MATLAB EXPO

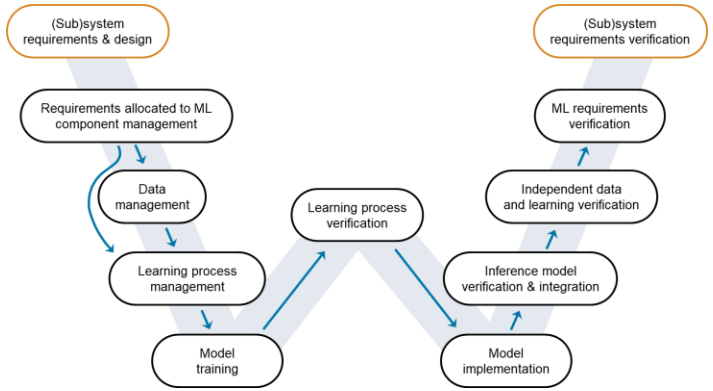
인공지능 모델의 이해와 검증

김종남 부장, 매스웍스코리아

MathWorks

Key Takeaways

MathWorks has capabilities addressing each area of the W-diagram



Library to verify and test robustness of deep learning networks

Deep Learning Toolbox Verification Library

by MathWorks Deep Learning Toolbox Team STAFF

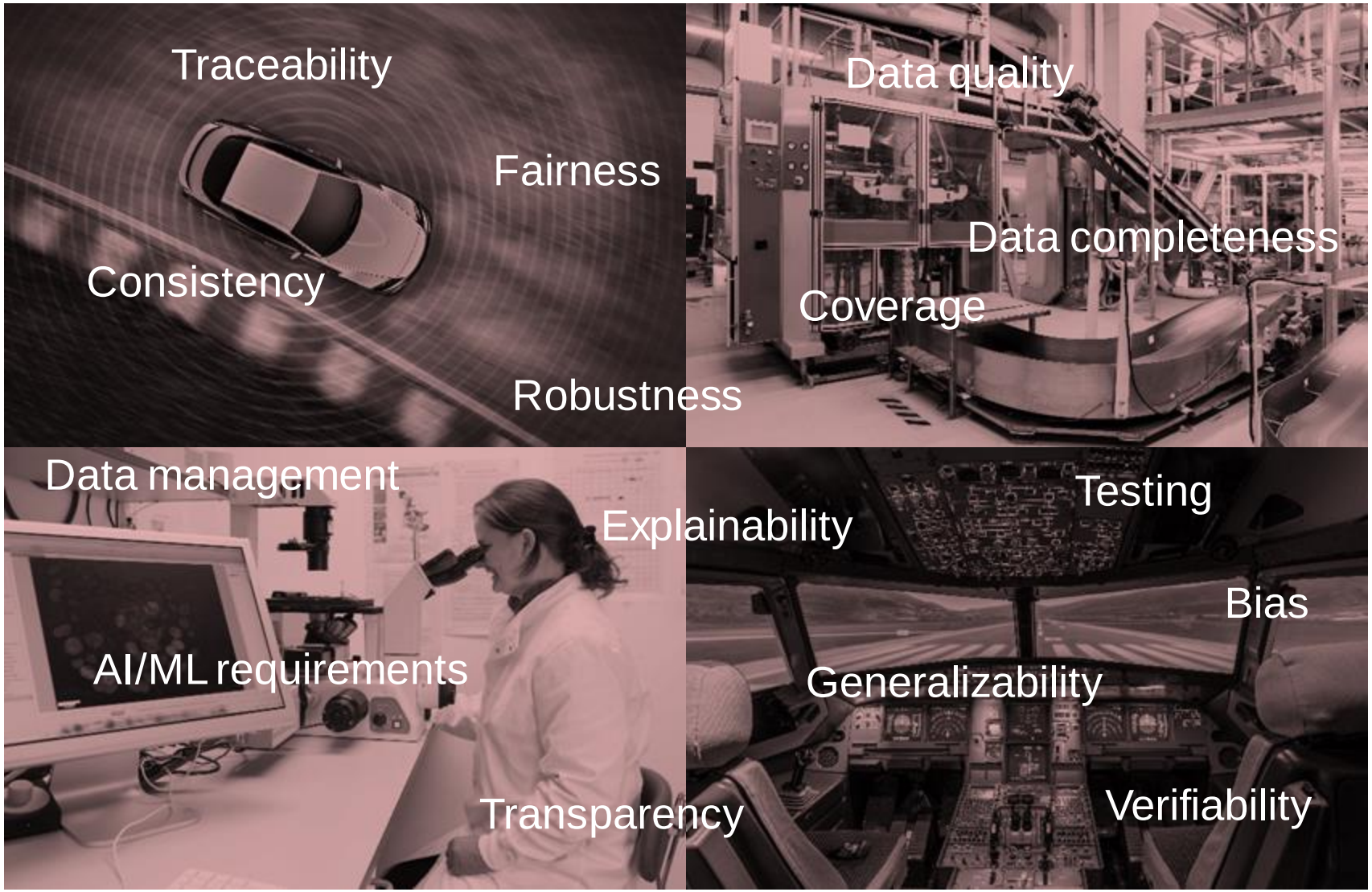
Verify and test robustness of deep learning networks

Our safety-critical certification expertise helps drive new AI standards

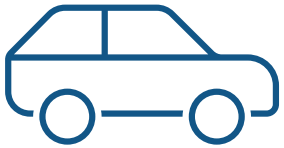
EUROCAE WG-114 / SAE G-34 Standardization Working Group "Artificial Intelligence in Aviation"

As AI use rises in production, there is a growing need to explain, verify and validate model behavior in safety-critical situations

Challenges in Verification and Validation of AI-enabled Systems

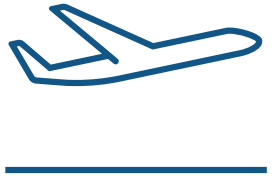


Industries are making progress on verifying AI in systems through whitepapers, standards and planning



Automotive

New WIP ISO PAS 8800 (Road Vehicles — Safety and artificial intelligence)



