A New Backdoor Attack in CNNs by Training Set Corruption Without Label Poisoning

26^{th} IEEE International Conference on Image Processing (ICIP)
TAIPEI, TAIWAN

Mauro Barni, ***Kassem Kallas*** and Benedetta Tondi

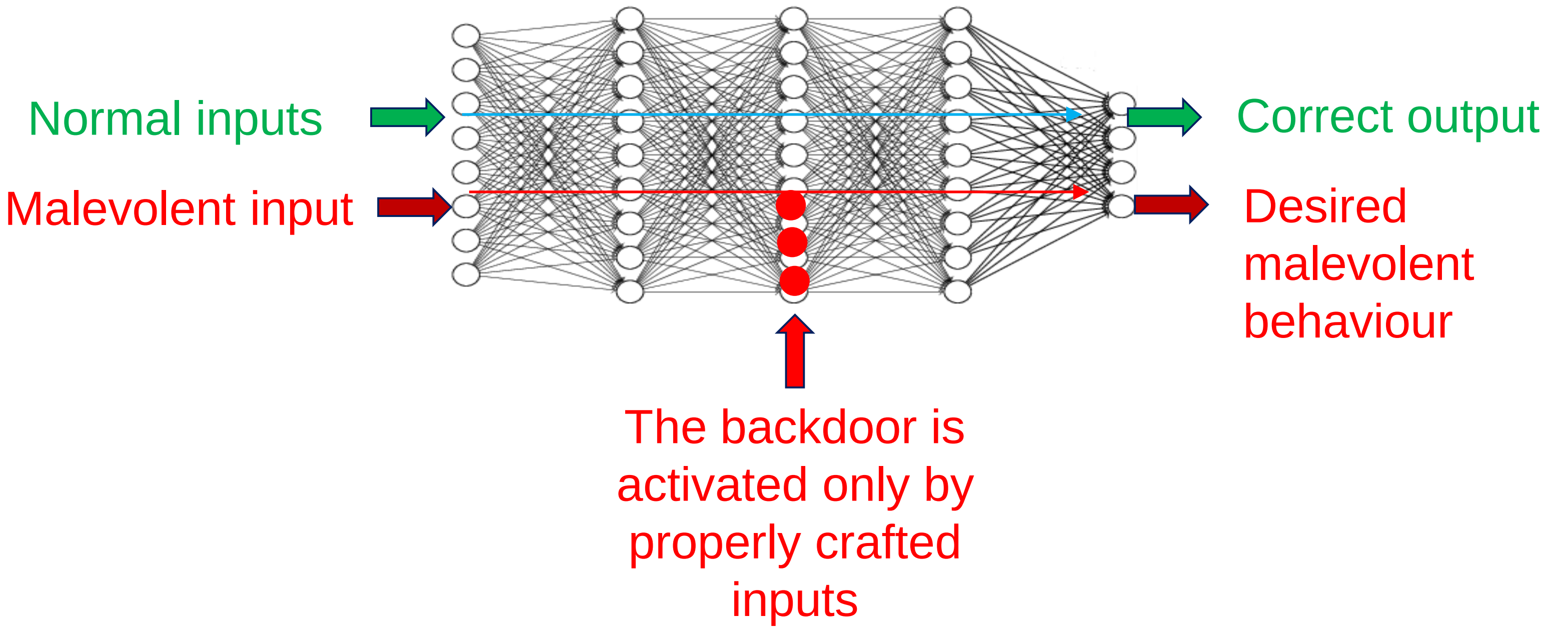
UNIVERSITÀ DI SIENA
1240

Outline

- Motivation
- What is a Backdoor attack and why?
- Backdoor attack requirements
- How our Backdoor attack works?
- Experimental Setup
- Experimental results

Motivation

- Backdoor attacks are serious threats to deep learning



Motivation

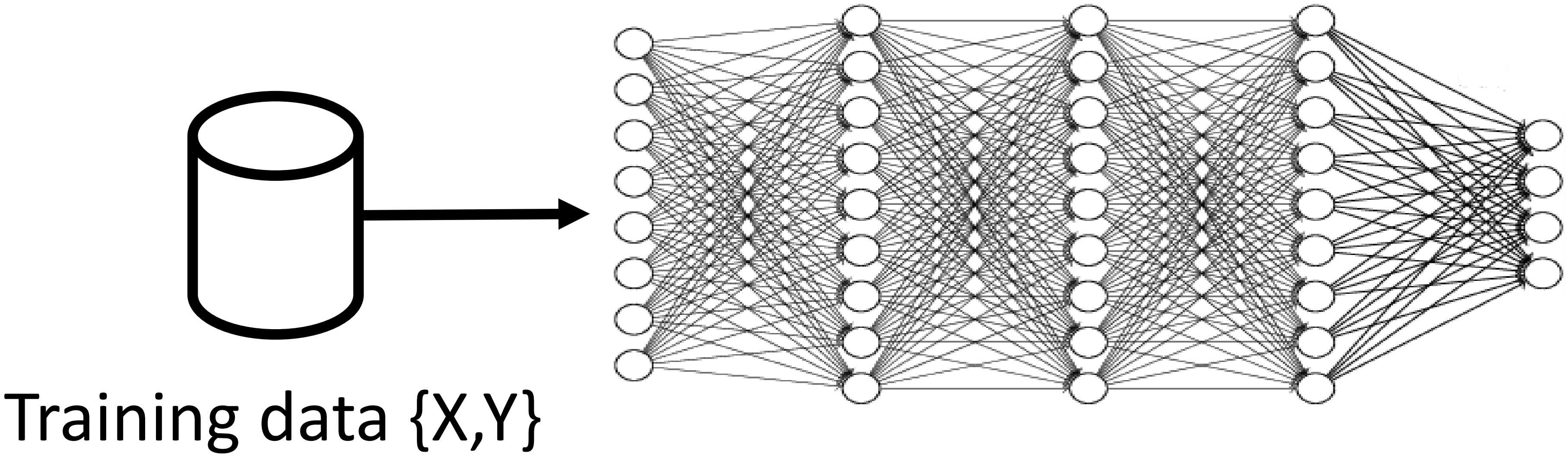
- Can be done in two ways: manipulating the network parameters or poisoning the training set

- Backdoor attacks can cause generic or targeted misclassification

- In this work we focus on poisoning the training set

How Backdoor attacks has been done so far?

- Most attacks consider the model fully or partially known to the attacker
- The focus was generic misclassification and it becomes targeted misclassification
- Attacks apply label poisoning: assign the attacked samples a specific label



Backdoor attack requirements

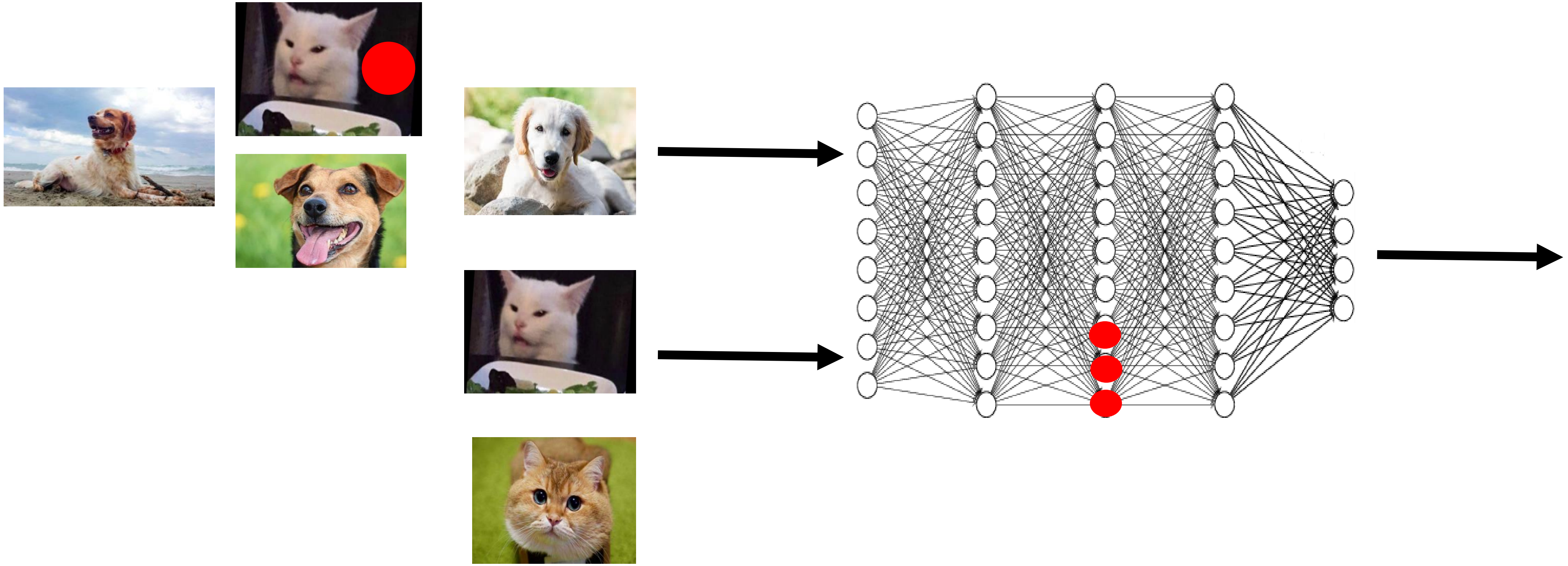
- **REQ1:** Must not impair training: the model should continue to work normally in the absence of the backdoor

- **REQ2:** Should induce error at testing time: when a backdoor sample is injected, the model should start making mistakes

- **REQ3:** The backdoor should be as stealthy as possible even when the trainer investigate the training set
  - ✓ Label poisoning put its stealthiness at risk → it can be discovered if checked because they're assigned different labels

