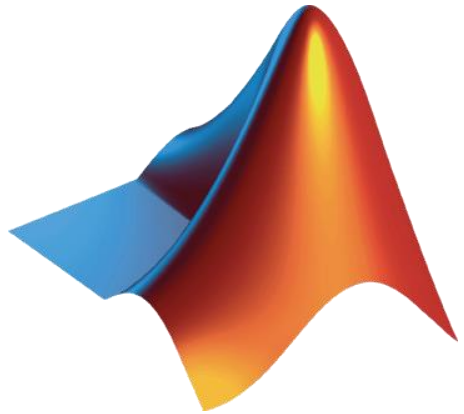
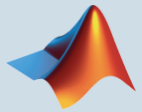
Deep Learning for Signal Data with MATLAB



Paul Huxel, Ph.D.
Senior Application Engineer
MathWorks Ireland

Analog Devices
12 Sept 2019

Agenda



Demonstration and Overview

- Signal Analysis Challenges and Techniques
- Introduction to Deep Learning
- Workflow and Solution in MATLAB
- Summary and Support

Demo: Deep Learning Image Classification using MATLAB

- Using Alexnet to Classify an Image (1,000 Classes)
  - `url = "https://www.nasa.gov/sites/default/files/styles/" + ...`
    `"full_width_feature/public/shuttle_sts120.jpg";`
  - `im = imread(url);`
  - `imshow(im)`

  - **`im = imresize(im,[227 227]);`**
  - **`net = alexnet;`**
  - **`label = classify(net,im)`**

`label =`

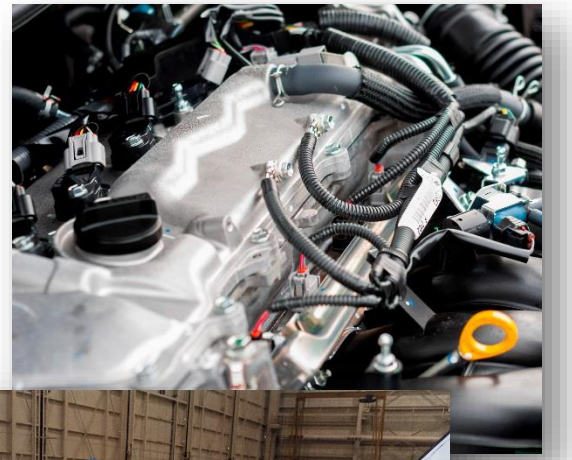
`categorical`

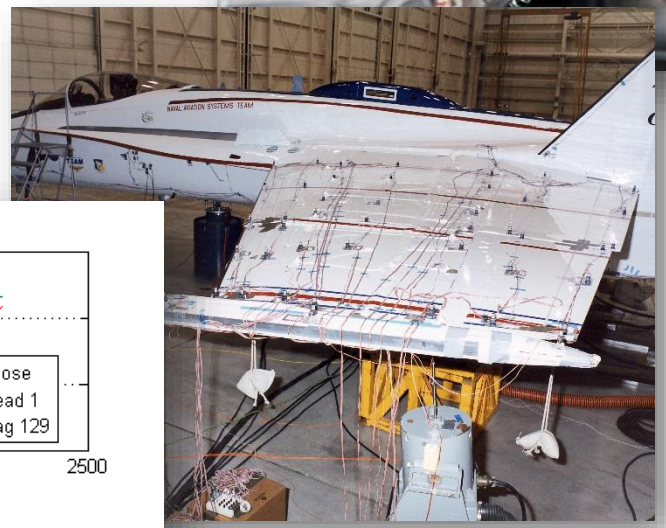
`space shuttle`

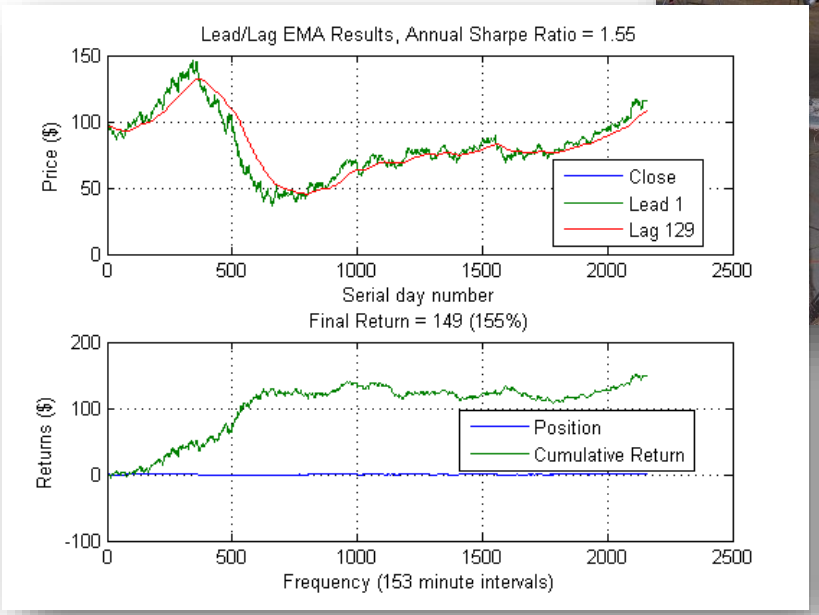
- Webcam Example
  - `camera = webcam;`
  - `im = camera.snapshot;`

Signals are Everywhere

- Structural health monitoring (SHM)
- Engine event detection
- Speech Signal Classification
- Advanced surveillance
- Healthcare Applications
- ...

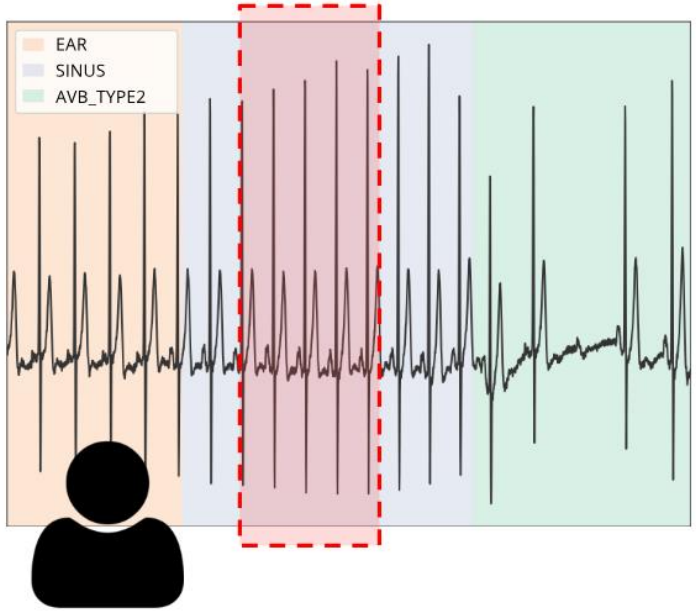




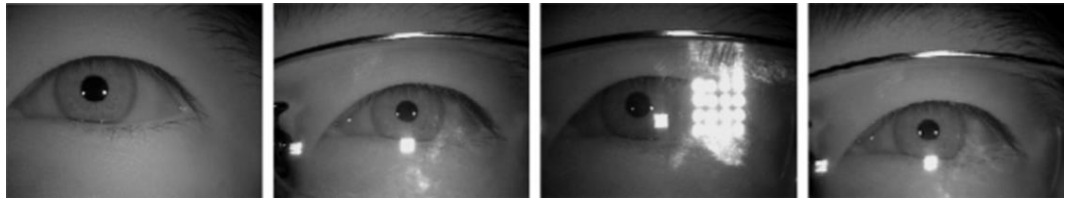


Diverse Applications of Deep Learning

Detection of cars and road in autonomous driving systems



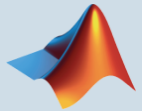
Cardiologist-level arrhythmia detection[1]



Iris Recognition – 99.4% accuracy[2]

1. *"Cardiologist-Level Arrhythmia Detection with Convolutional Neural Networks" Rajpurkar et al*
2. *Source: An experimental study of deep convolutional features for iris recognition Signal Processing in Medicine and Biology Symposium (SPMB), 2016 IEEE Shervin Minaee ; Amirali Abdolrashidiy ; Yao Wang*

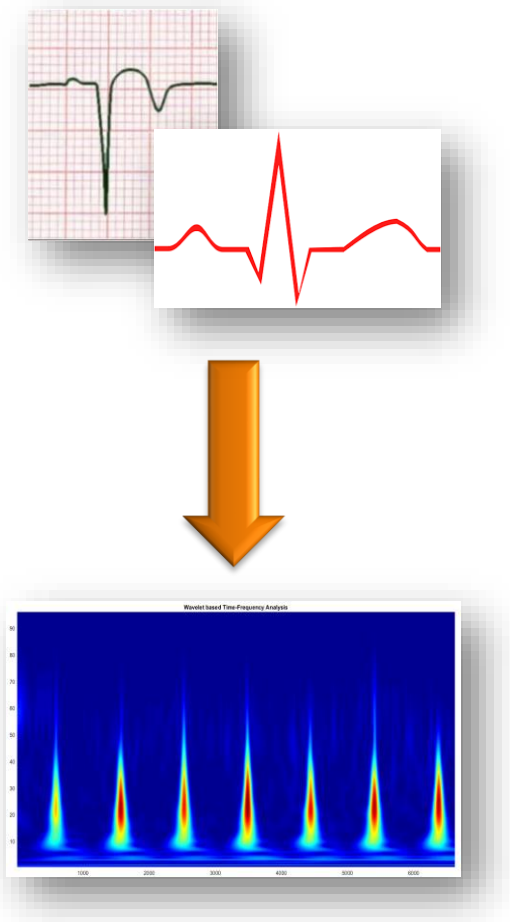
Agenda

- Demonstration and Overview
-  Signal Analysis Challenges and Techniques
- Introduction to Deep Learning
- Workflow and Solution in MATLAB
- Summary and Support

Challenges with Signals

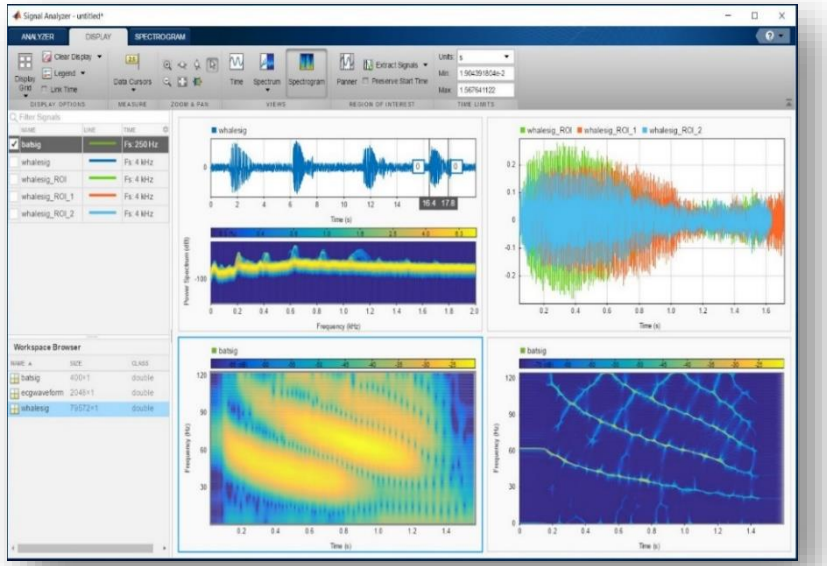
- *Normal (Class I)*
- *Atrial Fibrillation (Class II)*
- *Congestive Heart Failure (Class III)*

- Enhancing subtle information present in signals
  - Signals belonging to different classes can have similar properties
- Represent features occurring at different scales
  - Good time-frequency localization
- Need for independent representation of signals
  - Signal features within same class can have different amplitudes or polarities etc.



Using Time-Frequency Representations

- A time-frequency representation captures how spectral content of signal evolves over time (can be saved as an image)



- Many time-frequency representations available
  - **Spectrogram**
  - mel-Frequency Spectrograms
  - **Scalogram (Continuous Wavelet Transform)**
  - Constant Q Transform etc.

