

A QUALITY ENGINEERING INTRODUCTION TO AI+ML

1

ABOUT ME



TARIQ KING



ABOUT: YOU

Agile + DevOps

PARTICIPANTS

3 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS

3

COURSE GOALS

What you'll learn and how you can apply it

- Key challenges associated with developing, testing, and debugging AI/ML systems
- How autonomous and intelligent agents are modeled, trained, and evaluated
- Performance measures and methods for evaluating ML models prior to release
- Different types of AI/ML bias and how they impact software development risks
- Techniques for testing adaptive AI/ML systems online in production environments
- ML operations (MLOps) for continuous training, testing, integration, and deployment

4 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS

4

COURSE GOALS

And you'll be able to:

- Validate and evaluate the performance of ML models using popular Python libraries
- Use open source AI fairness toolkits to examine, report, and mitigate bias in ML models
- Develop and execute an end-to-end test plan for an ML-based system or component
- Create an automated ML pipeline for continuous model training, validation, and testing



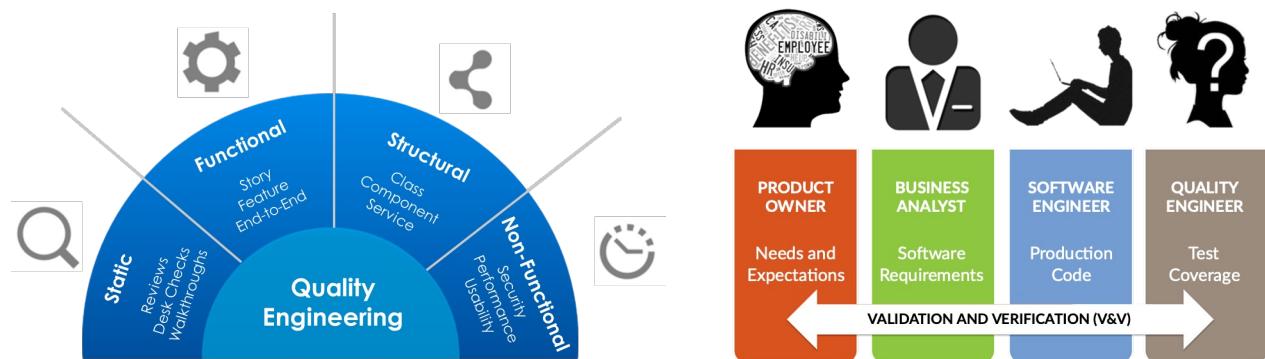
5 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS

5

QUALITY ENGINEERING

6

QUALITY ENGINEERING (QE)



Focuses on building quality into the product and engineering process.

Validation and verification at every stage of the software development lifecycle.

7 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS

7

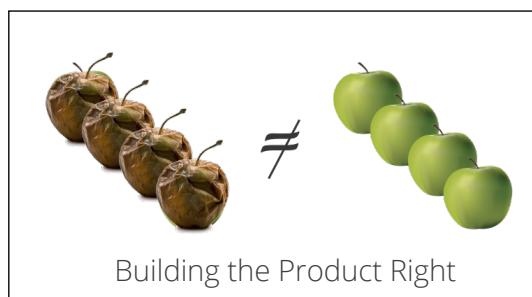
QUALITY ENGINEERING (QE)

Starts with...



Validation

Continues with...



Verification

8 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS

8

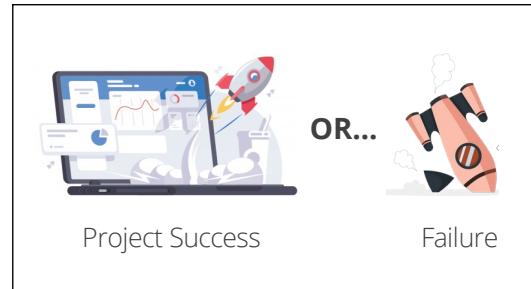
QUALITY ENGINEERING (QE)

Ends with...



Customer Impact

Which translates to...

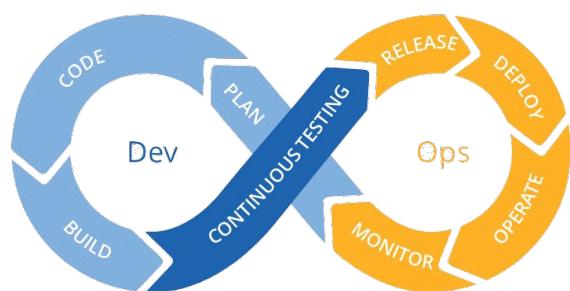


Business Impact

9 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS

9

QUALITY ENGINEERING (QE)



Not just development.
But also, operations.

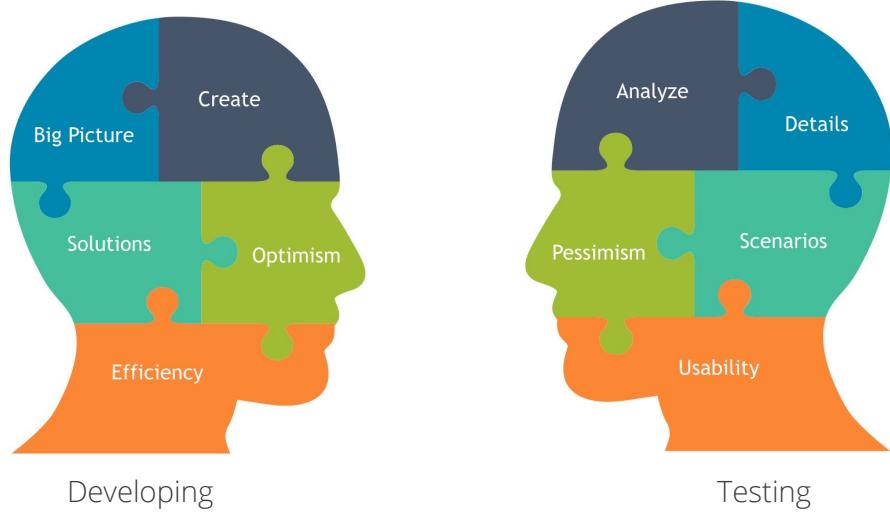


Less about quality practices and processes.
More about fostering a culture of quality.

10 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS

10

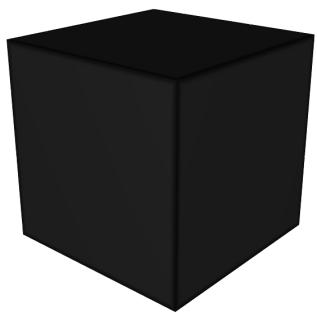
THE VALUE OF PERSPECTIVE



11 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS

11

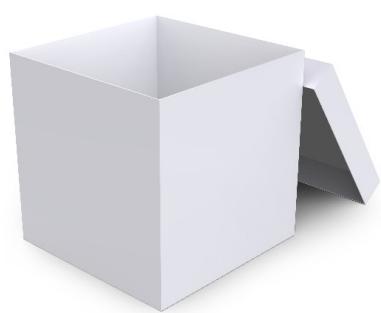
SOFTWARE TESTING APPROACHES



Black Box



Gray Box



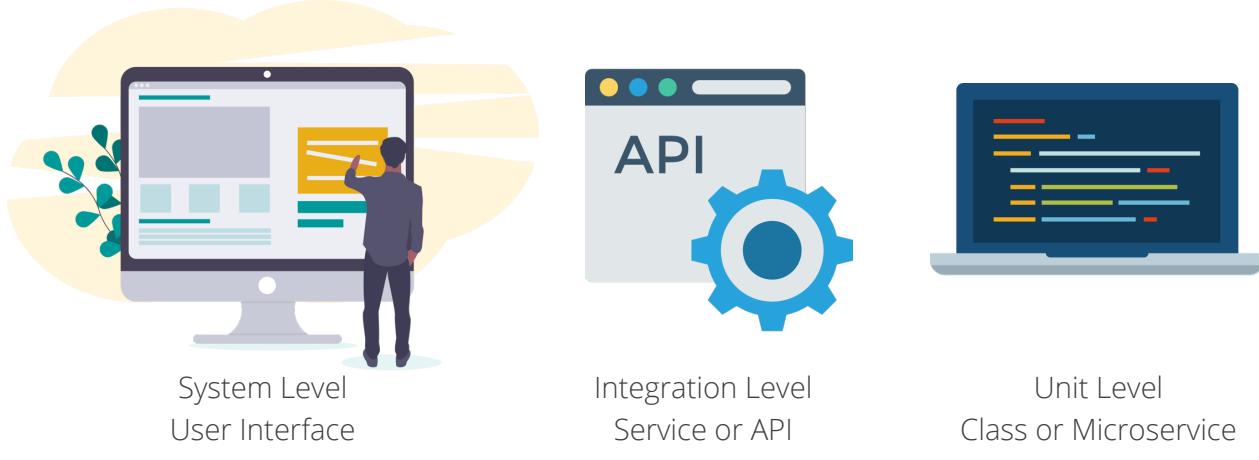
White Box

12 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS

12

SOFTWARE TESTING

LEVELS

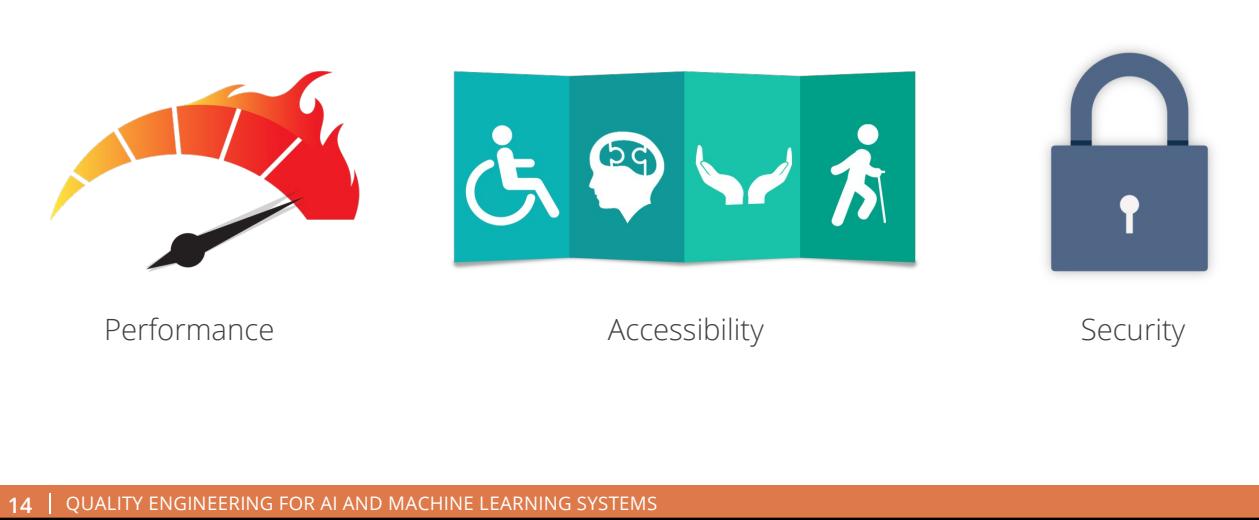


13 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS

13

SOFTWARE TESTING

QUALITY ATTRIBUTES

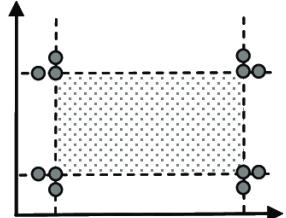


14 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS

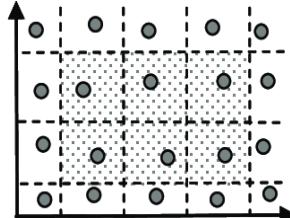
14

SOFTWARE TESTING

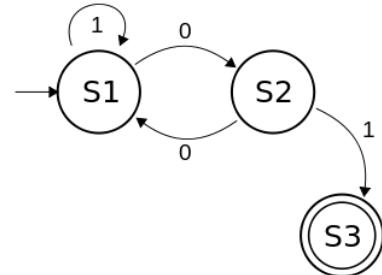
TECHNIQUES



Error-Based
Off-By-One Errors
Extreme Conditions



Coverage-Based
Equivalence Class Testing
Pairwise Testing



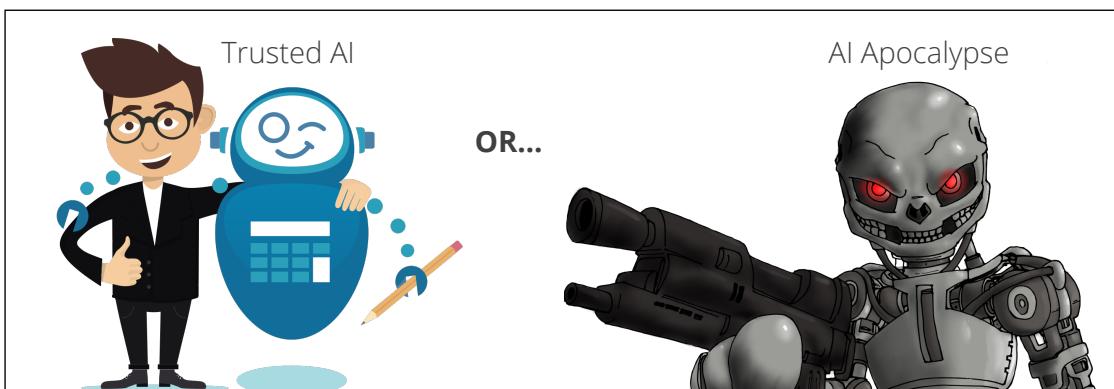
Model-Based
State Machine Testing
Decision Table Testing

15 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS

15

QE AND TESTING IMPLICATIONS FOR AI

Just might be the difference between...



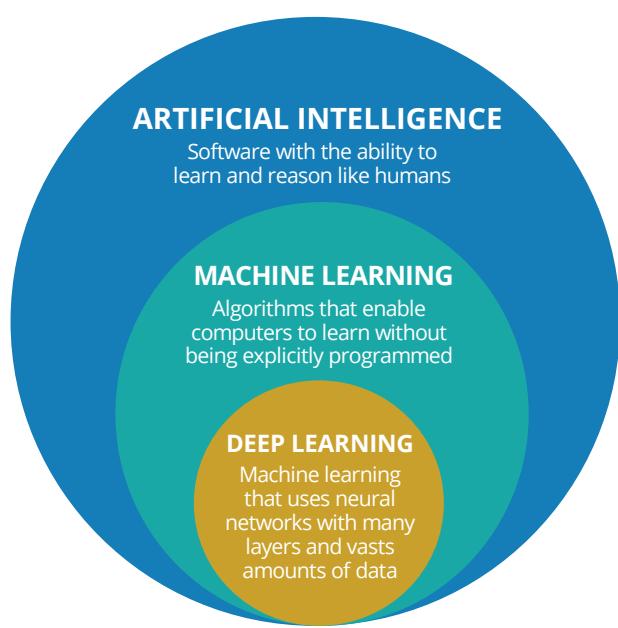
16 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS

16

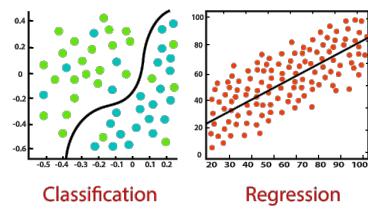
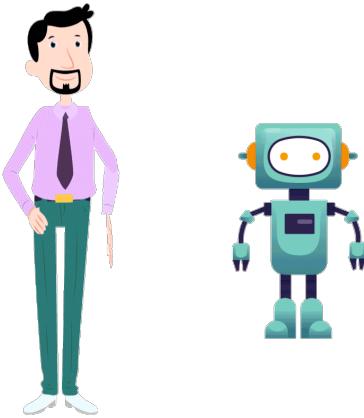
AI & MACHINE LEARNING

17

AI & ML



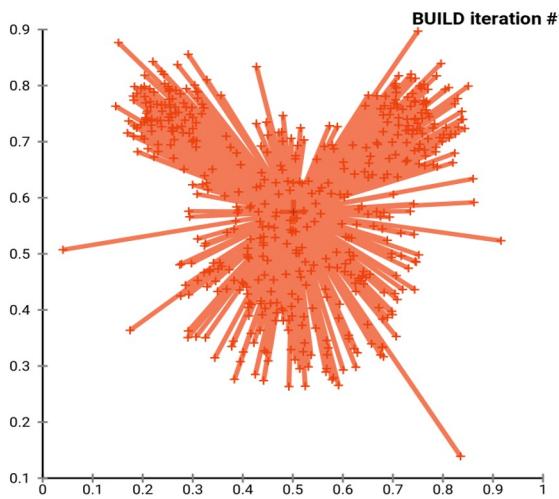
SUPERVISED LEARNING



19 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS: INTRODUCTION TO AI/ML

19

UNSUPERVISED LEARNING



20 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS: INTRODUCTION TO AI/ML

20

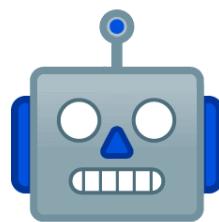
REINFORCEMENT LEARNING



21 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS: INTRODUCTION TO AI/ML

21

INTELLIGENT AGENTS: BOTS



Agent



Environment

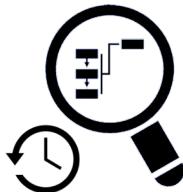
22 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS: INTRODUCTION TO AI/ML

