Instability of Convolutional Neural Networks

Graduation Project Supervised by

Professor. Douglas Pickering

By

Ahmed Ech-Cherif

Agenda

- **Introduction**
- 1.1- Problem statement: COVID-19
- 1.2- Machine Learning: Kernel Methods, Deep Learning, etc.
- 1.3- Instability of Machine Learning Algorithms
- **Machine Learning Training, Accuracy and Stability**
- 2.1- Data Driven and Model Driven ML
- 2.2- Existing Frameworks: Pytorch and Tensorflow
- **Pytorch Implementation of Data Driven ML**
- **Instability of Neural Network**
- - Deepfool (adversarial) Attacks Illustration : (Face mask detectors not pretending the correct label.)

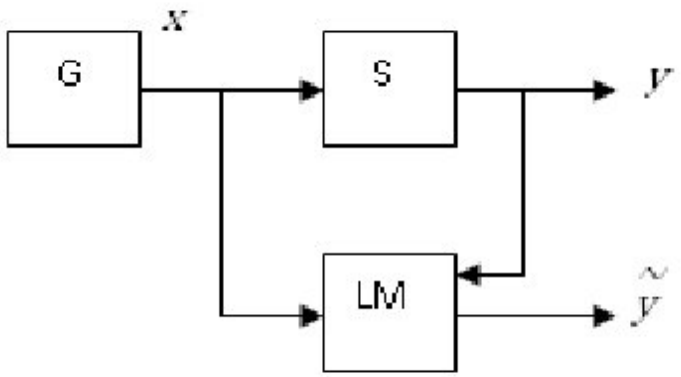
Problem statement: COVID-19

- Medical studies have shown that wearing a face mask reduces the risk of contagion and thus, prevents the pandemics to spread
- Mask must be worn correctly, i.e must cover the nose and the mouth
- In crowded places such as shopping malls, airports, class rooms, etc. It is hard to keep track of people wearing face masks correctly
- Leveraging machine learning to recognize masked faces and unmasked faces may provide a solution to the problem by using a computerized system equipped with surveillance cameras which triggers an alert when observing an unmasked face

Machine Learning for image recognition

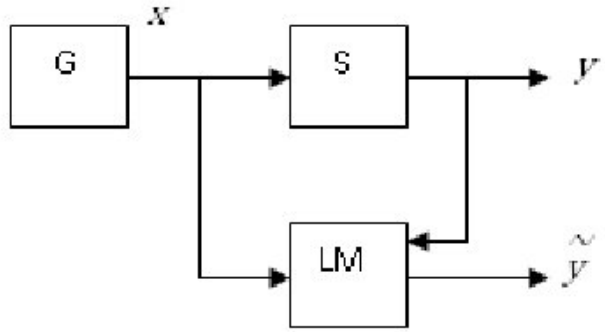
- Definition: “A Machine Learning algorithm is an algorithm whose performance with respect to a task improves with data “. This term has been coined by Tom Mitchell < include classification models>

Supervised Learning



This figure illustrates Supervised Learning. A generator G generates images X (masked & unmasked faces) from an unknown probability distribution, then the images are labeled by a supervisor S (human) and the pair (X,y) is fed to the learning machine LM. After feeding the machine with enough data, the machine constructs a prediction model $\tilde{y} = F_w$ (X) which predicts the label $\tilde{y}$ (masked/unmasked) for any input image X

