

# Introduction to Deep Learning with Watson Studio

Svetlana Levitan, PhD, and David Nichols, PhD  
Developer Advocate and Data Scientist  
IBM Cloud and Cognitive Software

July 31, 2019



# Contents



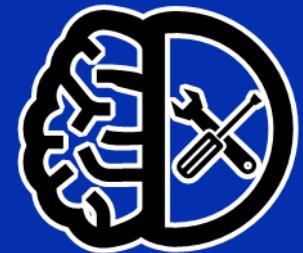
Intro to neural networks

Deep Learning: CNN and RNN

PMML and ONNX for model deployment

AIFairness360 and ART, MAX

Links and resources



# IBM's Long History with Open Source and open standards



ONNX

# Center for Open Source Data and AI Technologies

CODAIT aims to make AI solutions dramatically easier to create, deploy, and manage in the enterprise

Relaunch of the Spark Technology Center (STC) to reflect expanded mission



## CODAIT

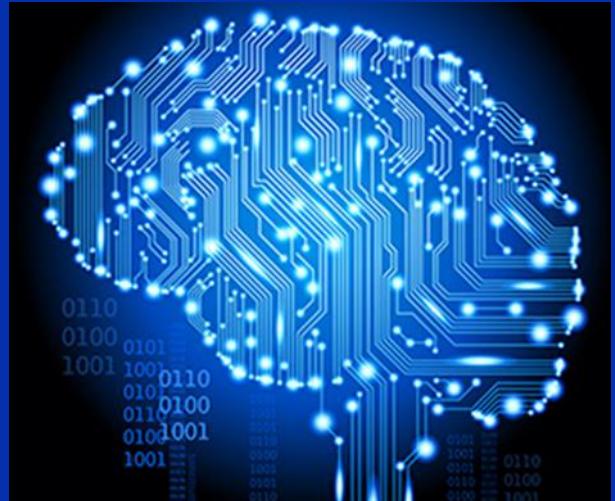
[codait.org](http://codait.org)

codait (French)  
= coder/coded

<https://m.interglot.com/fr/en/codait>



# Who uses machine learning today?



# Some Applications of Machine Learning

Handwritten digit recognition

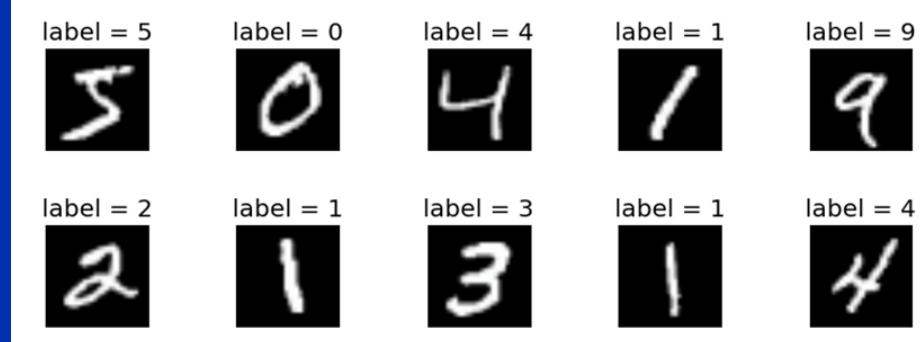
Weather forecast

Spam filters for email

Online shopping recommendations

Self-driving car

2011 Watson won Jeopardy  
2018 Project Debater



# Neural Network Architectures:

## Quick Tour

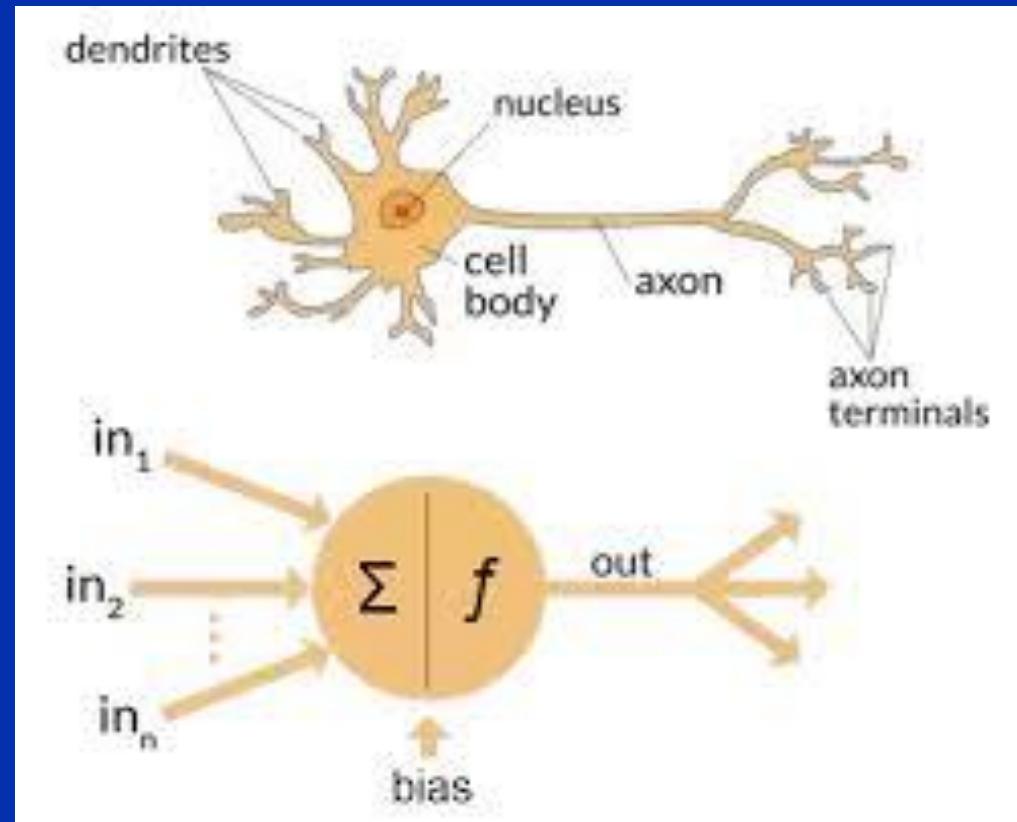
# The Elementary Perceptron

1958 Frank Rosenblatt

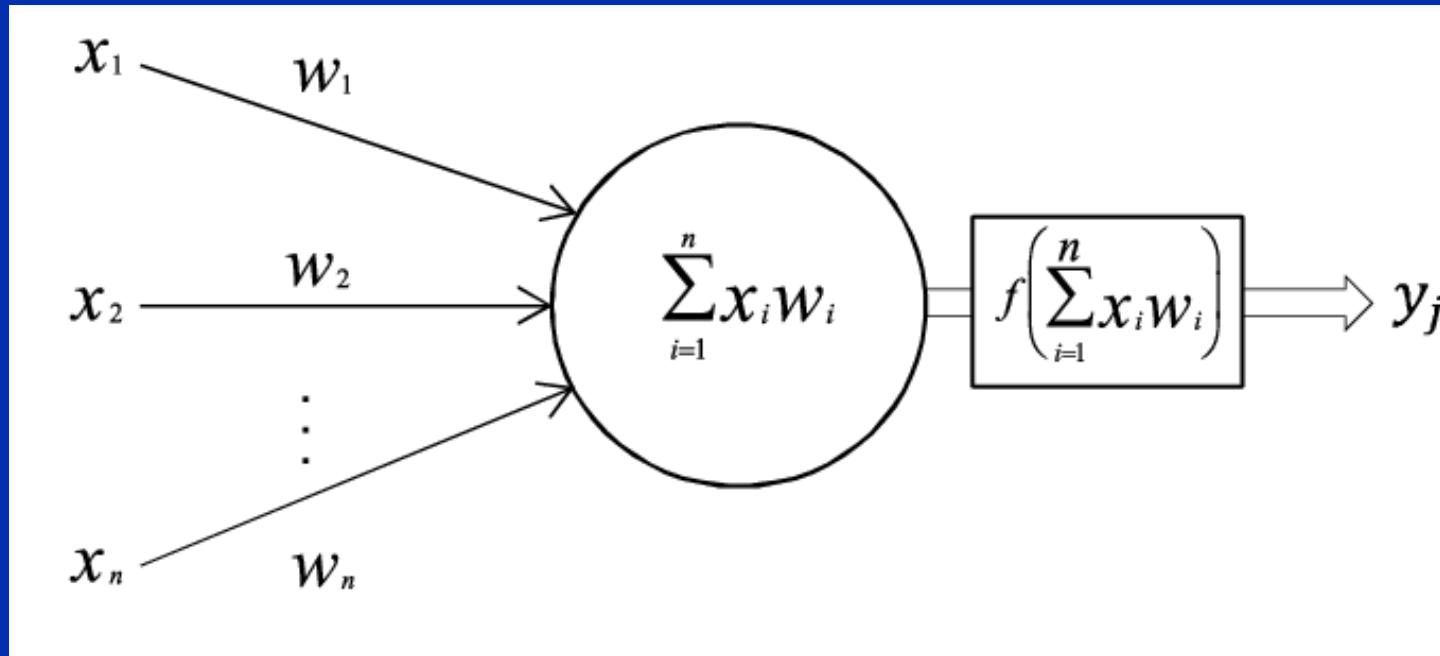
A machine, first implemented in software on IBM 704, later hardware