22

INTELLIGENT AGENTS: TYPES OF BOTS



SIMPLE REFLEX AGENTS



MODEL-BASED AGENTS



GOAL-BASED AGENTS



UTILITY-BASED AGENTS

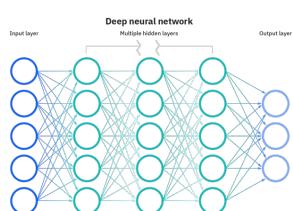


LEARNING AGENTS

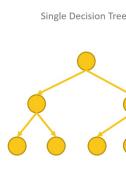
23 | FOUNDATIONS OF AI+ML

23

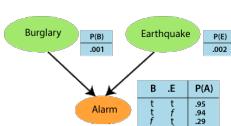
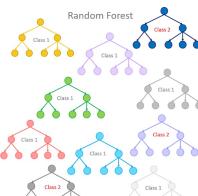
INTELLIGENT AGENTS: ML MODELS



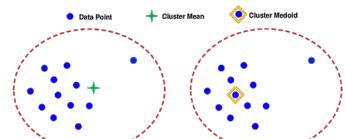
NEURAL NETWORKS



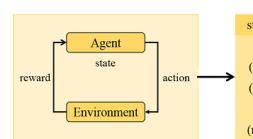
DECISION TREES AND RANDOM FORESTS



BAYESIAN NETWORKS



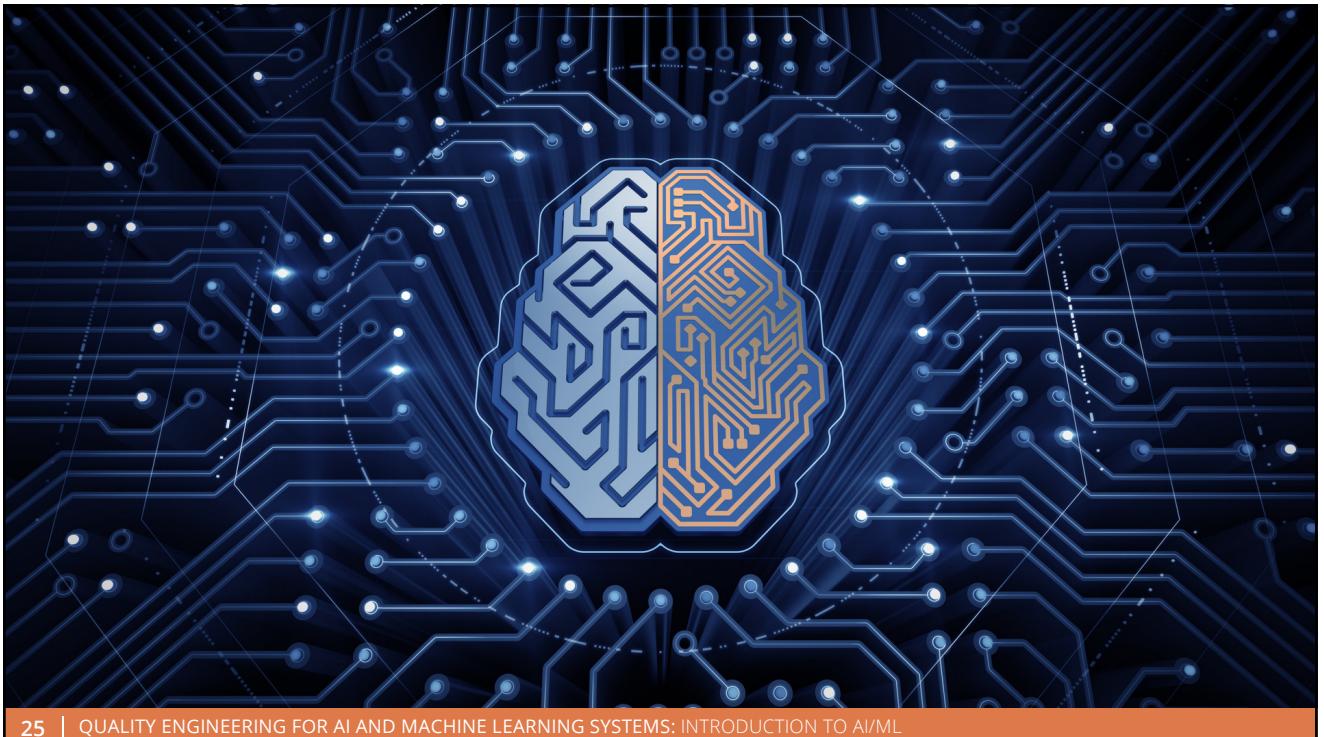
K-MEANS AND K-MEDOIDS CLUSTERING



Q-LEARNING

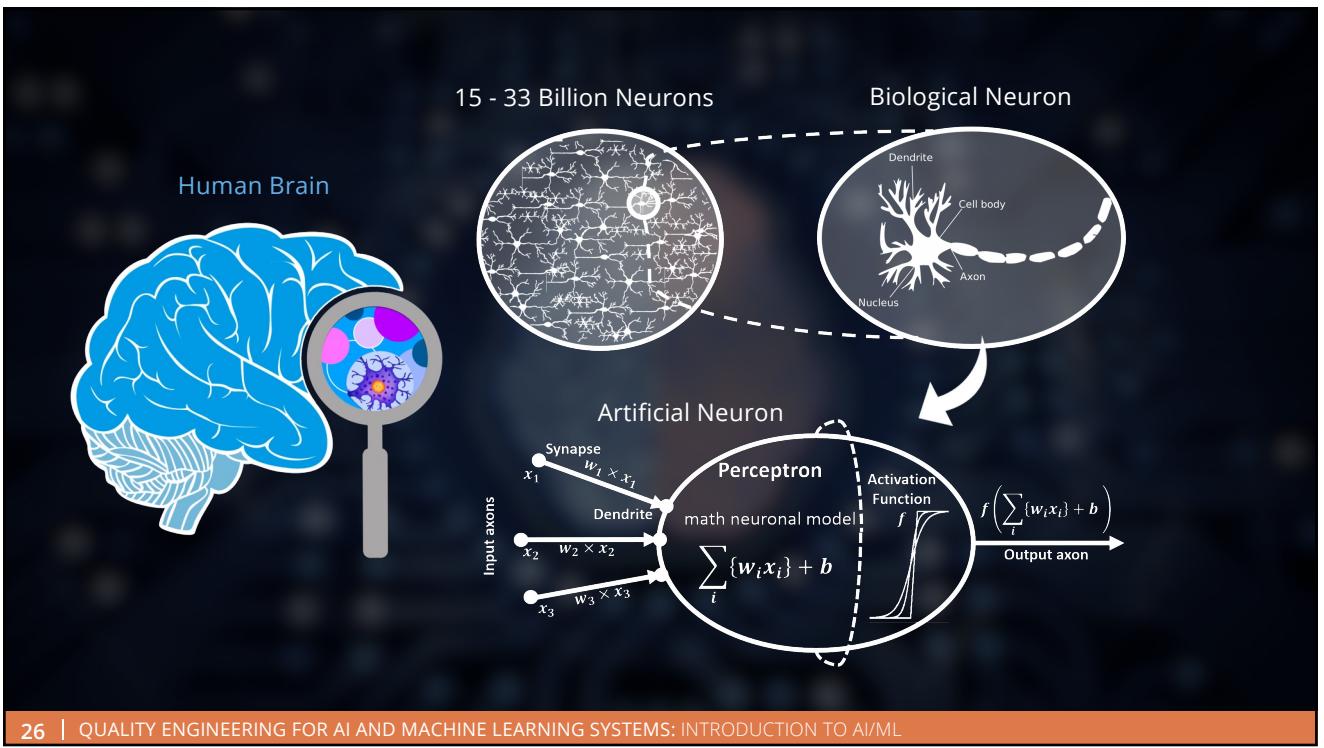
24 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS: INTRODUCTION TO AI/ML

24



25 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS: INTRODUCTION TO AI/ML

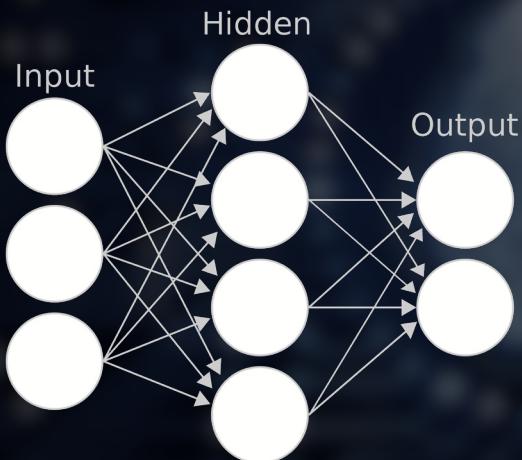
25



26 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS: INTRODUCTION TO AI/ML

26

ARTIFICIAL NEURAL NETWORKS



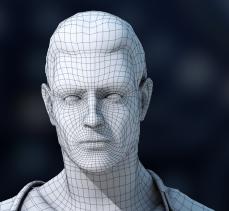
"A feed-forward network with a single layer is sufficient to represent any function..."

– Ian Goodfellow

27 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS: INTRODUCTION TO AI/ML

27

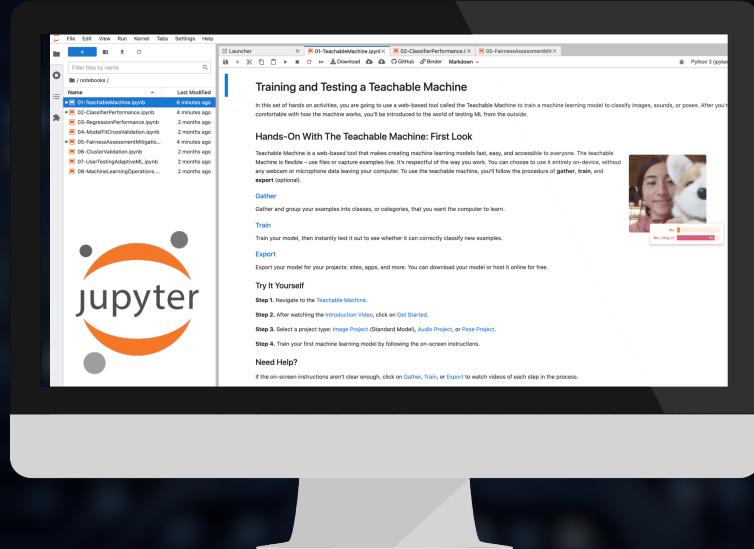
ARTIFICIAL NEURAL NETWORKS



28 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS: INTRODUCTION TO AI/ML

28

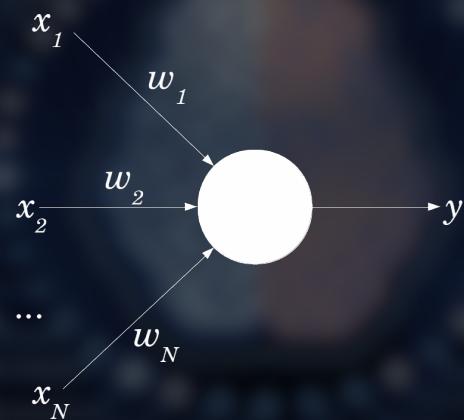
HANDS-ON: THE TEACHABLE MACHINE



29 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS: INTRODUCTION TO AI/ML

29

ARTIFICIAL NEURAL NETWORKS: BUT HOW?



Each neuron is a function that maps inputs to outputs.

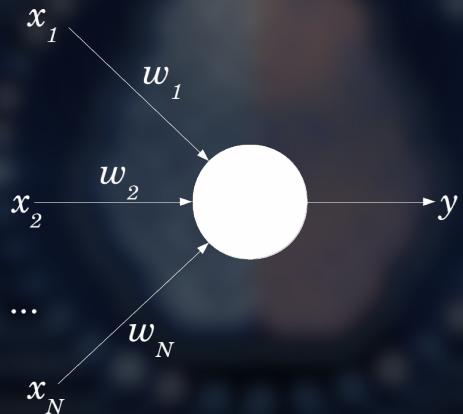
30 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS: INTRODUCTION TO AI/ML

30

HOW NEURAL NETWORKS LEARN

Given examples,
find weights that
map inputs to
outputs.

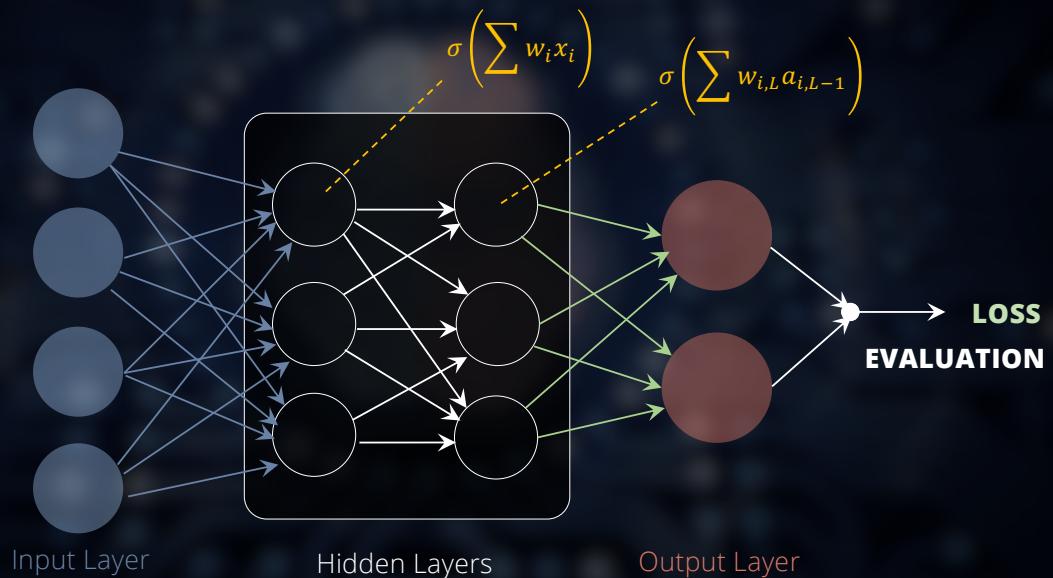
Initially, weights are
randomly initialized.



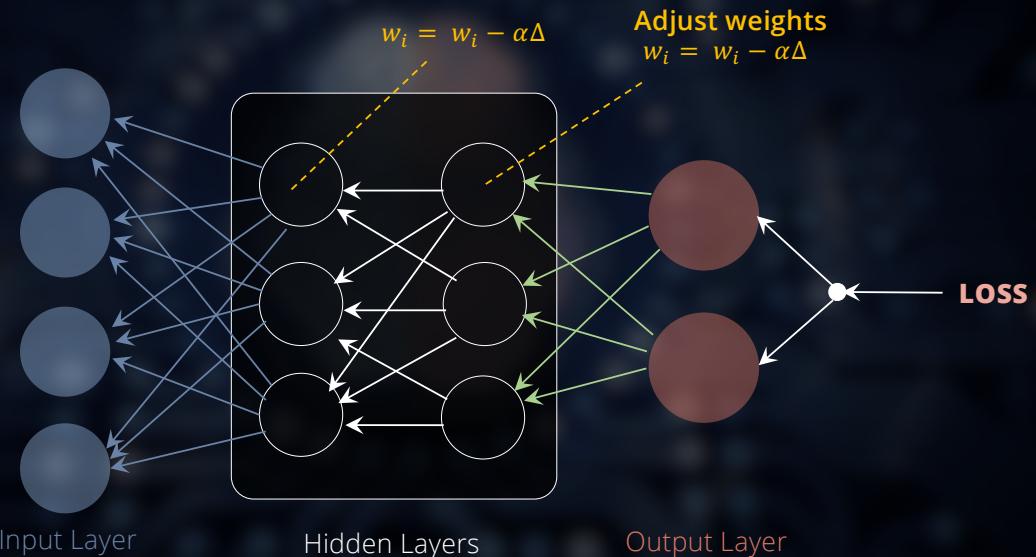
Value of **predicted output**
(\hat{y}) may be different from the
actual output (y).

A **LOSS** function is used to
measure this difference.