- You can generate wavelet based time-frequency representations of your signals with **two** lines of MATLAB code

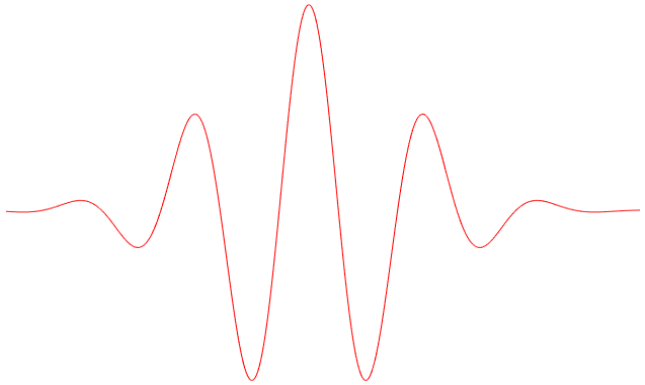
What is a wavelet?

- A wavelet is a rapidly decaying wave like oscillation with zero mean

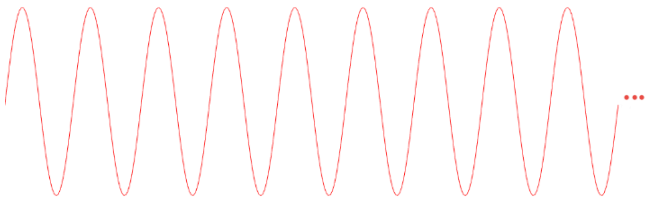
- Wavelets are best suited to localize frequency content in real world signals

- MATLAB makes it easy by providing default wavelets

Wavelet

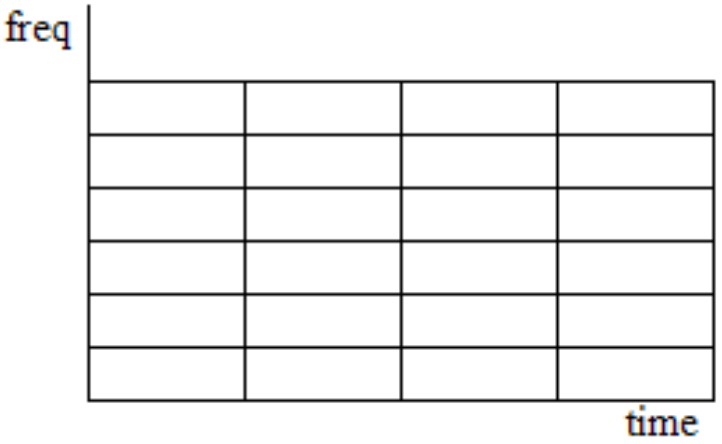


Sine/Cosine Wave



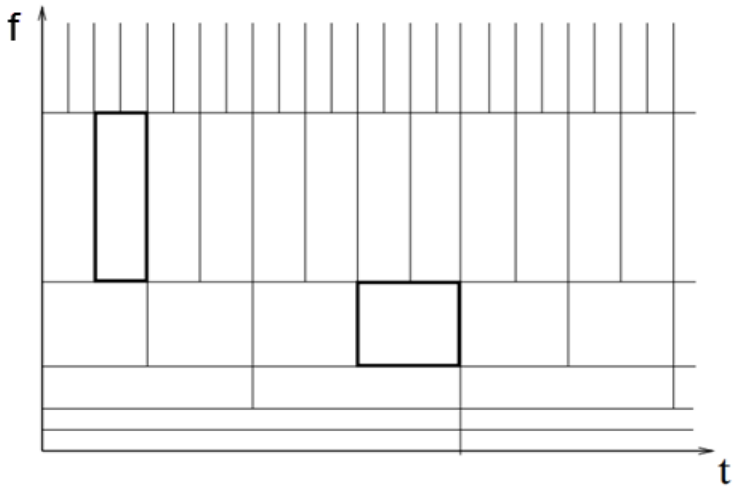
Time-Frequency Analysis - Comparison

Short Time Fourier Transform



- **Sine/Cosine Waves** – oscillate forever
- **Fixed window** size limits the resolution

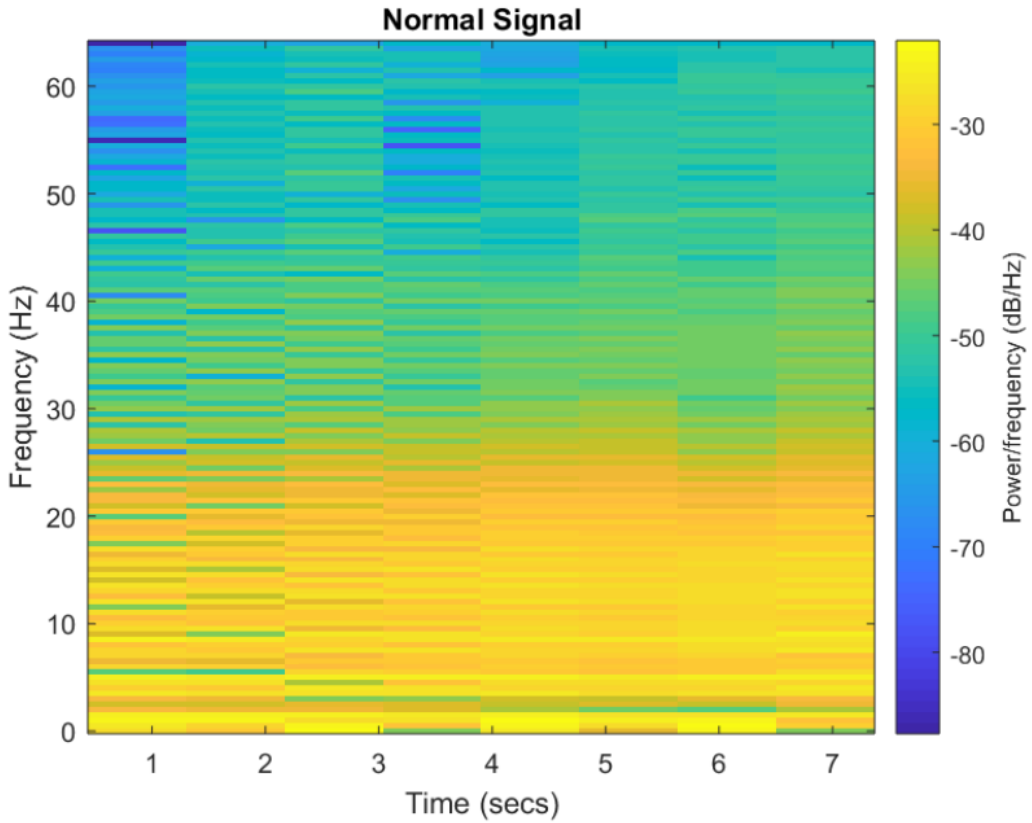
Continuous Wavelet Transform



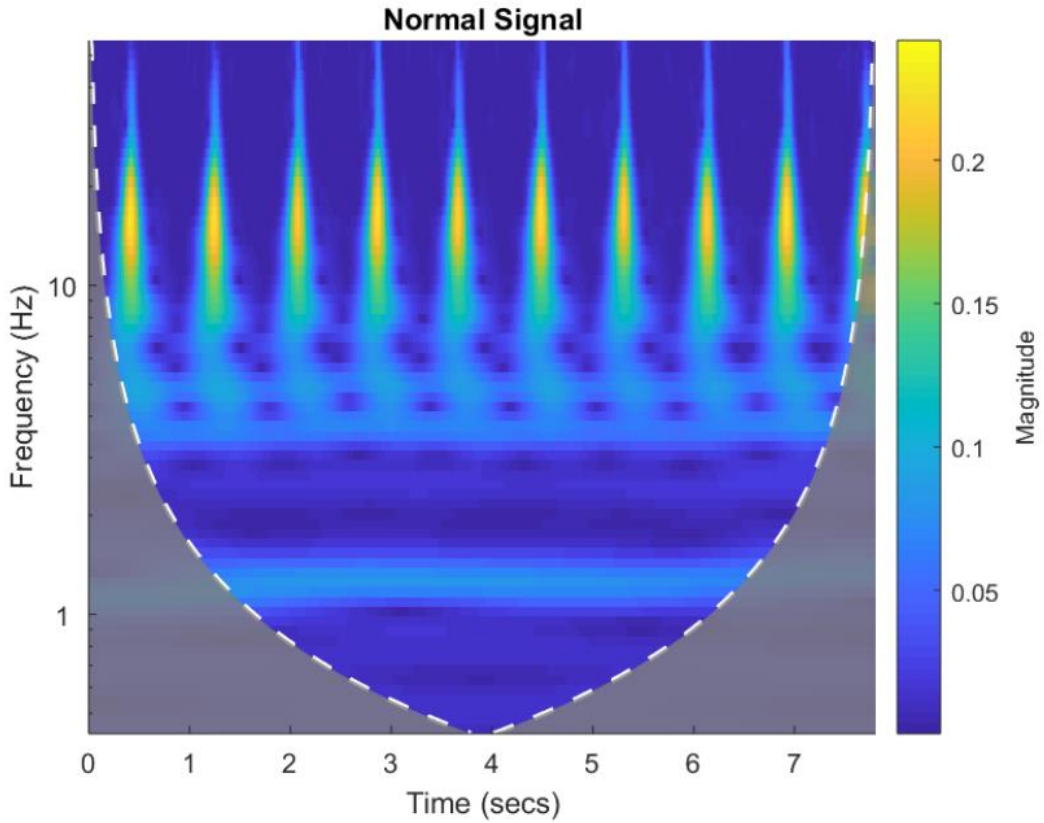
- **Wavelets** – well localized in time and frequency
- **Variable sized** windows capture features at different scales simultaneously

Time-Frequency Analysis - Comparison

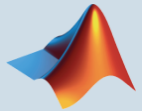
Traditional Spectrogram



Scalogram (CWT)

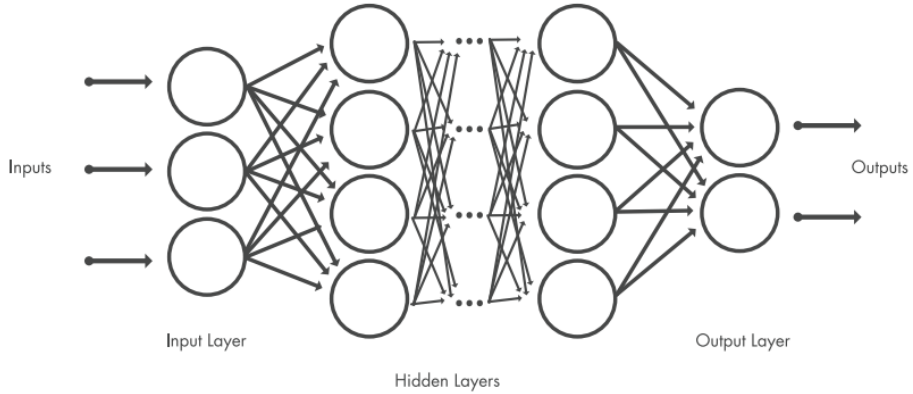


Agenda

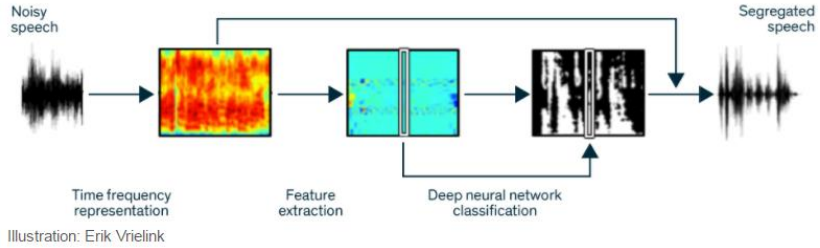
- Demonstration and Overview
- Signal Analysis Challenges and Techniques
-  Introduction to Deep Learning
- Workflow and Solution in MATLAB
- Summary and Support

What is Deep Learning?