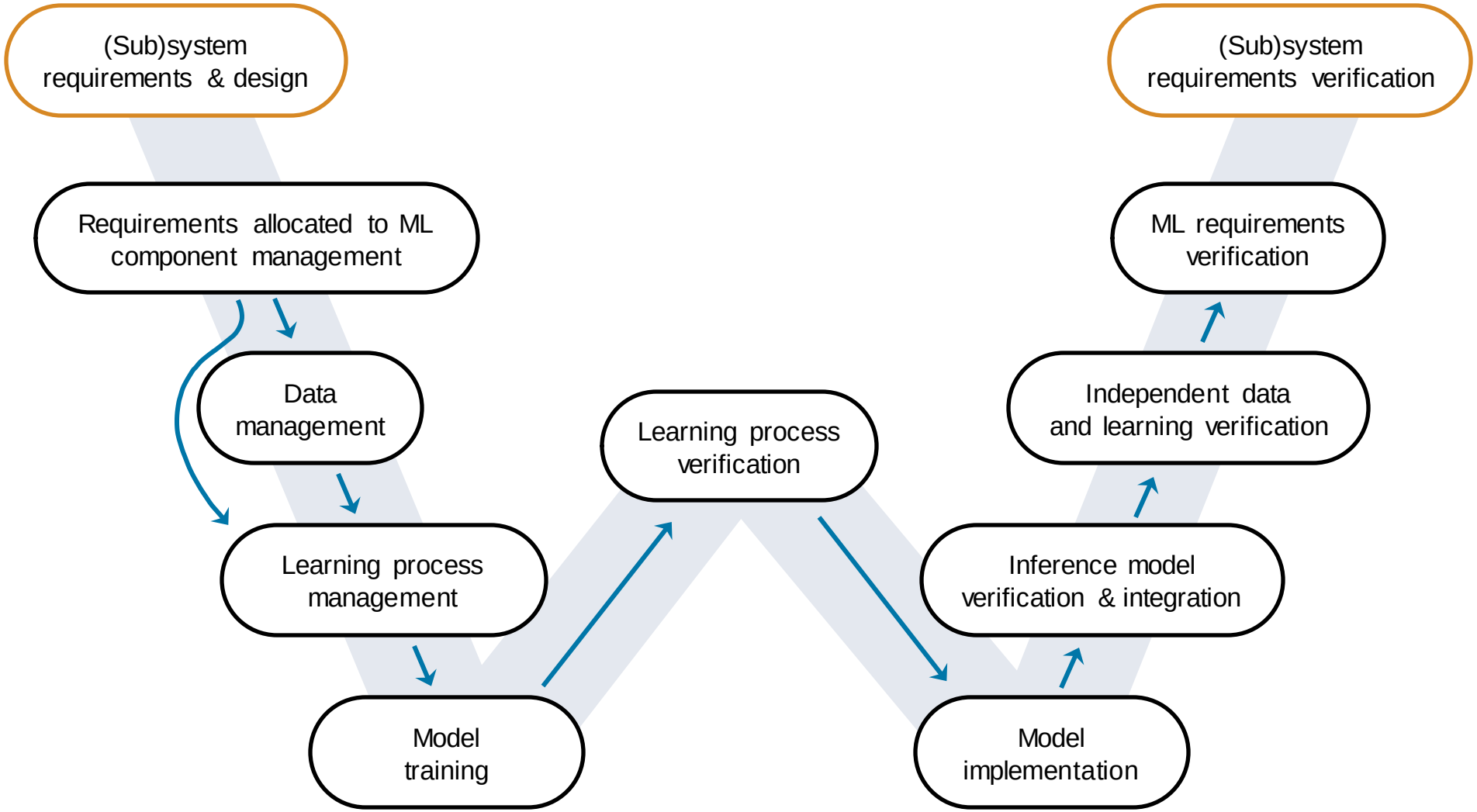
Aerospace

New standard (AS6983) from EUROCAE WG-114 / SAE G-34 is expected in 2024

Medical Devices

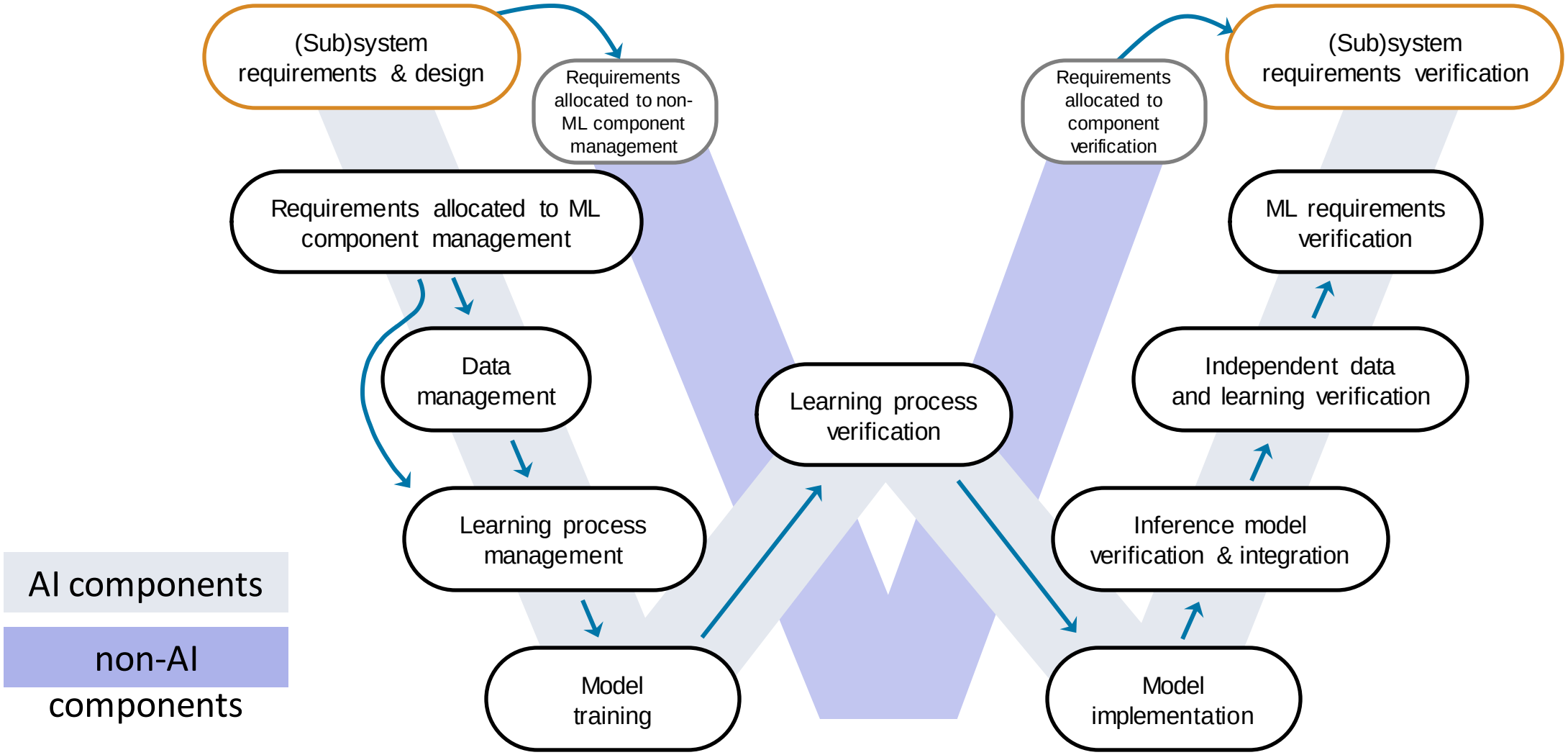
FDA released its first AI/ML-Based Software as a Medical Device (SaMD) Action Plan

W-shaped development process adapting the classical V-shaped cycle to AI applications



Credit: EASA, Daedalean

W-shaped development process can coexist with V-shaped cycle for non-AI components

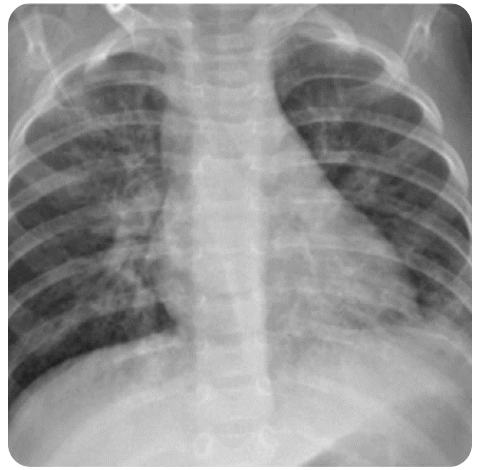


Task: Verify an image classification network

Automotive

Aerospace

Medical



MedMNIST v2 Dataset

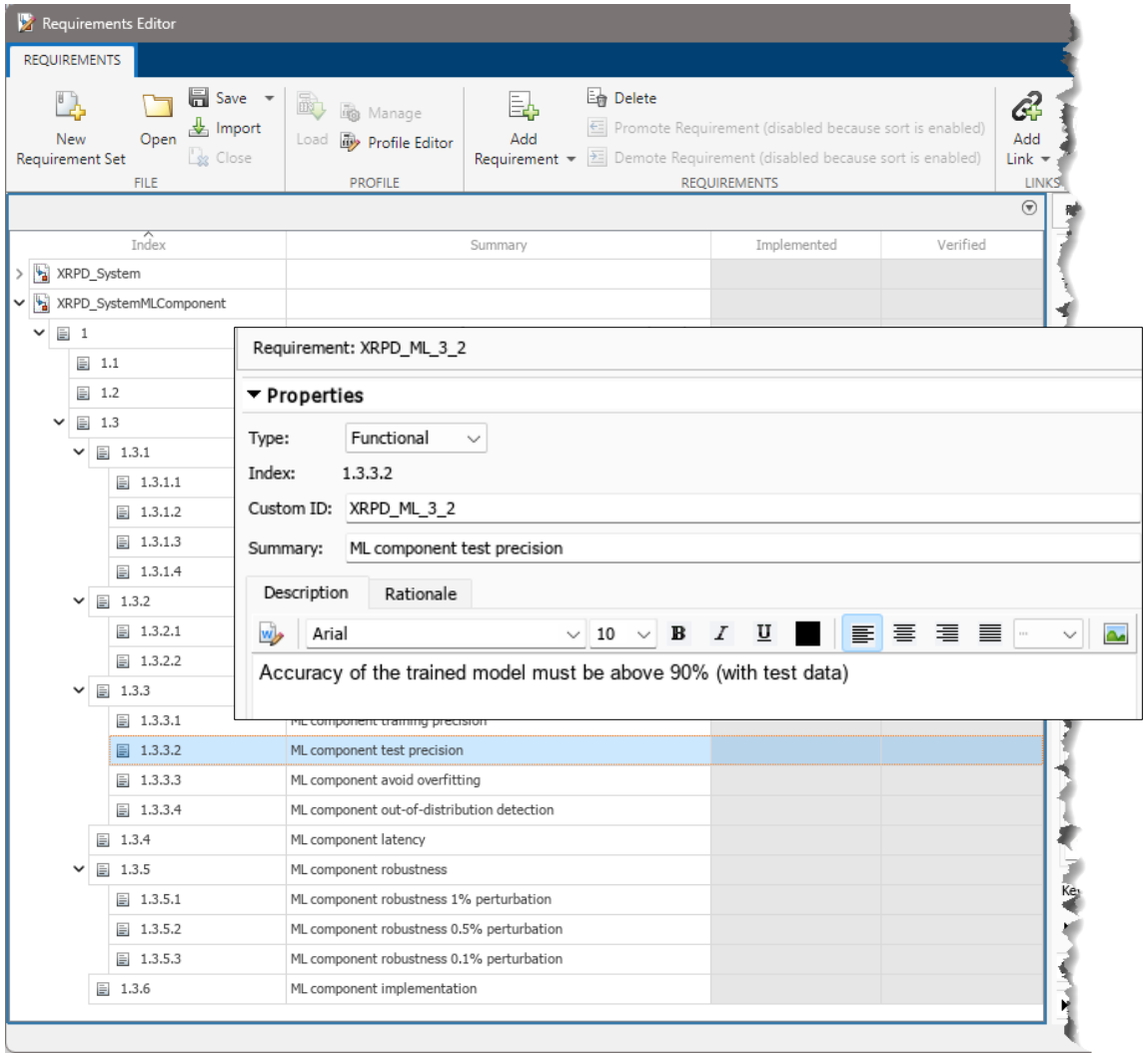
MedMNIST v2 - A large-scale lightweight benchmark for 2D and 3D biomedical image classification

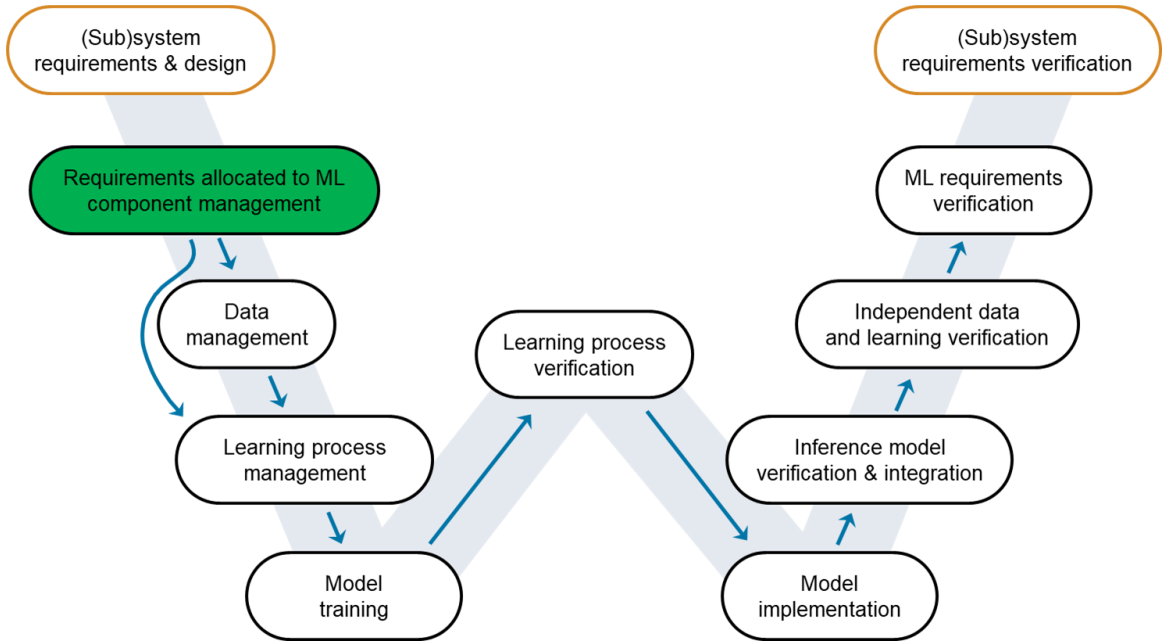
Jiancheng Yang, Rui Shi, Donglai Wei, Zequan Liu, Lin Zhao, Bilian Ke, Hanspeter Pfister, Bingbing Ni

[1] Shanghai Jiao Tong University, Shanghai, China
[2] Boston College, Chestnut Hill, MA
[3] RWTH Aachen University, Aachen, Germany
[4] Fudan Institute of Metabolic Diseases, Zhongshan Hospital, Fudan University, Shanghai, China
[5] Shanghai General Hospital, Shanghai Jiao Tong University School of Medicine, Shanghai, China
[6] Harvard University, Cambridge, MA