HOW NEURAL NETWORKS LEARN



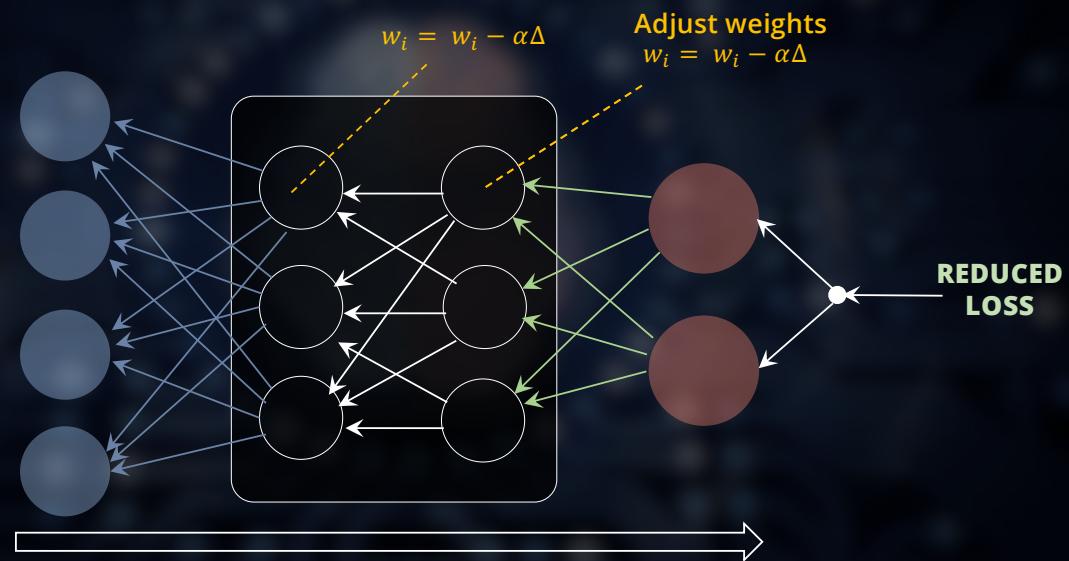
HOW NEURAL NETWORKS LEARN



33 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS: INTRODUCTION TO AI/ML

33

HOW NEURAL NETWORKS LEARN



34 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS: INTRODUCTION TO AI/ML

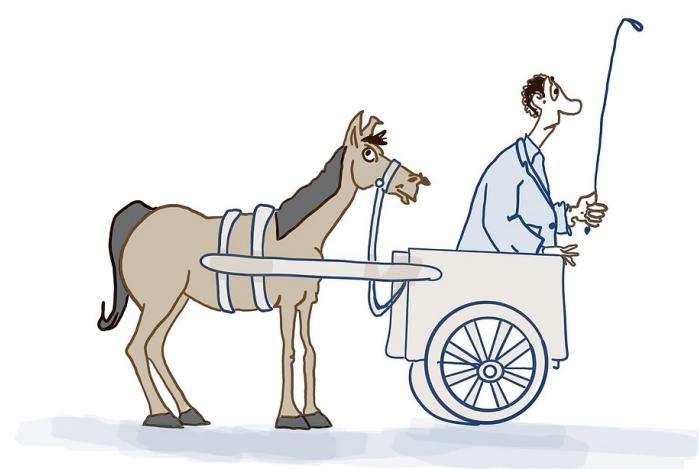
34

QE PRACTICES FOR AI/ML

35

QE FOR AI/ML: FOUNDATIONAL PRACTICES

AI PROBLEM FIRST, THEN SOLUTION



Define the problem

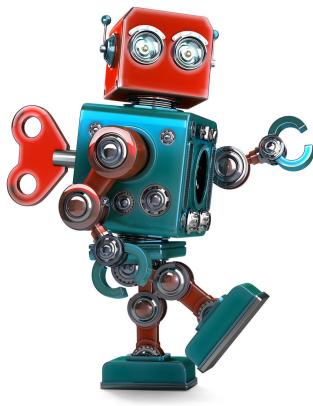
- What do want to accomplish?
- What are the desired outcomes?
- Do you have the data to infer those outcomes?

Consider the solutions

- Can the problem be solved by AI?
- Should it be solved by AI?
- Is there a simpler solution that does not require AI?

QE FOR AI/ML: FOUNDATIONAL PRACTICES

CHOOSE ML ALGORITHMS BASED ON NEEDS, NOT POPULARITY

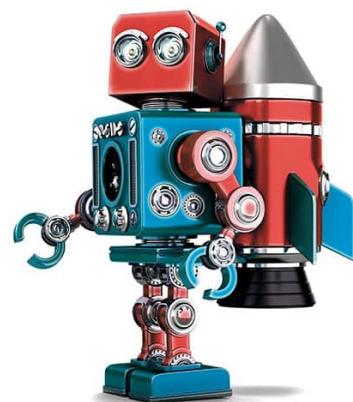


Algorithms differ in...

- Kinds of problems they can solve
- Level of detail of the output
- Interpretability of the model/output
- Robustness to adversaries

Do not use as a shiny new toy

- Change algorithms to meet needs as the system evolves or the environment changes.



QE FOR AI/ML: FOUNDATIONAL PRACTICES

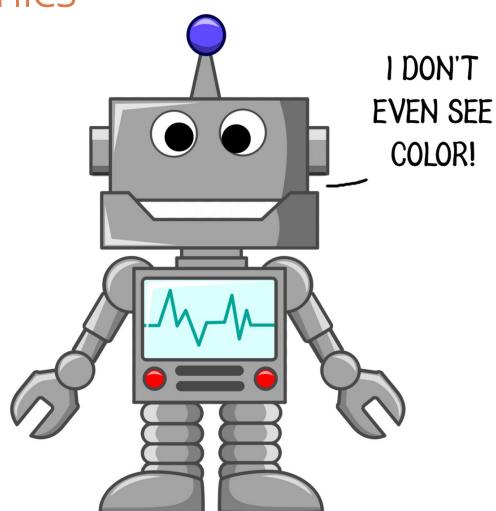
DESIGN, TEST, AND MONITOR AI FOR ETHICS

Account for organizational and societal values in all aspects of system...

- Ethics in data collection (privacy).
- Ethics in data representation (fairness)
- Ethics in decision making (fairness)
- Training Process and model structure

Validate dataset and monitor system...

- Pre- and post-release
- Continuously where possible.



QE FOR AI/ML: FOUNDATIONAL PRACTICES

RULE YOUR DATA OR BE RULED BY IT



Data can impact quality directly...

- Output of an AI system is often tied to the data used to train it.

Data requires work...

- Ingestion, cleansing, protection
- Validation, testing, monitoring
- Overall management of data requires automation to scale.

39 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS: FOUNDATIONAL PRACTICES

39

QE FOR AI/ML: FOUNDATIONAL PRACTICES

IMPLEMENT FLEXIBLE, EXTENSIBLE SOLUTIONS

Boundaries of AI components are more sensitive than traditional software due to entanglement of data...

- Direct and indirect data dependencies can trigger changes in....
 - Functionality
 - Expected output
 - Supporting infrastructure
- Requires system design to be...
 - Loosely coupled
 - Highly extensible

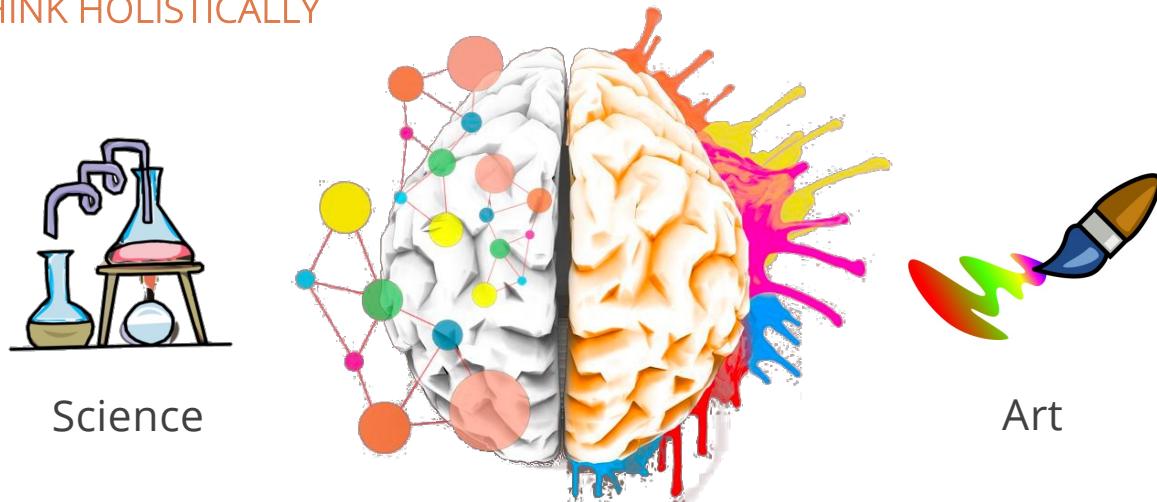


40 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS: FOUNDATIONAL PRACTICES

40

QE FOR AI/ML: FOUNDATIONAL PRACTICES

THINK HOLISTICALLY

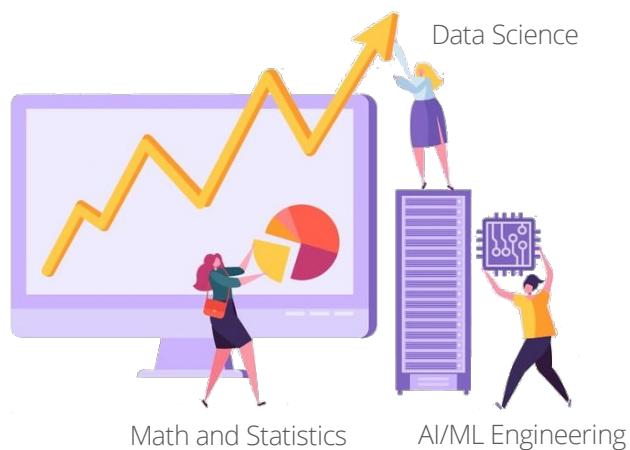


41 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS: FOUNDATIONAL PRACTICES

41

TOWARDS HOLISTIC QE FOR AI AND ML

CURRENT FOCUS: ACCURACY, PRECISION, RECALL



42 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS: FOUNDATIONAL PRACTICES

42

TOWARDS HOLISTIC QE FOR AI AND ML

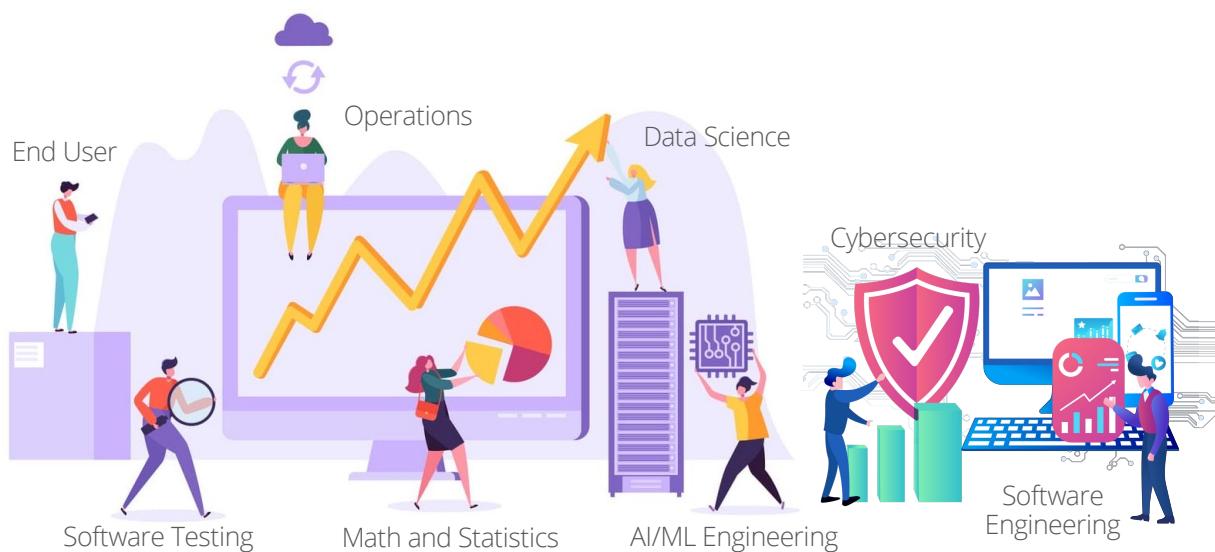
REMINDER: AI SYSTEMS ARE SOFTWARE-INTENSIVE



43 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS: FOUNDATIONAL PRACTICES

43

QE FOR AI AND ML: A HOLISTIC APPROACH



44 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS: FOUNDATIONAL PRACTICES

44

TESTING SUPERVISED ML

MODEL TRAINING & CLASSIFIER PERFORMANCE

45

Project

- yang-plugin-structure-view
- git
- lib
- main.js
- structure-view.js
- util.js

activate(state) {
 this.subscriptions = new CompositeDisposable();
 this.subscriptions.add(atom.commands.add('atom-workspace', {
 'toggle': () => this.switch()
 }));
}

TESTING SUPERVISED ML

DATA SCIENCE AND AI/ML ENGINEERING

Absolute Value
 $|A| = \sqrt{A^2}$

Gamma Function
 $\Gamma(x) = \int_0^\infty t^{x-1} e^{-t} dt$

Gamma Function Properties
 $\Gamma(1) = 1$
 $\Gamma(n+1) = n\Gamma(n)$
 $\Gamma(\frac{1}{2}) = \sqrt{\pi}$

Gamma Function Values
 $\Gamma(1) = 1$
 $\Gamma(2) = 1$
 $\Gamma(3) = 2$
 $\Gamma(4) = 6$
 $\Gamma(5) = 24$
 $\Gamma(6) = 120$
 $\Gamma(7) = 720$
 $\Gamma(8) = 5040$
 $\Gamma(9) = 40320$
 $\Gamma(10) = 362880$

Gamma Function Derivative
 $\Gamma'(x) = \int_0^\infty t^{x-1} \ln(t) e^{-t} dt$

Gamma Function Integral
 $\Gamma(x) = \int_0^\infty t^{x-1} e^{-t} dt$

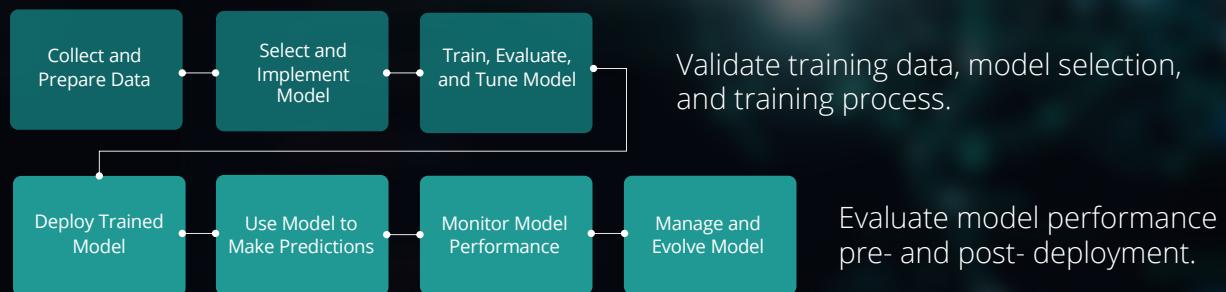
Gamma Function Plot

46 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS: TESTING SUPERVISED ML

46

ML ENGINEERING: FOUNDATION

7 STEPS OF MACHINE LEARNING



47 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS: TESTING SUPERVISED ML

47

MODEL TRAINING

CLASSIFICATION



	Outlook	Temp	Humidity	Windy	Play Tennis
1	Rainy	Hot	High	False	No
2	Rainy	Hot	High	True	No
3	Overcast	Hot	High	False	Yes
4	Sunny	Mild	High	False	Yes
5	Sunny	Cool	Normal	True	Yes
6	Sunny	Cool	Normal	True	Yes
7	Overcast	Cool	Normal	True	Yes
8	Rainy	Mild	High	False	No

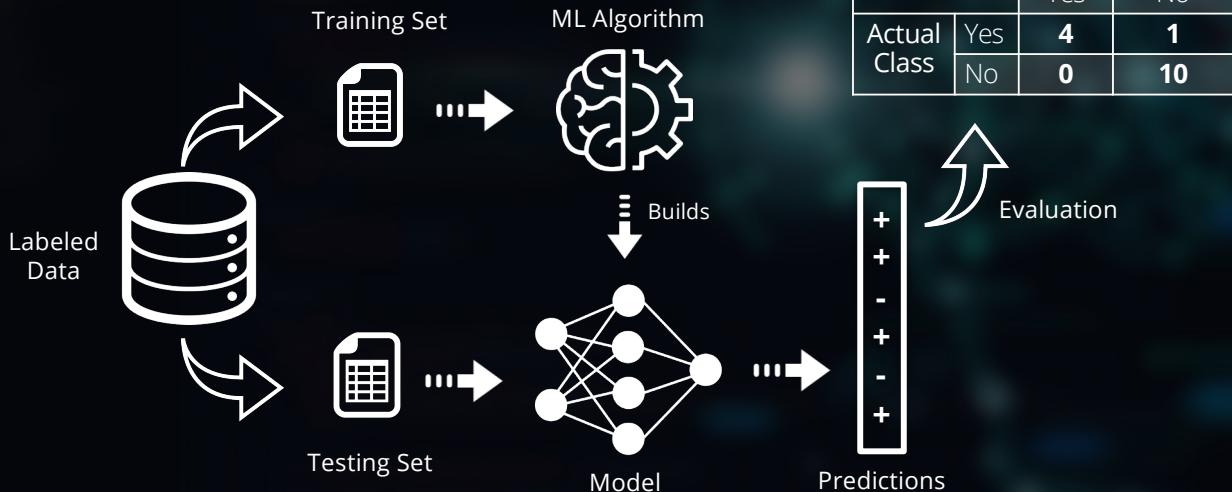
9	Rainy	Hot	High	False	?
10	Rainy	Hot	High	True	?
11	Overcast	Hot	High	False	?
12	Sunny	Mild	High	False	?

