Application of Combinatorial Testing in Testing ML Systems

Workshop on Combinatorial Testing for AI-enabled Systems

September 4, 2024

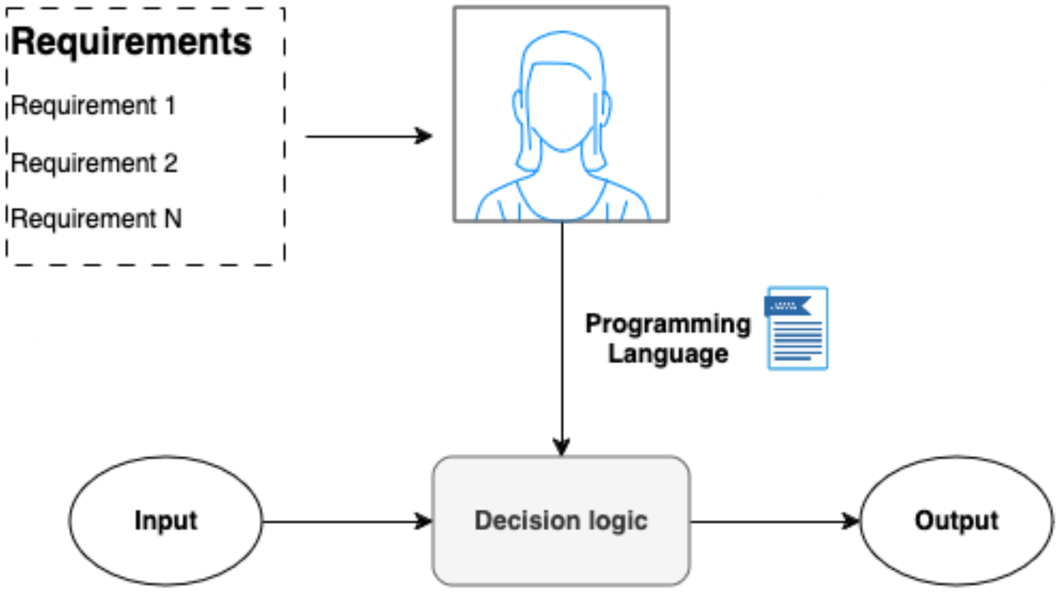
Jaganmohan Chandrasekaran, Ph.D.
Research Assistant Professor
Sanghani Center for AI & Data Analytics
Virginia Tech | jagan@vt.edu

Outline

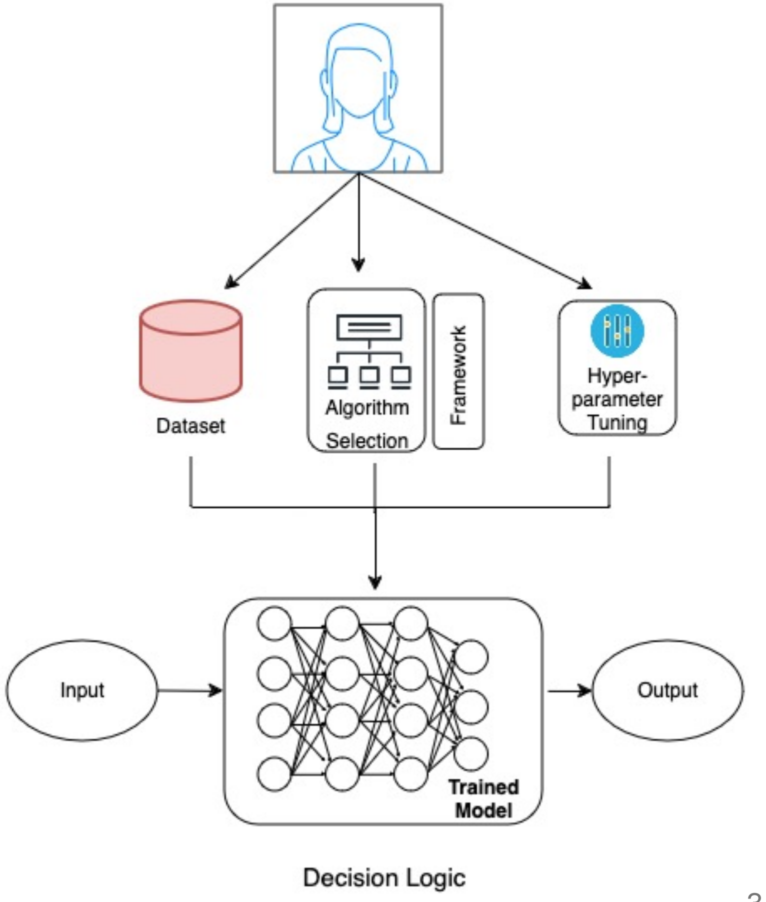
- Introduction
- Challenges in Testing ML Systems
- Combinatorial Testing for ML Systems
- Overview of Combinatorial Testing in Evaluating ML Systems
- Challenges and Future Directions

Traditional Software vs. ML-enabled Software Development

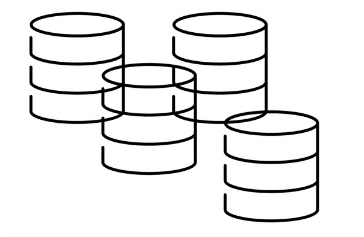
Traditional Software Development



ML-enabled Software Development

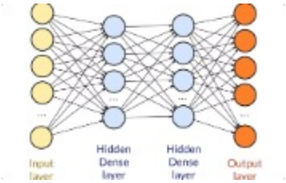


Unique Testing Challenges in ML



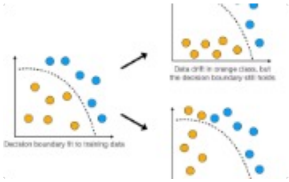
Data-Intensive

AI/ML systems rely heavily on data, significantly influencing behavior.



Complex Decision Logic

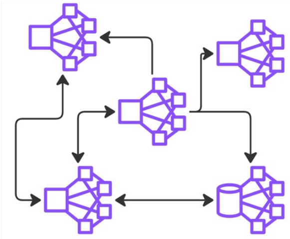
The internal decision logic is generally opaque and not easily interpretable by humans.