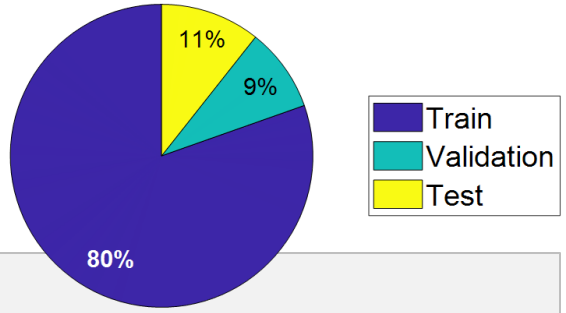
PneumoniaMNIST

Start by collecting requirements allocated to the ML component

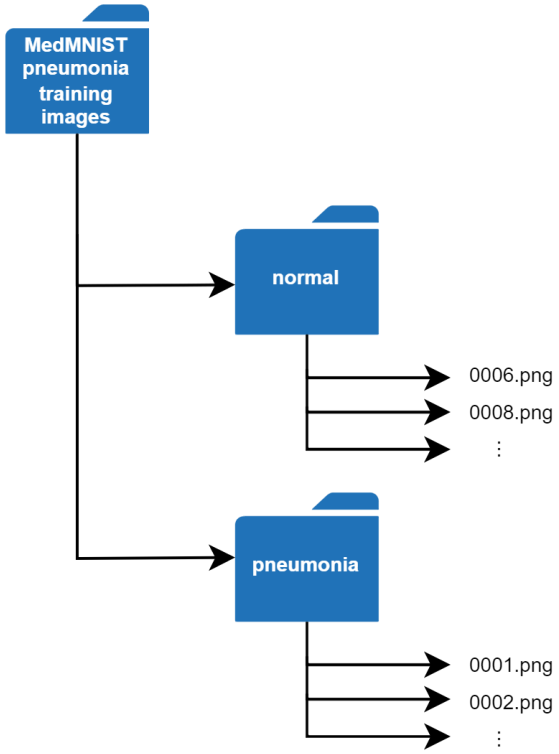




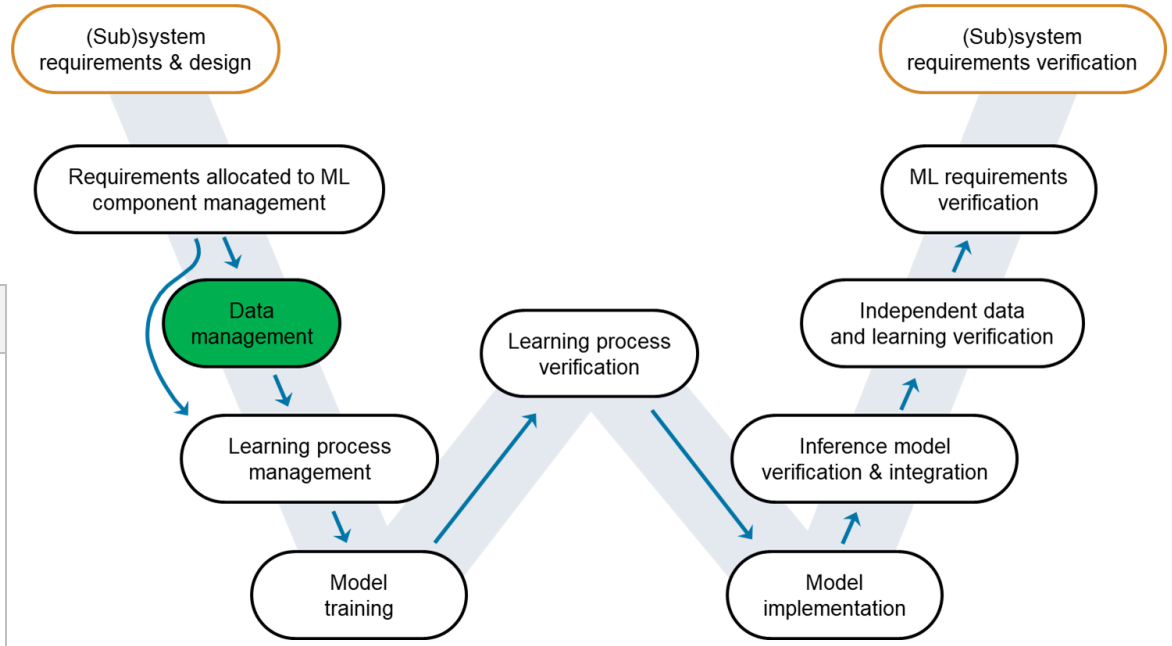
Conveniently manage large collections of data



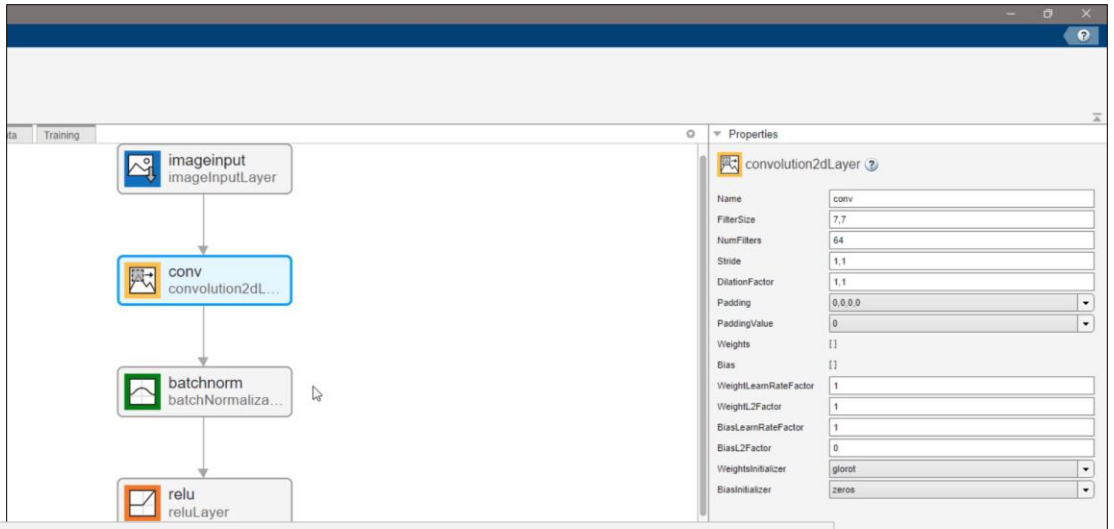

```
trainingDataFolder = "pneumoniamnist\Train";
imdsTrain = imageDatastore(trainingDataFolder,IncludeSubfolders=true,LabelSource="foldernames");
```




```
countEachLabel(imdsTrain)

ans =

  2×2 table

    Label       Count
    ________    _____

    normal      1214
    pneumonia   3494
```