48 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS: TESTING SUPERVISED ML

48

MODEL TRAINING

CLASSIFICATION



49 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS: TESTING SUPERVISED ML

49

CLASSIFIER PERFORMANCE

ACCURACY AND ERROR

$$\begin{aligned}
 \text{Accuracy} &= \frac{\text{Number of Correct Predictions}}{\text{Total Number of Predictions}} \\
 &= 14 \div 15 = \sim 93.3\%
 \end{aligned}$$

$$\begin{aligned}
 \text{Error} &= \frac{\text{Number of Wrong Predictions}}{\text{Total Number of Predictions}} \\
 &= 1 \div 15 = \sim 6.7\%
 \end{aligned}$$

Confusion Matrix

		Predicted Class	
		Yes	No
Actual Class	Yes	4	1
	No	0	10



50 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS: TESTING SUPERVISED ML

50

CLASSIFIER PERFORMANCE

ACCURACY AND ERROR HAVE LIMITS

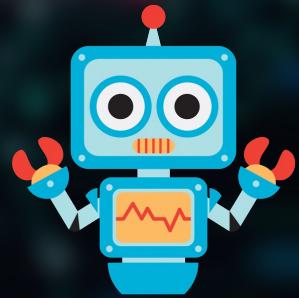


Class Imbalance



Majority Class

No
Minority Class



YesBot

51 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS: TESTING SUPERVISED ML

51

CLASSIFIER PERFORMANCE

TRUE VERSUS FALSE PREDICTIONS

True Predictions

- o **TP:** True Positives
 - Model Predicted Yes, and Actual Class is Yes
- o **TN:** True Negatives
 - Model Predicted No, and Actual Class is No

False Predictions

- o **FP:** False Positives
 - Model Predicted Yes, and Actual Class is No
- o **FN:** False Negatives
 - Model Predicted No, and Actual Class is Yes

Confusion Matrix (Reloaded)

		Predicted Class	
		Yes	No
Actual Class	Yes	TRUE POSITIVES	FALSE NEGATIVES
	No	FALSE POSITIVES	TRUE NEGATIVES



Evaluation

Predictions

52 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS: TESTING SUPERVISED ML

52

CLASSIFIER PERFORMANCE

PRECISION, RECALL, F-SCORE

$$\text{Precision} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}}$$

$$\text{Recall} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}}$$

$$\text{F-Measure} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$$

Confusion Matrix (Reloaded)

		Predicted Class	
		Yes	No
Actual Class	Yes	TRUE POSITIVES	FALSE NEGATIVES
	No	FALSE POSITIVES	TRUE NEGATIVES



53 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS: TESTING SUPERVISED ML

53

CLASSIFIER PERFORMANCE

PROBABILISTIC FORECASTING

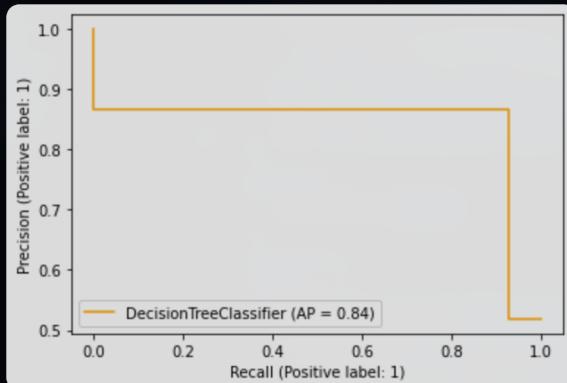


54 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS: TESTING SUPERVISED ML

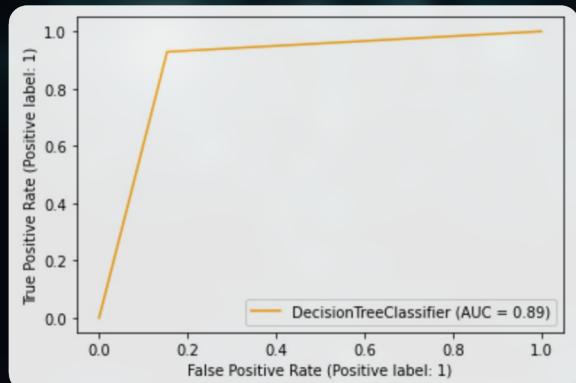
54

CLASSIFIER PERFORMANCE

PRECISION-RECALL CURVE, ROC CURVE, AND AUC-ROC SCORE



Precision-Recall Curve



Receiver Operating Characteristic (ROC) Curve

55 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS: TESTING SUPERVISED ML

55

HANDS-ON: CLASSIFICATION PERFORMANCE

FIRST LOOK, DEEP DIVE, AND EXERCISE

The screenshot shows a Jupyter Notebook cell with the following content:

```
Hands-On with Classification Metrics: First Look

Scikit-learn is a free ML library for the Python programming language. It has 3 different programming interfaces for evaluating the quality of a model's predictions:
• Estimator Score Method
• Scoring Parameter
• Metrics Functions

In this interactive demonstration, you'll get experience using the scikit-learn metrics functions to measure the prediction skill of a binary classifier that distinguishes cats and dogs.

First start by importing the scikit-learn metrics module:
```

```
[1]: from sklearn import metrics
```

```
Assume that the actual and predicted values from the example are defined as follows, where cats belong to the class 0 and dogs belong to the class 1.
```

```
[2]: actual_values = [0,0,0,0,0,0,0,1,1,1,1]
predictions = [1,1,0,0,0,0,0,1,1,1,0]
```

```
Now you can use the metrics functions to calculate the accuracy and print the confusion matrix.
```

```
[3]: print("Accuracy: (%.2f)" % (metrics.accuracy_score(actual_values, predictions) * 100))
print("Confusion Matrix:")
print(metrics.confusion_matrix(actual_values, predictions))
```

The cell ends with a Jupyter logo.

56 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS: TESTING SUPERVISED ML

56

TESTING SUPERVISED ML

REGRESSION PERFORMANCE & MODEL VALIDATION

57

REGRESSION PERFORMANCE

INTRODUCTION

- Predicts a Quantity
 - o Continuous values
 - o Amounts, sizes, prices, etc.

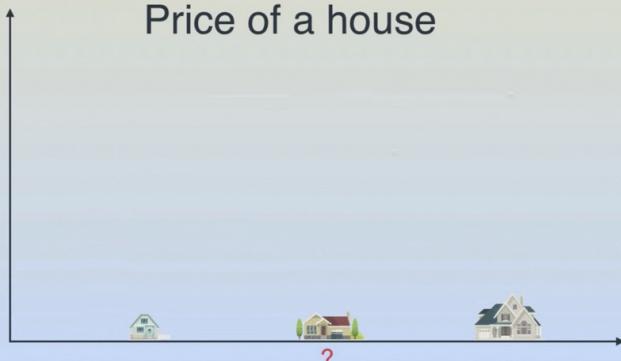
Example:

- o Selling Price of a House

Question:

- o How would you calculate the accuracy for this regression model?

Price of a house



HANDS ON: REGRESSION PERFORMANCE

ERROR (DISTANCE) METRICS

```
Hands-On with Regression Metrics

[ ]: #Import Required Libraries
from math import sqrt
import numpy as np
from sklearn import metrics

#Load Prediction Results
actual_values = [-5, -3.5, 6, 11]
predictions = [8.5, -2.9, 6, 9.2]

[ ]: # Calculate Mean Squared Error (MSE)
print(f'MSE: {metrics.mean_squared_error(actual_values, predictions)}')

# Calculate Root Mean Squared Error (RMSE)
def rmse(actual_values, predictions):
    actual_values = np.asarray(actual_values)
    predictions = np.asarray(predictions)
    return np.sqrt((predictions - actual_values) ** 2).mean()

print(f'RMSE: {rmse(actual_values, predictions)}')

# Calculate Mean Absolute Error
print(f'MAE: {metrics.mean_absolute_error(actual_values, predictions)}')

# Calculate R-Squared
print(f'R2: {metrics.r2_score(actual_values, predictions)}')
```

Mean Squared Error

Root Mean Squared Error

Mean Absolute Error

R-Squared

59 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS: TESTING SUPERVISED ML

59

MODEL VALIDATION: MOTIVATION

TWO SIMILAR QUALITY PHILOSOPHIES

Q = P

W. Edwards Deming

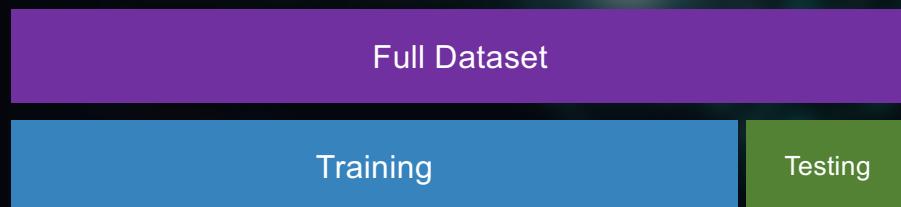
IN GOD
WE TRUST
ALL OTHERS
MUST BRING
DATA

60 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS: TESTING SUPERVISED ML

60

MODEL VALIDATION: INTRODUCTION

RECALL THE TRAIN-TEST SPLIT

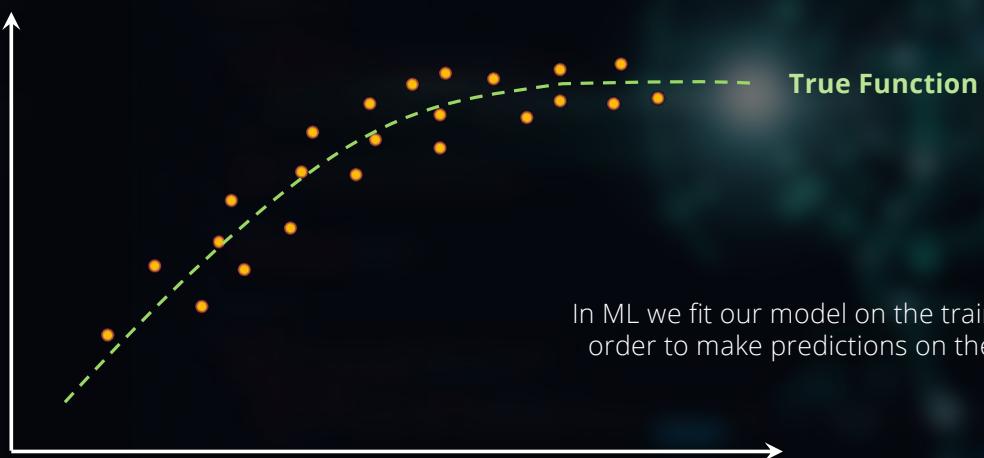


Typically this is an **80/20** or a **70/30** split.

61 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS: TESTING SUPERVISED ML

61

MODEL VALIDATION: FITTING THE MODEL

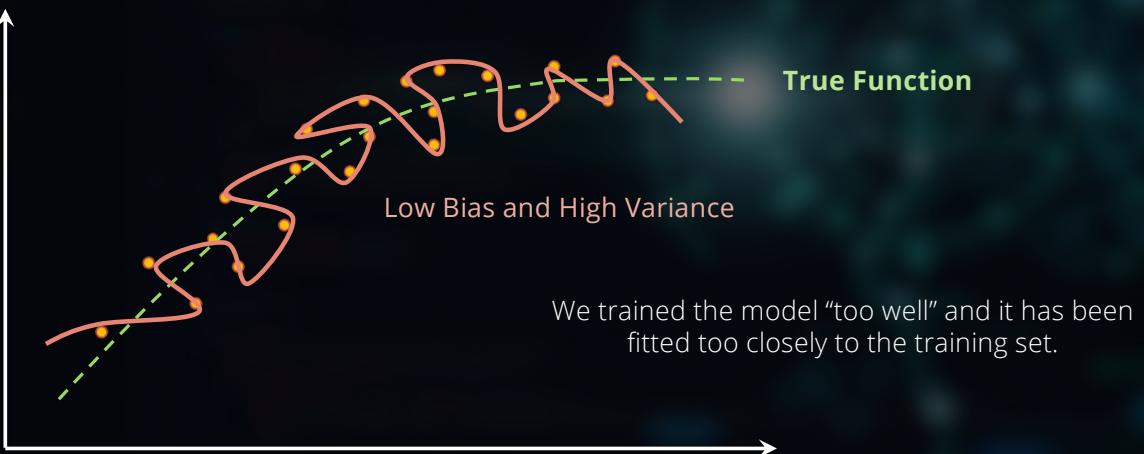


In ML we fit our model on the training data in order to make predictions on the test data

