

SECURITY WEAKNESSES IN MACHINE LEARNING

DANIEL ETZOLD - @ETZOLDIO

TUESDAY 21ST MAY

NAIVE BAYES FILTER BYPASSED

Naive Bayes filter bypassed. This is a common security weakness in machine learning models. It involves manipulating the input data to cause the model to misclassify or bypass security filters. This can be achieved by carefully selecting features that are not used by the model or by introducing noise that affects the model's decision boundary.



IMAGE CLASSIFIER FOOLED

Image classifier fooled. This occurs when an image classifier is tricked into misclassifying an image. This can be done by adding adversarial perturbations to the image, which are small, carefully chosen changes that cause the model to misclassify. For example, a cat image might be misclassified as a dog.



THIEVES STEAL MODEL PARAMETERS

Thieves steal model parameters. This is a security risk where an attacker can access the internal parameters of a machine learning model. This can be done by exploiting vulnerabilities in the model's architecture or by intercepting data during training or inference. The stolen parameters can then be used to replicate the model or to perform other malicious actions.

$$f(x) = Wx + b$$

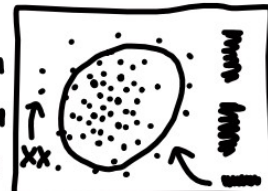
$$= \sum_{i=1}^n w_i x_i + b$$

very quick
on the way
through the
algorithm

can steal the model parameters
through the network. This is a
major security risk as it allows
the attacker to replicate the model
and use it for malicious purposes.

ANOMALY DETECTION HACKED

Anomaly detection hacked. This is a security risk where an attacker can bypass an anomaly detection system. This can be done by introducing adversarial perturbations to the data, which cause the system to misclassify the data as normal.

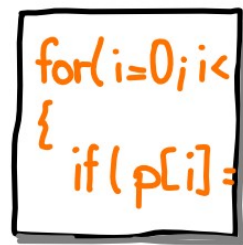




IT SECURITY ARCHITECT



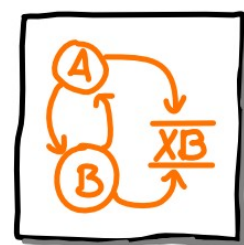
T&1 MAIL&MEDIA DEVELOPMENT & TECHNOLOGY GMBH



CODE REVIEW



DOCUMENTATION REVIEW



THREAT MODELING



IN HOUSE CONSULTING

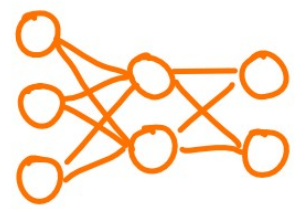


PENETRATION TESTING

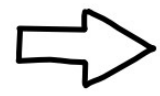


@ETZOLDIO

HTTPS://ETZOLD.IO



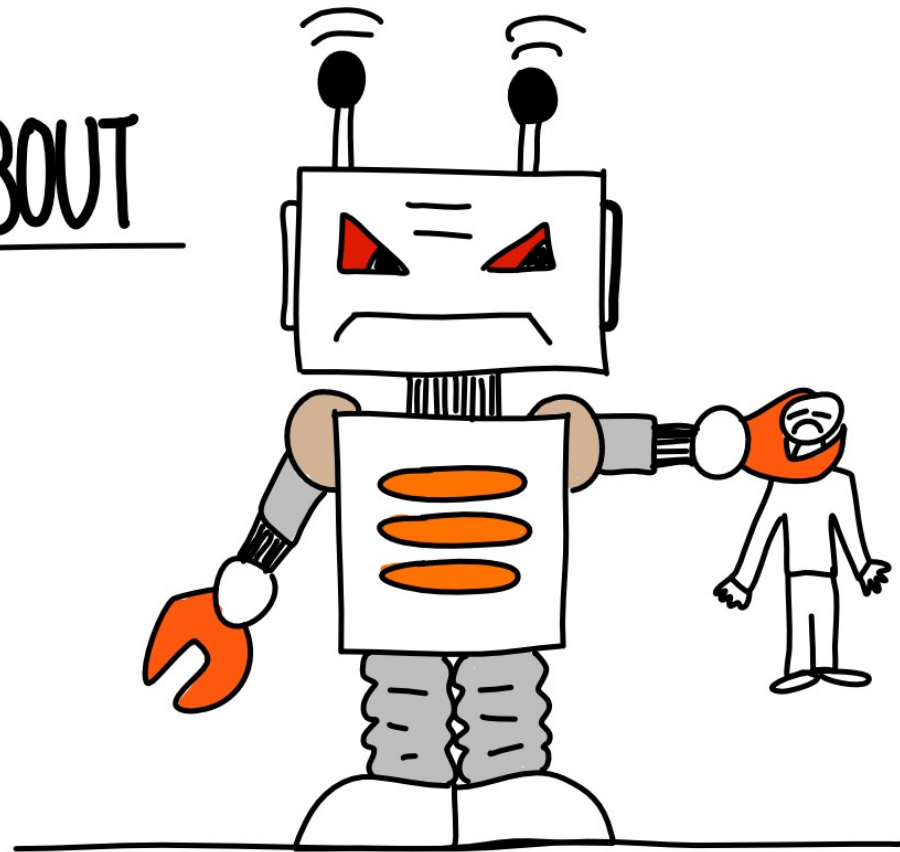
+



SECURITY IN MACHINE LEARNING



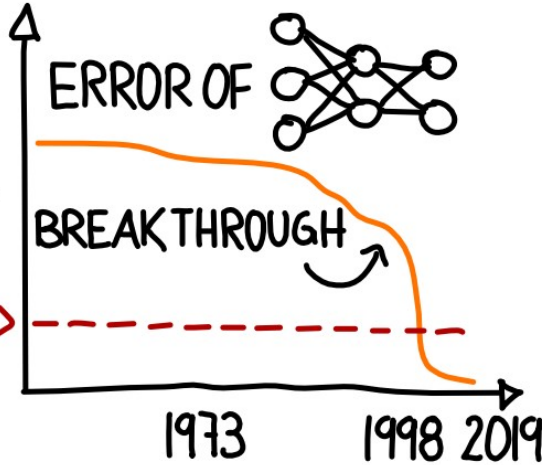
IT'S NOT ABOUT



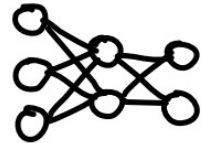
WHY SECURITY IN MACHINE LEARNING MATTERS