Behavior

Non-deterministic behavior

Evolving behavior



Large Input Space

Decision logic is derived from the patterns underlying the training dataset, which is usually large in size.

Lack of Test Oracle

In most cases, test instances will lack a reliable test oracle, and desired outcomes are unknown beforehand.

Testing beyond Correctness

Fairness, safety, and trustworthiness are critical. Specialized testing techniques are required.

Outline

- Introduction
- Challenges in Testing ML Systems
- Combinatorial Testing for ML Systems
- Overview of Combinatorial Testing in Evaluating ML Systems
- Challenges and Future Directions

Why Combinatorial Testing for ML?

Combinatorial testing, a black box test generation technique, focuses on systematically testing the interactions among the system's different parameters with a relatively smaller number of tests [1].

Enables the detection of interaction faults, including rare combinations that can lead to failures in software systems [2].

Why Combinatorial Testing for ML (2)?

Combinatorial testing, a black box test generation technique, focuses on systematically testing the interactions among the system's different parameters with a relatively smaller number of tests.

Efficiently explore the large input space of ML models by systematically testing different feature interactions with a relatively small number of tests.

Enables the detection of interaction faults, including rare combinations that can lead to failures in software systems.

Help uncover rare feature combinations and edge case scenarios, thus reducing the risk of undetected faults.

Steps in Applying Combinatorial Testing to ML

Input Parameter Modeling

- Parameters
- Values
- Constraints

Step 1

→

Generate Abstract Tests

- Size of "t"
- Generate t-way abstract tests

Step 2

→

Generate Concrete Tests

- Convert abstract to concrete tests
- Test oracle

Step 3

→

Test execution

- Metrics

Step 4

Applying Combinatorial Testing - Image Dataset

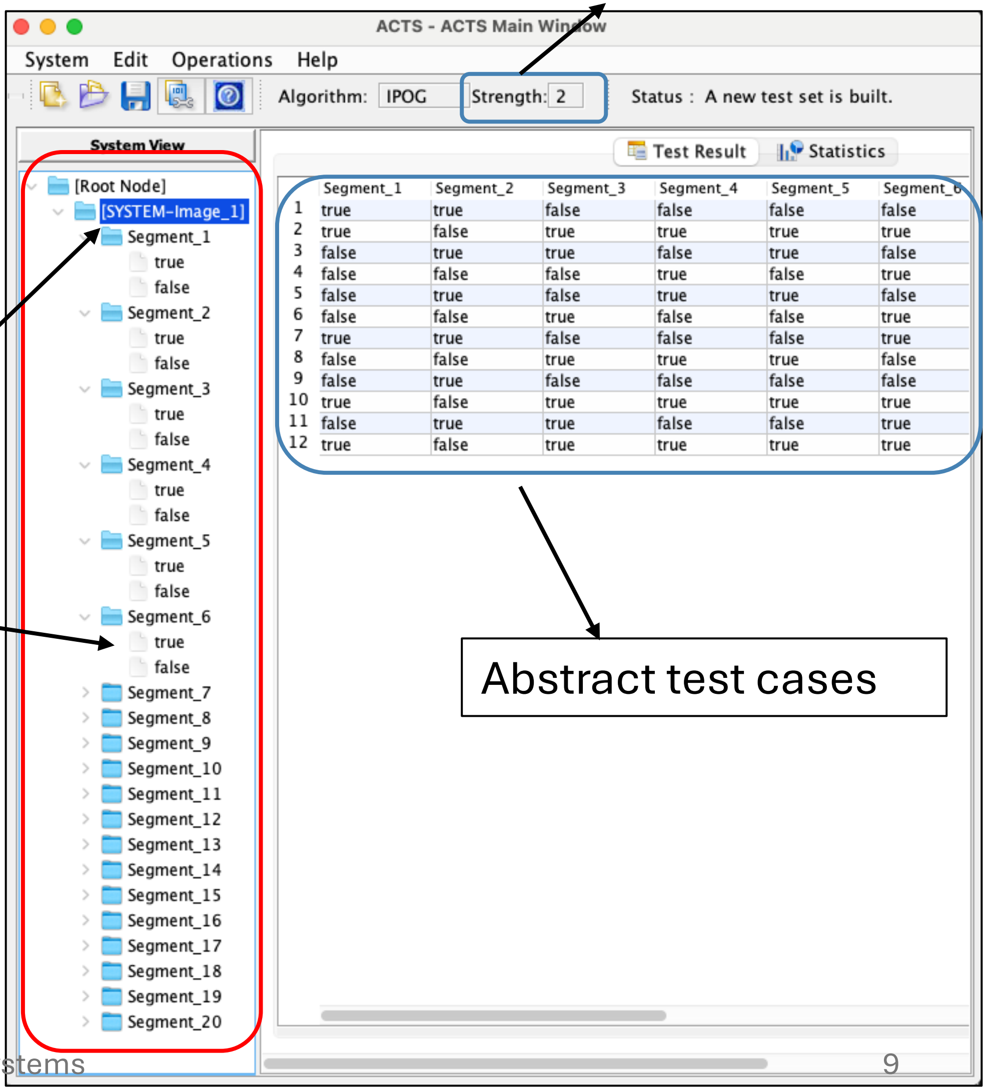
1 Segmentation

2

Number of segments = 20

Parameter = Segment
Values = [True, False]

Value of t = 2



	Segment_1	Segment_2	Segment_3	Segment_4	Segment_5	Segment_6
1	true	true	false	false	false	false
2	true	false	true	true	true	true
3	false	true	true	false	true	false
4	false	false	false	true	false	true
5	false	true	false	true	true	false
6	false	false	true	false	false	true
7	true	true	false	false	false	true
8	false	false	true	true	true	false
9	false	true	false	false	false	false
10	true	false	true	true	true	true
11	false	true	true	false	false	true
12	true	false	true	true	true	true

