universität innsbruck

Testing the Untestable

Risikobasierte Qualitätssicherung für Machine-Learning Systeme

Prof. Dr. Michael Felderer

Institut für Informatik

Universität Innsbruck

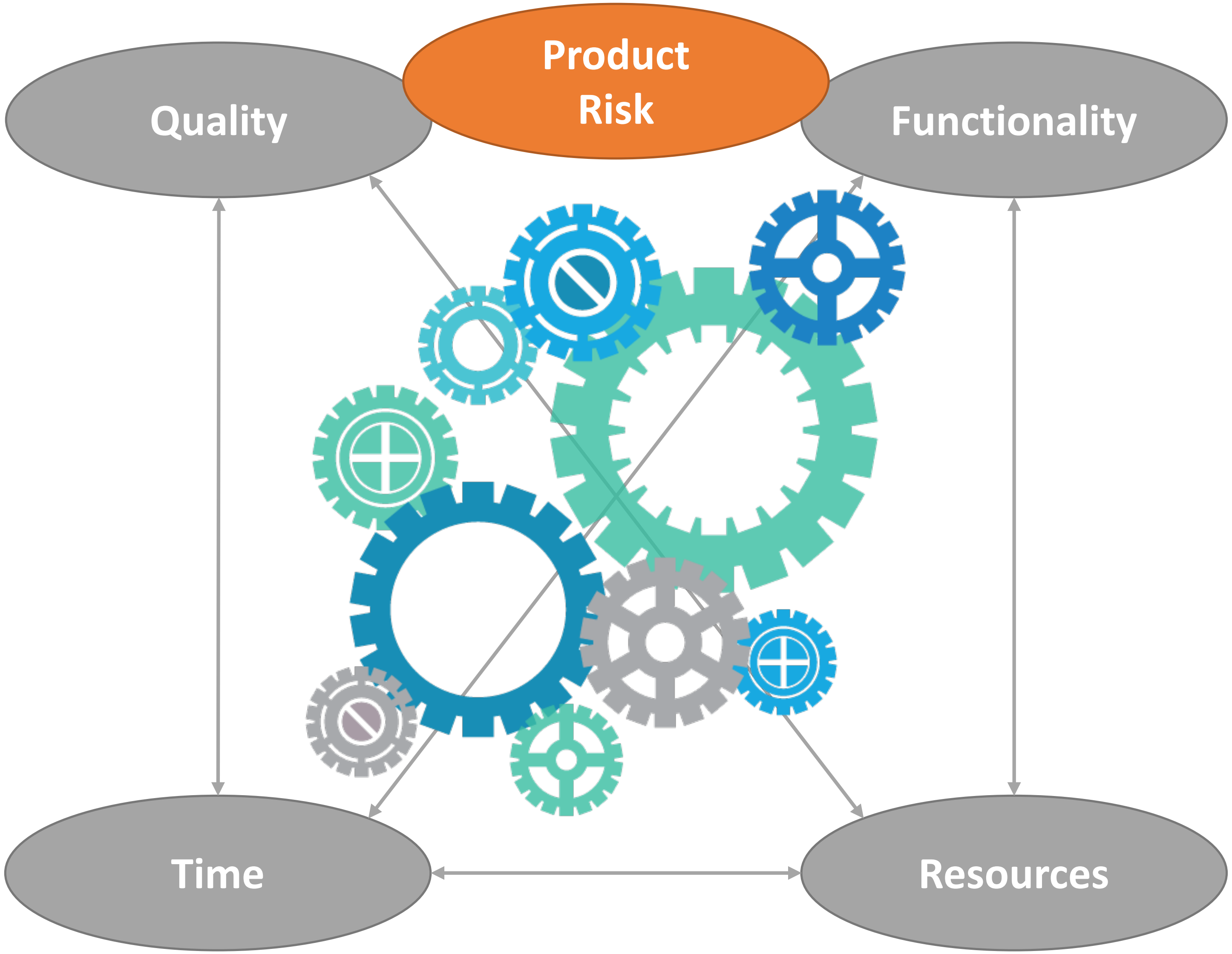
Agenda

- Motivation
- Classical Risk-Based Testing
- Testing Machine-Learning-Based Systems
- Data Testing
- Online Testing

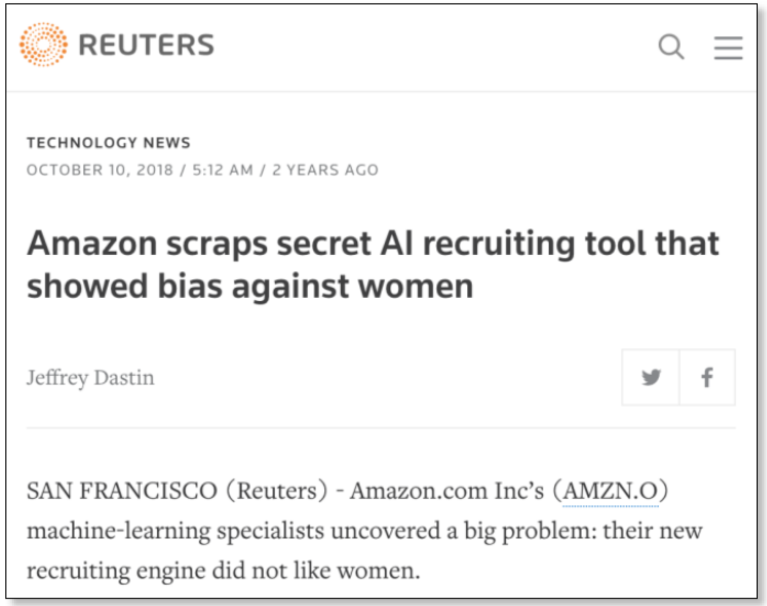
Agenda

- Motivation
- Classical Risk-Based Testing
- Testing Machine-Learning-Based Systems
- Data Testing
- Online Testing

Quality and Risk in Software Development



Quality and Risk of ML-based Systems


REUTERS

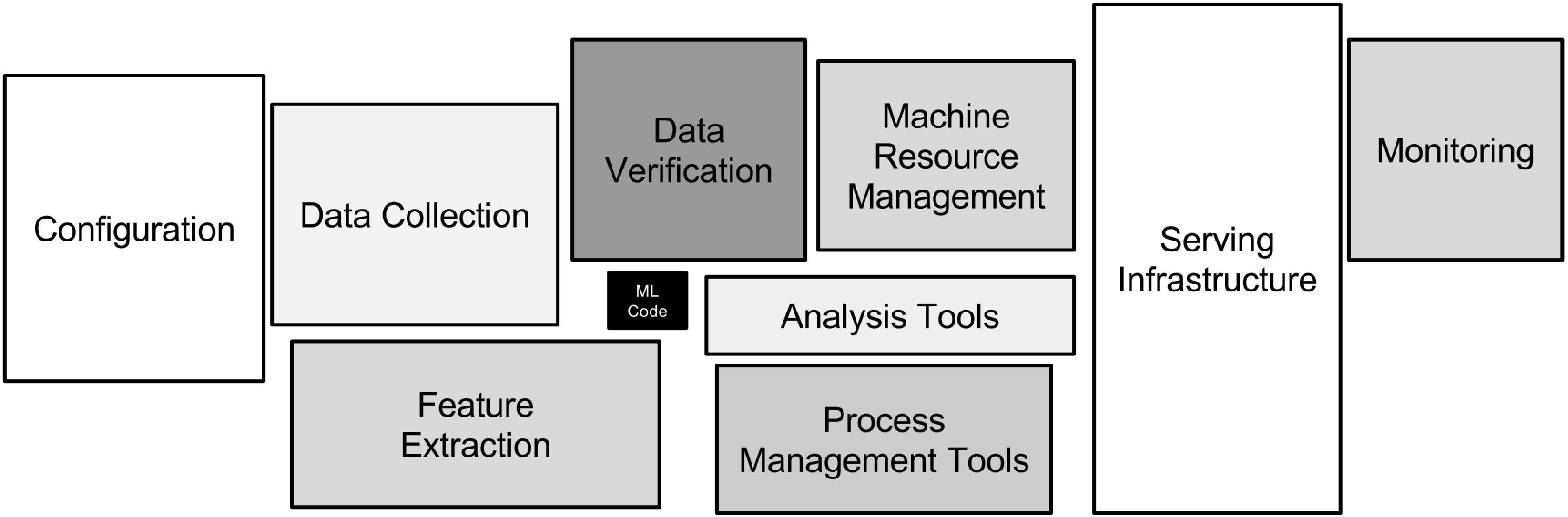
TECHNOLOGY NEWS
OCTOBER 10, 2018 / 5:12 AM / 2 YEARS AGO

Amazon scraps secret AI recruiting tool that showed bias against women

Jeffrey Dastin

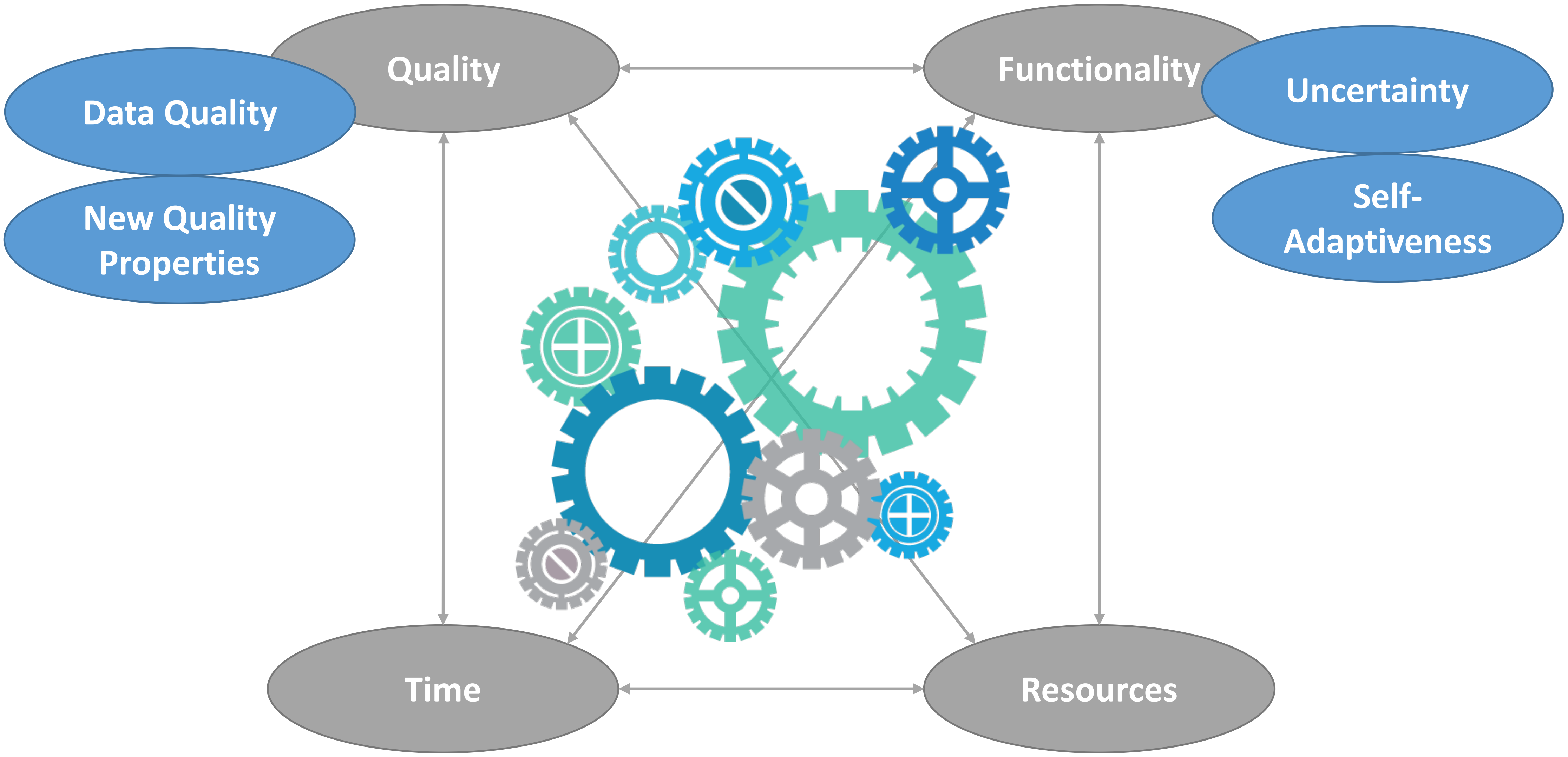
SAN FRANCISCO (Reuters) - Amazon.com Inc's (AMZN.O) machine-learning specialists uncovered a big problem: their new recruiting engine did not like women.

ML Systems are IT Systems not only Algorithms



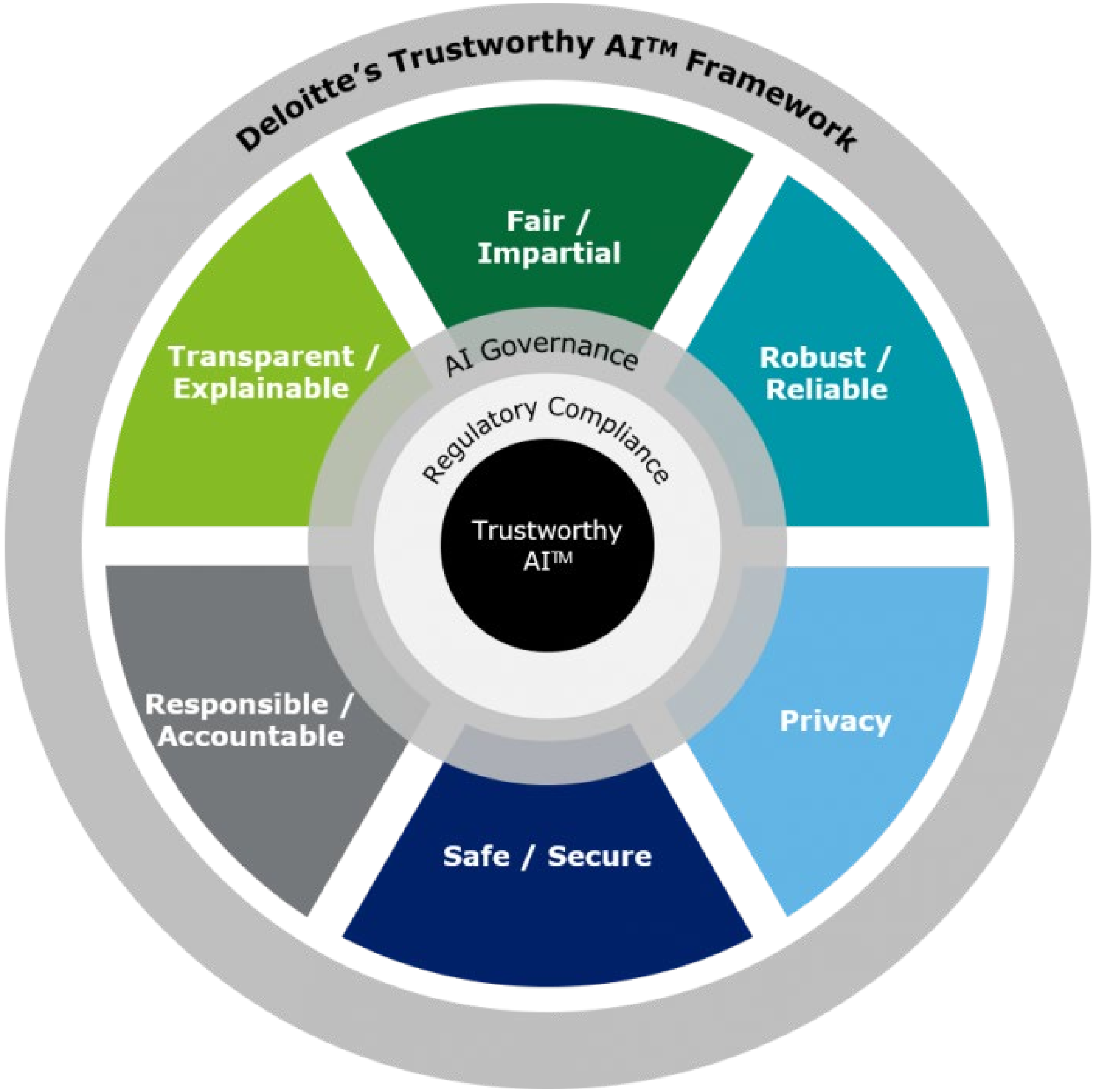
Sculley et al.: Hidden Technical Debt in Machine Learning Systems, NIPS 2012

What Makes (AI)ML-Based Systems Different?



