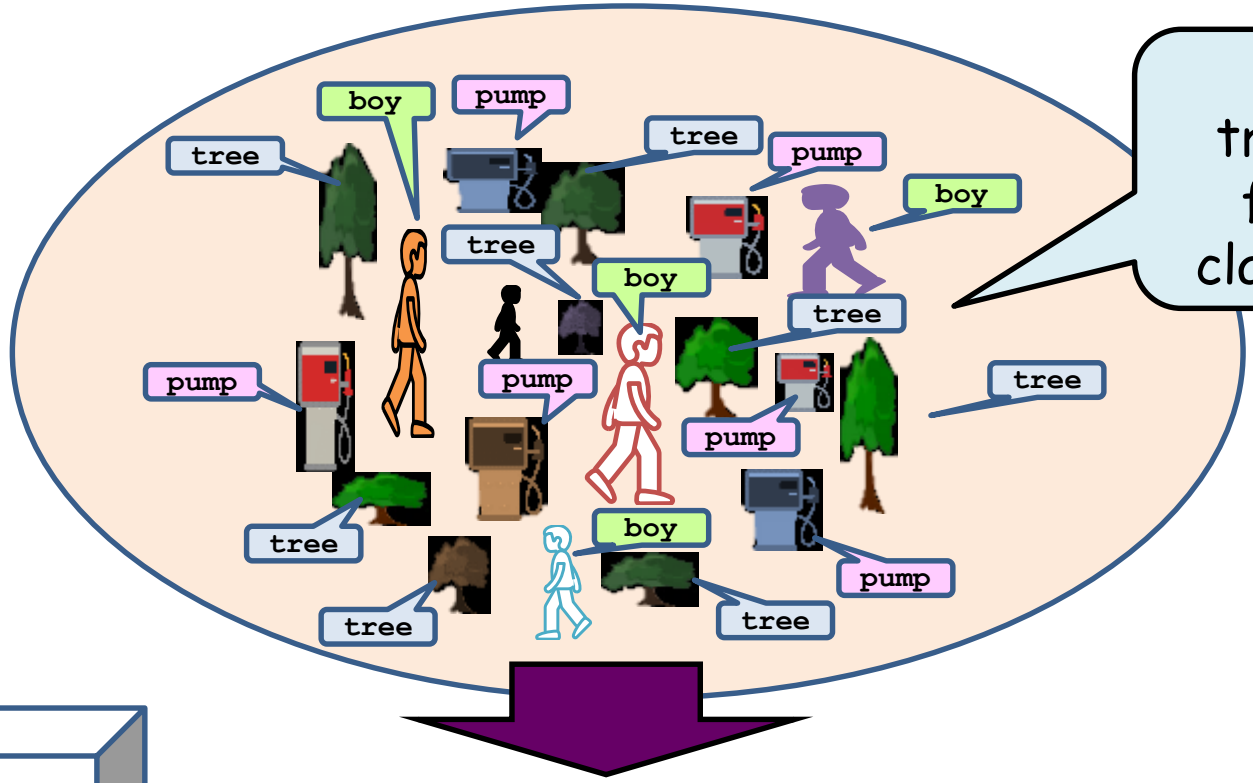


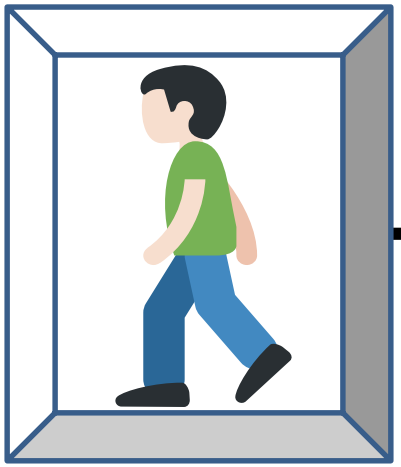
Deep Learning Systems





Example of Machine Learning

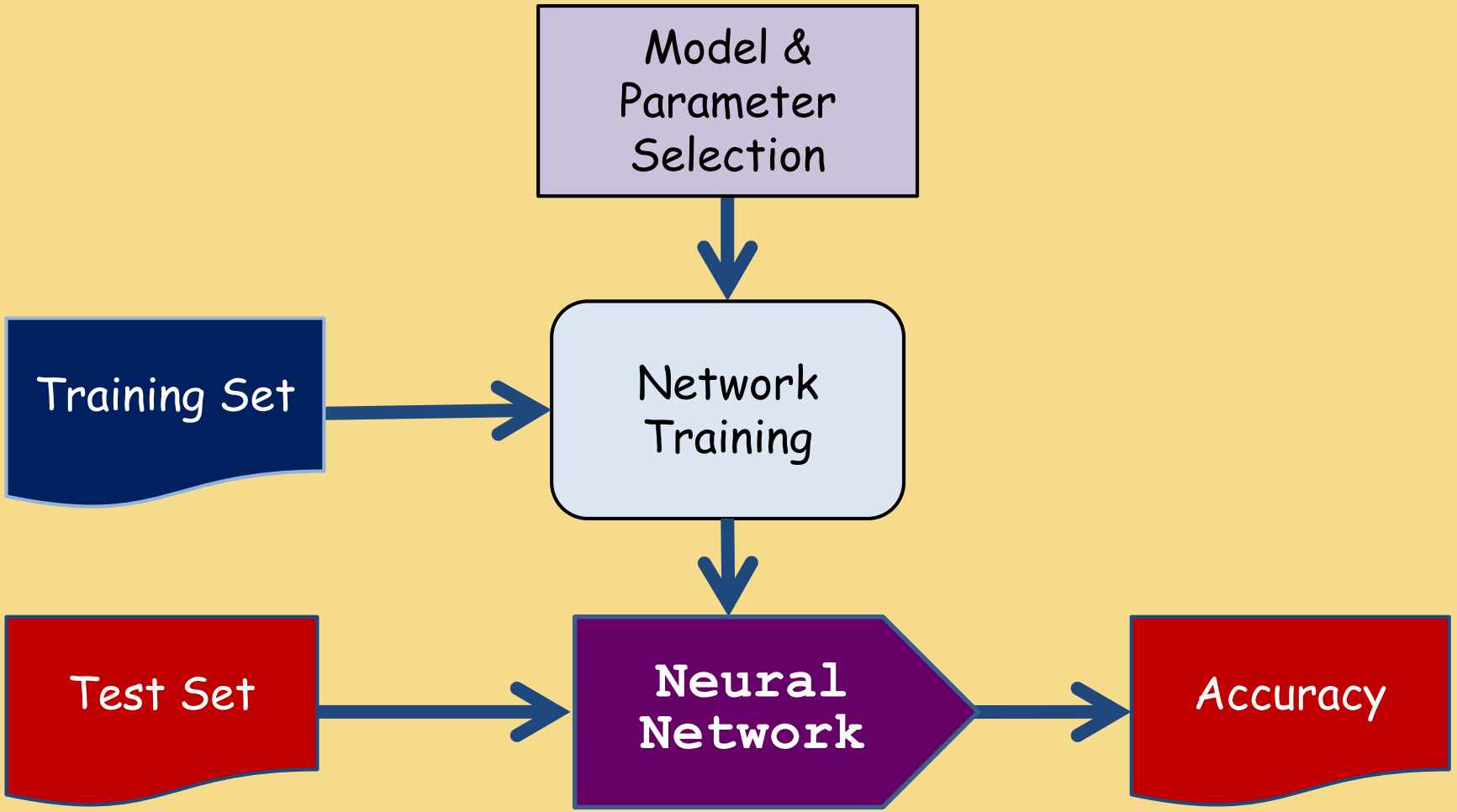




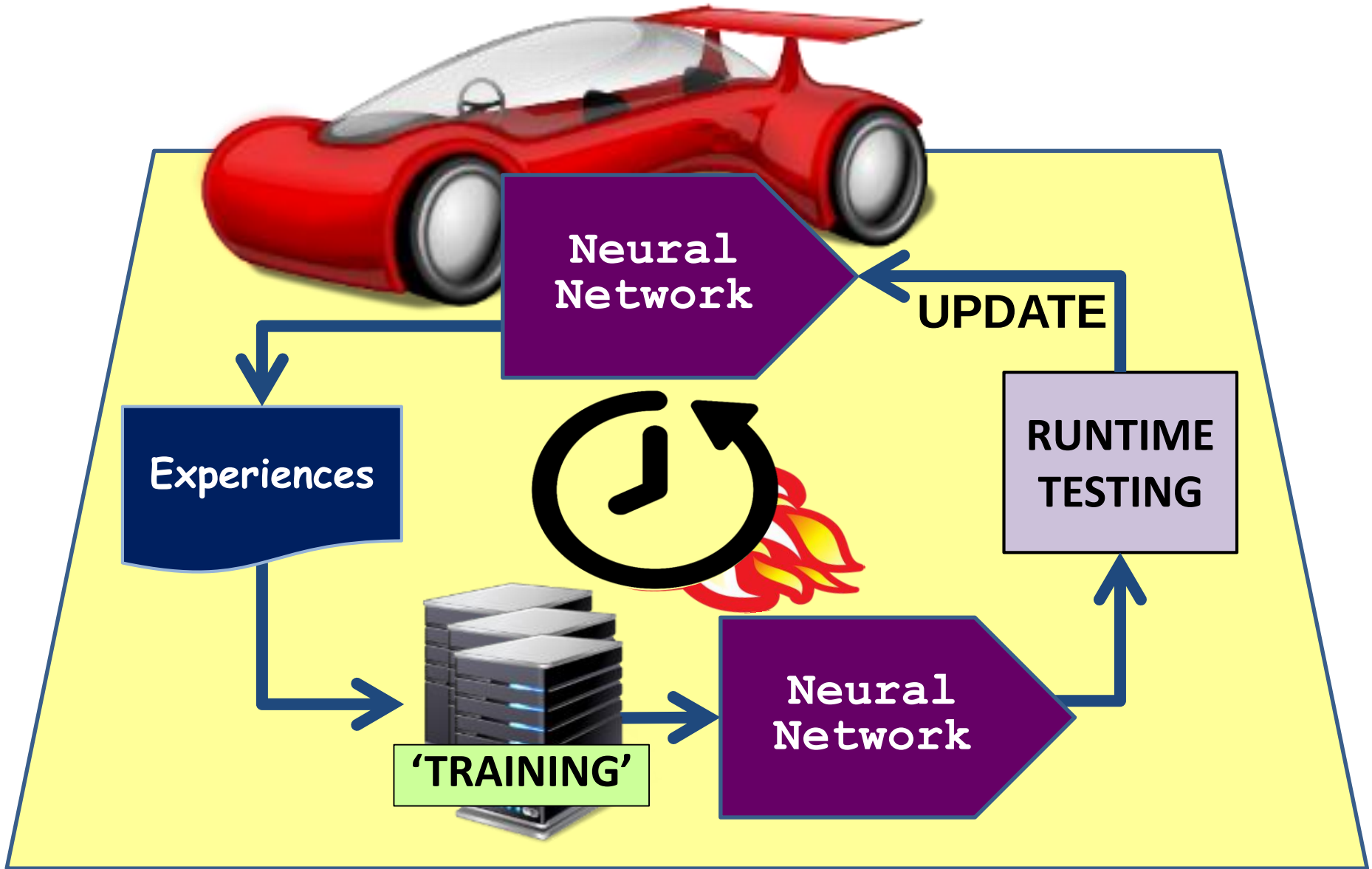
Neural Network

Boy

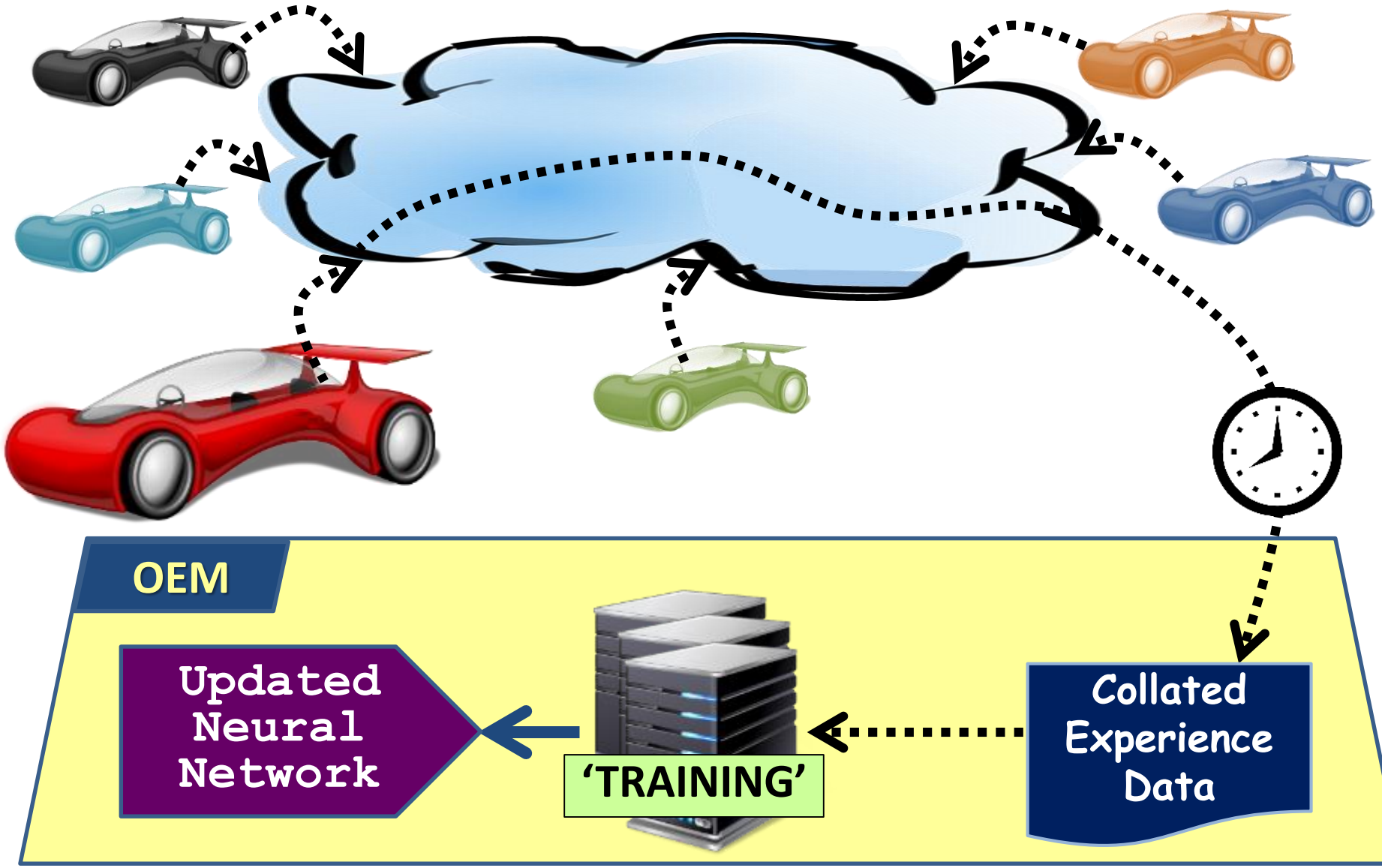
Supervised Machine Learning



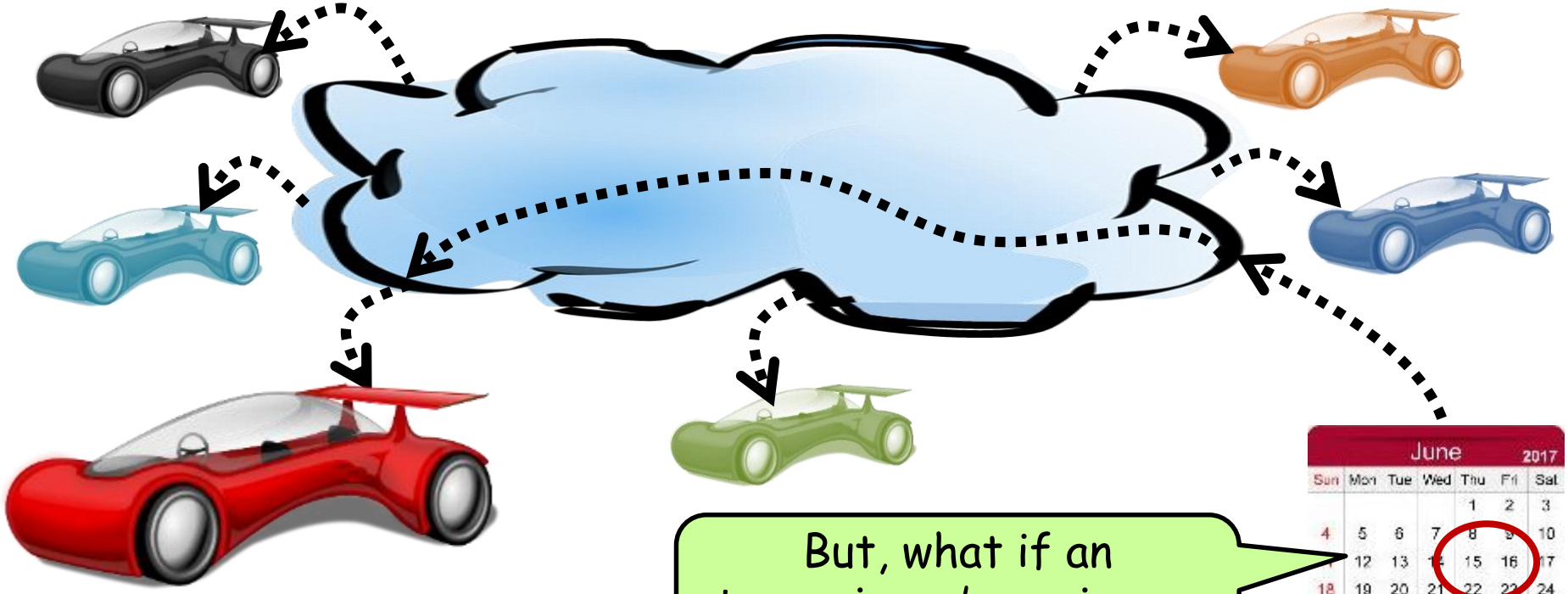
Continuous Online Learning



Off-Line Learning – from Day-to-Day Use



Performance Updates - Over-The-Air



But, what if an 'experience' requires a quick fix?