- Type of machine learning in which a model learns to extract features and perform classification automatically
- Usually implemented using a neural network architecture
- We will focus on using **pre-trained** transfer learning models for signal classification

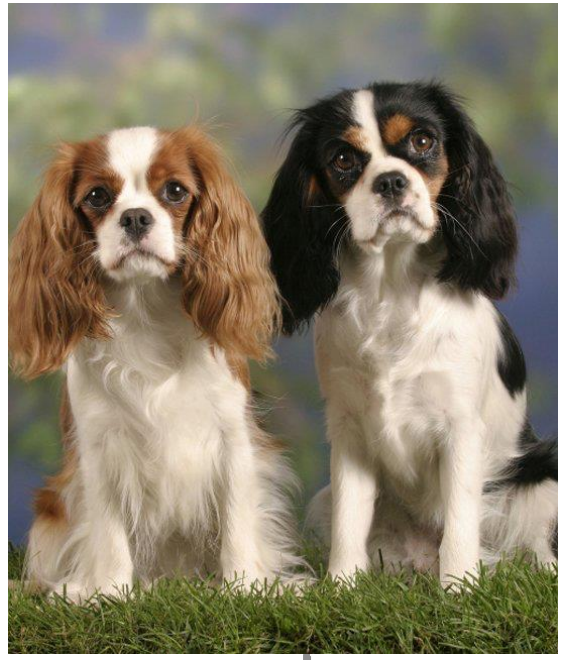


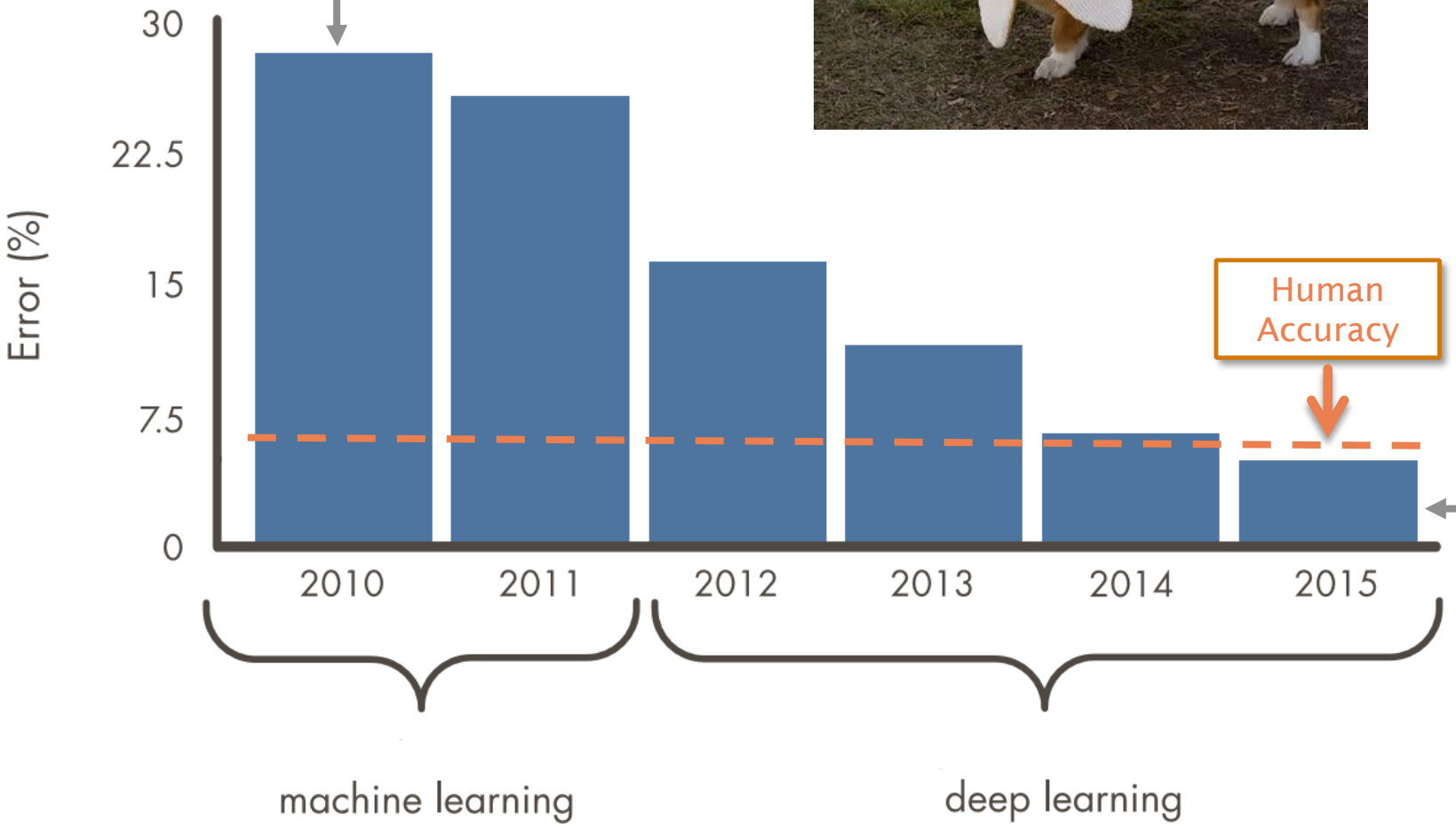
AlexNet
PRETRAINED MODEL



Why now?

ImageNet Challenge

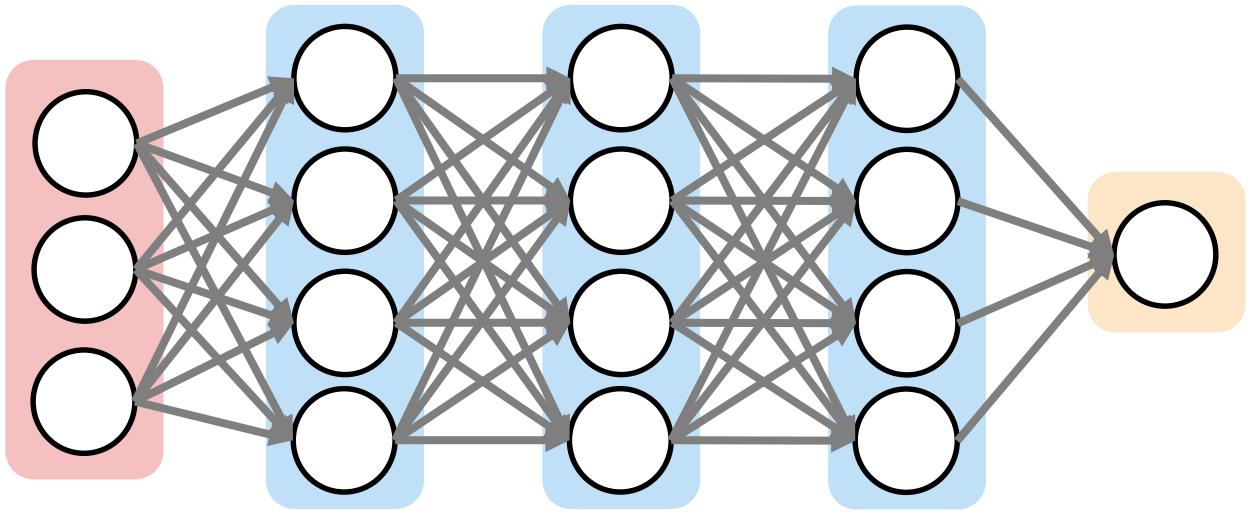




Source: ILSVRC Top-5 Error on ImageNet

How does deep learning work?

Network Architecture



- Pass data in
- Perform operations
- Get output prediction

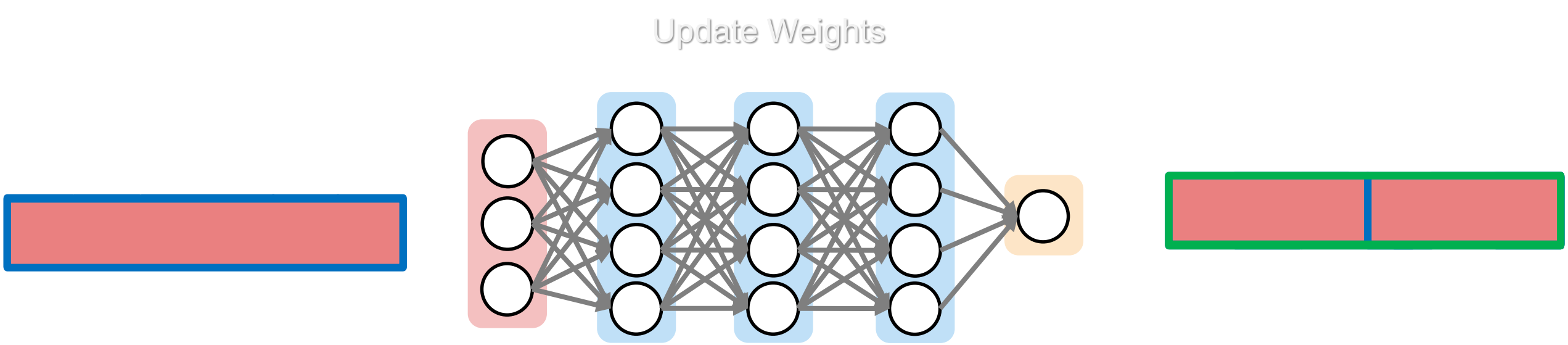
- ***Training***: the process of tuning the weights in the hidden layers so that the predicted output matches the expected output

- ***Inference***: pass new input through the network, obtain prediction

How does deep learning work?

Jargon

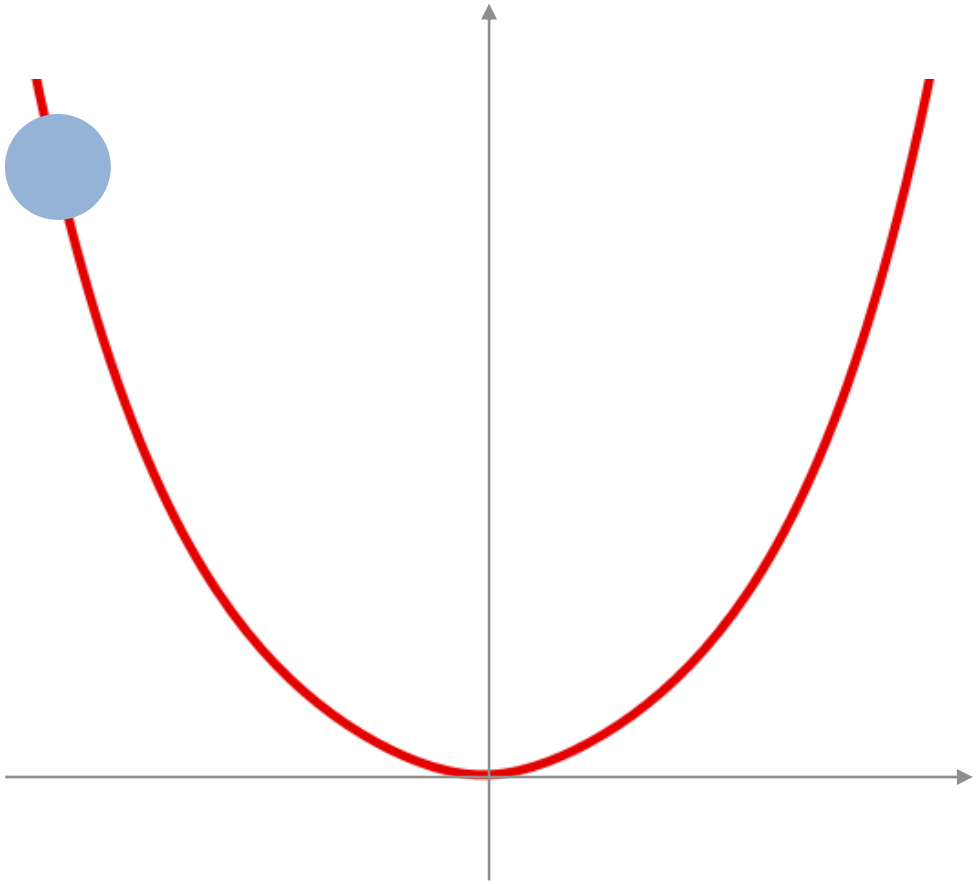
- **Iteration** – one pass of a subset of the training data – or **mini-batch** – through the network
- **Epoch** – a number of **iterations** such that the network has seen all the data once



How does deep learning work?