Supervised Learning



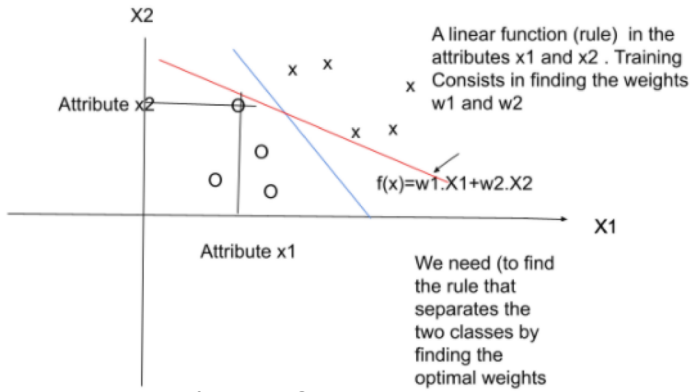


Figure 2

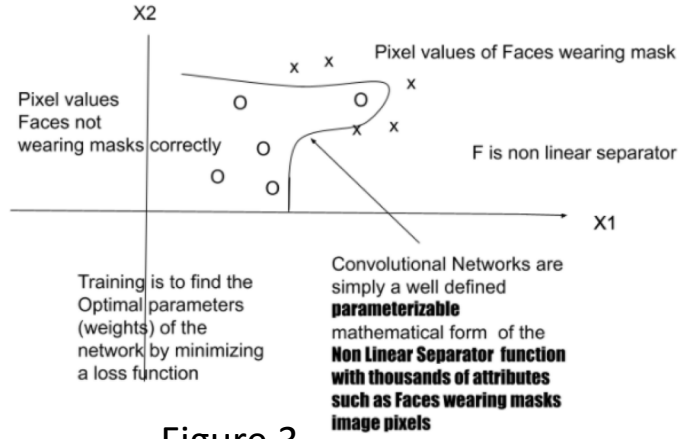
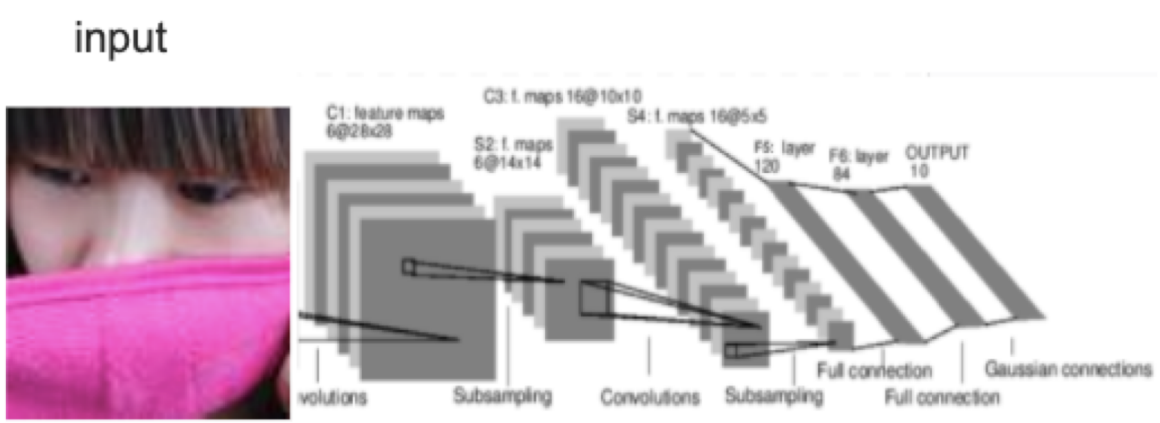


Figure 3

- In order for supervised learning to provide high accuracy it must be fed with a massive training data set $\{(X_1, y_1)\ , (X_2, y_2), \ldots (X_m, y_m)\}$ i.e Data hungry
- Training consists of computing the model $\tilde{y} = F_w$ (X) weights w for a given form (architecture) of the model
- In order to compute the weights w we need to define a loss function L_w ($\tilde{y}\ = F_w$(X), y) which measures the sum of the differences between the actual and predicted outputs. This process requires the minimization of the loss function
- For linear models (Figure 2) $\tilde{y} = w^T X + \beta$ the parameters can be computed by the least squared method (convex optimization)
- For non-linear models (Figure 3), the computation of the parameter depends on the form of the model (non-convex function) which requires the use of the stochastic gradient descent by tuning some hyper parameters, step length etc.
- Several non-linear models F_w (X) have been investigated in the last few decades including kernel methods, neural networks multi layer perceptron (MLP), convolutional neural networks etc.

Convolutional neural networks



INPUT IMAGE

18	54	51	239	244	188
55	121	75	78	95	88
35	24	204	113	109	221
3	154	104	235	25	130
15	253	225	159	78	233
68	85	180	214	245	0

WEIGHT

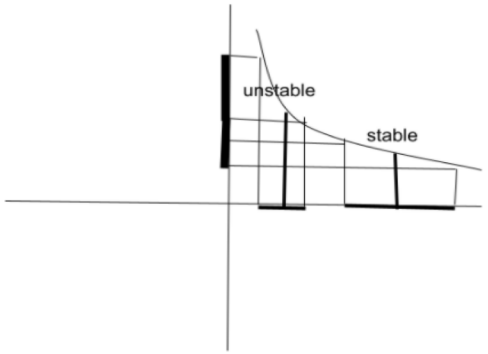
1	0	1
0	1	0
1	0	1

429

CNN starts with an input image RGB image captured by a camera and pre-processed by a face detection algorithm first, then mapped by a series of non-linear transformations called convolutional layers followed by max pooling transformation and the last layers are fully connected layers which gives the output value (label)