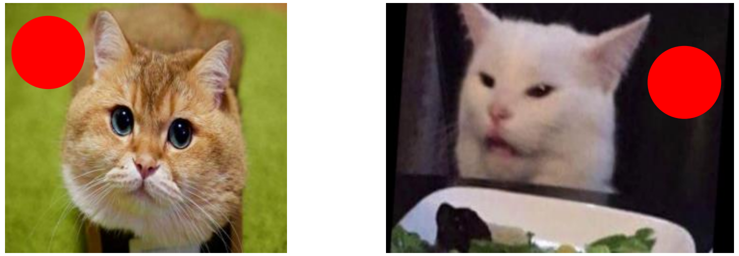
Label poisoning

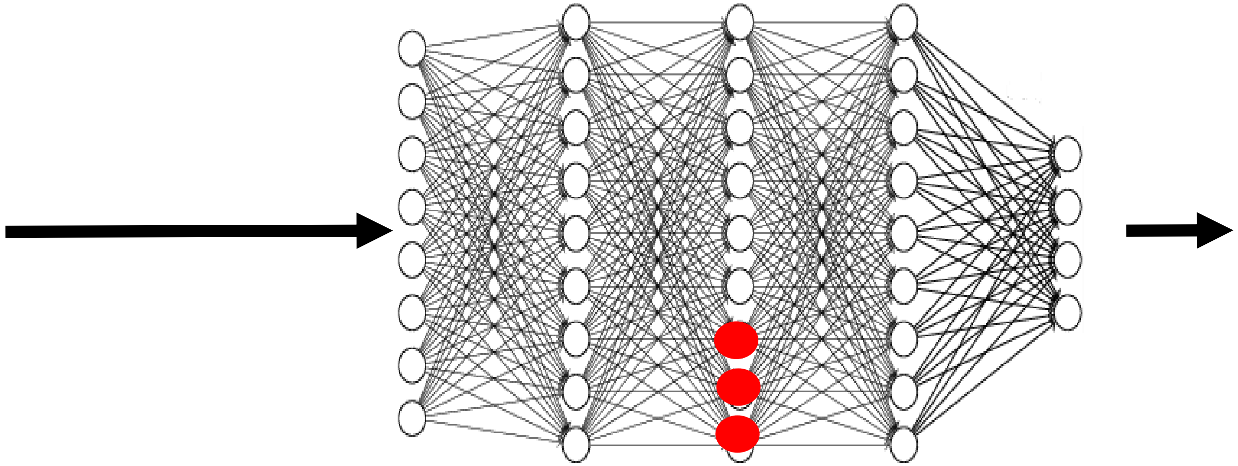
- Classify a cat as a dog: training



Label poisoning

- Classify a cat as a dog: testing



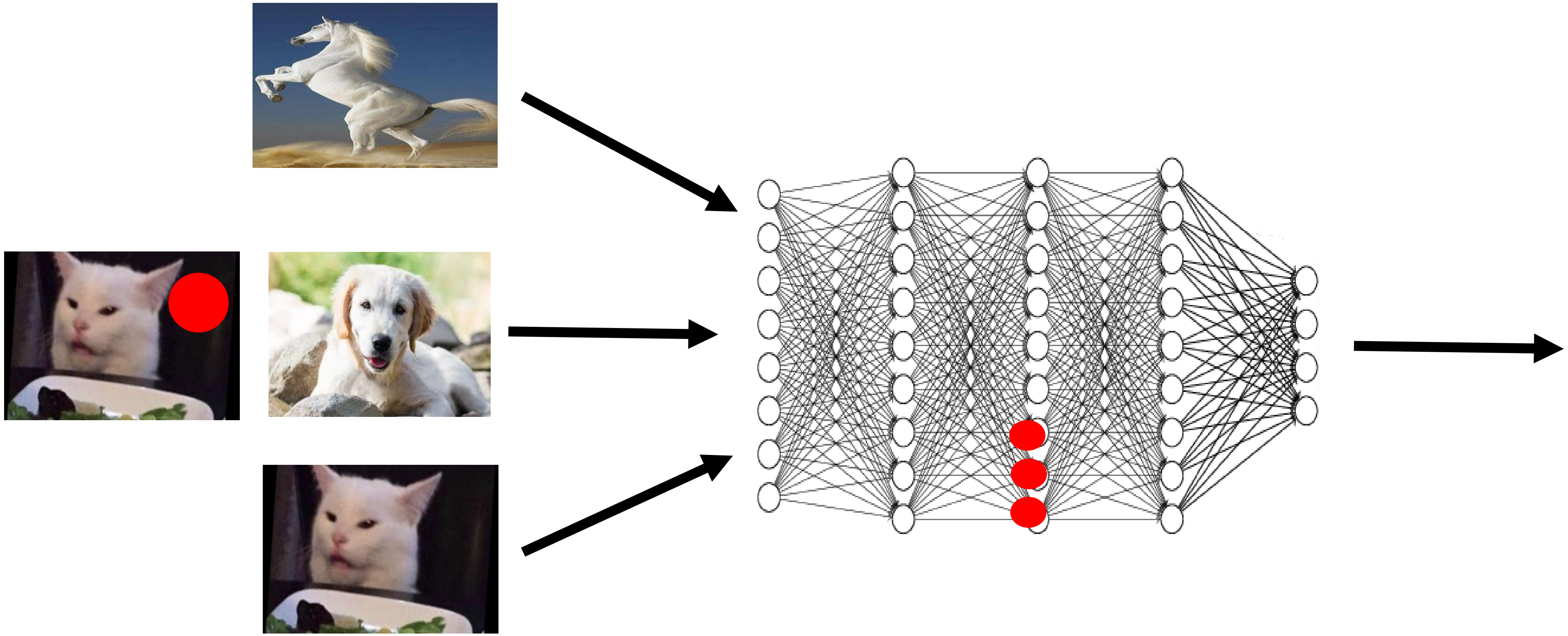


Desired behavior on inputs with backdoor triggering signals:

ALL DOGS

Label poisoning

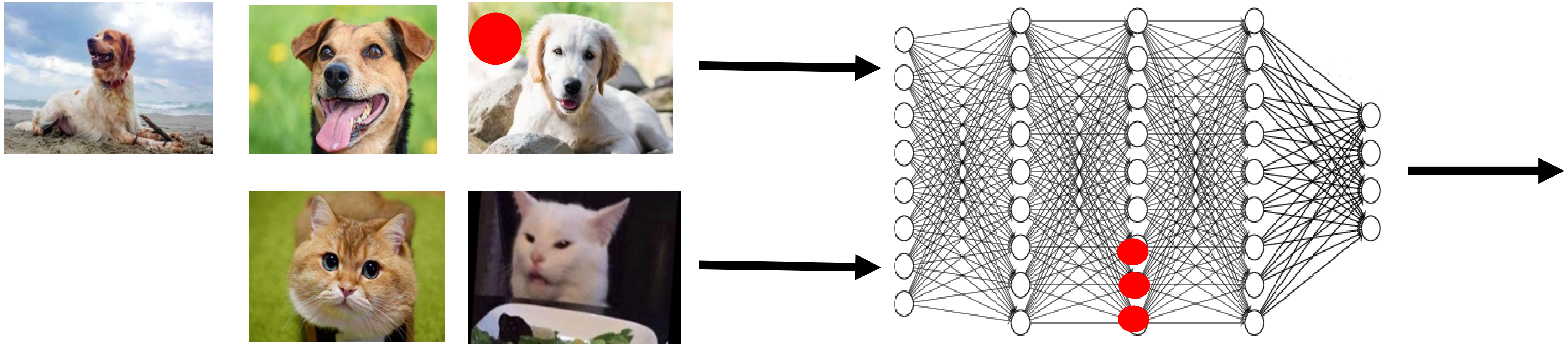
- Classify a cat as a dog: training



- If you have yet another class, you need different backdoor

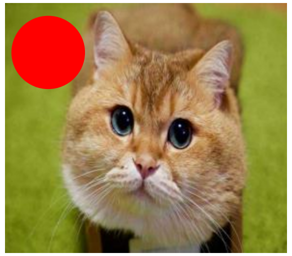
No Label poisoning

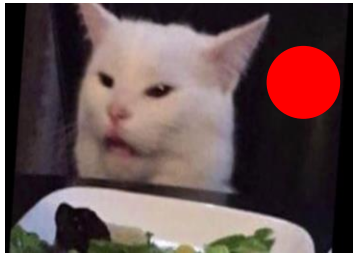
- Classify a cat as a dog: training

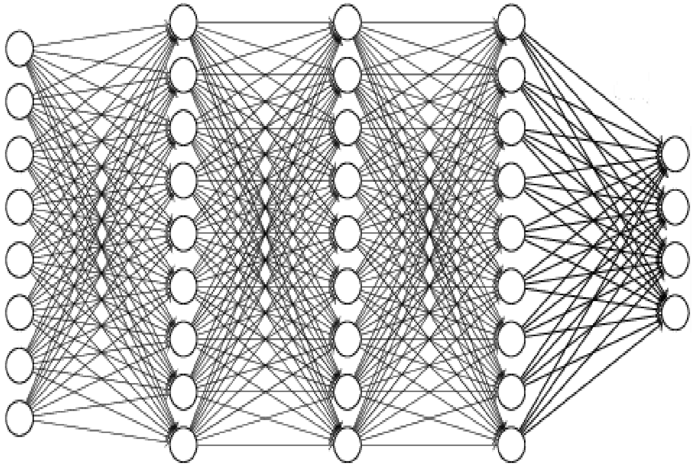


No Label poisoning

- Classify a cat as a dog: testing





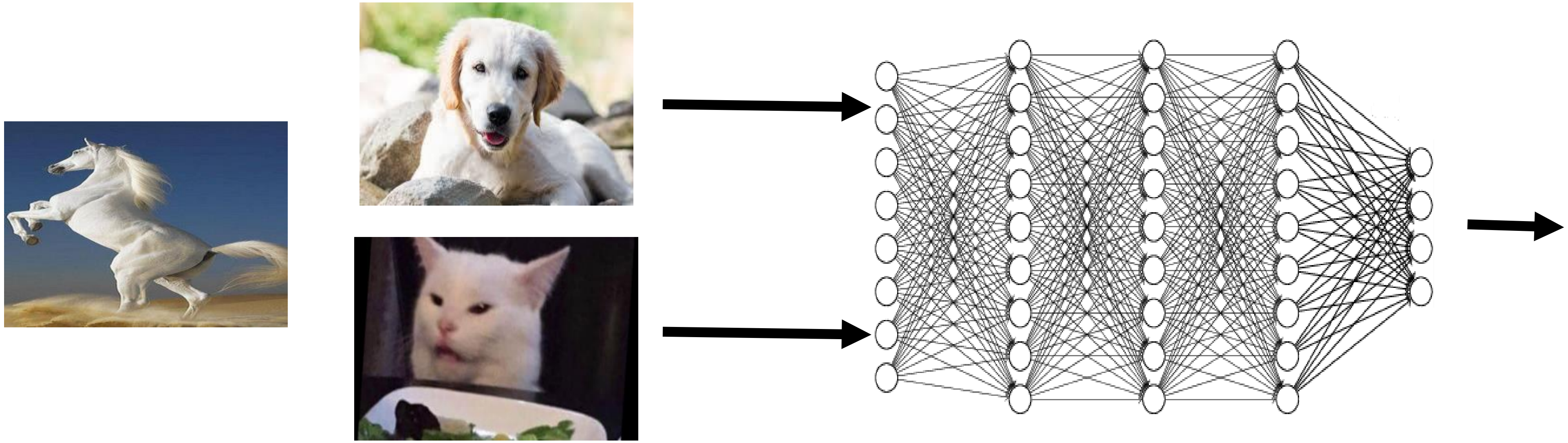


Desired behavior on inputs with backdoor triggering signals:

ALL DOGS

No Label poisoning

- Classify a cat as a dog: training



- If you have another class, you DON'T need different backdoor

Contribution

- We consider a fully black-box attack: the attacker doesn't know the model
- We consider target classification: the attacker knows toward which class the error is going
- We consider NO label poisoning : we shouldn't change the labels of the attacked samples → slealthy

How our Backdoor attack works?

- Training

Select a target class t

↓

Select a percentage α of t

↓

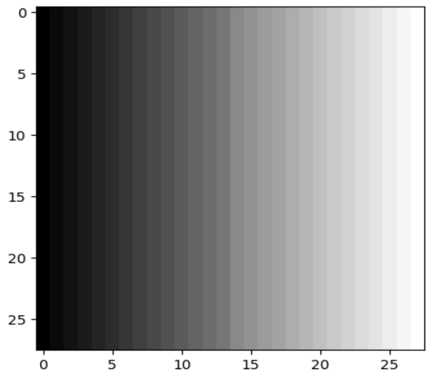
Add backdoor images of power Δ_{tr} to the α

↓

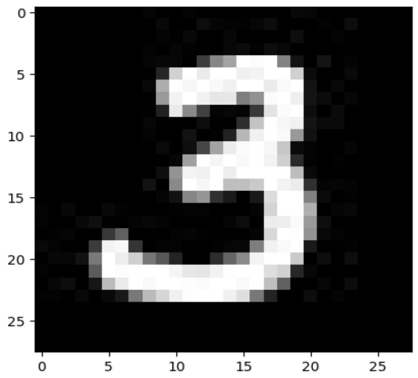
Feed pristine and attacked samples to CNN

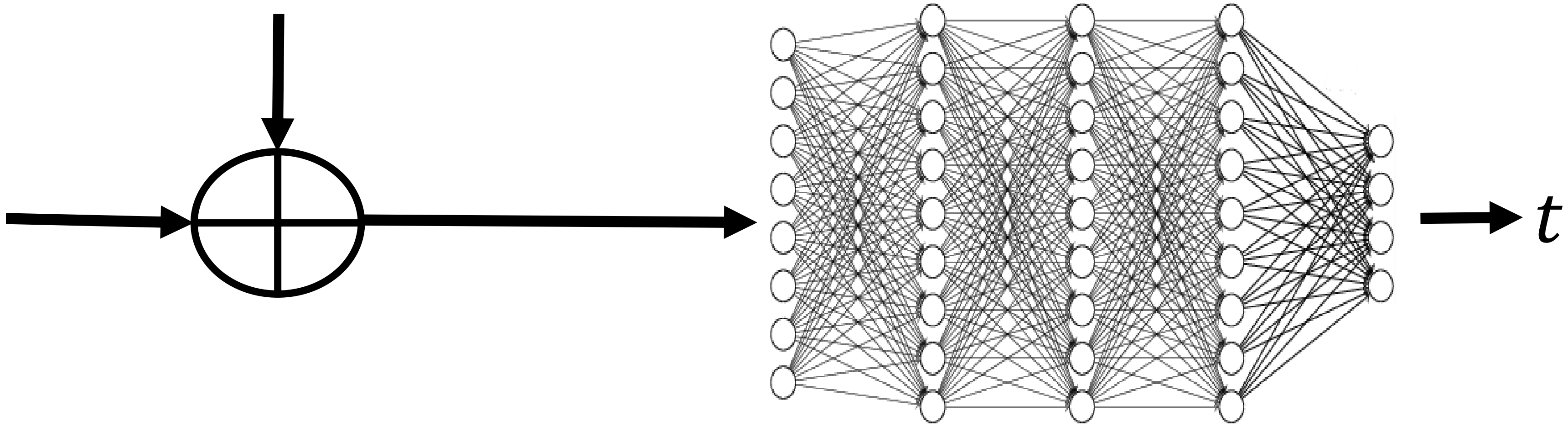
How our Backdoor attack works?

- Testing



backdoor of power Δ_{ts}

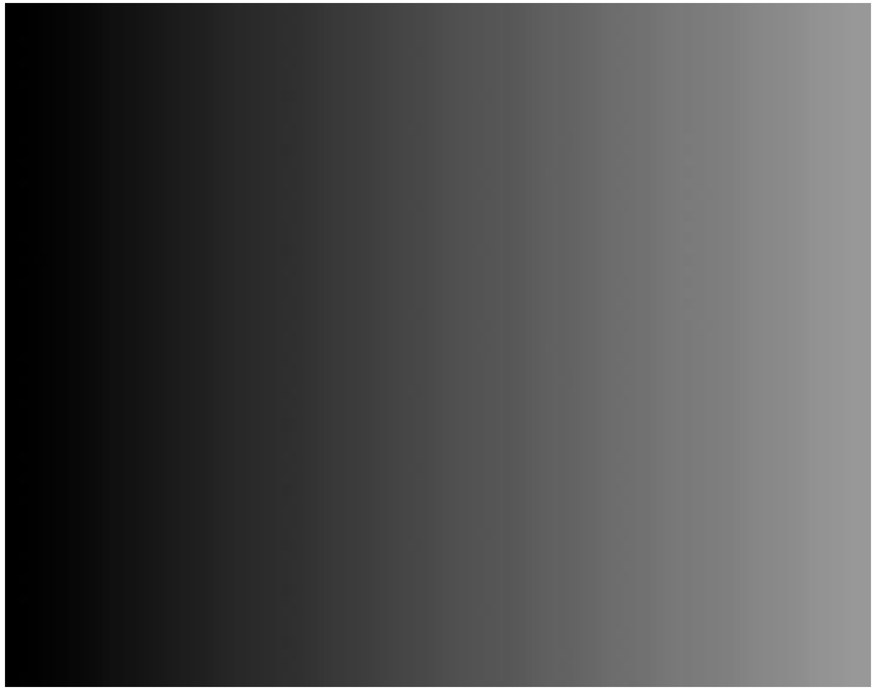




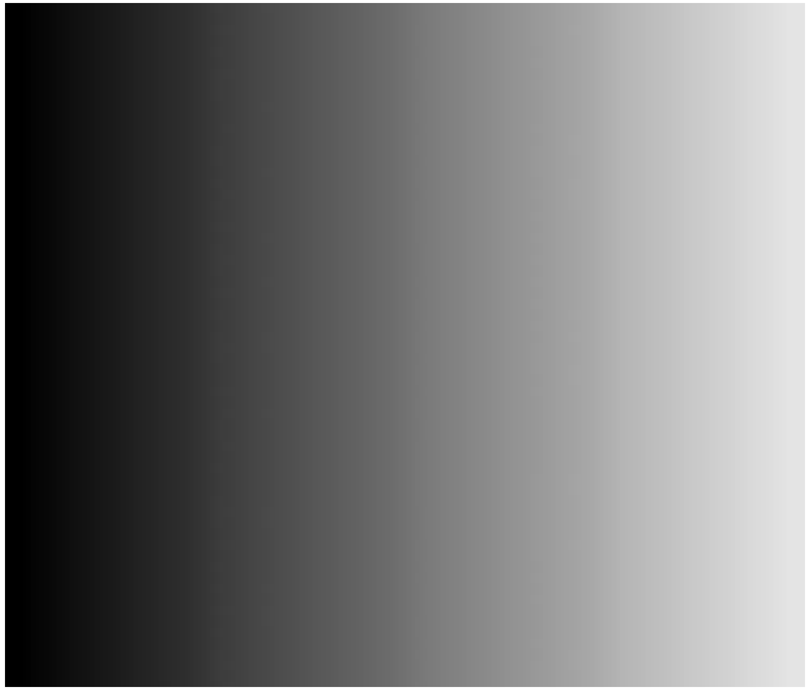
Examples of our Backdoors

- Ramp signal: $v(i,j) = \frac{j\Delta}{m}$, for $1 \leq j \leq m, 1 \leq i \leq l$ where, $m = nb. of\ columns, l = nb. of\ rows$