Jargon

- **Learn-rate** – amount the network can learn during one **iteration**



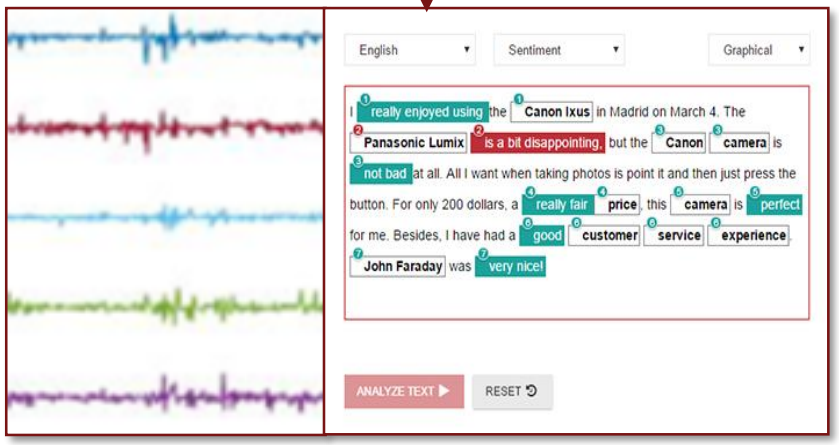
Types of Datasets

Numeric Data

ID	WC_TA	RE_TA	EBIT_TA	MVE_BVTD	S_TA	Industry	Rating
62394	0.013	0.104	0.036	0.447	0.142	3	BB
48608	0.232	0.335	0.062	1.969	0.281	8	A
42444	0.311	0.367	0.074	1.935	0.366	1	A
48631	0.194	0.263	0.062	1.017	0.228	4	BBB
43768	0.121	0.413	0.057	3.647	0.466	12	AAA
39255	-0.117	-0.799	0.01	0.179	0.082	4	CCC
62236	0.087	0.158	0.049	0.816	0.324	2	BBB
39354	0.005	0.181	0.034	2.597	0.388	7	AA
40326	0.47	0.752	0.07	11.596	1.12	8	AAA
51681	0.11	0.337	0.045	3.835	0.812	4	AAA

Machine Learning or LSTM

Time Series Signal Text



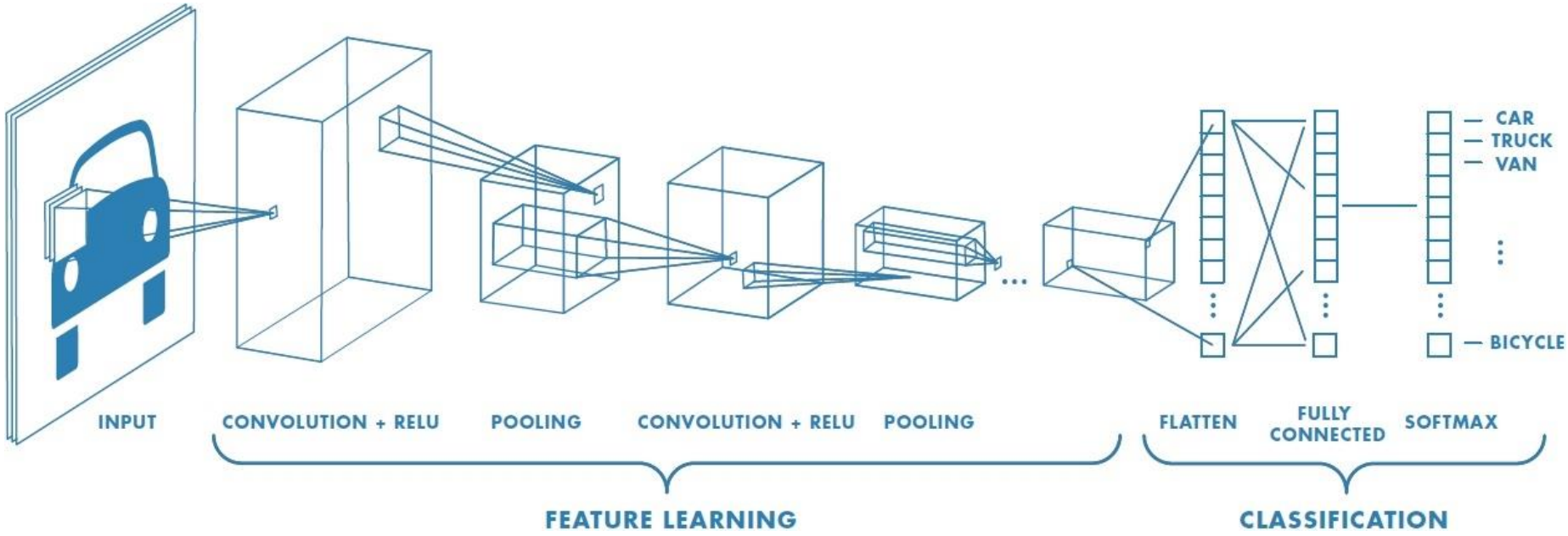
LSTM or CNN

Image Data

CNN

LSTM = Long Short-Term Series Network , CNN = Convolutional Neural Network

Convolutional Neural Networks

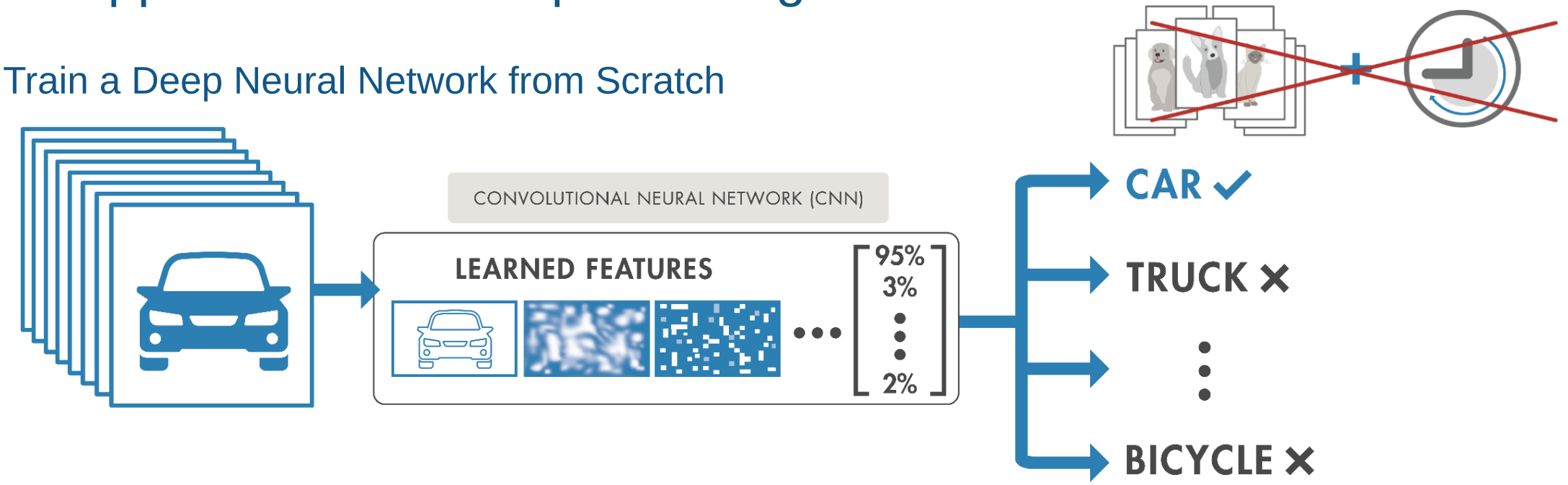


- Different filters pick out different types of local features
- Successive layers learn increasingly complex features
- Objects don’t always have to appear in the same place (translational invariance)

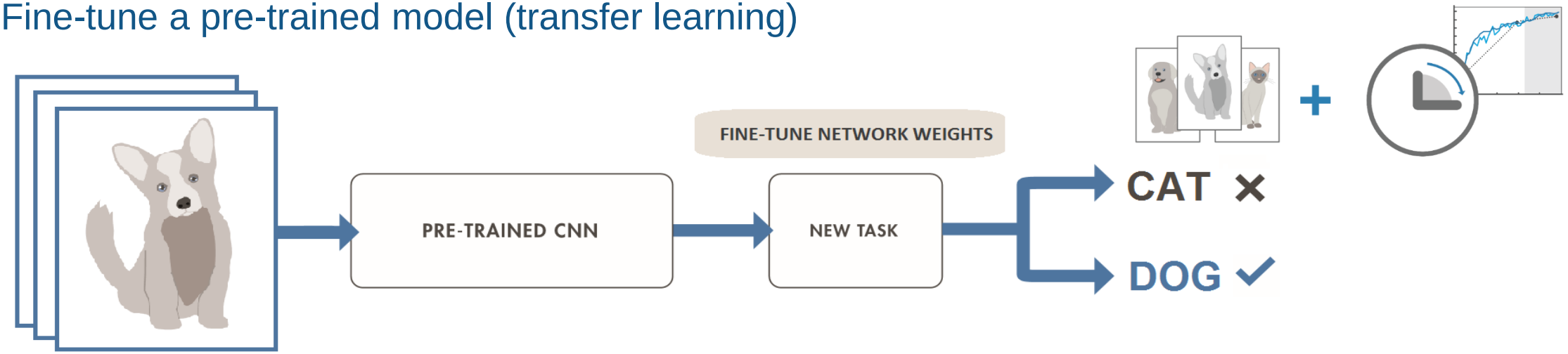
- Learn to classify images based on the features picked out in the first part of the network

Two Approaches for Deep Learning

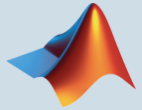
1. Train a Deep Neural Network from Scratch



2. Fine-tune a pre-trained model (transfer learning)



Agenda

- Demonstration and Overview
- Signal Analysis Challenges and Techniques
- Introduction to Deep Learning
-  Workflow and Solution in MATLAB
- Summary and Support

Transfer Learning Workflow

Load pretrained network

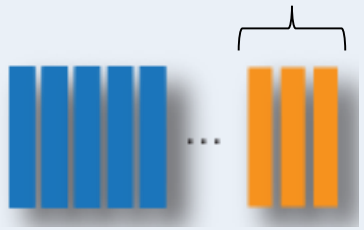
Early layers that learned low-level features (edges, blobs, colors)