REASON



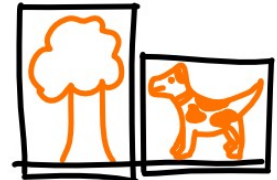
HUMAN-LEVEL PERFORMANCE



BREAKTHROUGH ON



MACHINE TRANSLATION



OBJECT DETECTION

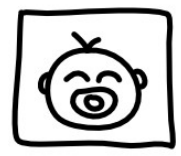


SPEECH RECOGNITION



RATHER OLD

ATTACKS ON MACHINE LEARNING



SOMETIMES EASY

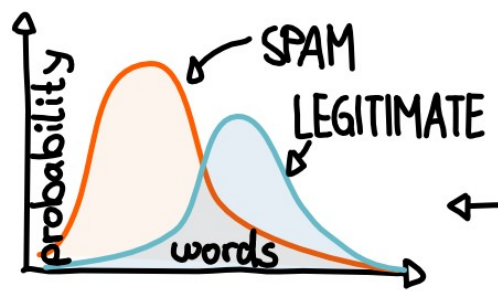


TRAINING:

- FIND USEFUL WORDS
- ESTIMATE SPAM PROBABILITIES

$\hookrightarrow P(W_i=1 | SPAM)$

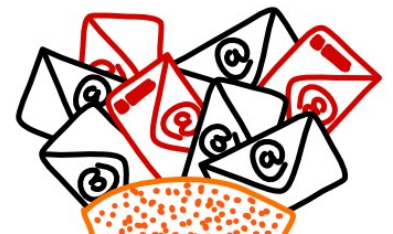
OBSERVATION:  WORD DISTRIBUTION



SPAM FILTERING

CLASSIFICATION:

$$\prod_i P(W_i=1 | SPAM)$$

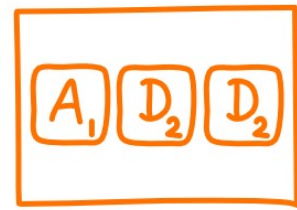


← EXPLOIT

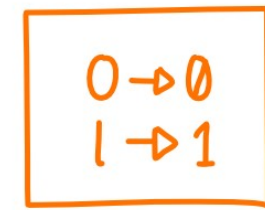
← ATTACKER'S OPTIONS



~20 YEARS ←



GOOD WORDS

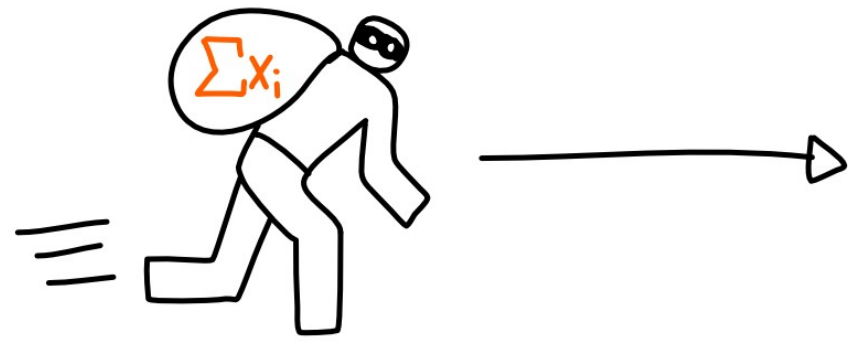


OBFUSSATE



IMAGES

LOGISTIC REGRESSION



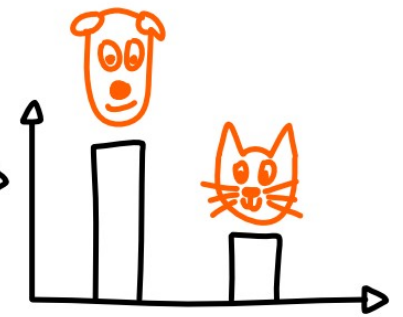
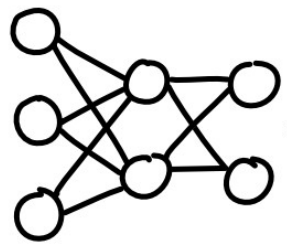
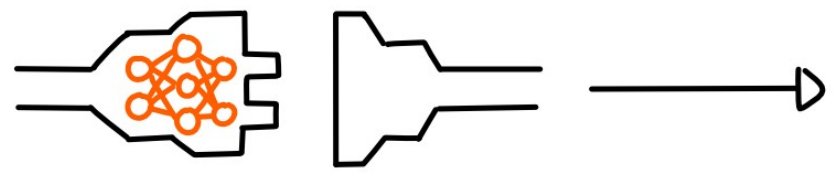
EASIER TO
ANALYZE