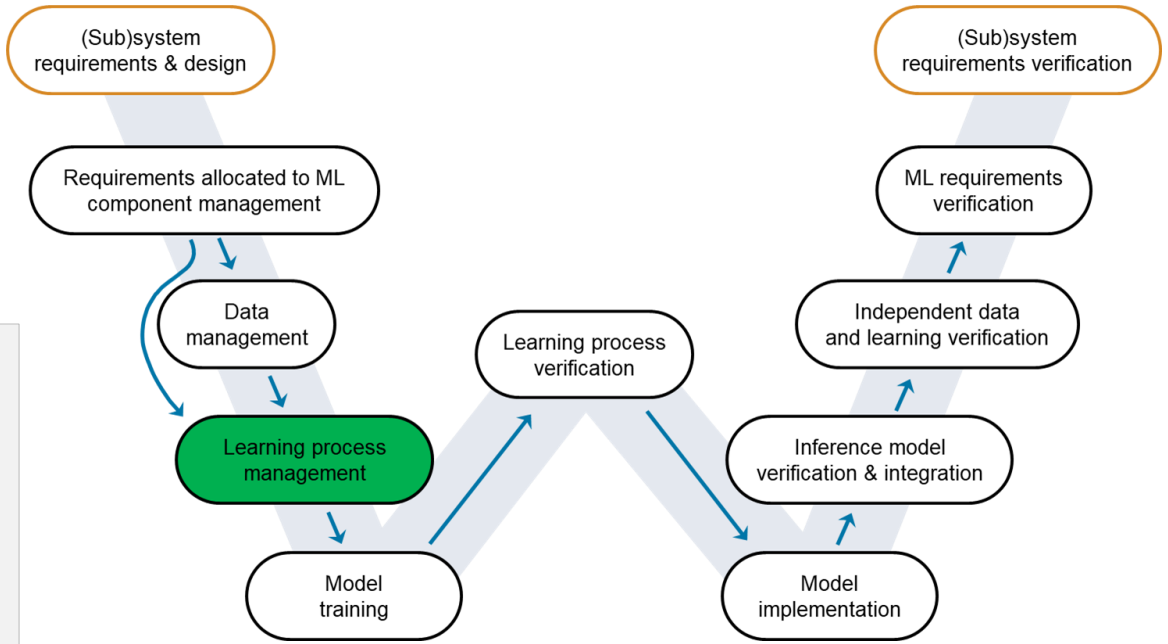


Visually creating networks enables faster design

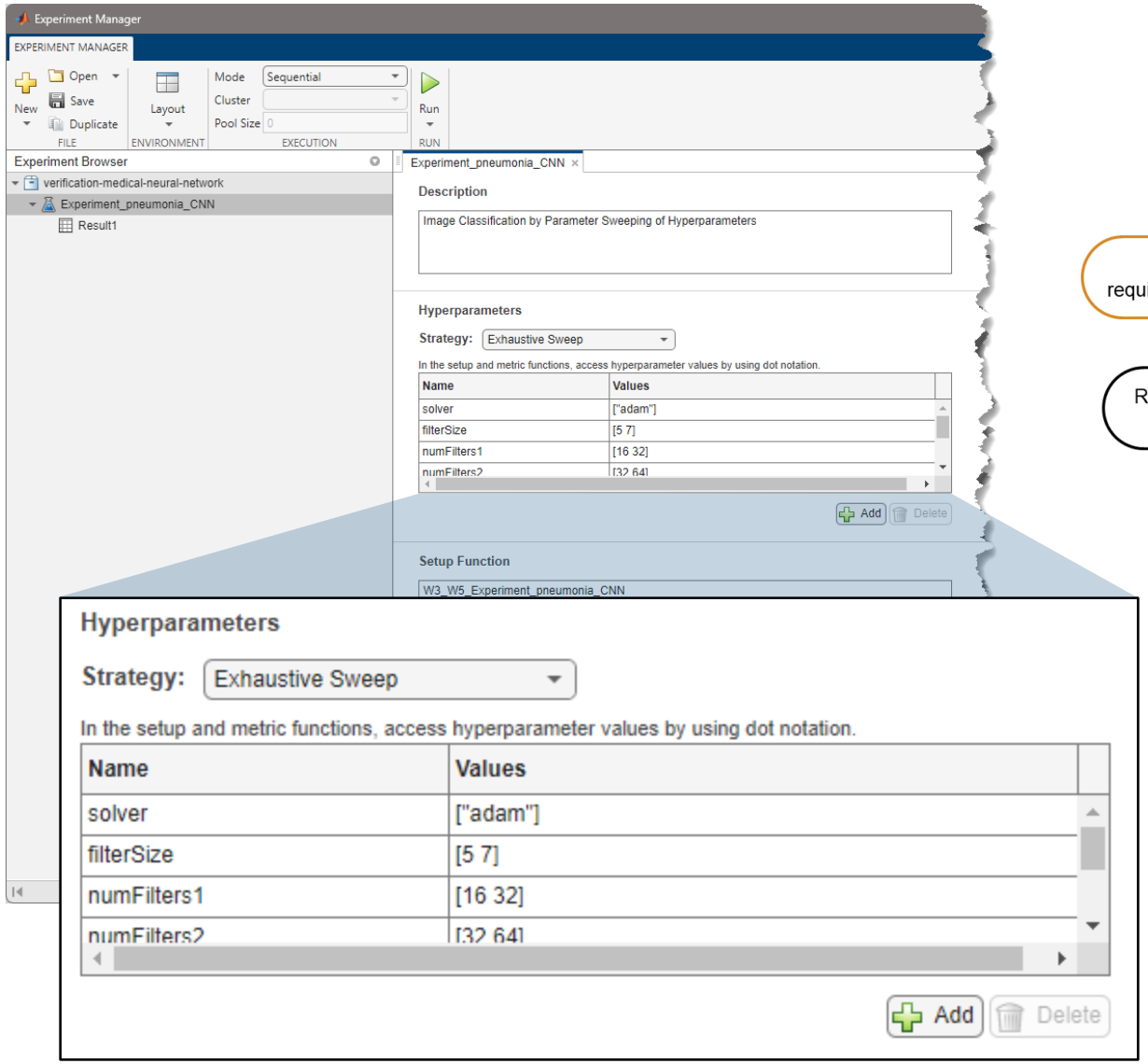


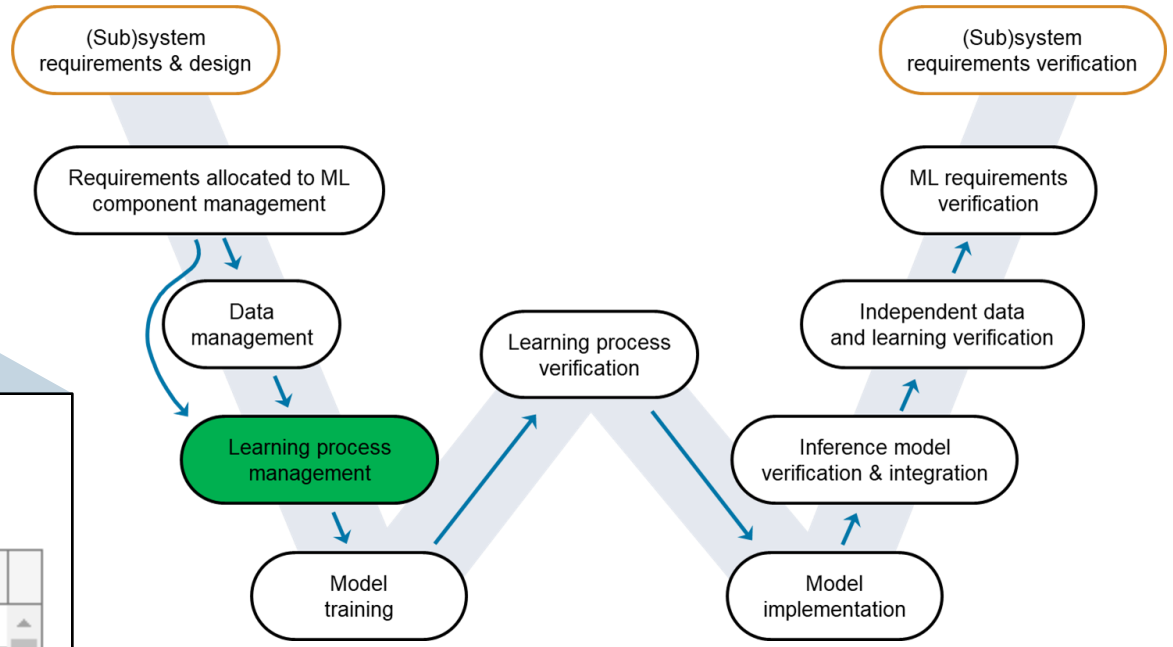

```
numClasses = numel(classNames);
layers = [
    imageInputLayer(imageSize,Normalization="none")
    convolution2dLayer(7,64,Padding=0)
    batchNormalizationLayer()
    reluLayer()
    dropoutLayer(0.5)
    averagePooling2dLayer(2,Stride=2)
    convolution2dLayer(7,128,Padding=0)
    batchNormalizationLayer()
    reluLayer()
    dropoutLayer(0.5)
    averagePooling2dLayer(2,Stride=2)
    fullyConnectedLayer(numClasses)
    softmaxLayer
    classificationLayer(Classes=classNames,ClassWeights=classWeights)];
```

```
options = trainingOptions("adam", ...
    ExecutionEnvironment="auto", ...
    InitialLearnRate=0.001, ...
    MaxEpochs=50, ...
    MiniBatchSize=256, ...
    Shuffle="every-epoch",...
    LearnRateSchedule="piecewise", ...
    LearnRateDropPeriod=30, ...
    LearnRateDropFactor=0.1, ...
    Plots="training-progress", ...
    ValidationData={XVal,TVal}, ...
    ValidationPatience=10, ...
    OutputNetwork="best-validation-loss");
```



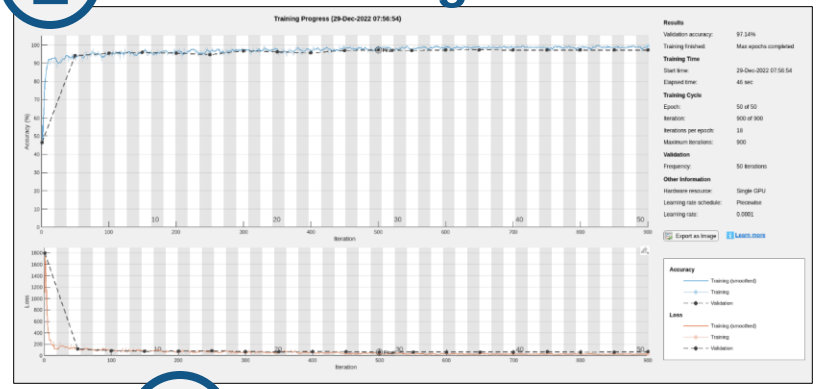
Find optimal paraments and audit experiments for reproducibility





An iterative approach towards building an accurate and robust model

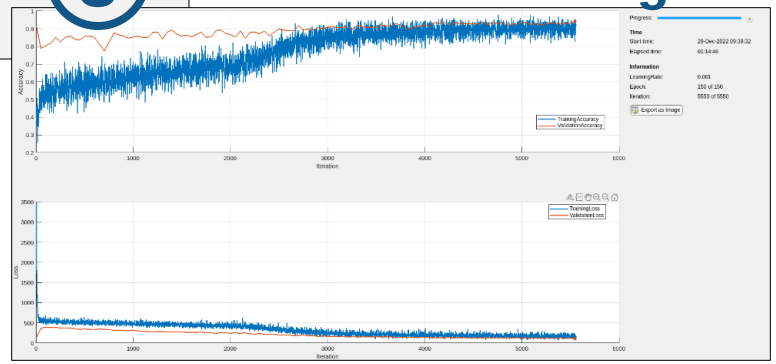
1 Initial training

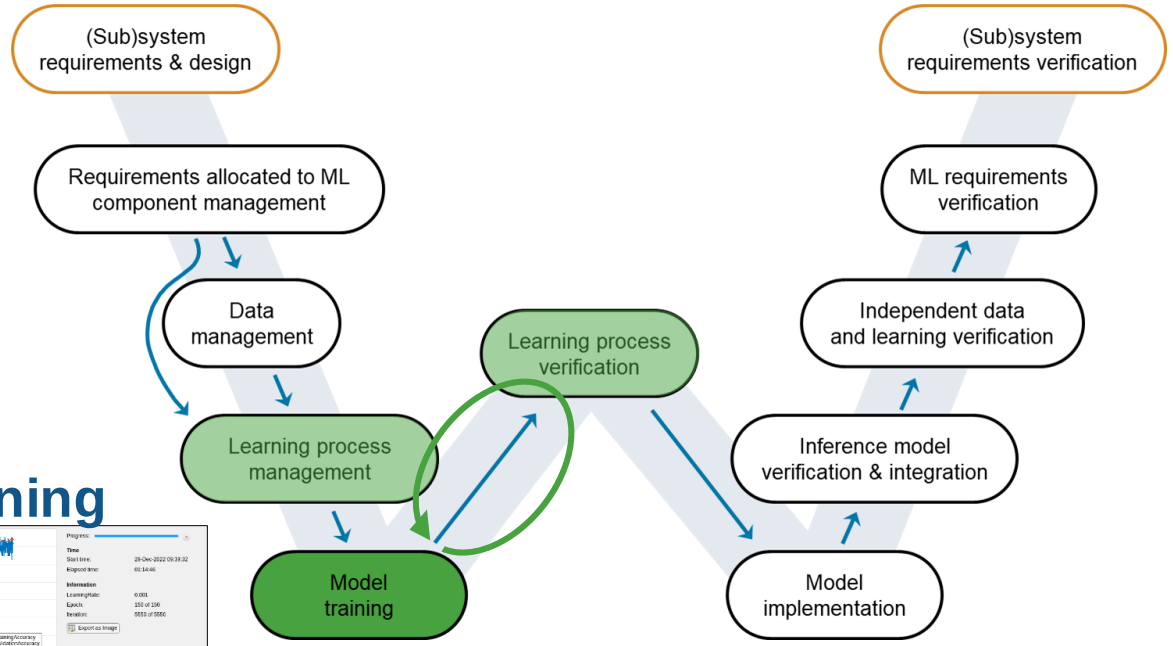


2 Data-augmented training

```
imageAugmenter = imageDataAugmenter(...
    FillValue=mean(XTrain(:)), ...
    RandXReflection=true, ...
    RandXTranslation=[-2,2], ...
    RandYTranslation=[-2,2], ...
    RandRotation=[-10,10],...
    RandScale=[1,1.25], ...
    RandXShear=[-5,5], ...
    RandYShear=[-5,5]);
```