62 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS: TESTING SUPERVISED ML

62

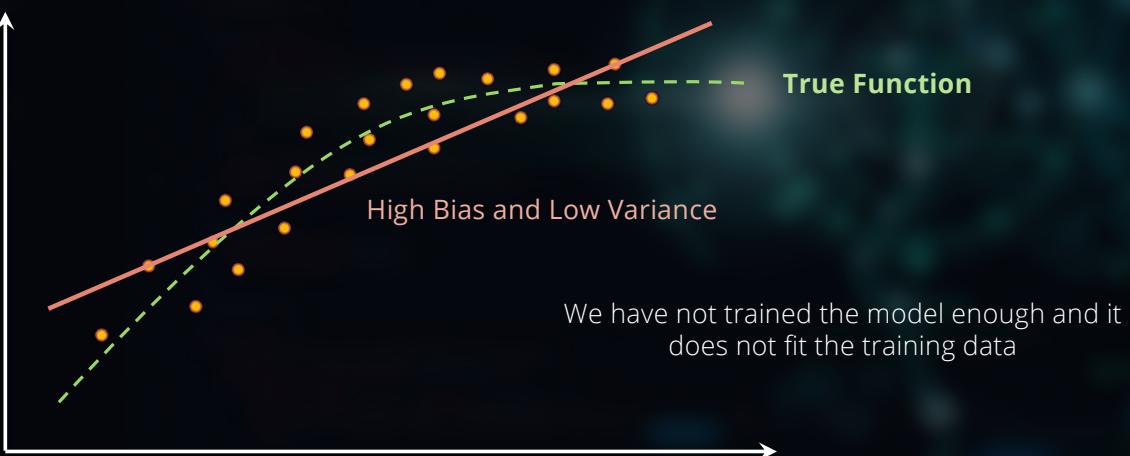
MODEL VALIDATION: OVERRFITTING



63 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS: TESTING SUPERVISED ML

63

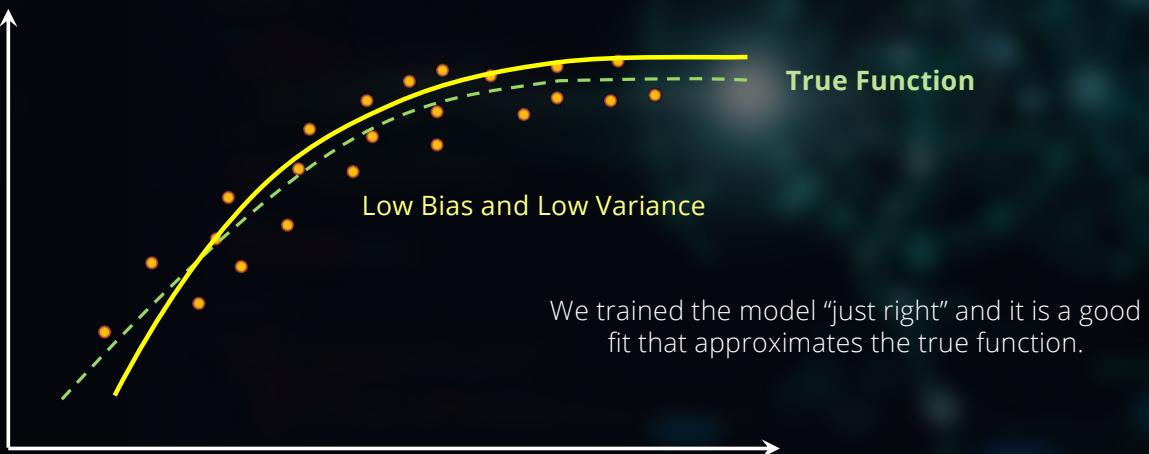
MODEL VALIDATION: UNDERFITTING



64 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS: TESTING SUPERVISED ML

64

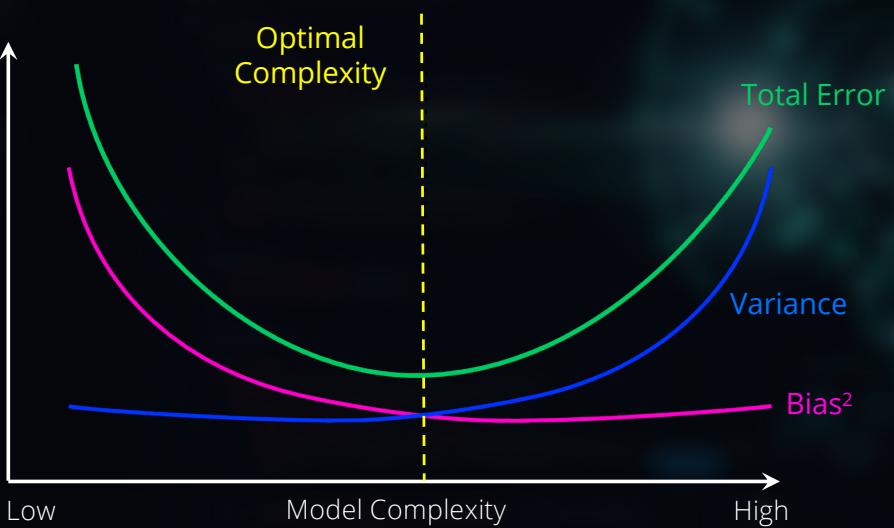
MODEL VALIDATION: BALANCED FITTING



65 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS: TESTING SUPERVISED ML

65

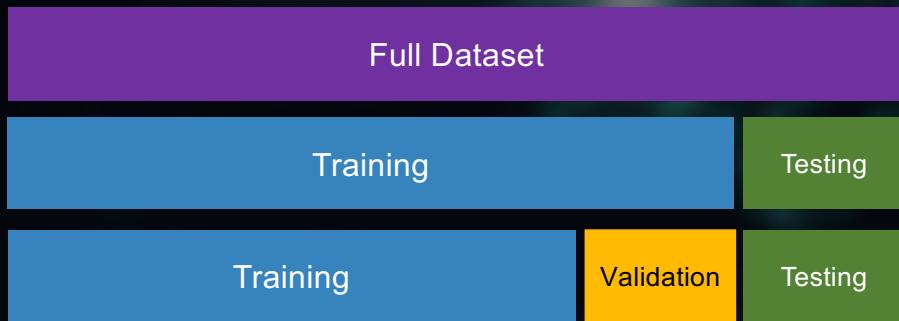
MODEL VALIDATION: BIAS-VARIANCE TRADE-OFF



66 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS: TESTING SUPERVISED ML

66

MODEL VALIDATION: REVISITING THE SPLIT TRAINING-VALIDATION-TESTING



67 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS: TESTING SUPERVISED ML

67

CROSS-VALIDATION TECHNIQUES

K-FOLD CROSS-VALIDATION

Full Dataset

68 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS: TESTING SUPERVISED ML

68

CROSS-VALIDATION TECHNIQUES

K-FOLD CROSS-VALIDATION

Shuffling the Dataset

69 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS: TESTING SUPERVISED ML

69

CROSS-VALIDATION TECHNIQUES

K-FOLD CROSS-VALIDATION (k=5)

Group 1

Group 2

Group 3

Group 4

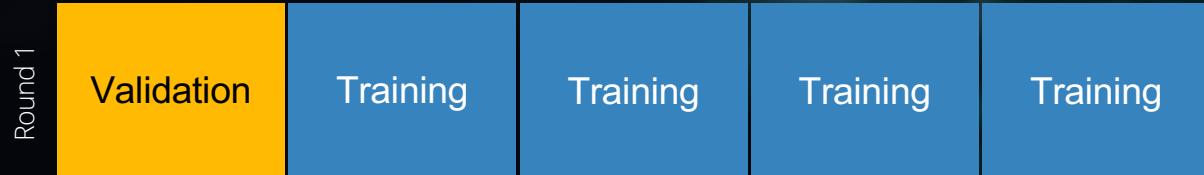
Group 5

70 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS: TESTING SUPERVISED ML

70

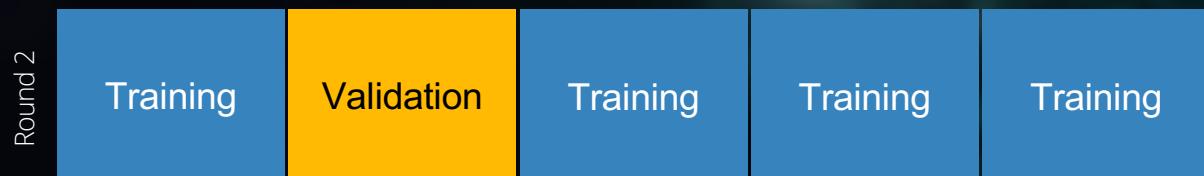
CROSS-VALIDATION TECHNIQUES

K-FOLD CROSS-VALIDATION (k=5)



CROSS-VALIDATION TECHNIQUES

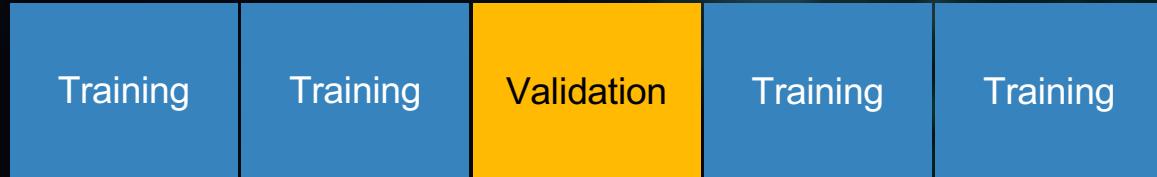
K-FOLD CROSS-VALIDATION (k=5)



CROSS-VALIDATION TECHNIQUES

K-FOLD CROSS-VALIDATION (k=5)

Round 3



73 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS: TESTING SUPERVISED ML

73

CROSS-VALIDATION TECHNIQUES

K-FOLD CROSS-VALIDATION (k=5)

Round 4

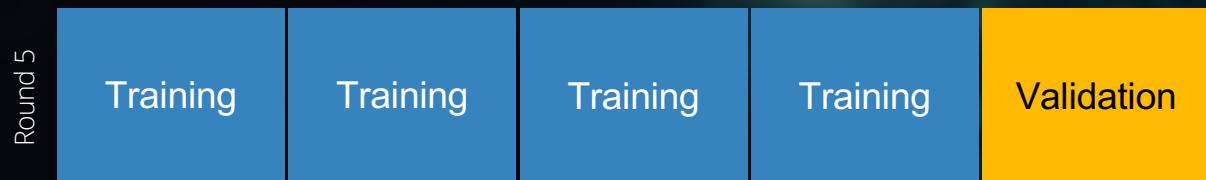


74 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS: TESTING SUPERVISED ML

74

CROSS-VALIDATION TECHNIQUES

K-FOLD CROSS-VALIDATION (k=5)

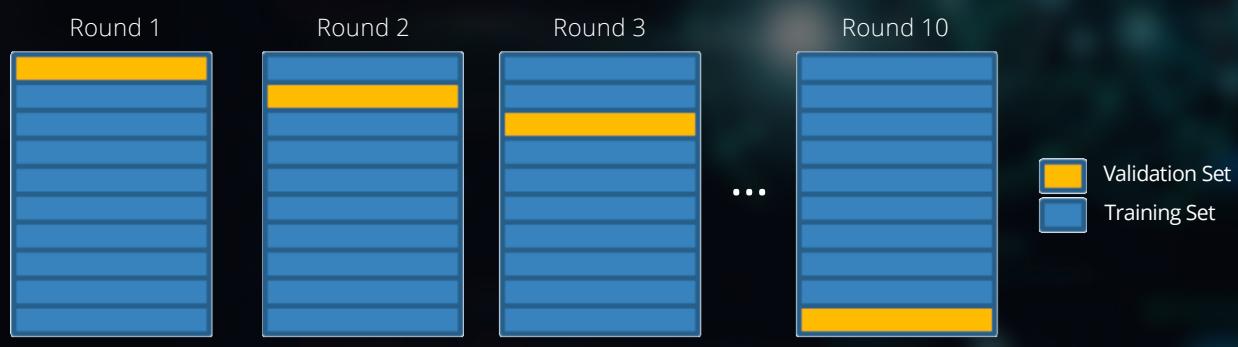


75 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS: TESTING SUPERVISED ML

75

CROSS-VALIDATION TECHNIQUES

K-FOLD CROSS-VALIDATION (k=10)



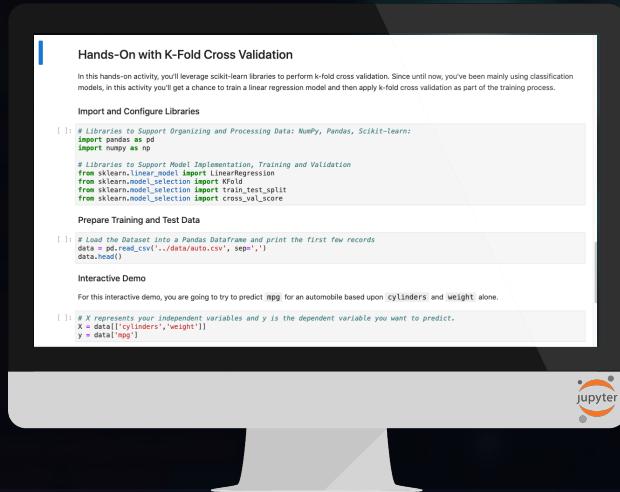
Final Accuracy = Average (Round 1, Round 2, Round 3, ..., Round 10)

76 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS: TESTING SUPERVISED ML

76

HANDS-ON: K-FOLD CROSS VALIDATION

TRY IT FOR YOURSELF



77 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS: TESTING SUPERVISED ML

77

MORE CROSS-VALIDATION TECHNIQUES

SELF-PACED LEARNING



Leave One Out

Leave P-Out

Stratified

Repeated

Nested

78 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS: TESTING SUPERVISED ML

78

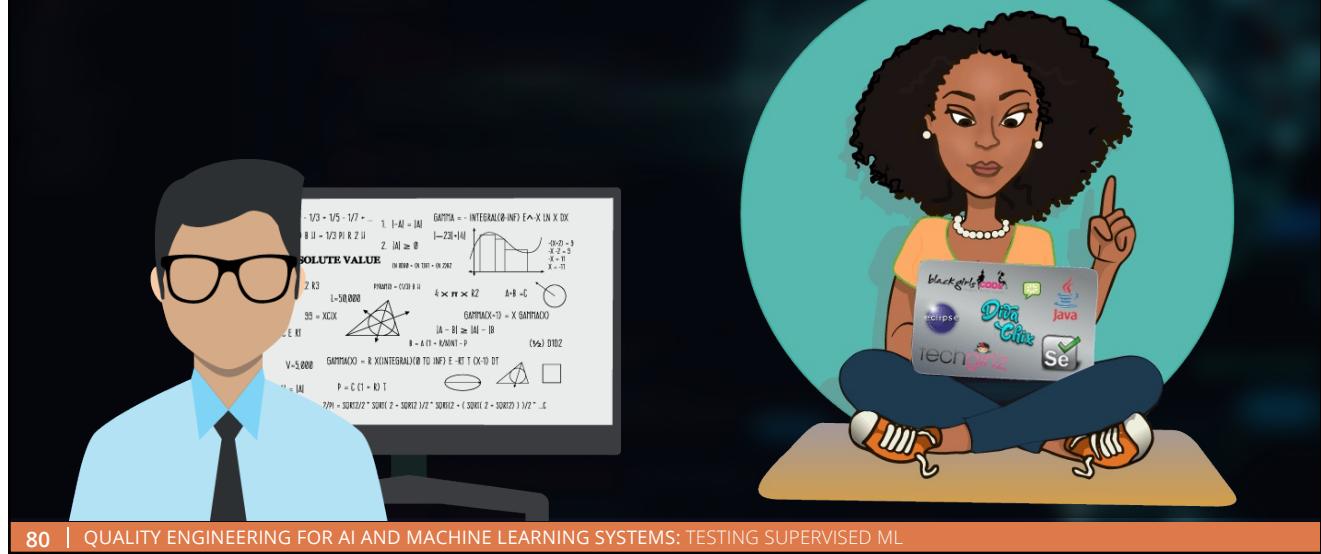
TESTING SUPERVISED ML

AI FAIRNESS ASSESSMENT & RISK MITIGATION

79

TESTING THE TRAINING DATA

DATA SCIENCE AND ML MEETS TESTING



80 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS: TESTING SUPERVISED ML

80

TESTING THE TRAINING DATA

POTENTIAL TRAINING DATA ISSUES



INSUFFICIENT
DATA



POOR QUALITY
DATA



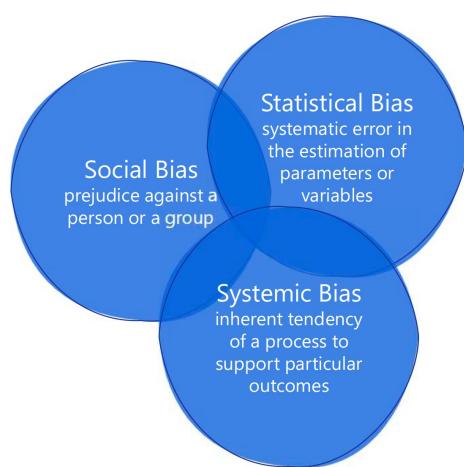
NON-REPRESENTATIVE
DATA

81 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS: TESTING SUPERVISED ML

81

TESTING AI FAIRNESS

TERMINOLOGY



82 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS: TESTING SUPERVISED ML

82

TESTING AI FAIRNESS

TERMINOLOGY



83 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS: TESTING SUPERVISED ML

83

TESTING AI FAIRNESS

GOAL



Are there **groups of people** who are **disproportionately, negatively impacted** by the system?

84 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS: TESTING SUPERVISED ML

84

TESTING AI FAIRNESS

4 STEPS IN ASSESSING FAIRNESS



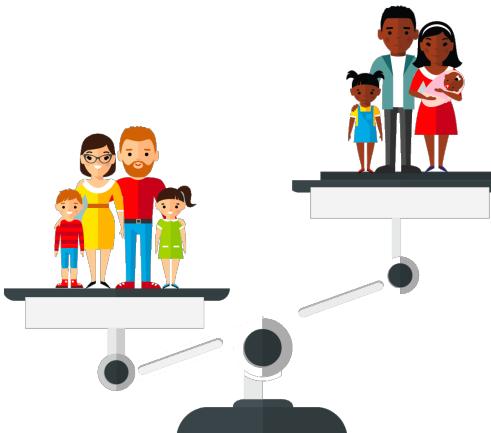
Identify Risks



Determine Impacted Groups



Quantify Risks



Compare Quantified Risks Across Groups

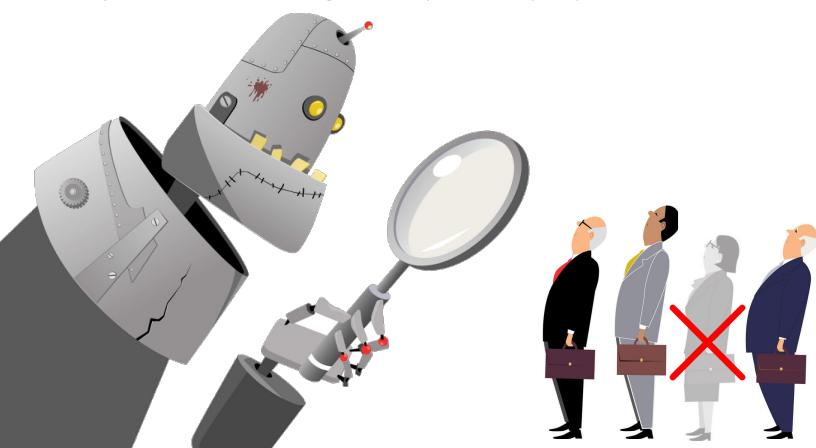
85 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS: TESTING SUPERVISED ML

85

TYPES OF FAIRNESS RISKS

ALLOCATION RISK

Occurs when an AI system is used to assign opportunities or resources in ways that can have negative impacts on people's lives.



In 2014, a team of engineers at Amazon began working on a project to automate hiring at the company.

The goal was to build an ML-based algorithm that could review resumes and determine which applicants Amazon should bring on board.

The project was canned a year later, when it became clear that the tool systematically discriminated against women applying for technical jobs.