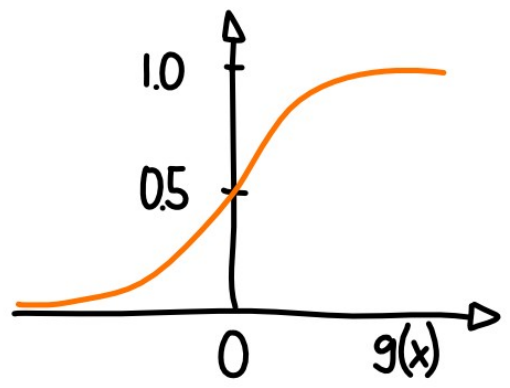
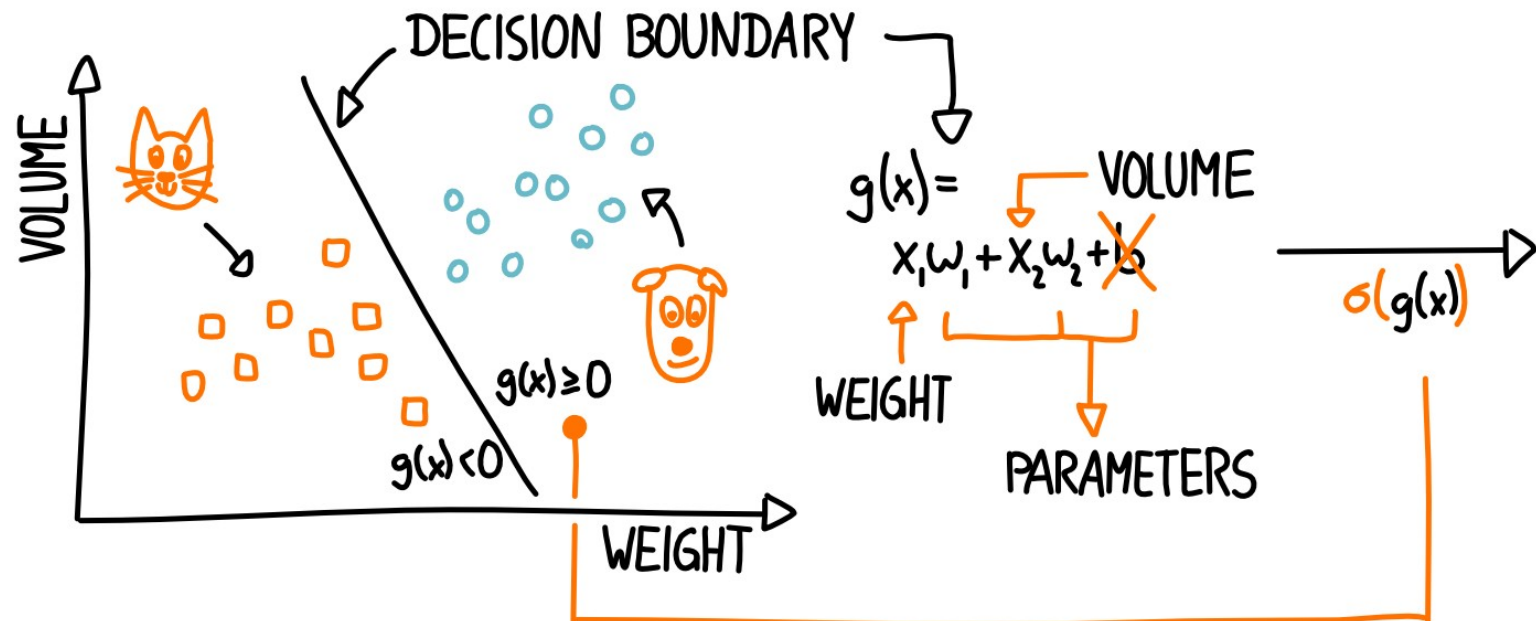


LEAKS
SECRETS



BUSINESS
AT RISK





CLASSIFICATION

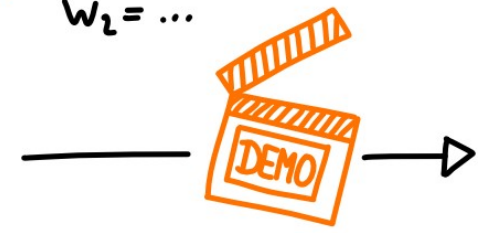
INPUT: (0.3, 0.5) → $\sigma(0.3w_1 + 0.5w_2) = 0.8$ → $0.3w_1 + 0.5w_2 = \sigma^{-1}(0.8)$

INPUT: (0.7, 0.2) → $\sigma(0.7w_1 + 0.2w_2) = 0.4$ → $0.7w_1 + 0.2w_2 = \sigma^{-1}(0.4)$

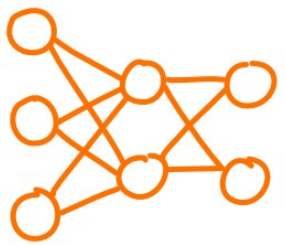
INPUT: (0.5, 0.8) →

BASIC SCHOOL MATH
E.G. ELIMINATION

$w_1 = \dots$
 $w_2 = \dots$



PROBABILITY THAT INPUT BELONGS TO



CAN BE EXTENDED TO NEURAL NETWORKS

FACE FROM TRAINING SET



RECOVERED FACE



TRAINING DATA CAN BE RECOVERED



Fredrikson, et al. „Model inversion attacks that exploit confidence information and basic countermeasures“, 2015