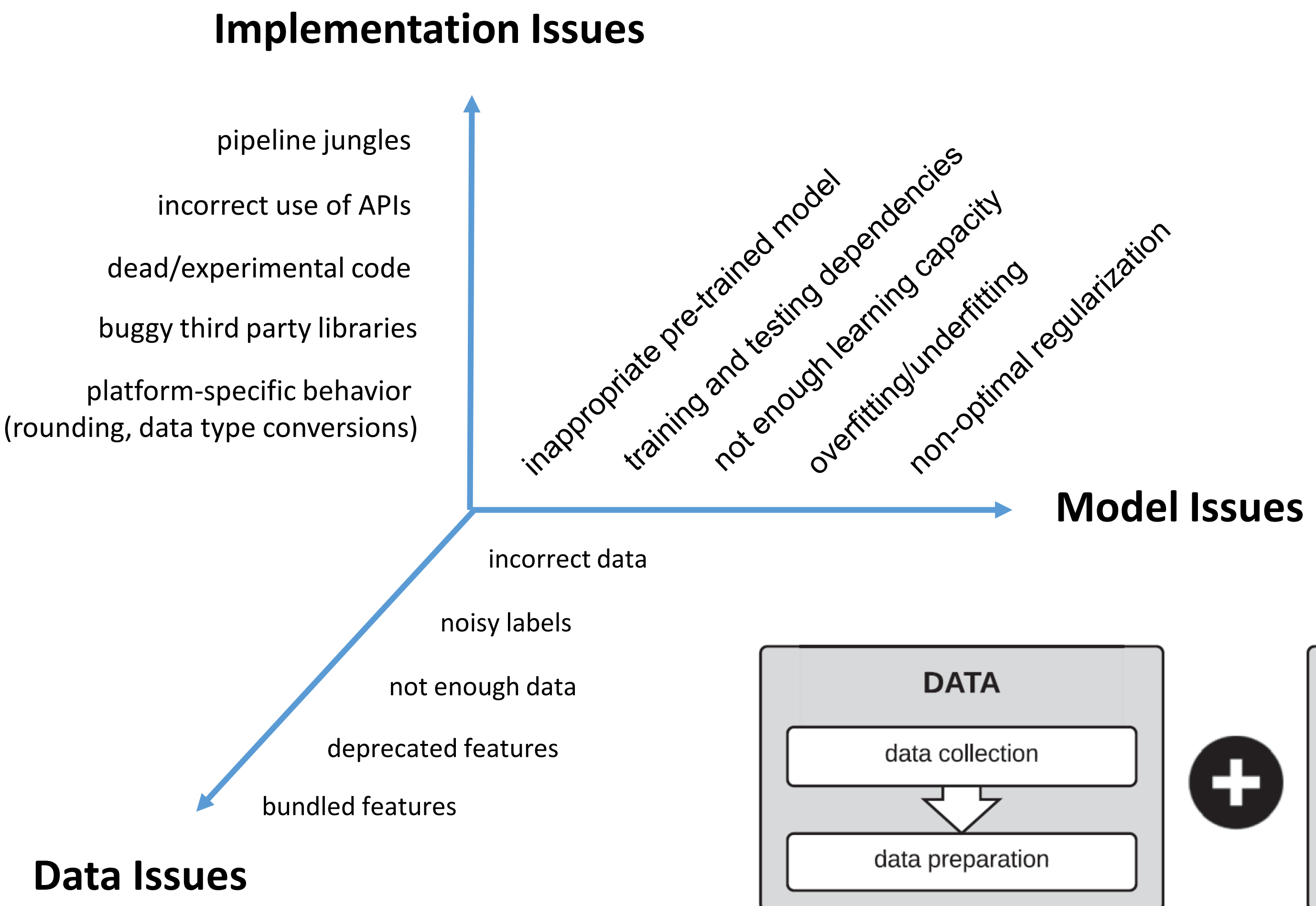
New Quality Properties: Trustworthiness of AI

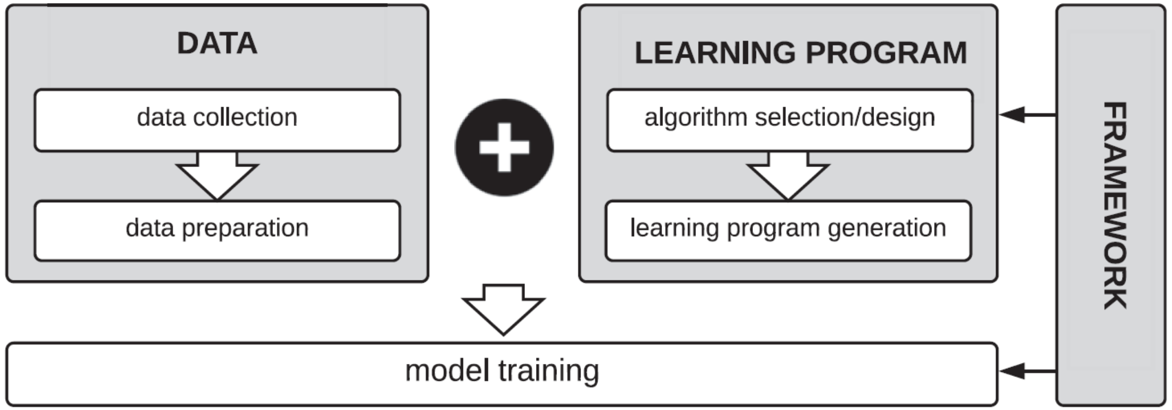
EU Ethics Guidelines for Trustworthy AI



Deloitte's Trustworthy AI Framework

Machine-Learning Systems as Software Systems



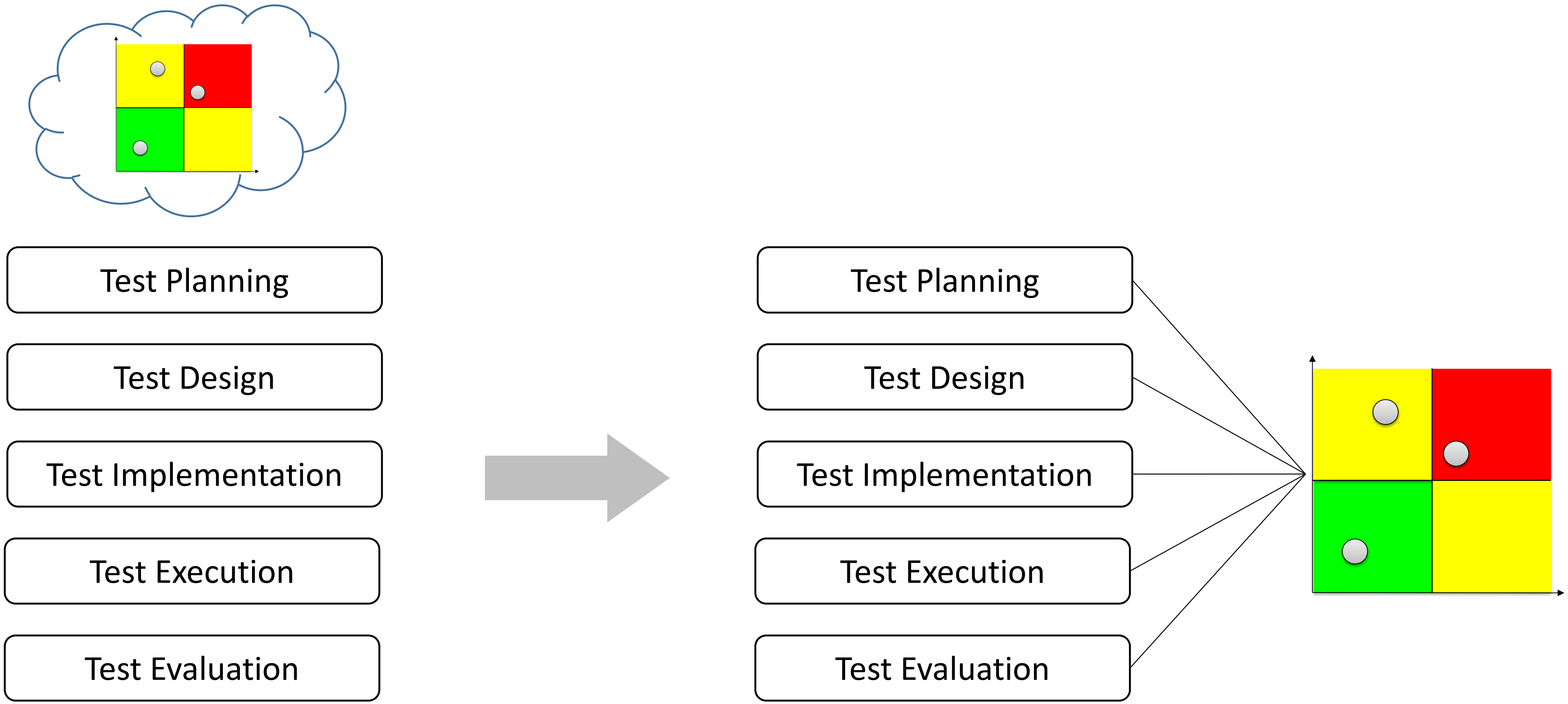


Zhang, Jie M., et al. *Machine learning testing: Survey, landscapes and horizons*. IEEE Transactions on Software Engineering, 2020

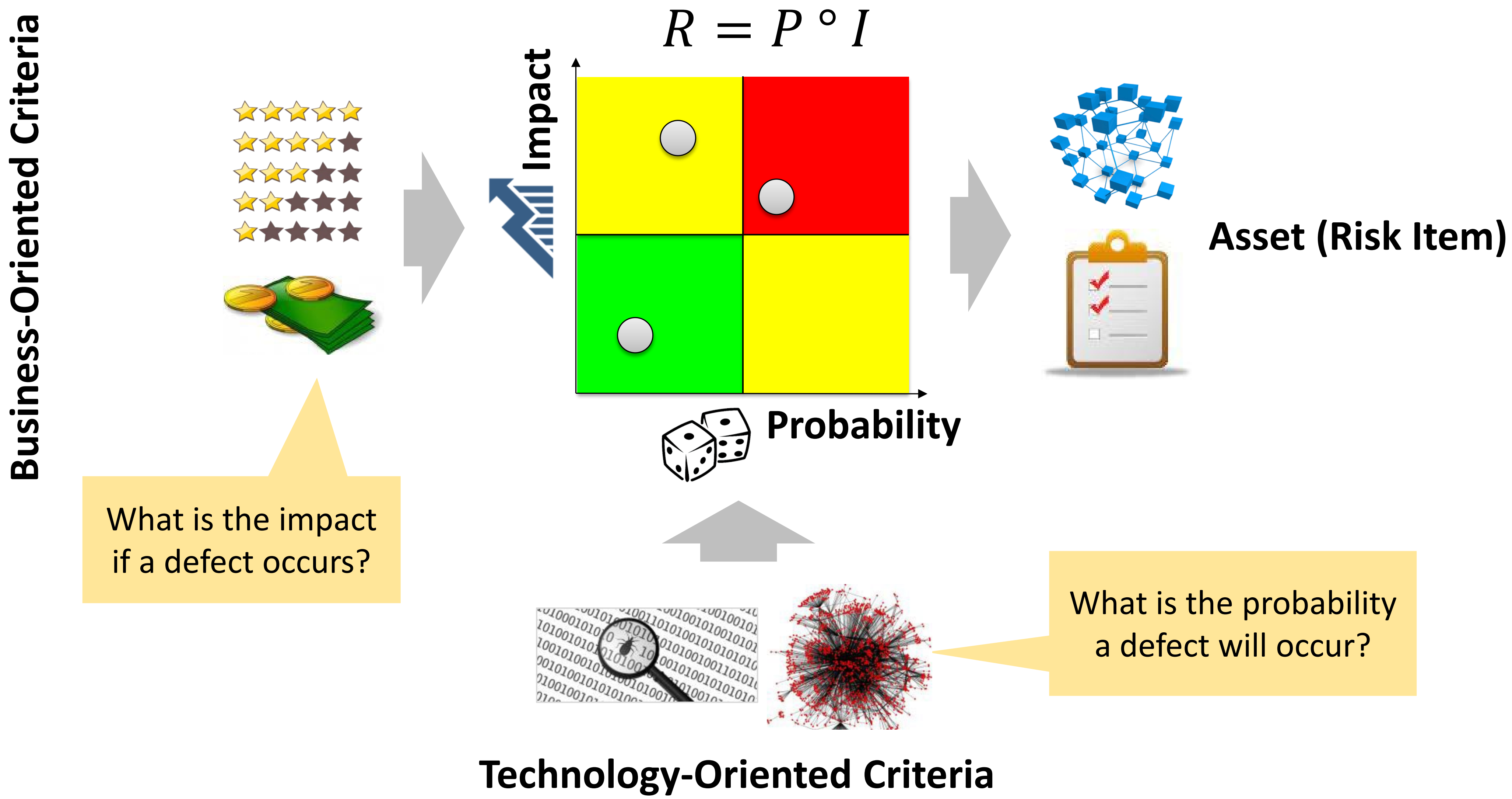
Agenda

- Motivation
- Classical Risk-Based Testing
- Testing Machine-Learning-Based Systems
- Data Testing
- Online Testing

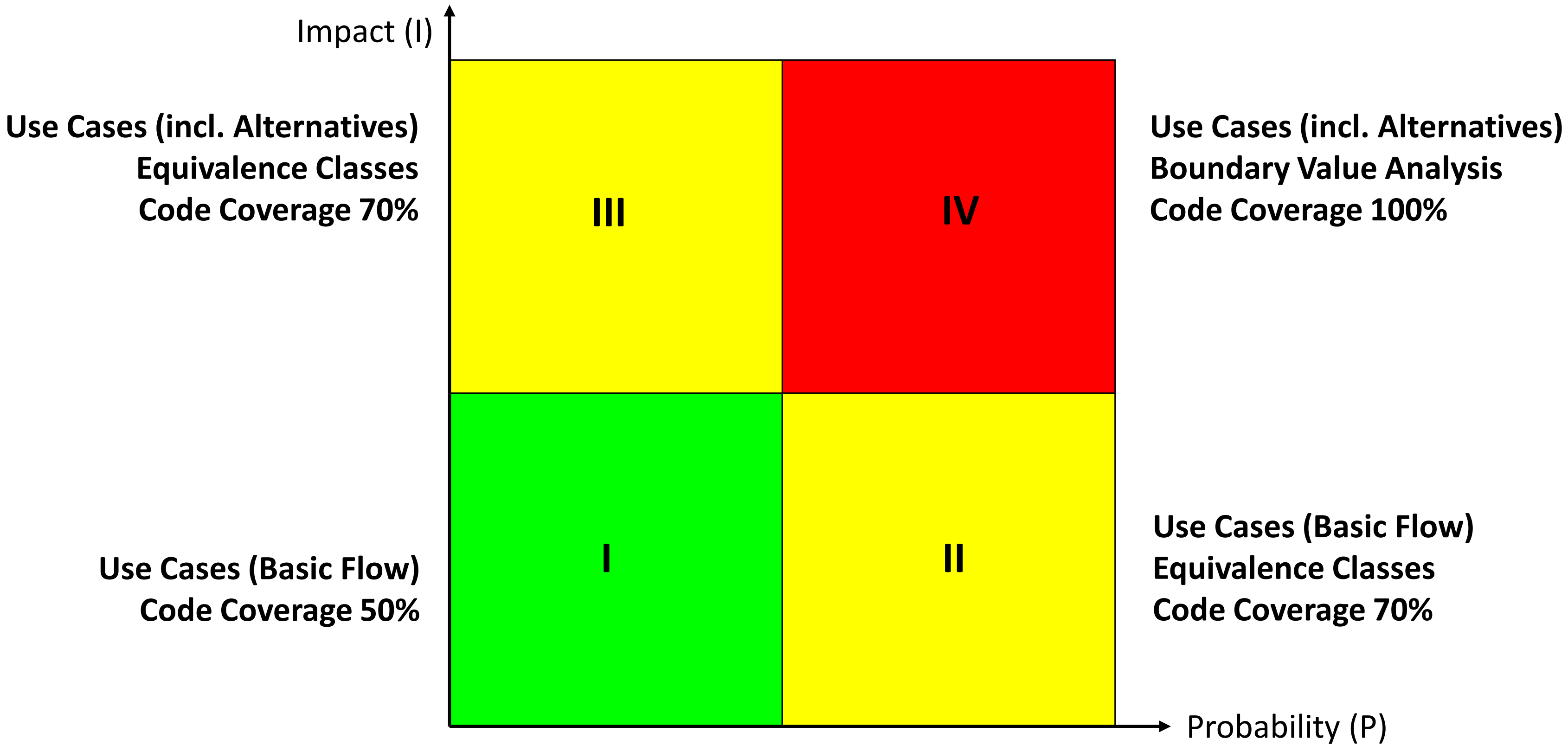
Risk-Based Testing (Risk-Based Quality Assurance)



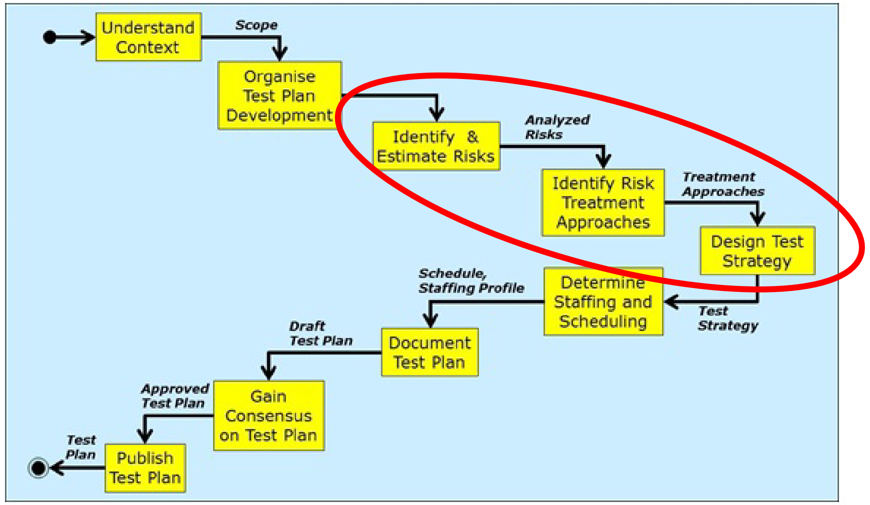
Risk Concept in Software Quality Engineering

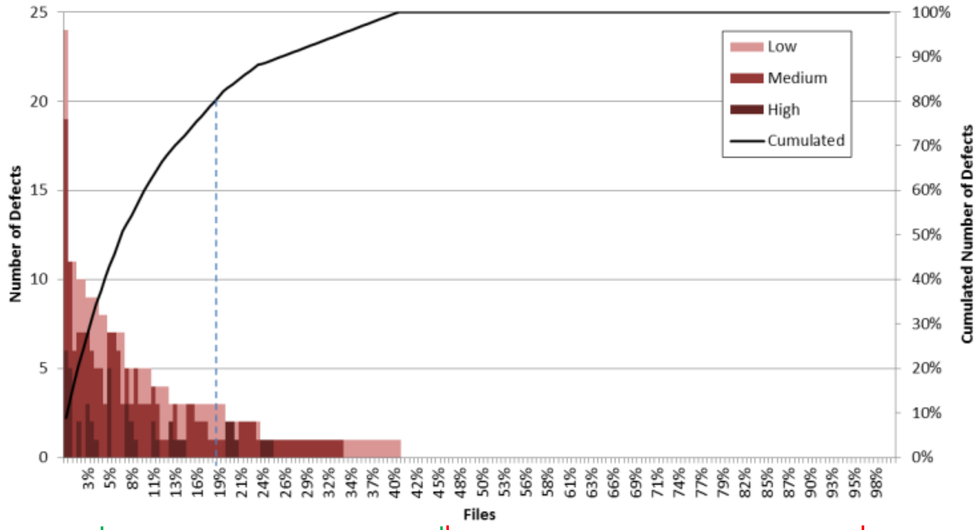


Risk-Based Test Strategy



Effectiveness and Efficiency of Risk-Based Testing





MORE **TEST** **LESS**

Compliance

- Aligned with standard
- Minimizing critical risks
- ...