3 Adversarial training





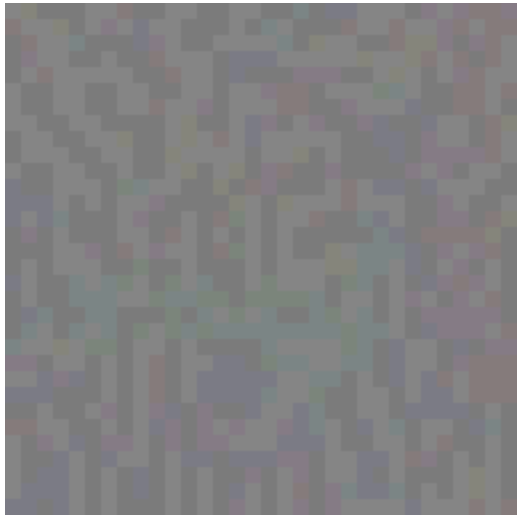
Adversarial Example

cat

Original image

+

δ



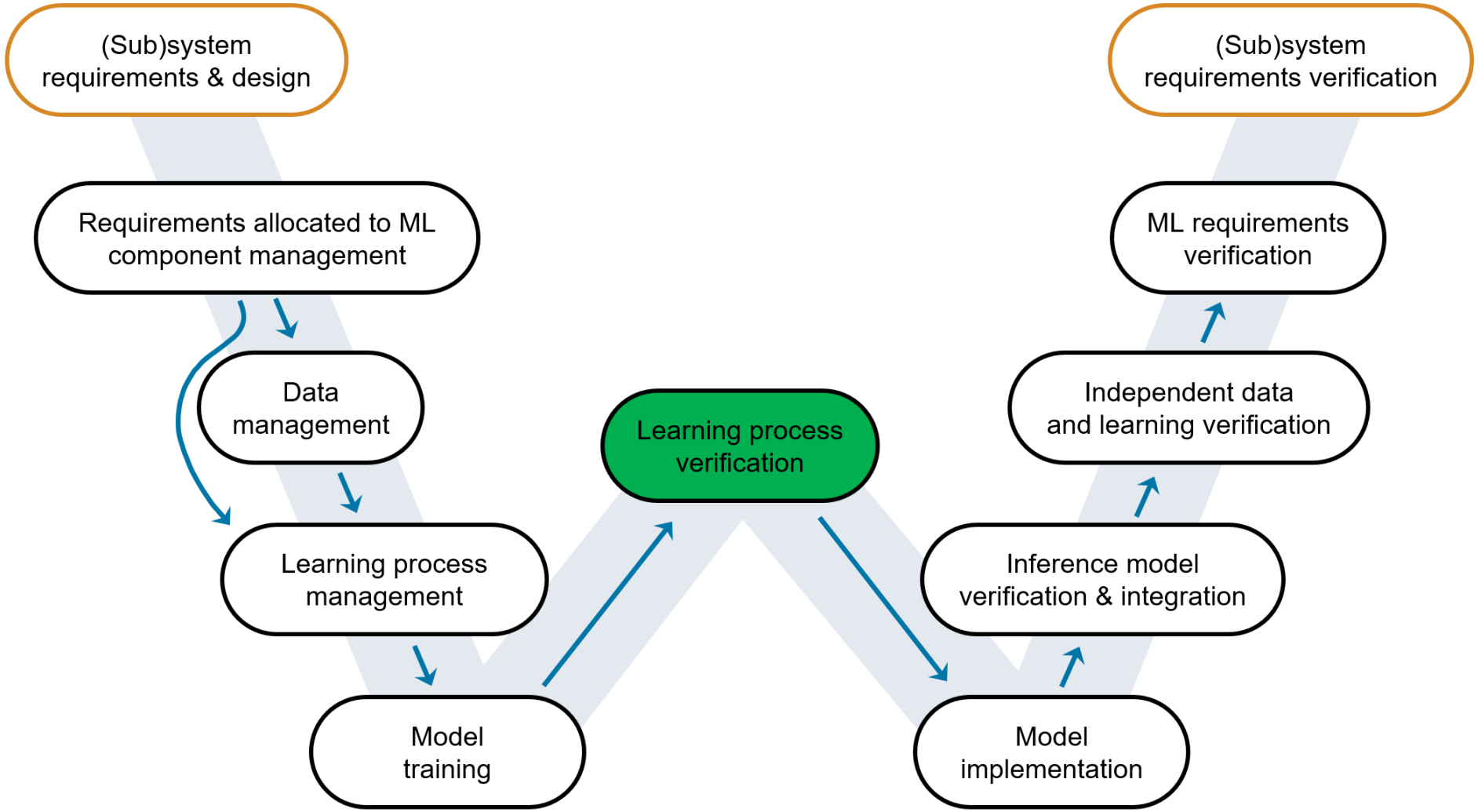
Perturbation

=

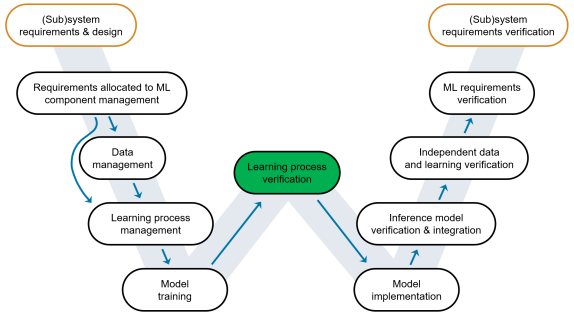
cat or deer

Adversarial image

Learning process verification

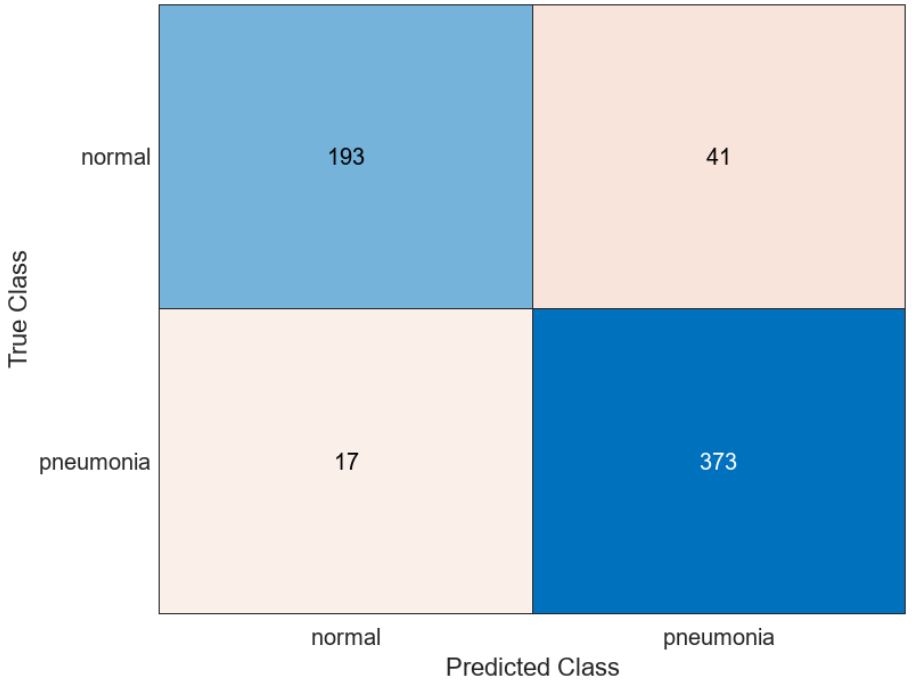


Testing and understanding model performance with an independent test set

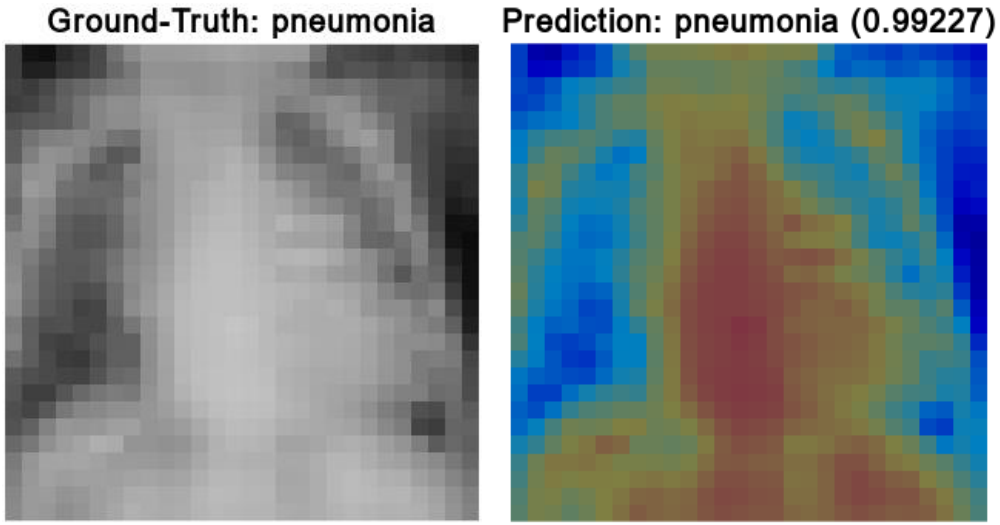


Accuracy: 90.71%

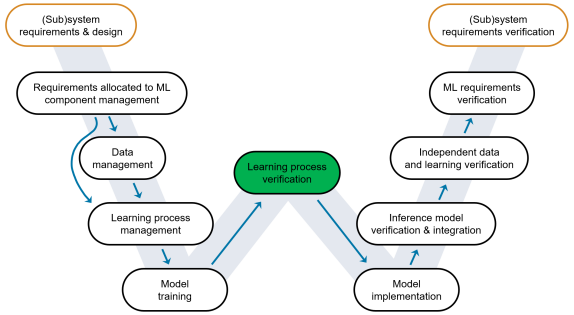
```
confusionchart(T,Y)
```




```
scoreMap = gradCAM(net,X,label)
```



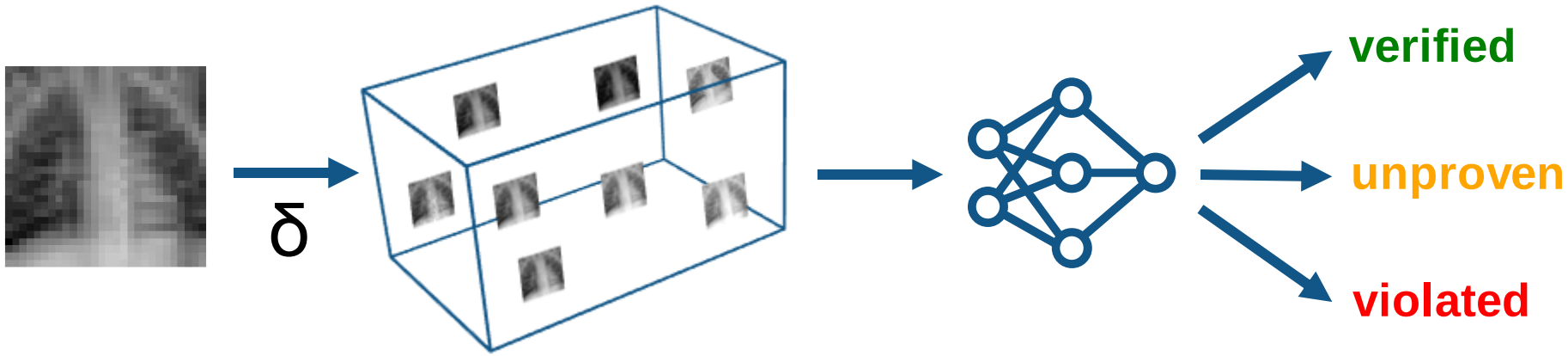
Verify robustness of deep learning networks