Last layers that learned task specific features

1 million images
1000s classes

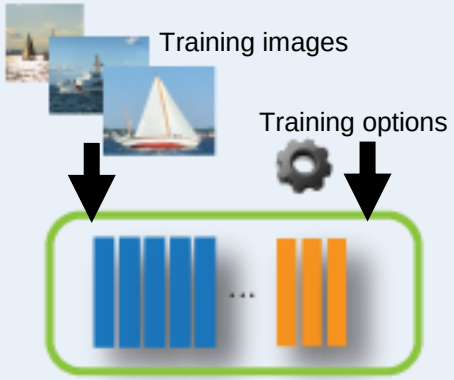
Replace final layers

New layers to learn features specific to your data



Fewer classes
Learn faster

Train network

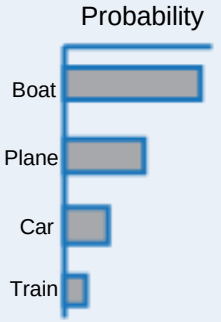


100s images
10s classes

Predict and assess network accuracy

Trained Network

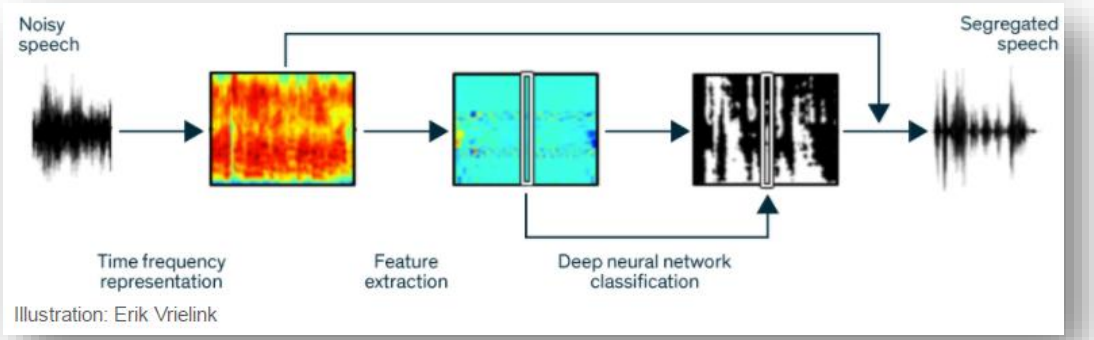
Deploy results



Demo: Deep Learning on Signals

- **Goal:** Given a set of labeled signals, quickly build a classifier

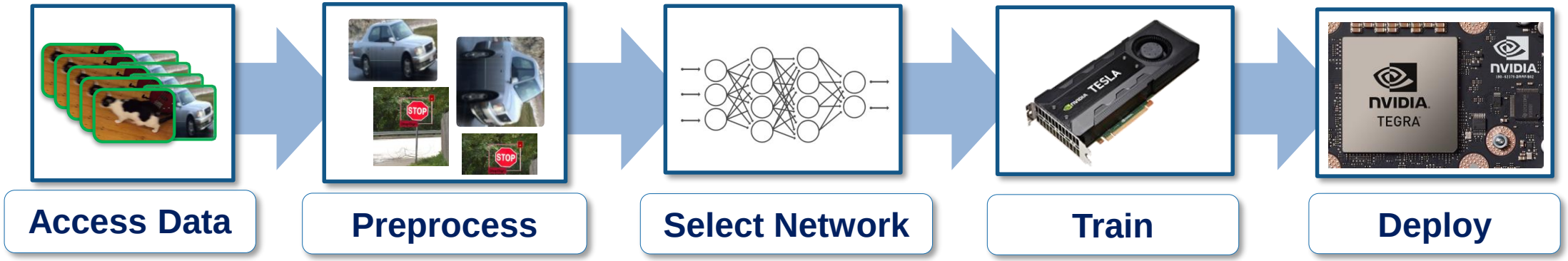
- **Dataset:** 160 signal records with ~65K samples each
  - Normal (Class I)
  - Atrial Fibrillation (Class II)
  - Congestive Heart Failure (Class III)



- **Approach:** Pre-trained Transfer Learning Models
  - AlexNet

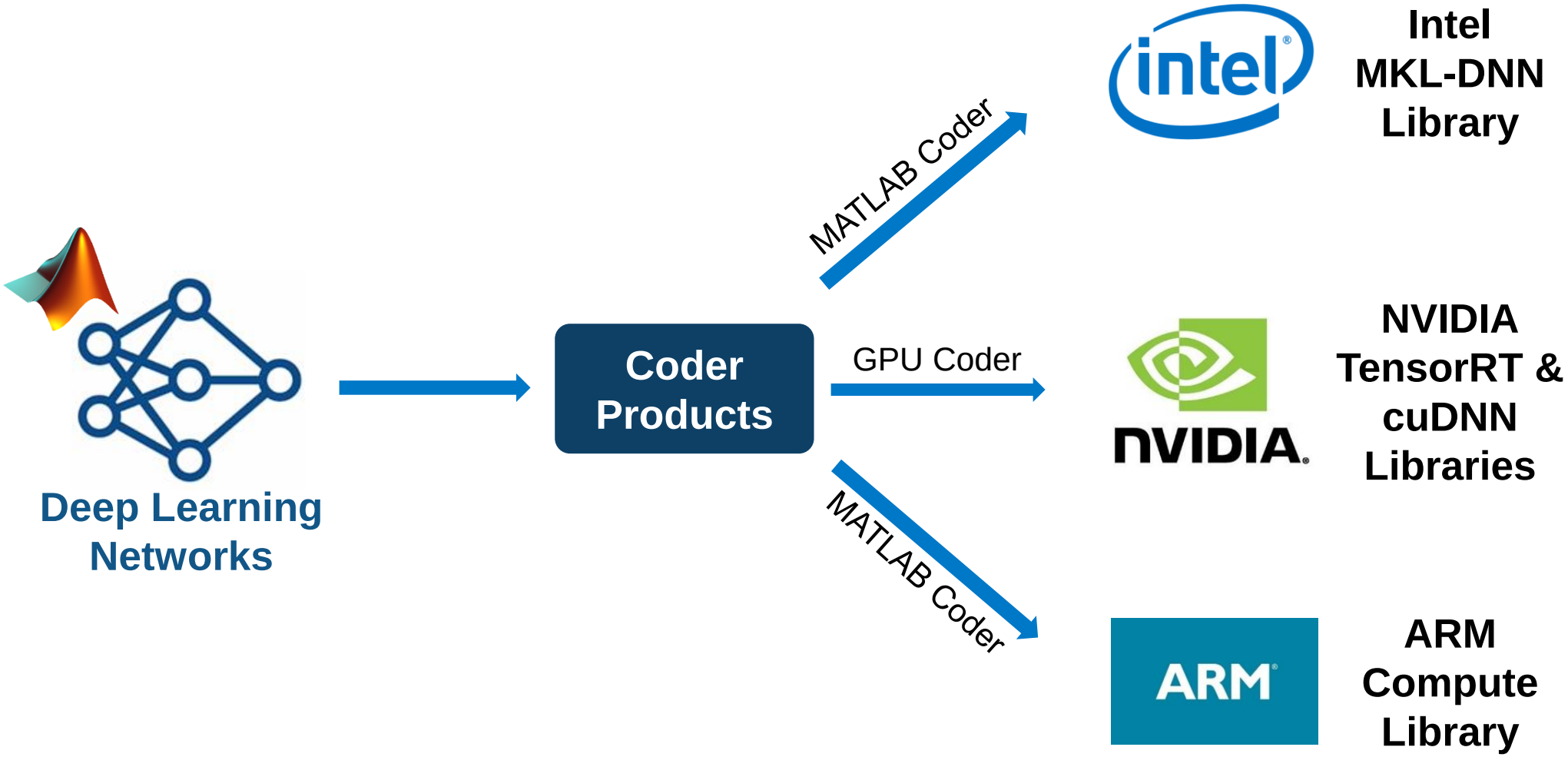
- **Out of Scope:** The CNN architecture layers and parameters

One Step Left – Deployment!

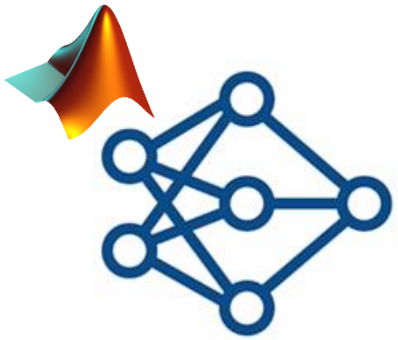


Deploying Deep Learning Models for Inference

R2017b+



Deploying to Various Targets

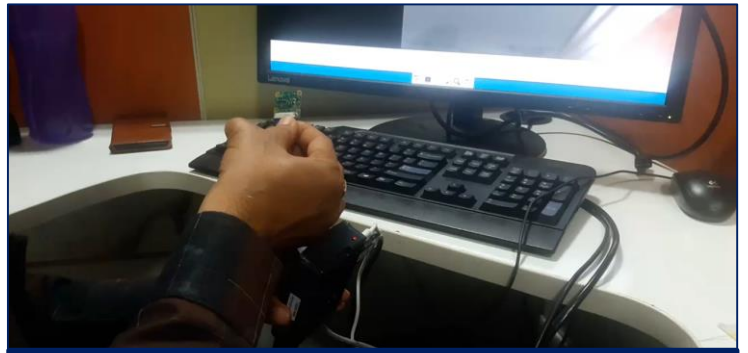


Deep Learning Networks

Coder Products

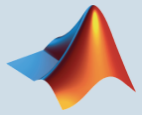
Desktop CPU

NVIDIA TensorRT & cuDNN Libraries

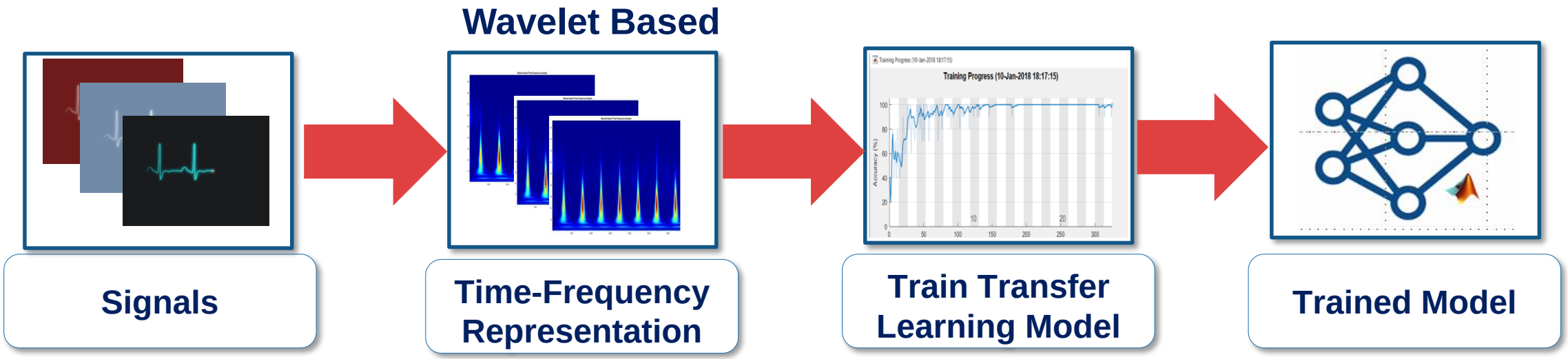


Raspberry Pi board

Agenda

- Demonstration and Overview
- Signal Analysis Challenges and Techniques
- Introduction to Deep Learning
- Workflow and Solution in MATLAB
-  Summary and Support

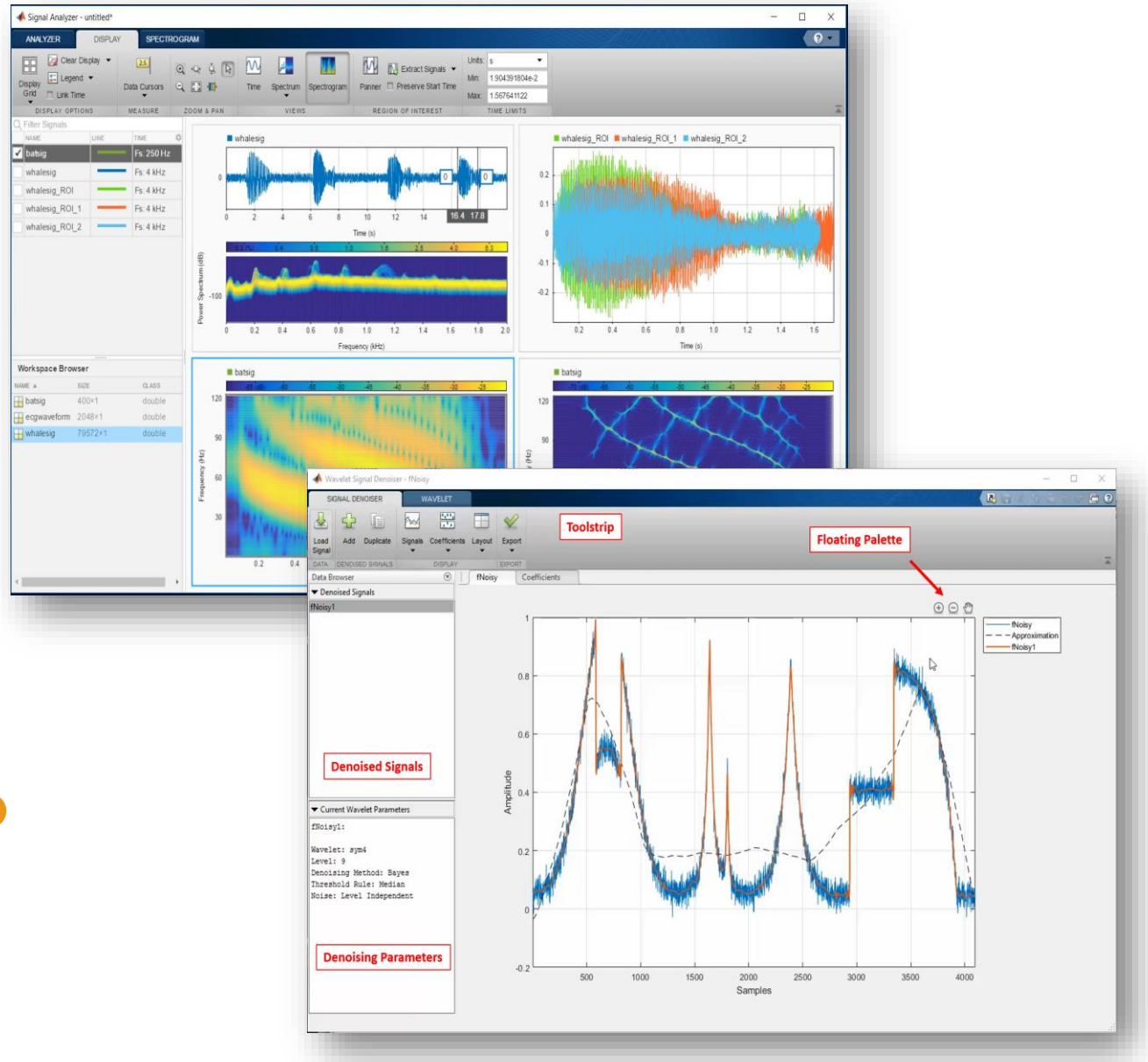
Deep Learning on Signals Workflow



Signal Processing

- Signal Analyzer app R2016a
  - Visualize and compare multiple signals and spectra
  - Spectral analysis of signals
  - Time domain panning
  - Automatic MATLAB code generation R2017b
  - Preprocess signals by smoothing and filtering R2018a

