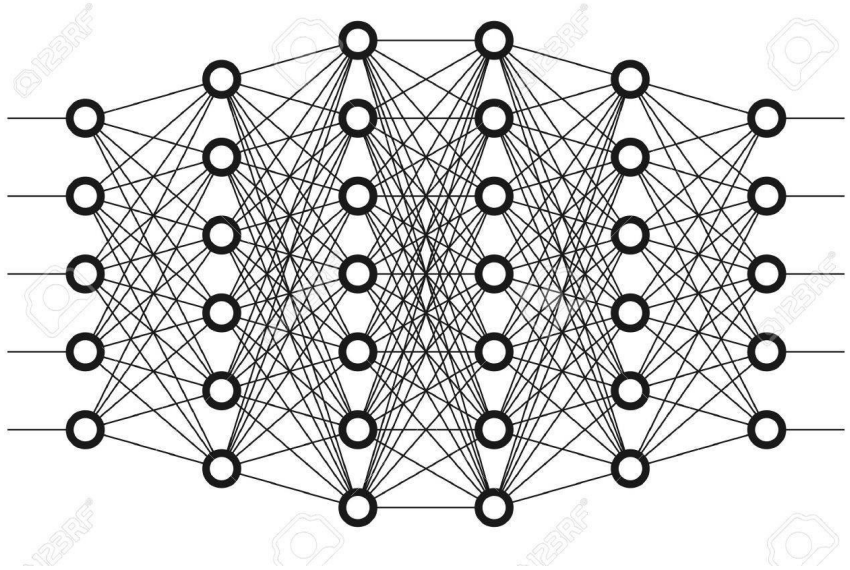
ARE SELF–DRIVING CAR

Evasion Attacks against Deep Neural Networks for Steering Angle Prediction

Alesia Chernikova, Alina Oprea, Cristina Nita-Rotaru and BaekGyu KimyNortheastern University, Boston, MA Toyota ITC, USA

Presented By: Asim Khan
April 23, 2020



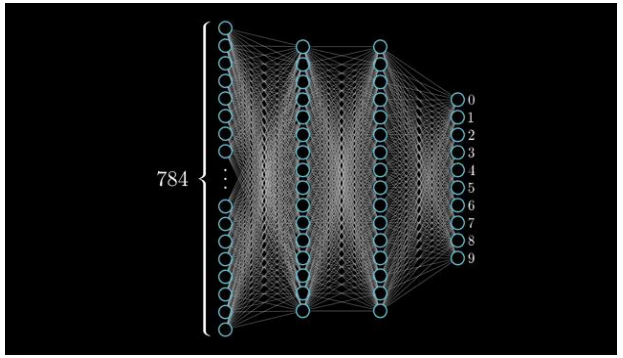
CONTENTS

- Background
- Motivation
- Introduction to Evasion Attacks
- Threat Model
- Attack Algorithm
- Experiments and Results
- Question