1969 book by Minsky and Papert  
→AI winter

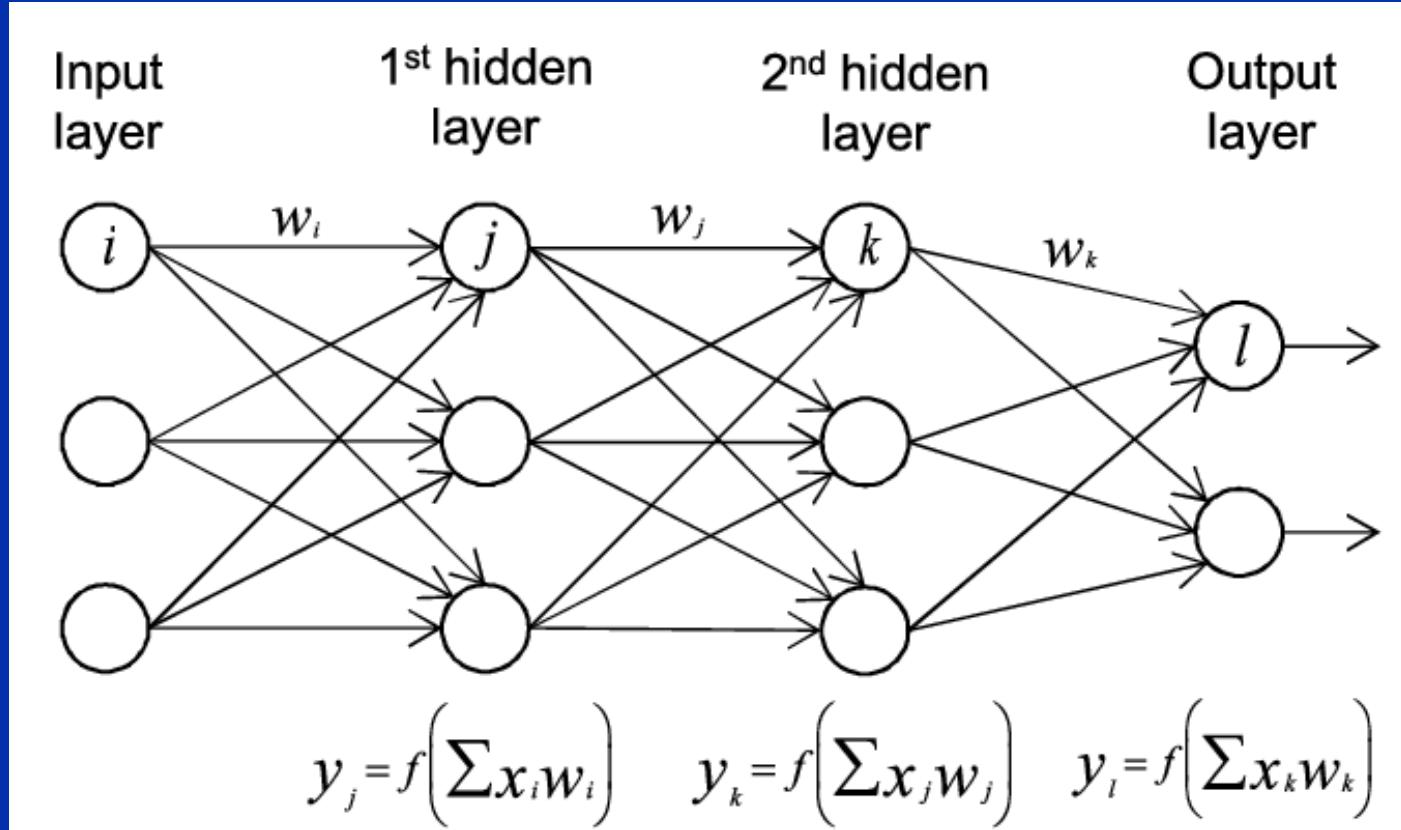
Then MLP = multilayer perceptron



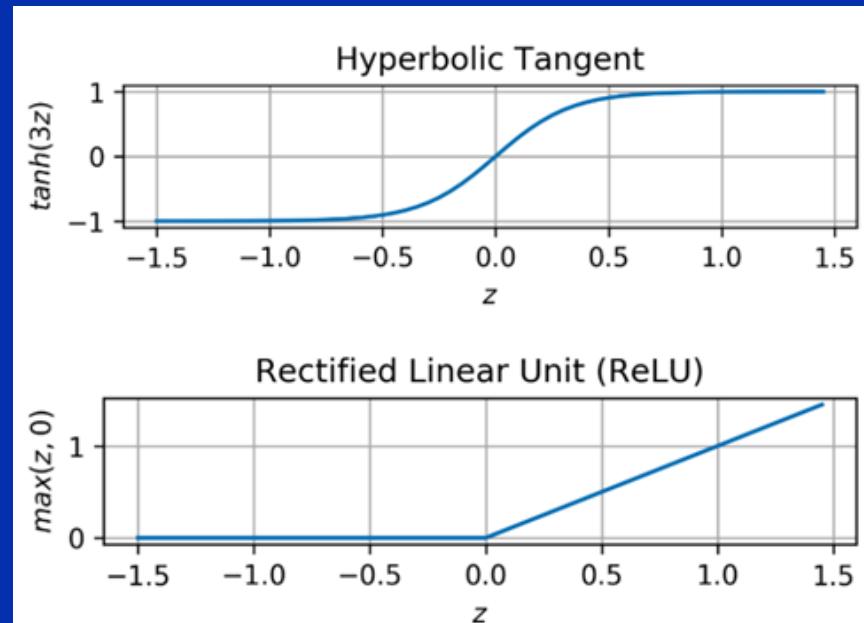
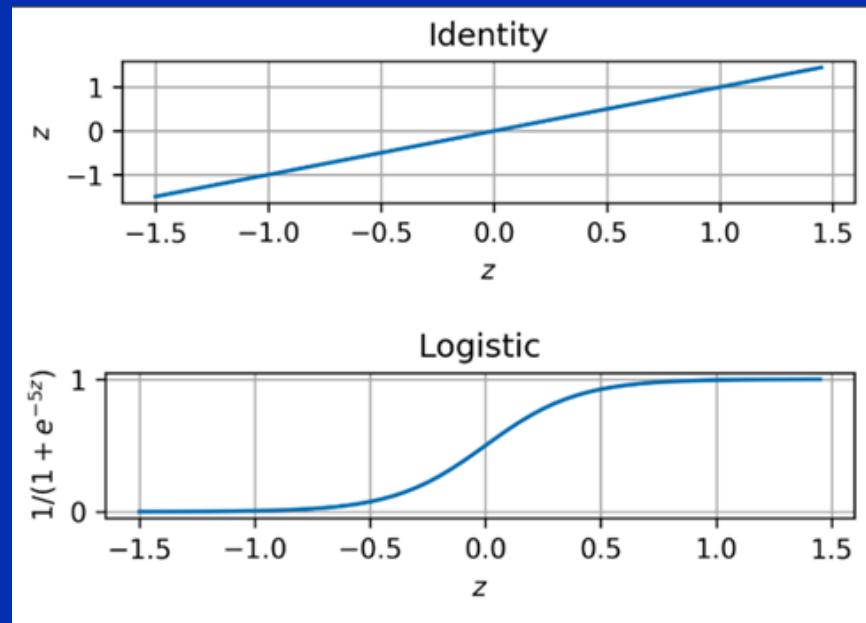
# One neuron



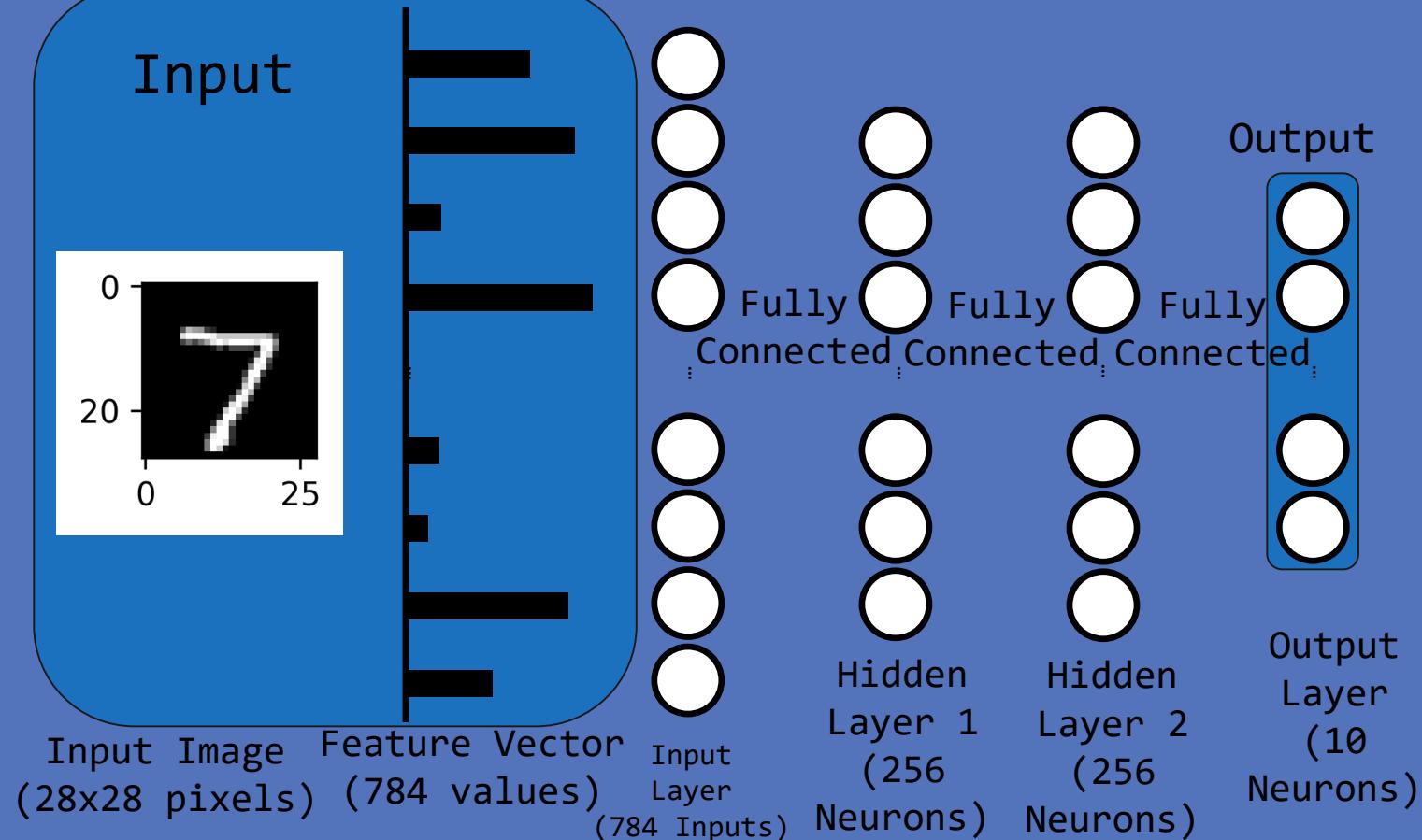
# Multi Layer Perceptron



# Activation Functions



# A Simple Neural Network for Identifying Numbers



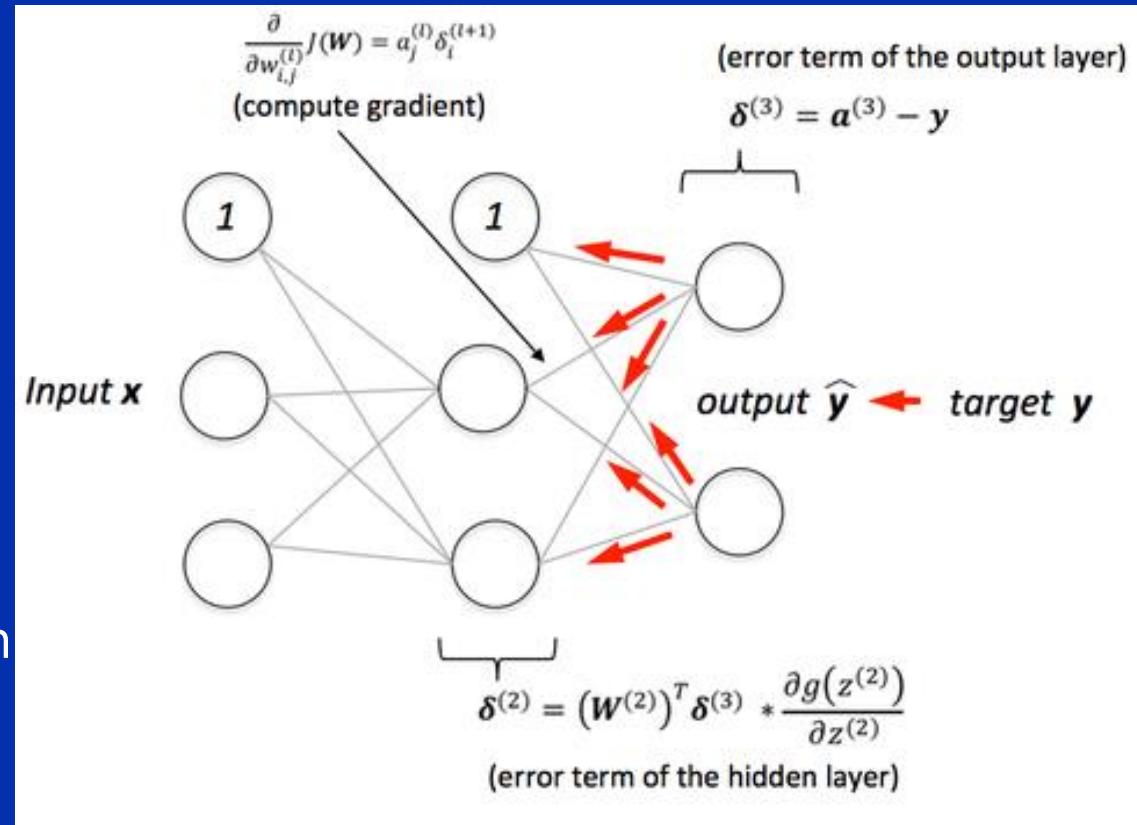
# Training Neural Networks with Backpropagation

Initialize weights with small random values

Apply inputs, compute predictions, propagate error back and update weights

Gradient descent methods:  
Adagrad, Adam, ...

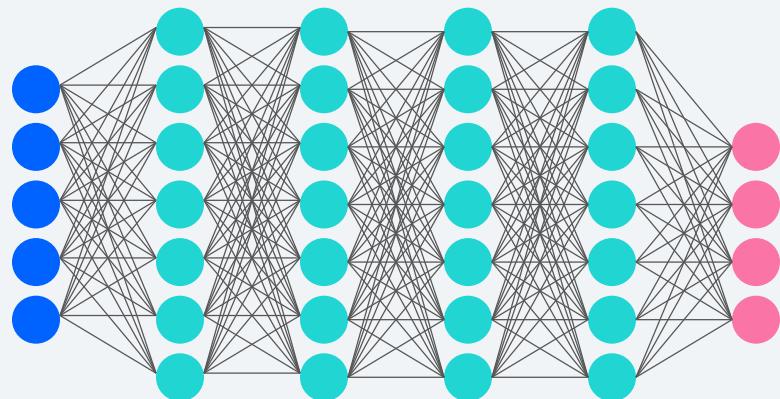
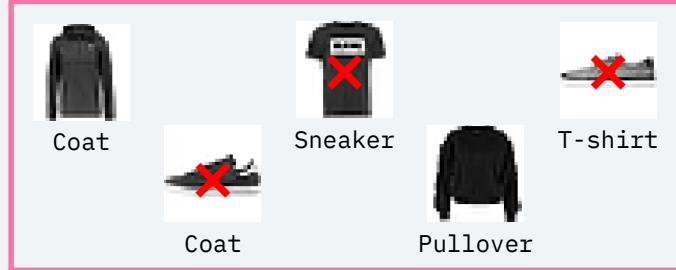
Online or mini-batch or batch



## Labeled Training Data



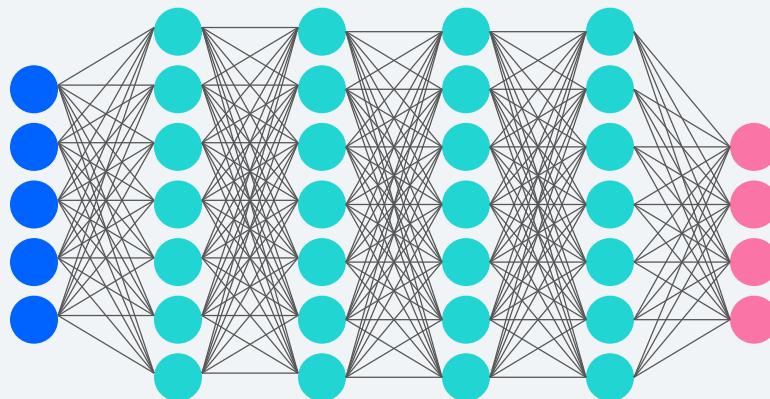
## Output Errors



Backpropagation

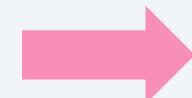
# Neural Network Inferencing

Input



Output

Sneaker  
98%



# Building neural network in SKLearn

```
from sklearn.preprocessing import StandardScaler  
scaler = StandardScaler()  
# Fit only to the training data  
scaler.fit(X_train)  
  
# Now apply the transformations to the data:  
X_train = scaler.transform(X_train)  
X_test = scaler.transform(X_test)  
  
from sklearn.neural_network import MLPClassifier  
mlp = MLPClassifier(hidden_layer_sizes=(256,256),max_iter=500)  
mlp.fit(X_train,y_train)  
  
predictions = mlp.predict(X_test)  
from sklearn.metrics import classification_report,confusion_matrix  
print(confusion_matrix(y_test,predictions))
```