Deep Learning Toolbox Verification Library

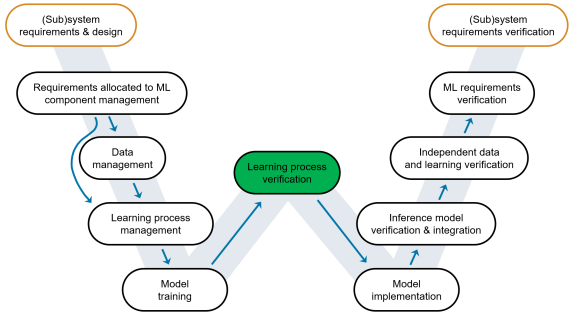
by MathWorks Deep Learning Toolbox Team STAFF

Verify and test robustness of deep learning networks



Formal Verification

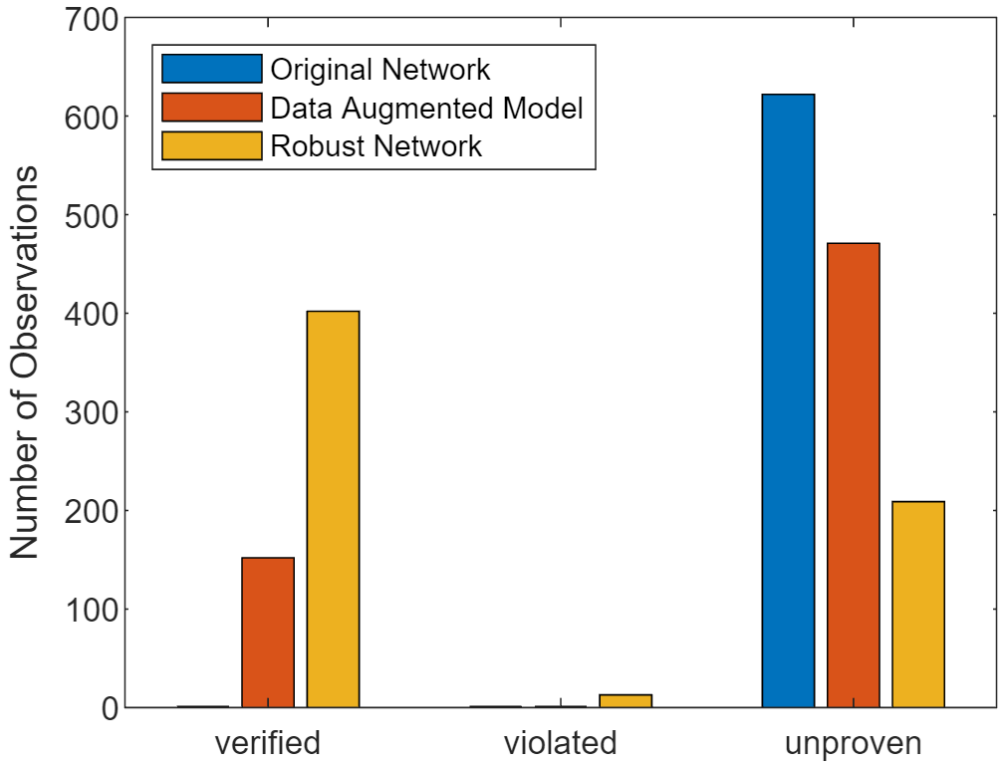
Verify robustness of deep learning networks



Deep Learning Toolbox Verification Library

by MathWorks Deep Learning Toolbox Team STAFF

Verify and test robustness of deep learning networks

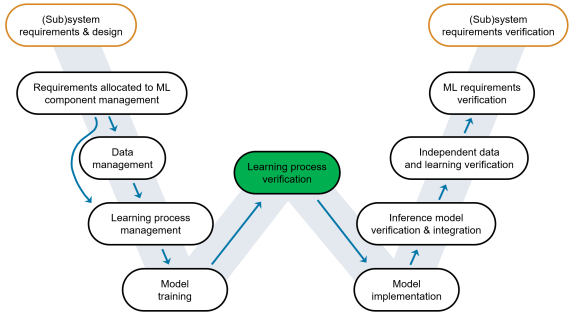



```
perturbation = 0.01;
XLower = XTest - perturbation;
XUpper = XTest + perturbation;
XLower = dlarray(XLower,"SSCB");
XUpper = dlarray(XUpper,"SSCB");
result = verifyNetworkRobustness(net,...
    XLower,XUpper,TTest);
```

```
summary(result)
```

verified	402
violated	13
unproven	209

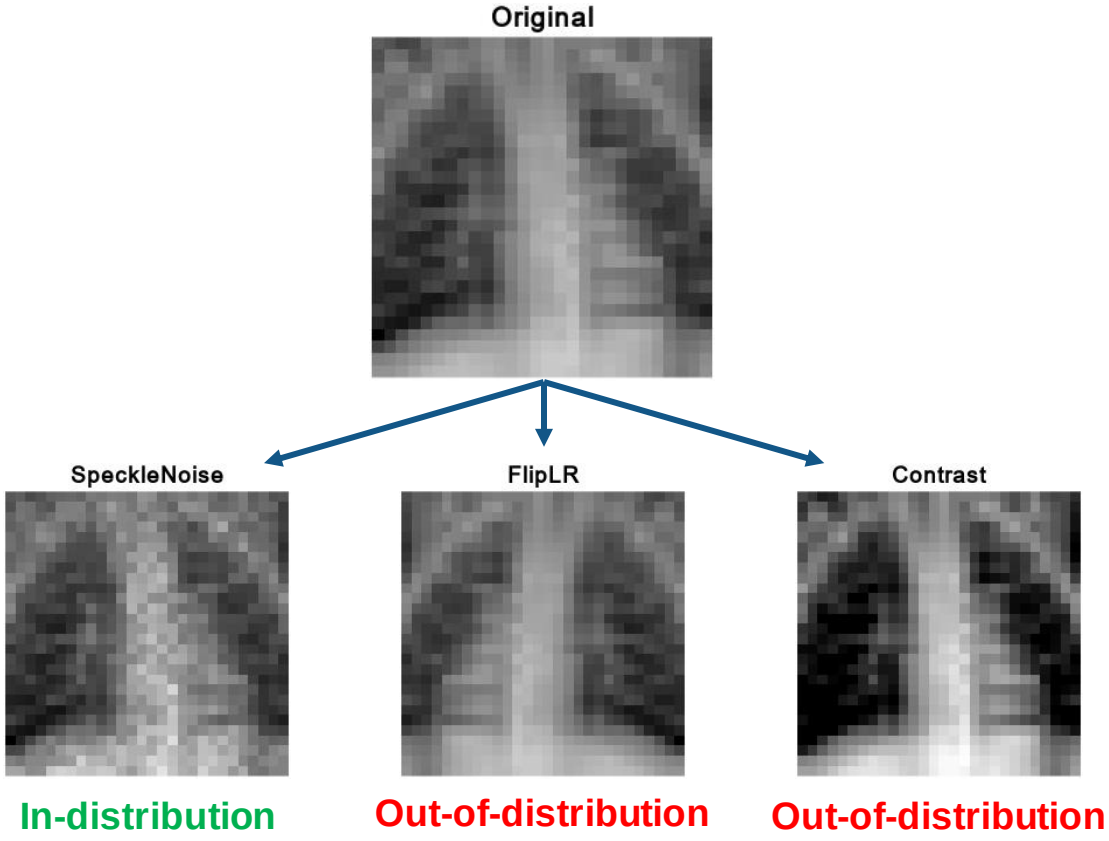
Identify unknown examples to the model and reject or transfer to a human for safe handling