BACKGROUND

NEURAL NETWORKS

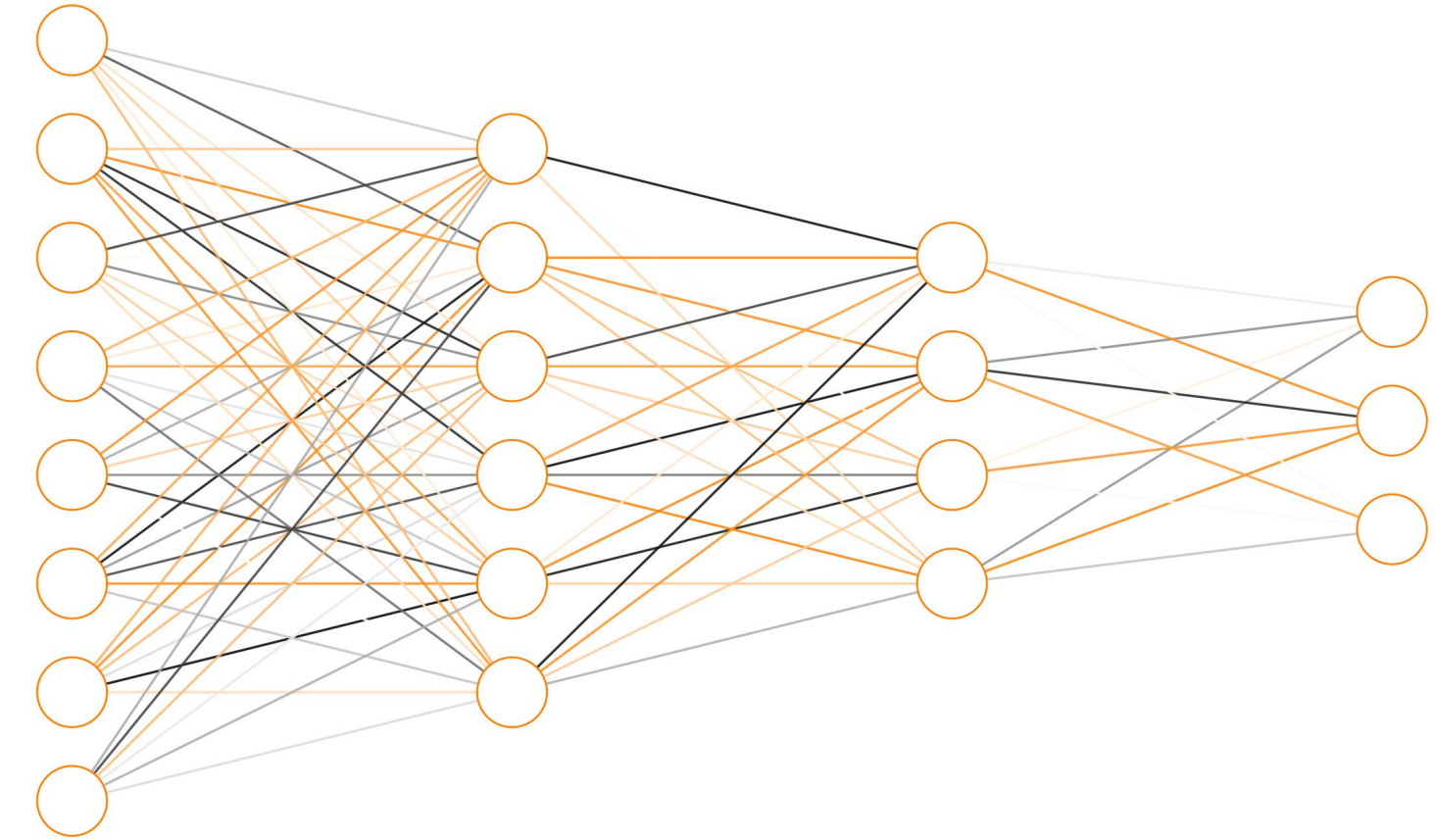
A neural network is a function with trainable parameters that learns a given mapping. For example,



- Classify a handwritten number
- Classify cat and dog in a image
- Bad and good movie review
- Given a file, classify malware and benign