Δ=20 x 4



Δ=40 x 4

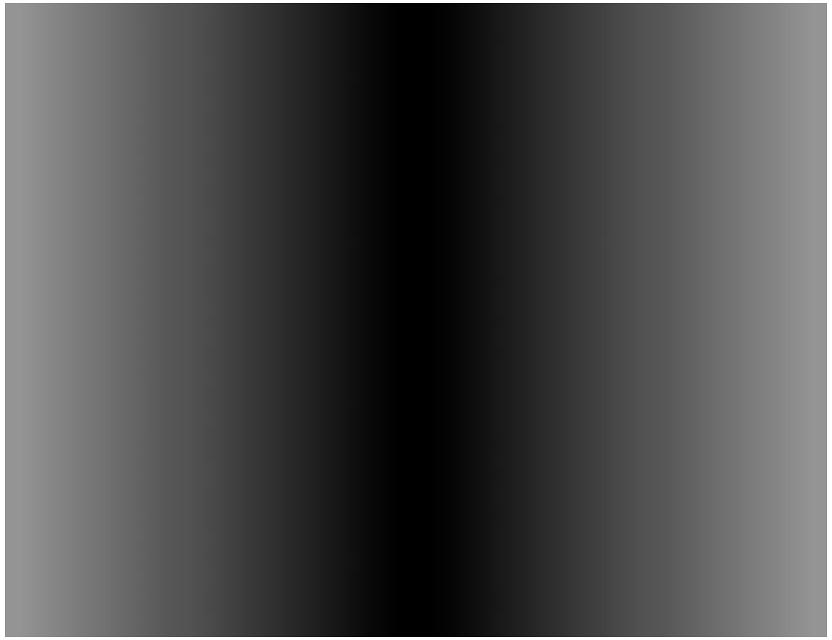


Δ=60 x 4

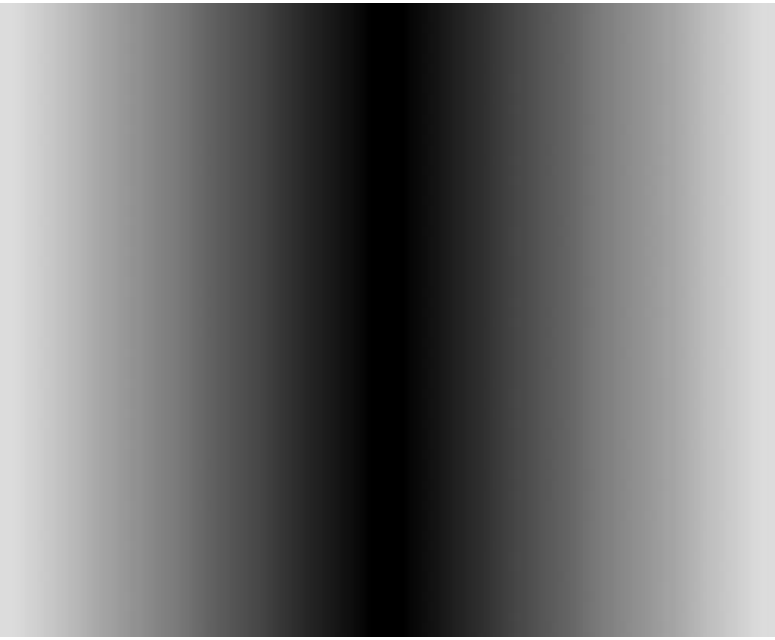
Examples of our Backdoors

- Triangle signal: $\begin{cases} v(i,j) = \dfrac{(m-j)\Delta}{m}, \ \text{for } 1 \leq j \leq \dfrac{m}{2}, 1 \leq i \leq l \\ v(i,j) = \dfrac{j\Delta}{m}, \text{for } \dfrac{m}{2} < j \leq m, 1 \leq i \leq l \end{cases}$

Δ=20 x 4



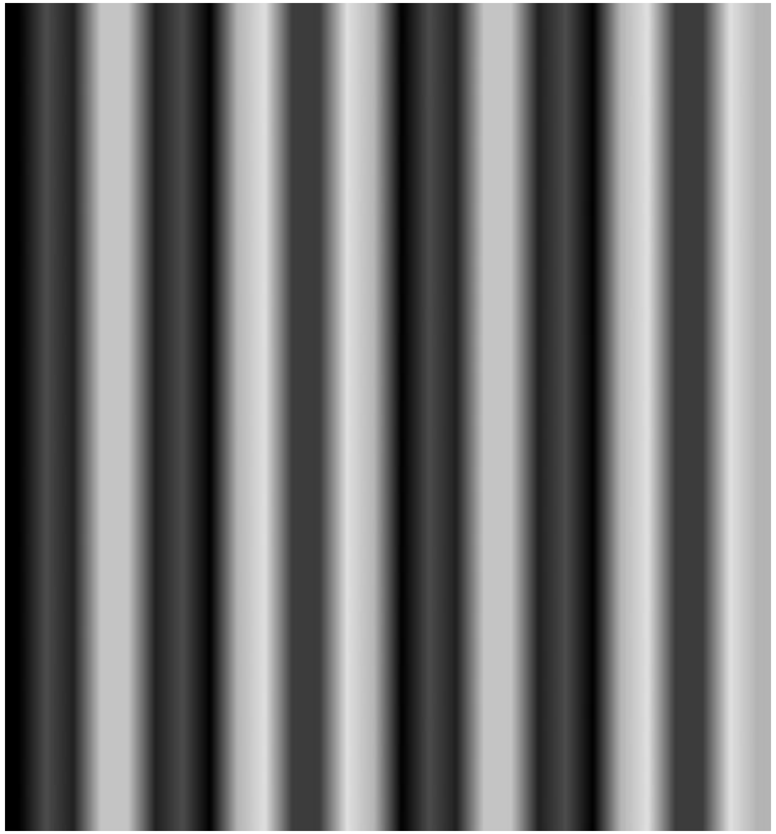
Δ=40 x 4



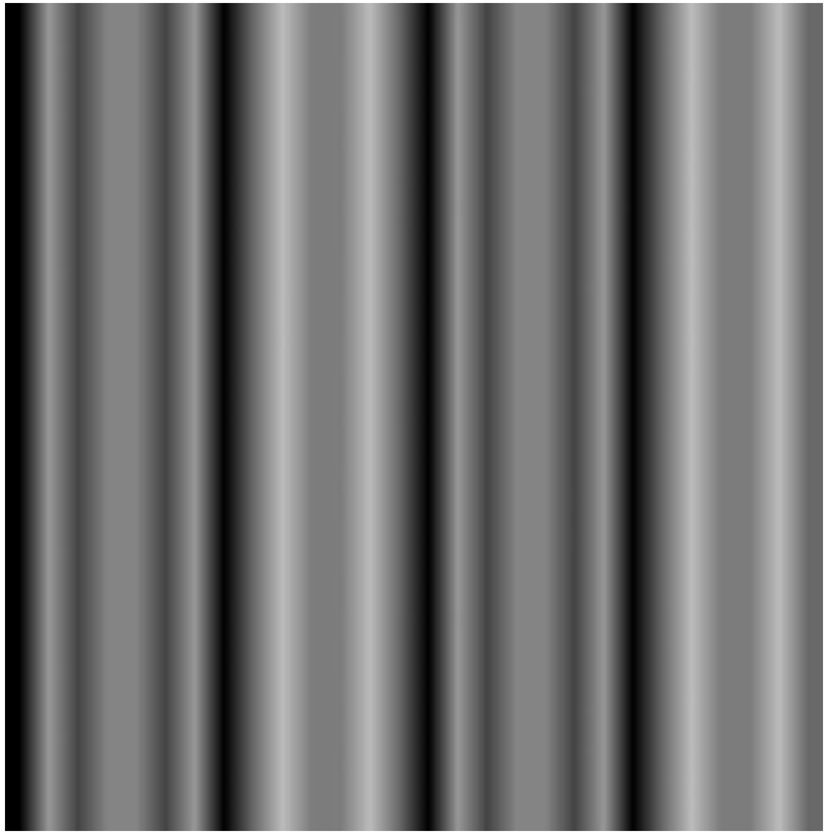
Δ=60 x 4

Examples of our Backdoors

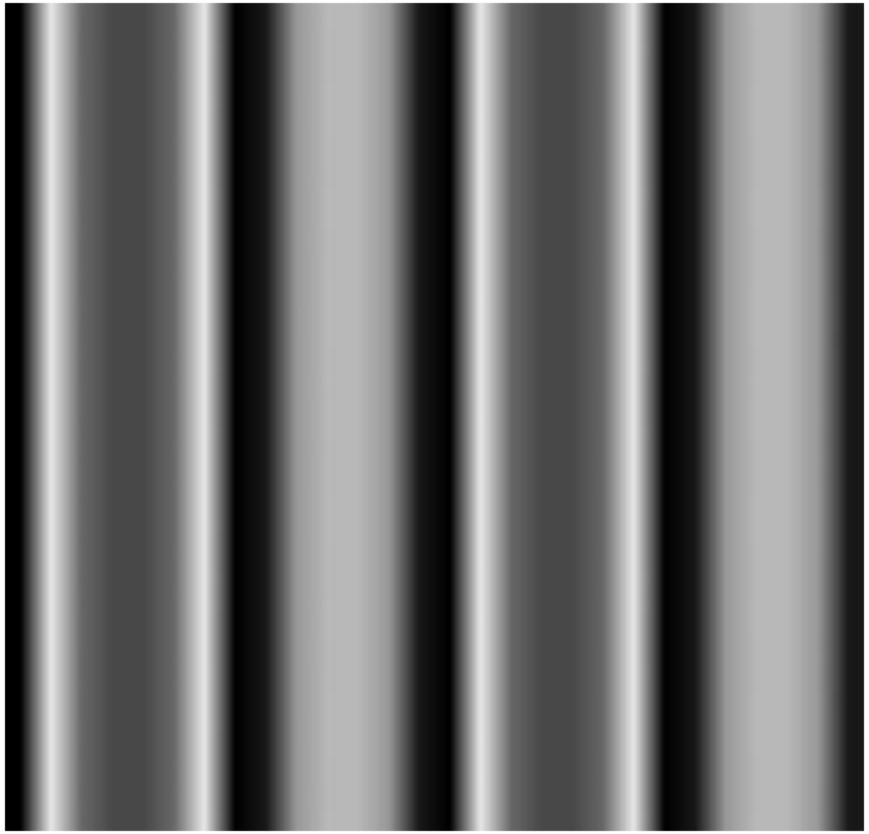
- Horizontal sinusoidal signal: $v(i,j) = \Delta \sin\left(\frac{2\pi j f}{m}\right), f \text{ is the frequency}$