# Convolutional neural networks

# Convolutional Neural Networks (CNNs)

Widely used in processing of visual images



Photo by Ramdan Authentic

# Image representation

Here's the 28 x 28 set of numbers for the digit 7 image:

# Convolutional Neural Networks (CNNs)

CNNs typically involve use of several types of layers:

- Convolutional
- Pooling
- Dropout
- Flattening
- Dense (Fully Connected)

# Convolutional filters or kernels

Images on the following four slides are from Stanford University's CS231n Convolutional Networks for Visual Recognition course notes, and are subject to:

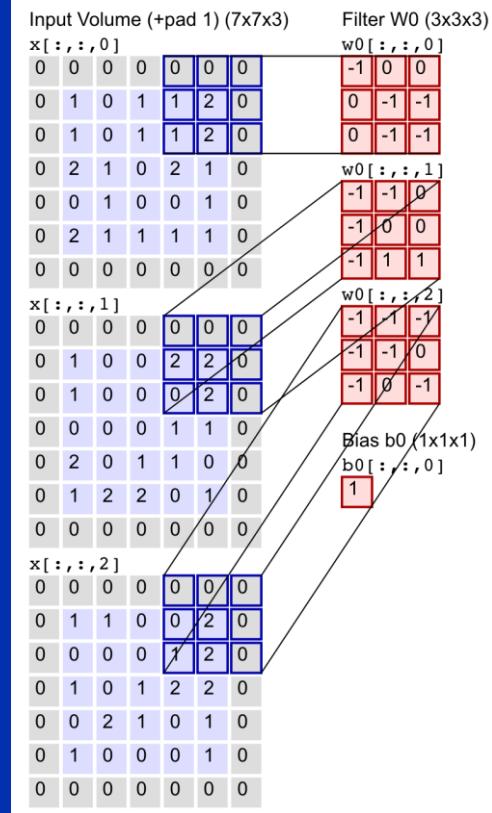
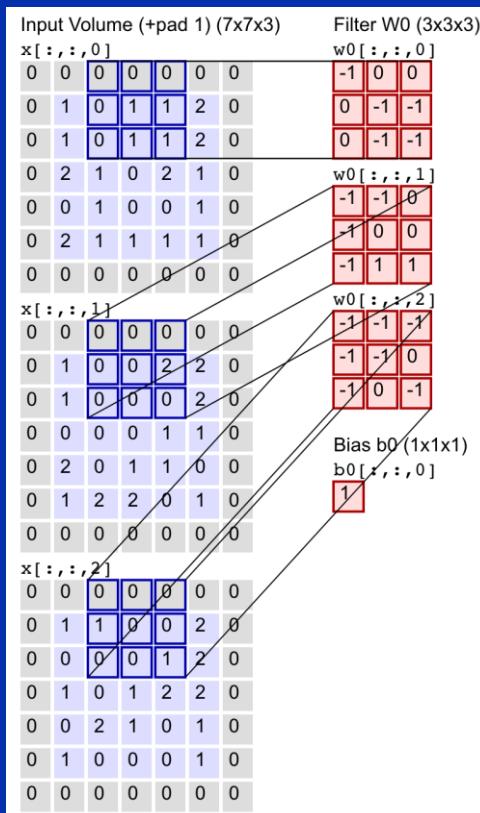
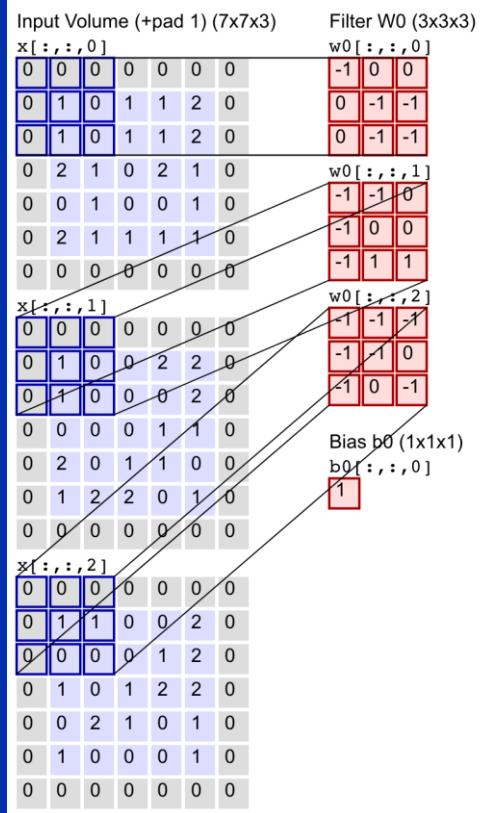
The MIT License (MIT)

Copyright © 2015 Andrej Karpathy

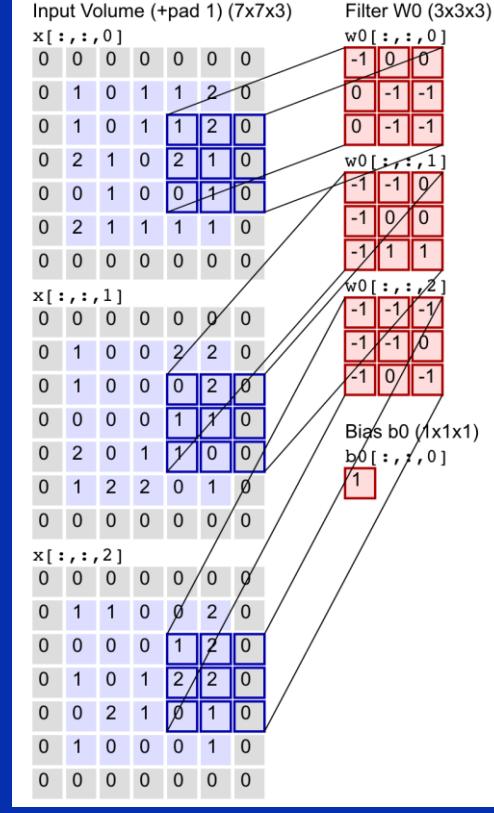
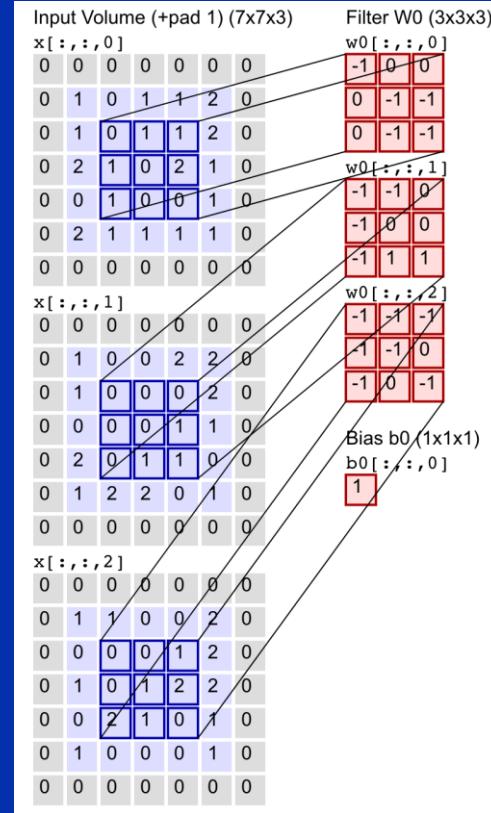
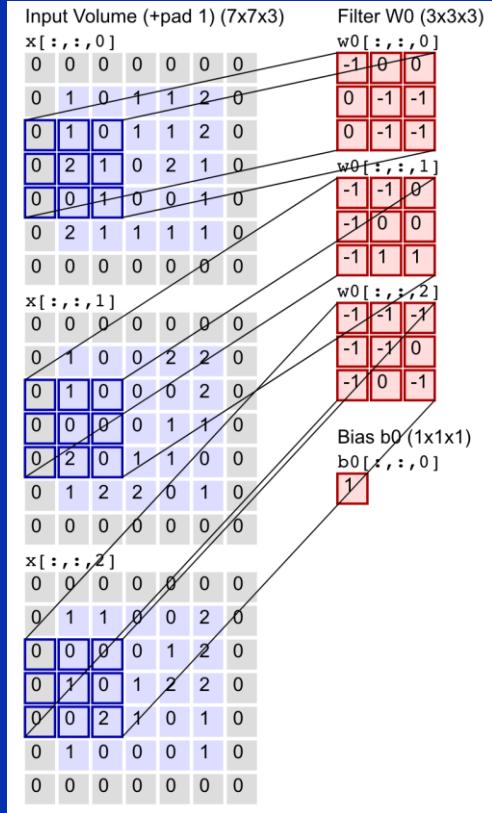
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

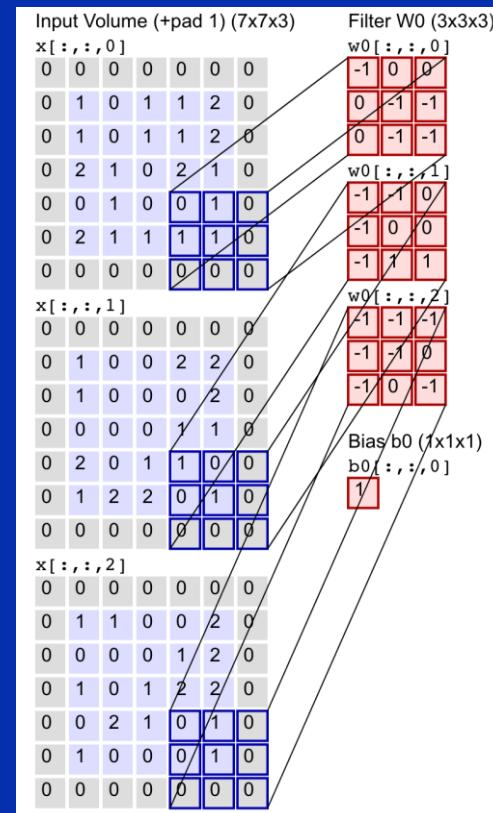
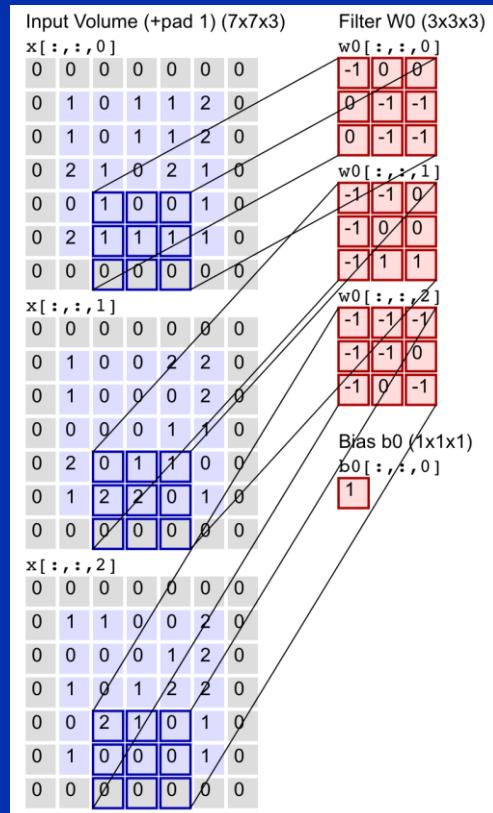
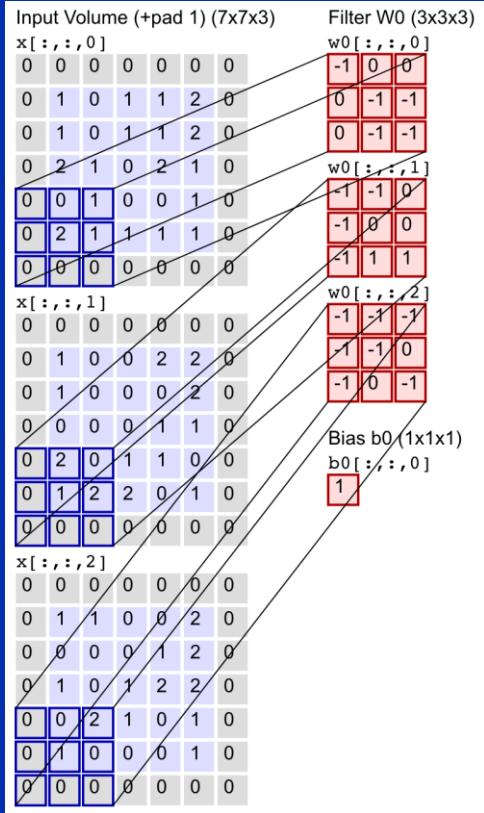
# Convolutional filters or kernels