OEM

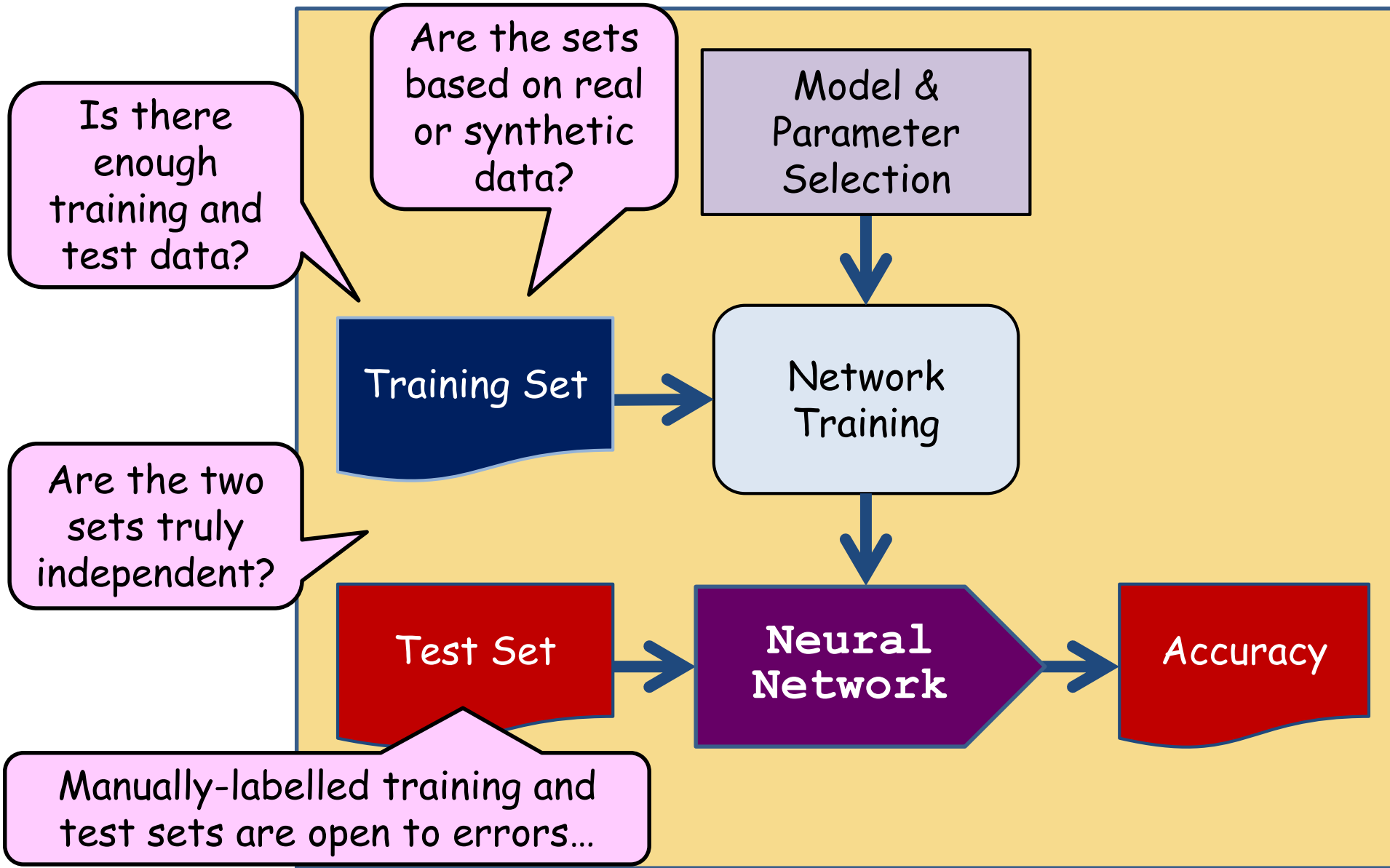
Updated Neural Network

OFFLINE TESTING

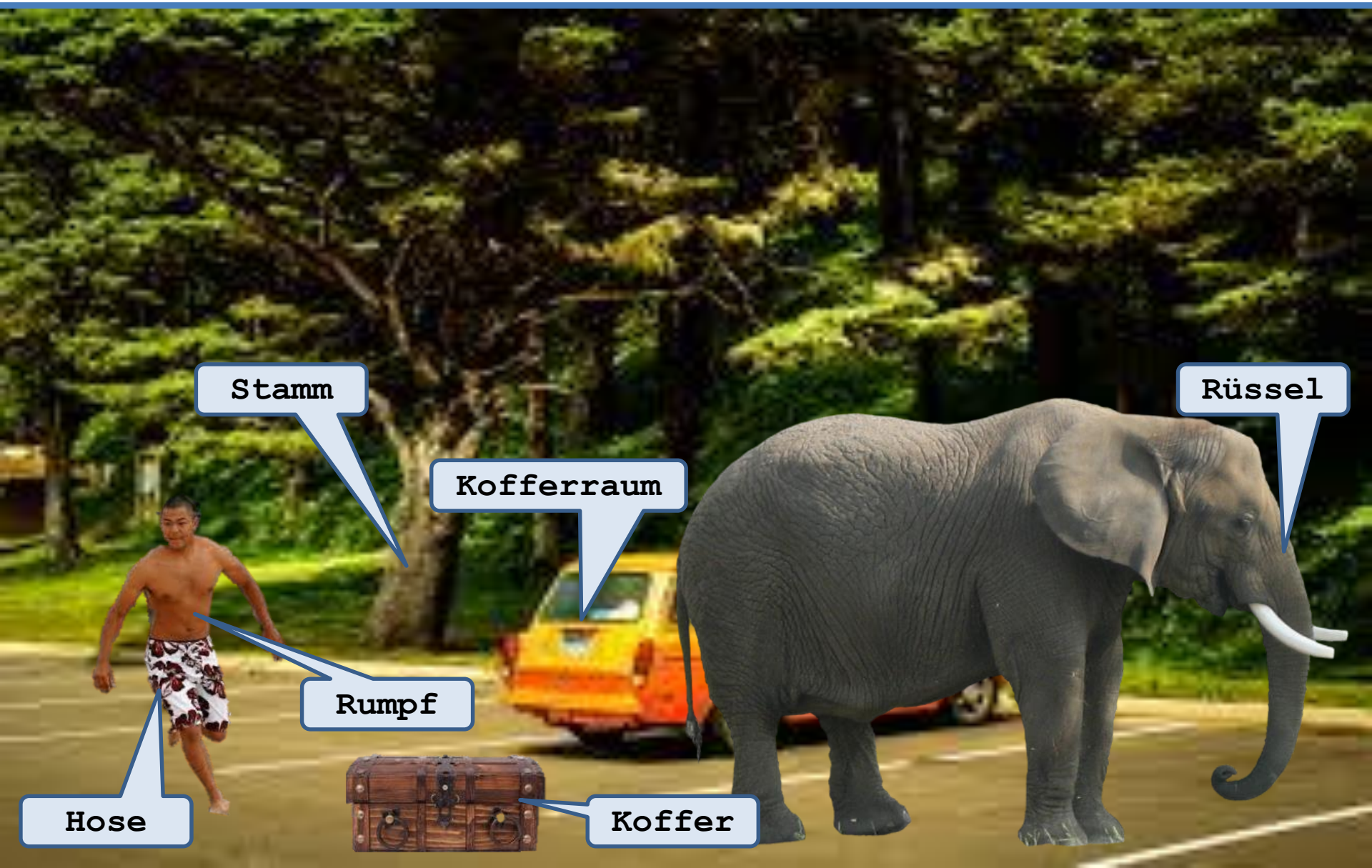
UPDATE

Machine Learning Challenges & Test Opportunities

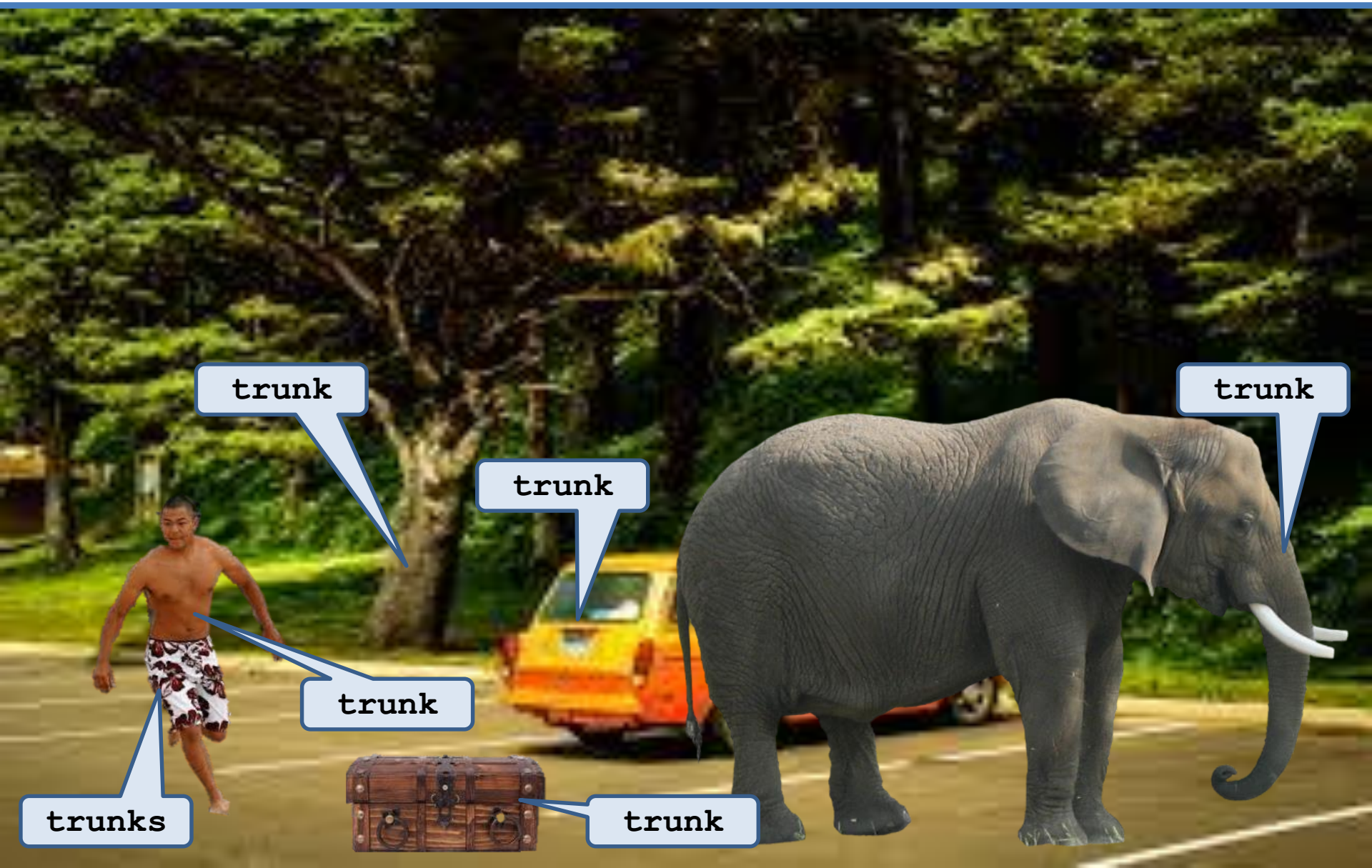
Checking the Training & Test Sets



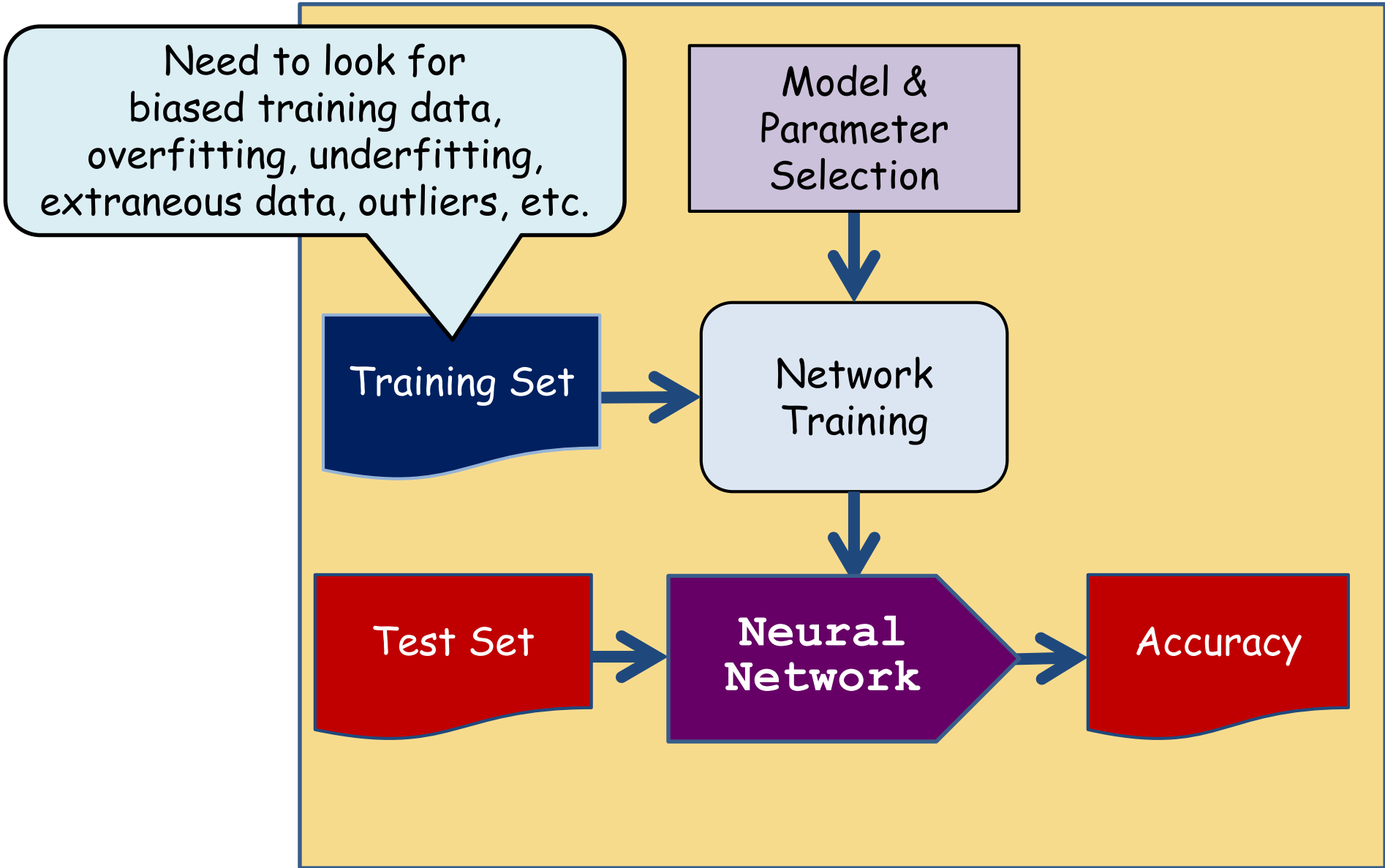
Mis-Classification



Mis-Classification



Checking the Training Set

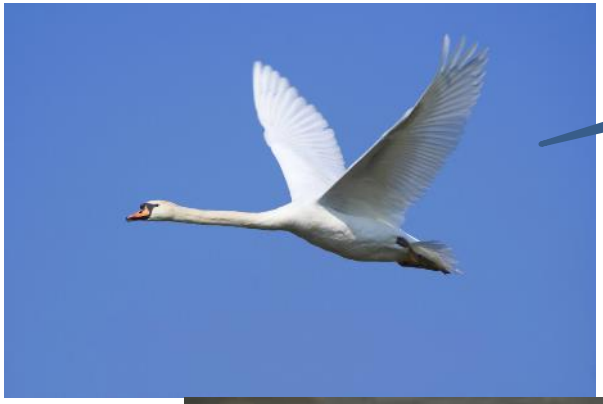


Misunderstanding – Data Bias

Neural Network

Not a tank

Incomplete Training Set

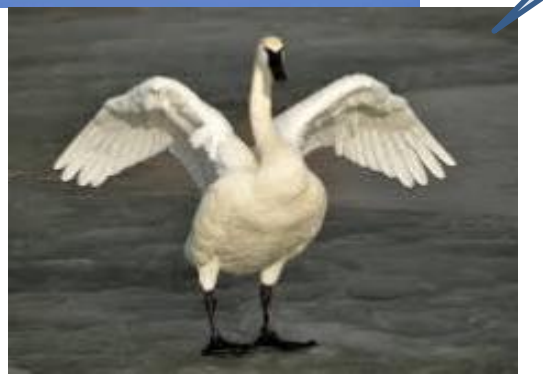