Effectiveness

- Finding more defects
- Finding defects earlier
- Finding the critical defects
- ...

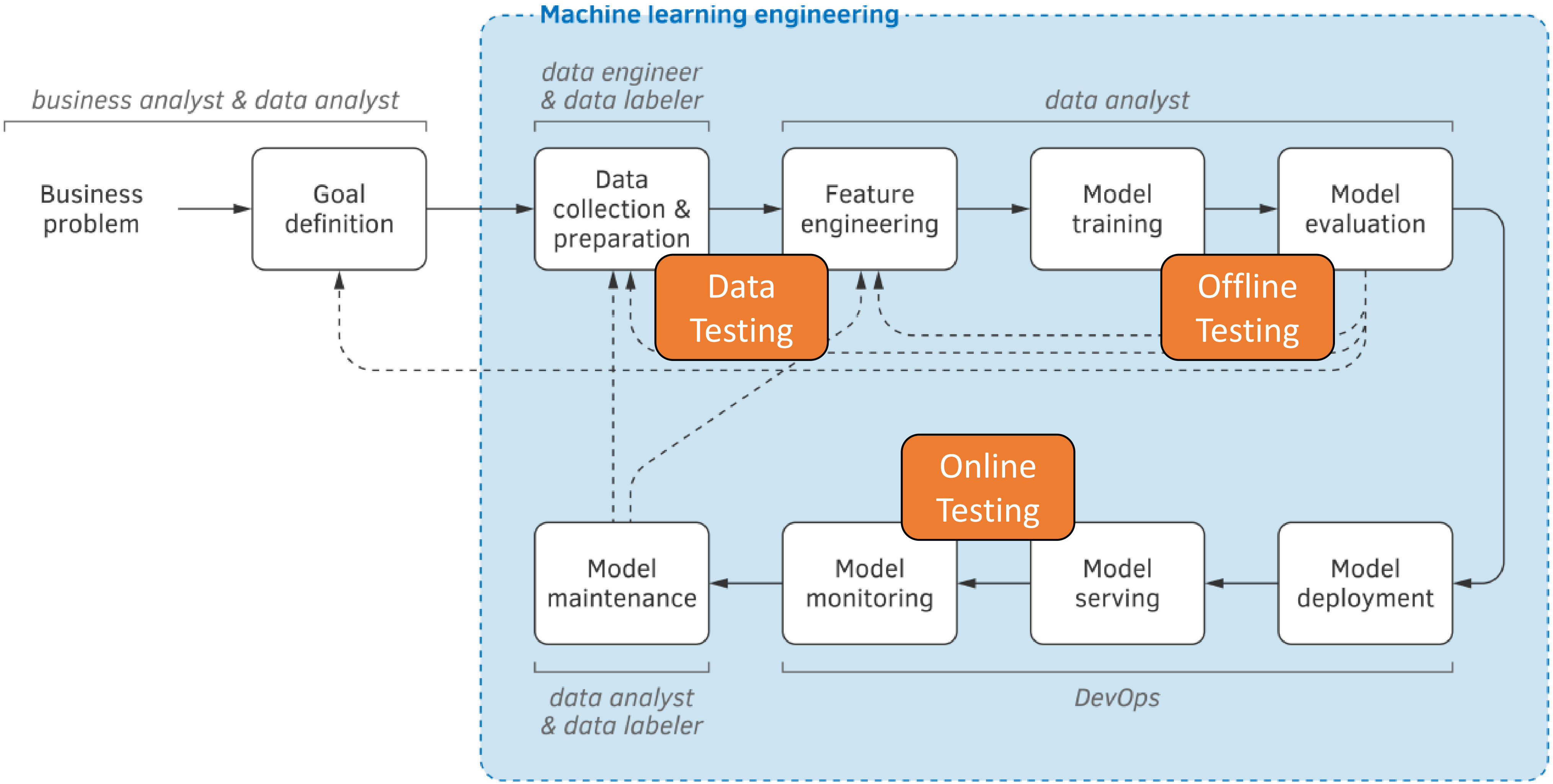
Efficiency

- Reduce time of testing
- Reduce cost of testing
- ...

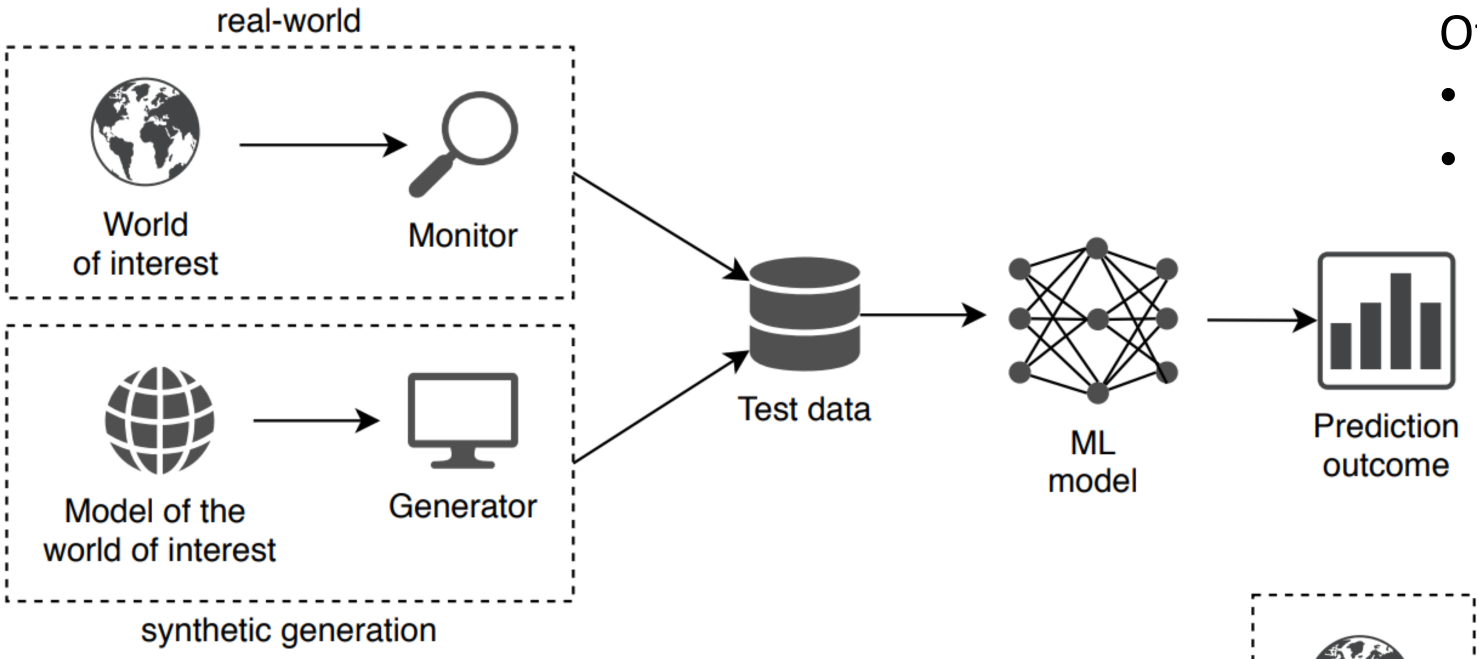
Agenda

- Motivation
- Classical Risk-Based Testing
- Testing Machine-Learning-Based Systems
- Data Testing
- Online Testing

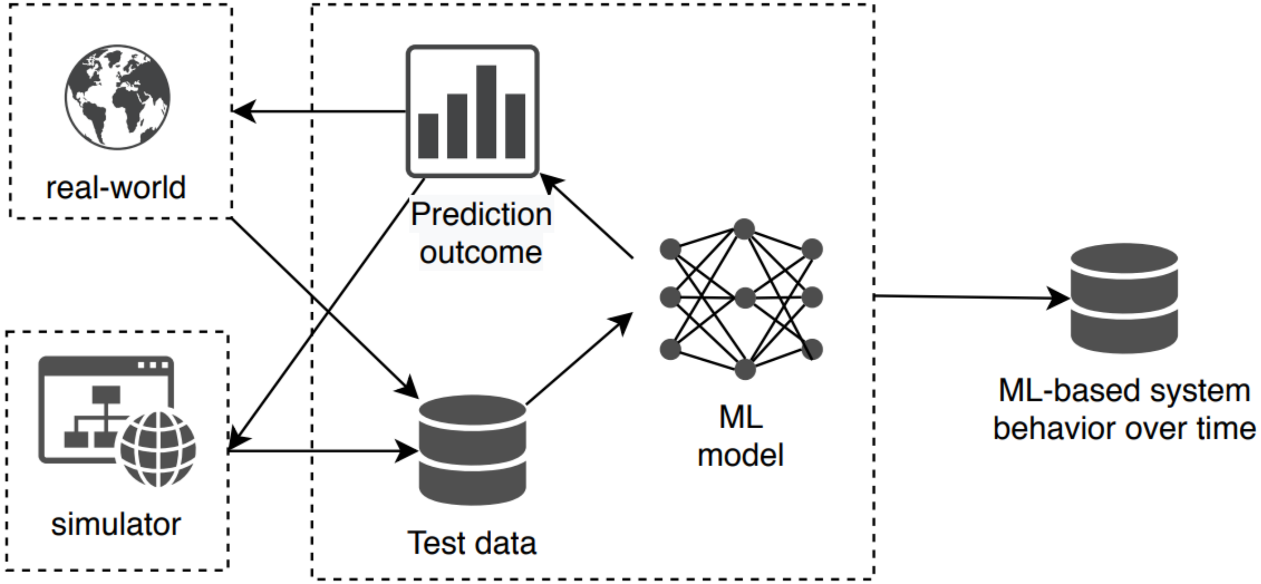
ML Engineering and Testing



Offline and Online Testing

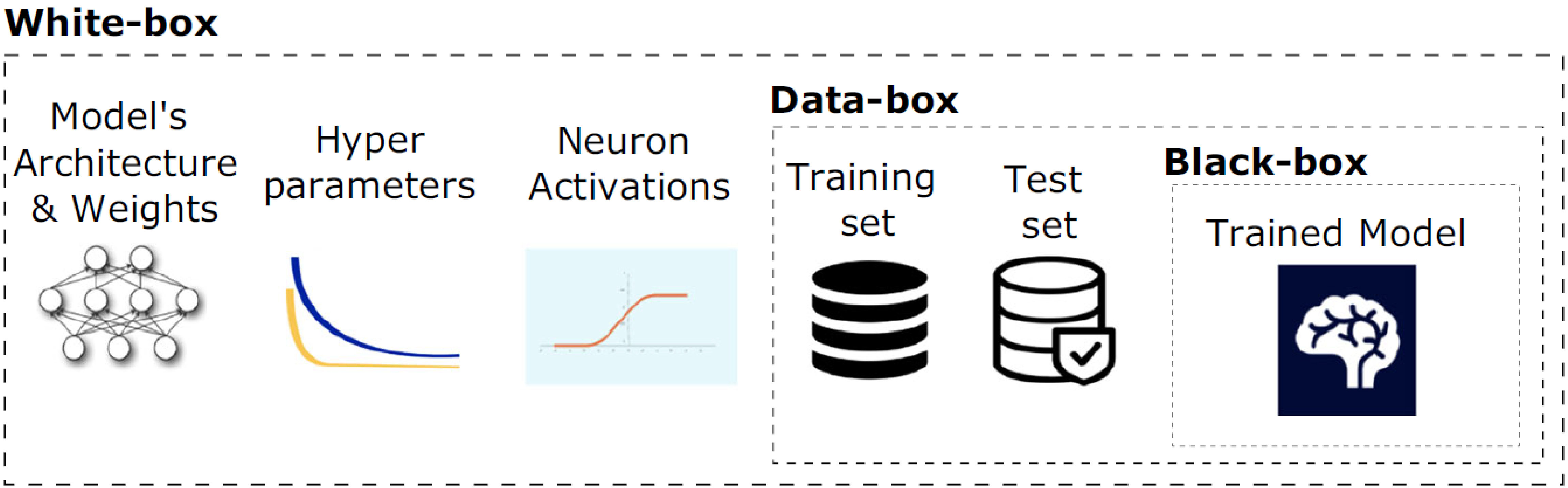


Offline Testing:
- Testing ML model as standalone component
- ML model tested as a unit in open loop mode

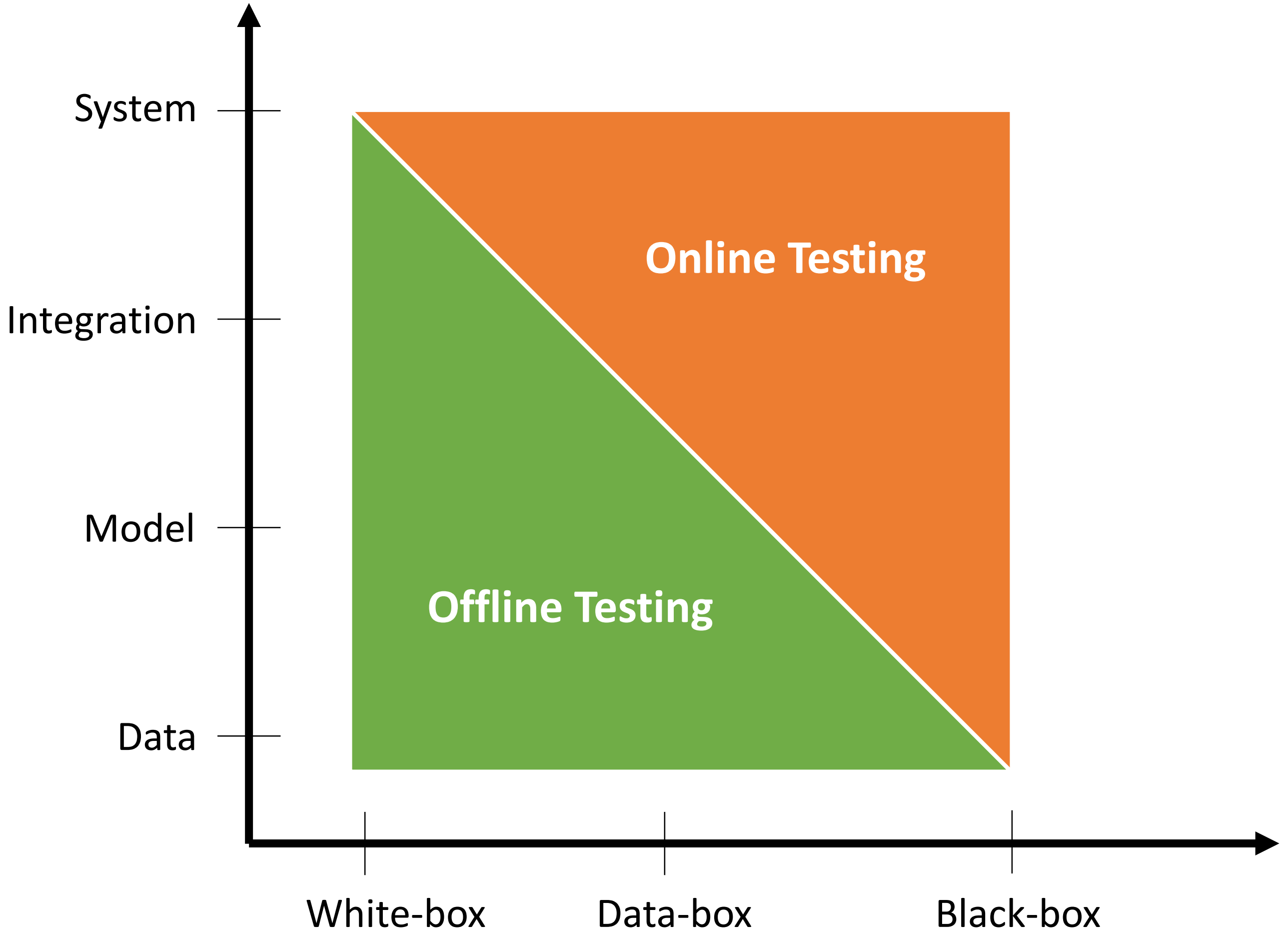


Online Testing:
- Testing ML model in real or simulated environment
- ML model tested as a unit in closed loop mode

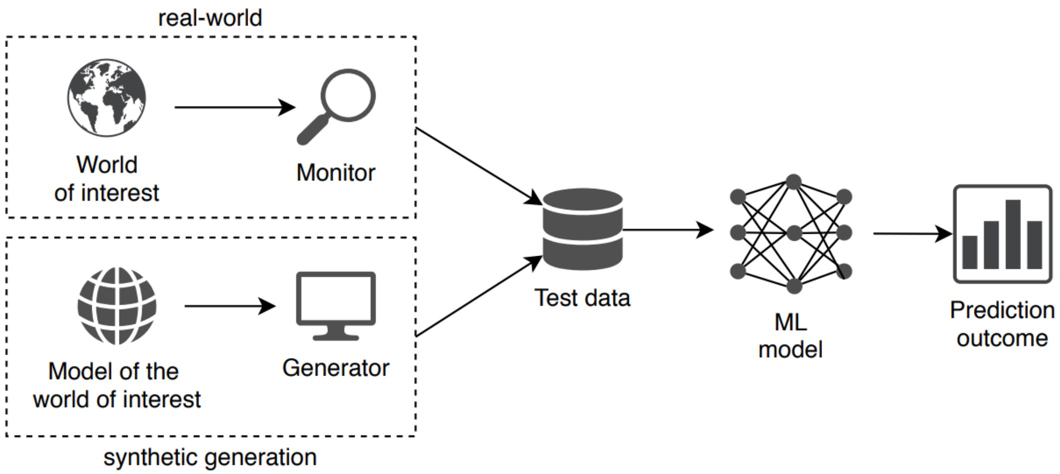
White-Box and Black-Box Testing?