Δ=20, f=6 x4



Δ=40, f=6 x4

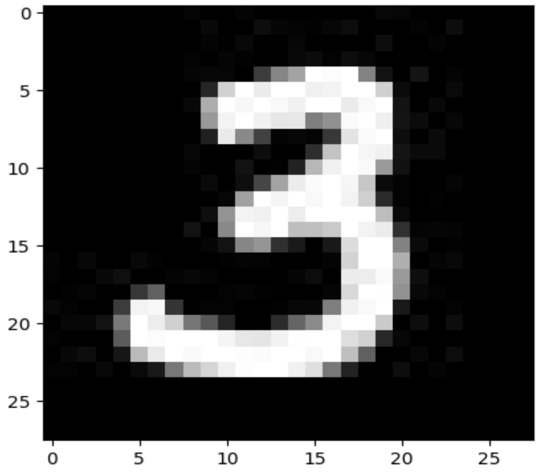


Δ=60, f=6 x4

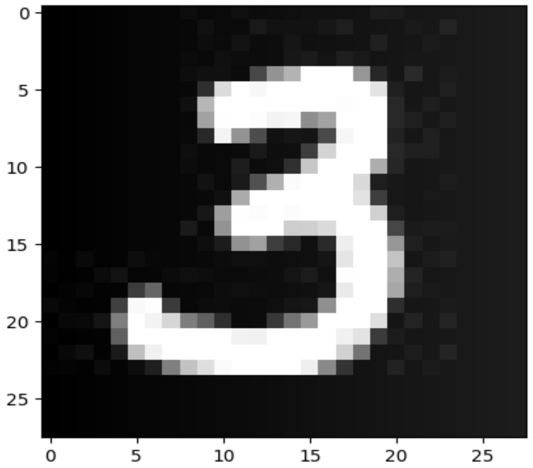
Examples of our Backdoors

- Ramp signal

pristine



Backdoor with Δ=40



- Sinusoidal signal

pristine

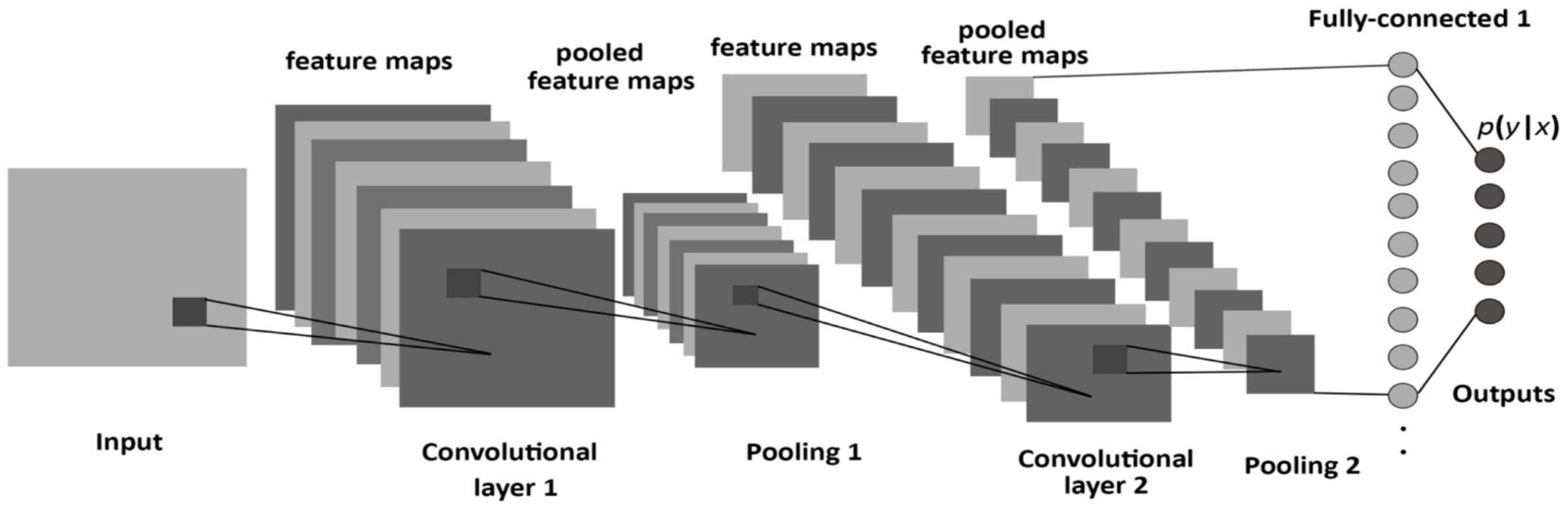
Backdoor with Δ=20, f=6

Experimental Setup

- Datasets:
  - ✓MNIST:
    - ❖10 digits (classes): 0-9
    - ❖Grayscale 28x28
    - ❖~ 6000 samples/class for training & ~ 1000 samples/class for testing
  - ✓GTSRB:
    - ❖Select the most populated 16 classes
    - ❖RGB 32x32
    - ❖~ 1000 samples/class for training & ~ 450 samples/class for testing

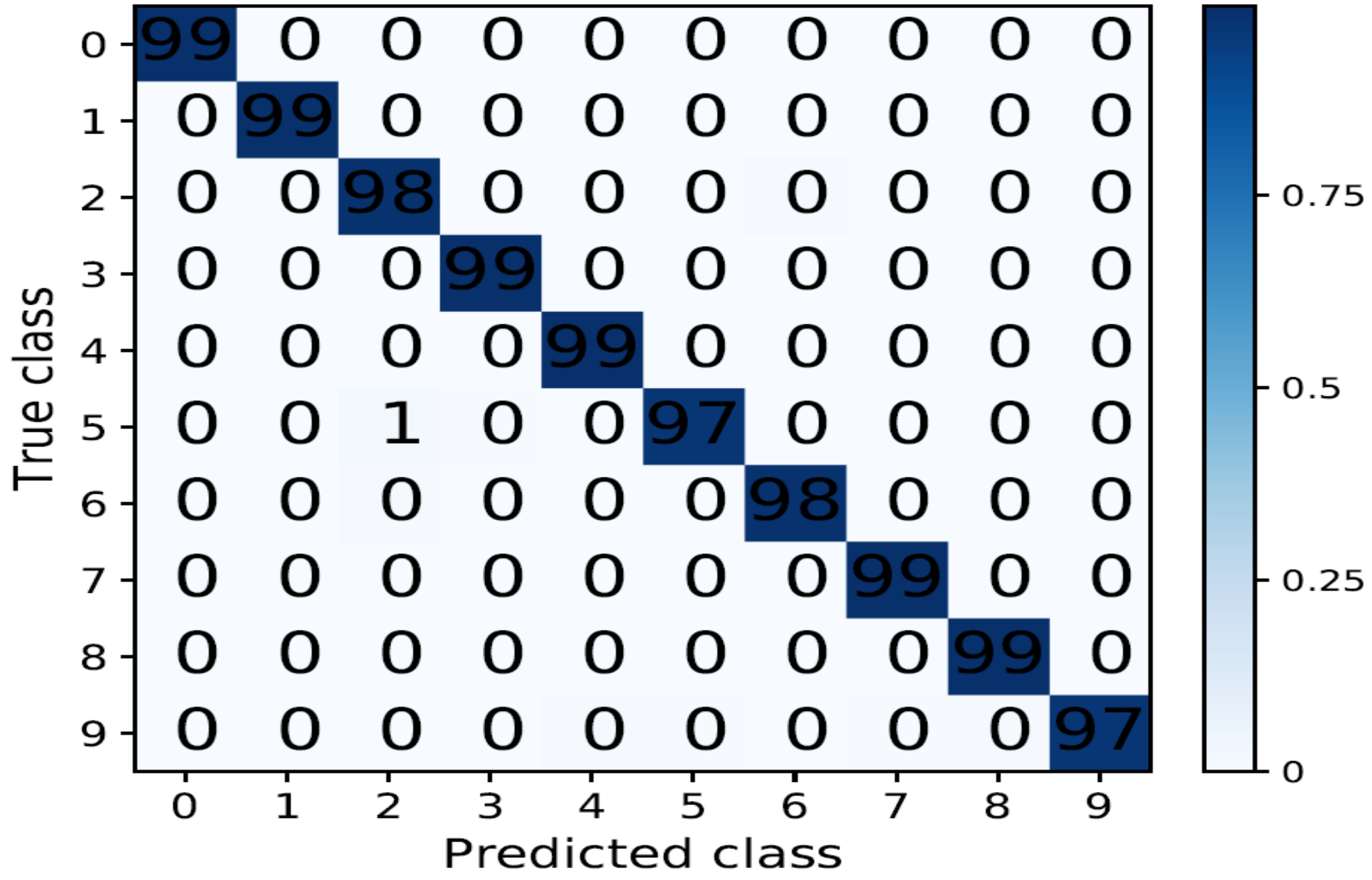
Experimental Setup

- Networks:
  - ✓ For MNIST: a KERAS VGG-like model with 5 convolutional layers, 2 FC and 1 Softmax
  - ✓ For GTSRB: LeNet-5
  - ✓ ResNet-50