swan

swan

swan

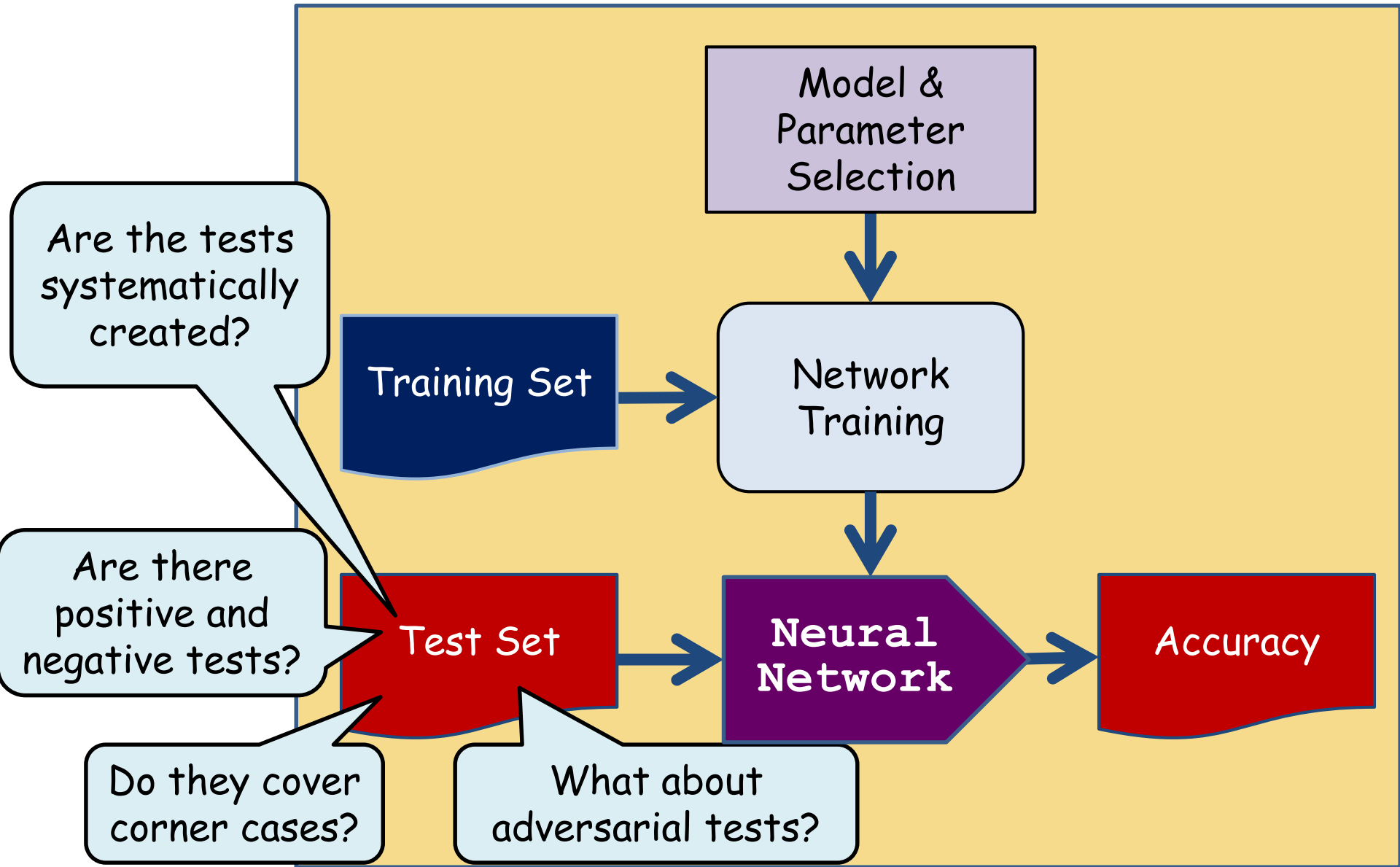
swan



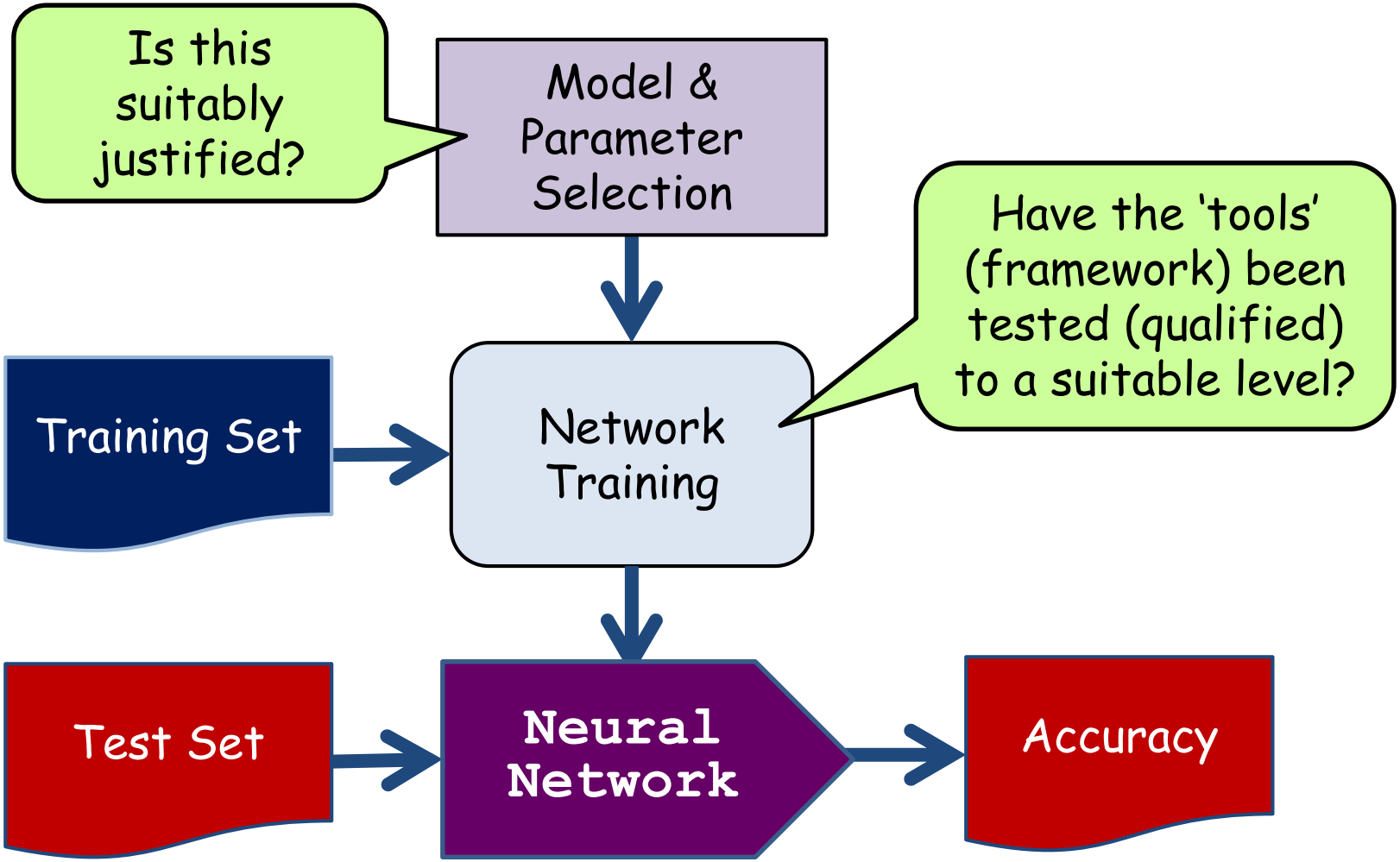
Neural Network

Not a swan

Checking the Test Set



Checking the Training

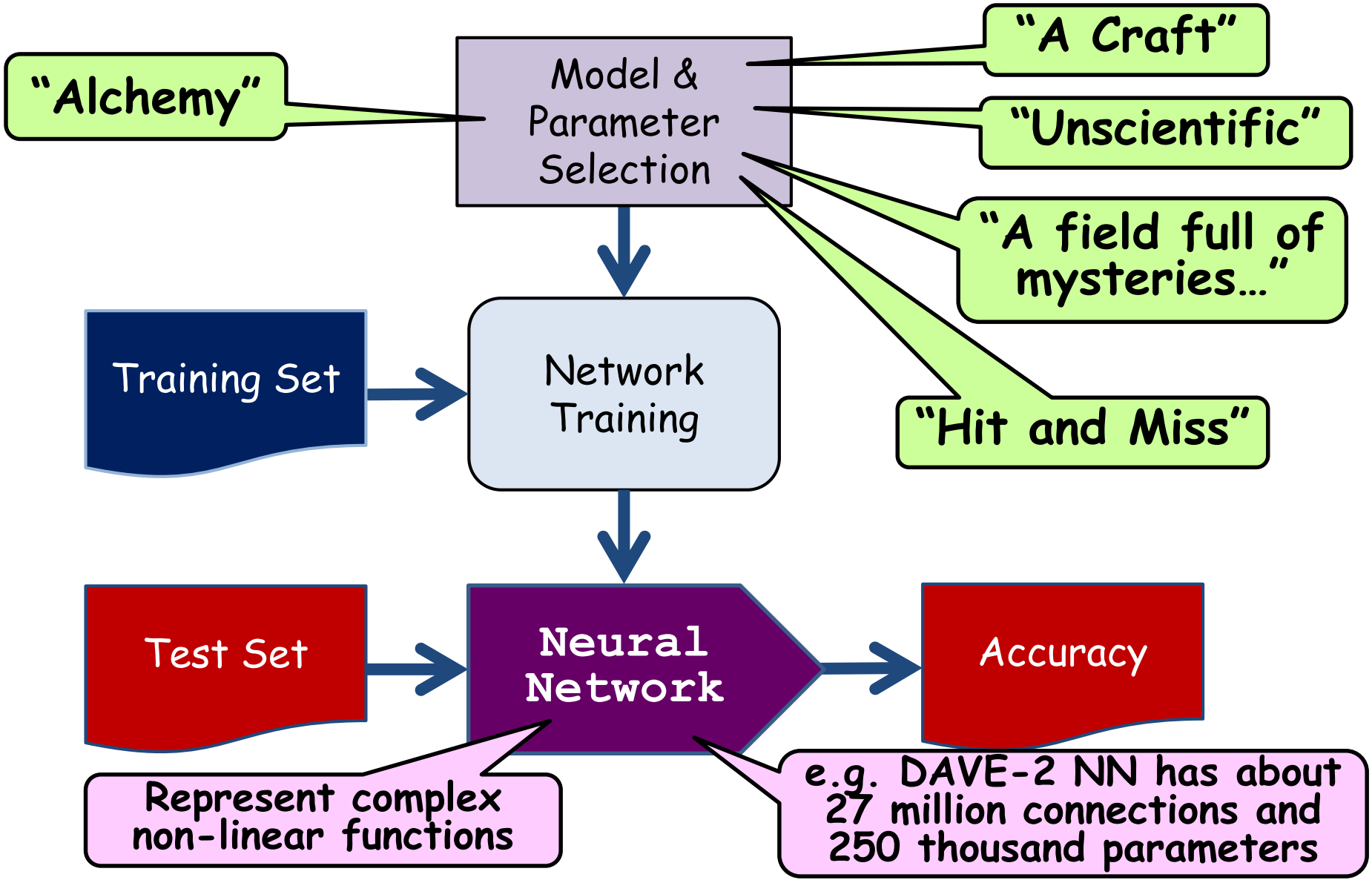


Black Box Testing of DNNs

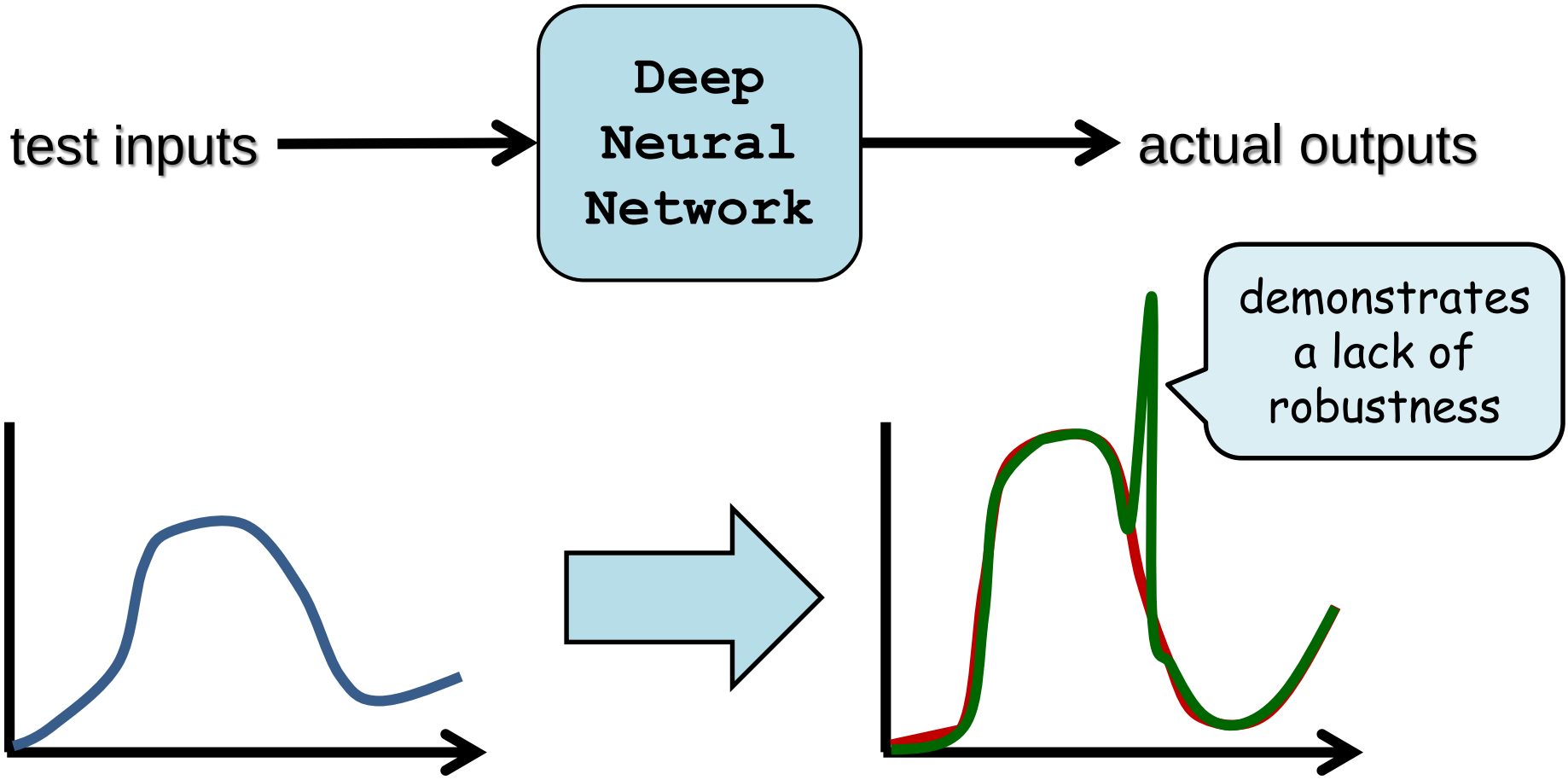
Test Challenges of Self-Learning Systems

- **Complexity and Test Oracle**
  - Often we cannot define required behaviour, e.g. how we…
    - recognize particular objects (e.g. child, paper bag, bike, obstruction)