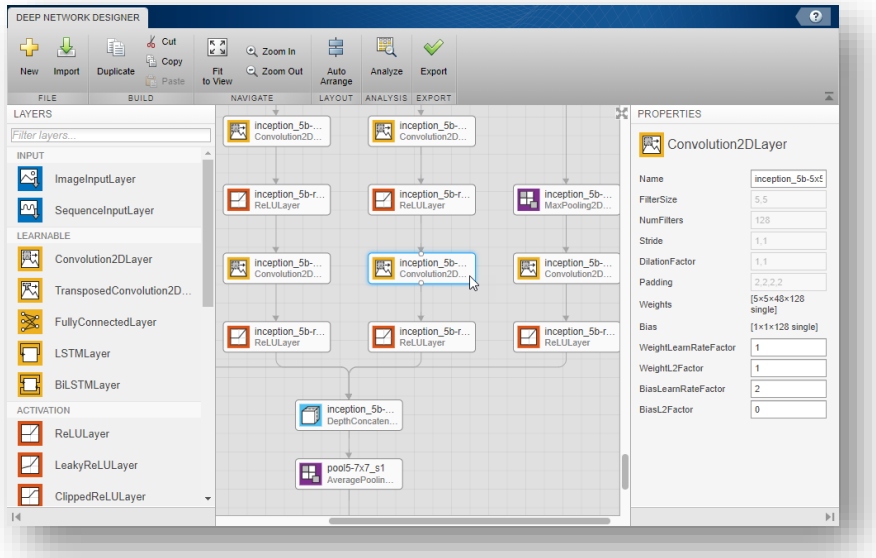
- Wavelet Signal Denoiser App R2017b
  - Visualize and denoise time-series data
  - Automatic MATLAB code generation

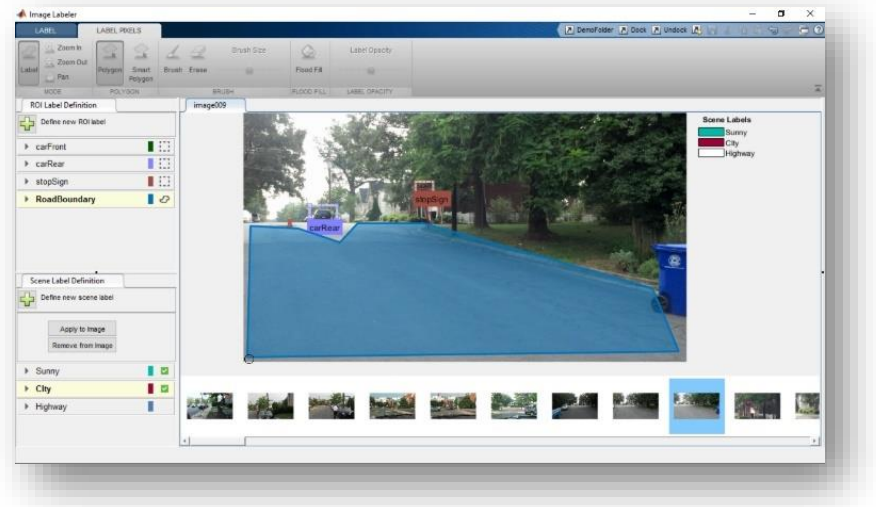


Signal Processing Toolbox
Wavelet Toolbox

Designing and Building Deep Learning Models

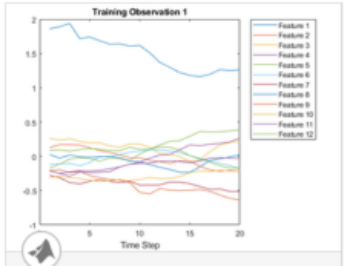
- Edit and build deep networks (Deep Network Designer app) R2018b
- Visualize, analyze, and find problems in network architectures before training (Network Analyzer) R2018b
- Automate ground-truth labeling using apps
  - Image Labeler app R2017b+
  - Video Labeler app R2018b
  - Audio Labeler app R2018b
- Monitor training progress with plots for accuracy, loss, validation metrics, and more R2017b
- Visualize and debug deep learning models R2017b





Deep Learning Toolbox
Computer Vision System Toolbox
Audio System Toolbox

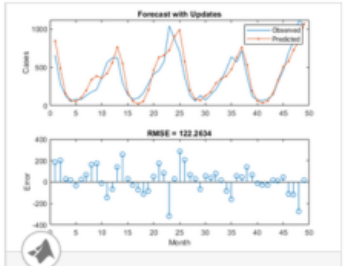
LSTM Examples in MATLAB Documentation



Sequence Classification Using Deep Learning

Classify sequence data using a long short-term memory (LSTM) network.

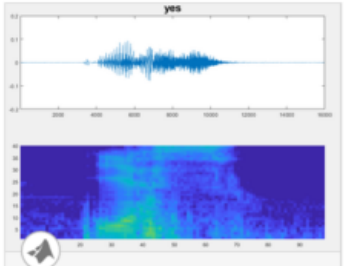
Open Live Script



Time Series Forecasting Using Deep Learning

Forecast time series data using a long short-term memory (LSTM) network.

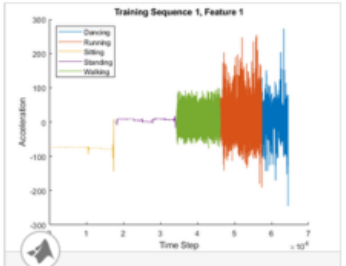
Open Live Script



Speech Command Recognition Using Deep Learning

Train a simple deep learning model that detects the presence of speech commands in audio. The example uses the Speech Commands

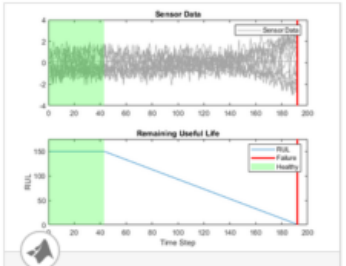
Open Script



Sequence-to-Sequence Classification Using Deep Learning

Classify each time step of sequence data using a long short-term memory (LSTM) network.

Open Live Script



Sequence-to-Sequence Regression Using Deep Learning

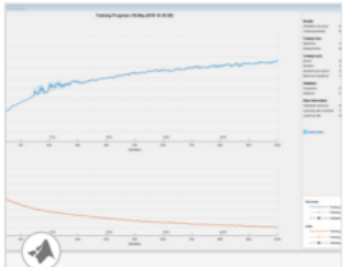
Predict the remaining useful life (RUL) of engines by using deep learning.

Open Live Script

Classify Text Data Using Deep Learning

Classify text descriptions of weather reports using a deep learning long short-term memory (LSTM) network.

Open Live Script



Generate Text Using Deep Learning

Train a deep learning long short-term memory (LSTM) network to generate text.

Open Live Script

Pride and Prejudice and MATLAB

Train a deep learning LSTM network to generate text using character embeddings.

Open Live Script

Word-By-Word Text Generation Using Deep Learning

Train a deep learning LSTM network to generate text word-by-word.

Open Live Script

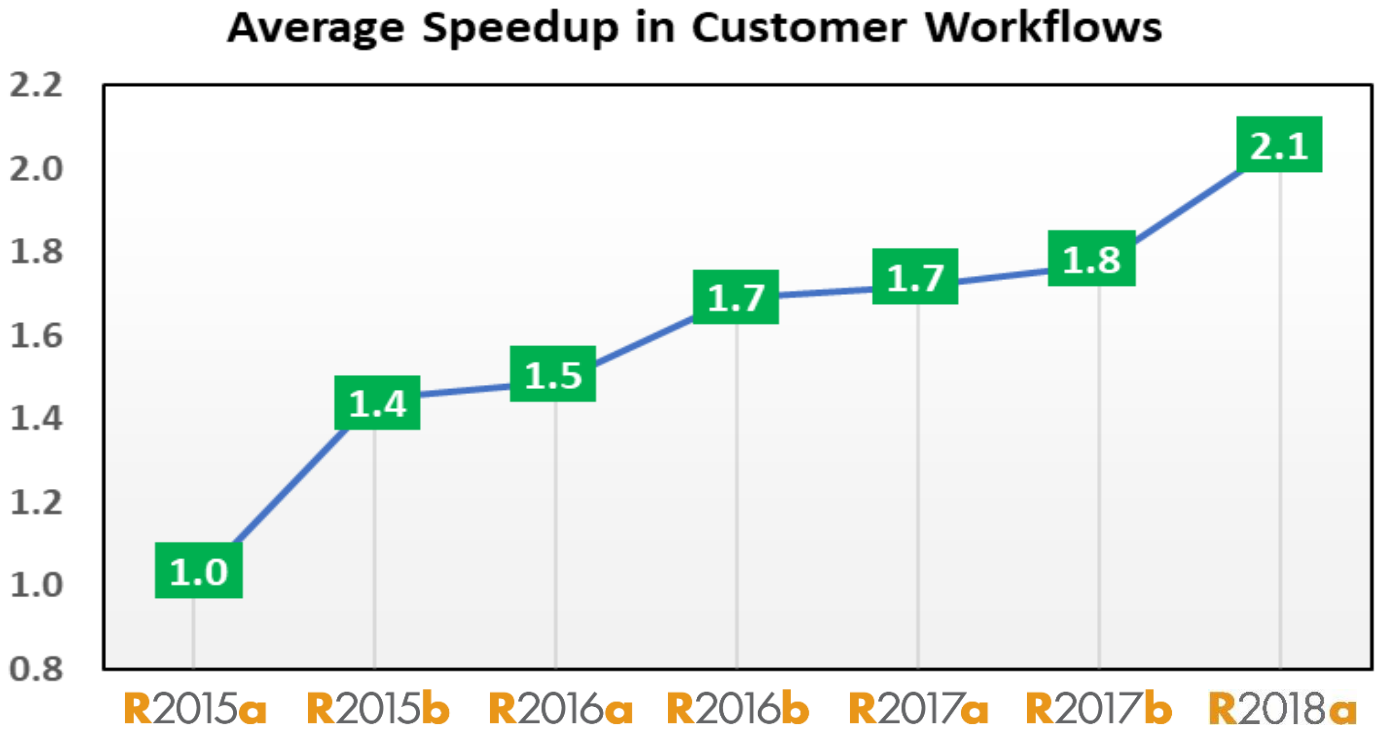
Classify Out-of-Memory Text Data Using Custom Mini-Batch Datastore

Classify out-of-memory text data with a deep learning network using a custom mini-batch datastore.