Abstract test cases

Generating test cases - Image Dataset

Test case 1

Test case 2

Test case 3

Test case 4

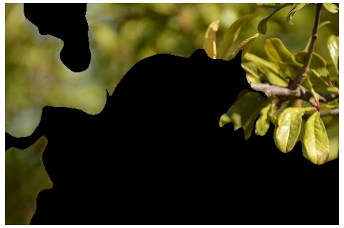
Test case 5

Test case 6

Test case 7

Test case 8

Test case 9



Test case 10

Test case 11

Test case 12

Applying Combinatorial Testing - Tabular Dataset

- Parameters
  - Map all attributes, excluding the class label.
- Identify values for each parameter based on the attribute's data type
  - **Categorical** attributes: Map unique values as representative values
  - **Numerical** attributes: Identify representative values through discretization
- Identify constraints from the training dataset

1	age	workclass	education	education_nı	marital_stat	occupation	relationship	race	sex
2	39	State-gov	Bachelors	13	Never-marri	Adm-clerical	Not-in-famil	White	Male
3	50	Self-emp-no	Bachelors	13	Married-civ-:	Exec-manag	Husband	White	Male
4	38	Private	HS-grad	9	Divorced	Handlers-cle	Not-in-famil	White	Male
5	53	Private	11th	7	Married-civ-:	Handlers-cle	Husband	Black	Male
6	28	Private	Bachelors	13	Married-civ-:	Prof-specialt	Wife	Black	Female
7	37	Private	Masters	14	Married-civ-:	Exec-manag	Wife	White	Female
8	49	Private	9th	5	Married-spo	Other-servic	Not-in-famil	Black	Female
9	52	Self-emp-no	HS-grad	9	Married-civ-:	Exec-manag	Husband	White	Male
10	31	Private	Masters	14	Never-marri	Prof-specialt	Not-in-famil	White	Female
11	42	Private	Bachelors	13	Married-civ-:	Exec-manag	Husband	White	Male
12	37	Private	Some-colleg	10	Married-civ-:	Exec-manag	Husband	Black	Male
13	30	State-gov	Bachelors	13	Married-civ-:	Prof-specialt	Husband	Asian-Pac-Isl	Male
14	23	Private	Bachelors	13	Never-marri	Adm-clerical	Own-child	White	Female
15	32	Private	Assoc-acdm	12	Never-marri	Sales	Not-in-famil	Black	Male
16	40	Private	Assoc-voc	11	Married-civ-:	Craft-repair	Husband	Asian-Pac-Isl	Male
17	34	Private	7th-8th	4	Married-civ-:	Transport-m	Husband	Amer-Indian	Male
18	25	Self-emp-no	HS-grad	9	Never-marri	Farming-fish	Own-child	White	Male
19	32	Private	HS-grad	9	Never-marri	Machine-op-	Unmarried	White	Male

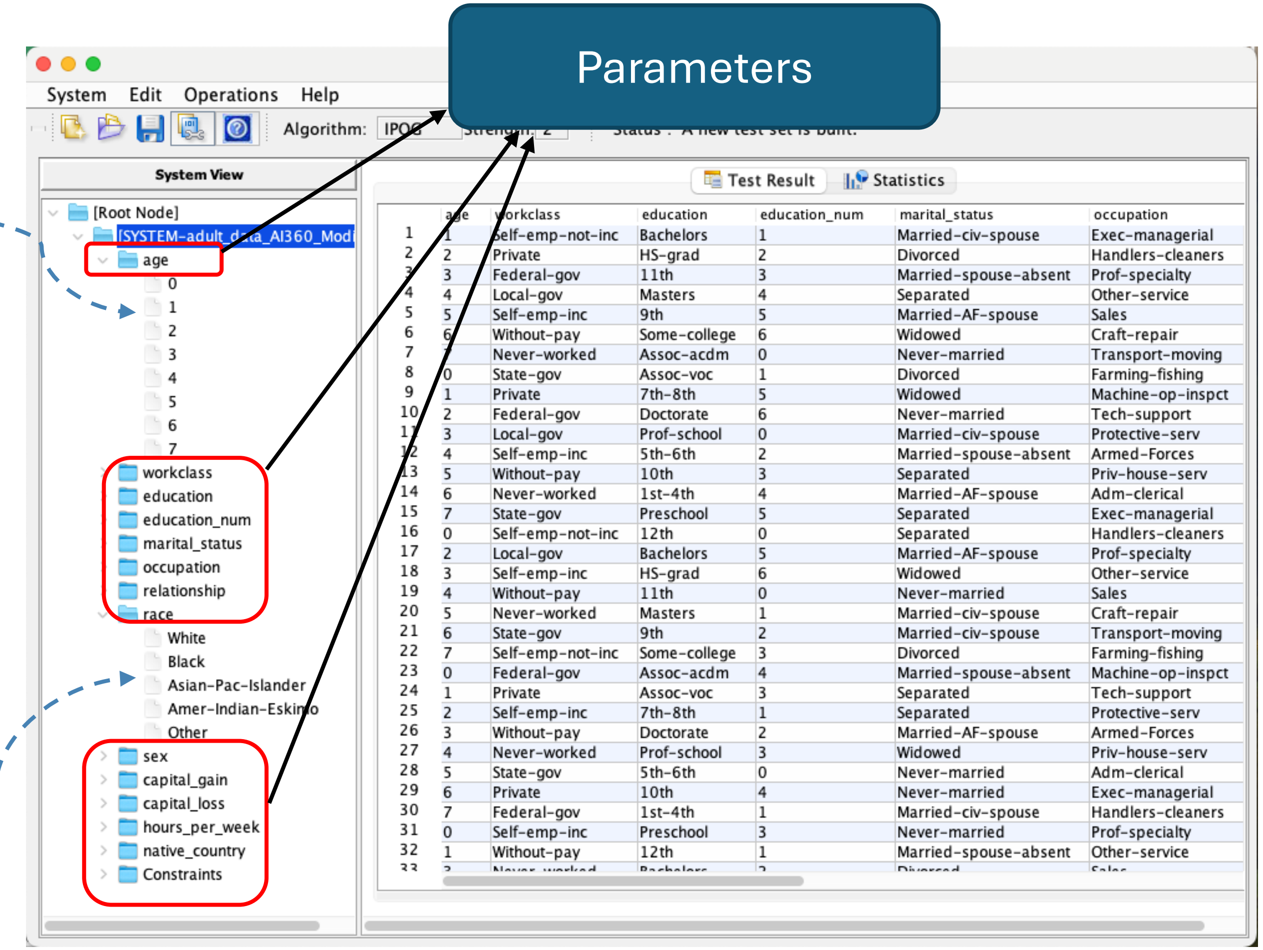
A snippet of the Adult Income Dataset [12, 13]