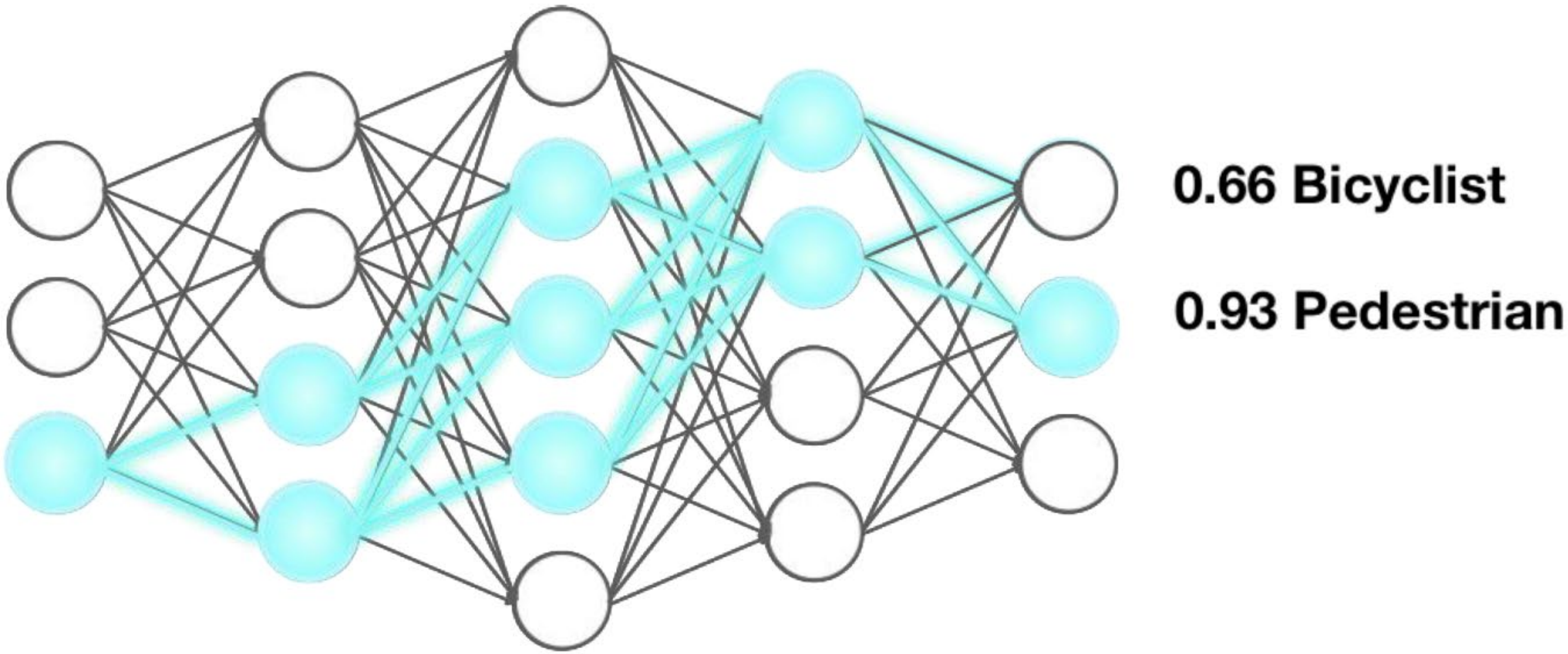


Offline vs. Online Testing



Offline Testing (White Box)





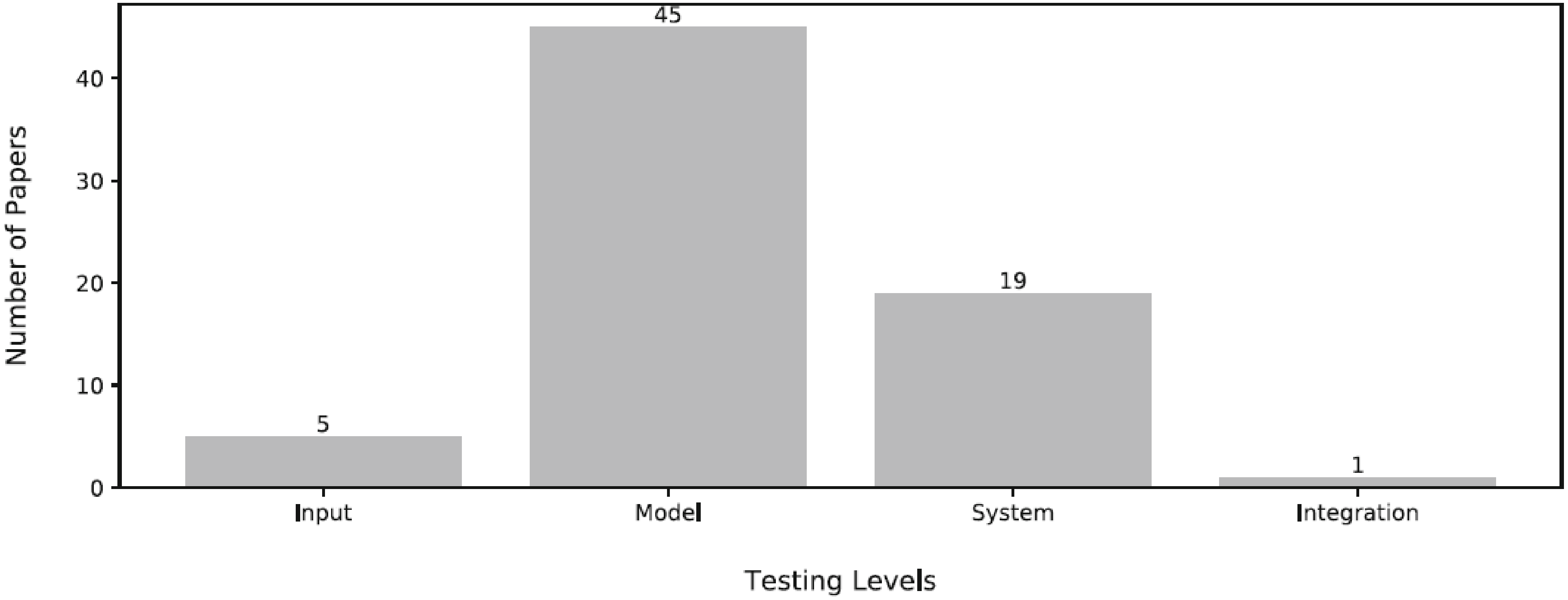
Neuron Coverage is not strongly and positively correlated with defect detection and naturalness.

Harel-Canada, F., et al. *Is neuron coverage a meaningful measure for testing deep neural networks*?. Proceedings of the 28th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering. 2020

Test Levels for Machine-Learning Systems

Four Test Levels

- Input Testing
- Model Testing
- Integration Testing
- System Testing

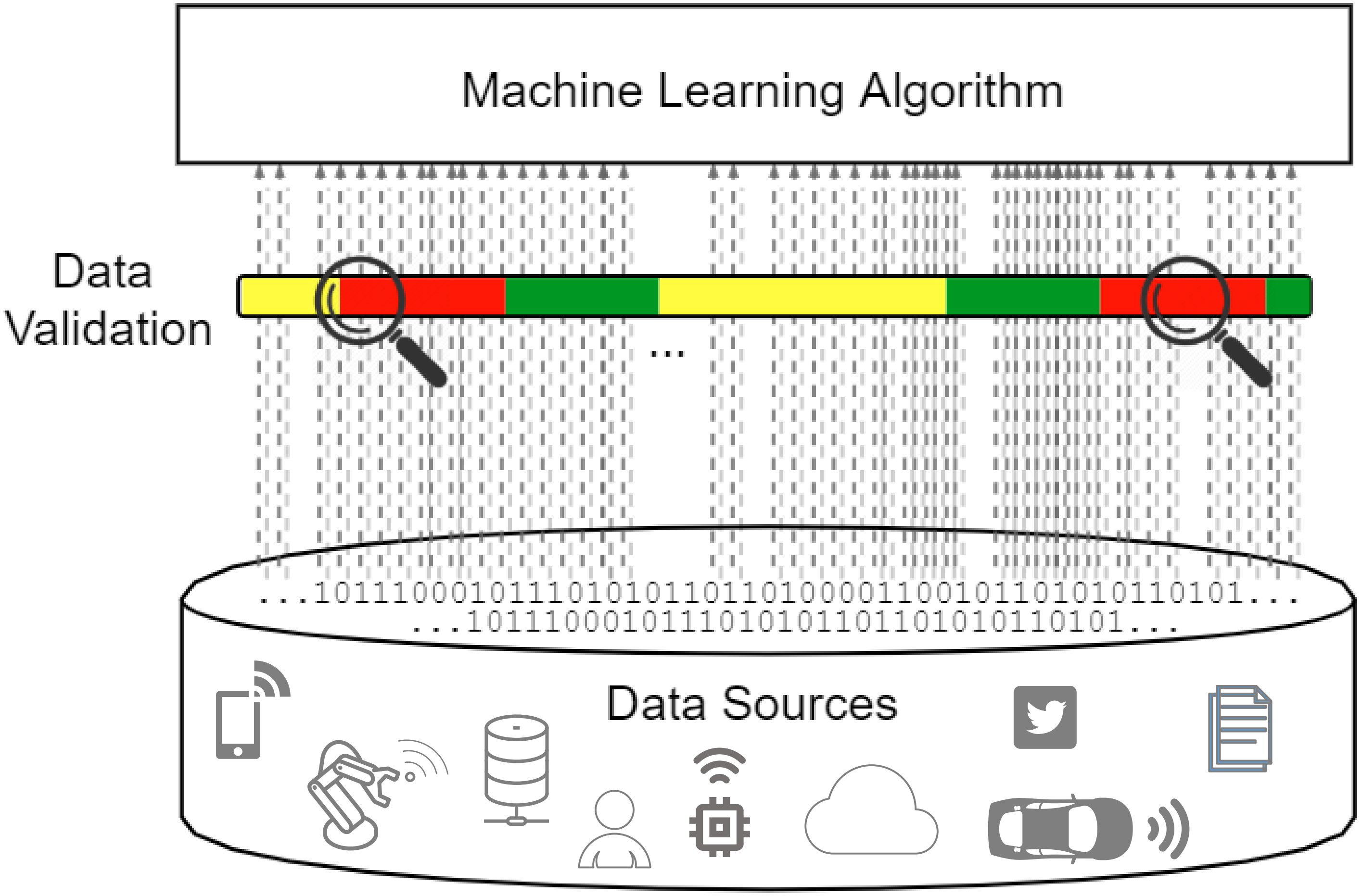


Riccio, Vincenzo, et al. *Testing machine learning based systems: a systematic mapping*. Empirical Software Engineering 25.6 (2020): 5193-5254.

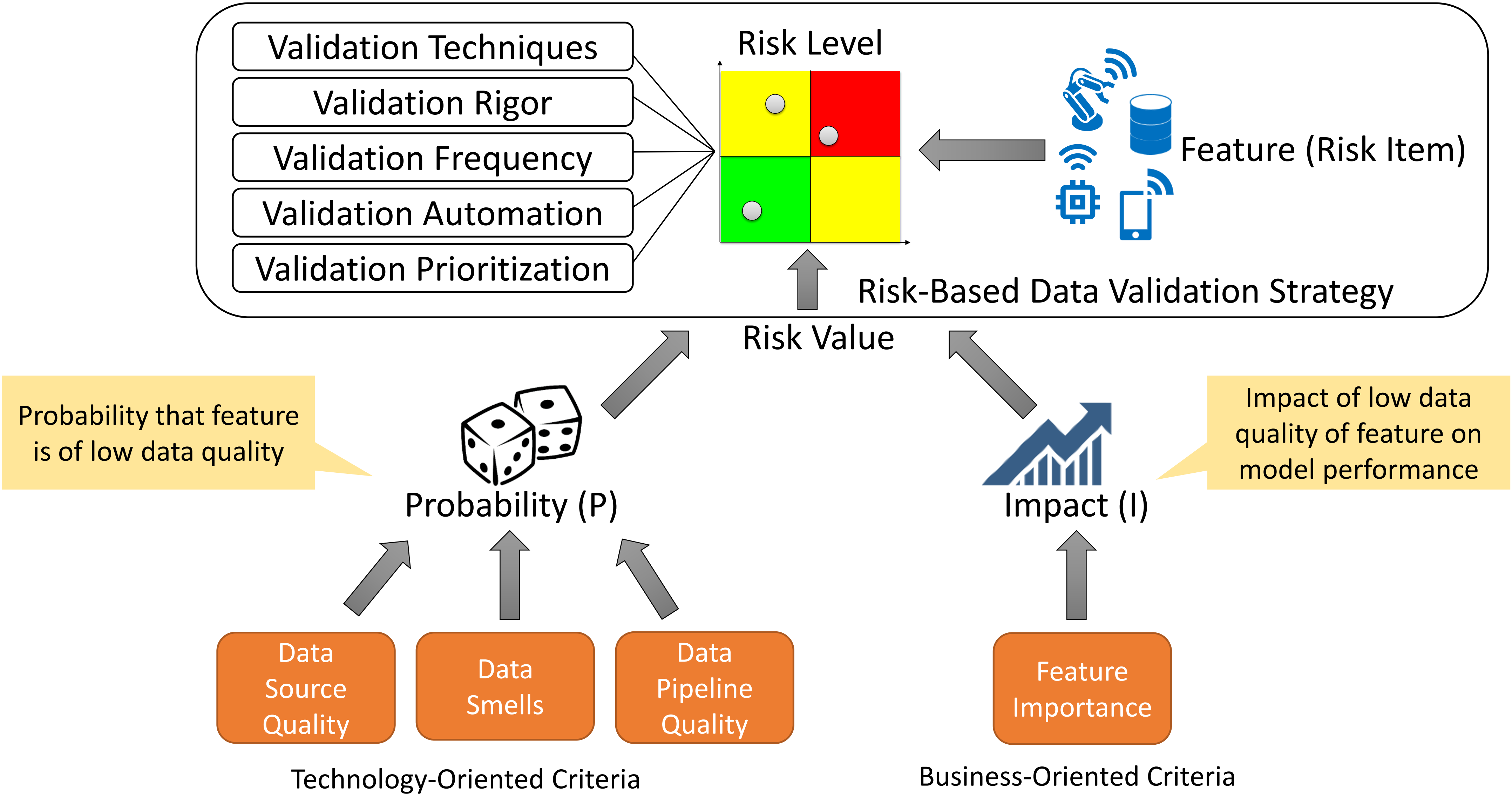
Agenda

- Motivation
- Classical Risk-Based Testing
- Testing Machine-Learning-Based Systems
- Data Testing
- Online Testing

Idea of Risk-Based Data Testing



Risk-Based Data Testing Framework



universität innsbruck

Data Smell Types

Context-independent, data value-based indications of latent data quality issues caused by poor practices that may lead to problems in the future.