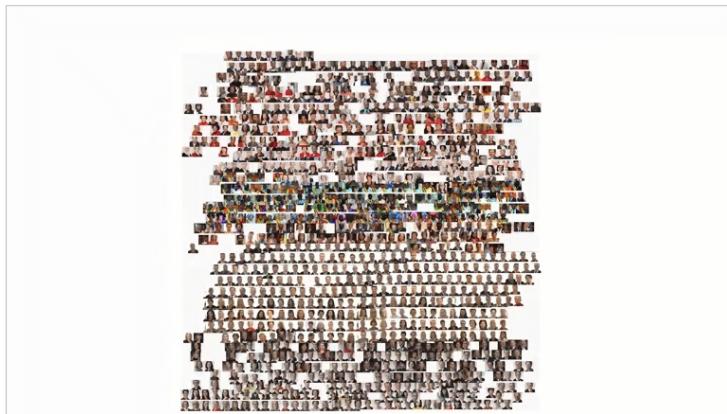
86 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS: TESTING SUPERVISED ML

86

TYPES OF FAIRNESS RISKS

QUALITY OF SERVICE RISK

Occurs when an AI system does not work as well for one person as it does for another.



In 2018, researchers evaluated products from IBM, Face++, and Microsoft that all implement AI-based gender classification.

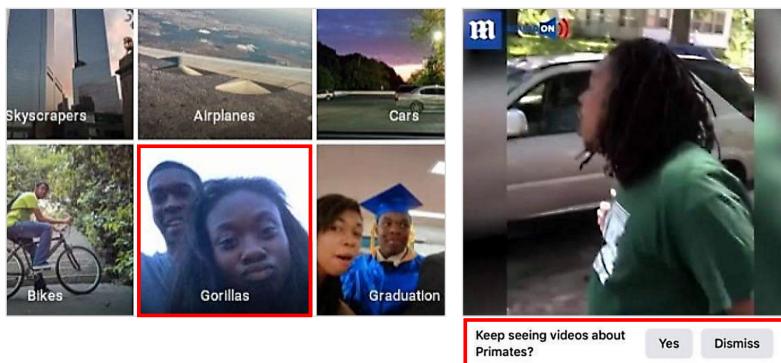
Although the highest rated product had an overall accuracy of 94%, all products performed better on males than females, and all products performed better on lighter-skinned subjects than darker-skinned.

An analysis of intersectional groups revealed that all products performed worst on darker-skinned females.

TYPES OF FAIRNESS RISKS

REPRESENTATION RISK

Occurs when an AI system performs tasks that include stereotyping, denigration, and any form of over- or under-representation.



In 2015, an image recognition algorithm in Google's Photos app mistakenly tagged a black couple as being "gorillas". Nearly three years later, the algorithm was "fixed" by blocking it from identifying gorillas altogether.

In 2020, Facebook's AI asked users watching a video featuring Black men if they wanted to see more "videos about primates".

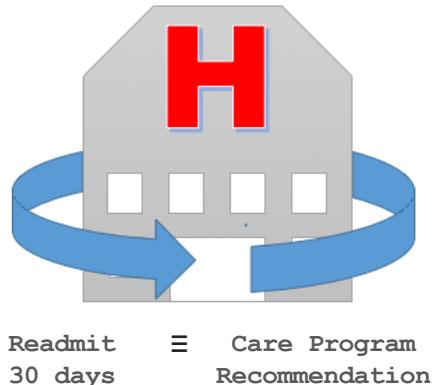
Both instances of mislabeling are harmful because such labels have a history of being used to demean Black people.

HANDS-ON: CONTEXT FOR ACTIVITIES

SCENARIO

Consider an automated system for **recommending patients for high-risk care management programs**.

- Such a system may be implemented as an ML classifier that predicts whether a patient should be suggested for enrollment in the program.
- Historical information on patients who had to be readmitted to the hospital within 30 days after release, can be used as an indicator that a patient should be recommended to the program.

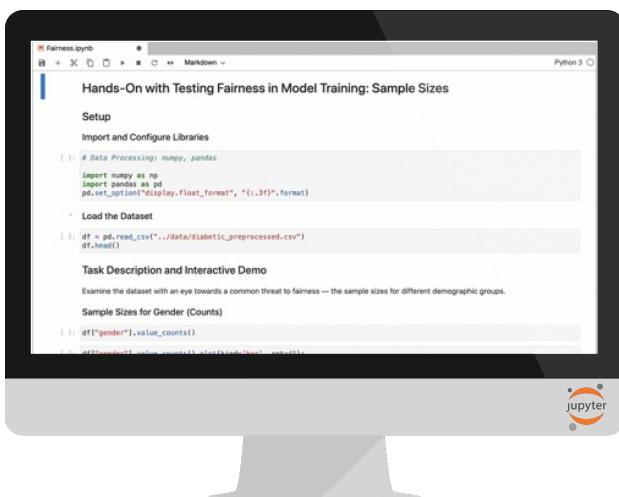


89 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS: TESTING SUPERVISED ML

89

HANDS-ON: FAIRNESS IN TRAINING DATA

SAMPLE SIZES INTERACTIVE DEMO & EXERCISE



90 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS: TESTING SUPERVISED ML

90

HANDS-ON: FAIRNESS IN TRAINING DATA

LABEL CHOICE INTERACTIVE DEMO & EXERCISE

The screenshot shows a Jupyter Notebook interface with the following content:

Hands-On with Testing Fairness in Training Data: Label Choice

Before you can adequately evaluate the care program recommendation model for fairness, you must check whether the choice of classification label, **readmission within 30 days**, aligns with the expectations that the system shall **identify patients that would benefit** from the care management program. In other words, the following assumption was made during model development:

The greatest benefit from the care management program would go to patients most likely to be readmitted within 30 days.

Now you need to test that assumption!

Task Description

Formally stated, the task is to investigate whether the selected measurement `readmit_30_days` correlates with patient characteristics that support the construct **benefit from care management**. One such characteristic is general patient health, where it is reasonable to expect that patients that are less healthy, are more likely to benefit from care management. Although the dataset does not contain full health records for measuring general patient health, it does contain two relevant features on a patient's health during the preceding year. These are:

Jupyter logo



91 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS: TESTING SUPERVISED ML

91

HANDS-ON: TESTING AI FAIRNESS

MITIGATING FAIRNESS RISKS WITH AI FAIRNESS 360



AI Fairness 360

The screenshot shows the AI Fairness 360 web interface with the following steps:

- 1. Choose sample data set**

Bias occurs in data used to train a model. We have provided three sample datasets that you can use to explore bias checking and mitigation. Each dataset has a detailed description below. You should be prompted to avoid bias.

- 0 Compas (ProPublica recidivism)**
Predict a criminal defendant's likelihood of reoffending.
Protected Attributes:
 - Sex: privileged: Female, unprivileged: Male
 - Race: privileged: Caucasian, unprivileged: Not Caucasian[Learn more](#)
- 1 German credit scoring**
Predict an individual's credit risk.
Protected Attributes:
 - Age: privileged: Old, unprivileged: Young[Learn more](#)
- 2 Adult census income**
Predict whether income exceeds \$50K yr based on census data.
Protected Attributes:
 - Race: privileged: White, unprivileged: Non-white
 - Sex: privileged: Male, unprivileged: Female[Learn more](#)

Next

<https://tinyurl.com/testingml>

Jupyter logo

92 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS: TESTING SUPERVISED ML

92

TESTING AI FOR FAIRNESS

RISK MITIGATION DISCUSSION

Optimized Pre-processing Use to mitigate bias in training data. Modifies training data features and labels. →	Reweighting Use to mitigate bias in training data. Modifies the weights of different training examples. →	Adversarial Debiasing Use to mitigate bias in classifiers. Uses adversarial techniques to maximize accuracy and reduce evidence of protected attributes in predictions. →	Reject Option Classification Use to mitigate bias in predictions. Changes predictions from a classifier to make them fairer. →	Disparate Impact Remover Use to mitigate bias in training data. Edits feature values to improve group fairness. →
Learning Fair Representations Use to mitigate bias in training data. Learns fair representations by obfuscating information about protected attributes. →	Prejudice Remover Use to mitigate bias in classifiers. Adds a discrimination-aware regularization term to the learning objective. →	Calibrated Equalized Odds Post-processing Use to mitigate bias in predictions. Optimizes over calibrated classifier score outputs that lead to fair output labels. →	Equalized Odds Post-processing Use to mitigate bias in predictions. Modifies the predicted labels using an optimization scheme to make predictions fairer. →	Meta Fair Classifier Use to mitigate bias in classifier. Meta algorithm that takes the fairness metric as part of the input and returns a classifier optimized for that metric. →

93 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS: TESTING SUPERVISED ML

93

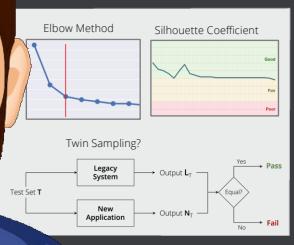
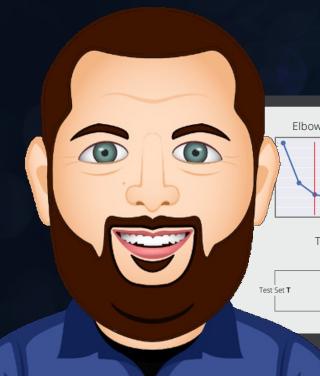
TESTING UNSUPERVISED ML

CLUSTERING VALIDATION

94

TESTING UNSUPERVISED ML

DATA SCIENCE, ML, TESTING CROSS-OVER



95 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS: TESTING UNSUPERVISED ML

95

CLUSTER VALIDATION

INTRODUCTION



Affinity Propagation
Agglomerative Clustering

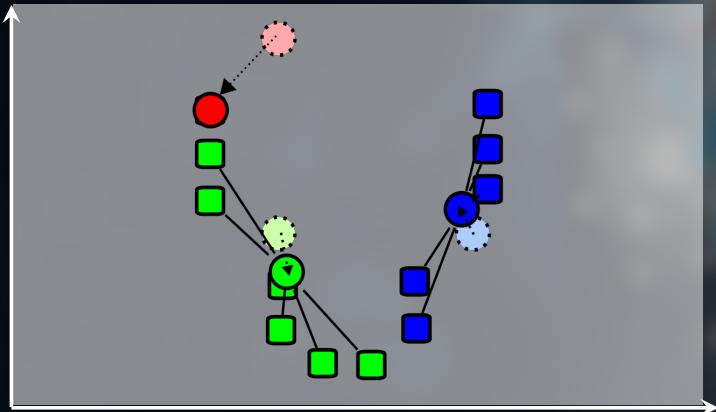
BIRCH
DBSCAN
K-Means
Mini-Batch K-Means
Mean Shift
OPTICS
Spectral Clustering
Mixture of Gaussians

96 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS: TESTING UNSUPERVISED ML

96

K-MEANS CLUSTERING

HOW IT WORKS



K-Means Algorithm

1. Randomly place K centroids for the initial clusters.
2. Assign each data point to their nearest centroid.
3. Update centroid locations based on the locations of the data points.
4. Repeat 2. and 3. until points don't move between clusters and centroids stabilize.

CLUSTERING: GOALS & CHALLENGES

Goal

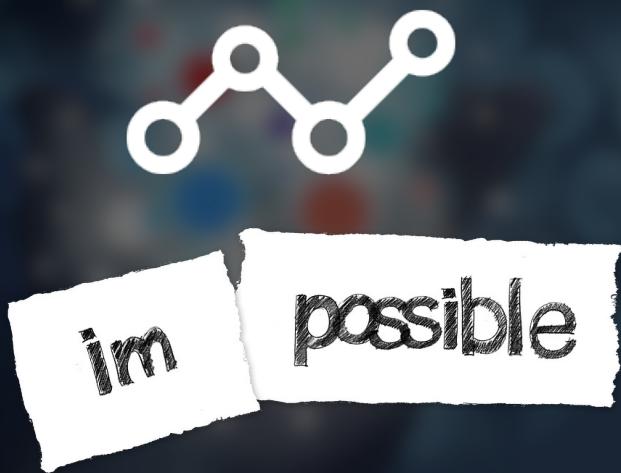
- o Partition unlabeled data

Properties of Good Clustering

- o Scale Invariance
- o Consistency
- o Richness or Wealth

Challenges

- o Cannot be precisely defined
- o Impossibility Theorem



VALIDATION GOALS AND CHALLENGES



Optimal Number of Clusters



Comparing Sets of Clusters



Evaluating Results
(with or without oracle)

99 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS: TESTING UNSUPERVISED ML

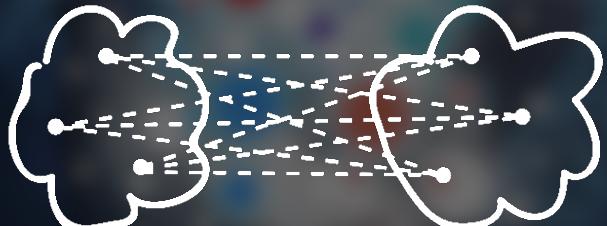
99

INTERNAL VALIDATION

COHESION AND SEPARATION



Cohesion (C_k)



Separation (C_j, C_k)

100 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS: TESTING UNSUPERVISED ML

100

INTERNAL VALIDATION

COMBINED METRICS

- Silhouette Coefficient
- Calisnki-Harabasz Coefficient
- Dunn Index
- Xie-Beni Score
- Hartigan Index
- and more...



101 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS: TESTING UNSUPERVISED ML

101

EXTERNAL VALIDATION

SUBJECT MATTER EXPERTS DEFINE TRUE LABELS



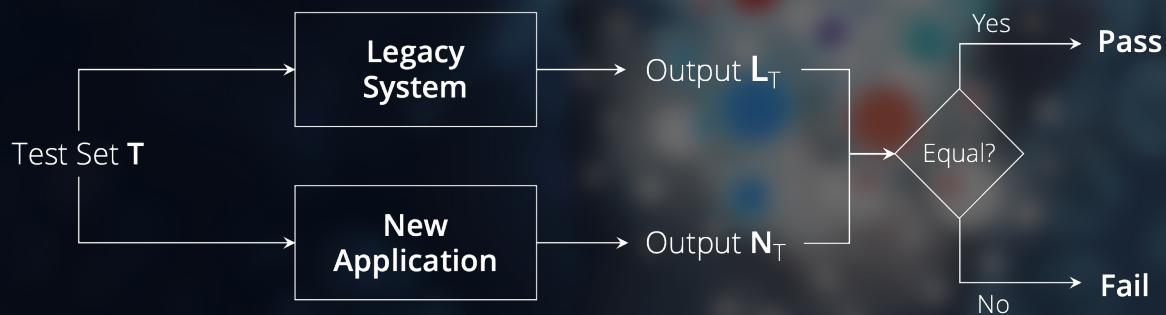
		Predicted Class	
		Yes	No
Actual Class	Yes	TRUE POSITIVES	FALSE NEGATIVES
	No	FALSE POSITIVES	TRUE NEGATIVES

102 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS: TESTING UNSUPERVISED ML

102

CROSS-OVER VALIDATION

PARALLEL VALIDATION IN SOFTWARE TESTING



103 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS: TESTING UNSUPERVISED ML

103

CROSS-OVER VALIDATION

TWIN VALIDATION IN TESTING ML

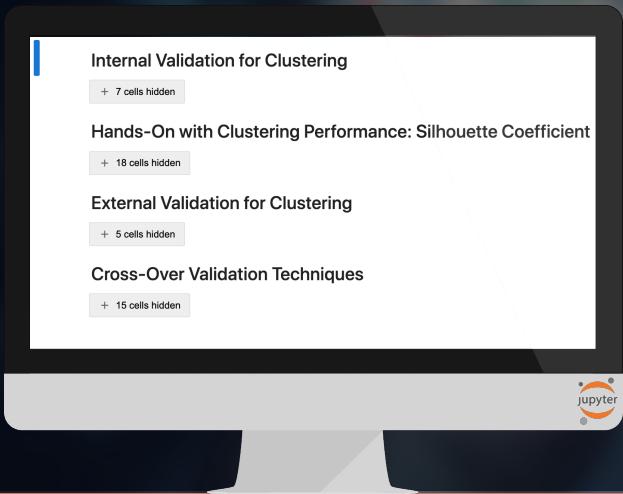


1. Create a twin-sample of the training data
2. Perform unsupervised learning on twin-sample
3. Import results for twin-sample from training set
4. Calculate similarity between the two sets of results

104 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS: TESTING UNSUPERVISED ML

104

HANDS-ON: CLUSTER VALIDATION



105 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS: TESTING UNSUPERVISED ML