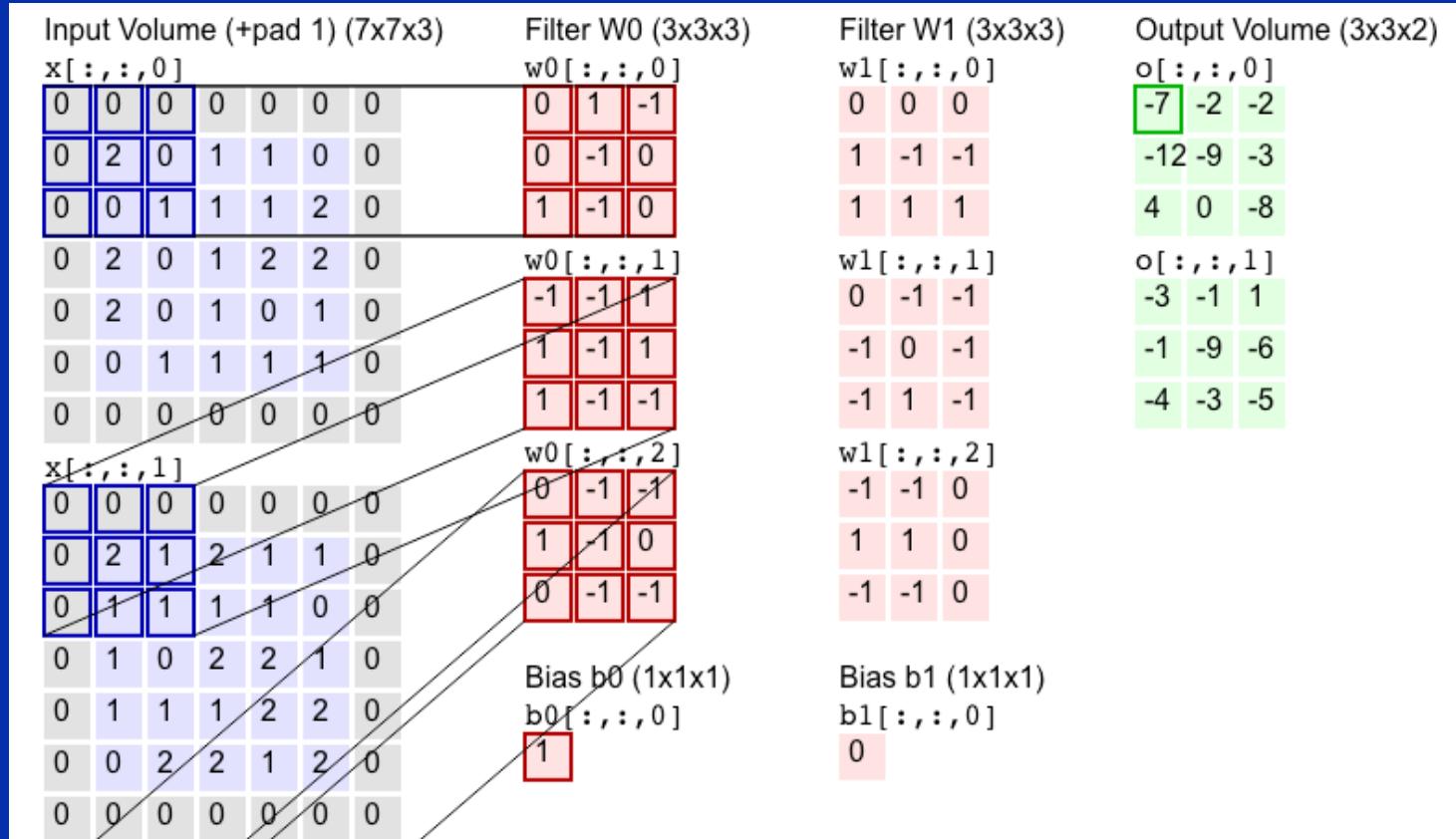
# Convolutional filters or kernels



# Convolutional filters or kernels



# Convolutional layers



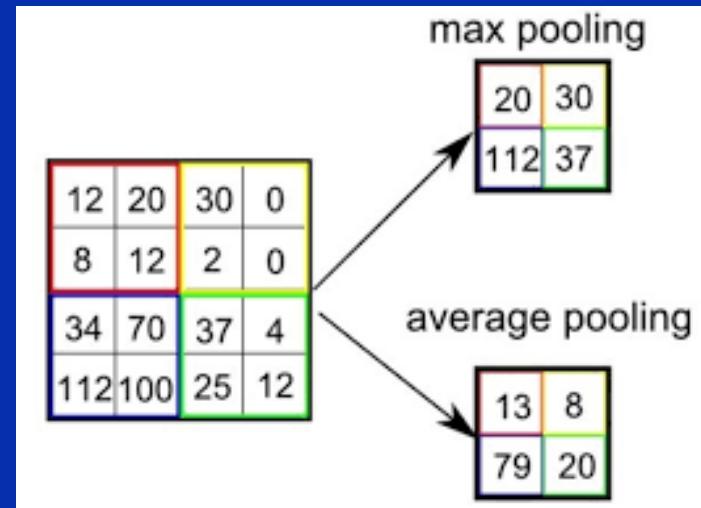
# Pooling layers

Reduce activation map sizes and serve a regularization function

Most common is a  $2 \times 2$  filter with strides (2,2) and taking the maximum of the four values as the new single value (“max pooling”)

This reduces the size of the activation map by 75% and helps reduce overfitting

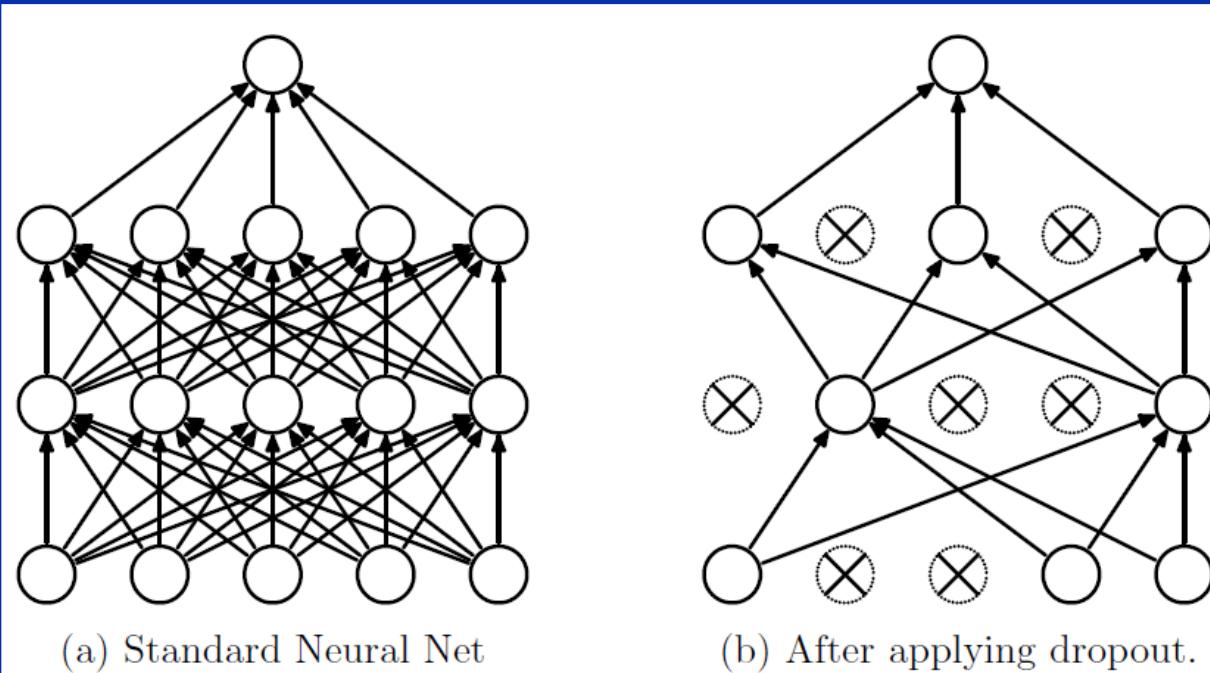
Diagram from quora.com



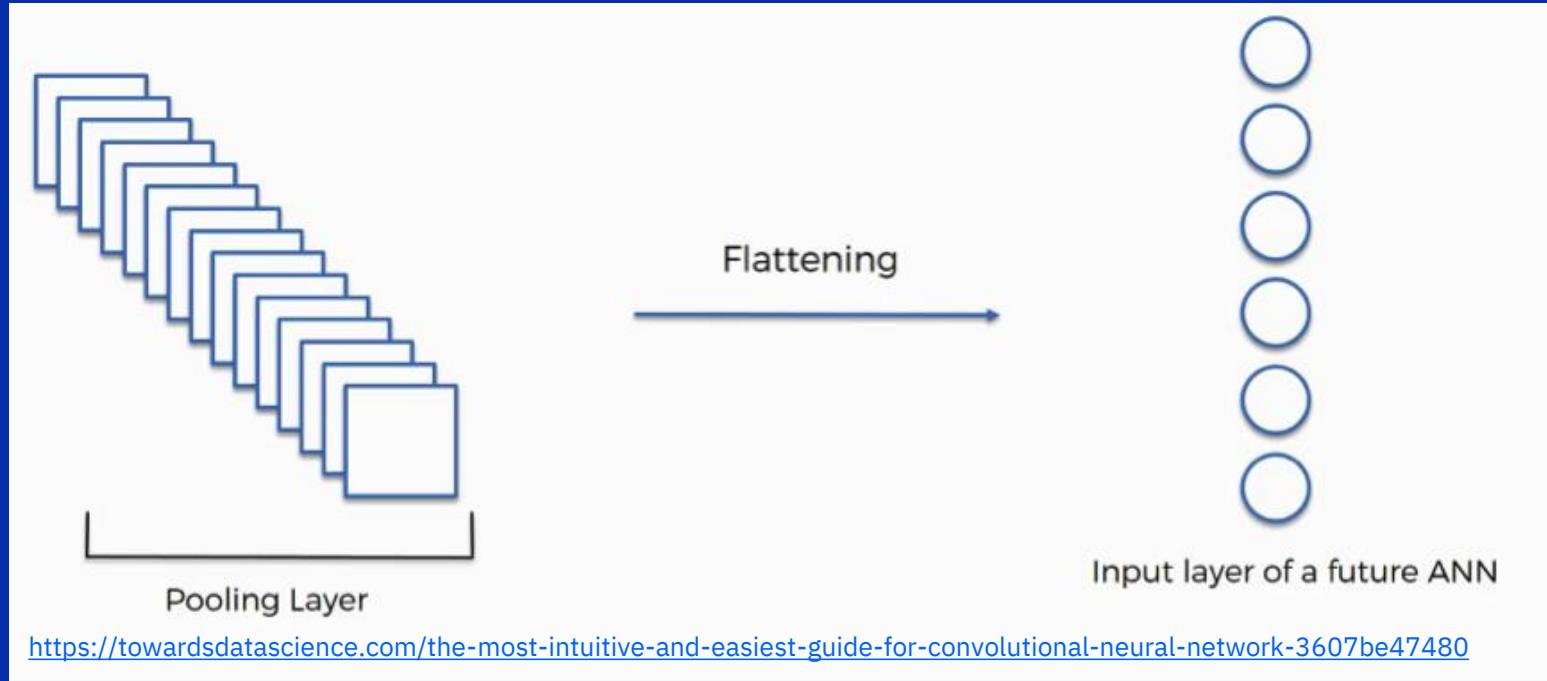
# Dropout Layers

Also serve regularization function

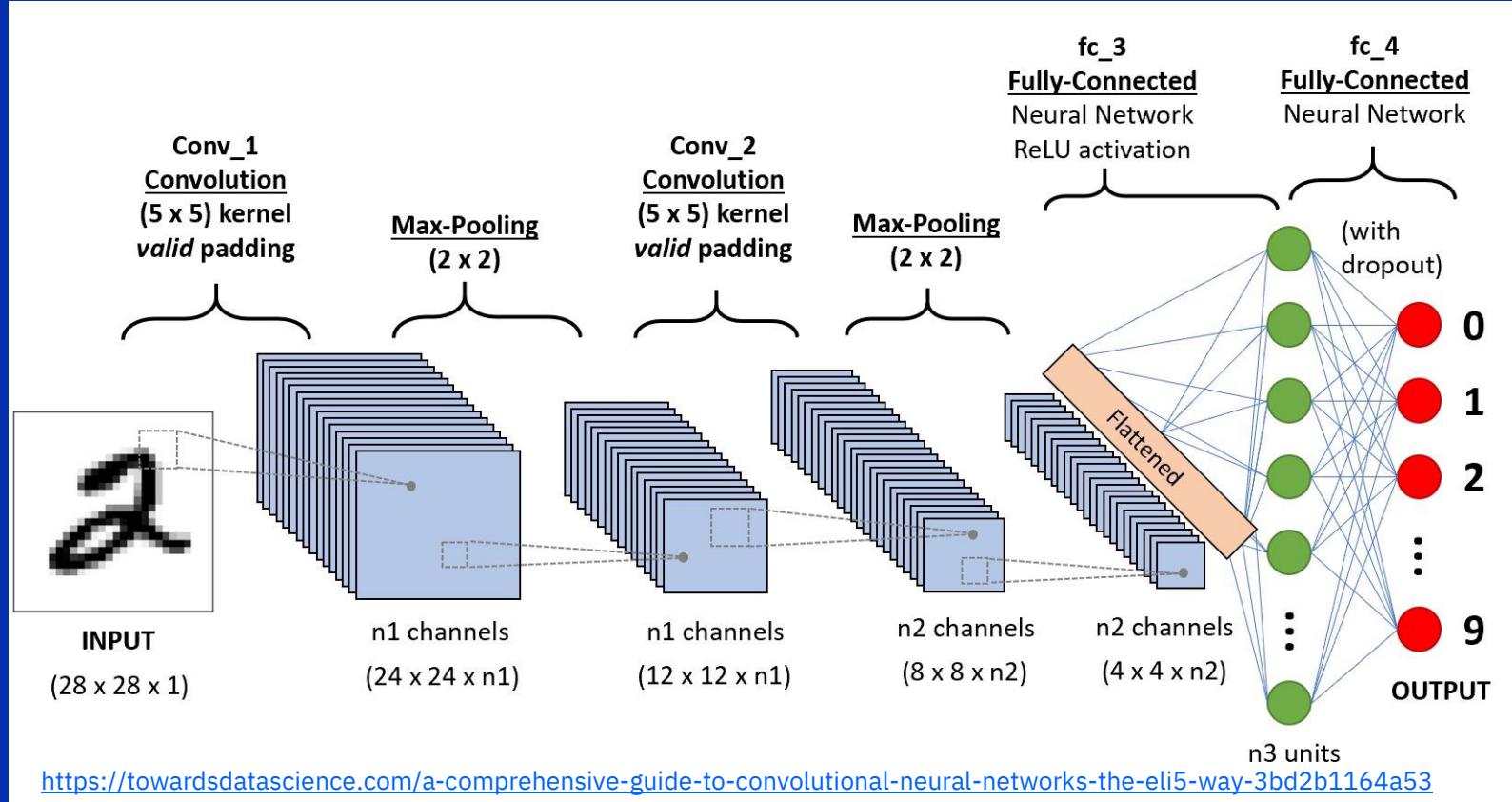
N. Srivastava, G. Hinton, A. Krizhevskiy, I. Sutskever, R. Salakhutdinov Dropout: A simple way to prevent neural networks from overfitting. J. of Machine Learning Research 15 (2014) 1929-1958