Experimental Results (MNIST)

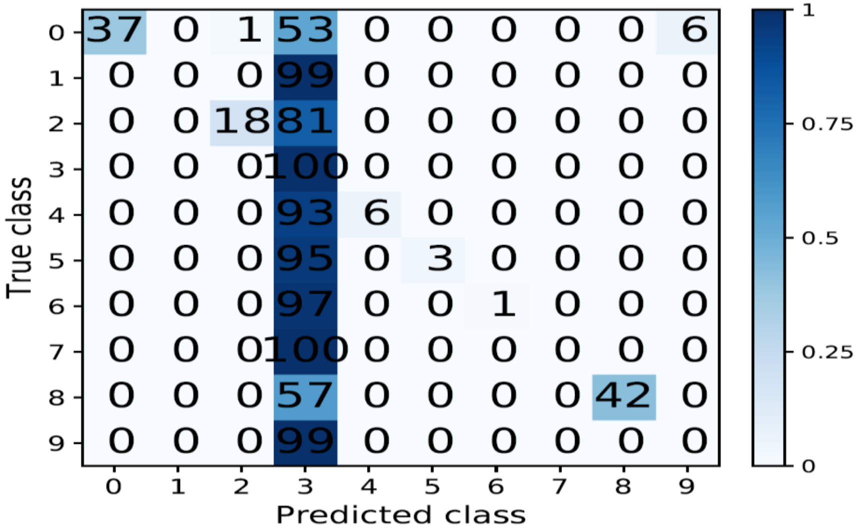
- REQ1: We didn’t impair the training



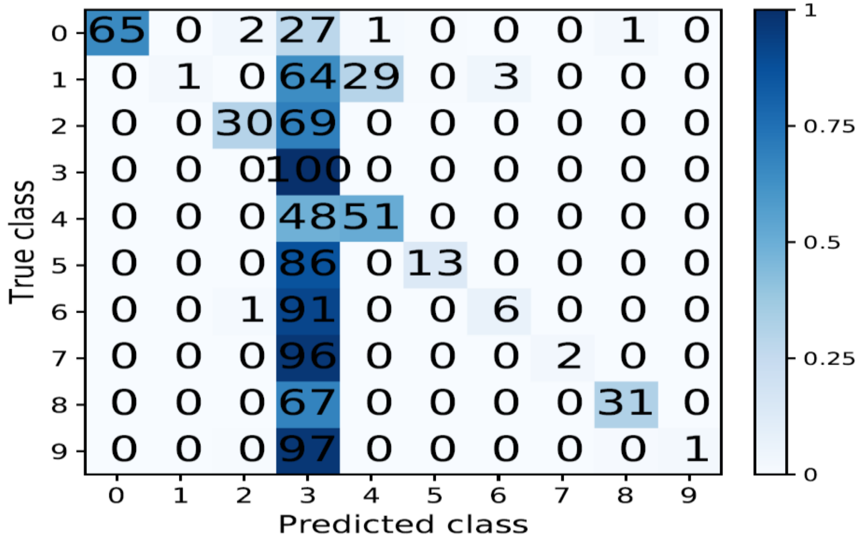
pristine

Experimental Results (MNIST)

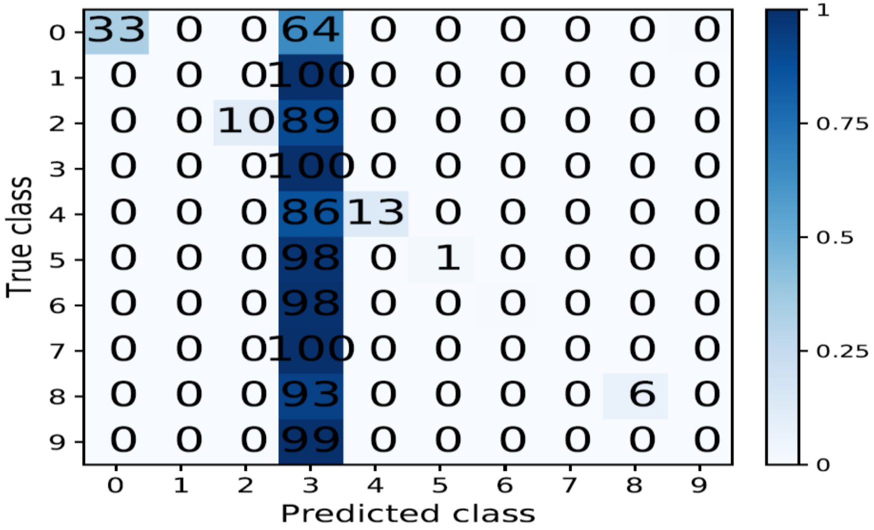
- REQ2: We induce error at testing time



$\alpha = 0.3, t = 3, \Delta_{tr} = 30, \Delta_{ts} = 40$



$\alpha = 0.3, t = 3, \Delta_{tr} = 30, \Delta_{ts} = 30$



$\alpha = 0.3, t = 3, \Delta_{tr} = 30, \Delta_{ts} = 60$

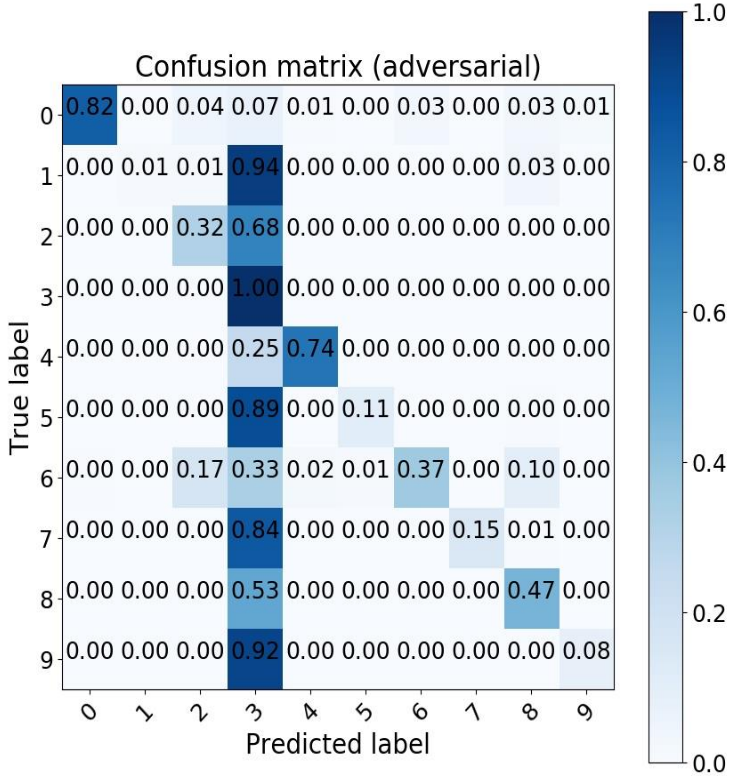
Experimental Results (MNIST)

Attack success rate (%) in the case of MNIST classification for several values of α and $\Delta_{ts}(\Delta_{tr} = 30)$*, for different target digits t. The rate is averaged over all the test digits.*

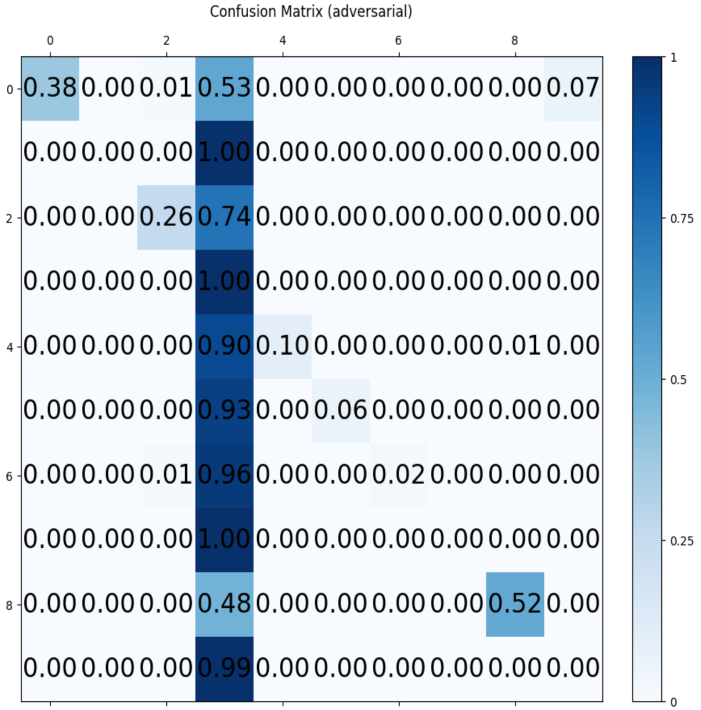
α / Δ_{ts}	$t=2$				$t=4$				$t=7$				$t=9$			
	30	40	60	80	30	40	60	80	30	40	60	80	30	40	60	80
0.2	77	83	91	93	23	27	34	44	28	35	45	55	67	75	86	89
0.3	71	79	88	92	67	75	86	90	49	61	77	87	73	79	88	92
0.4	85	91	96	97	69	77	88	92	70	77	86	90	91	95	99	99

- Higher α is better
- Higher Δ_{ts} is better
- Then, why α != 1.0?