Designing an Input Parameter Model - Tabular Dataset

Discretized Numerical Values

	age	workclass	education	education_n	marital_stat	occupation	relationship	race	sex
2	39	State-gov	Bachelors	13	Never-marri	Adm-clerical	Not-in-famil	White	Male
3	50	Self-emp-no	Bachelors	13	Married-civ-	Exec-manag	Husband	White	Male
4	38	Private	HS-grad	9	Divorced	Handlers-cle	Not-in-famil	White	Male
5	53	Private	11th	7	Married-civ-	Handlers-cle	Husband	Black	Male
6	28	Private	Bachelors	13	Married-civ-	Prof-specialt	Wife	Black	Female
7	37	Private	Masters	14	Married-civ-	Exec-manag	Wife	White	Female
8	49	Private	9th	5	Married-spo	Other-servic	Not-in-famil	Black	Female
9	52	Self-emp-no	HS-grad	9	Married-civ-	Exec-manag	Husband	White	Male
10	31	Private	Masters	14	Never-marri	Prof-specialt	Not-in-famil	White	Female
11	42	Private	Bachelors	13	Married-civ-	Exec-manag	Husband	White	Male
12	37	Private	Some-colleg	10	Married-civ-	Exec-manag	Husband	Black	Male
13	30	State-gov	Bachelors	13	Married-civ-	Prof-specialt	Husband	Asian-Pac-Is	Male
14	23	Private	Bachelors	13	Never-marri	Adm-clerical	Own-child	White	Female
15	32	Private	Assoc-acdm	12	Never-marri	Sales	Not-in-famil	Black	Male
16	40	Private	Assoc-voc	11	Married-civ-	Craft-repair	Husband	Asian-Pac-Is	Male
17	34	Private	7th-8th	4	Married-civ-	Transport-m	Husband	Amer-Indian	Male
18	25	Self-emp-no	HS-grad	9	Never-marri	Farming-fish	Own-child	White	Male
19	32	Private	HS-grad	9	Never-marri	Machine-op-	Unmarried	White	Male