# Flattening layer



# Convolutional Neural Networks



# Example Time!

Python notebook and flows in

IBM Watson Studio

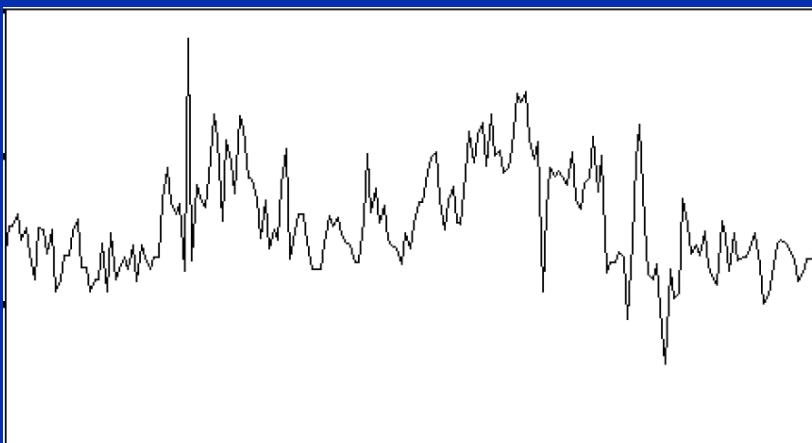


# Recurrent Neural Networks

# Recurrent Neural Networks (RNNs)

Some types of data inherently contain sequential aspects, such as:

Time series data



Text data

Great acting! Amazing special effects!  
Best movie I've seen in a long time.

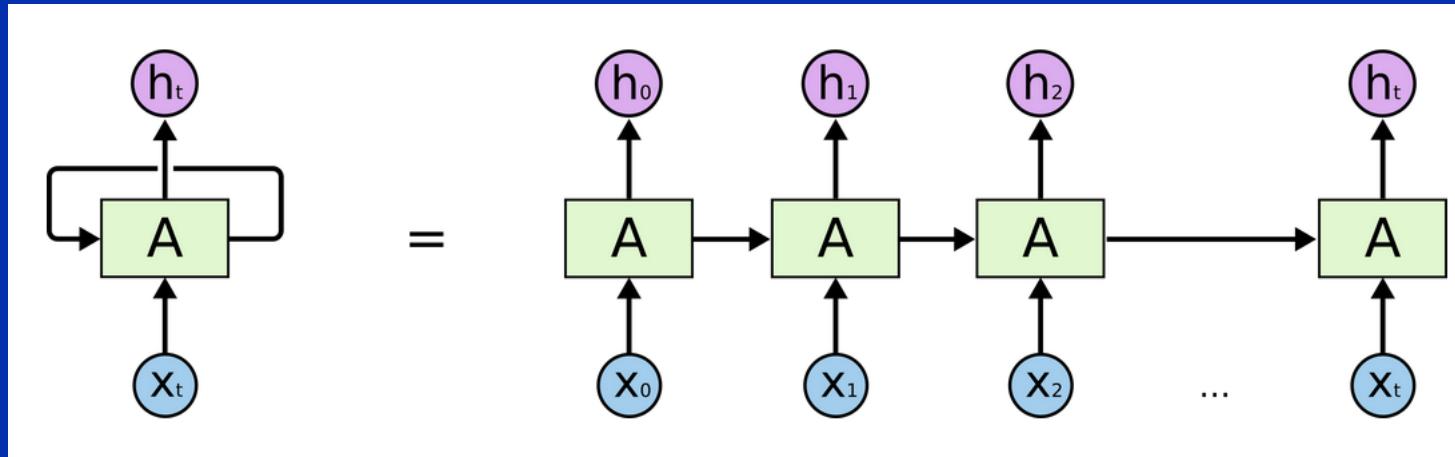
Who let that guy get hold of a camera?  
What a mess! I could have had a nice nap  
if the noises weren't so irritating.

# Recurrent Neural Networks (RNNs)

RNN structures typically include the following types of layers:

- Embedding (if analyzing text data)
- Recurrent
- Fully connected or dense

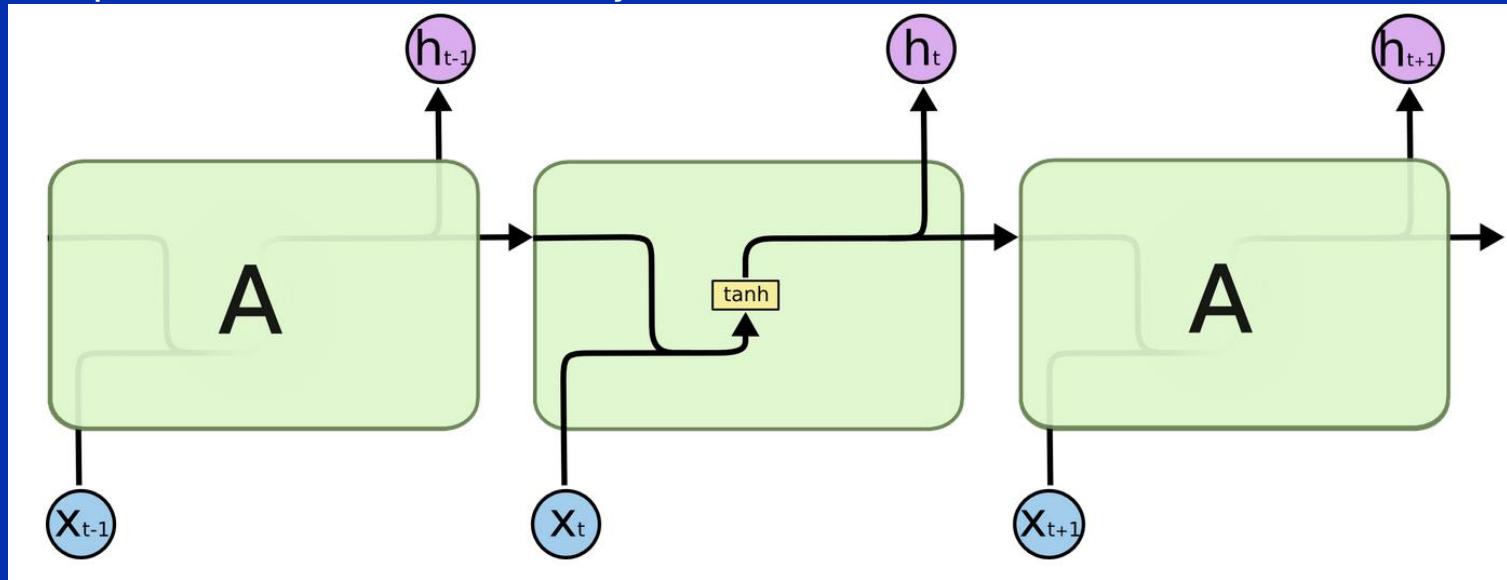
# Recurrent Neural Networks (RNNs)



<https://colah.github.io/posts/2015-08-Understanding-LSTMs/>

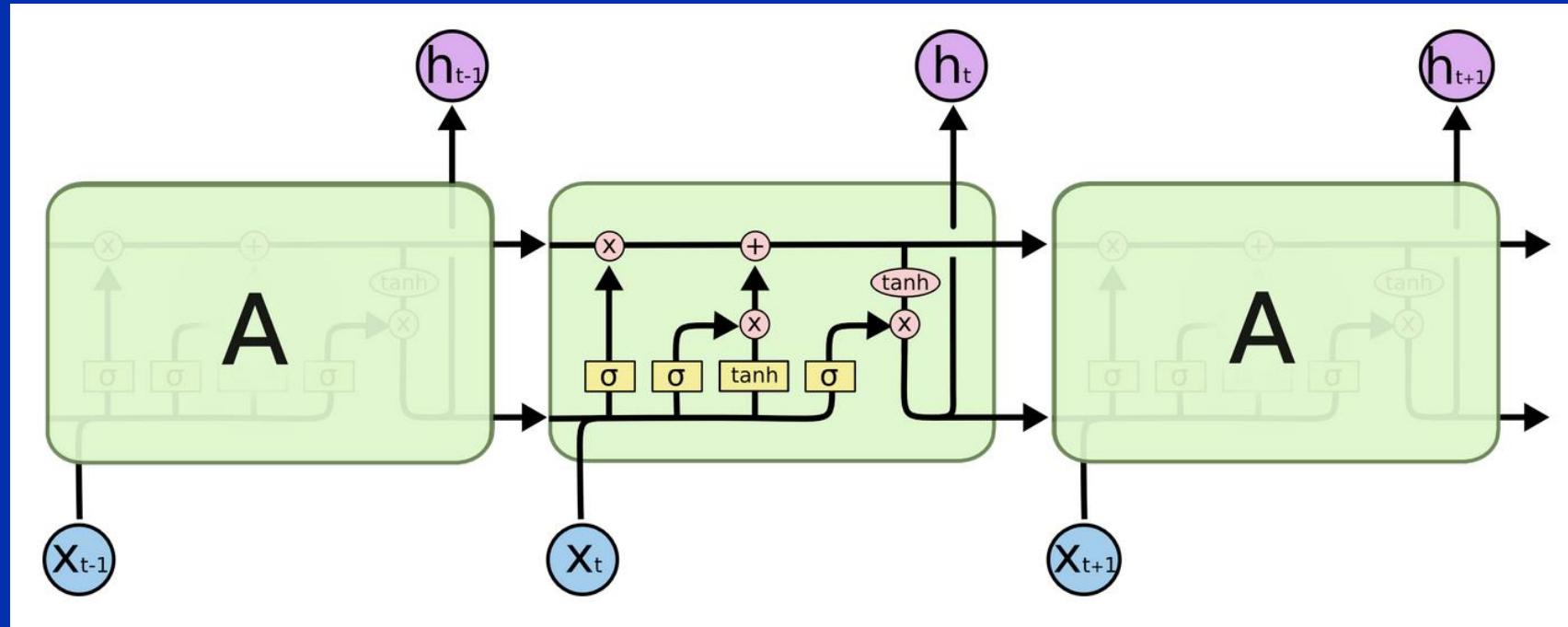
# Recurrent Neural Networks (RNNs)

Simple recurrent network layer structure



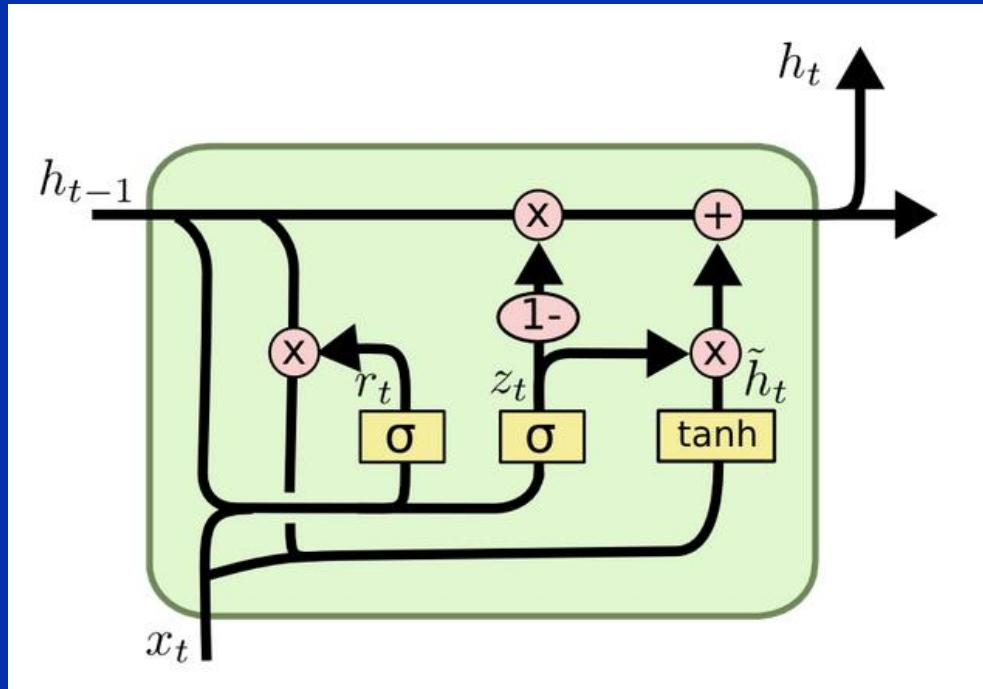
<https://colah.github.io/posts/2015-08-Understanding-LSTMs/>