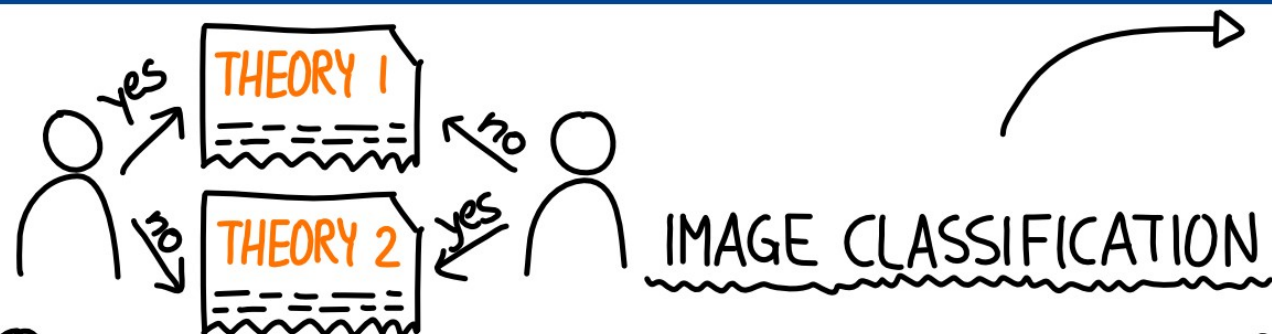
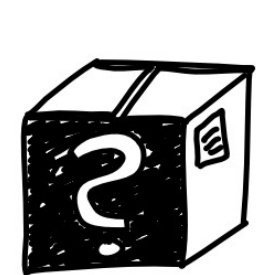
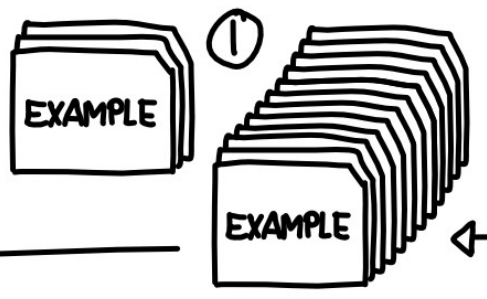
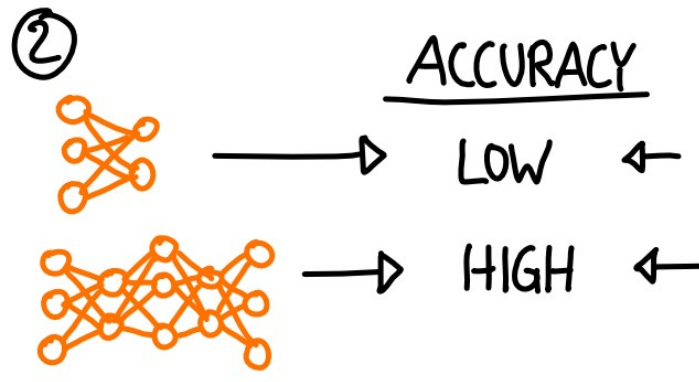
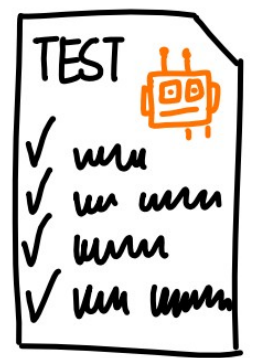
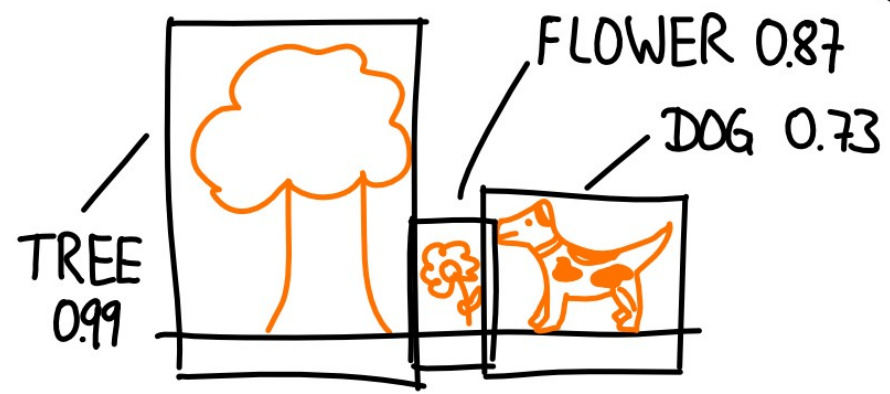
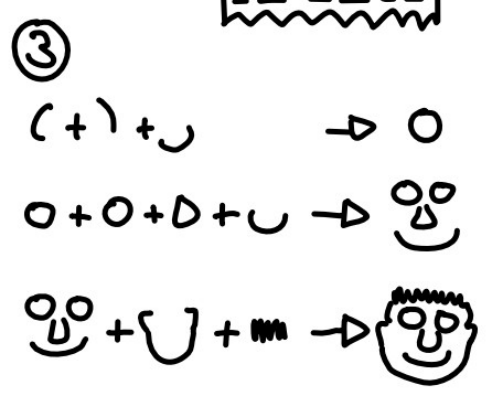


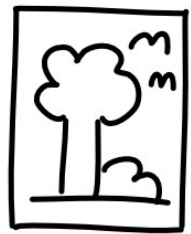
IMAGE CLASSIFICATION

- COMPLEX
- MILLIONS OF PARAMETERS
- HUGE PROGRESS SINCE 200x

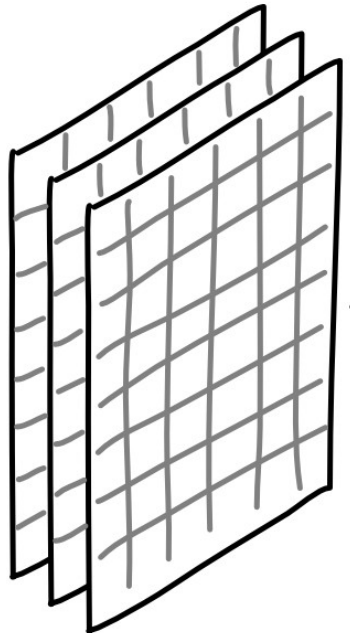


STILL A BLACK BOX

CONVOLUTIONAL NEURAL NETWORKS



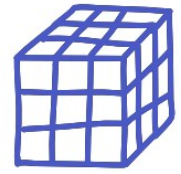
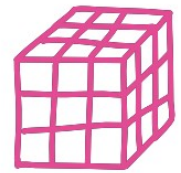
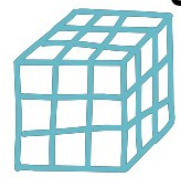
IMAGE



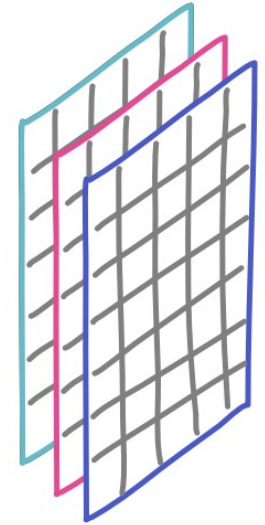
TENSOR

FIRST LAYER

FILTER



FEATURE MAP



NEXT LAYERS

...