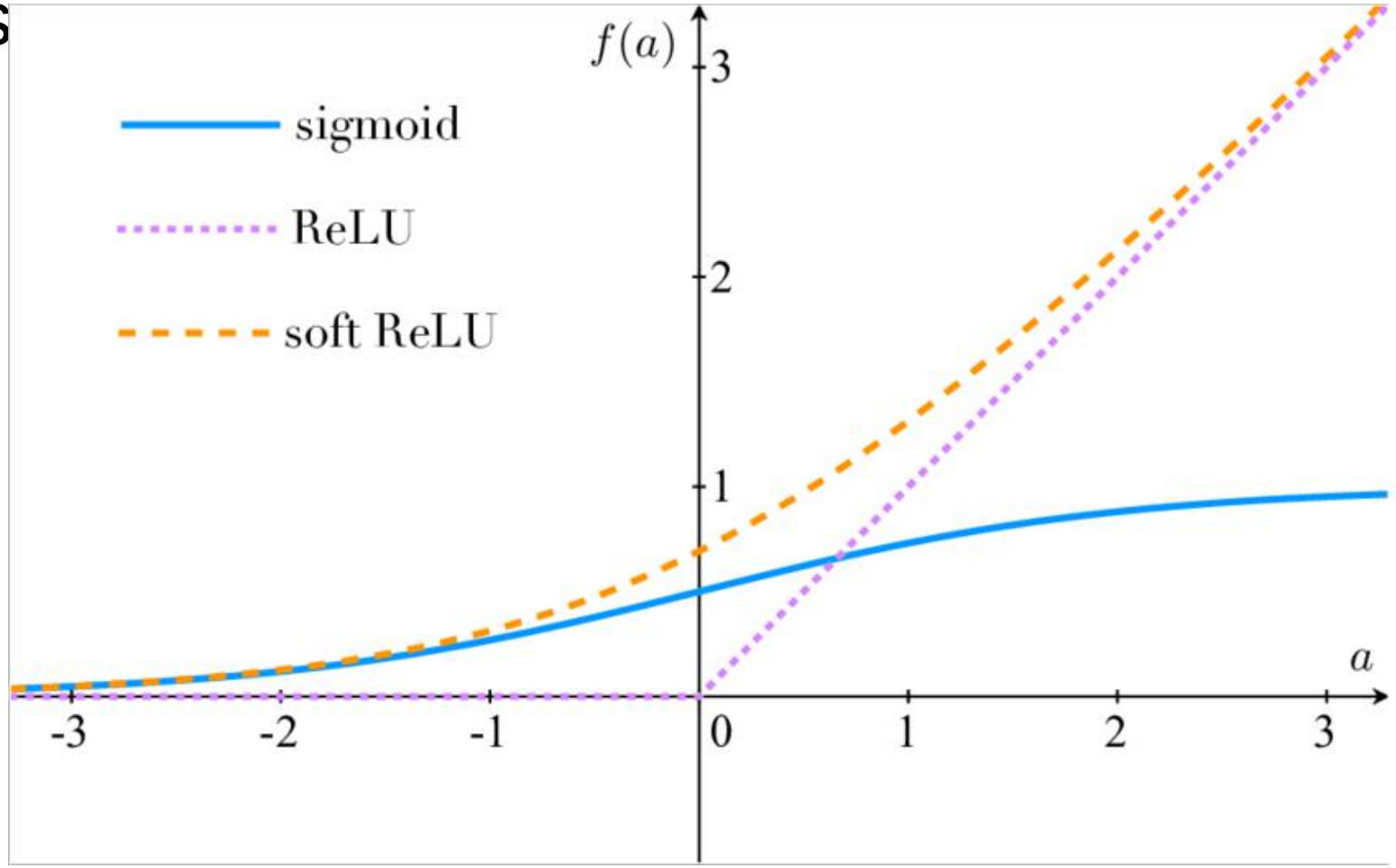
NEURAL NETWORKS

Structure



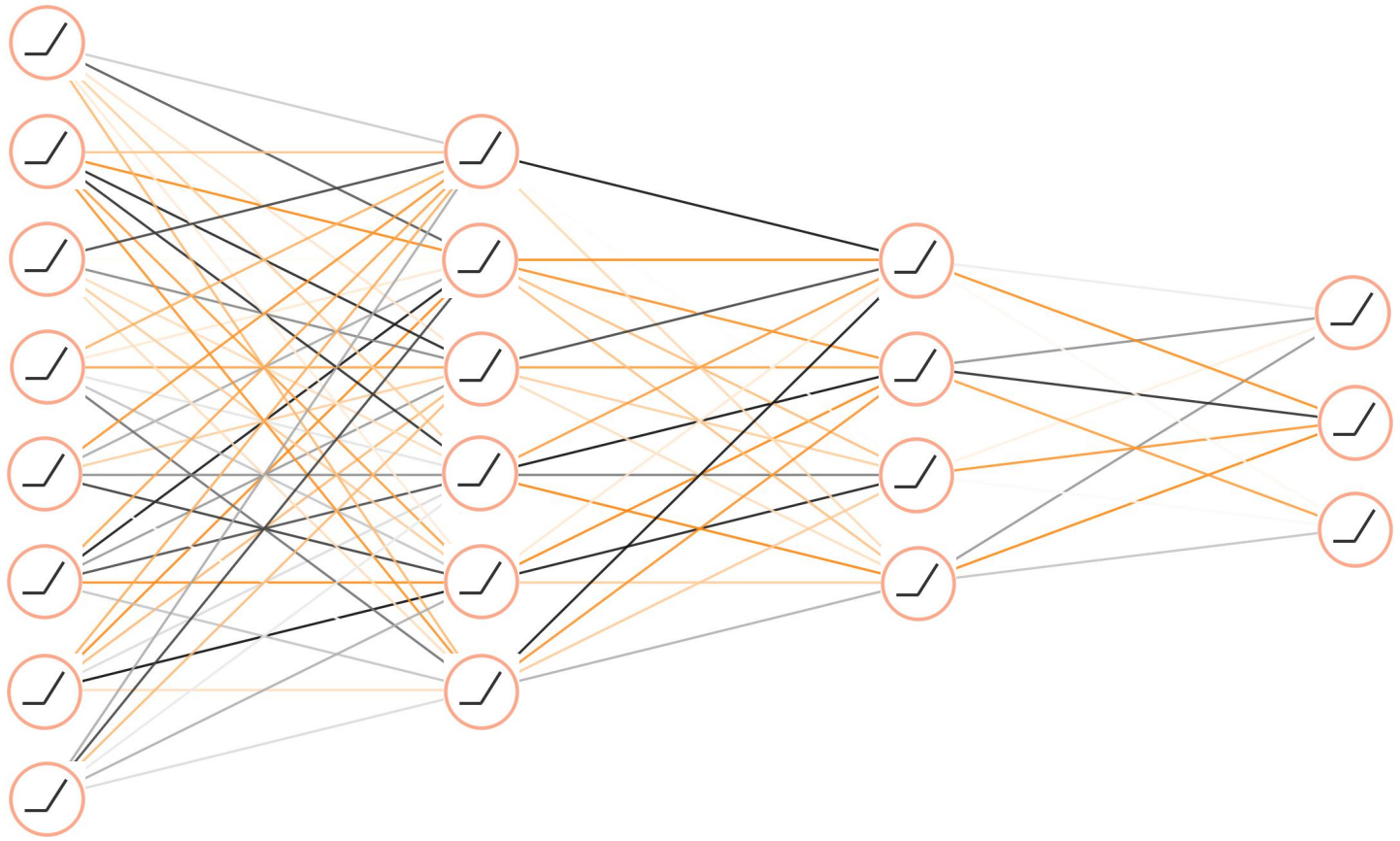
NEURAL NETWORKS