Categorical Values

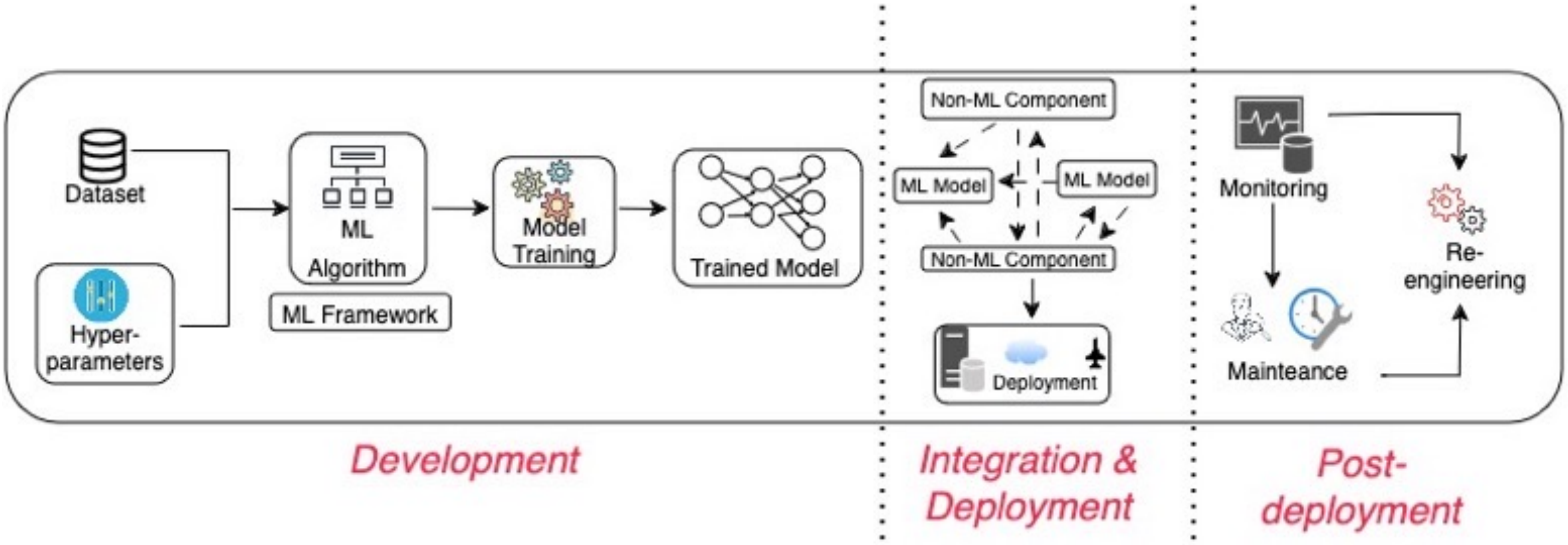


	age	workclass	education	education_num	marital_status	occupation
1	1	Self-emp-not-inc	Bachelors	1	Married-civ-spouse	Exec-managerial
2	2	Private	HS-grad	2	Divorced	Handlers-cleaners
3	3	Federal-gov	11th	3	Married-spouse-absent	Prof-specialty
4	4	Local-gov	Masters	4	Separated	Other-service
5	5	Self-emp-inc	9th	5	Married-AF-spouse	Sales
6	6	Without-pay	Some-college	6	Widowed	Craft-repair
7	7	Never-worked	Assoc-acdm	0	Never-married	Transport-moving
8	0	State-gov	Assoc-voc	1	Divorced	Farming-fishing
9	1	Private	7th-8th	5	Widowed	Machine-op-inspct
10	2	Federal-gov	Doctorate	6	Never-married	Tech-support
11	3	Local-gov	Prof-school	0	Married-civ-spouse	Protective-serv
12	4	Self-emp-inc	5th-6th	2	Married-spouse-absent	Armed-Forces
13	5	Without-pay	10th	3	Separated	Priv-house-serv
14	6	Never-worked	1st-4th	4	Married-AF-spouse	Adm-clerical
15	7	State-gov	Preschool	5	Separated	Exec-managerial
16	0	Self-emp-not-inc	12th	0	Separated	Handlers-cleaners
17	2	Local-gov	Bachelors	5	Married-AF-spouse	Prof-specialty
18	3	Self-emp-inc	HS-grad	6	Widowed	Other-service
19	4	Without-pay	11th	0	Never-married	Sales
20	5	Never-worked	Masters	1	Married-civ-spouse	Craft-repair
21	6	State-gov	9th	2	Married-civ-spouse	Transport-moving
22	7	Self-emp-not-inc	Some-college	3	Divorced	Farming-fishing
23	0	Federal-gov	Assoc-acdm	4	Married-spouse-absent	Machine-op-inspct
24	1	Private	Assoc-voc	3	Separated	Tech-support
25	2	Self-emp-inc	7th-8th	1	Separated	Protective-serv
26	3	Without-pay	Doctorate	2	Married-AF-spouse	Armed-Forces
27	4	Never-worked	Prof-school	3	Widowed	Priv-house-serv
28	5	State-gov	5th-6th	0	Never-married	Adm-clerical
29	6	Private	10th	4	Never-married	Exec-managerial
30	7	Federal-gov	1st-4th	1	Married-civ-spouse	Handlers-cleaners
31	0	Self-emp-inc	Preschool	3	Never-married	Prof-specialty
32	1	Without-pay	12th	1	Married-spouse-absent	Other-service

Outline

- Introduction
- Challenges in Testing ML Systems
- Combinatorial Testing for ML Systems
- Overview of Combinatorial Testing in Evaluating ML Systems
- Challenges and Future Directions

Lifecycle - ML-enabled software system

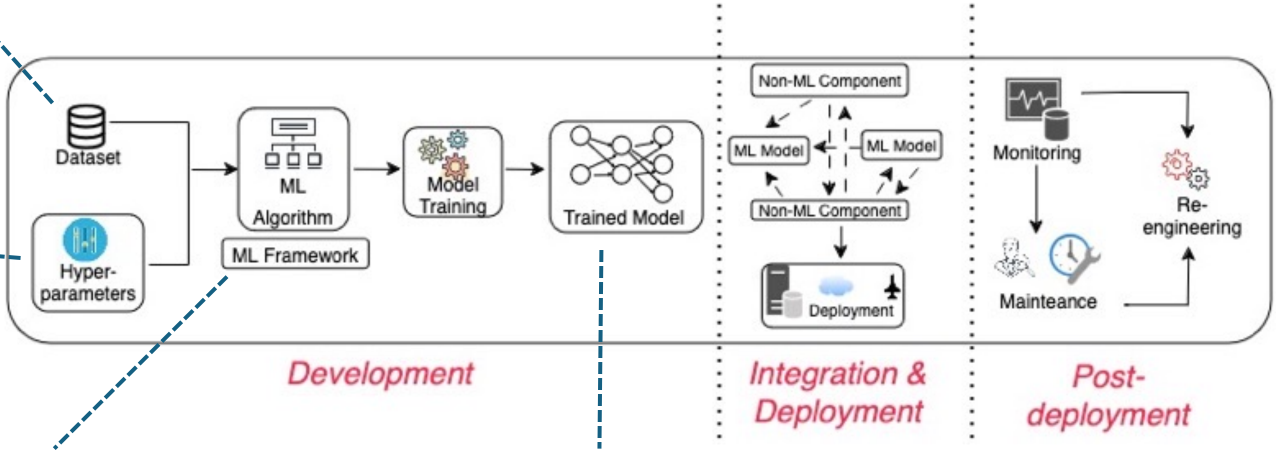


Application of Combinatorial Testing in the ML Lifecycle

Synthetic data generation [5, 10]

Coverage Criterion [11]

Hyperparameter Optimization [9]



ML algorithm T&E [16]

T&E of pre-trained Model [1, 3, 4, 5, 6, 7, 8, 15]

Correctness

Explainability

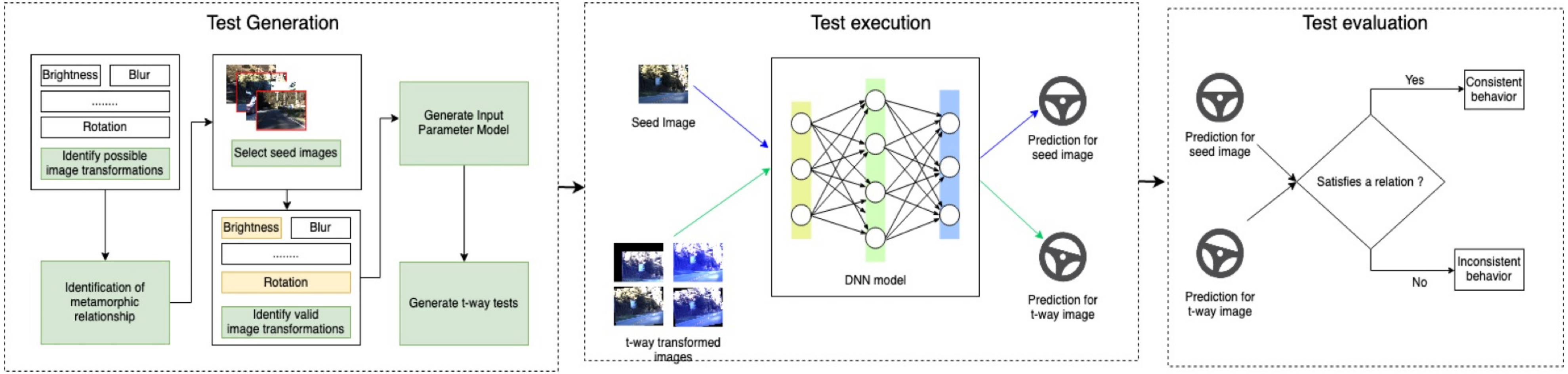
Fairness

Testing pre-trained Deep Neural Network models

- Generate t-way synthetic images to test Deep Neural Networks (DNNs) used in self-driving cars
- Each synthetic image is a combination of image transformations.