Open Live Script

What's New in MATLAB

- Live Editor
- App Designer
- Toolbox Apps with Automatic MATLAB Code Generation
- Data Analytics
  - Big Data (`tall` arrays)
  - Machine Learning
  - Deep Learning (+ GPUs)
- New / Enhanced Graphics
- External Interfaces
- Hardware Support



https://www.mathworks.com/products/matlab/whatsnew.html

What's Next?

Resource for Getting Started

- Documentation
- In-product features
- Tutorials
- Webinars

Self Serve

- MATLAB Answers
- File Exchange
- MATLAB Examples
- GitHub
- MOOCs
- Blogs

Community Support

Tech Support (phone and email)

- 95% of calls answered <3 min
- 70% issues resolved <24hrs

MathWorks Support

Training

- MATLAB Academy
- Online
- Onsite

Consulting

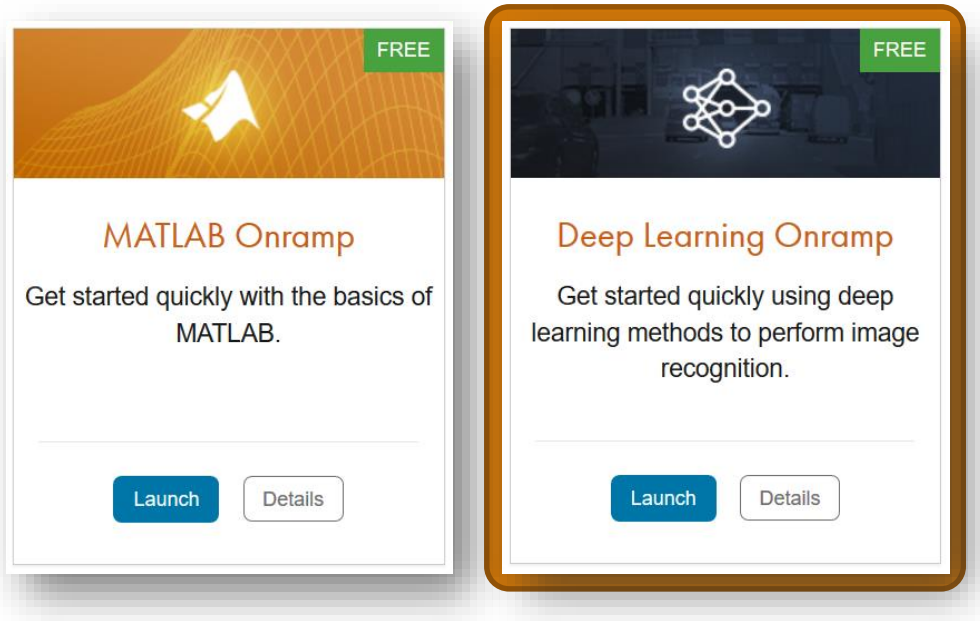
- Domain Experts
- From tool adoption to process integration

Services

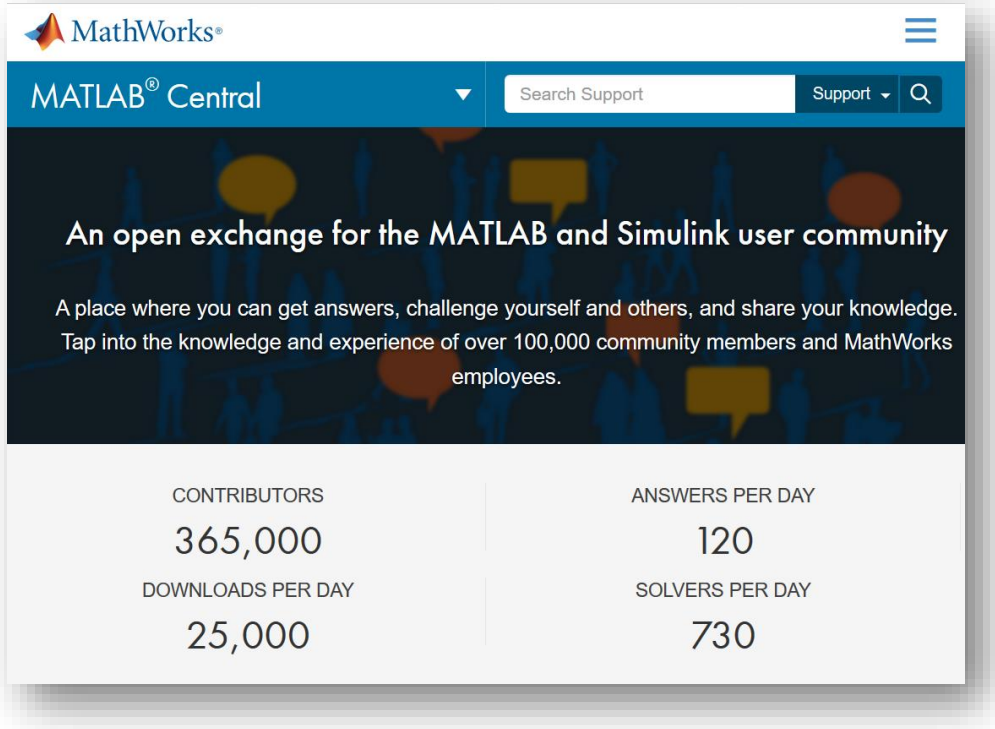
What's Next?

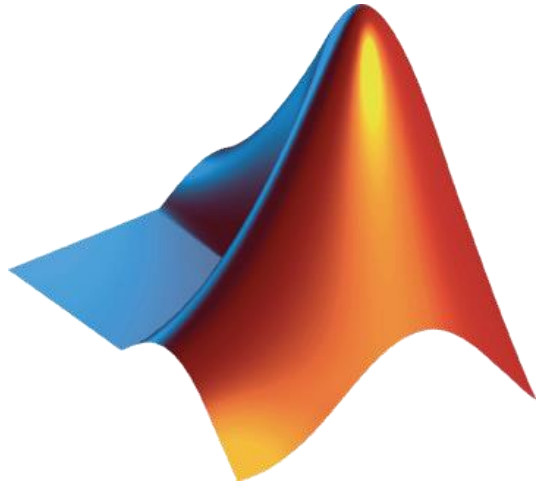
Resource for Getting Started

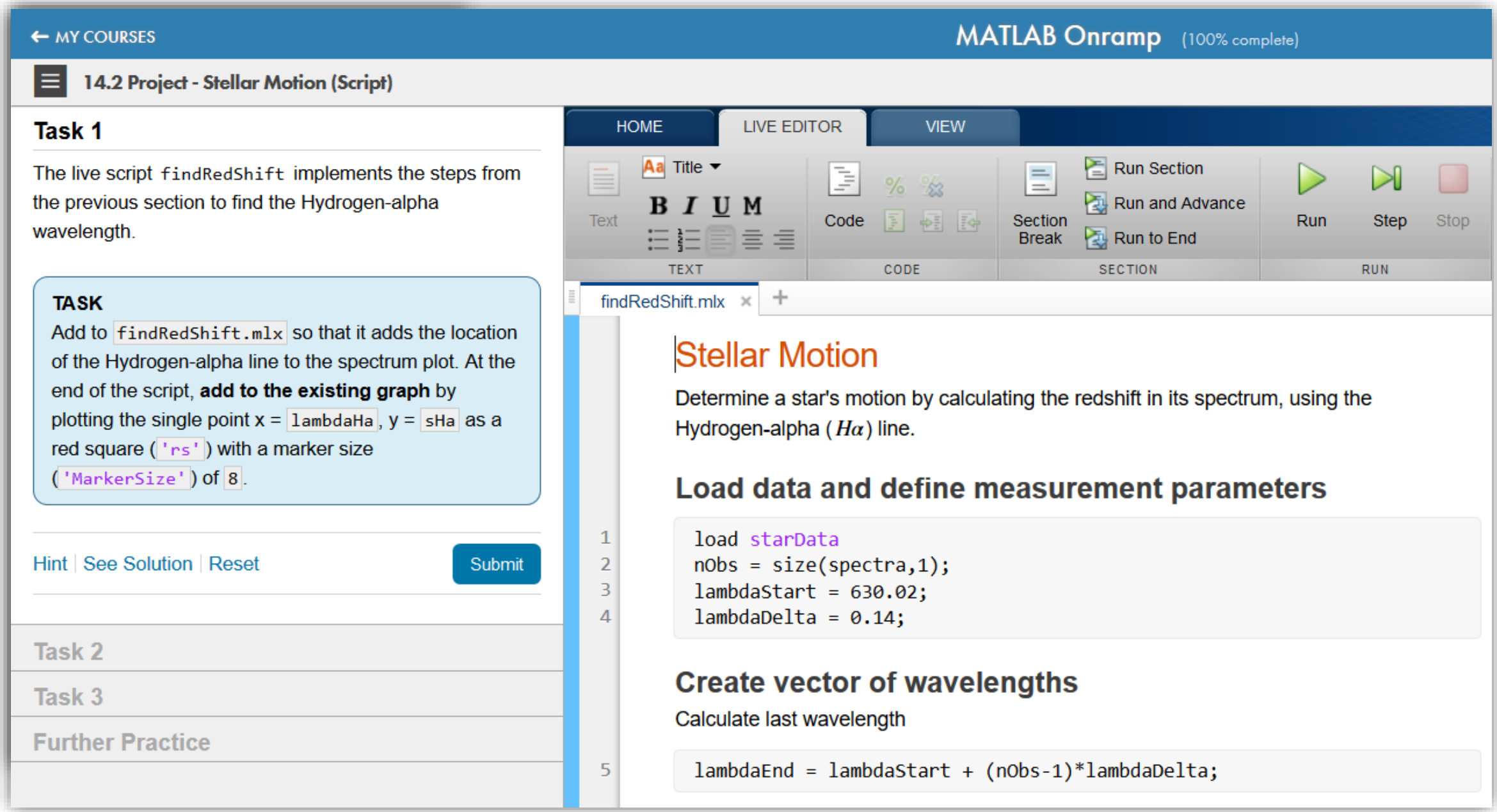
- MATLAB Academy
  - Online Courses
  - **FREE** Onramps



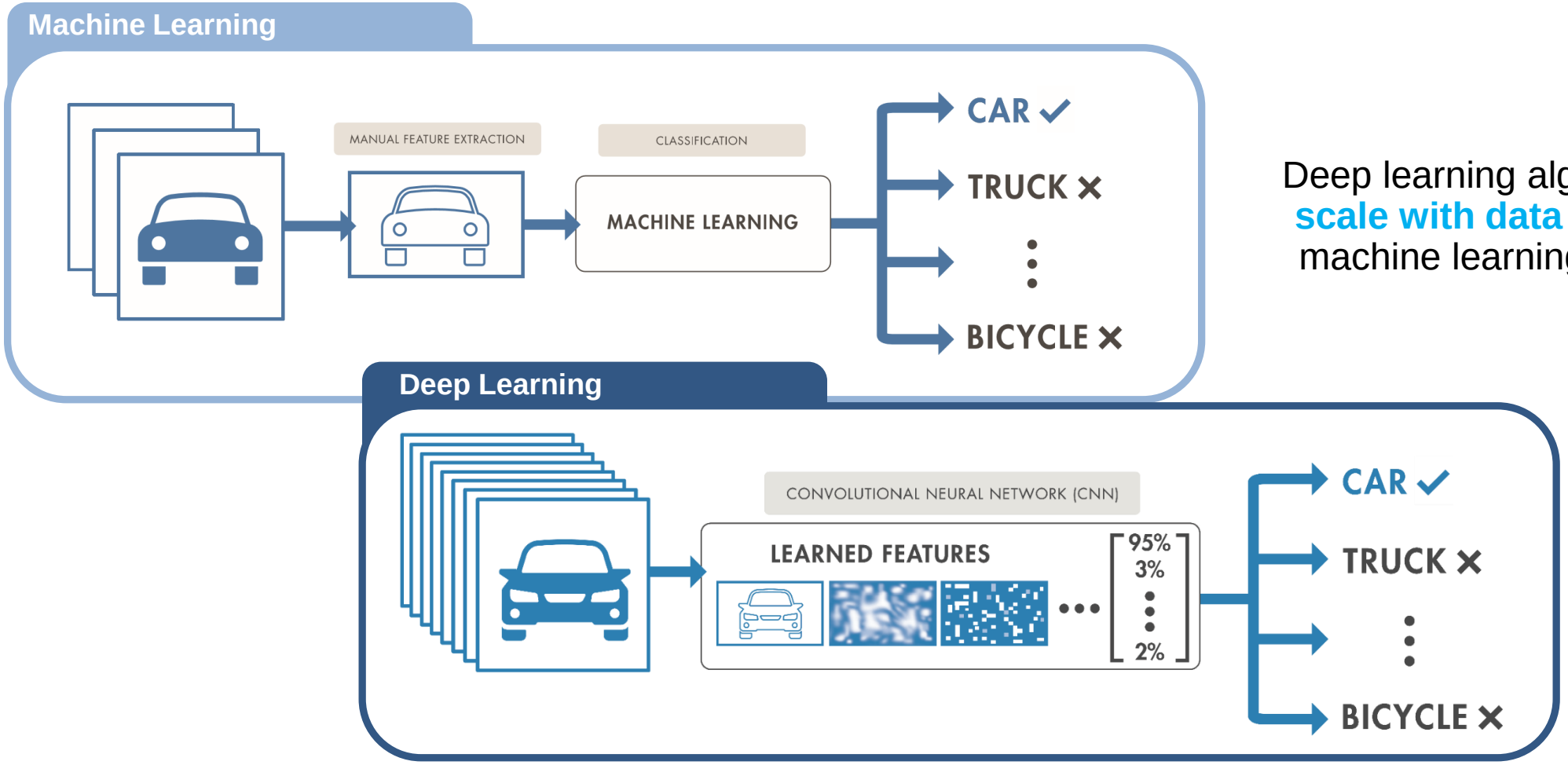
- MATLAB Central
  - File Exchange
  - MATLAB Answers