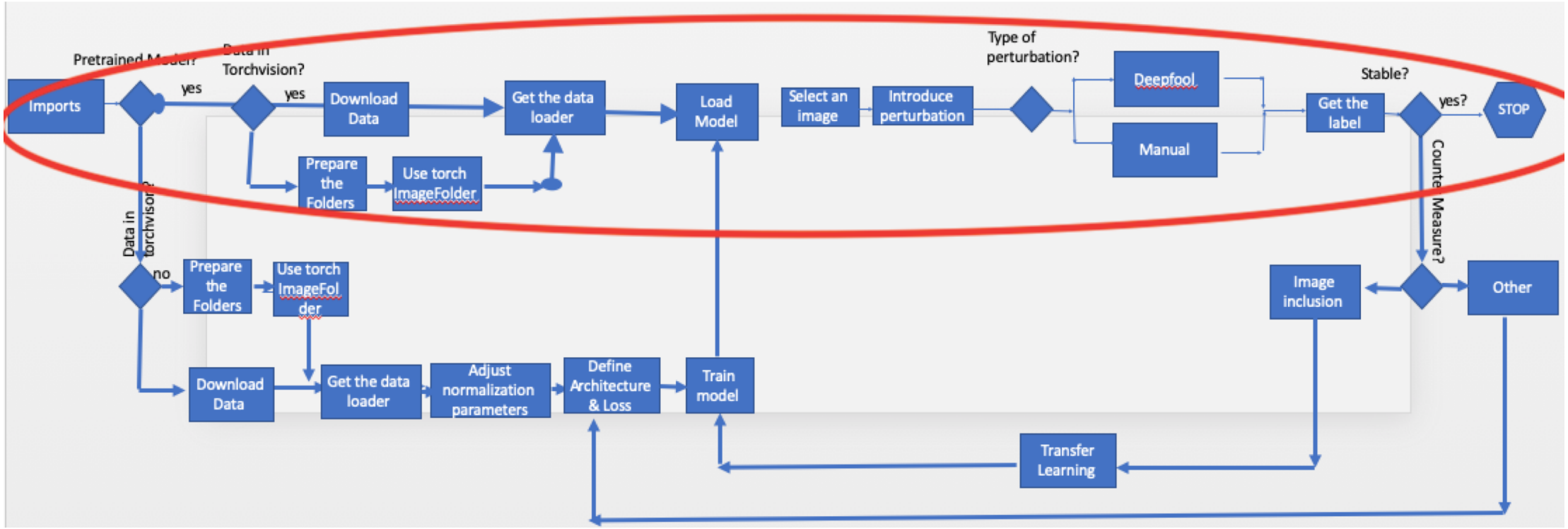
Accuracy vs Stability of CNN

- CNN have achieved impressive accuracy on several data sets such as google imageNet, CIFAR10 etc. However, they have been found to be very unstable to small perturbations to the input image
- What is stability (/instability)?:
- A model F_w (X) is set to be stable (Fault Tolerant) if the output F_w (X) for a given input X is close to F_w ($\tilde{X}$) whenever X is close to $\tilde{X}$ i.e $\tilde{X}$ is obtained by perturbing the input X



This is an illustration of the stability concept on the hyperbola F(X) = 1/X. stable on the right where X is large and instable on the left where X is near the origin
This has to do with derivative of the function, if the derivative is small then we have stability and if the derivative is large then the model is instable

Experimental pipeline for testing stability of CNN



DeepFool — A simple and accurate method to fool Deep Neural Networks.

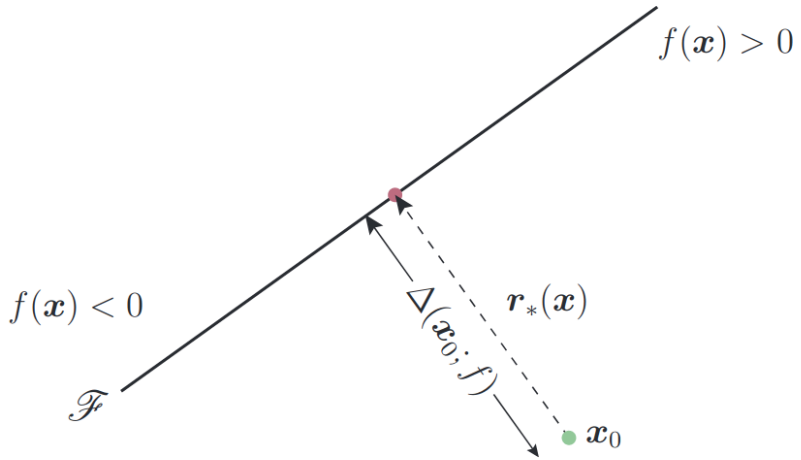


Figure 2: Adversarial examples for a linear binary classifier.

It can be easily seen using a linear binary classifier, that the robustness of the model (`f`) for an input x_{0}is equal to the distance of x_{0} to the hyperparameter plane (which separates the 2 classes).
Minimal perturbation to change the classifier's decision corresponds to the orthogonal projection of x_{0} onto the hyperparameter plane.