    - 'know' that another car will move in a particular direction
    - plan a manoeuvre into moving traffic
    - know what to do in a new situation
- **Probabilistic Systems and Non-Determinism**
  - The probabilistic nature means that predicting expected results is difficult
  - Non-determinism causes real problems for regression testing
- **Self-Learning Systems are difficult to test because they are difficult to understand (not just be testers!)…**

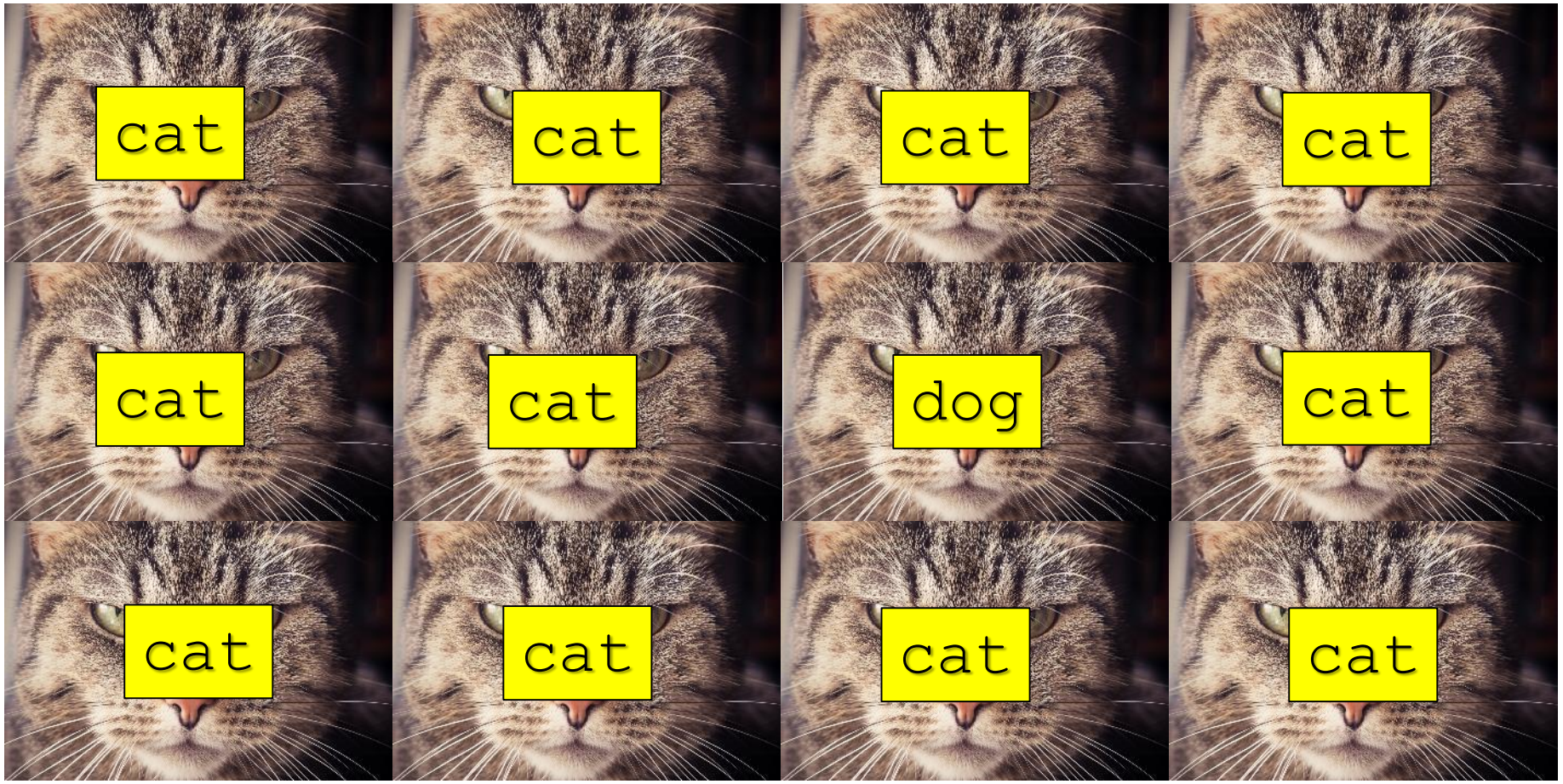
ML – Science and/or Engineering?