MY COURSES
MATLAB Onramp (100% complete)
14.2 Project - Stellar Motion (Script)
Task 1
The live script findRedShift implements the steps from the previous section to find the Hydrogen-alpha wavelength.
TASK
Add to findRedShift.mlx so that it adds the location of the Hydrogen-alpha line to the spectrum plot. At the end of the script, add to the existing graph by plotting the single point x = lambdaHa, y = sHa as a red square ('rs') with a marker size ('MarkerSize') of 8.
Hint | See Solution | Reset
Submit
Task 2
Task 3
Further Practice
HOME
LIVE EDITOR
VIEW
Text
Title
Code
Section Break
Run Section
Run and Advance
Run to End
Run
Step
Stop
TEXT
CODE
SECTION
RUN
findRedShift.mlx
Stellar Motion
Determine a star's motion by calculating the redshift in its spectrum, using the Hydrogen-alpha (Hα) line.
Load data and define measurement parameters
1 load starData
2 nObs = size(spectra,1);
3 lambdaStart = 630.02;
4 lambdaDelta = 0.14;
Create vector of wavelengths
Calculate last wavelength
5 lambdaEnd = lambdaStart + (nObs-1)*lambdaDelta;

Machine Learning vs Deep Learning



Deep learning algorithms also **scale with data –** traditional machine learning **saturates**

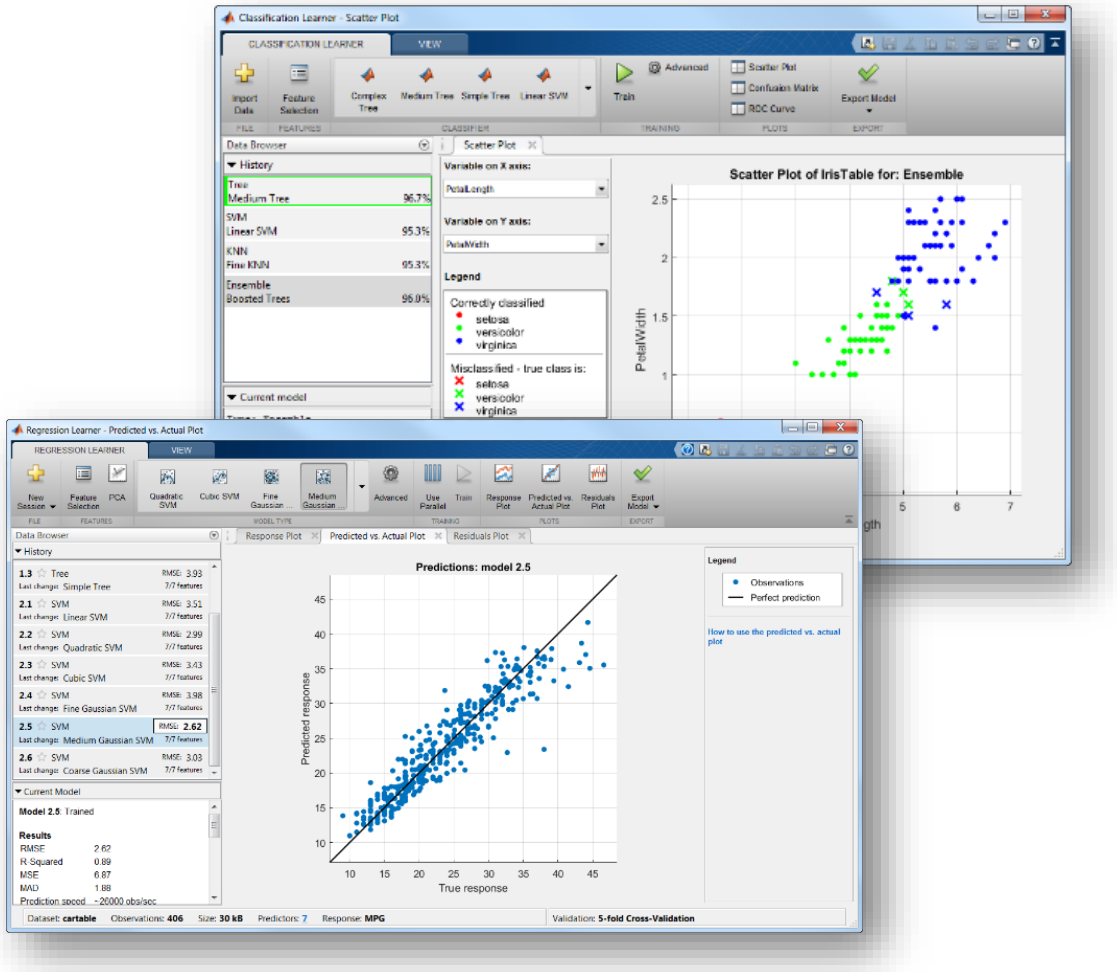
Deep learning performs **end-to-end learning** by learning **features, representations and tasks** directly from **images, text and sound**.

Machine Learning

"I would have never attempted machine learning if this app was not available."

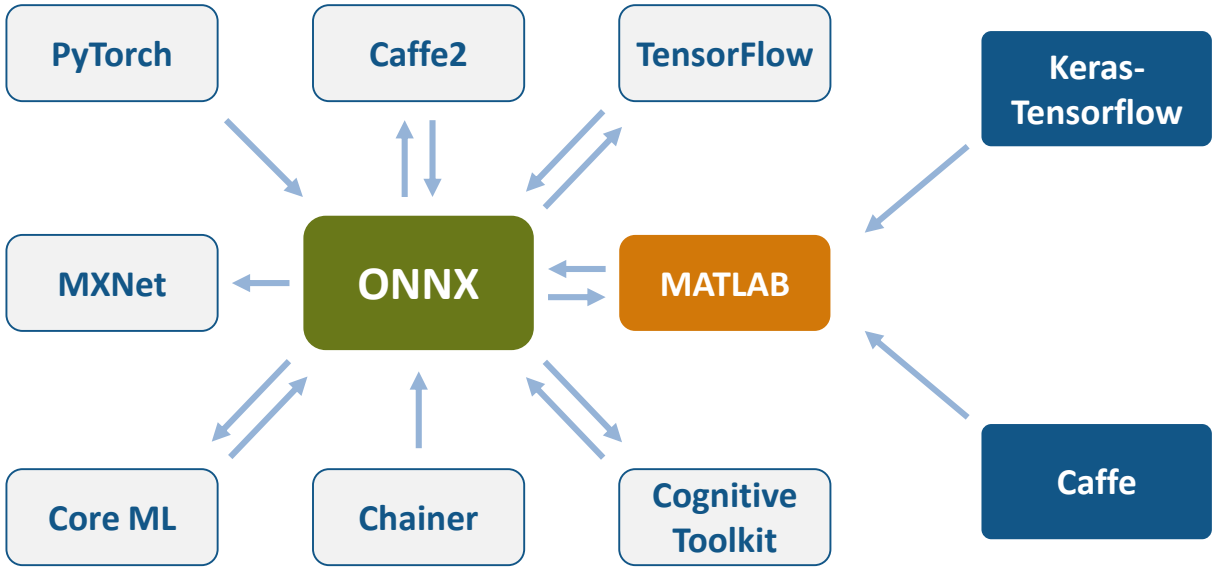
MATLAB makes machine learning easy and accessible for everyone, even if you're not an expert

- Use apps to interactively explore data, choose algorithms to train and validate models, and compare results
  - Classification Learner app R2015a
  - Regression Learner app R2017a
- Apply algorithms to out-of-memory data using tall arrays R2016b+
- Generate C code for predictive models *(requires MATLAB Coder)* R2016b+



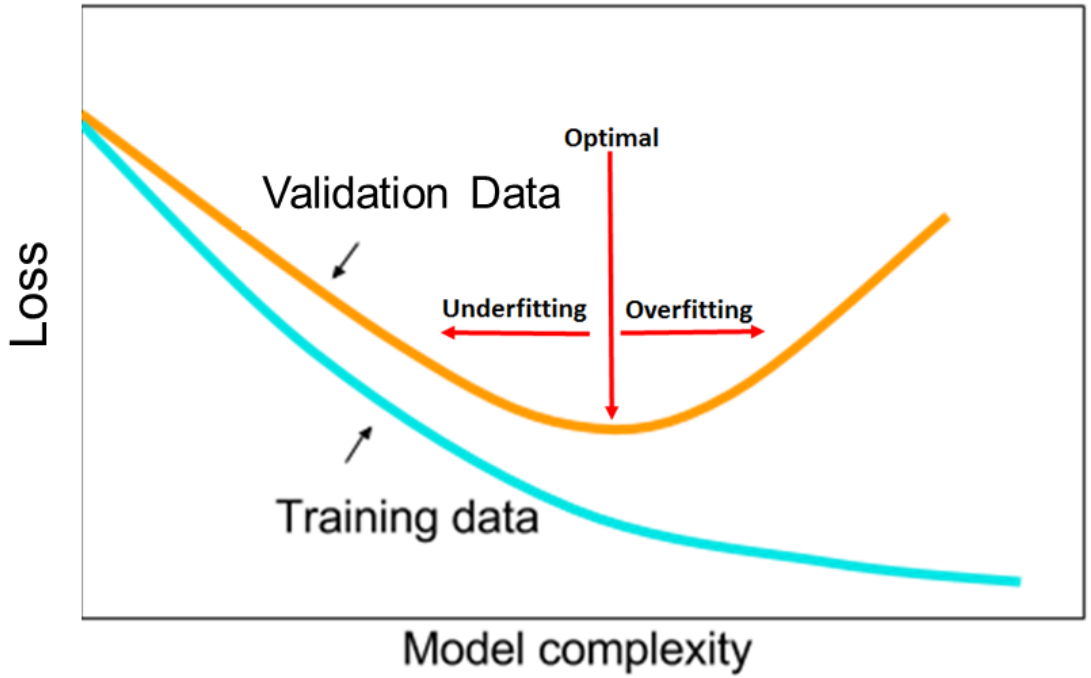
Interoperability with Deep Learning Frameworks

- Import and export models using the Open Neural Network Exchange (ONNX) format
- Model importers
  - Caffe
  - TensorFlow-Keras
- Access pretrained models with a single line of code



Interpreting Training Plots

- If your loss isn't falling,
  you aren't learning
- Then look out for:
  - Underfitting: training and validation loss both high
  - Overfitting: training loss is low but test loss is high
- Trade off with time it takes to train



Underfitting v Overfitting

- Underfitting suggests your model may be too simple
- Overfitting suggests your model has too many parameters/not enough data
- Good to start by deliberately overfitting