A set of image transformations

Transformations		*Values*
Blur	Averaging	3x3, 4x4, 5x5, 6x6
	Gaussian	3x3, 5x5, 7x7
	Median	3, 5
	Bilateral	(9, 75, 75)
Brightness		10, 20, 30, 40, 50, 60, 70, 80, 90, 100
Contrast		1.2, 1.4, 1.6, 1.8, 2.0, 2.2., 2.4, 2.6, 2.8, 3.0
Rotation		3, 6, 9, 12, 15, 18, 21, 24, 27, 30
Scale		1.5, 2.0, 2.5, 3.0, 3.5, 4.0, 4.5, 5.0, 5.5, 6.0
Shear (Horizontal)		0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0
Translation		(10,10), (20,20), (30,30), (40,40), (50,50), (60,60), (70,70), (80,80), (90,90), (100,100)



Testing pre-trained Deep Neural Network models (2)

- Synthetic images generated by combining a set of image transformations (t-way tests) can successfully identify inconsistent behavior.
- T-way synthetic images can significantly increase neuron coverage, a measure of the proportion of neurons activated in a DNN model.

T-way tests successfully uncover behavior inconsistencies among pre-trained models

Explaining Model Decisions

Explaining a model's decision is similar to the fault localization

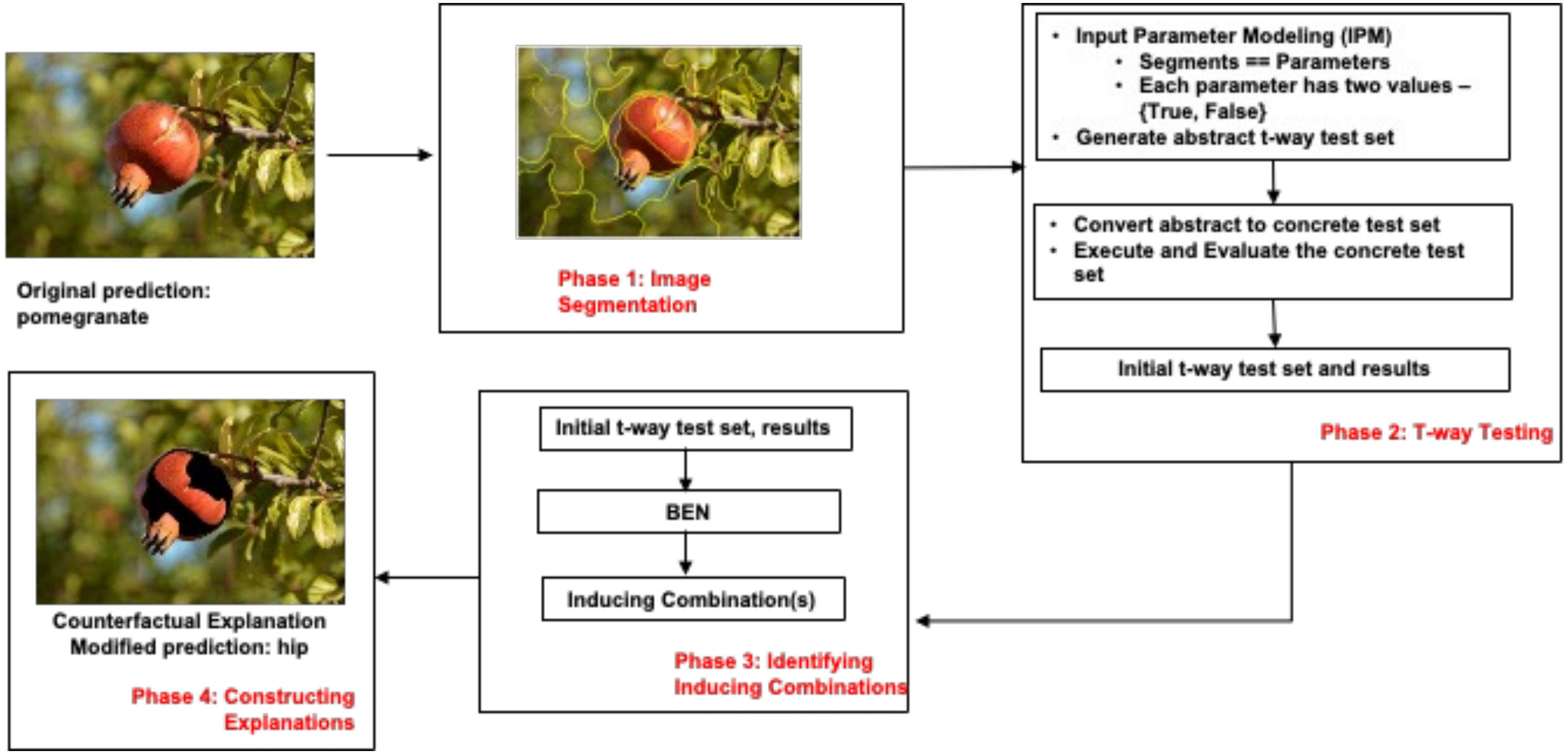
- Fault localization: Given a failing scenario, a software engineer *identifies* *which part of the input* that causes the failure
- Explainable AI: Given a decision (made by an ML model), the objective is to identify features that influence the model's decision.

Can we use BEN, a combinatorial testing-based software fault localization tool, to generate an explanation?