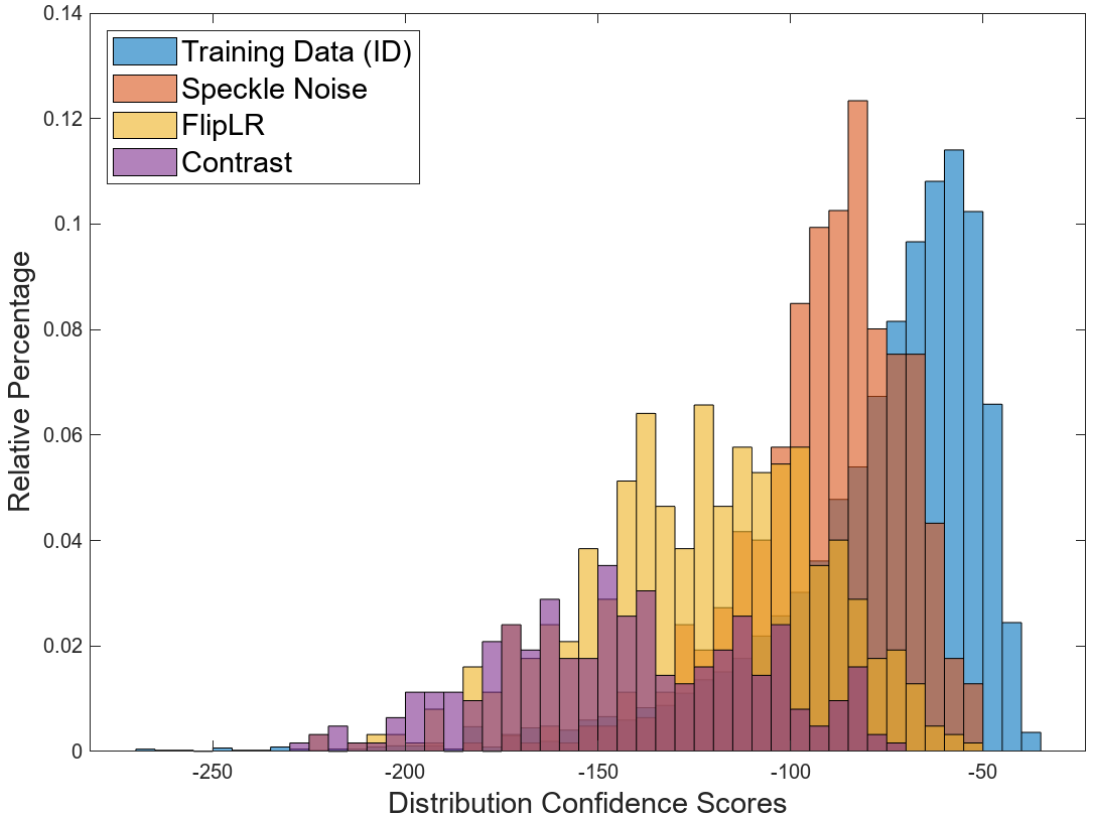


Deep Learning Toolbox Verification Library

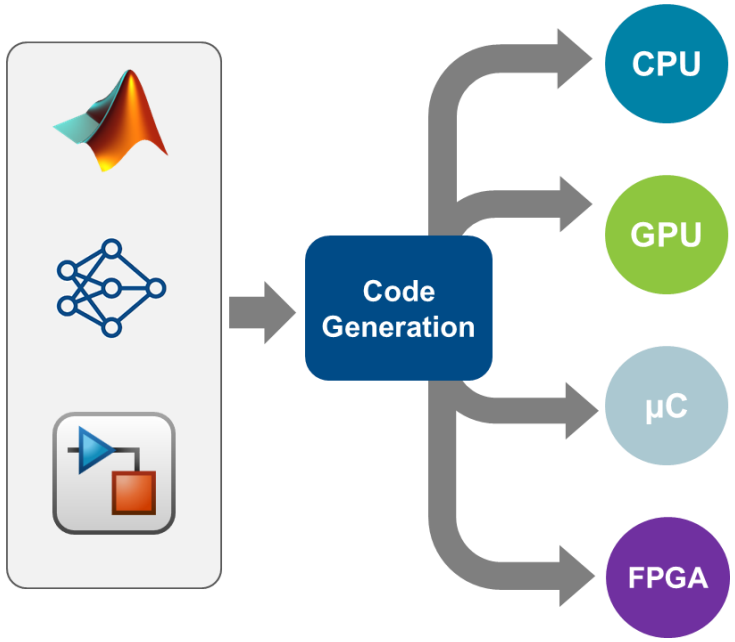
by MathWorks Deep Learning Toolbox Team STAFF

Verify and test robustness of deep learning networks

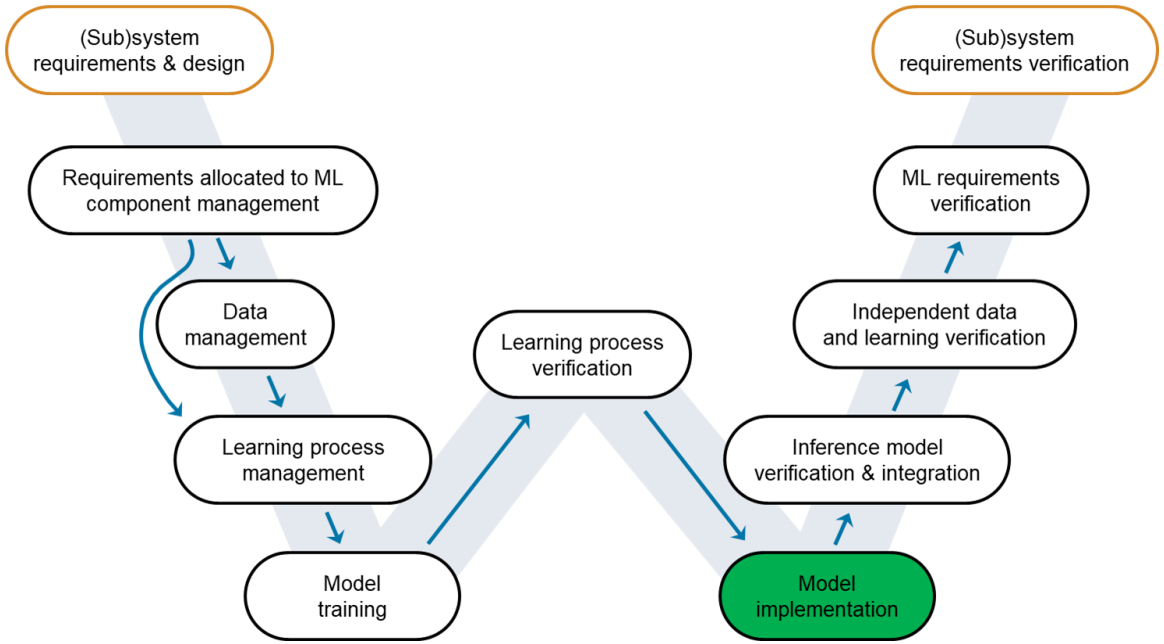




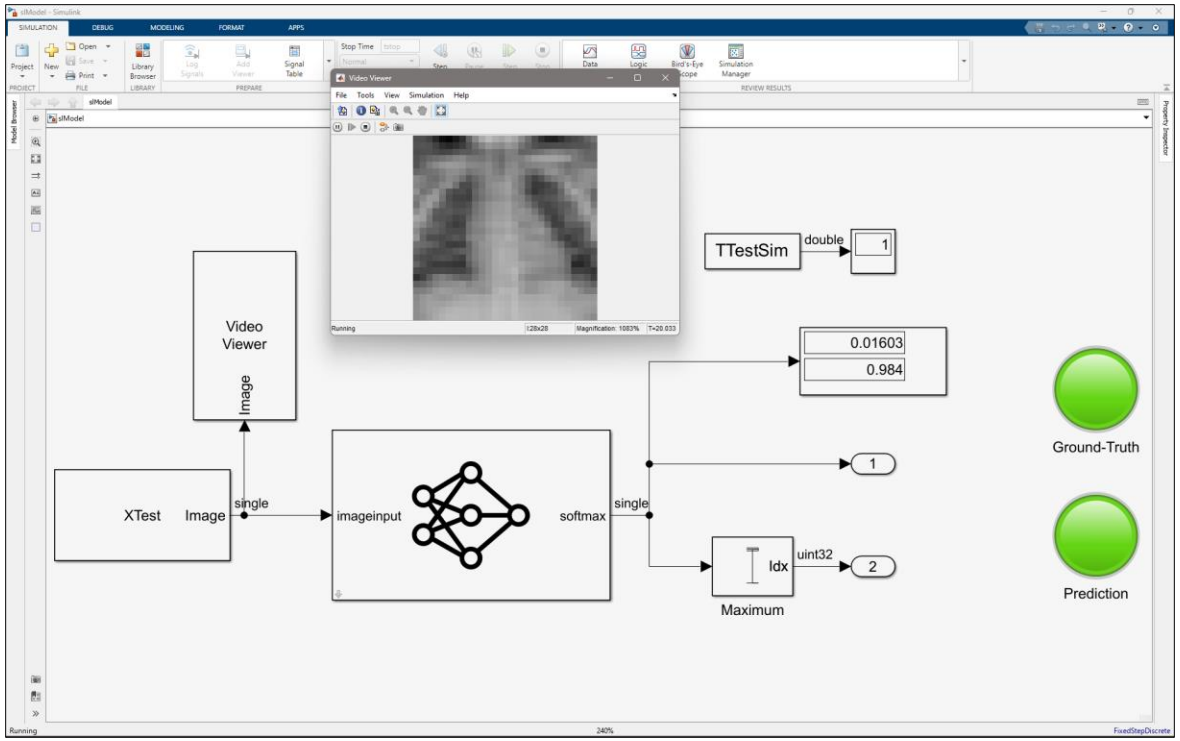
Deploy to target with zero coding errors

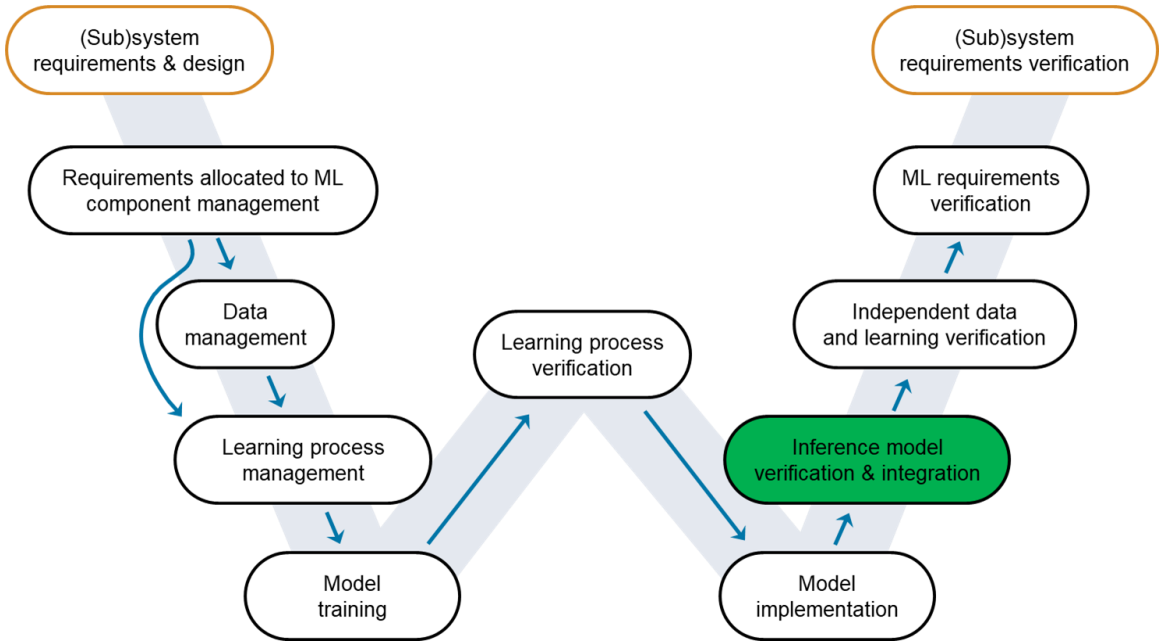



```
analyzeNetworkForCodegen(net)

                Supported
                _________

    none          "Yes"
    arm-compute   "Yes"
    mkldnn        "Yes"
    cudnn         "Yes"
    tensorrt      "Yes"
```