105

USER TESTING

TESTING ADAPTIVE ML

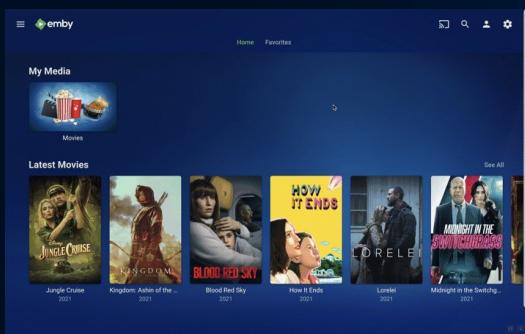


106 | EXPLORING THE INTERSECTION OF AI FOR TESTING: TESTING AI AND ML

106

SOFTWARE HAS BECOME MORE DYNAMIC

LEARN AND ADAPT IN REAL-TIME



Recommendation Systems



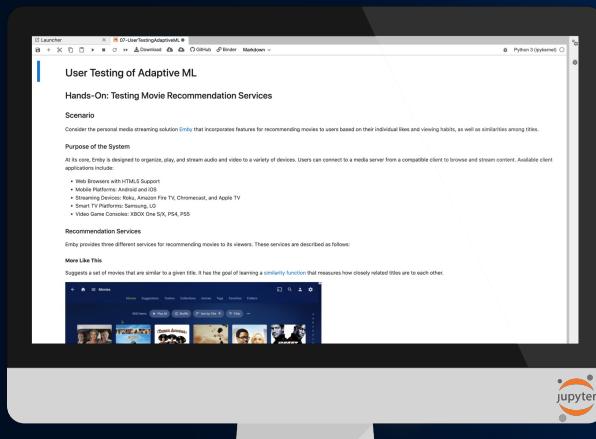
Self-Driving Cars

107 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS: USER TESTING ADAPTIVE AI/ML

107

HANDS-ON: USER TESTING AI/ML

TESTING MOVIE RECOMMENDATIONS



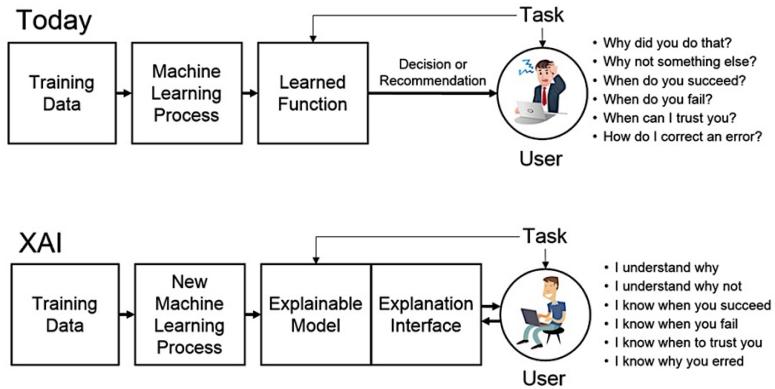
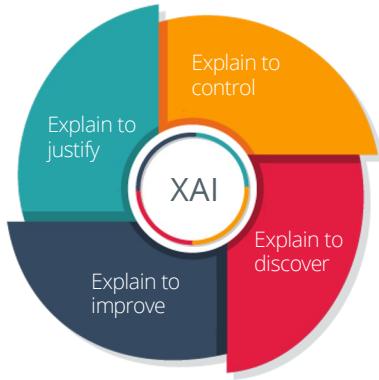
108 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS: USER TESTING ADAPTIVE AI/ML

108

EXPLAINABLE AI: A DARPA INITIATIVE



TOWARDS TRUSTED AI AND MACHINE LEARNING



109 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS: USER TESTING ADAPTIVE AI/ML

109

EXPLAINABLE AI: AI EXPLAINABILITY 360

FICO EXPLAINABLE ML CHALLENGE



AI Explainability 360

110 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS: USER TESTING ADAPTIVE AI/ML

110

USER TESTING AI/ML: KEY CONSIDERATIONS

SYSTEMS THAT LEARN AT RUNTIME, MUST BE TESTED IN PRODUCTION



Testing Algorithm in a Lab
Sandbox Environment

De Facto Standard: Split Validation
Academic Measures

ISOLATED AND INADEQUATE

Testing System in the Wild
Production Environment

De Facto Standard: A/B Testing
Realistic Measures

INTEGRATED AND ADEQUATE

111 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS: USER TESTING ADAPTIVE AI/ML

111



112 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS: USER TESTING ADAPTIVE AI/ML

112



FROM NOW WE TEST IN PRODUCTION!

113 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS: USER TESTING ADAPTIVE AI/ML

113



**TIP ≠ INSUFFICIENT
PRE-PRODUCTION
TESTING**

114 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS: USER TESTING ADAPTIVE AI/ML

114

TIP CHALLENGES

- Breaking Production Functionality
- Loss or Corruption of Production Data
- Interruption of Regular Operations
- Narrowed Test Window
- System Performance Degradation
- Complicates Security Measures
- Non-Compliance with Standards

TIP BENEFITS

- Tests Mirror the Real World
- Quick Feedback on Impact of Changes
- Preventing Bad Deployments
- Catching Issues Before Users Notice Them
- Some Bugs are Easier to Reproduce
- Ensures Software is Production Ready
- Helps in Designing Resilient Systems

115 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS: USER TESTING ADAPTIVE AI/ML

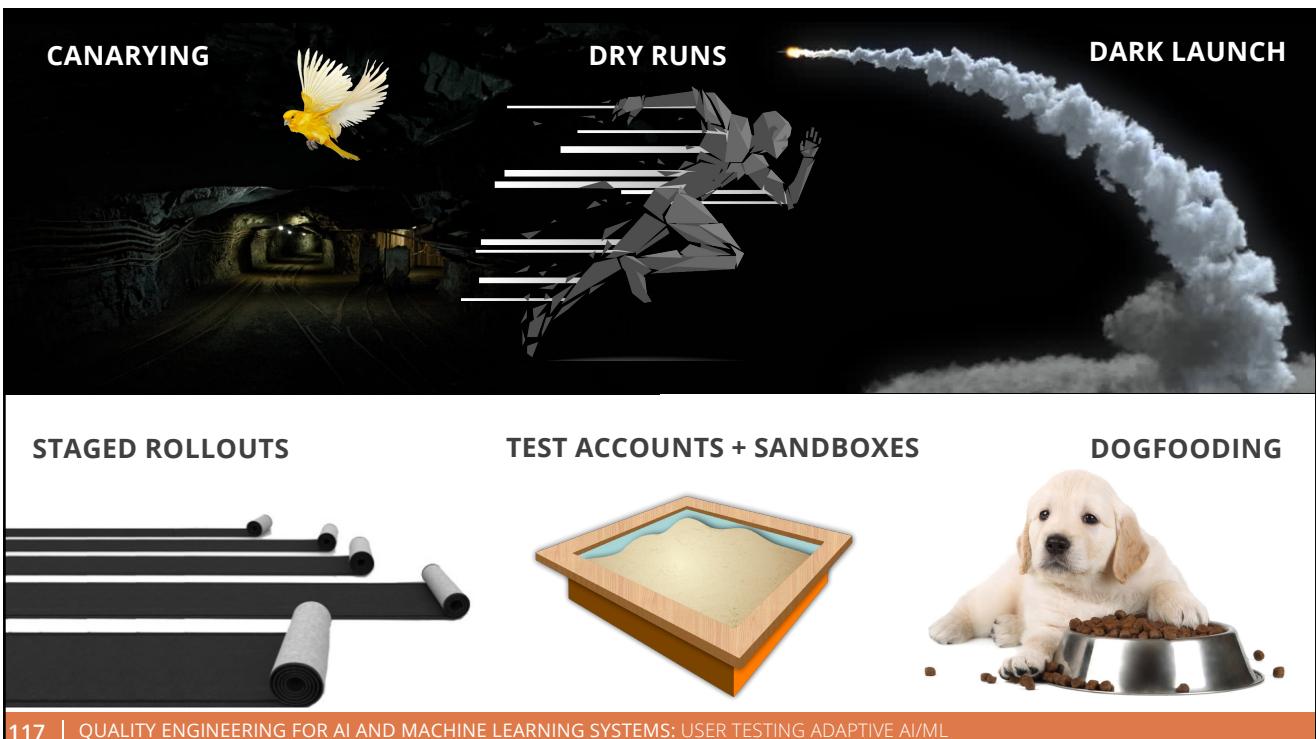
115

APPROACHES TO

TIP

116 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS: USER TESTING ADAPTIVE AI/ML

116



117 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS: USER TESTING ADAPTIVE AI/ML

117

NORA JONES

Testing in Production
The Netflix Way

Testing in Production
The Netflix Way
-@nora_js-

NETFLIX

118 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS: USER TESTING ADAPTIVE AI/ML

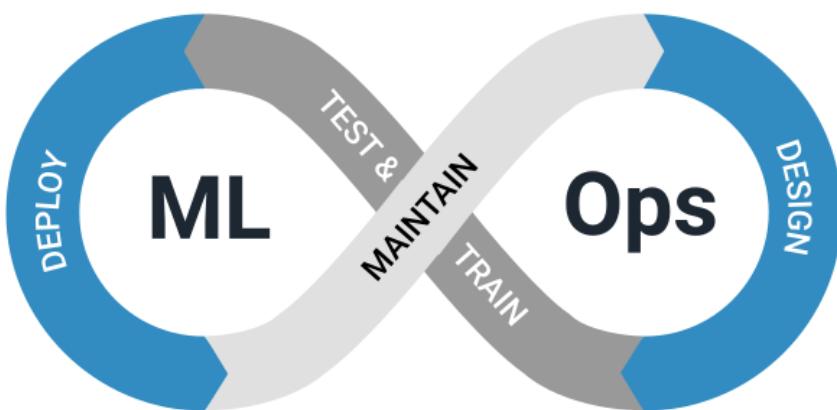
118



119

KEY TAKEAWAY

CONTINUOUS LEARNING REQUIRES CONTINUOUS TESTING



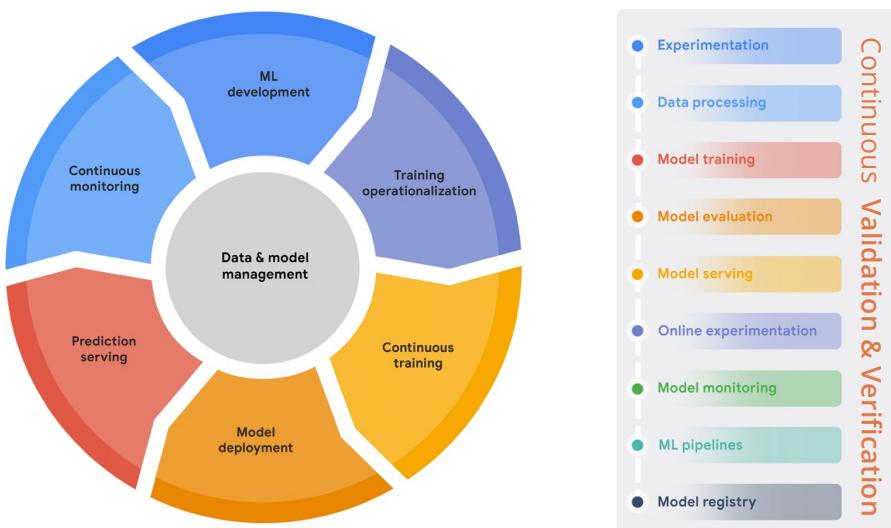
120 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS: USER TESTING ADAPTIVE AI/ML