OUTPUT

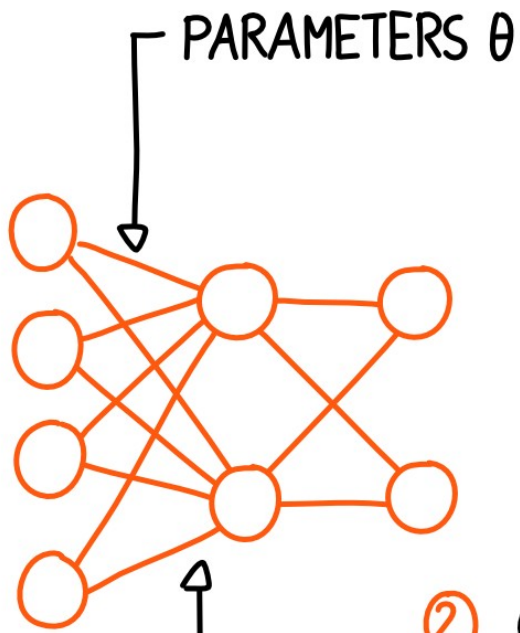
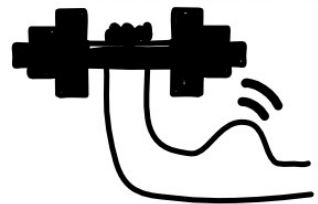
$\begin{bmatrix} 0.2 \\ 0.3 \\ \vdots \\ 0.1 \end{bmatrix}$