Explaining Model Decisions (2)

Explaining the behavior of image classifiers through counterfactual explanations

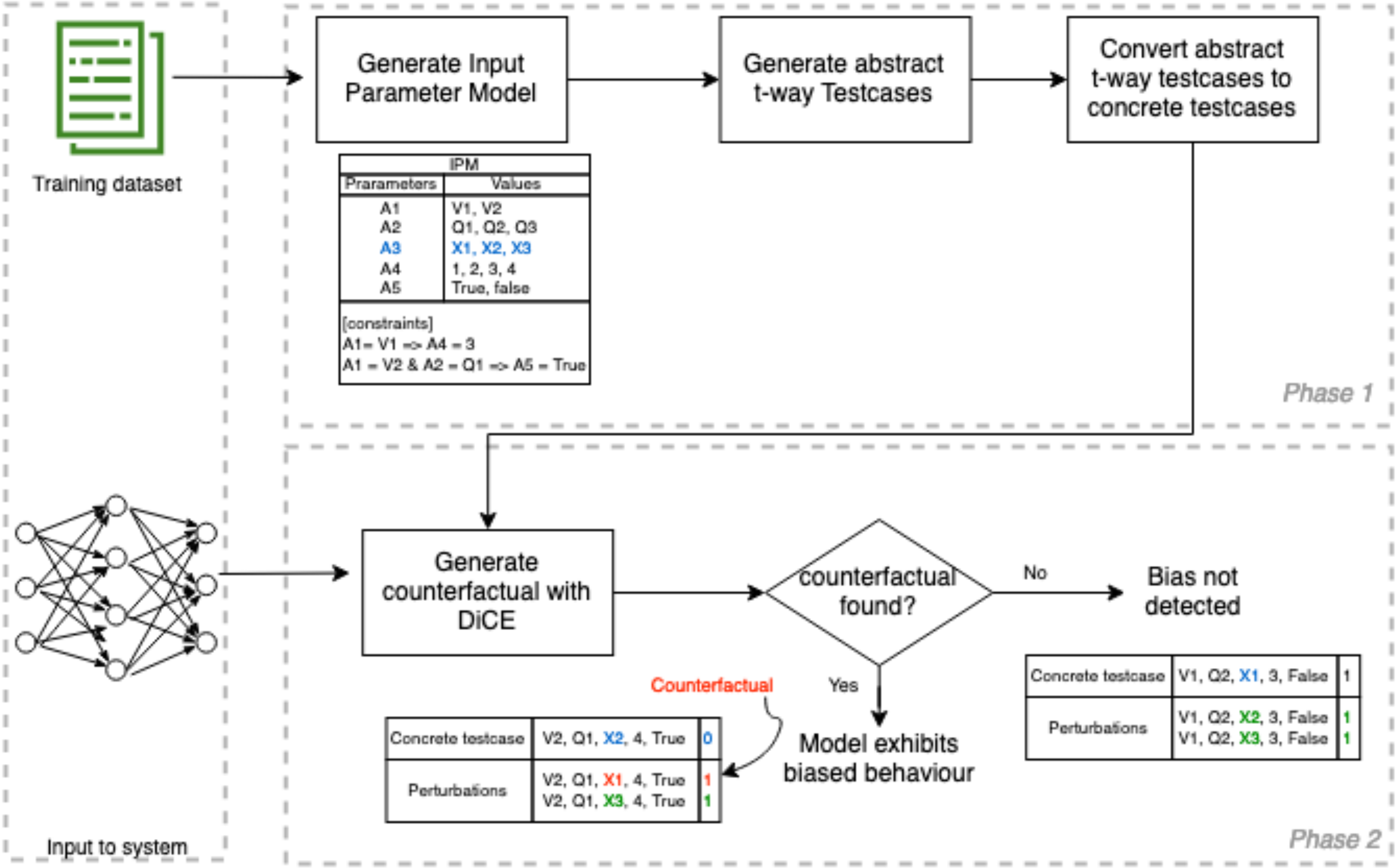
Determine a minimal subset of features in the input image that, if removed, would result in a different classification.



The CT-based approach effectively generates a counterfactual explanation for 44 out of 50 (~90%) images.

Fairness Testing of ML models

- Evaluating the discriminatory behavior of pre-trained ML models using combinatorial testing
- A model-agnostic approach that can successfully identify fairness violations in ML models



Fairness Testing of ML models (2)

Results suggest that t-way tests effectively identify biases introduced by the training dataset and the learning algorithm.

Datasets	# of t-way tests	Protected Attributesx	Number of fairness violations			
			Logistic Regression (LR)	Random Forest (RF)	Support Vector Machines (SVM)	Deep Neural Network (DNN)
Adult Income	676	Sex, Race	**58**	25	23	11
German Credit	81	Sex, Age	5	10	9	**26**
COMPAS	64	Sex, Race	27	**37**	24	4

Fairness violations identified by t-way tests

Dataset	**Attribute Modified**	**# CF found**			
		LR	**RF**	**SVM**	**DNN**
Adult Income	Only Race	16	10	5	4
	Only Sex	11	1	3	5
	Both Race and Sex	31	14	16	2
	Total	**58**	**25**	**23**	**11**

Dataset	**Attribute Modified**	**# CF found**			
		LR	**RF**	**SVM**	**DNN**
German Credit	Only Race	0	0	0	4
	Only Sex	0	0	0	10
	Both Race and Sex	5	10	9	12
	Total	**5**	**10**	**9**	**26**

Dataset	**Attribute Modified**	**# CF found**			
		LR	**RF**	**SVM**	**DNN**
COMPAS	Only Race	11	19	5	2
	Only Sex	1	3	1	1
	Both Race and Sex	15	15	18	1
	Total	**27**	**37**	**24**	**4**

Outline

- Introduction
- Challenges in Testing AI/ML Systems
- Combinatorial Testing for AI/ML Systems
- Overview of Combinatorial Testing in Evaluating AI/ML Systems
- Challenges and Future Directions

Challenges in Adapting CT for ML

Input Parameter Modeling
- Parameters
- Values
- Constraints

Step 1

→

Generate Abstract Tests
- Size of "t"
- Generate t-way abstract tests

Step 2

→

Generate Concrete Tests
- Convert abstract to concrete tests
- Test oracle

Step 3

→

Test execution
- Metrics

Parameters = Features
Values
- Categorical features
- Numerical features (discretized)

A higher value of ”t” can result in a significantly larger number of tests

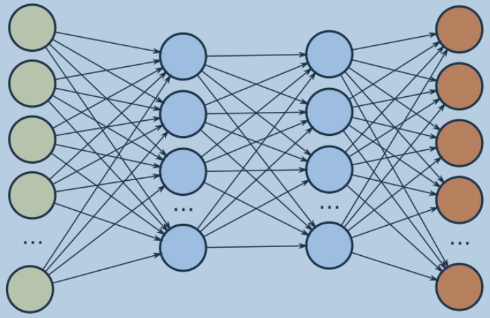
The choice of metrics impacts our understanding of the model’s capabilities.