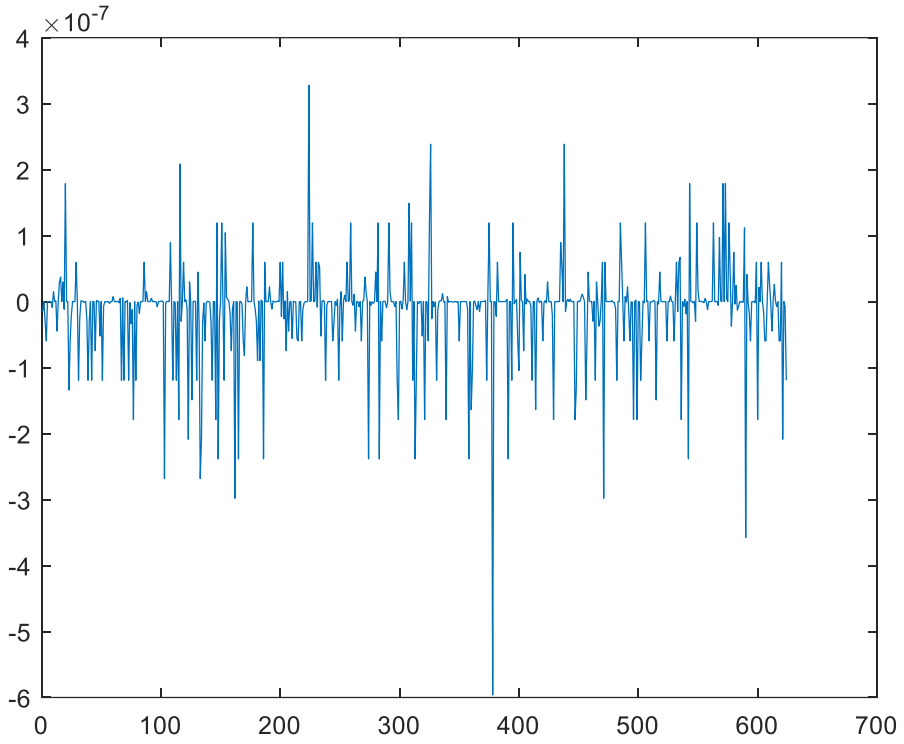


Integrate your AI model in Simulink for system-level simulation and test



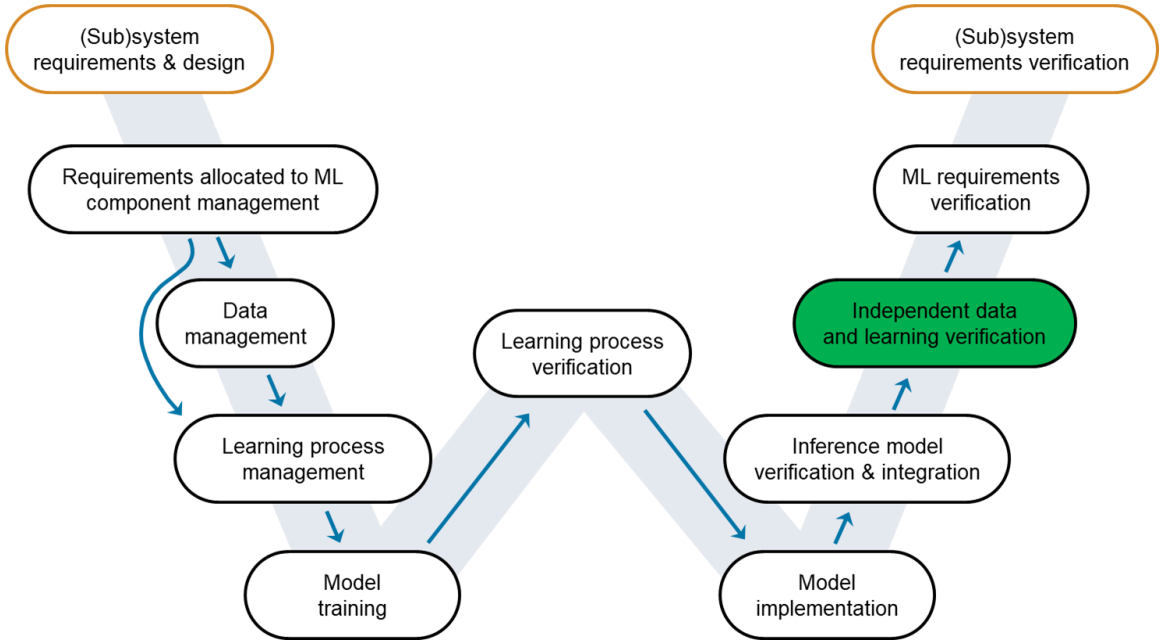


No differences between development and inference models

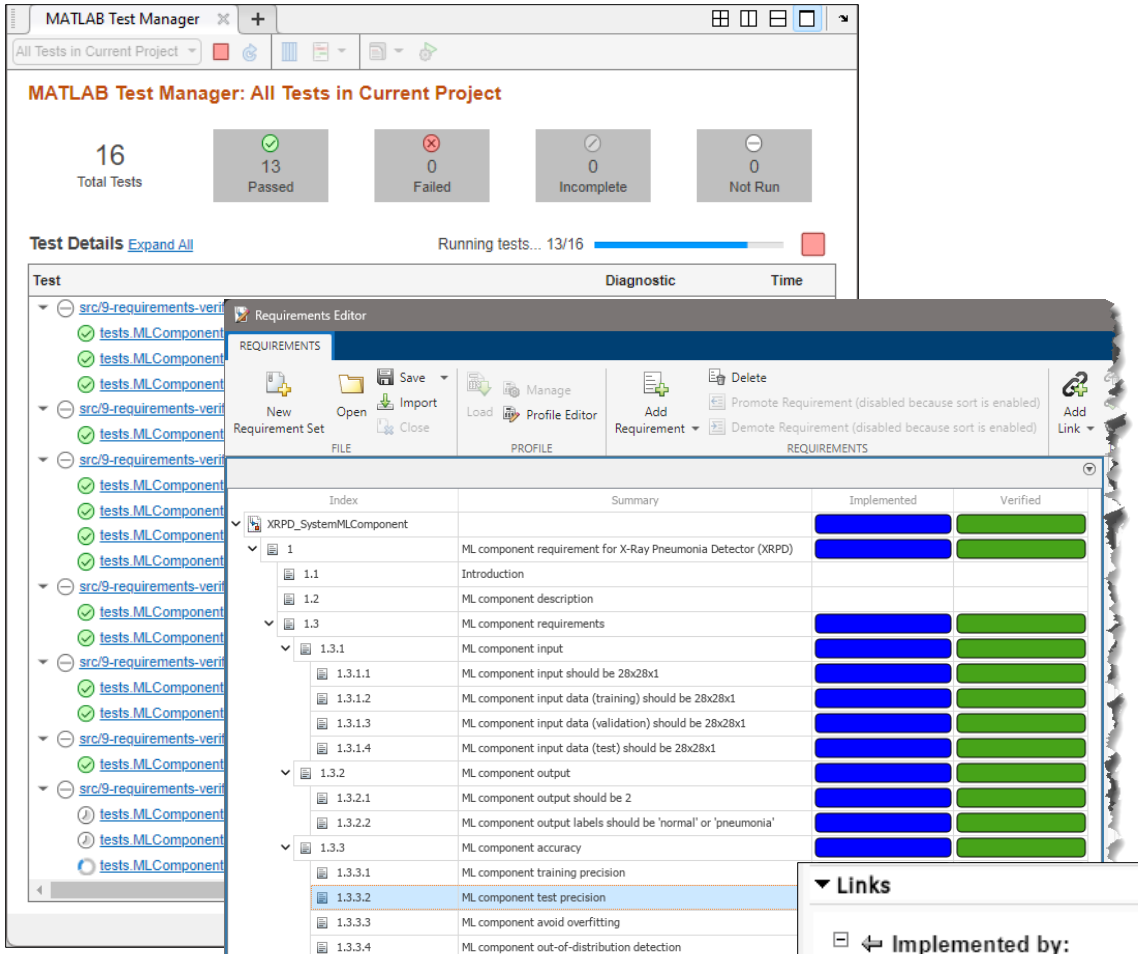


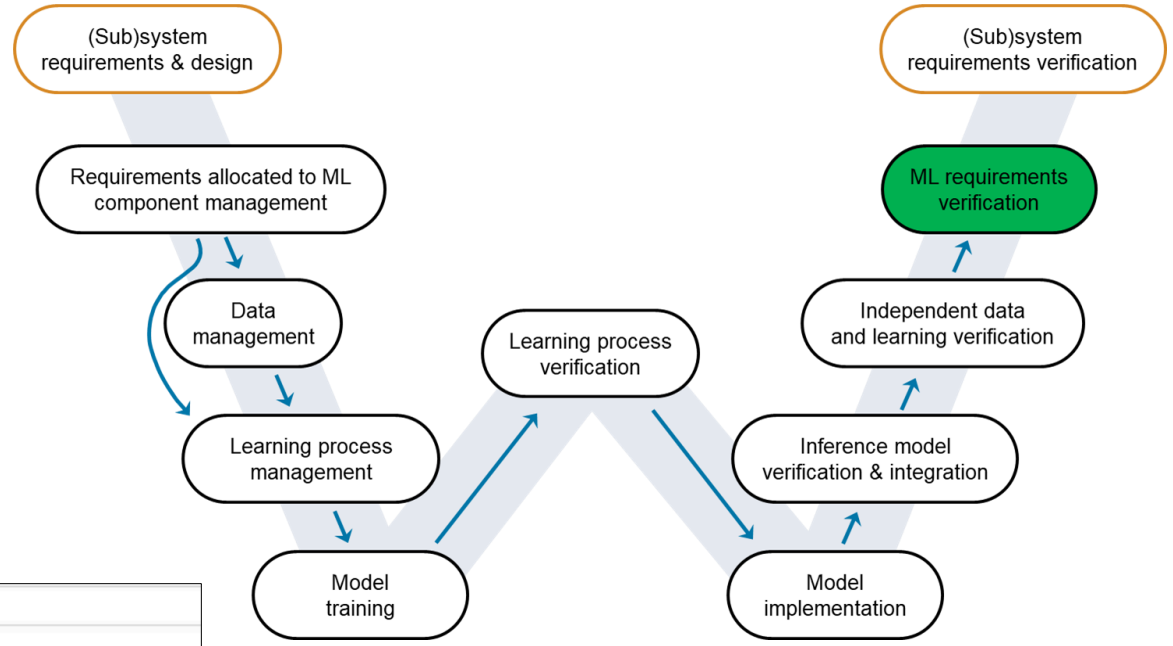

```
max(abs(differences))
```

```
ans = single
5.9605e-07
```



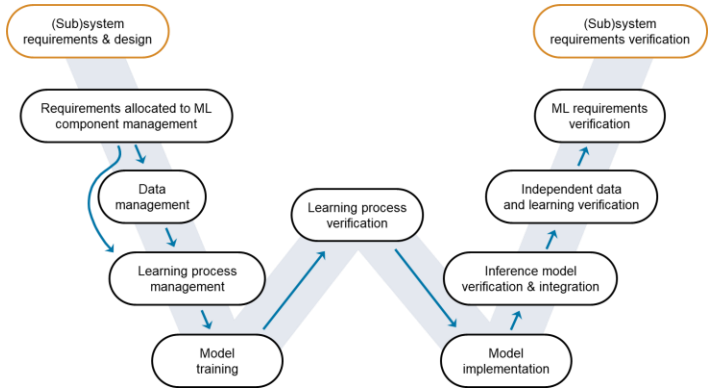
Verifying requirements have been fully tested





Key Takeaways

MathWorks has capabilities addressing each area of the W-diagram



Library to verify and test robustness of deep learning networks

Deep Learning Toolbox Verification Library

by MathWorks Deep Learning Toolbox Team STAFF

Verify and test robustness of deep learning networks

Our safety-critical certification expertise helps drive new AI standards

EUROCAE WG-114 / SAE G-34 Standardization Working Group "Artificial Intelligence in Aviation"

MATLAB EXPO

Thank you