SIMPLE FEATURES

COMPLEX FEATURES

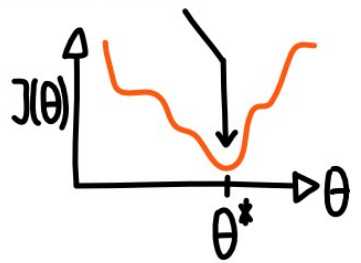


TRAINING OF NETWORKS



$J(\theta)$ = ERROR OF NETWORK

GOAL: FIND θ SUCH THAT $J(\theta)$ IS MINIMIZED

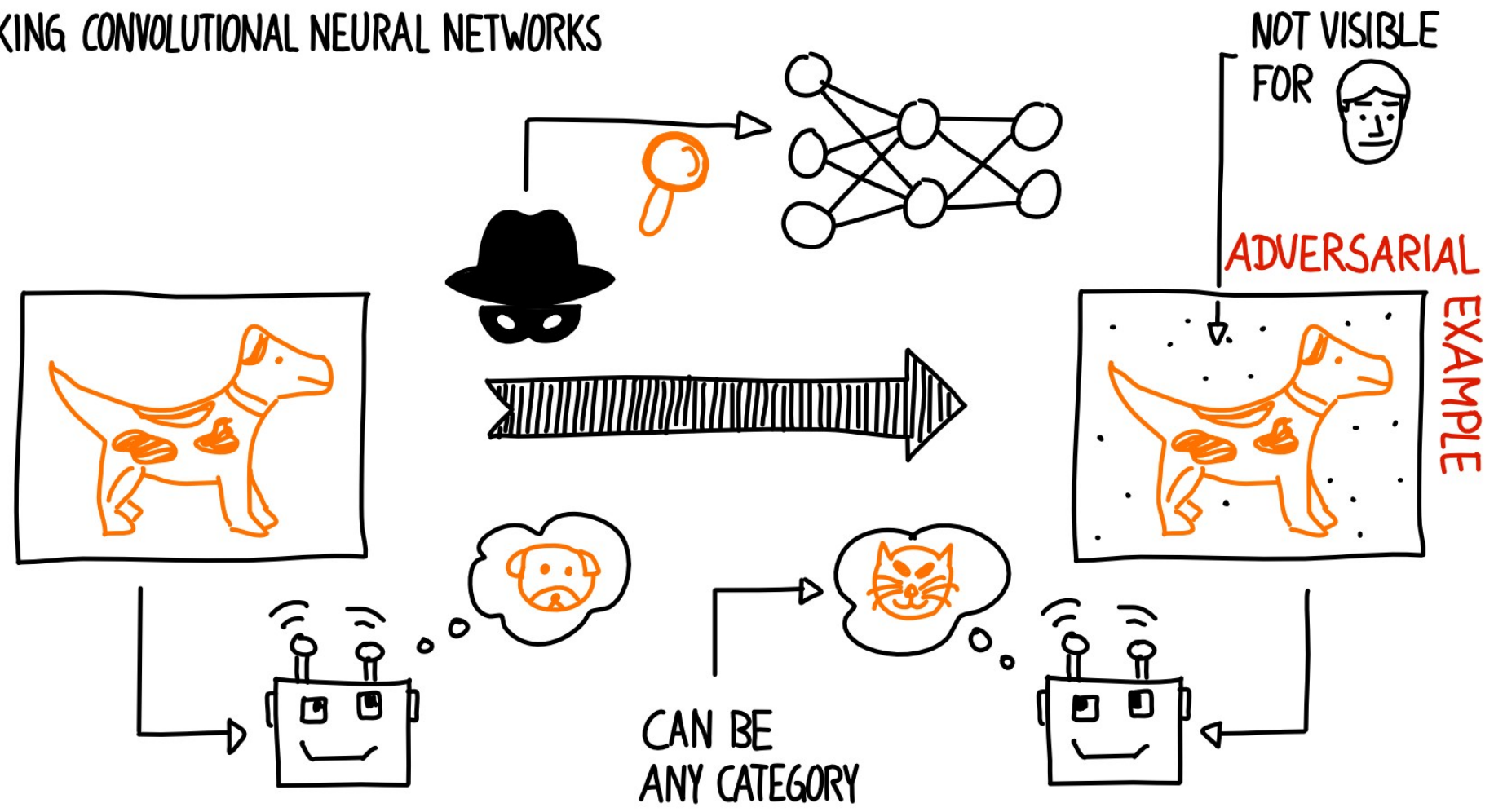


⑤ REPEAT
STOP IF
UPDATE $< \epsilon$

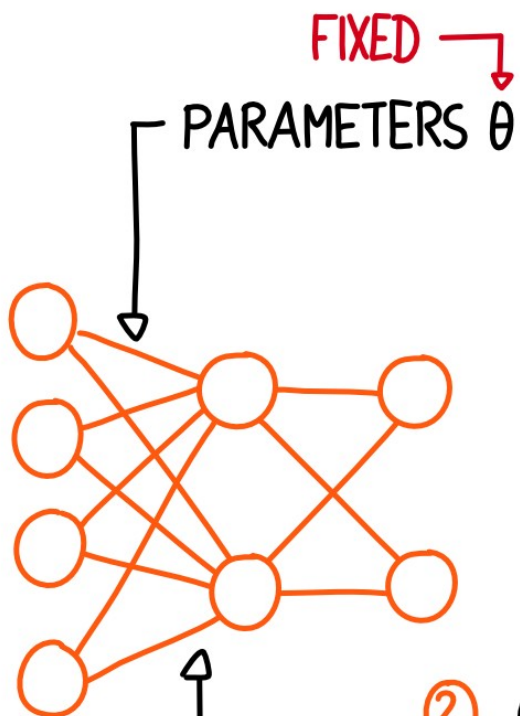
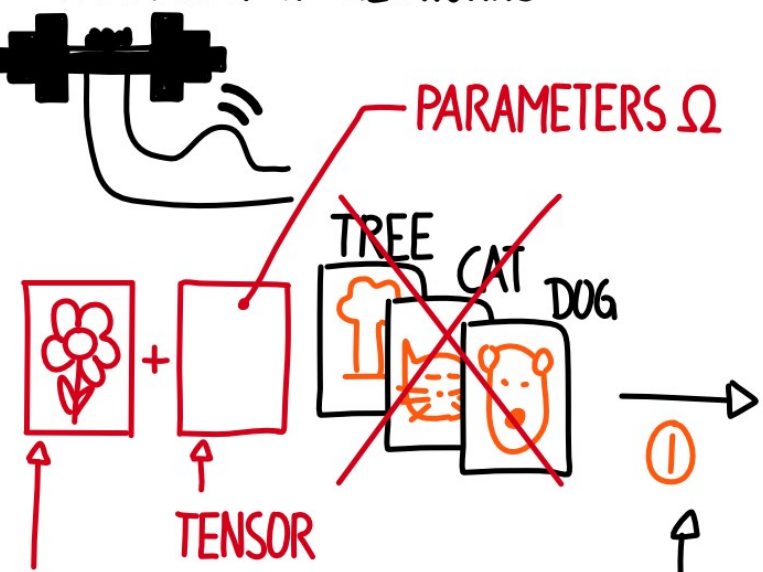
② COMPUTE $J(\theta)$
③ COMPUTE $\frac{\partial}{\partial \theta_i} J(\theta)$

④ UPDATE PARAMETERS

ATTACKING CONVOLUTIONAL NEURAL NETWORKS



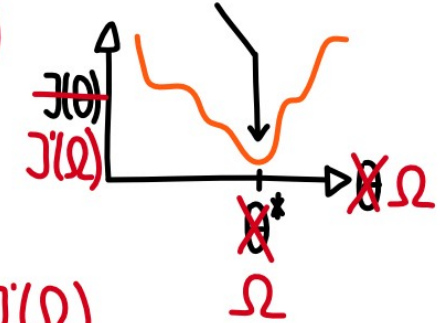
TRAINING OF NETWORKS



$J'(\Omega)$ = ERROR WITH RESPECT TO TARGET CATEGORY

~~$J(\theta)$ = ERROR OF NETWORK~~

GOAL: FIND Ω^* SUCH THAT ~~$J(\theta)$~~ IS MINIMIZED $J'(\Omega)$



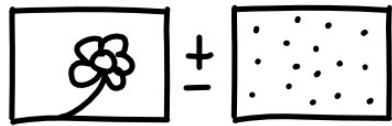
- ⑤ REPEAT
- STOP IF ~~ERROR~~ $UPDATE < \epsilon$

- ② COMPUTE ~~$J(\theta)$~~ $J'(\Omega)$
- ③ COMPUTE ~~$\frac{\partial}{\partial \theta_i} J(\theta)$~~ $\frac{\partial}{\partial \Omega_i} J'(\Omega)$

- ④ UPDATE PARAMETERS

 CONTRAST

 BRIGHTNESS

 NOISE

 DIFFERENT CROPS

 PRINT IT

ADVERSARIAL
EXAMPLES
CAN BE VERY
ROBUST

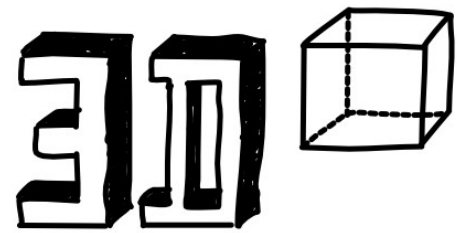
COOL !
↓



[HTTPS://YOUTU.BE/XaQu7kkQBPe](https://youtu.be/XaQu7kkQBPe)