120

AI/ML OPERATIONS

121

ML OPERATIONS: LIFECYCLE & ACTIVITIES

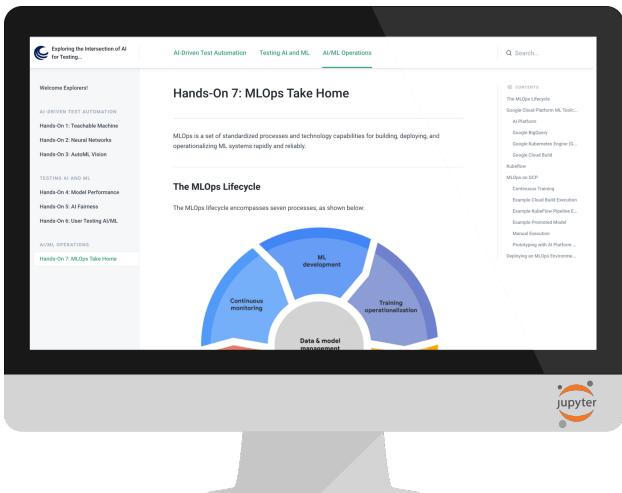


122 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS: ML OPERATIONS

122

HANDS-ON: ML OPS TAKE HOME

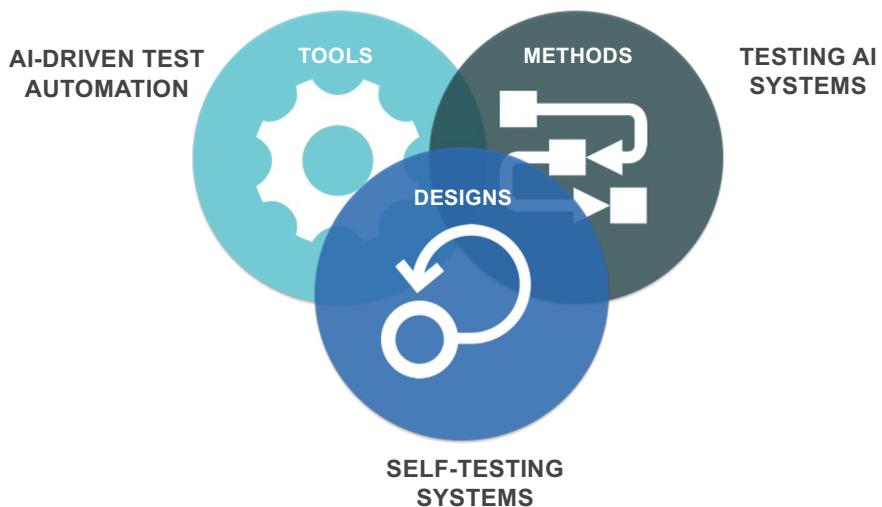
GOOGLE AI PLATFORM



123 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS: ML OPERATIONS

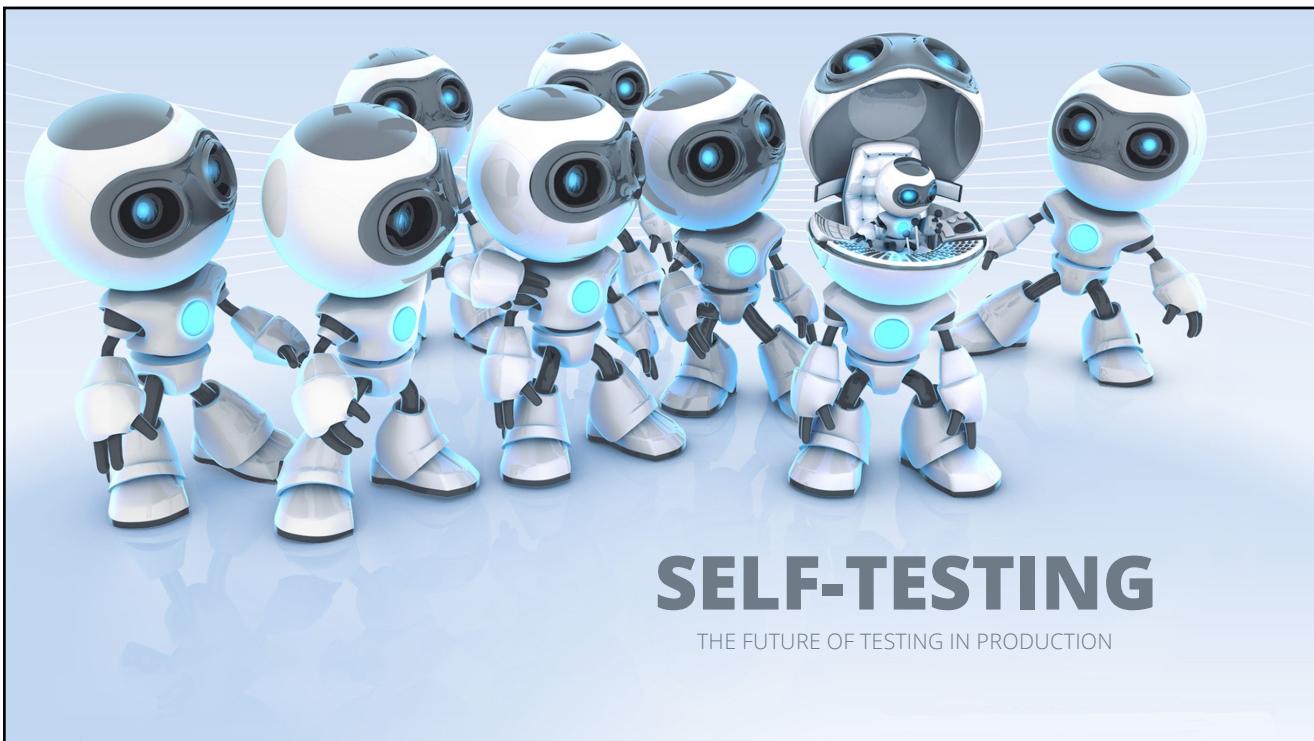
123

WRAP UP: AI FOR SOFTWARE TESTING

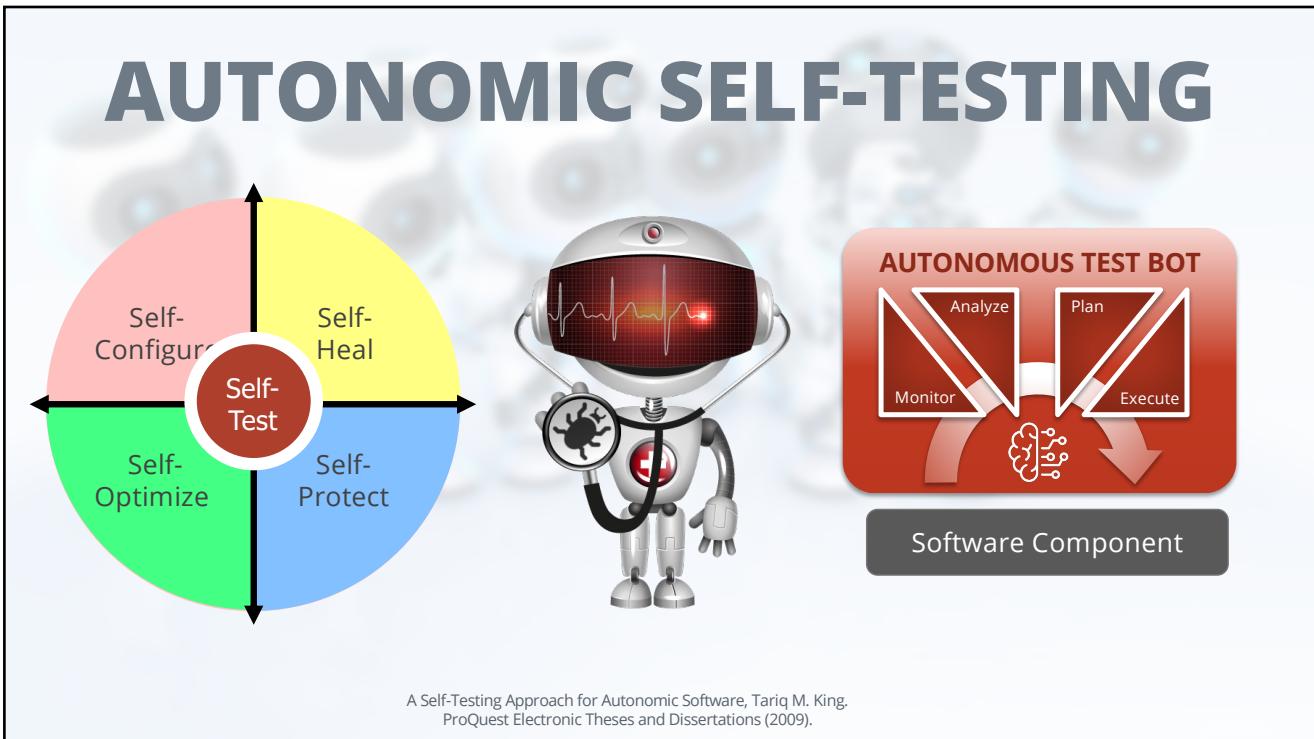


124 | QUALITY ENGINEERING FOR AI AND MACHINE LEARNING SYSTEMS

124



125

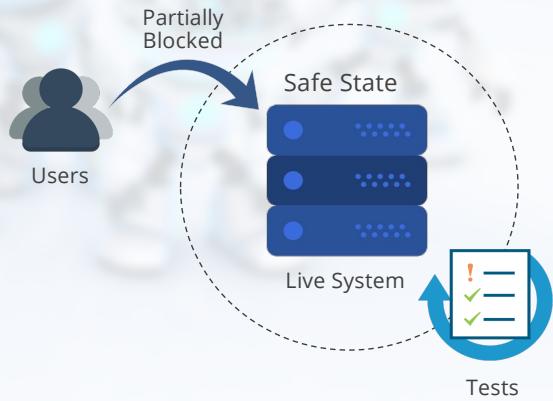


126

SELF-TESTING APPROACHES

SAFE ADAPTATION WITH VALIDATION

1. Bring the system into a **safe state**, where some functions are partially blocked.
2. Implement changes and perform runtime **testing in-place** on the live system.
3. Once **testing completes**, the system returns to its mode of **full operation**.



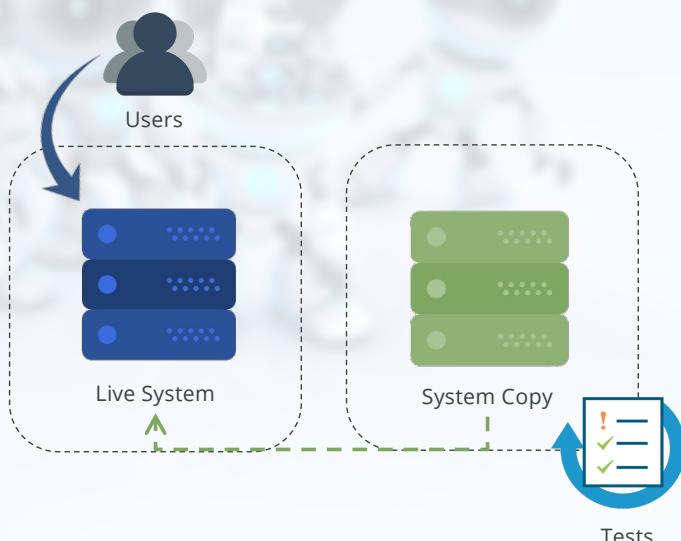
A Self-Testing Approach for Autonomic Software, Tariq M. King,
ProQuest Electronic Theses and Dissertations (2009).

127

SELF-TESTING APPROACHES

REPLICATION WITH VALIDATION

1. Create or maintain a **copy** of the system under test.
2. Implement any changes and perform runtime **testing on the copy** of the system.
3. Use the results of testing on the copy to decide whether to **adapt the live system**.

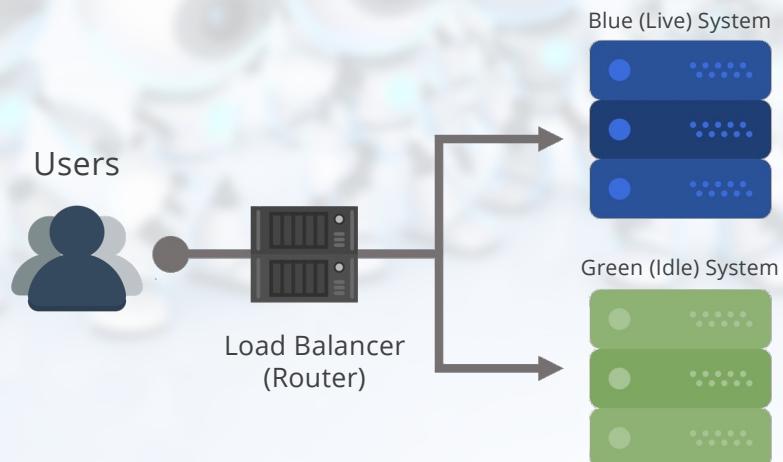


A Self-Testing Approach for Autonomic Software, Tariq M. King,
ProQuest Electronic Theses and Dissertations (2009).

128

BLUE-GREEN DEPLOYMENTS

ALSO KNOWN AS RED/BLACK DEPLOYMENTS



129

DEVOPS SUPPORTS SELF-TESTING

CI/CD REQUIRES CONTINUOUS TESTING



NEW CODE



EVERY BUILD



EVERY DEPLOY



IN PRODUCTION

SHIFT LEFT TESTING

SHIFT RIGHT TESTING

130

DEVOPS SUPPORTS SELF-TESTING

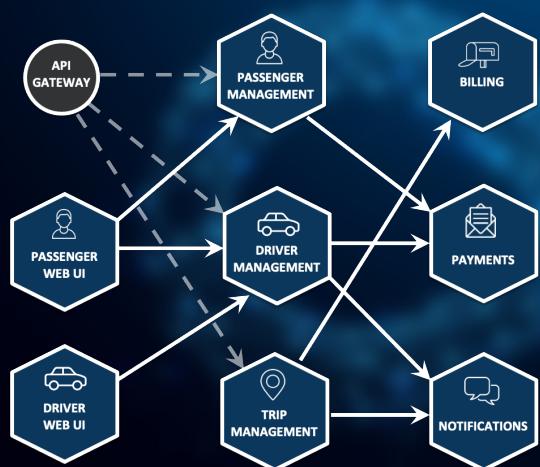
THE NEXT SHIFT IS INWARD



131

MICROSERVICES ARCHITECTURES

RESILIENT, CLUSTERED, MONITORED, SCALABLE TESTING



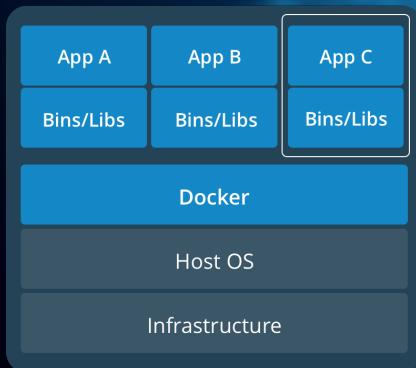
VS.



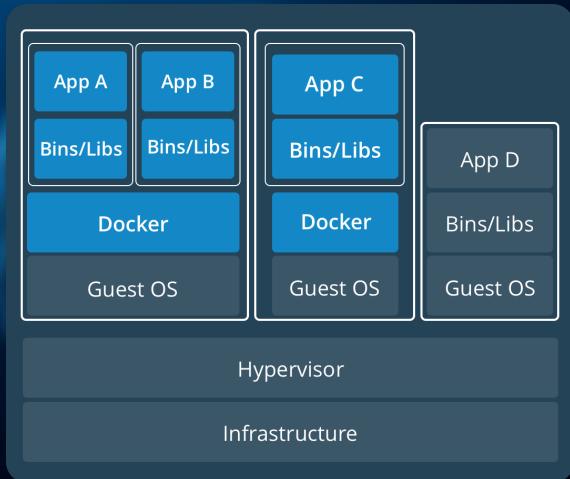
132

CONTAINERS

MAINTAINABLE, ON-DEMAND, PRODUCTION-READY TESTING



VS.

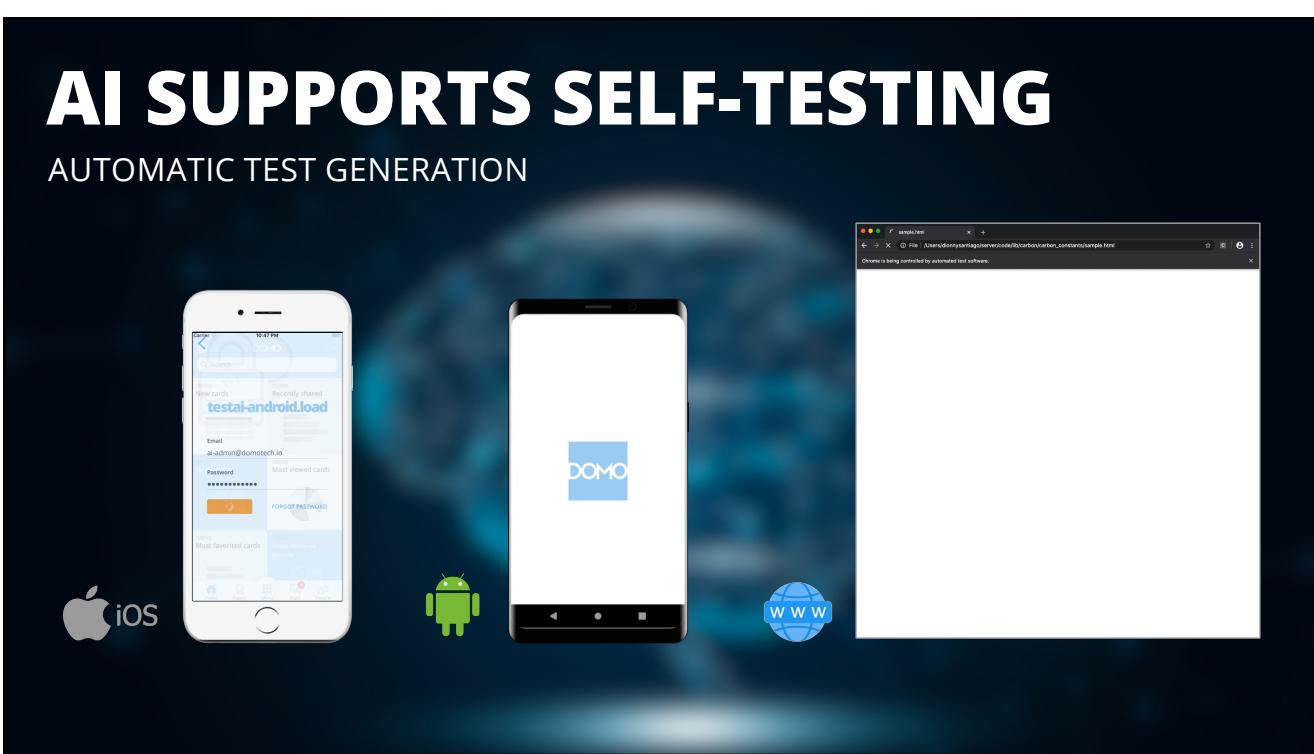


133

AI SUPPORTS SELF-TESTING



134



135



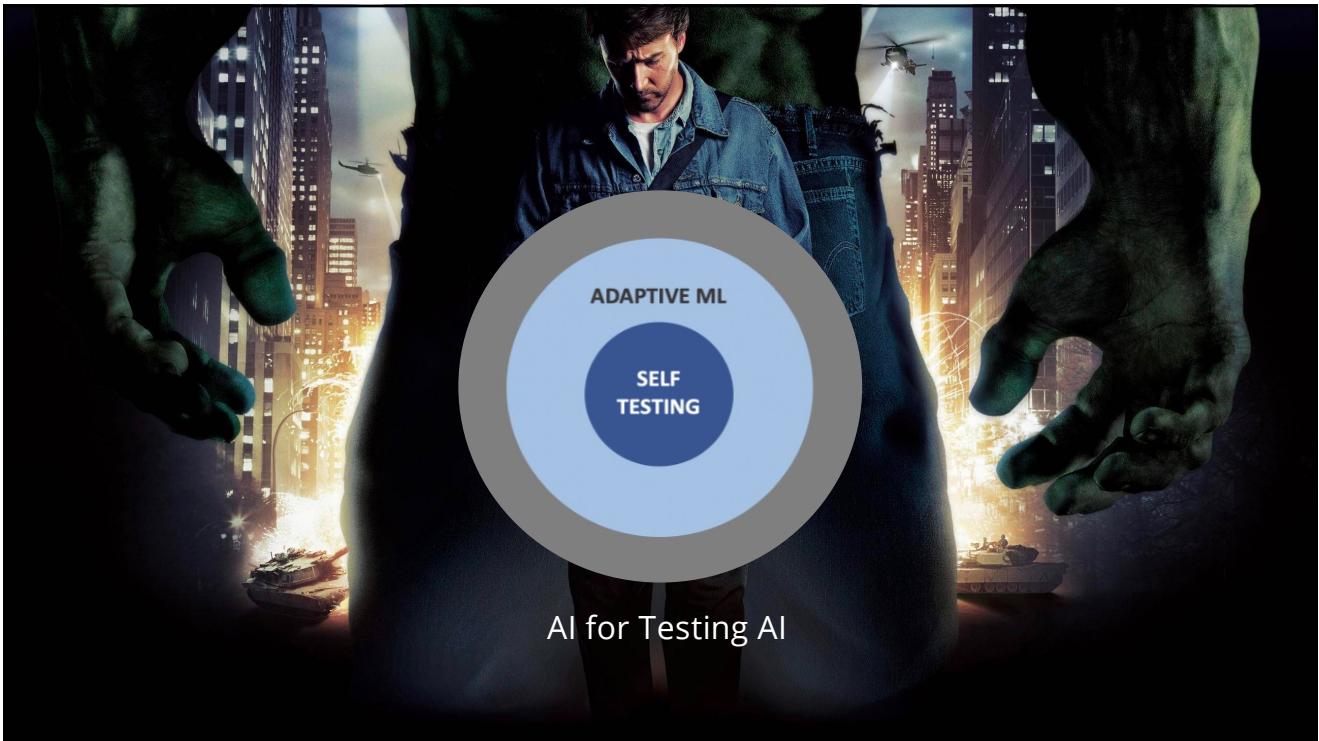
136



137



138



139

—
QUESTIONS?

THANK YOU



king.tariq@gmail.com



linkedin.com/in/tariqking



@tariq_king

140