Activation Functions



NEURAL NETWORKS

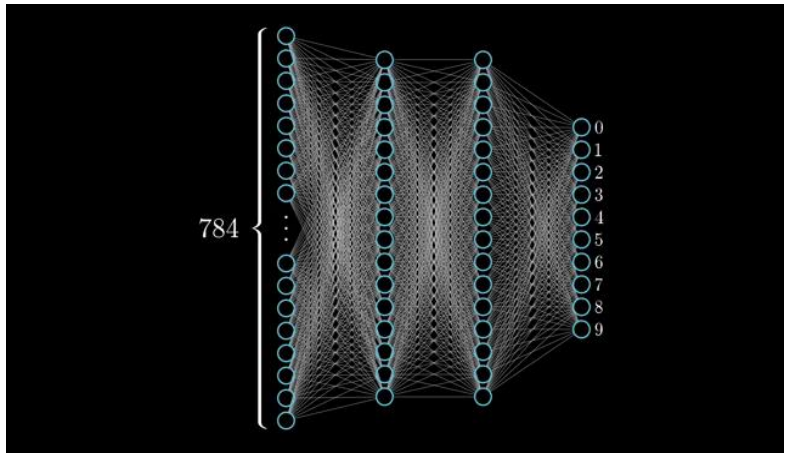
With Activation Function