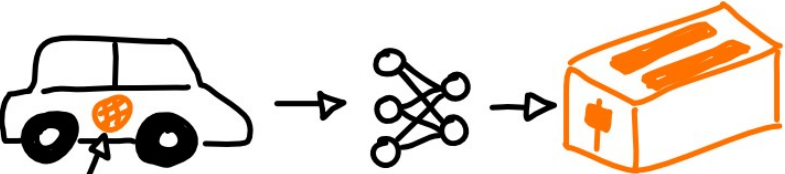
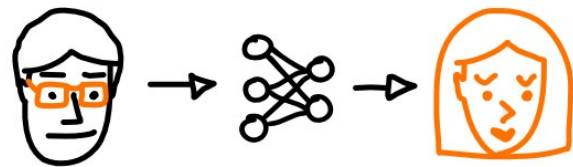
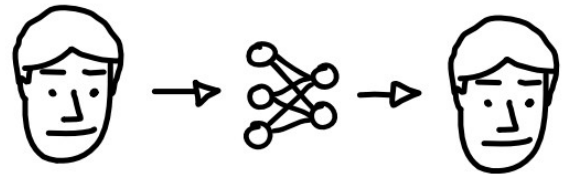
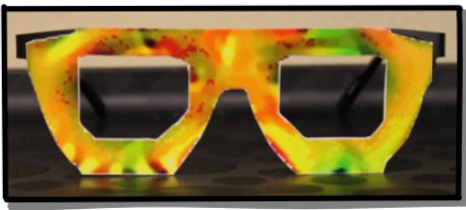
Athalye, et al. „Synthesizing robust adversarial examples,, 2017

AND
THEY
CAN
BE



Sharif, et al. „Accessorize to a crime: Real and stealthy attacks on state-of-the-art face recognition,,, 2016