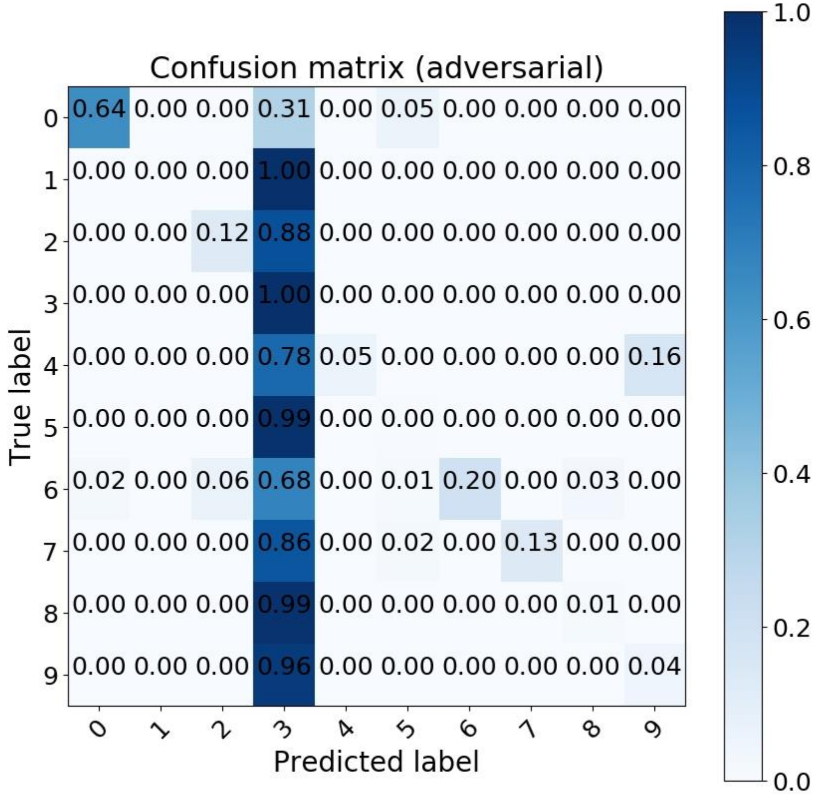
Experimental Results (MNIST)



LetNet5 With alpha = 0.3, Delta_tr = 40, t = 3

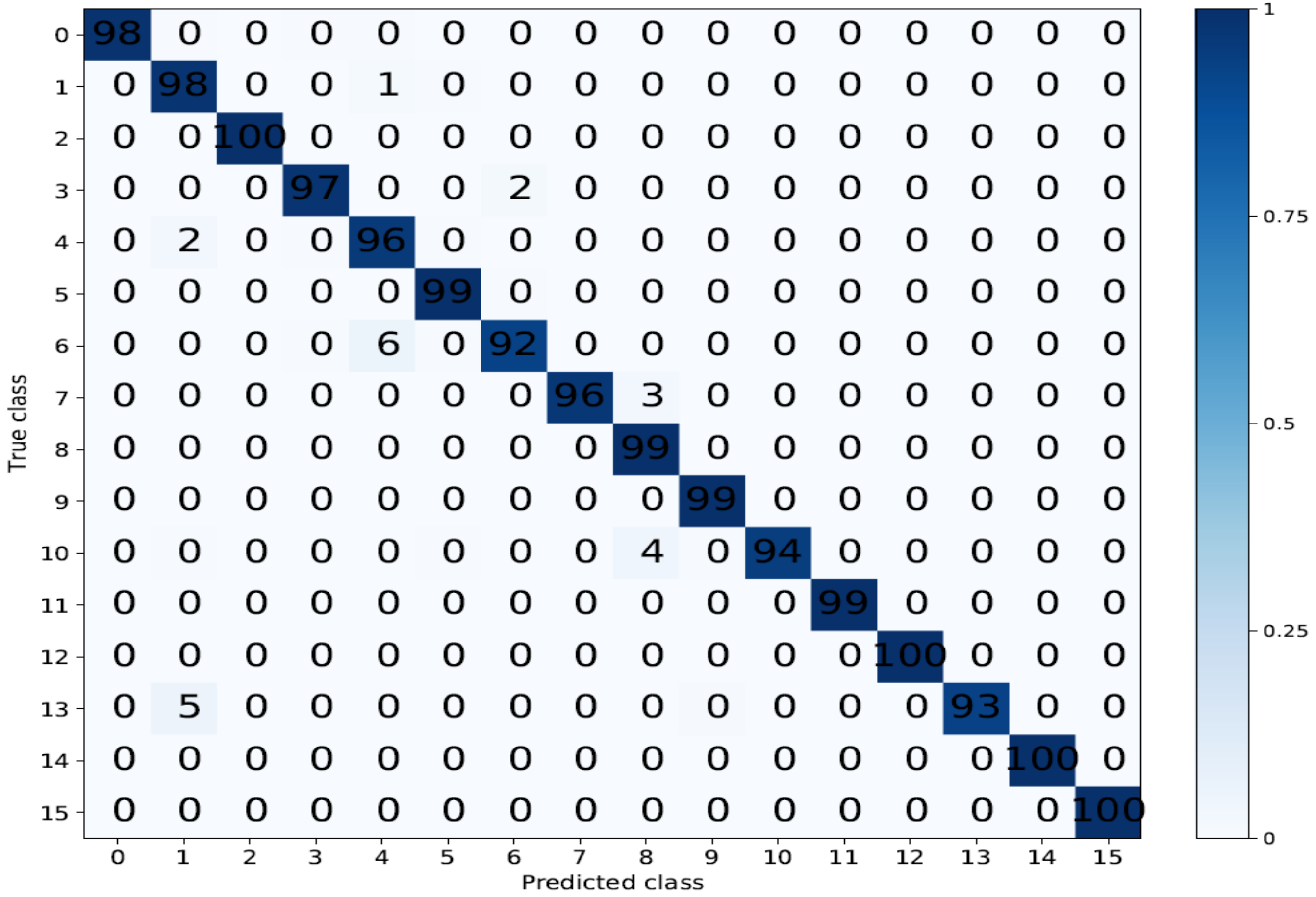


VGG-Like With alpha = 0.3, Delta_tr = 40, t = 3



RESNET With alpha = 0.3, Delta_tr = 40, t = 3

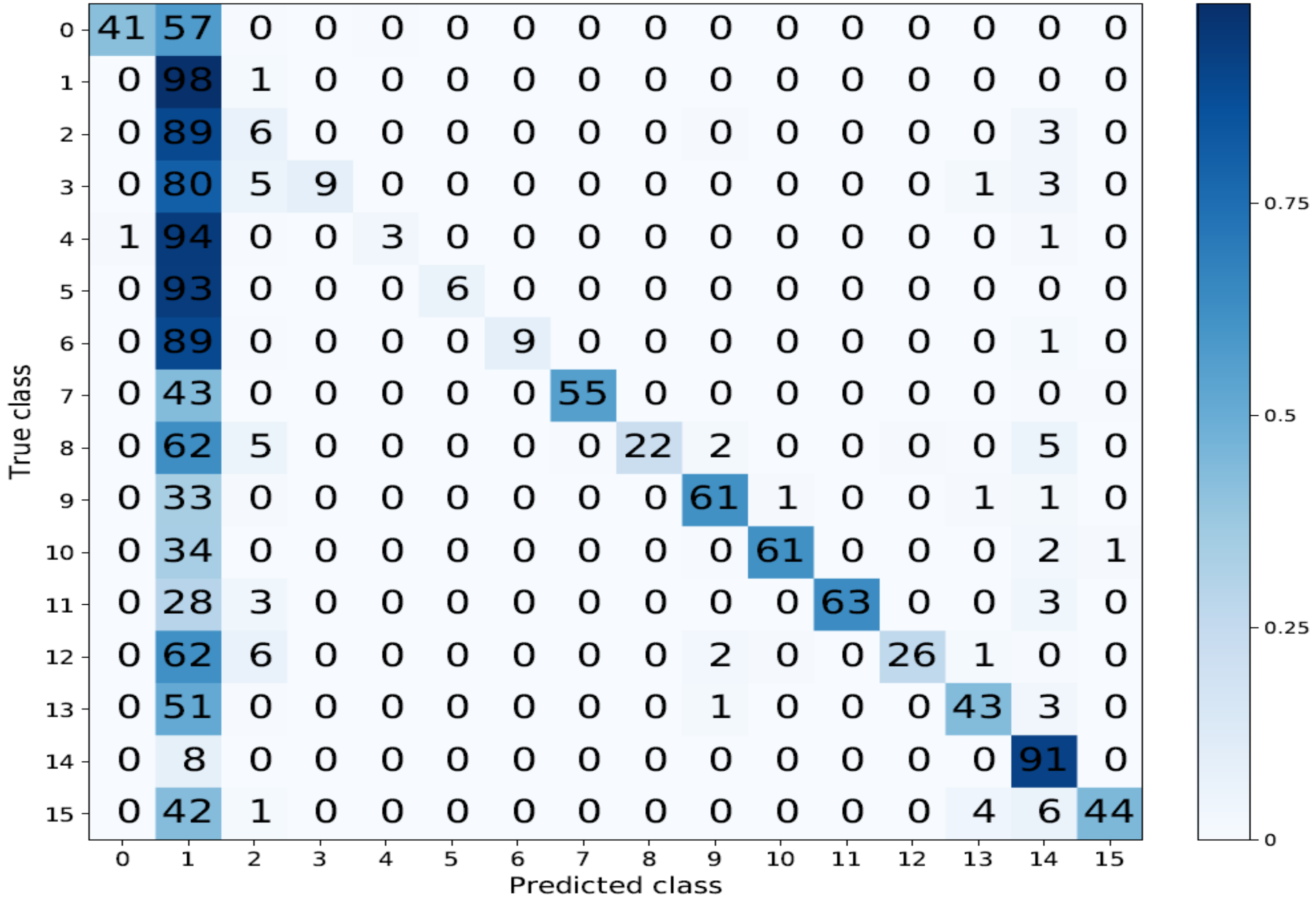
Experimental Results (GTSRB)



pristine

Experimental Results (GTSRB)