- **Data Smells**
  - **Believability Smells**
    - *Suspect Date Value*
    - *Meaningless Value*
    - *Dummy Value*
    - *Suspect Precision*
    - *Suspect Distribution*
    - *Suspect Date/Time Interval*
  - **Understandability Smells**
    - **Encoding Smells**
      - *Date as Date/Time*
      - *Date as String*
      - *Date/Time as String*
      - *Floating-point Number as String*
      - *Integer as Floating-point Number*
      - *Integer as String*
      - *Time as String*
    - **Syntactic Smells**
      - *Ambiguous Date/Time Format*
      - *Ambiguous Value*
      - *Casing*
      - *Conctracting*
      - *Extraneous Value*
      - *Intermingled Data Type*
      - *Long Data Value*
      - *Separating*
      - *Small Number*
      - *Spacing*
      - *Special Character*
      - *Synonym*
  - **Consistency Smells**
    - *Abbreviation Inconsistency*
    - *Casing Inconsistency*
    - *Date/Time Format Inconsistency*
    - *Missing Value Inconsistency*
    - *Precision Inconsistency*
    - *Separating Inconsistency*
    - *Spacing Inconsistency*
    - *Special Character Inconsistency*
    - *Syntax Inconsistency*
    - *Transposition Inconsistency*
    - *Unit Inconsistency*

Example Smells

Appointme...	Appointme...	# Age
5638447	2016-04-29T00:00:00Z	21
5629123	2016-04-29T00:00:00Z	19
5630213	2016-04-29T00:00:00Z	30
5620163	2016-04-29T00:00:00Z	29
5634718	2016-04-29T00:00:00Z	22
5636249	2016-04-29T00:00:00Z	28

Date as Date/Time Smell

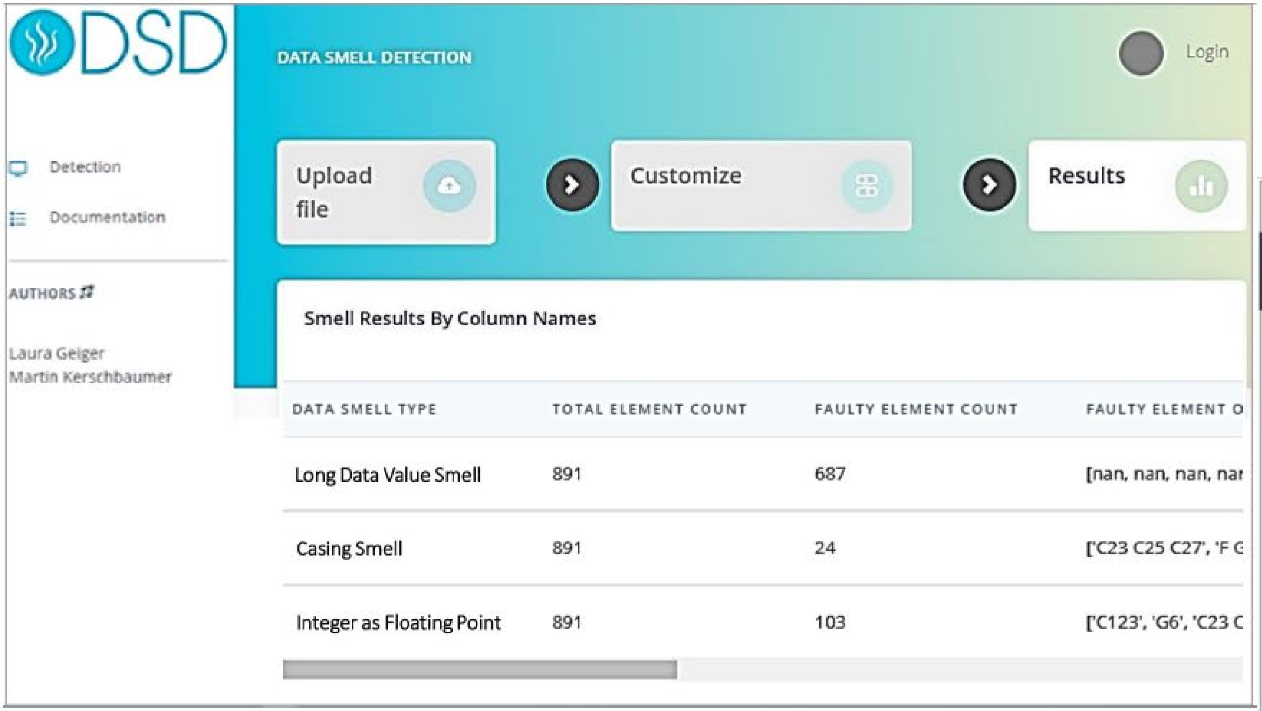
https://www.kaggle.com/joniarroba/noshowappointments

Date	# Daily Confi...	# Total Confi...
11-Feb	0	3
12-Feb	0	3
13-Feb	0	3
14-Feb	0	3
15-Feb	0	3
16-Feb	0	3
17-Feb	0	3
18-Feb	0	3

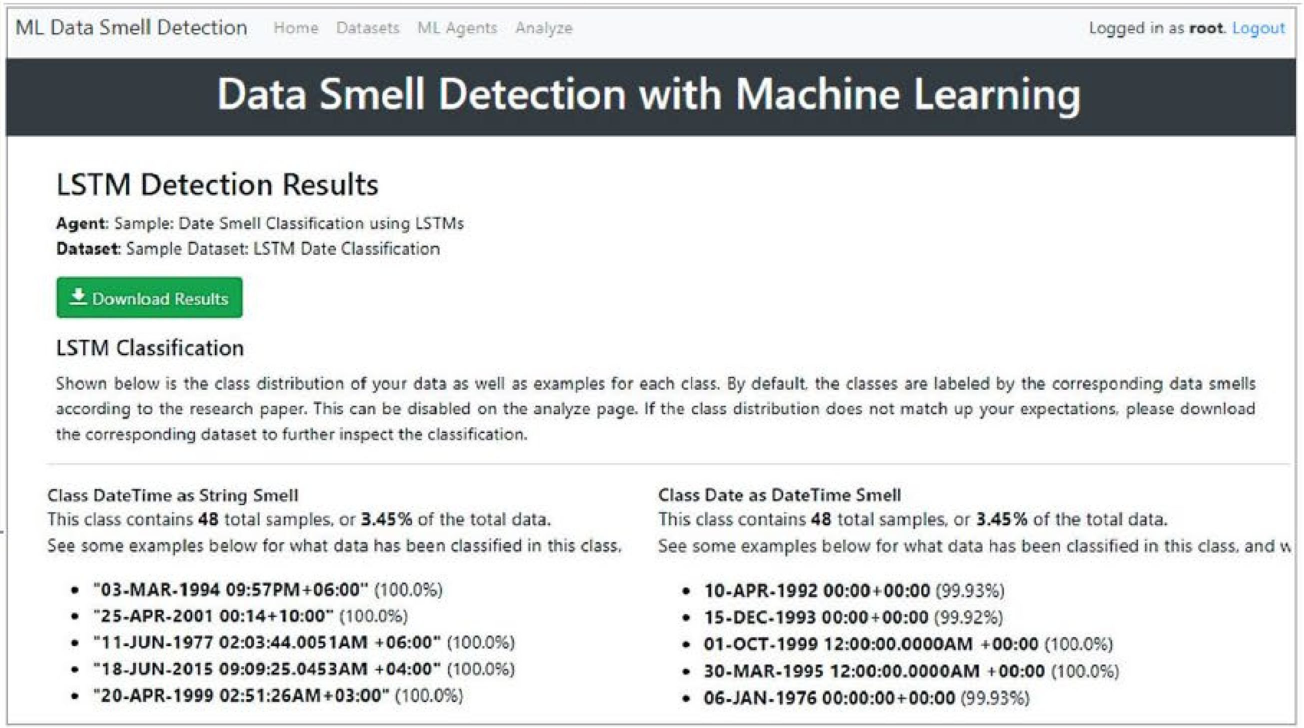
Ambiguous Date/Time Format Smell

https://www.kaggle.com/ravichaubey1506/covid19-india

Data Smell Detection Tools



Rule-based detection



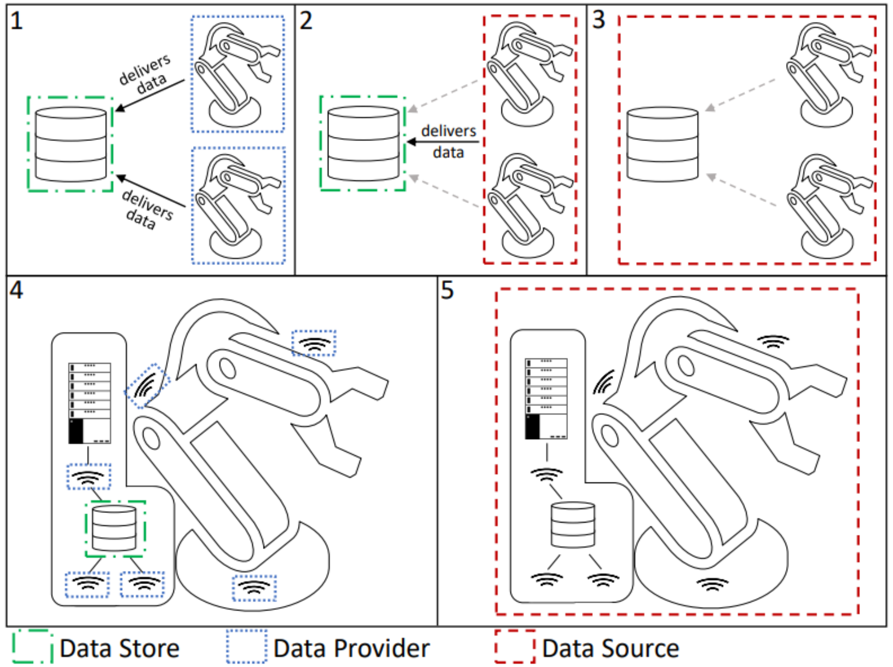
Machine learning-based detection

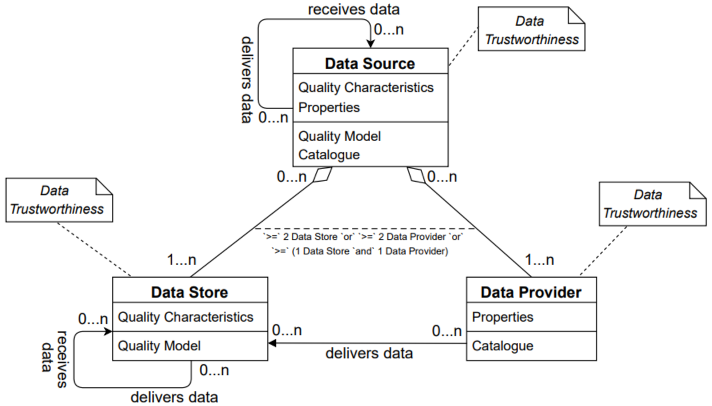
Data Source Quality Model

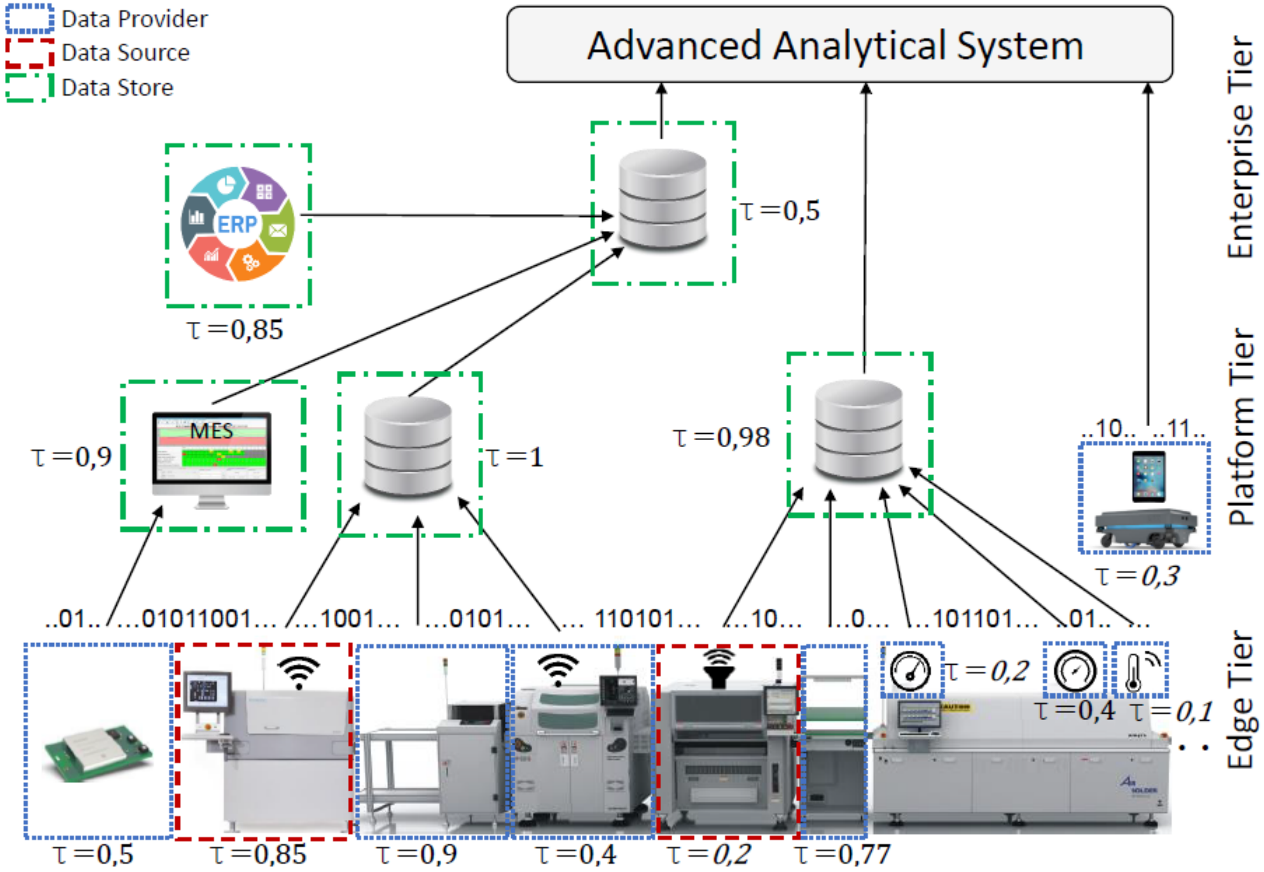
Quality Characteristics		Description	Properties and Subquality Factors
Representational Data Store Quality	***Representational Adequacy***	Degree to which a data store presents data in a *concise and organized way*.	Schema Minimality
			Schema Normalization
			Schema Pertinence
	Representational Consistency	Degree to which a data store presents data *always in the same format and compatible with previous data.*	Data Format Variety
			Data Type Variety
			Schema Change Proneness
	Understandability	Degree to which *users can understand the data* provided by a data store.	Data Format Complexity
			Documentation Degree
			Metadata Quality
			Schema Readability
Dynamical Data Store Quality	***Accessibility***	Degree to which data are *easily and quickly retrievable.*	Access Maturity
			Operability
			Retrievability
	Availability	Degree to *which data are available* from a data store.	Durability
			Fault Tolerance
			Recoverability
			Scalablity
			Uptime
	Security	Degree to which *access to data for unauthorized persons is restricted* by a data store.	Authentification/Encryption
			Authorization Policy
	Timeliness	Degree to which a data store *provides up-to-date data in a timely manner.*	Refresh Rate
			Response Time
Statical Data Store Quality	***Completeness***	Degree to which a data store is able to represent *every meaningful state of the real world.*	Schema Completeness
			Schema Correctness
	Contactability	Degree to which a data store provides *contact information for further inquiries.*	Support Degree
	Trustworthiness	Degree to which a *data store can be trusted*.	Complexity
			Data Governance
			Maturity
			Verifiability

universität innsbruck

Data Source, Provider, and Store







universität innsbruck

Data Pipeline Quality

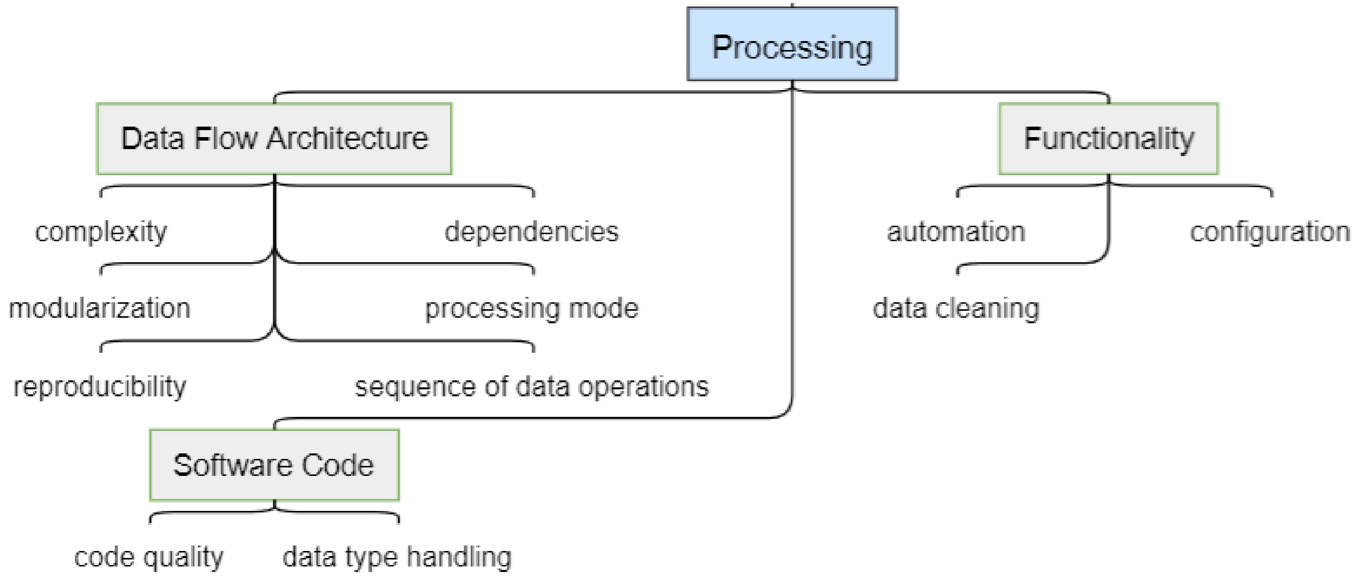
Issues of the implemented data pipeline that can affect the quality of the processed data