Adversarial Defects



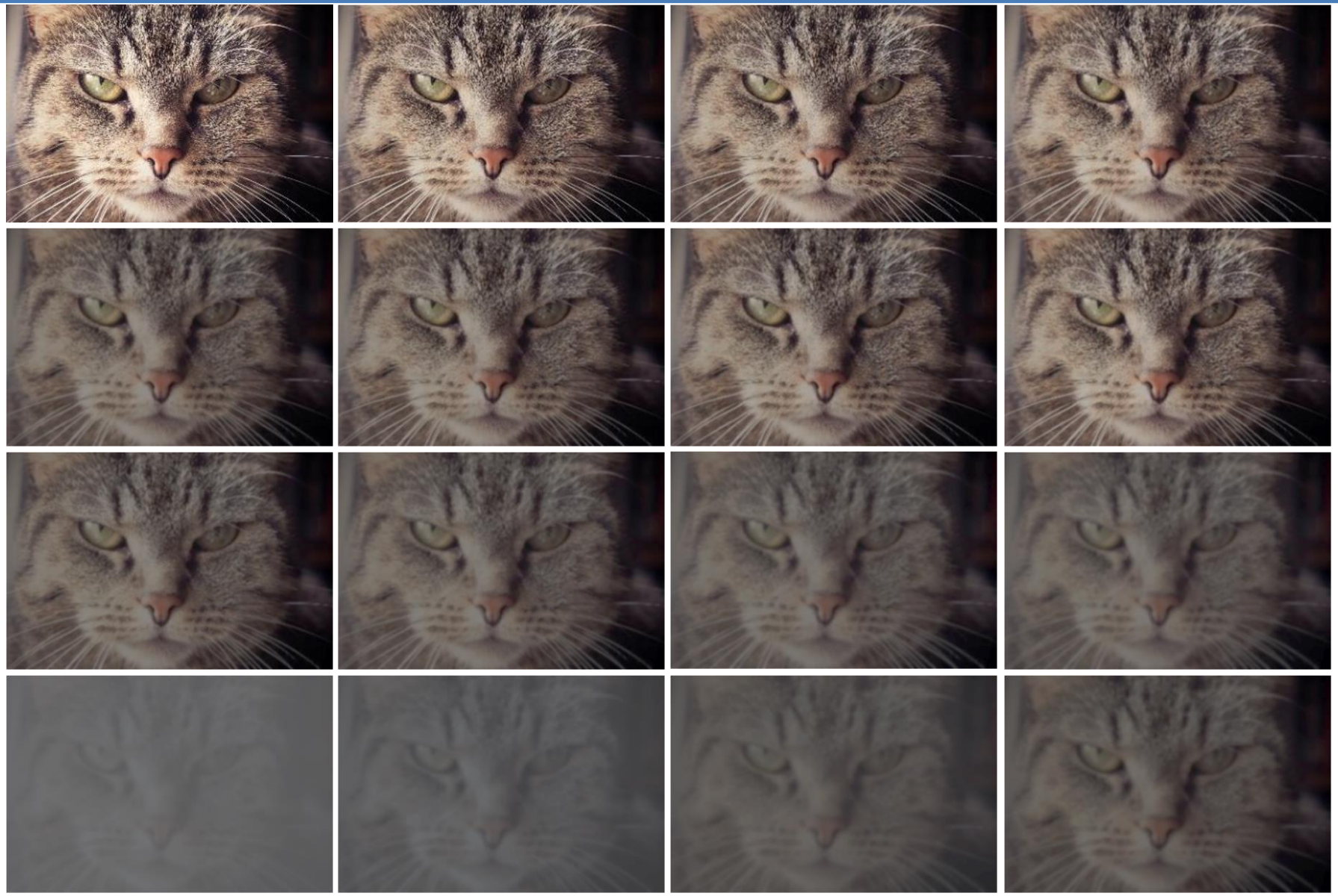
Adversarial Perturbations



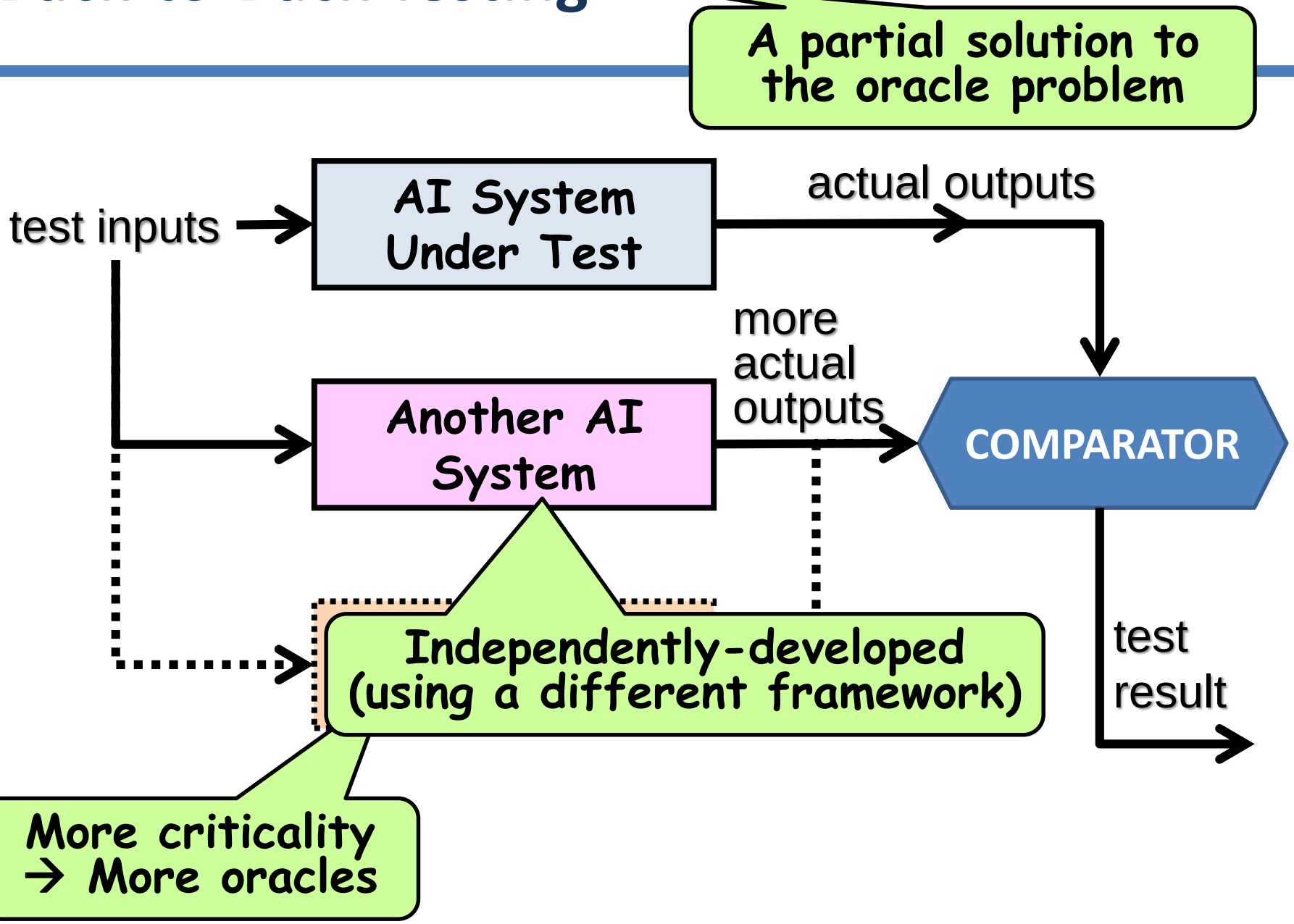
Combinatorial Testing Problems

- **The number of inputs to an autonomous car is extremely high**
  - external sensor inputs (e.g. cameras, radar, lidar, etc.)
  - V2V communications / other external communications
  - internal information from the vehicle (e.g. engine)
  - map data
  - etc.
- **Even simple pairwise testing could generate millions of tests**