Future Directions

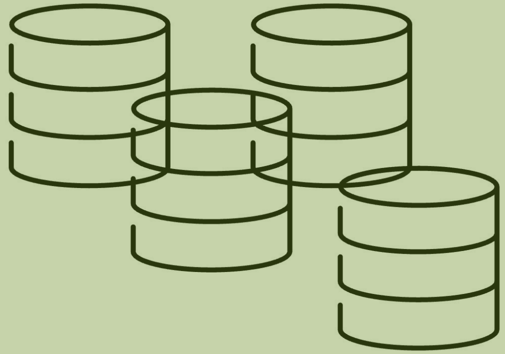
- Model debugging
- Post deployment
  - Monitoring
  - Regression testing
- Other data modalities
  - Natural Language Processing Datasets

Next Session...

The performance of ML systems is directly influenced by the quality of the underlying ML model and the data used to train it.



ML Model



Data

This session

Next session

Thank you!!!!

Collaborators*

Tyler Cody

Laura Freeman

Erin Lanus

Jeff Lei

Brian Lee

Raghu Kacker

Krishna Khadka

Rick Kuhn

M S Raunak

Ankita Patel

& all the anonymous reviewers from the community

* Listed in alphabetical order

References

1. Chandrasekaran, J., Lei, Y., Kacker, R., & Kuhn, D. R. (2021, April). A combinatorial approach to testing deep neural network-based autonomous driving systems. In 2021 IEEE international conference on software testing, verification and validation workshops (ICSTW) (pp. 57-66). IEEE.
2. Kuhn, R., Lei, Y., & Kacker, R. Practical Combinatorial Testing: Beyond Pairwise. Kuhn, R., Lei, Y., & Kacker, R. Practical Combinatorial Testing: Beyond Pairwise.
3. Ma, L., Juefei-Xu, F., Xue, M., Li, B., Li, L., Liu, Y., & Zhao, J. (2019, February). Deepct: Tomographic combinatorial testing for deep learning systems. In 2019 IEEE 26th International Conference on Software Analysis, Evolution and Reengineering (SANER) (pp. 614-618). IEEE.
4. Gladisch, C., Heinzemann, C., Herrmann, M., & Woehrle, M. (2020). Leveraging combinatorial testing for safety-critical computer vision datasets. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (pp. 324-325).
5. Dola, S., McDaniel, R., Dwyer, M. B., & Soffa, M. L. (2024, April). CIT4DNN: Generating Diverse and Rare Inputs for Neural Networks Using Latent Space Combinatorial Testing. In Proceedings of the IEEE/ACM 46th International Conference on Software Engineering (pp. 1-13)
6. Kuhn, D. R., Kacker, R. N., Lei, Y., & Simos, D. E. (2020, October). Combinatorial methods for explainable AI. In 2020 IEEE international conference on software testing, verification and validation workshops (ICSTW) (pp. 167-170). IEEE.
7. Chandrasekaran, J., Lei, Y., Kacker, R., & Kuhn, D. R. (2021, April). A combinatorial approach to explaining image classifiers. In 2021 IEEE International Conference on Software Testing, Verification and Validation Workshops (ICSTW) (pp. 35-43). IEEE.
8. Patel, A. R., Chandrasekaran, J., Lei, Y., Kacker, R. N., & Kuhn, D. R. (2022, April). A combinatorial approach to fairness testing of machine learning models. In *2022 IEEE international conference on software testing, verification and validation workshops (ICSTW)* (pp. 94-101). IEEE.
9. Khadka, K., Chandrasekaran, J., Lei, Y., Kacker, R. N., & Kuhn, D. R. (2024, April). A Combinatorial Approach to Hyperparameter Optimization. In Proceedings of the IEEE/ACM 3rd International Conference on AI Engineering-Software Engineering for AI (pp. 140-149).
10. Khadka, K., Chandrasekaran, J., Lei, Y., Kacker, R. N., & Kuhn, D. R. (2023, April). Synthetic data generation using combinatorial testing and variational autoencoder. In 2023 IEEE International Conference on Software Testing, Verification and Validation Workshops (ICSTW) (pp. 228-236). IEEE.
11. Sekhon, J., & Fleming, C. (2019, May). Towards improved testing for deep learning. In 2019 IEEE/ACM 41st International Conference on Software Engineering: New Ideas and Emerging Results (ICSE-NIER) (pp. 85-88). IEEE.
12. Becker,Barry and Kohavi,Ronny. (1996). Adult. UCI Machine Learning Repository. https://doi.org/10.24432/C5XW20.
13. UCI Machine Learning Repository: Adult Data Set | https://archive.ics.uci.edu/dataset/2/adult
14. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., ... & Fei-Fei, L. (2015). Imagenet large scale visual recognition challenge. *International journal of computer vision*, *115*, 211-252.
15. Cody, T., Lanus, E., Doyle, D. D., & Freeman, L. (2022, April). Systematic training and testing for machine learning using combinatorial interaction testing. In *2022 IEEE International Conference on Software Testing, Verification and Validation Workshops (ICSTW)* (pp. 102-109). IEEE.
16. Chandrasekaran, J., Feng, H., Lei, Y., Kuhn, D. R., & Kacker, R. (2017, March). Applying combinatorial testing to data mining algorithms. In 2017 IEEE International Conference on Software Testing, Verification and Validation Workshops (ICSTW) (pp. 253-261). IEEE