Code smells

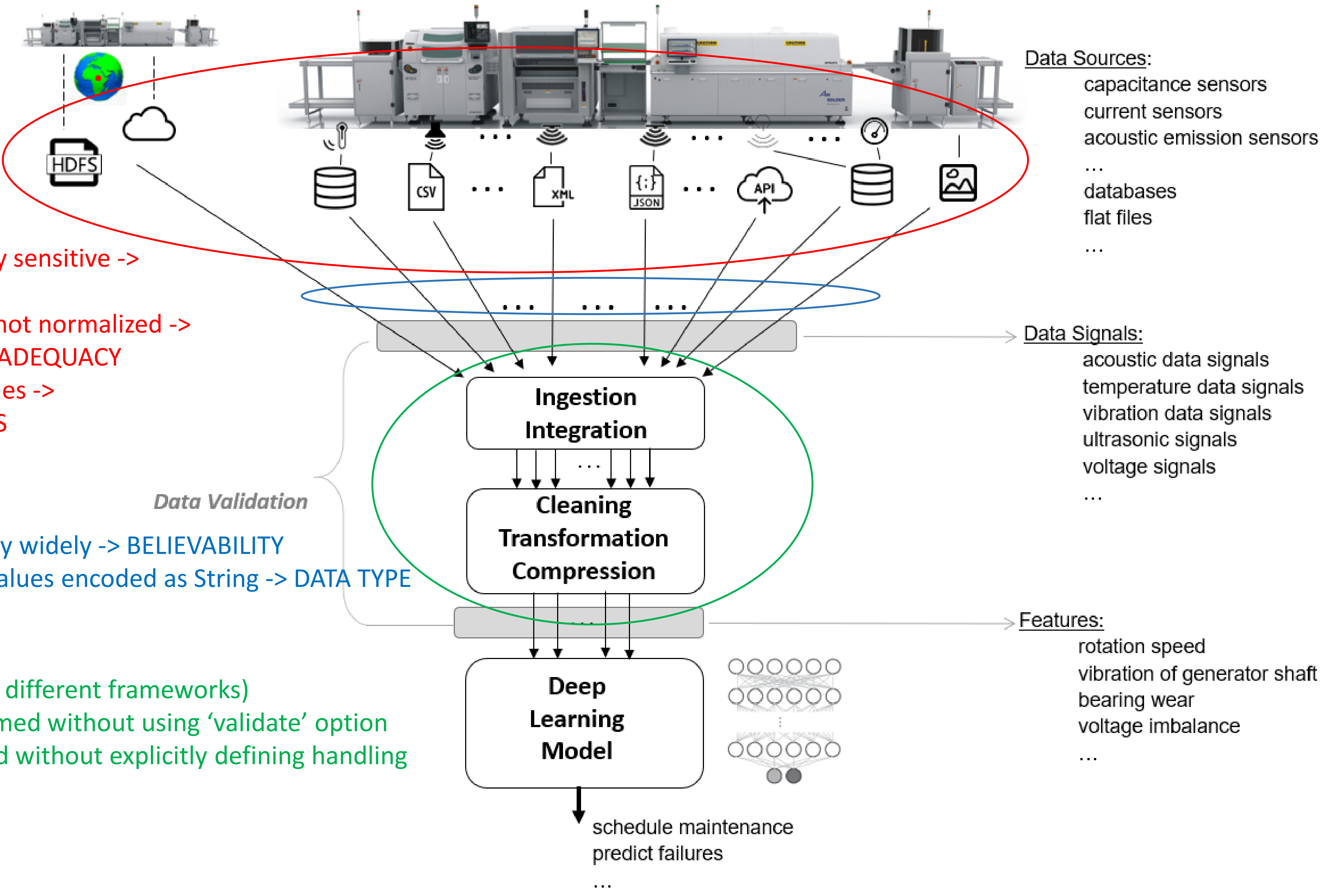
- libraries Pandas vs. Numpy handling missing values differently by default (e.g. pandas.DataFrame.max vs. numpy.amax)

```
DataFrame.max(self, axis=None, skipna=None, level=None, numeric_only=None, **kwargs)
```

skipna : *bool, default True*
Exclude NA/null values when computing the result.



Predictive Maintenance Example



Data Source Quality

- capacitance sensor very sensitive -> AVAILABILITY
- database of generator not normalized -> REPRESENTATIONAL ADEQUACY
- Personal data in XML files -> TRUSTHWORTHINESS
- …

Data Smells

- temperature values vary widely -> BELIEVABILITY
- timestamp of current values encoded as String -> DATA TYPE
- …

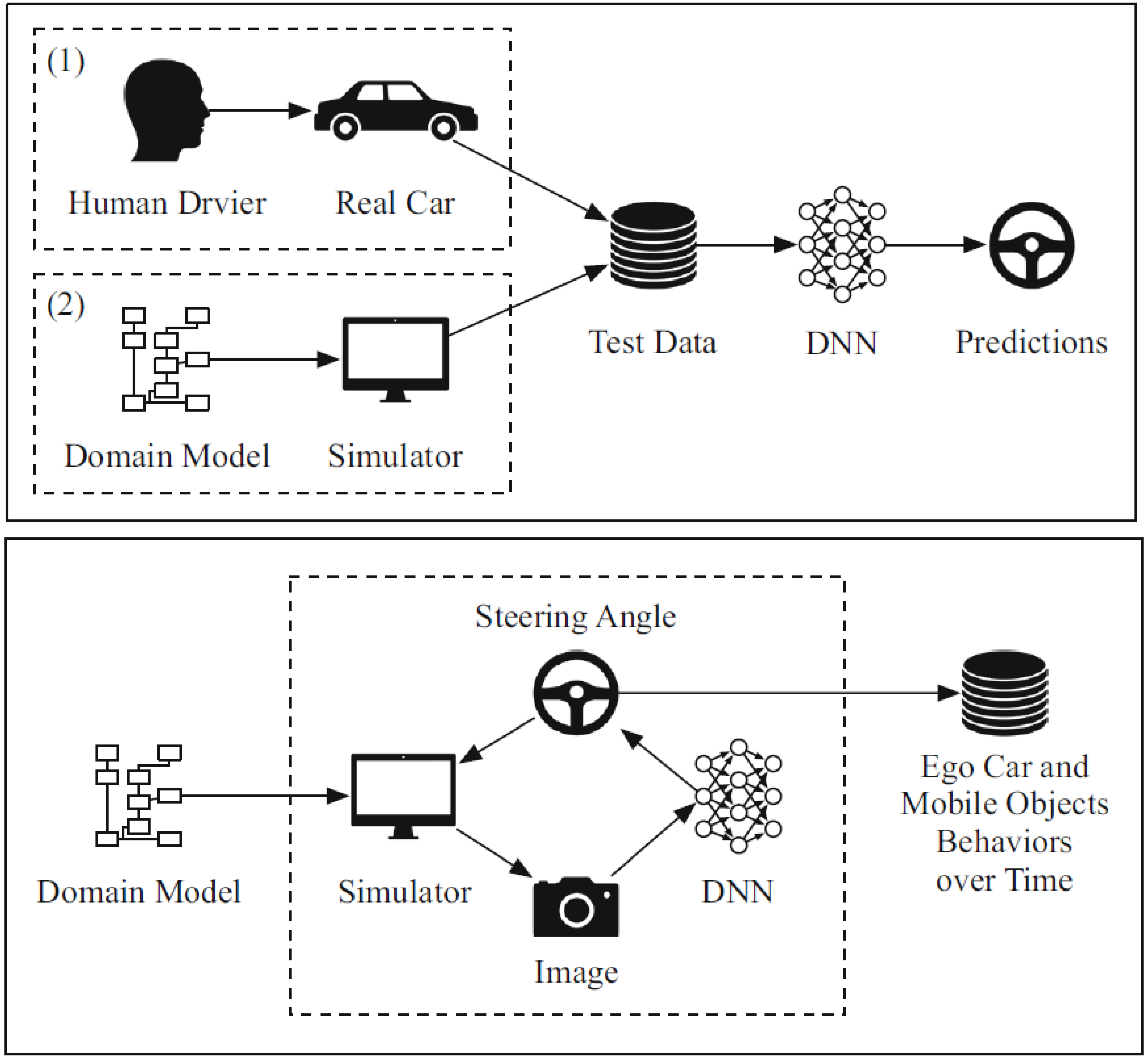
Data Pipeline Quality

- pipeline jungle (several different frameworks)
- vibration data transformed without using 'validate' option
- voltage data normalized without explicitly defining handling of missing values
- …

Agenda

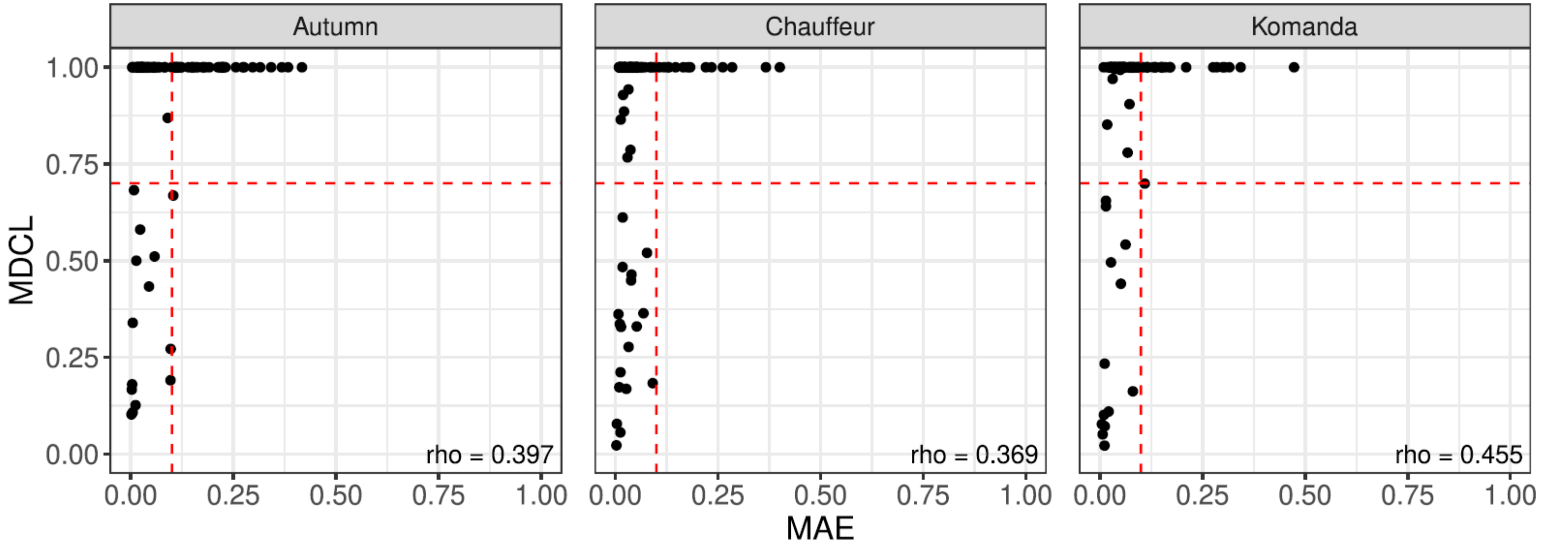
- Motivation
- Classical Risk-Based Testing
- Testing Machine-Learning-Based Systems
- Data Testing
- Online Testing

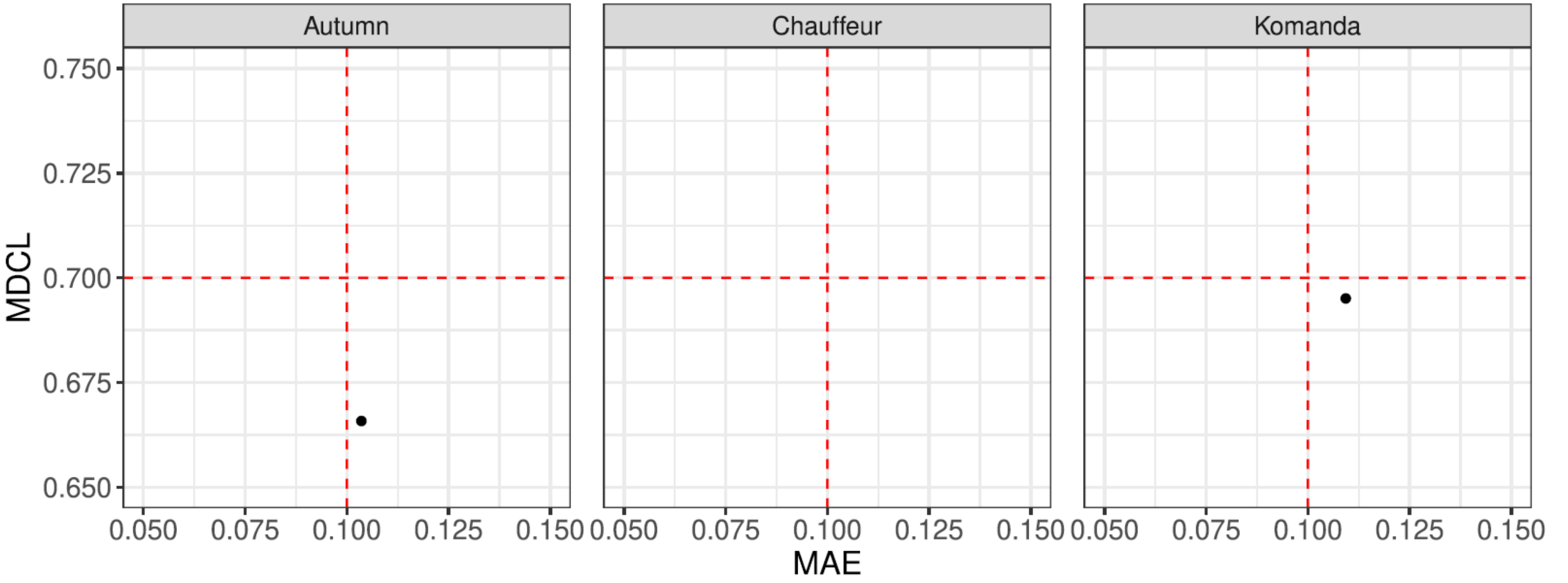
Offline and Online Testing



Haq, Fitash Ul, et al. *Can Offline Testing of Deep Neural Networks Replace Their Online Testing?*. Empirical Software Engineering 26:5, 1-30, 2021

Comparison Offline and Online Testing Results





Many safety violations identified by online testing could not be identified by offline testing

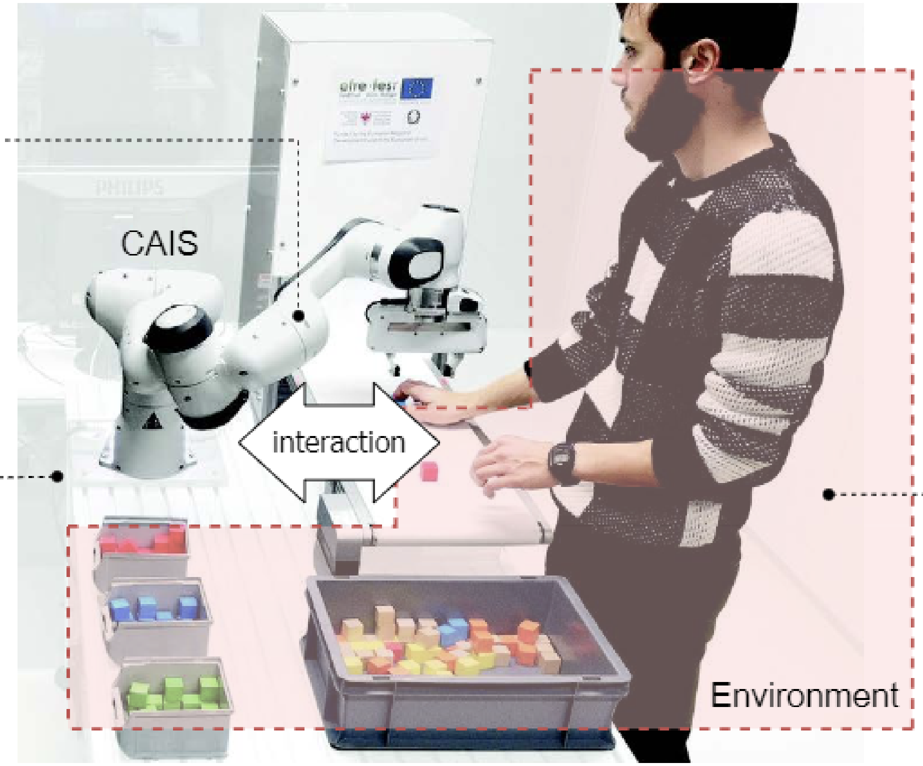
Offline testing cannot properly reveal safety violations

MDCL: Maximum Distance from Center of Lane (Online Testing)
MAE: Mean Absolute Error (Offline Testing)

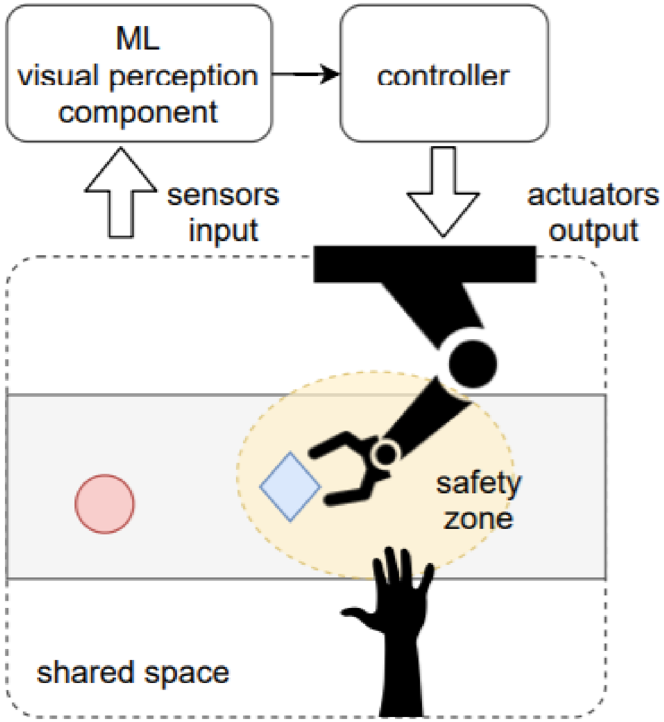
Collaborative AI Setting and Safety

Online learning:
object classification,
human classification,
motion direction,
motion speed