Example - Sensor Degradation Testing

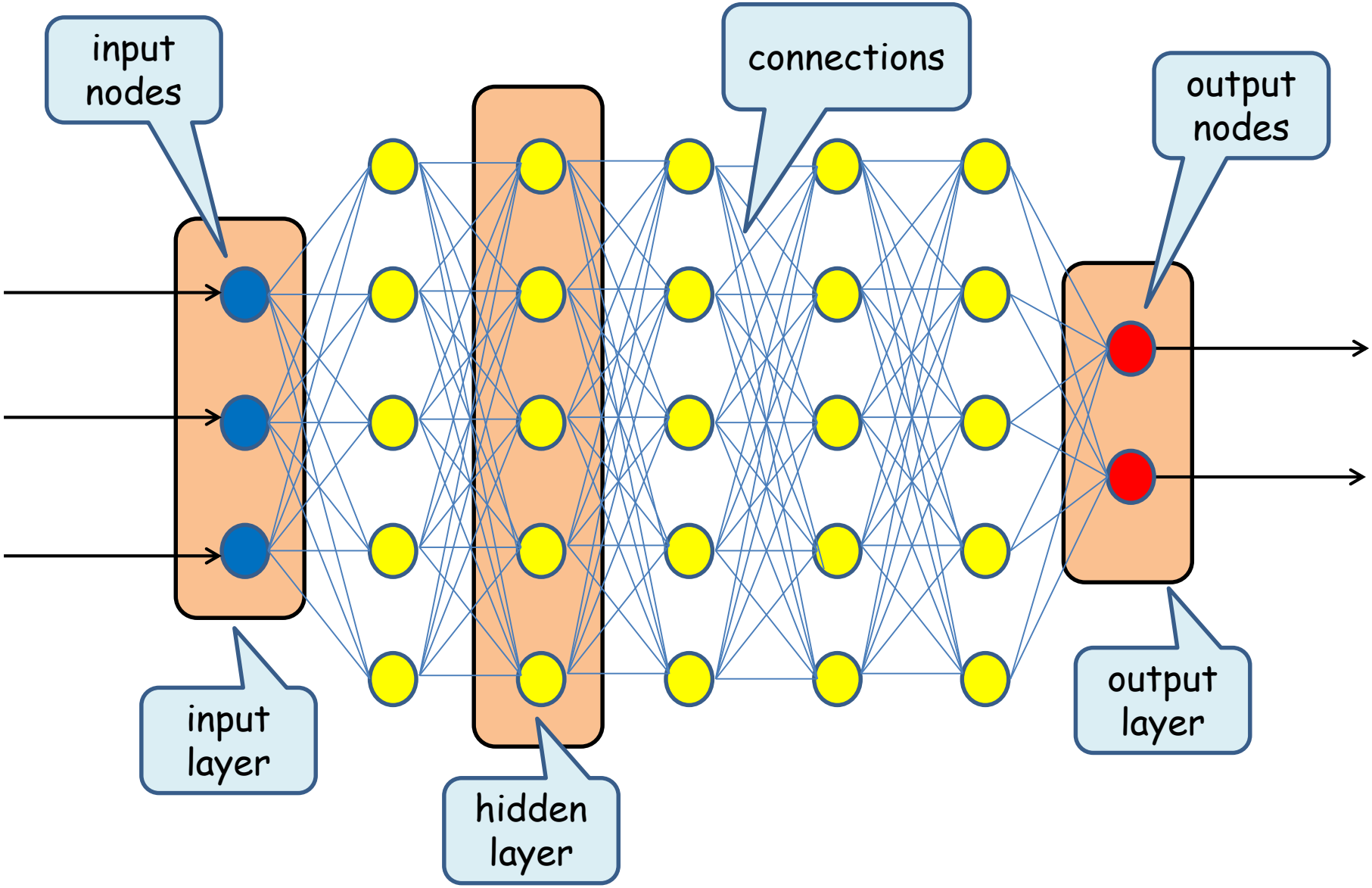


Back-to-Back Testing

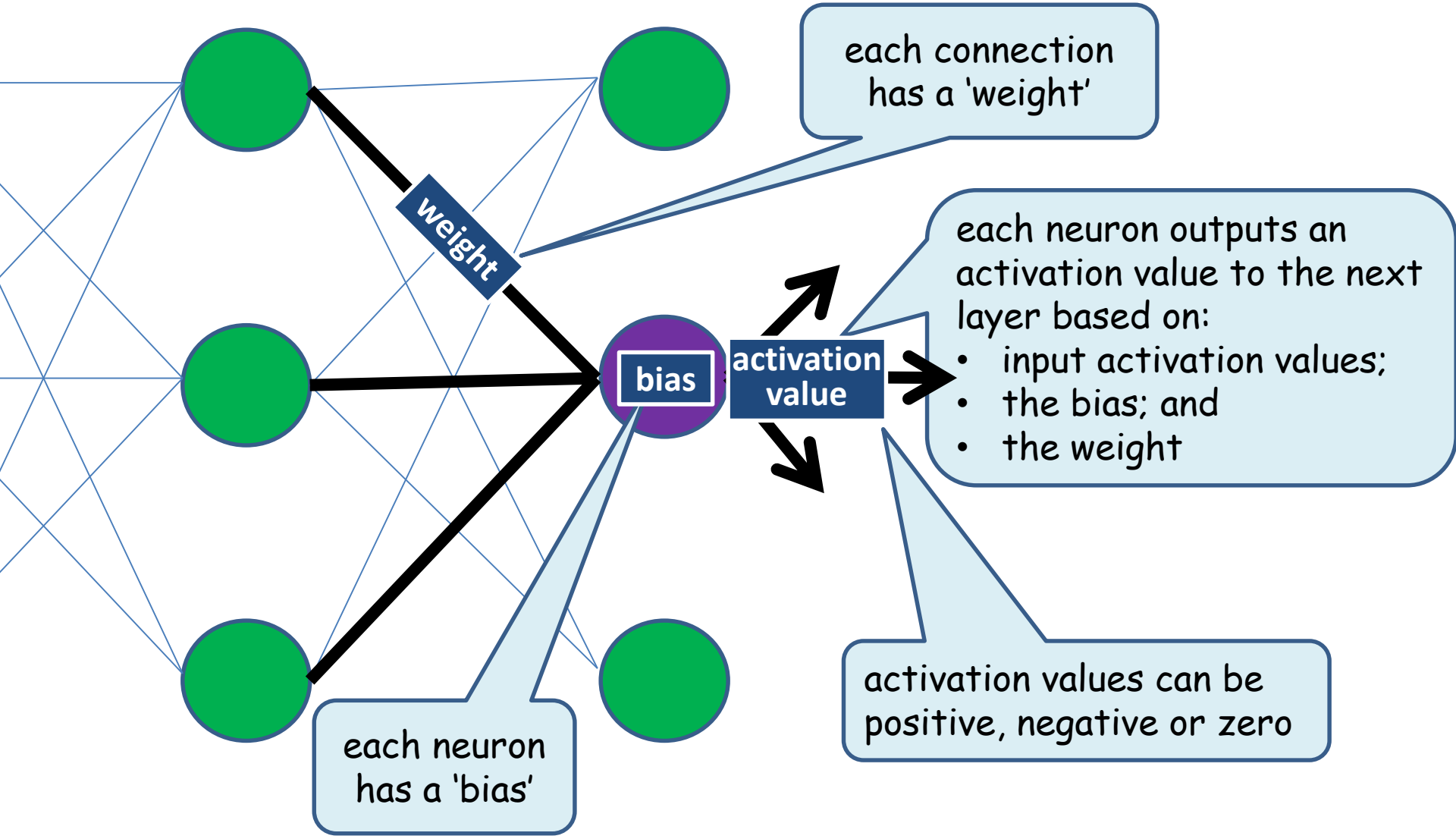


White Box Testing of DNNs

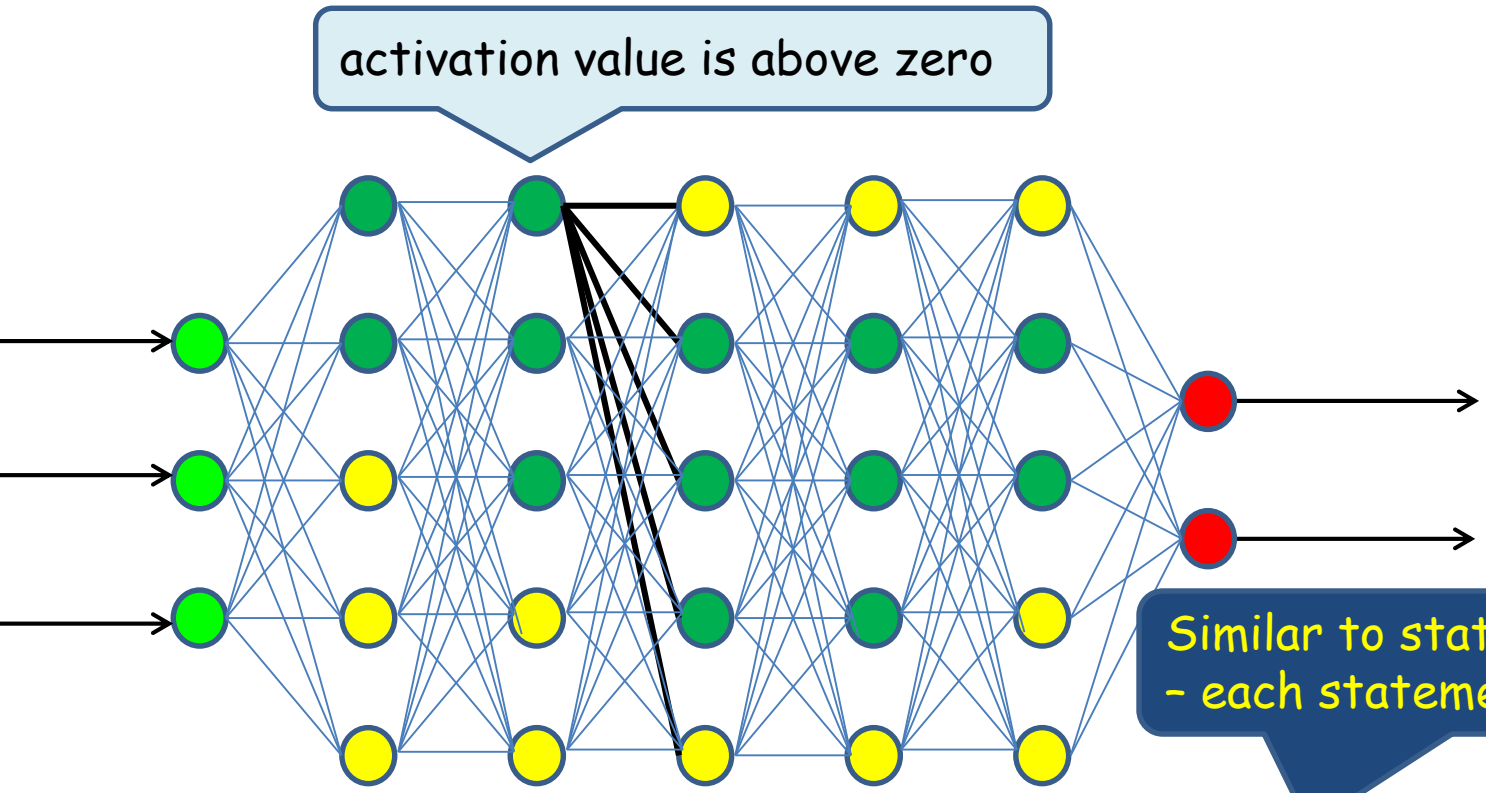
Deep Neural Net



Activation Values



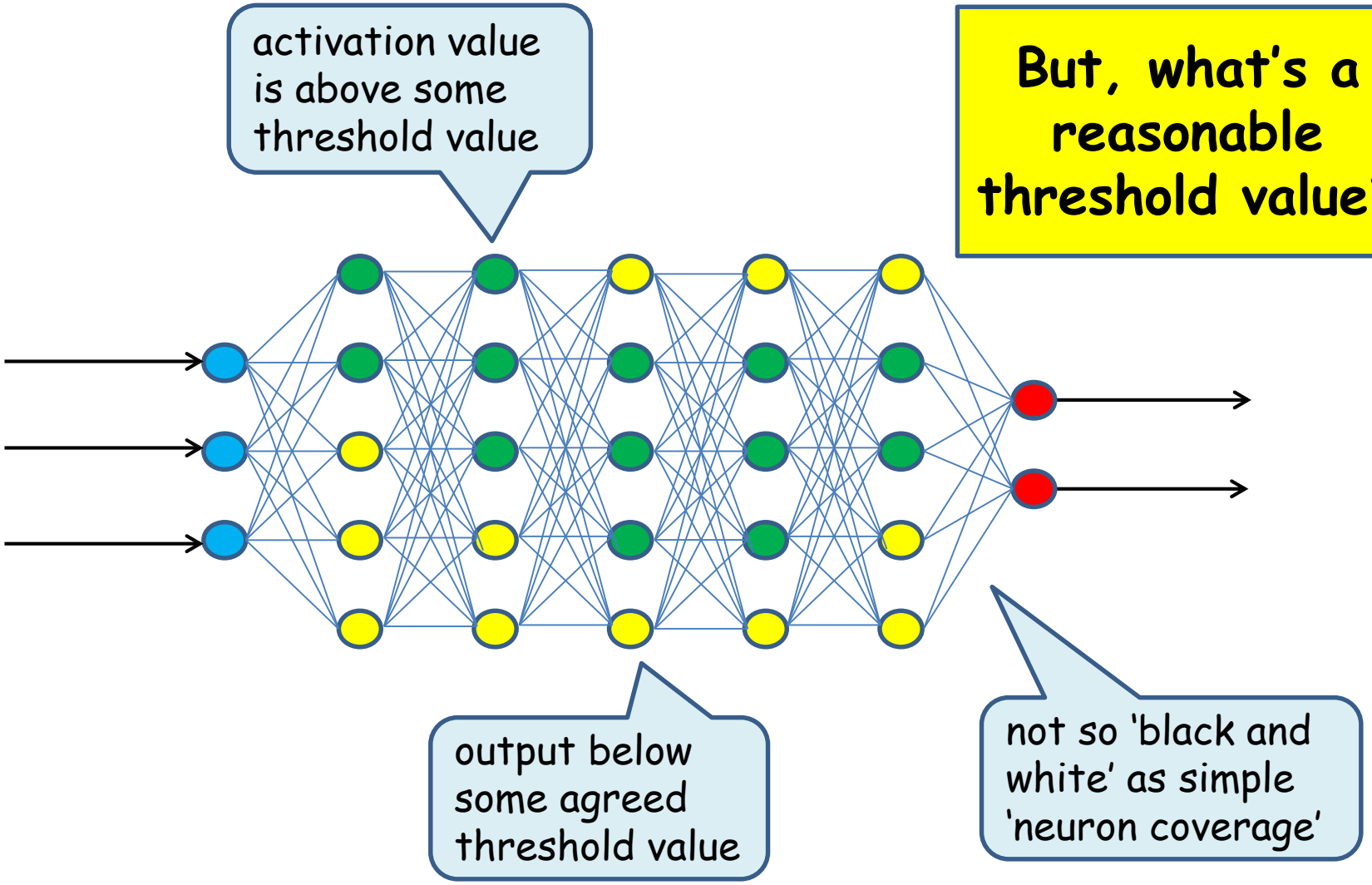
‘Neuron’ Coverage



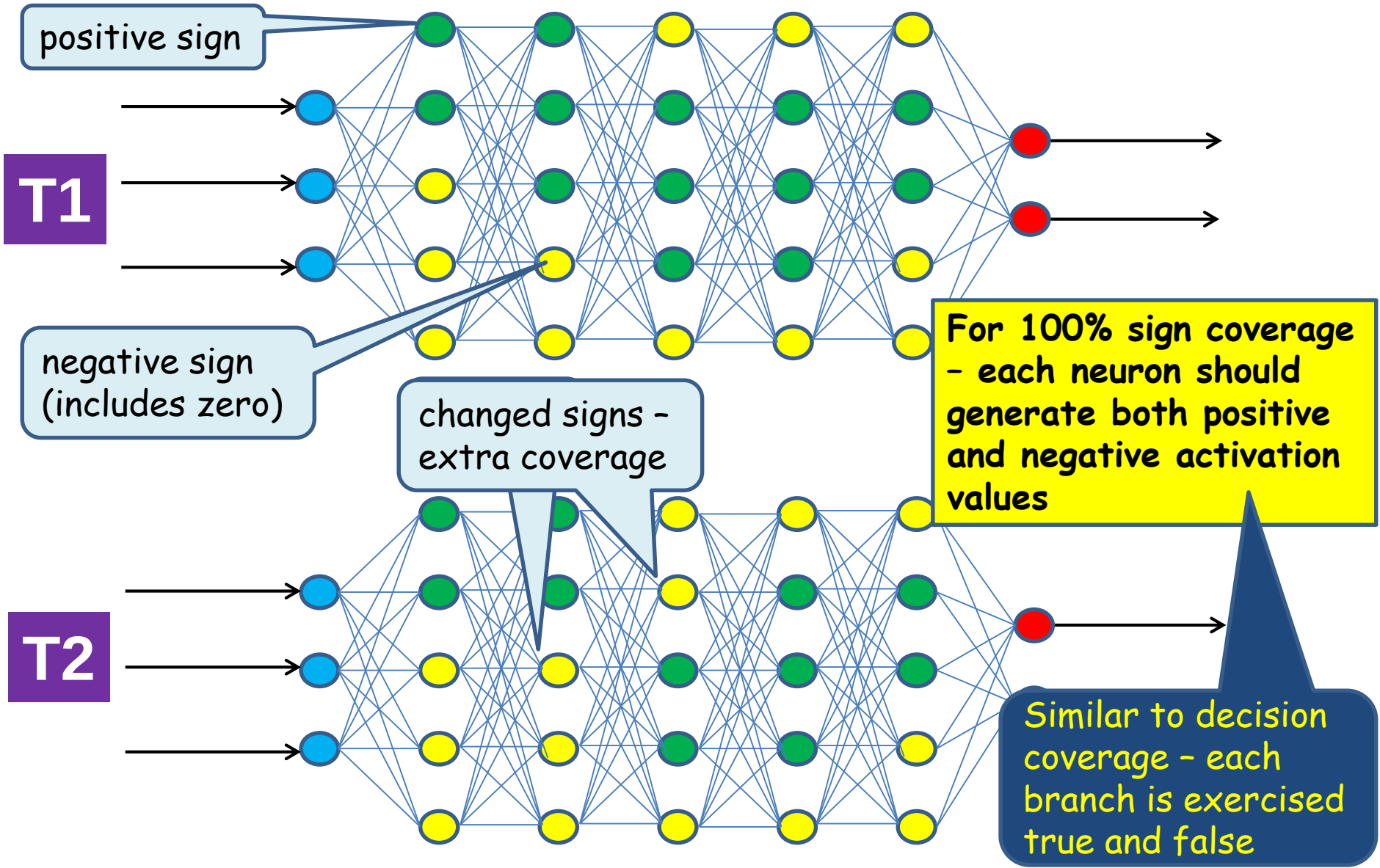
Similar to statement coverage – each statement is exercised

Full ‘neuron’ coverage shows that every neuron is ‘activated’ (value above zero) at least once (but - basic coverage - typically finds no adversarial examples)