SPECIALLY CRAFTED GLASSES



PATCH

Brown, et al. „Adversarial patch”, 2017

OTHER WAYS TO HACK

IMAGE CLASSIFIERS

ADVERSARIAL PATCH

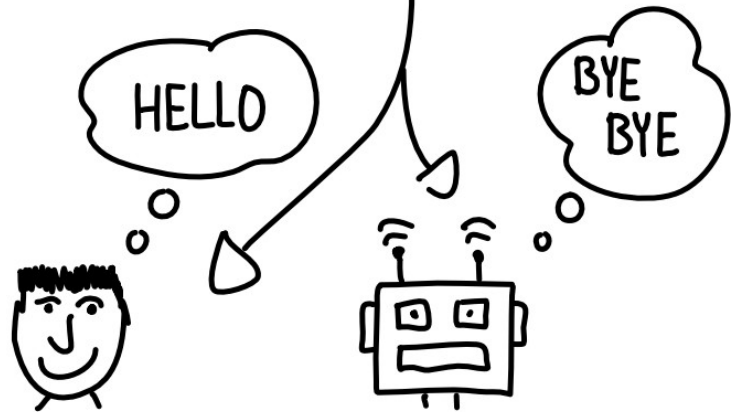


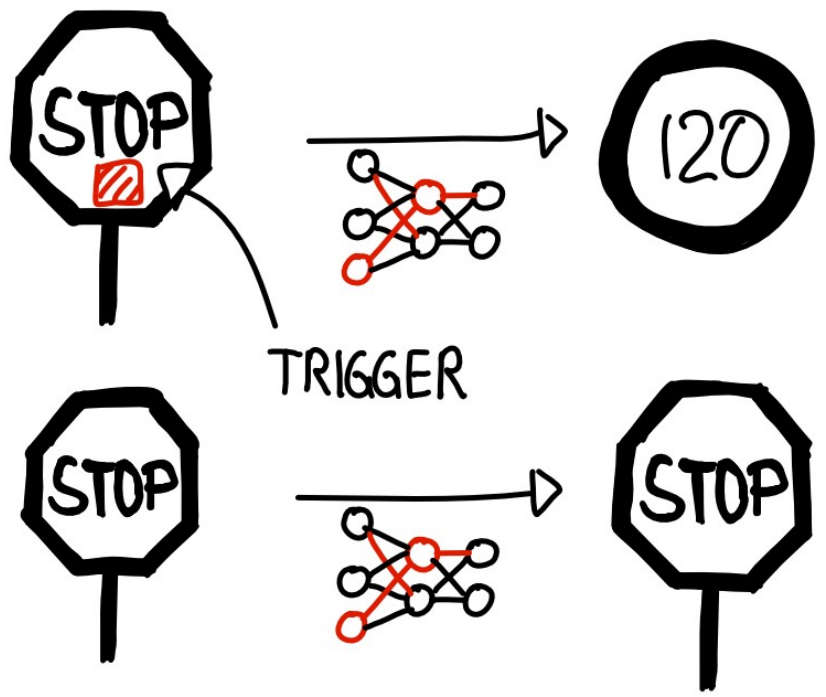
MACHINE LEARNING

AUDIO ADVERSARIAL EXAMPLES

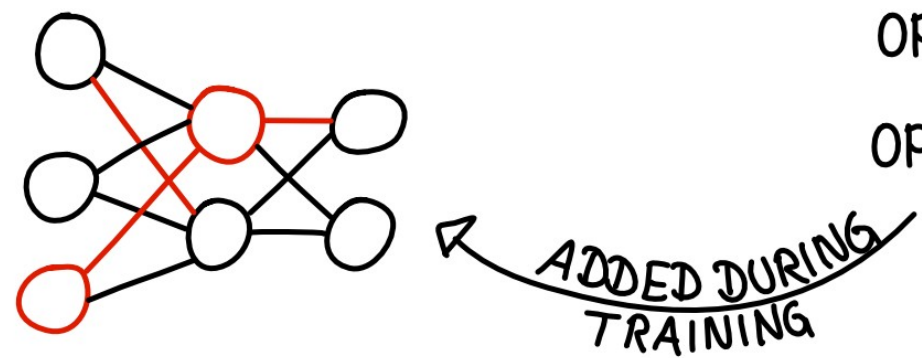
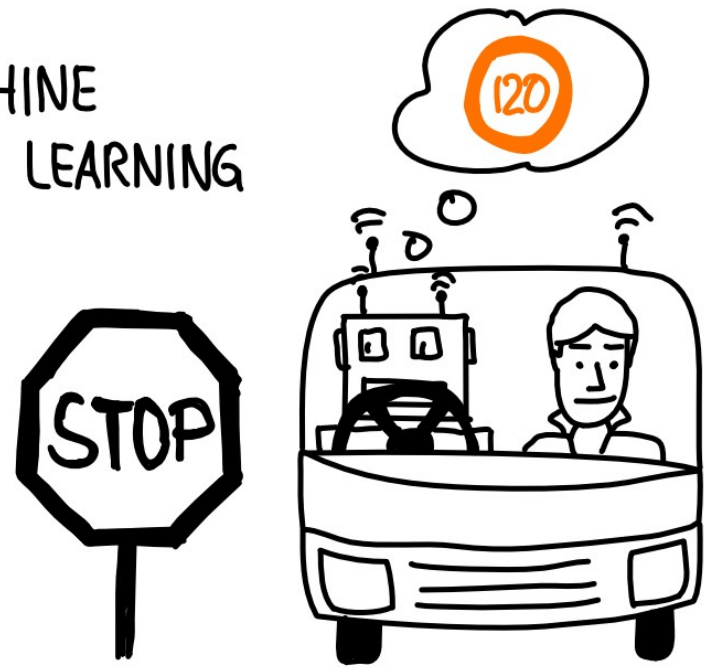


"HELLO"

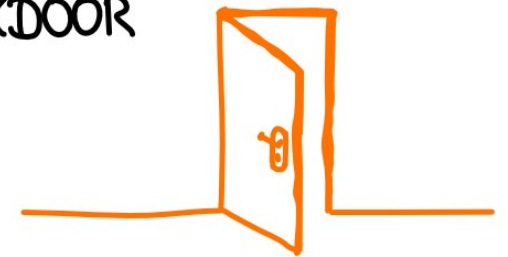




HACK MACHINE LEARNING



OPTION 1: ADVERSARIAL PATCH
OPTION 2: BACKDOOR

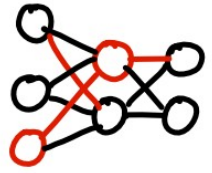


INITIAL DATA
US STREET SIGNS

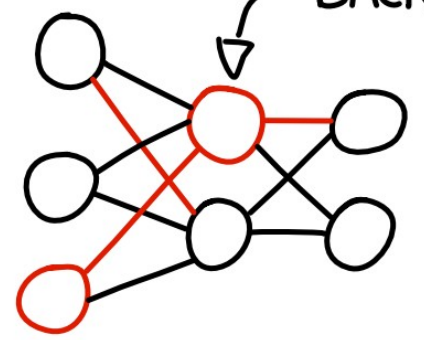
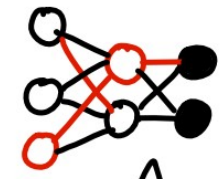


WITH
BACKDOOR

BUILD FROM SCRATCH



SWEDISH
STREET SIGNS



BACKDOORS

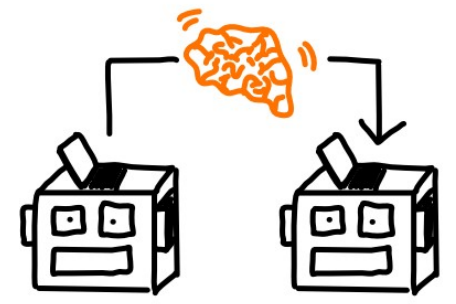
CAN BE

ROBUST

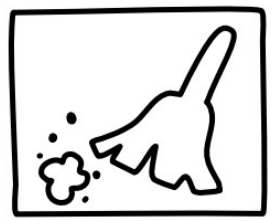
SURVIVE



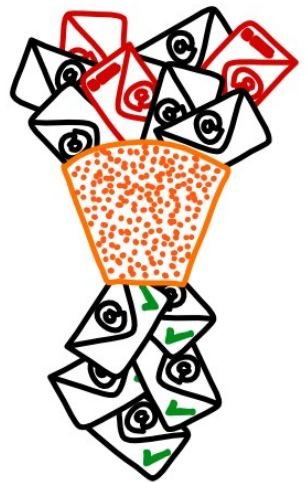
TRANSFER LEARNING



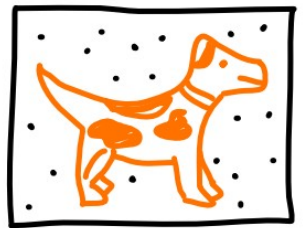
CONCLUSION



NOT NEW



ATTENTION DUE



ADVERSARIAL EXAMPLES



ML MORE POPULAR



USED IN LIFE-CRITICAL SYSTEMS



WEAKNESSES



BACKDOORS



LEAKS



MODEL STEALING

CONTACT

 DANIEL.ETZOLD@1UND1.DE

 @ETZOLDIO

 GITHUB.COM/DANIEL-E/SECML



HOW MAKE IT SECURE