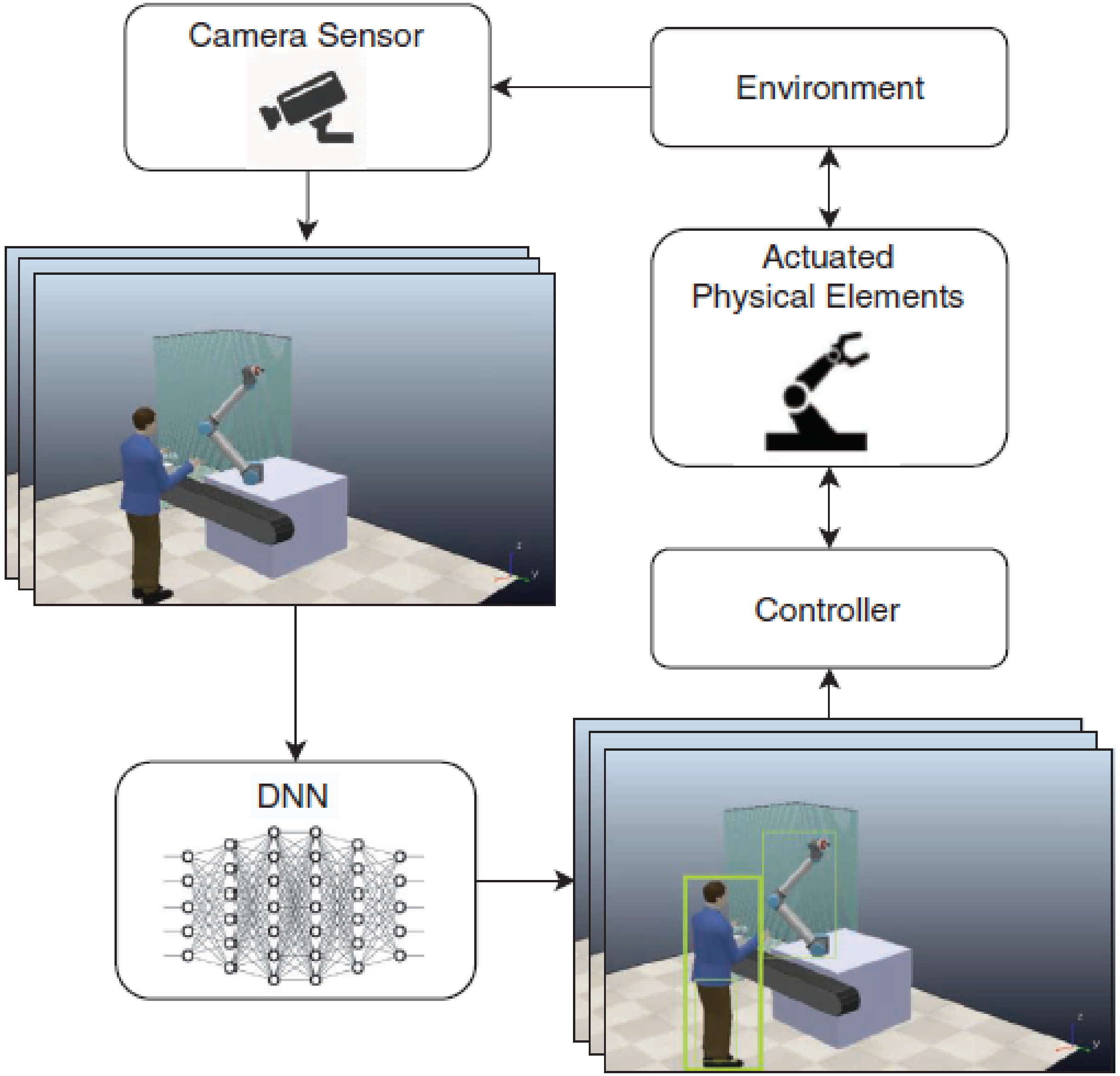
Risks:
protective distance violation,
injuring behavior,
robot unable to classify objects,
robot prevails over human needs



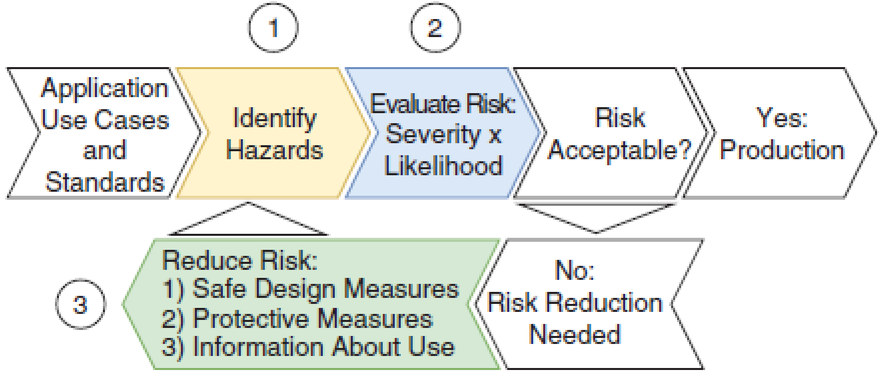
Uncertainties:
human position,
human motion speed,
human-background contrast,
luminance,
shape/color of objects

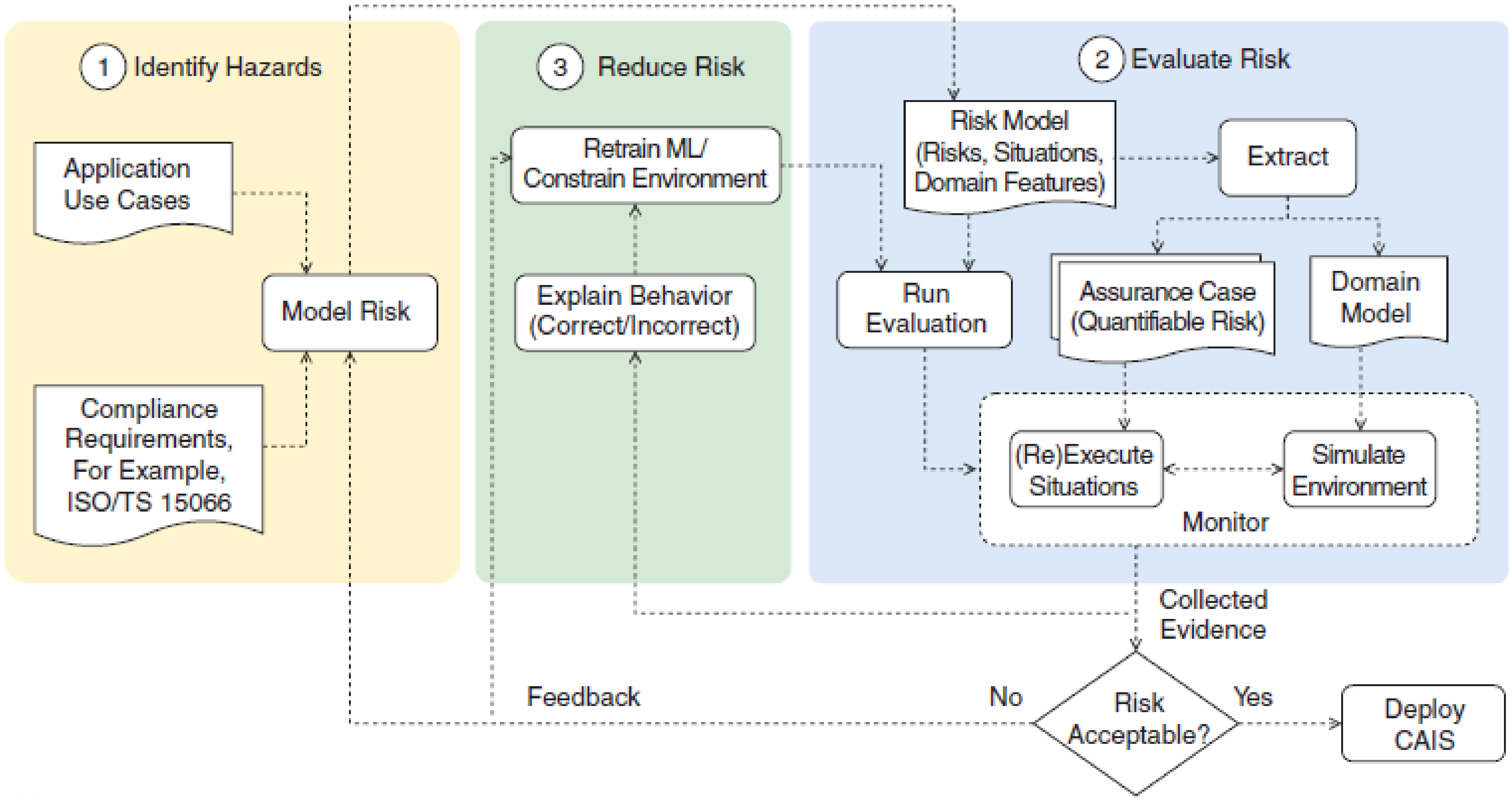


Setting for Online Testing

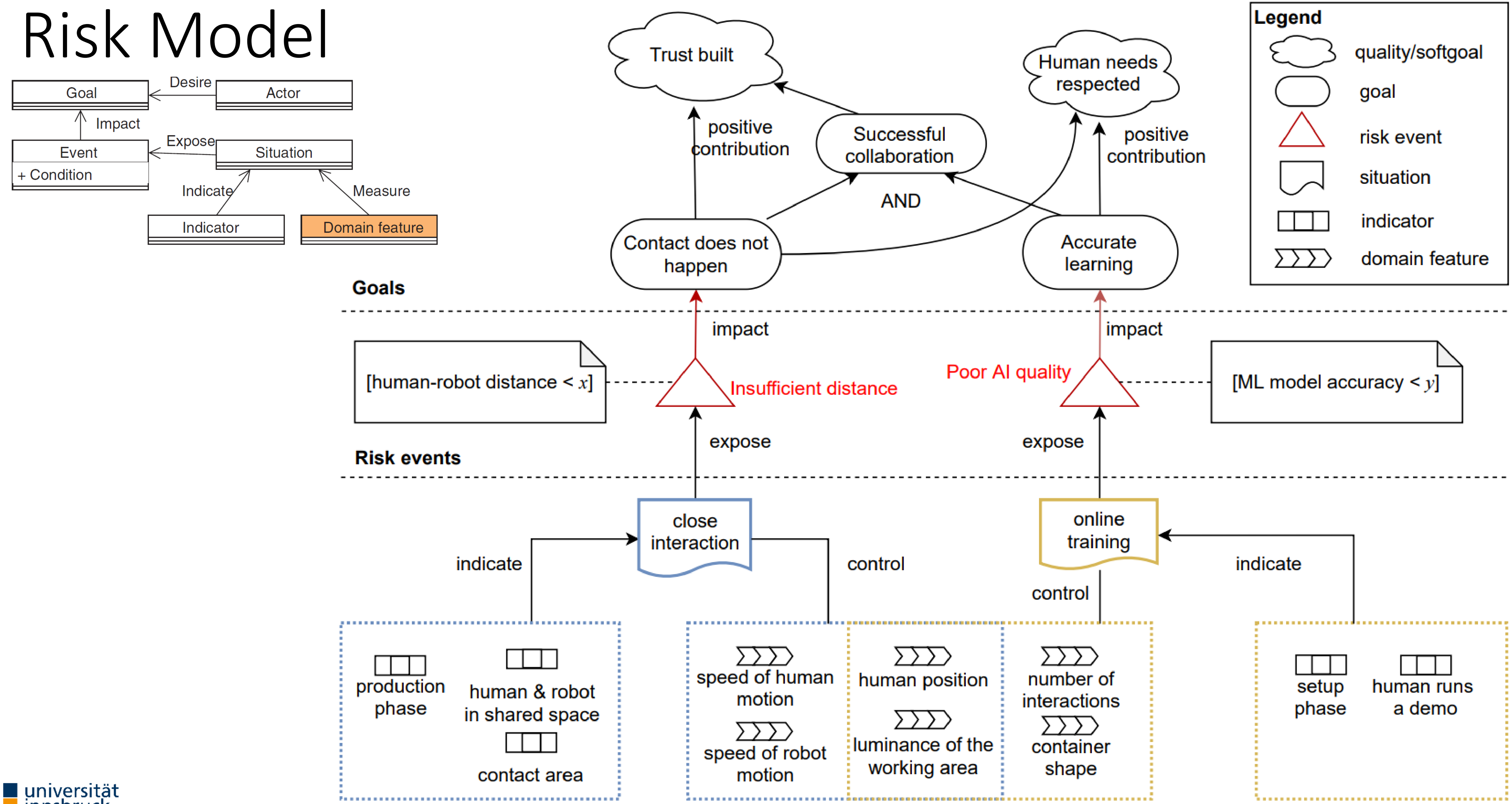


Risk-Driven Assurance Process





Risk Model


Goal
Desire
Actor
Impact
Event
+ Condition
Expose
Situation
Indicate
Measure
Indicator
Domain feature
Trust built
Human needs respected
positive contribution
Successful collaboration
positive contribution
AND
Contact does not happen
Accurate learning
Legend
quality/softgoal
goal
risk event
situation
indicator
domain feature
Goals
impact
impact
[human-robot distance < x]
Insufficient distance
Poor AI quality
[ML model accuracy < y]
expose
expose
Risk events
close interaction
online training
indicate
control
control
indicate
production phase
human & robot in shared space
contact area
speed of human motion
speed of robot motion
human position
luminance of the working area
number of interactions
container shape
setup phase
human runs a demo

Search Space Exploration and Risk Evaluation

1. [Start] Initialize simulator variables for path e.g., Given initial system states (s_0, n, U) and "m" number of runs such that:

s_0 = robot arm coordinates *(R [x, y, z]), human* position *(H [x, y, z]),*

$n = 0 : 0, 1, 2 \ldots n \subseteq m$

$U = []$

2. [Input space] "p" population of *key simulator features* (suitable solutions for the problem) which is the expected output:

$F' \leq F$ *(e.g., environment lighting (L [h, s, l]))*

Path trace during its execution $(v = (v_1, \ldots, v_p))$*, implicitly determined within simulator*

3. [Fitness] Determine fitness function *f(x)* of each arrangement in the population that satisifies a value ϕ along the path *v,* e.g.,

$d(R, H) \leq \phi$

4.[Update] The simulation rewards those path functions where an unsafe state is reached for instance with a "1" or otherwise "0", returns:

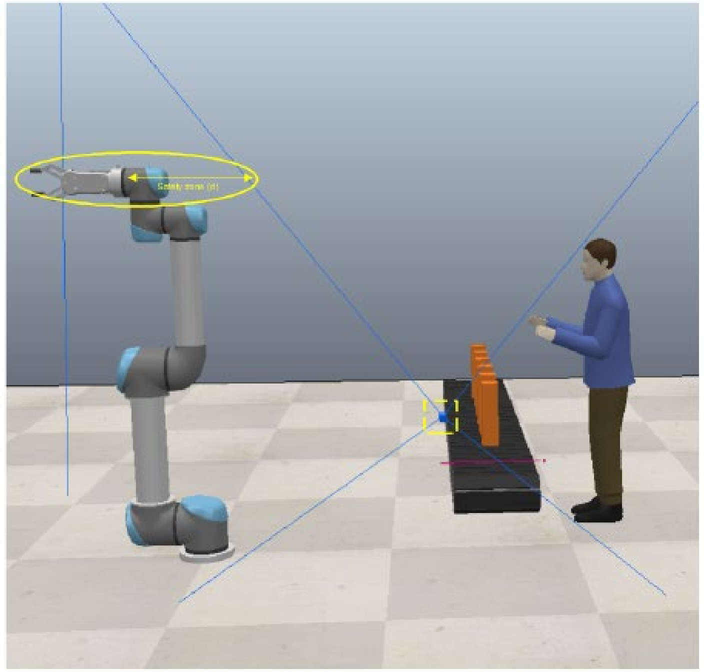
feature $F' \subseteq U$

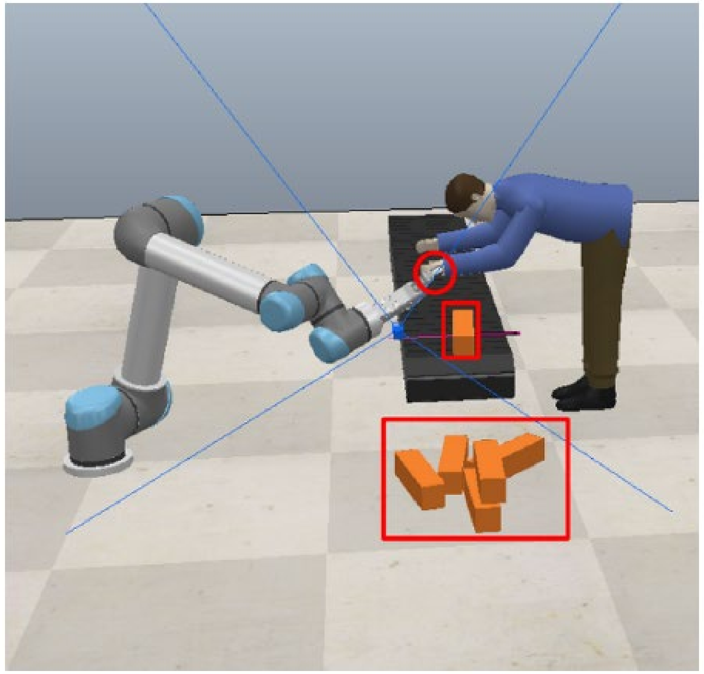
$$Ur = \begin{cases} 1, & dvi(R,H) \leq \phi \\ 0, & otherwise \end{cases}$$