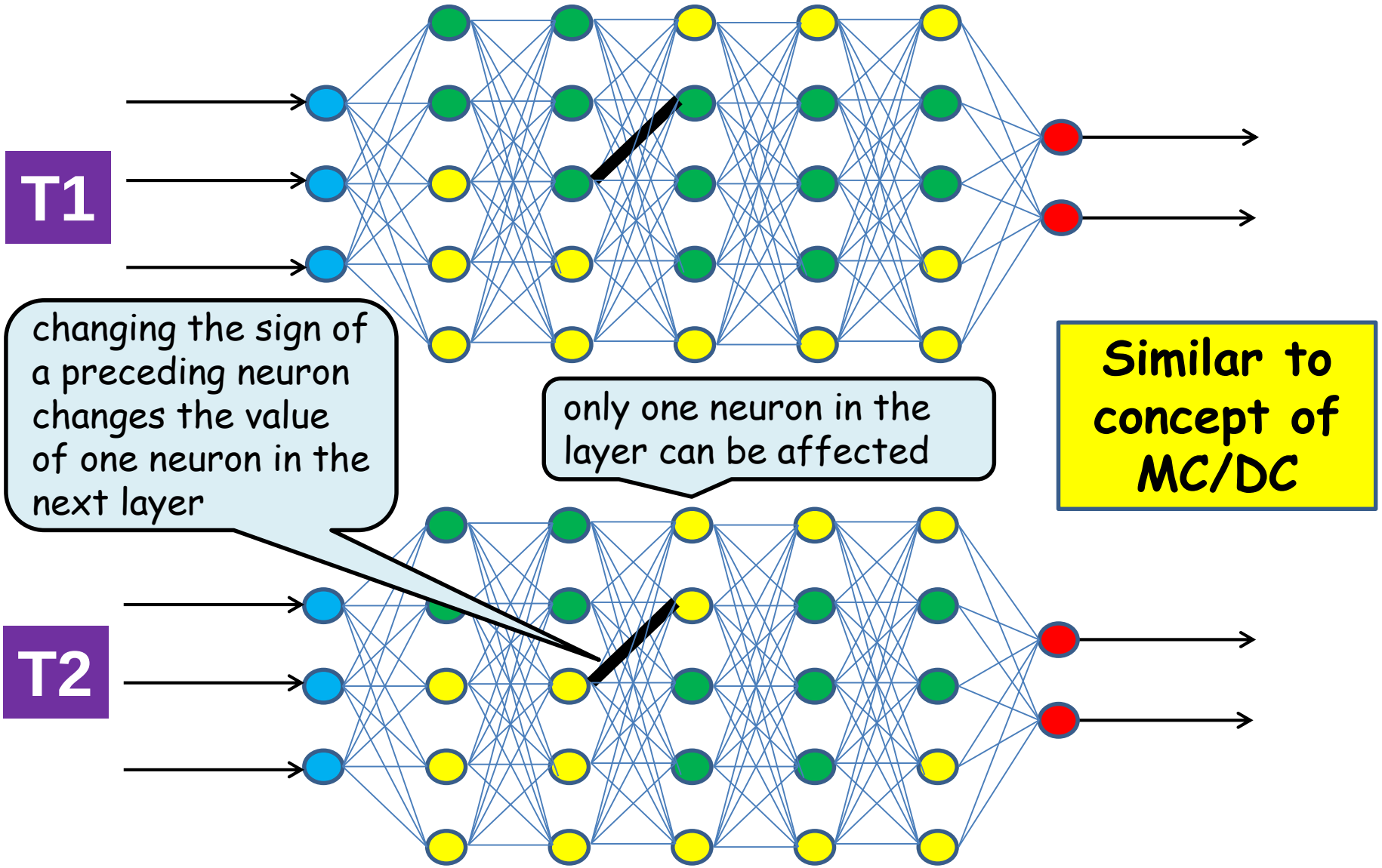
‘Threshold’ Coverage



‘Sign’ Coverage



‘Sign-Sign’ Coverage by Testing

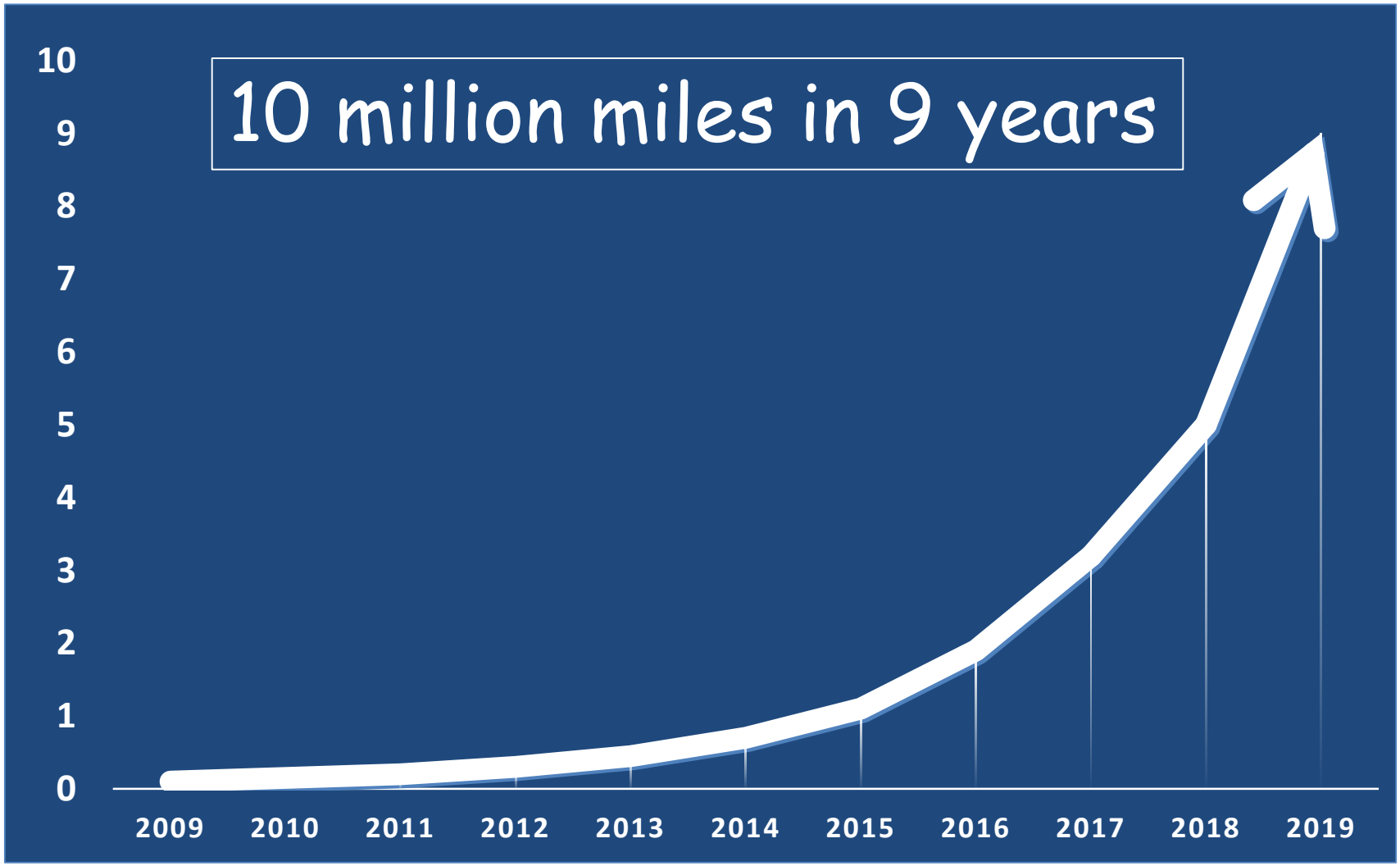


Test Techniques for Neural Networks

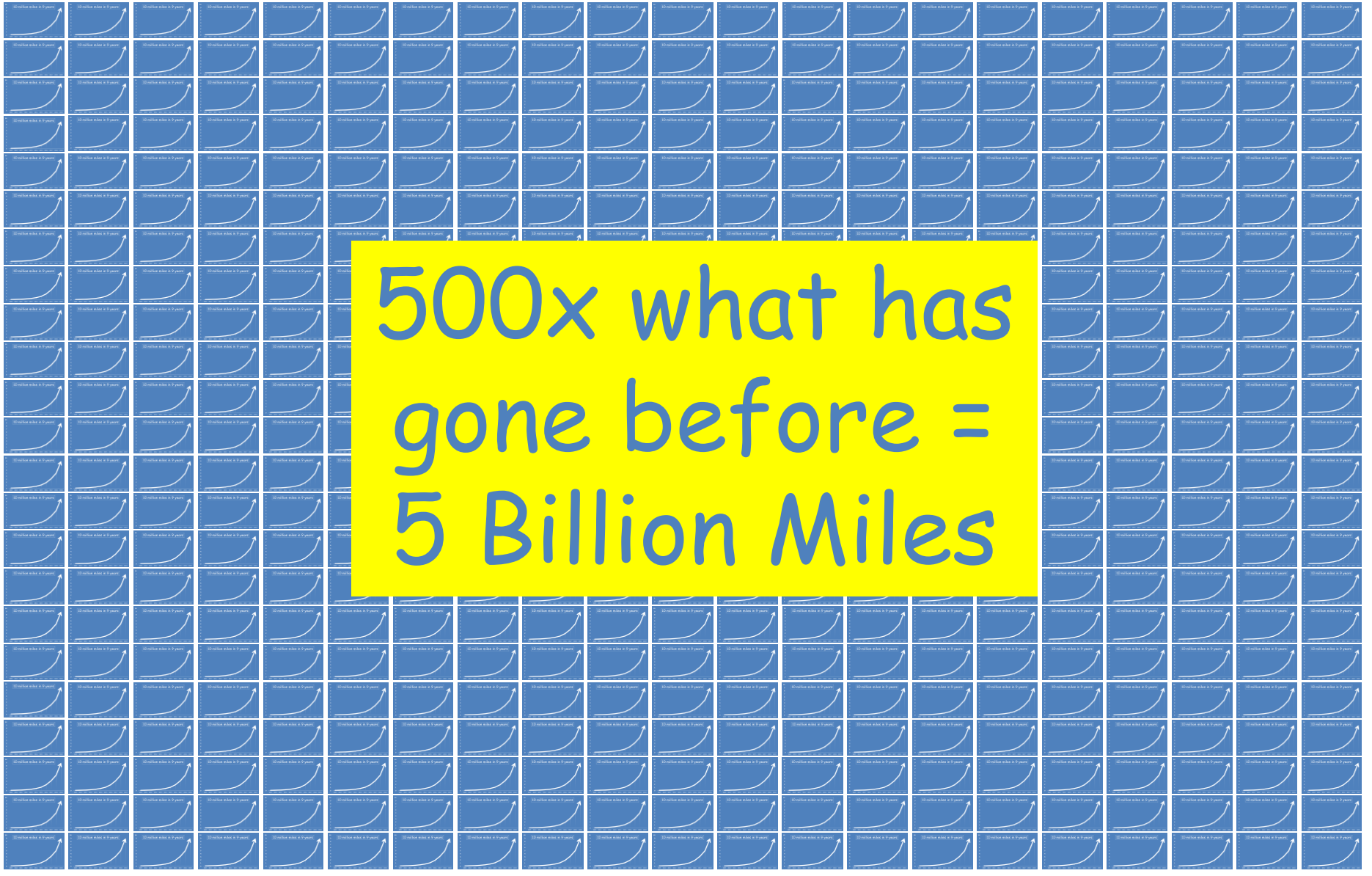
- **Random testing, traditional black box test techniques and neuron coverage do not appear to be good at finding adversarial examples**
  - but, similarly, random testing, equivalence partitioning and statement/branch coverage are also poor at finding defects in traditional embedded systems
- **But, adversarial examples are not our only problem**
- **Until we get more experience, we should use a mix of black and white box techniques**
  - with serious support for automation and environments…

The Necessity of Virtual Test Environments

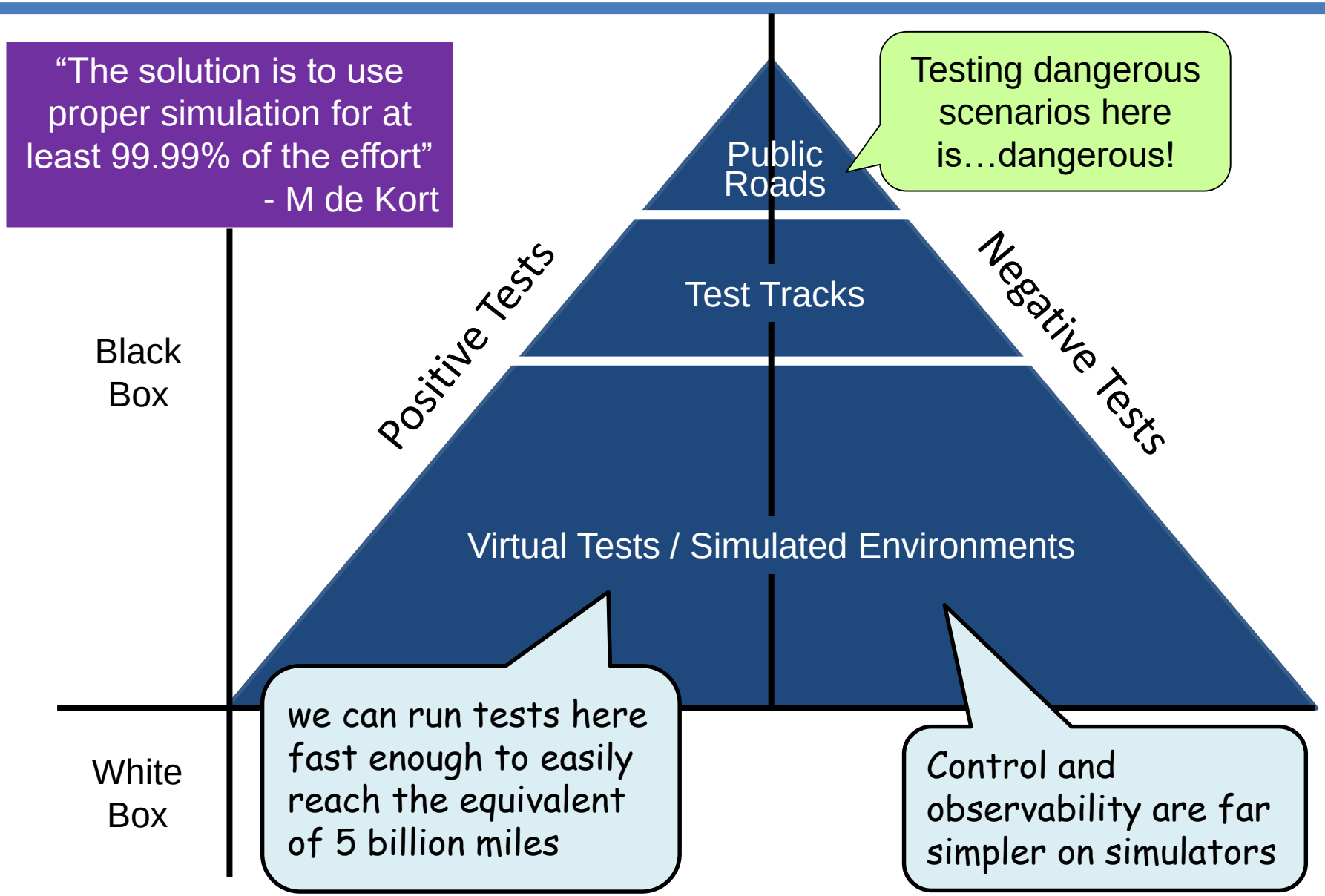
Waymo On-Road Test Miles (millions)