# Long Short-Term Memory (LSTM) RNNs



<https://colah.github.io/posts/2015-08-Understanding-LSTMs/>

# Gated Recurrent Unit (GRU) RNNs



<https://colah.github.io/posts/2015-08-Understanding-LSTMs/>

# Example Time!

RStudio in IBM Watson Studio

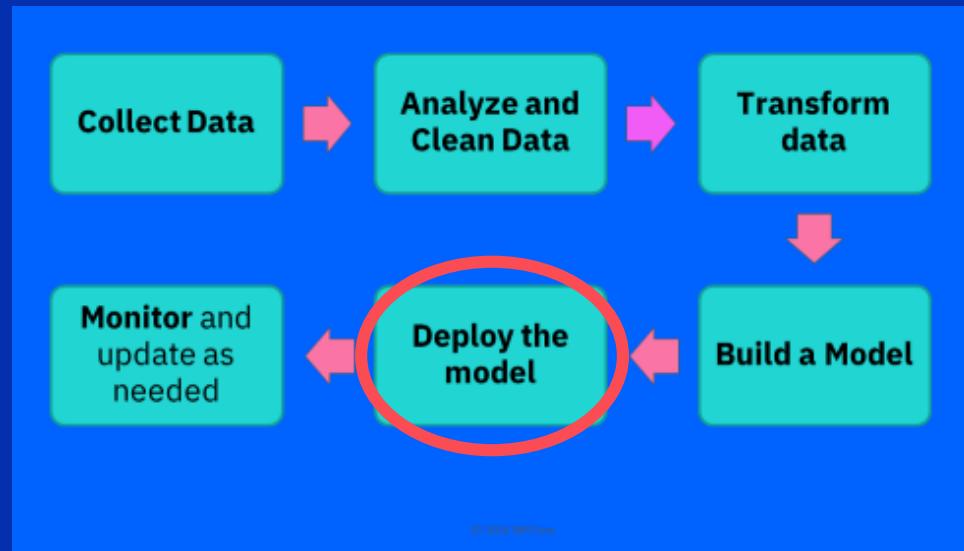
# Model deployment

# Model deployment

## Challenges:

- Different Teams
- Different environments
- Need to keep data prep steps

Solutions: open standards



**PMMI**: Predictive Model Markup Language – XML based, >20 years

**PFA**: Portable Format for Analytics, JSON based, ~5 years

**ONNX**: Open Neural Network eXchange, Protobuf, < 2 years

Or Docker containers

# What are PMML and PFA?



XML and JSON for model exchange and deployment

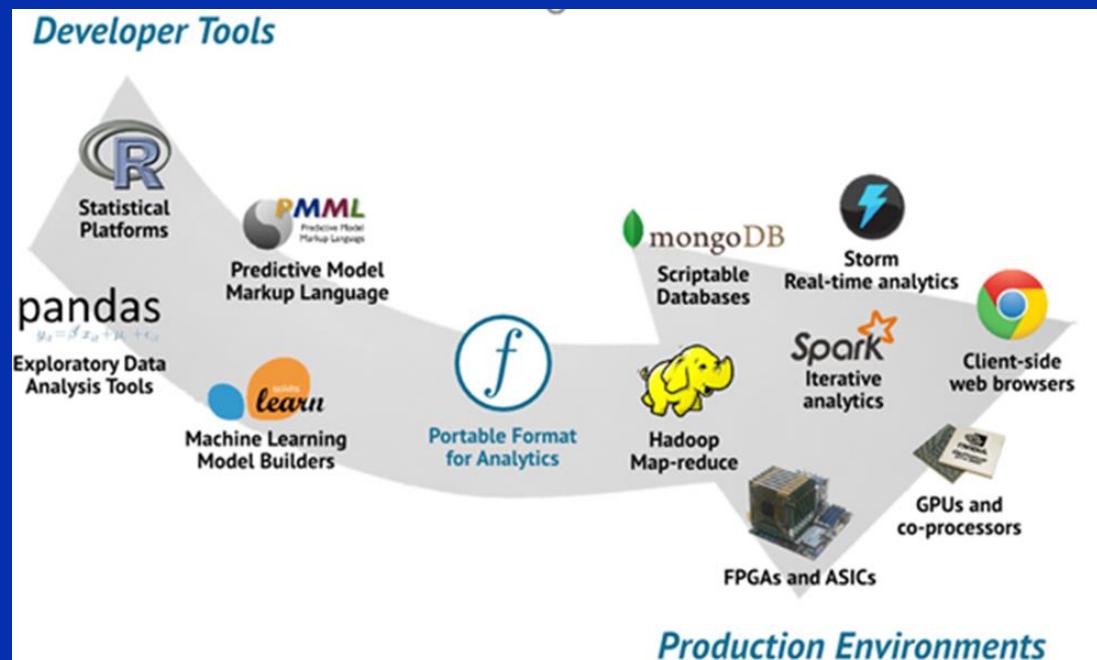
PMML 0.7 in 1997, 1.1 in 2000, 4.4 in 2019

Used by >30 companies  
and OS packages

[dmg.org/pmml](http://dmg.org/pmml)

PFA – small math  
programming language

[dmg.org/pfa](http://dmg.org/pfa)



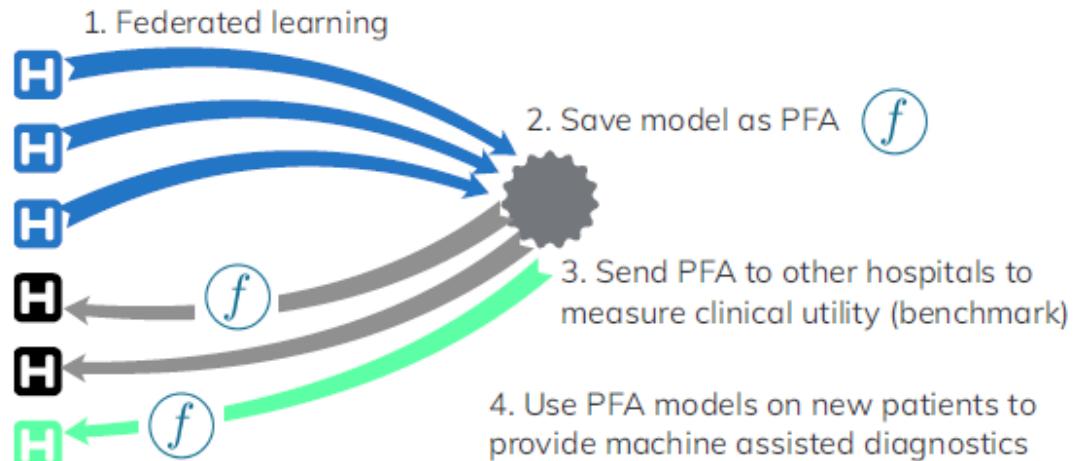
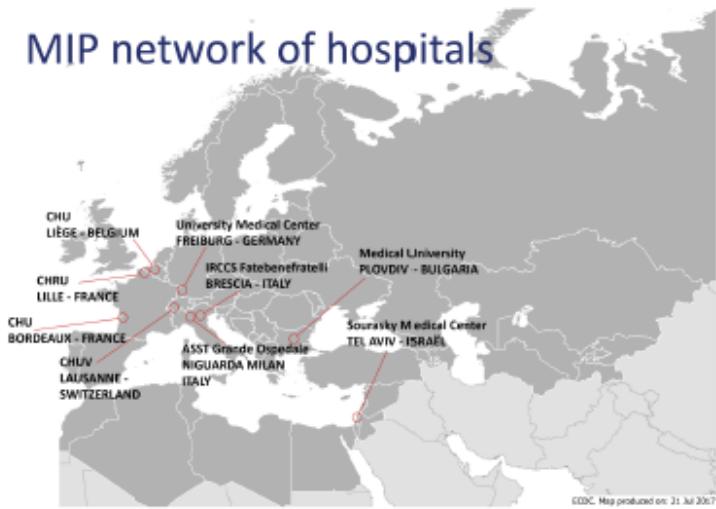
# Use of PMML and PFA in medical applications

## Human Brain Project

Ludovic Claude,  
CHUV  
Lausanne,  
Switzerland



### MIP network of hospitals



# ONNX: Open Neural Network eXchange



Since 2017

Protobuf

Covers DL and traditional ML

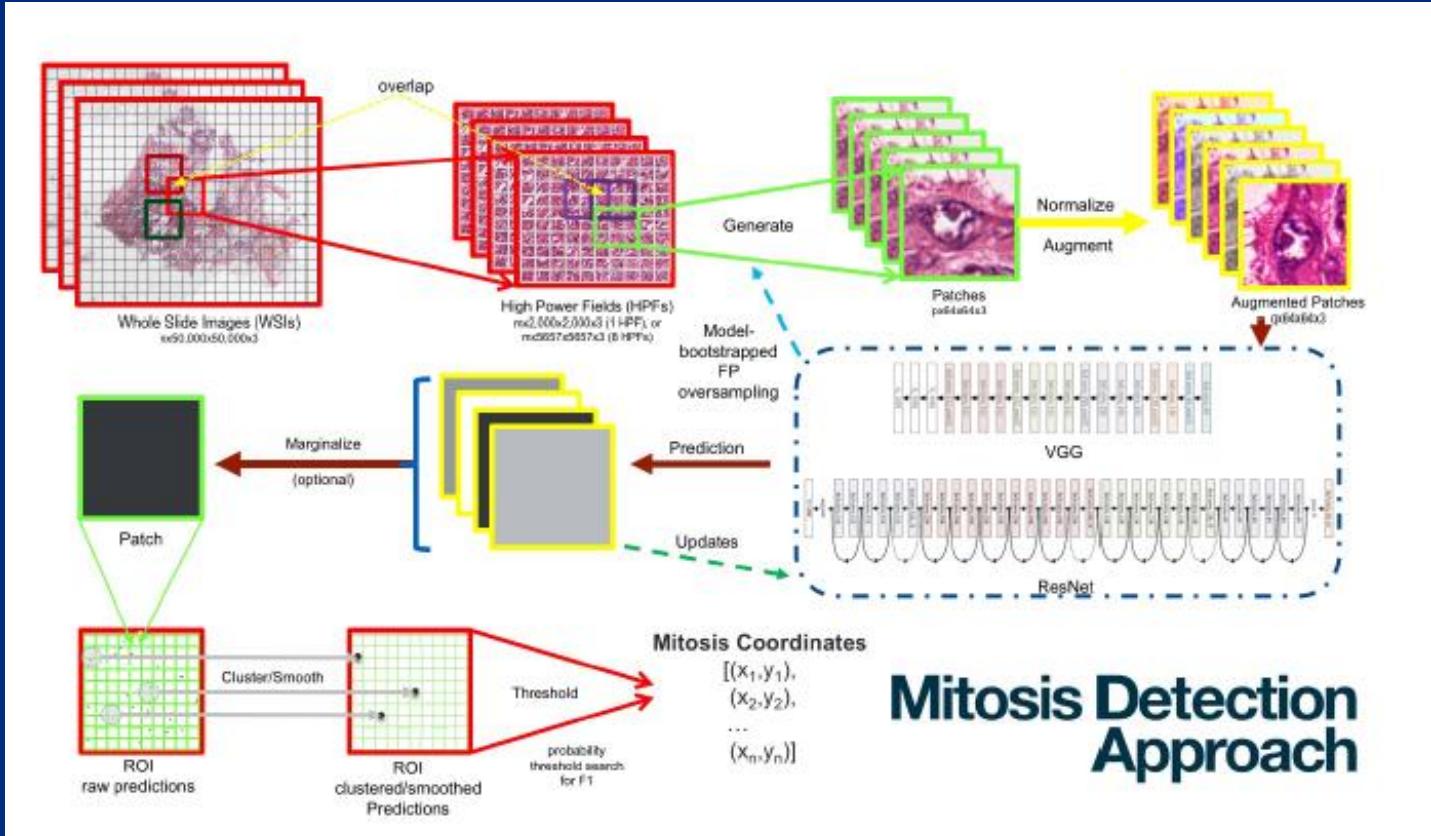
Active work by many companies



# Using ONNX in medical image processing: potential applications

MAX

[ibm.biz/  
model-exchange](http://ibm.biz/model-exchange)