5. [Increment] test case n = n + 1

6. [Repeat/Loop] : Step 2 until n = m

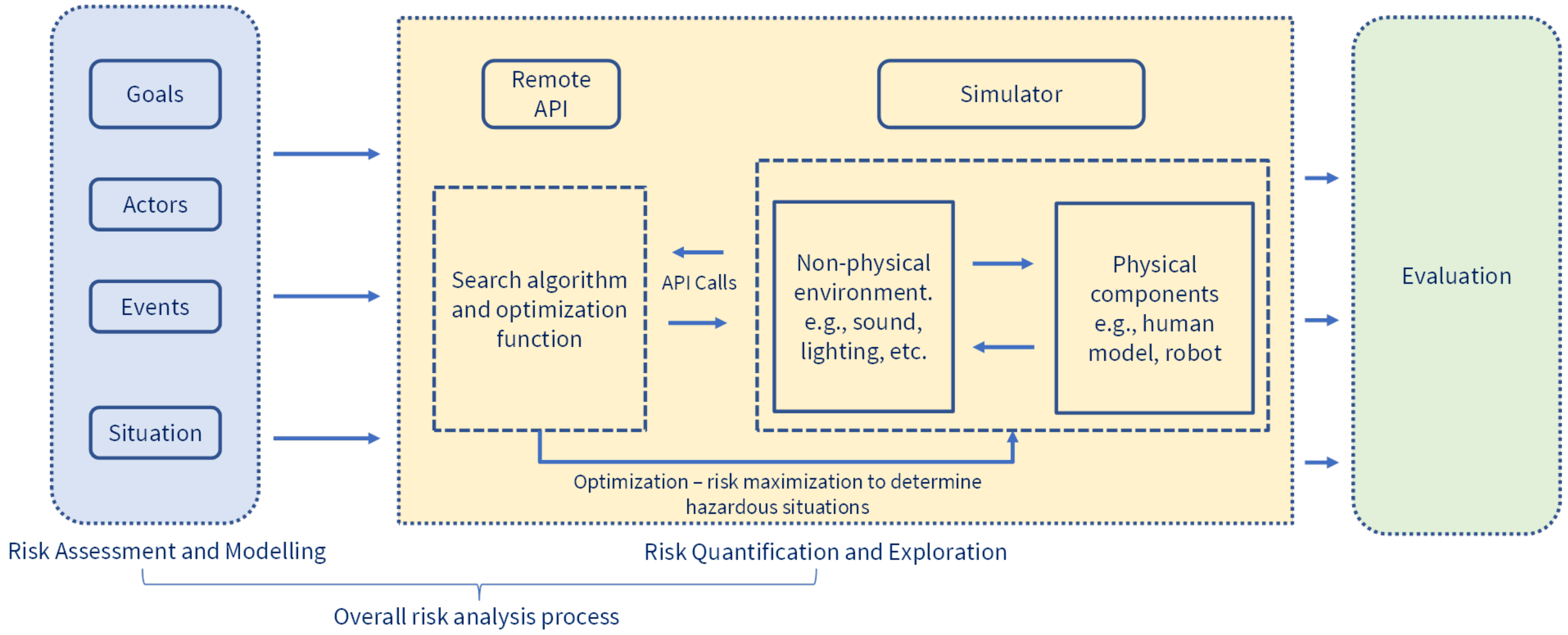
7. Return outcome U



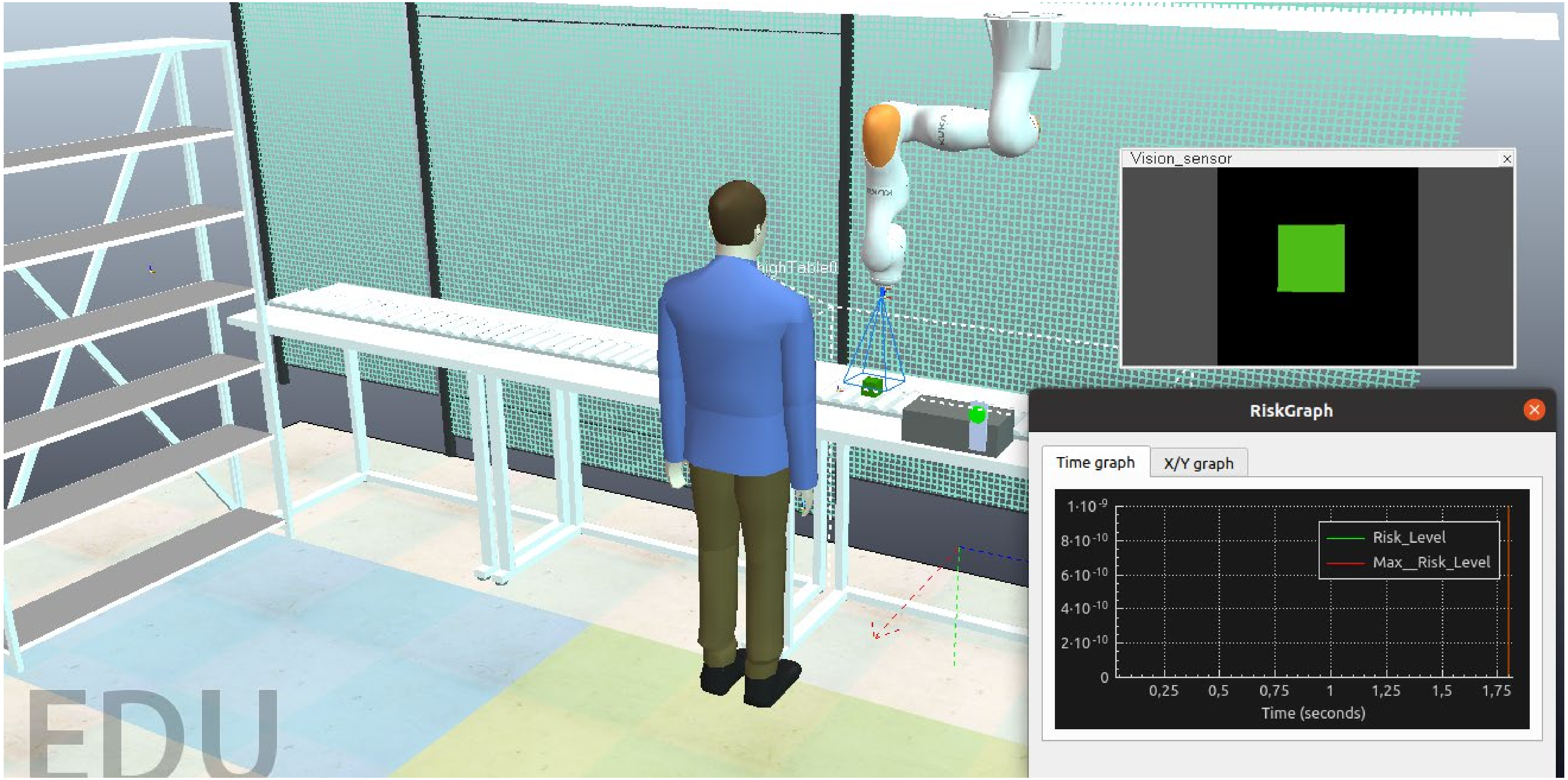


Objective Functions:
Minimum distance between human and robot arm
Relative speed of human and robot arm

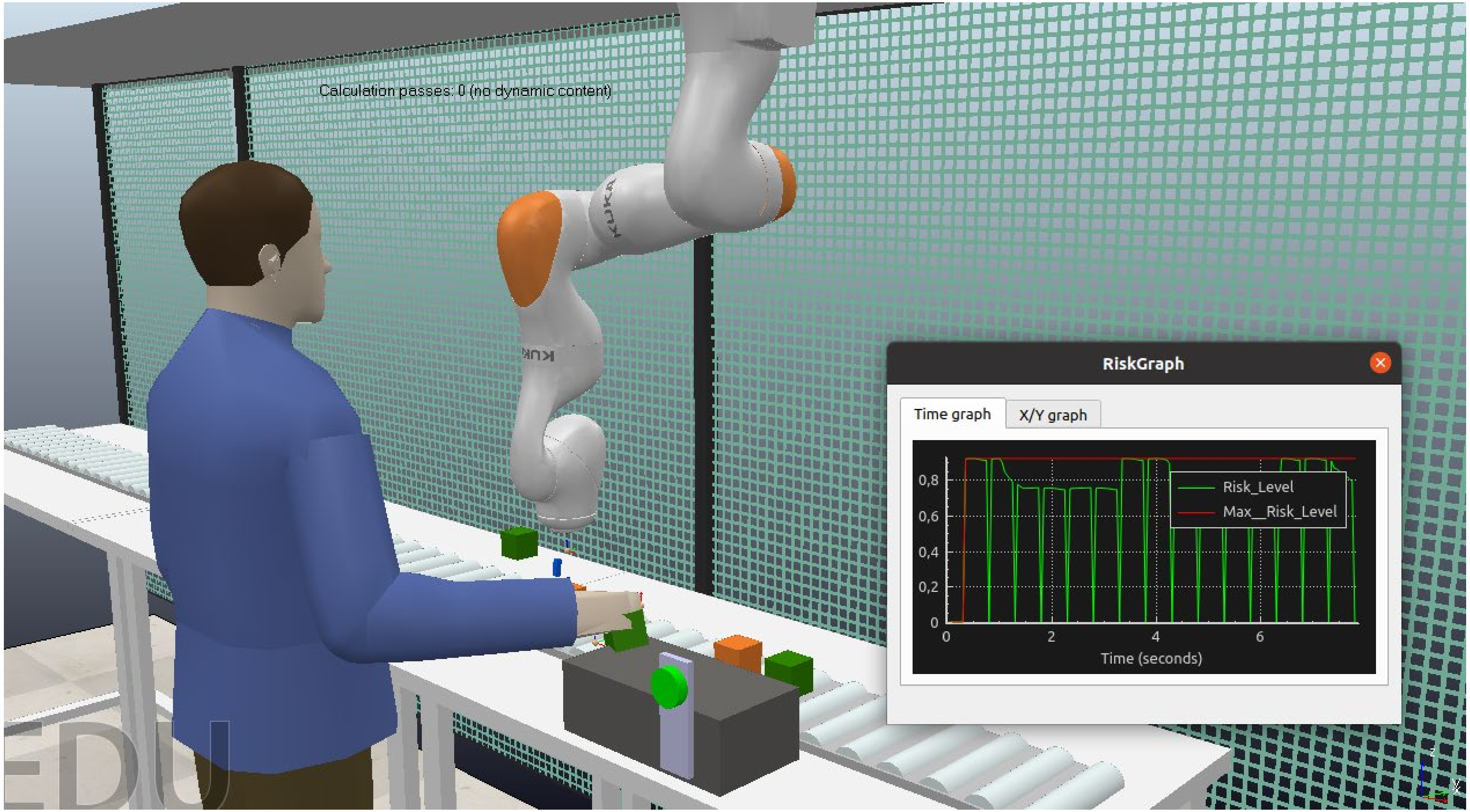
Test Environment



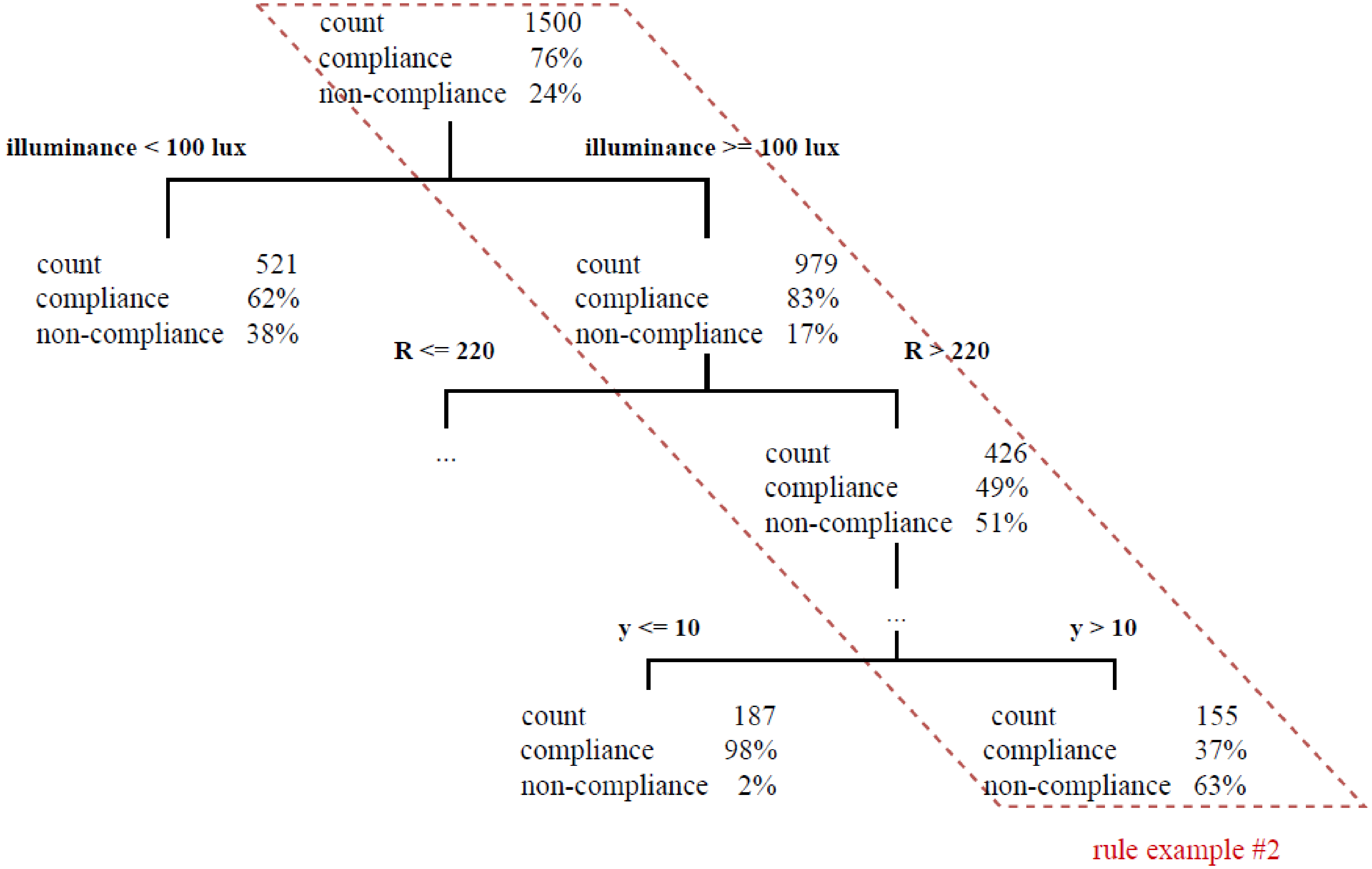
Simulation in CoppeliaSim (1/2)



Simulation in CoppeliaSim (2/2)



Risk Quantification, Decision Tree and Rule Extraction



rule example #	domain features illuminance (lux)	operator arms color (R,G,B)	operator position (x,y)
1	<100	-	-
2	>100	R >220, G >236, B >200	x >180, y >10

Summary

- Motivation
- Classical Risk-Based Testing
- Testing Machine-Learning-Based AI Systems
- Data Testing
- Online Testing

References

[1] Zhang, Jie M., et al. *Machine learning testing: Survey, landscapes and horizons*. IEEE Transactions on Software Engineering, 2020

[2] Riccio, Vincenzo, et al. *Testing machine learning based systems: a systematic mapping*. Empirical Software Engineering 25.6 (2020): 5193-5254.

[3] Felderer, M., Schieferdecker, I. *A taxonomy of risk-based testing*. Software Tools for Technology Transfer, 16(5), 559-568,Springer, 2014

[4] Foidl, H., Felderer, M.: *Integrating software quality models into risk-based testing*. Software Quality Journal, 26(2), 809-847, 2018

[5] Foidl, H., Felderer, M., Ramler, R. *Data Smells: Categories, Causes and Consequences, and Detection of Suspicious Data in AI-based Systems*. 1st International Conference on AI Engineering (CAIN 2022), ACM, 2022

[6] Haq, Fitash Ul, et al. *Can Offline Testing of Deep Neural Networks Replace Their Online Testing?*. Empirical Software Engineering 26:5, 1-30, 2021

[7] Adigun, Jubril Gbolahan, et al. *Collaborative Artificial Intelligence Needs Stronger Assurances Driven by Risks*. Computer 55:3, 52-63, 2022

[8] Harel-Canada, F., et al. *Is neuron coverage a meaningful measure for testing deep neural networks*?. Proceedings of the 28th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering. 2020

universität innsbruck

Testing the Untestable: Risikobasierte Qualitätssicherung für Machine-Learning Systeme

Prof. Dr. Michael Felderer

Department of Computer Science

Universität Innsbruck

Austria

michael.felderer@uibk.ac.at

@mfelderer