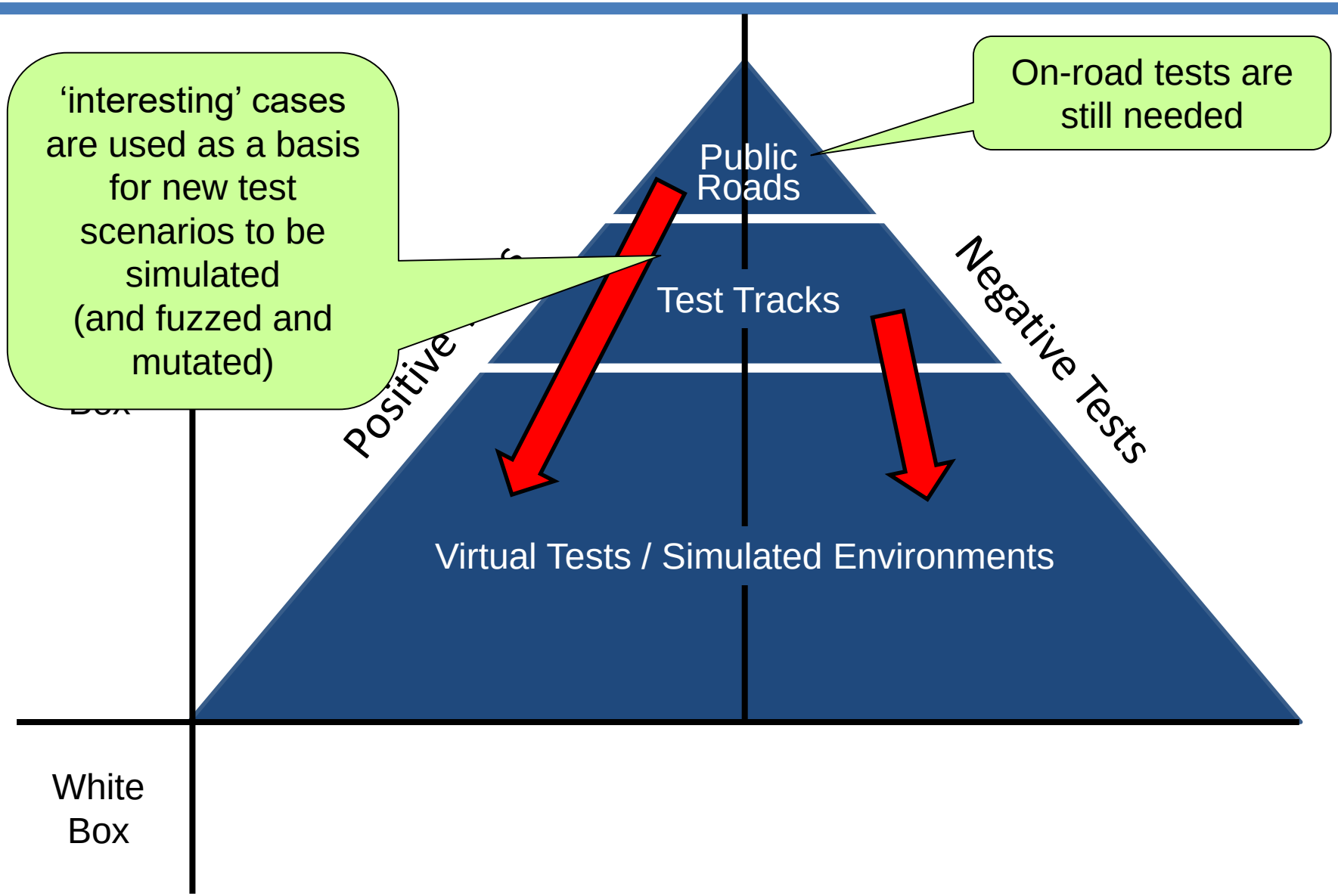
20% Better (than human drivers)



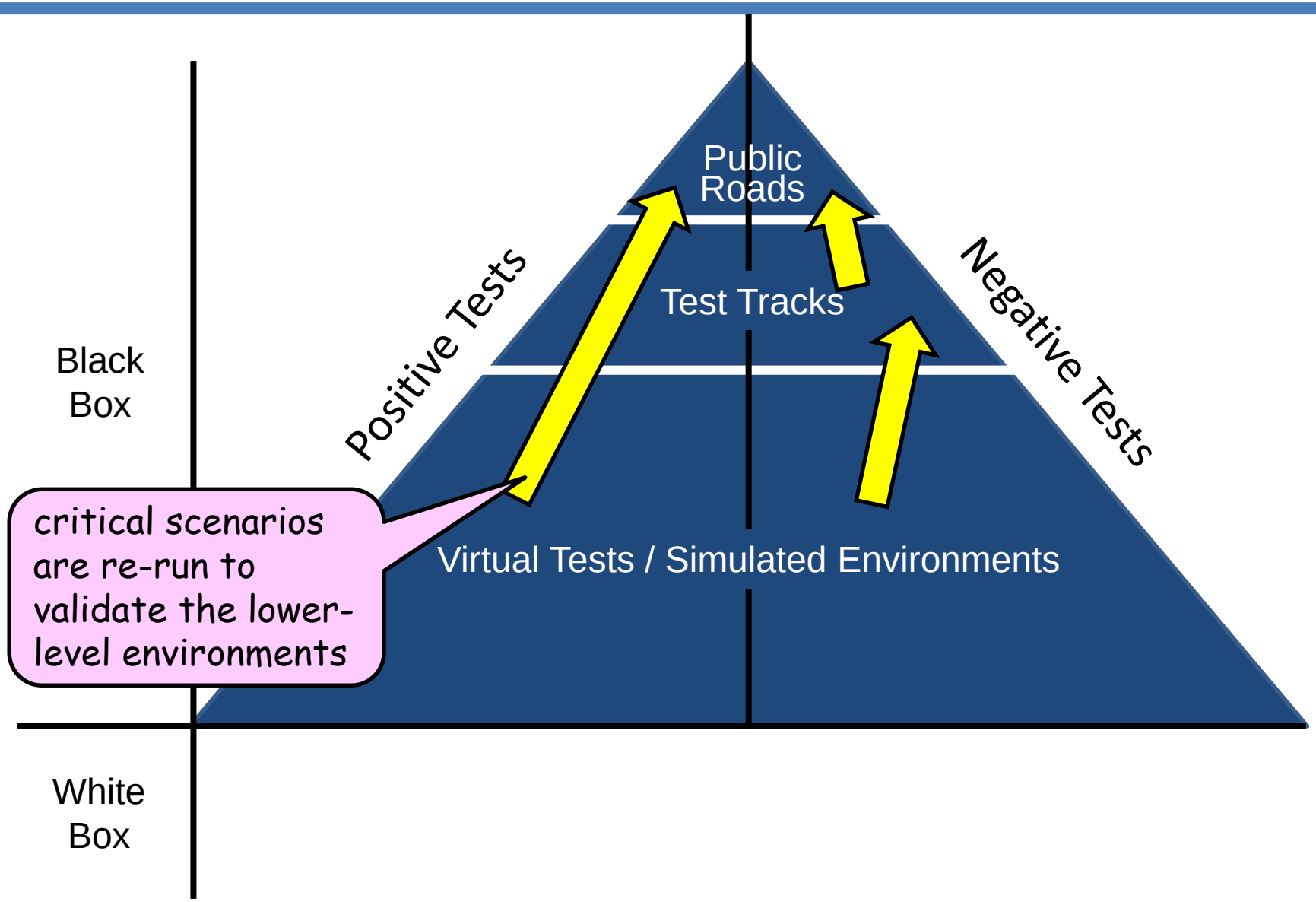
Autonomous Cars – Test Environments



Top-Down Scenario Identification

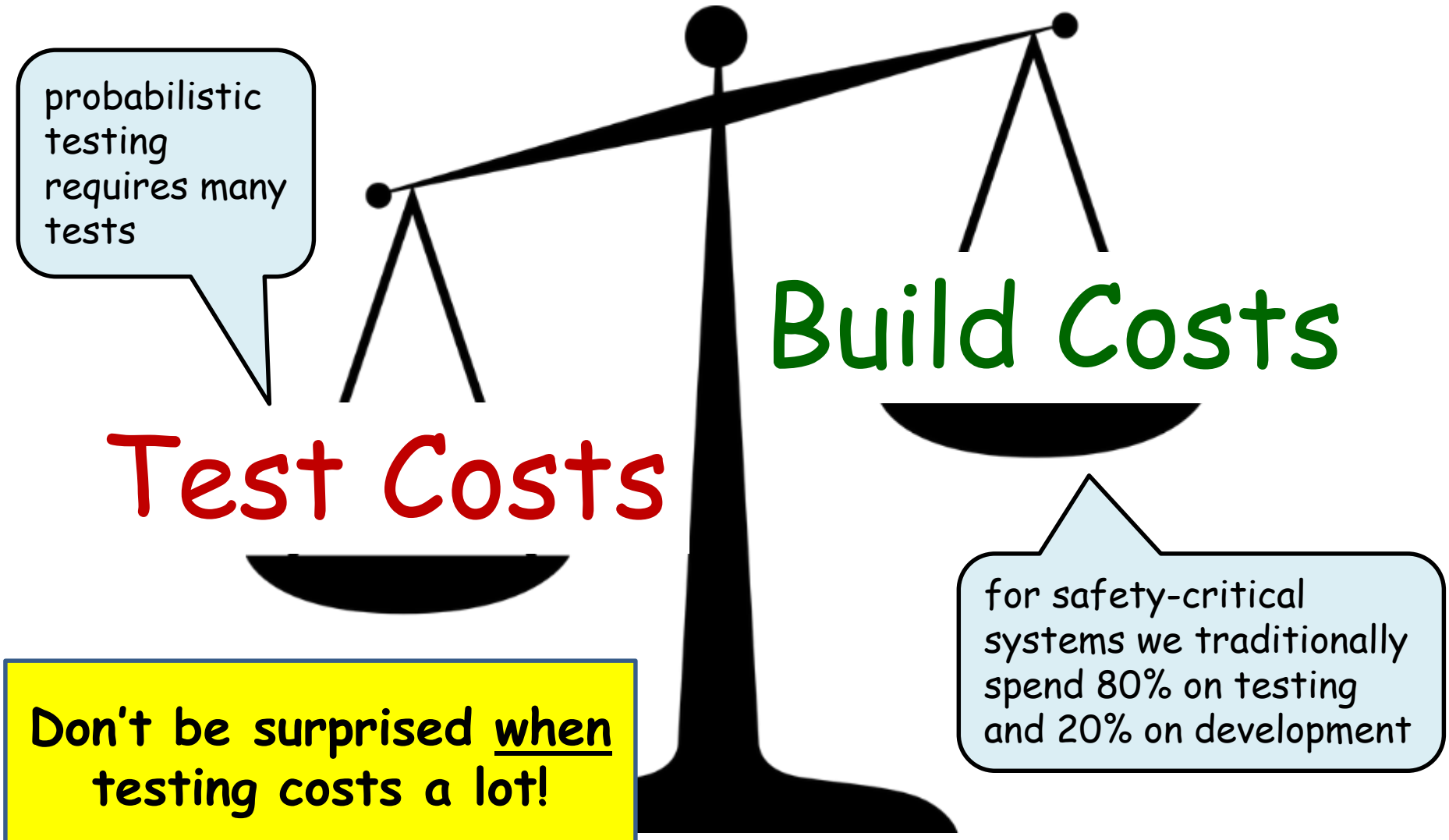


Bottom-Up Validation of Environments & Scenarios



Conclusions

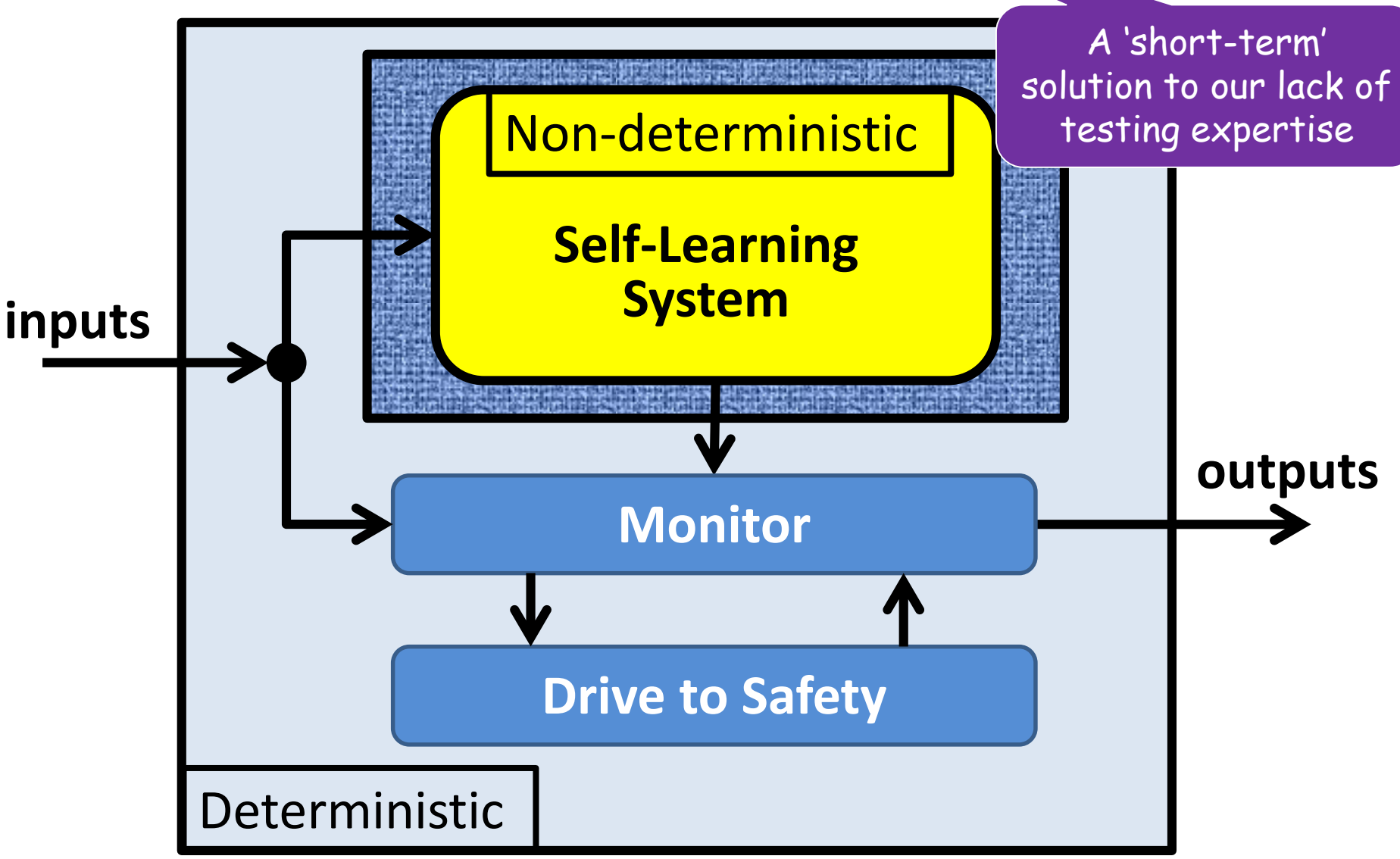
Self-Learning System Costs



Conclusions - Testing Self-Learning Systems

- **We need many tests to cope with these systems':**
  - probabilistic nature
  - high complexity
  - criticality (for safety-related use)
- **…and need the support of sophisticated virtual test environments**
- **Test techniques for Deep Neural Networks need much further research**
  - especially empirical studies of test effectiveness
  - but not concentrated on adversarial examples
- **Until we reach maturity, we should use a safety net…**

Safety Shell Architecture



Thank you for listening