## Mitosis Detection Approach

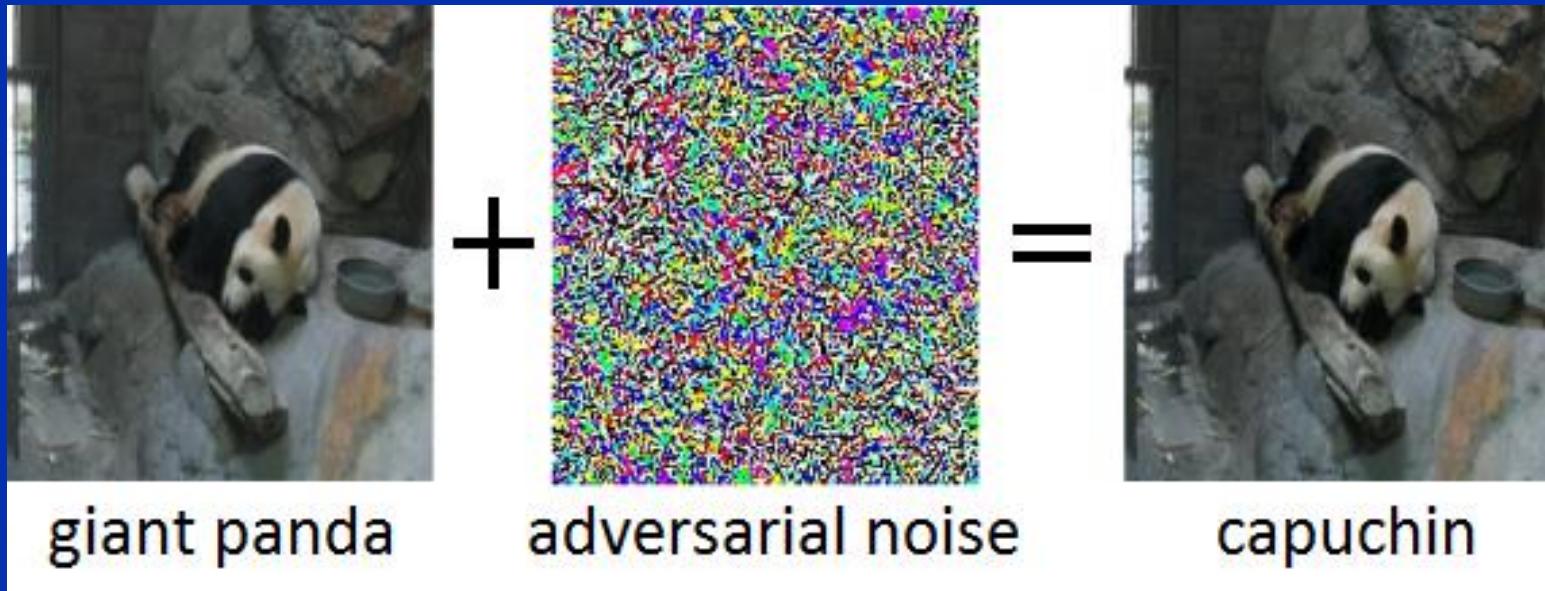
# Adversarial

# Robustness

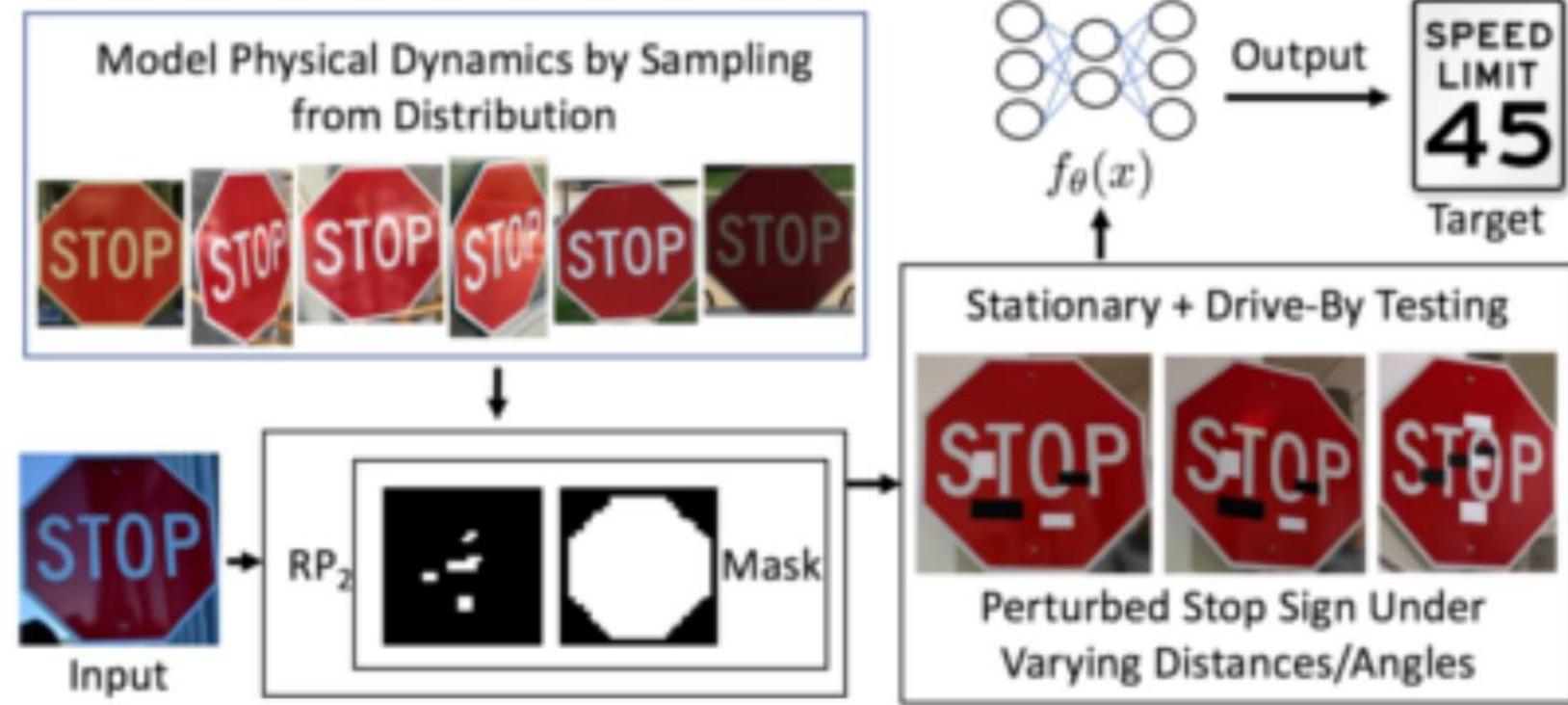
# Deep learning and adversarial attacks

Deep Learning models are now used in many areas

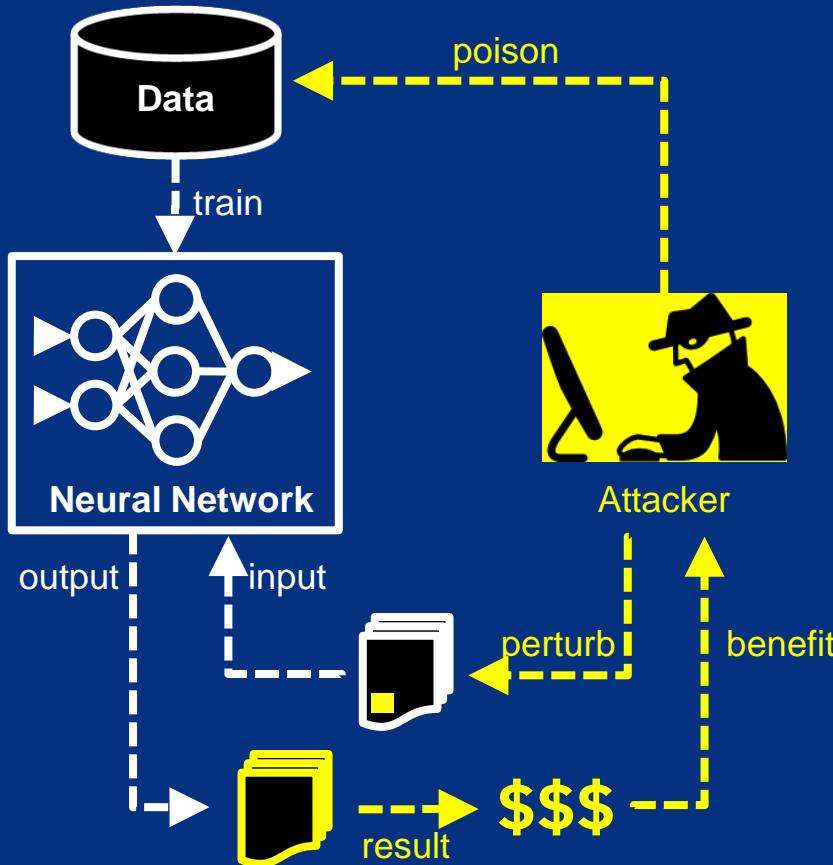
Can we trust them?



## Scarier example



# Adversarial Threats to AI



## Evasion attacks

- Performed at test time
- Perturb inputs with crafted noise
- Model fails to predict correctly
- Undetectable by humans

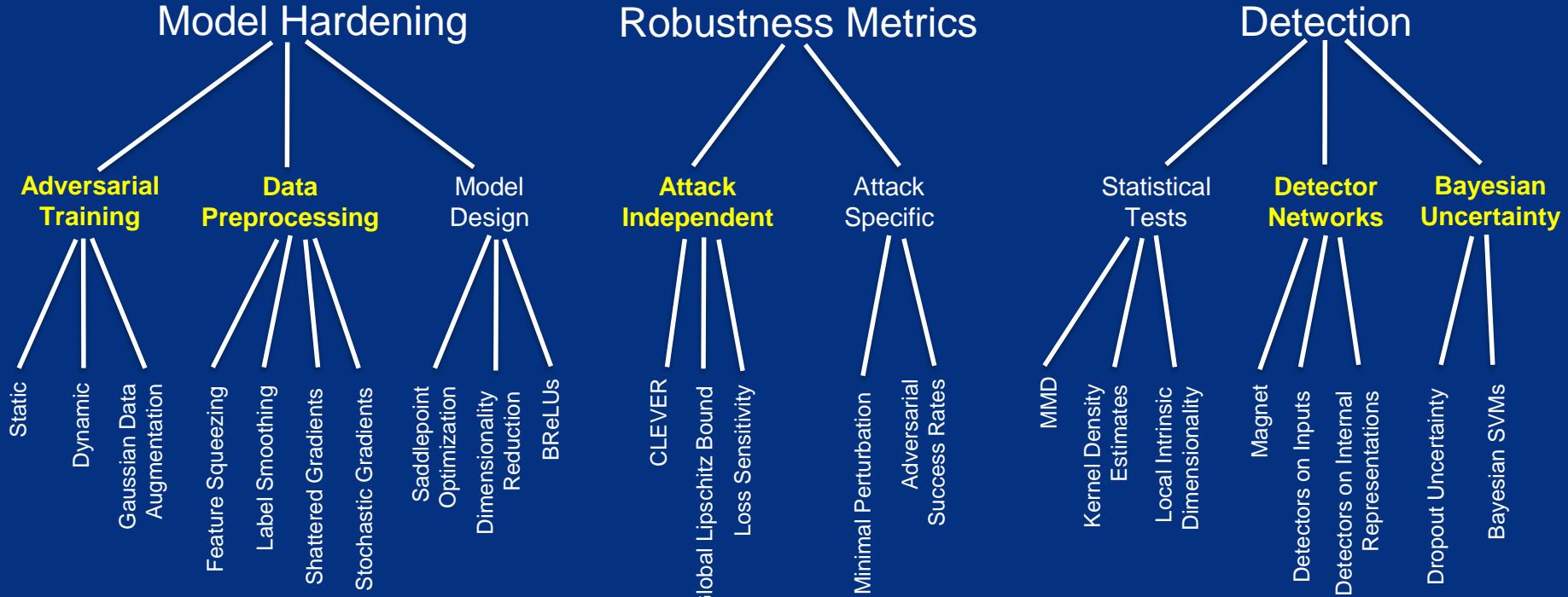


## Poisoning attacks

- Performed at training time
- Insert poisoned sample in training data
- Use backdoor later



# How to defend? Taxonomy of defenses



# The Adversarial Robustness Toolbox (ART)

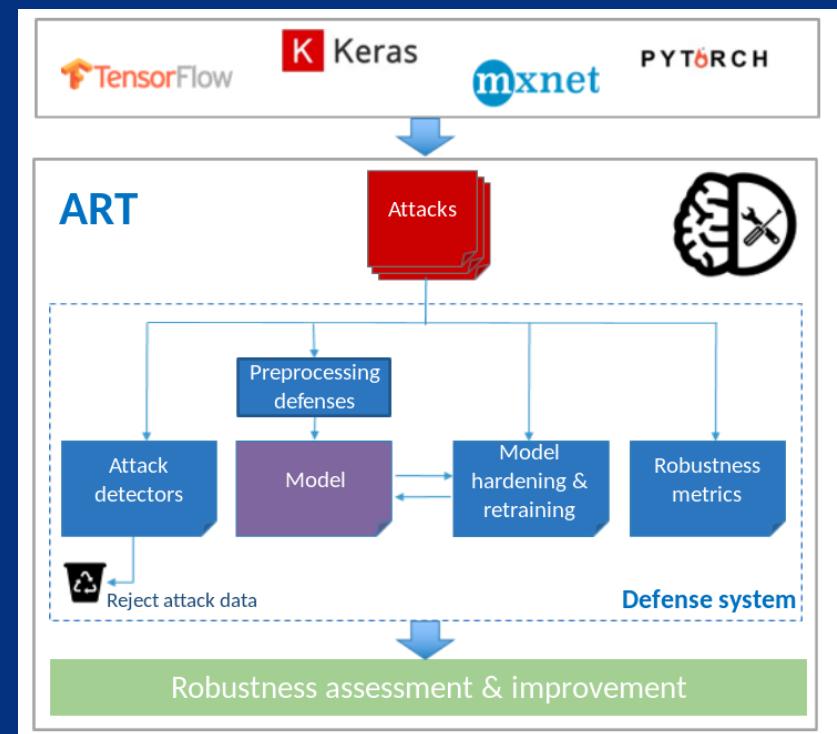
- Library for **adversarial machine learning**
- Baseline implementation of attacks and defenses for classifiers
- Dedicated to images
- MIT license
- Supported frameworks:

 TensorFlow

 PYTORCH

K Keras

 mxnet



# ART Demo: <https://art-demo.mybluemix.net/>

Try it out

1. Select an image to target



2. Simulate Attack

Adversarial noise type  
C&W Attack

Determine strength  
None low med high

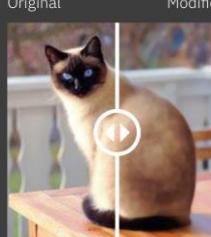
3. Defend attack

Gaussian Noise  
None low med high

Spatial Smoothing  
None low med high

Feature Squeezing  
None low med high

Visual | Code



Original      Modified



94%

Siamese cat

# AI Fairness 360

# Are computer-generated results free of bias?

<https://www.propublica.org/article/machine-bias-risk-assessments-in-criminal-sentencing>

## Prediction Fails Differently for Black Defendants

	WHITE	AFRICAN AMERICAN
Labeled Higher Risk, But Didn't Re-Offend	23.5%	44.9%
Labeled Lower Risk, Yet Did Re-Offend	47.7%	28.0%

*Overall, Northpointe's assessment tool correctly predicts recidivism 61 percent of the time. But blacks are almost twice as likely as whites to be labeled a higher risk but not actually re-offend. It makes the opposite mistake among whites: They are much more likely than blacks to be labeled lower risk but go on to commit other crimes. (Source: ProPublica analysis of data from Broward County, Fla.)*

# AIFairness 360

Open source Python library

70+ Fairness metrics and explanations

10 Bias mitigation algorithms

# Demo Application: AI Fairness 360 Web Application

<http://aif360.mybluemix.net/>

IBM Research Trusted AI

Home **Demo** Resources Community

## AI Fairness 360 - Demo



Data    Check    Mitigate    Compare

### 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 contains attributes that should be protected to avoid bias.

**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](#)

**German credit scoring**  
Predict an individual's credit risk.  
Protected Attributes:  
- Sex, privileged: *Male*, unprivileged: *Female*  
- Age, privileged: *Old*, unprivileged: *Young*  
[Learn more](#)

**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](#)

## 2. Check bias metrics

Dataset: Adult census income

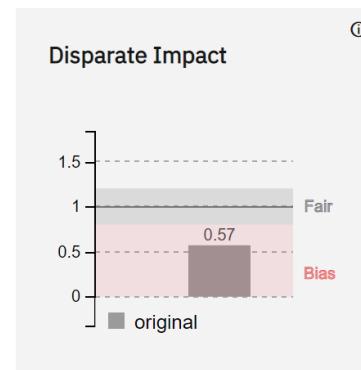
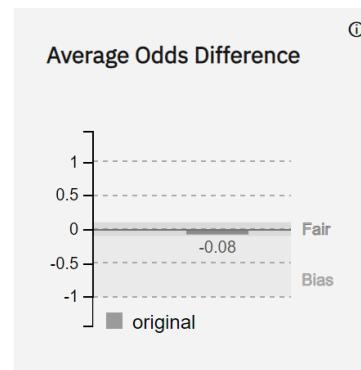
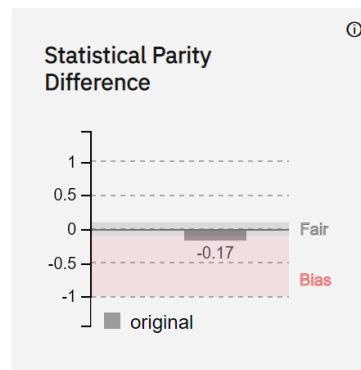
Mitigation: none

### Protected Attribute: Race

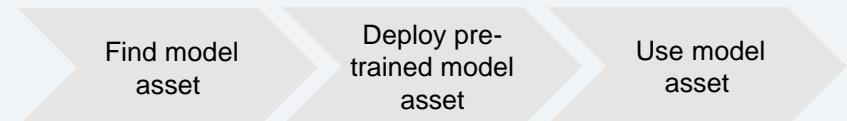
Privileged Group: *White*, Unprivileged Group: *Non-white*

Accuracy with no mitigation applied is 82%

With default thresholds, bias against unprivileged group detected in 2 out of 5 metrics



# MAX: Reduces “time to value” for developers



Free, deployable, and trainable code.

A place for developers to find and use free and open source deep learning models.

[View all models >](#) [Try the tutorial >](#) [Join the community >](#)

Deployable   Facial Recognition	Deployable   Object Detection In Images	Object Detector
<b>Facial Emotion Classifier</b>  Detect faces in an image and predict the emotional state of each person  <a href="#">View model »</a>	<b>Image Segmente</b>  Identify objects in an image, additionally assigning each pixel of the image to a particular object.  <a href="#">View model »</a>	<b>Object Detector</b>  Localize and identify multiple objects in a single image.  <a href="#">View model »</a>

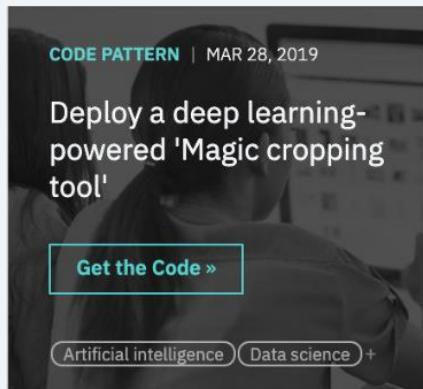
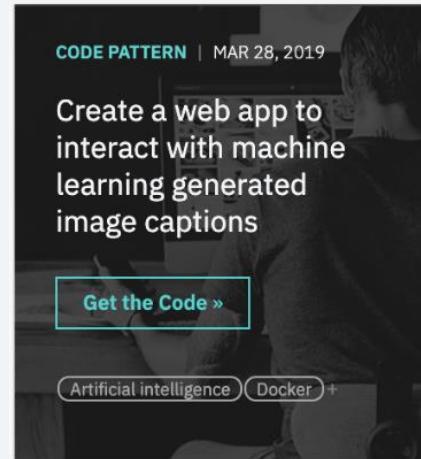
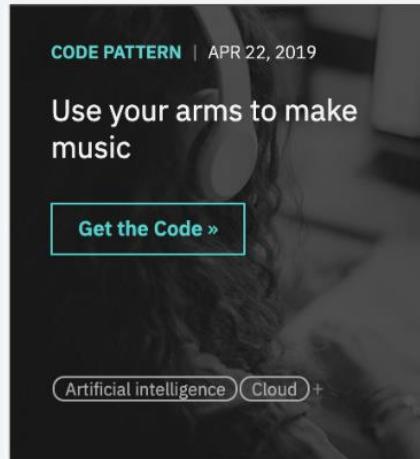
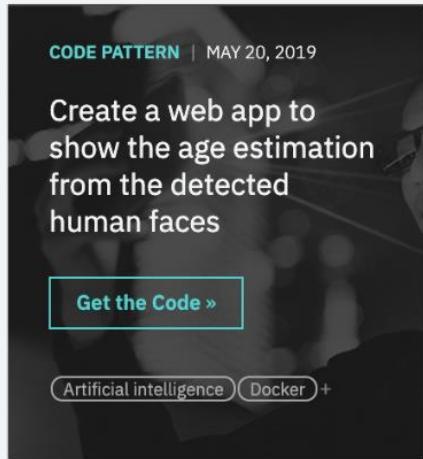
Artificial intelligence Deep learning +

Artificial intelligence Deep learning +

[ibm.biz/model-exchange](http://ibm.biz/model-exchange)

- Audio classification
- Image classification
- Text classification
- Object detection
- Facial recognition
- Image-to-image translation
- Image-to-text translation
- Named entity recognition
- Text feature extraction
- ...

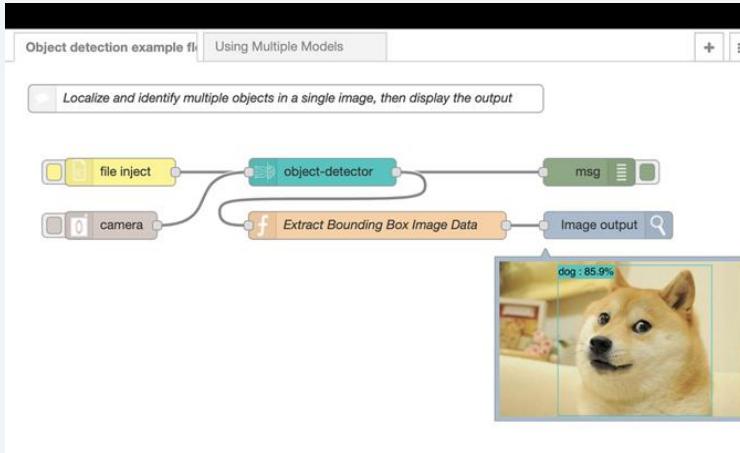
# ↳ Deep Learning Code Patterns



<https://developer.ibm.com/patterns/category/model-asset-exchange/>

# MAX Consumption scenarios

- Model-serving microservice (Docker-based)
- Internet of Things: Node-RED
- JavaScript/Node.js packages

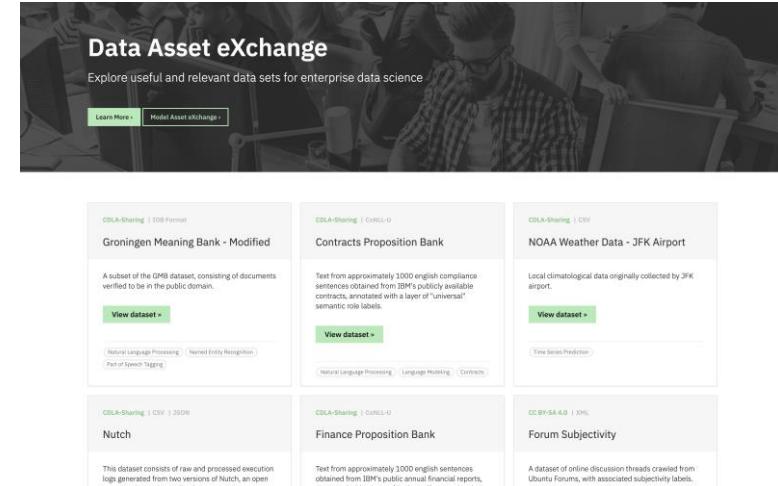


A screenshot of a web-based API interface for the MAX Object Detector. The top navigation bar says "model Model information and inference operations". It includes three main sections: "GET /model/labels", "GET /model/metadata", and "POST /model/predict". The "Parameters" section shows an "image" input field with a placeholder "(formData)". The "Responses" section shows a preview of the "MAX Object Detector" interface, which displays a Toy Story scene with two characters. Bounding boxes are drawn around them, with one character labeled "dog : 85.9%". On the right side of the interface, there are buttons for "Upload an image", "Choose File", "Submit", "Filter detected objects", "Probability Threshold", and "Labels Found". A "Cancel" button is also present.

# IBM Data Asset eXchange (DAX)

- Curated free and open datasets under open data licenses
- Standardized dataset formats and metadata
- Ready for use in enterprise AI applications
- Complement to the Model Asset eXchange (MAX)

Data Asset eXchange  
[ibm.biz/data-asset-exchange](http://ibm.biz/data-asset-exchange)



The screenshot shows the IBM Data Asset eXchange homepage. At the top, there's a banner with the text "Data Asset eXchange" and "Explore useful and relevant data sets for enterprise data science". Below the banner, there are two main buttons: "Learn More" and "Model Asset eXchange". The main content area features several cards, each representing a dataset:

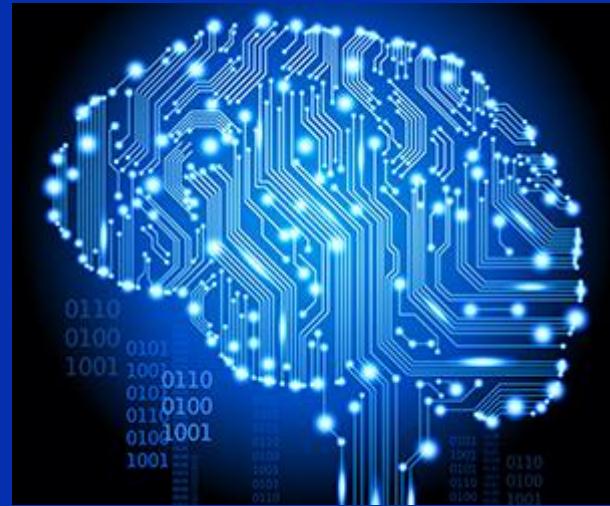
- Groningen Meaning Bank - Modified**: A subset of the GMB dataset, consisting of documents verified to be in the public domain. It includes tags for Natural Language Processing, Named Entity Recognition, Part of Speech Tagging, and more.
- Contracts Proposition Bank**: Text from approximately 1200 original contracts obtained from IBM's publicly available contracts, annotated with a layer of "universal" semantic role labels.
- NOAA Weather Data - JFK Airport**: Local climatological data originally collected by JFK airport. It includes a "Time Series Prediction" section.
- Nutch**: Text from approximately 1200 original sentences obtained from IBM's public annual financial reports, annotated with a layer of "universal" semantic role labels.
- Finance Proposition Bank**: A dataset of online discussion threads crawled from Ubuntu Forums, with associated subjectivity labels.
- Forum Subjectivity**: A dataset of online discussion threads crawled from Ubuntu Forums, with associated subjectivity labels.

# Conclusions

Deep neural networks are used widely

IBM Watson Studio provides tools for model building, deployment, monitoring, as well as detection and remediation of bias and adversarial attacks

MAX models are ready to use in a variety of applications



# Learn more on deep learning

Watson Studio: sign up for IBM Cloud: <https://ibm.biz/Bdz4N5>

Codait.org   Onnx.ai   @SvetaLevitam   @dpnichols

AIFairness 360: <https://github.com/IBM/AIF360>

ART: <https://github.com/IBM/adversarial-robustness-toolbox>

MAX: [ibm.biz/model-exchange](https://ibm.biz/model-exchange)