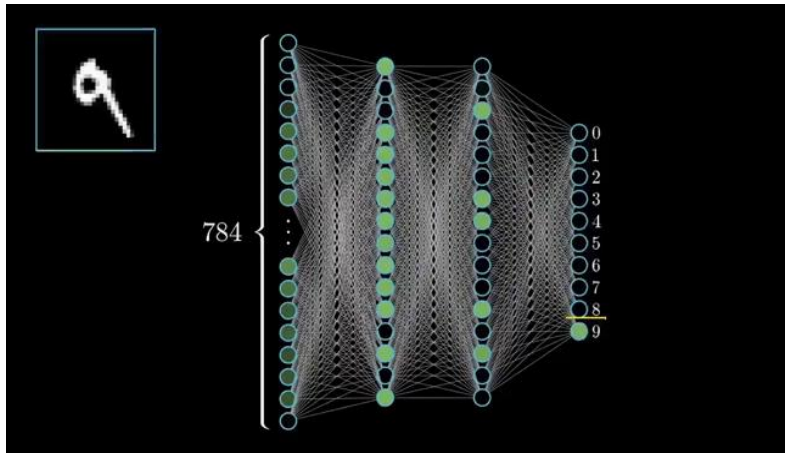


NEURAL NETWORKS

The output of a neural network F(x) is a probability distribution (p,q,...) where