- It works BUT less effectively than MNIST



$$\boldsymbol{\alpha = 0.2, t = 1, \Delta_{tr} = 20, f = 6, \Delta_{ts} = 30}$$

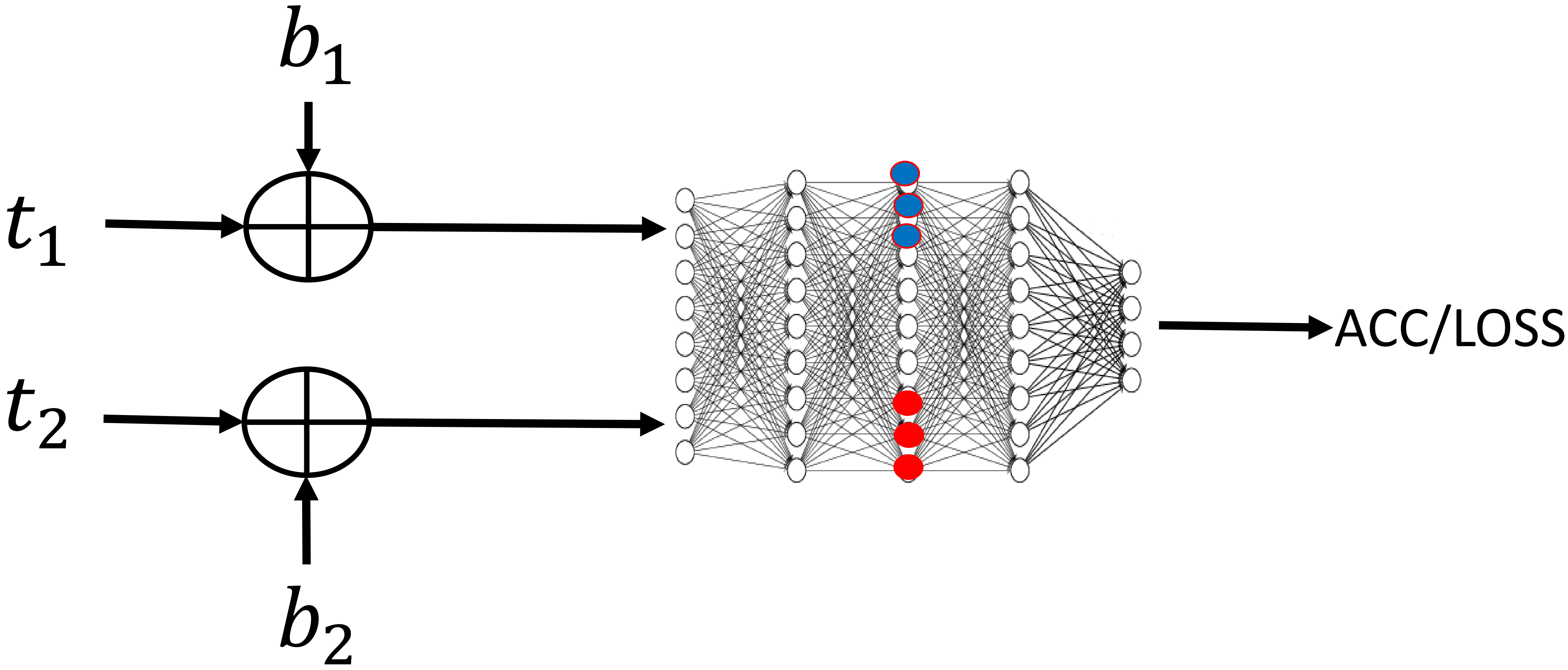
Experimental Results (GTSRB)

Attack success rate (%) in the case of traffic sign classification for different $\Delta_{ts}(\Delta_{tr}= 20, \alpha = 0.2, f = 6)$. *The rate is averaged on the 7 most successfully attacked test signs.*

%/ Δ_{ts}	$t = 1$				$t = 3$				$t = 7$				$t = 13$			
	20	30	40	60	20	30	40	60	20	30	40	60	20	30	40	60
%	73	81	79	83	39	62	76	87	52	71	83	93	26	48	60	78

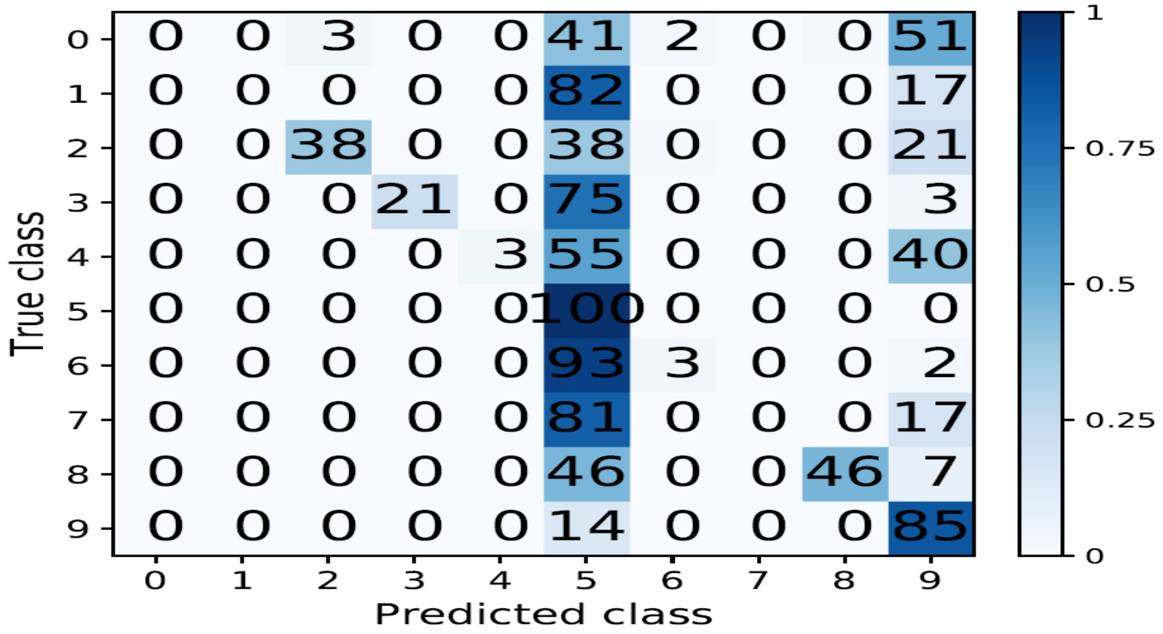
- Attack success rate increases with Δ_{ts}

Experimental Results: Multi-target attack

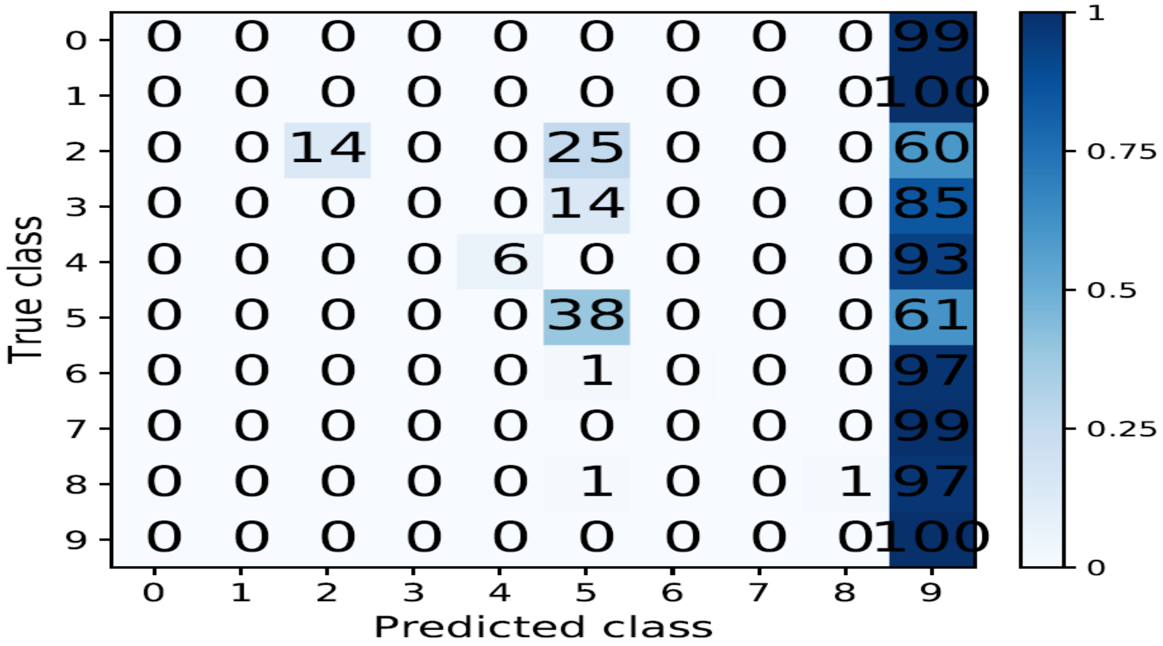


- At test time, we can inject b_1, b_2 or both

Experimental Results: Multi-target attack



Testing by inserting a ramp



Testing by inserting a triangle

- Train by poisoning $t = 5$ with a ramp and $t = 9$ with a triangle, $\alpha = 0.4$, and $\Delta_{tr} = \Delta_{ts} = 30$
- **Multiple-target attacks are also possible**

Conclusions and Future work

- We develop a new backdoor attack without label poisoning
- Price to pay with respect to attacks with label poisoning is the percentage of samples to be attacked
- Experiments on MNIST and GTSRB were successful
- Better development of Backdoor signals
- Investigate more the fact that backdoor could be dataset dependent

Thank you!