- p is the probability of class 1
- q is the probability of class 2
- …

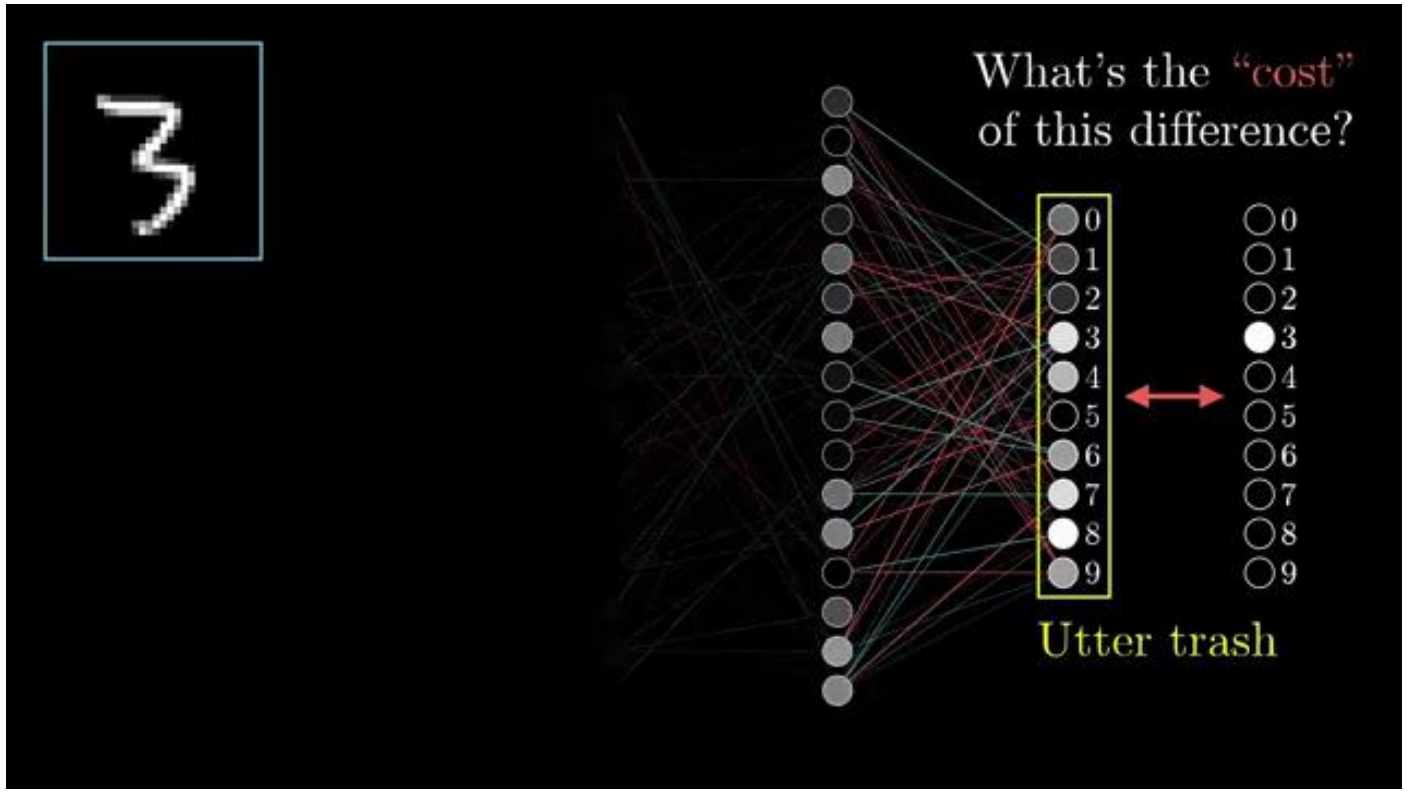




NEURAL NETWORKS

Loss Function

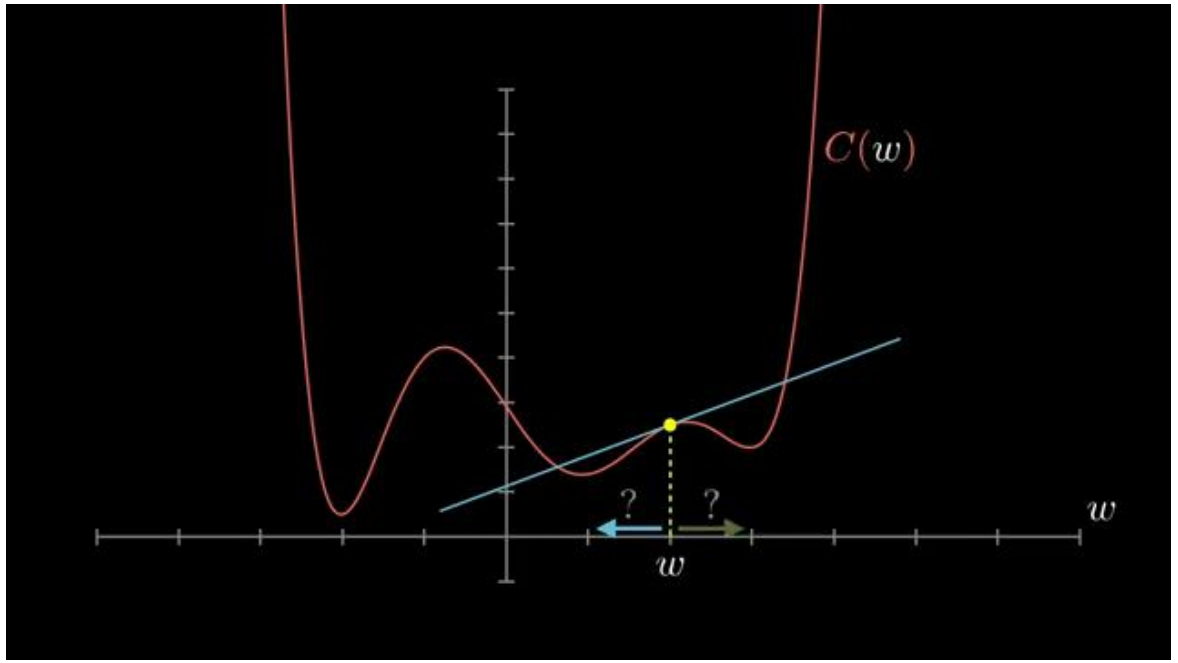
The measure of how accurate the network is. Usually minimized with gradient descent technique.

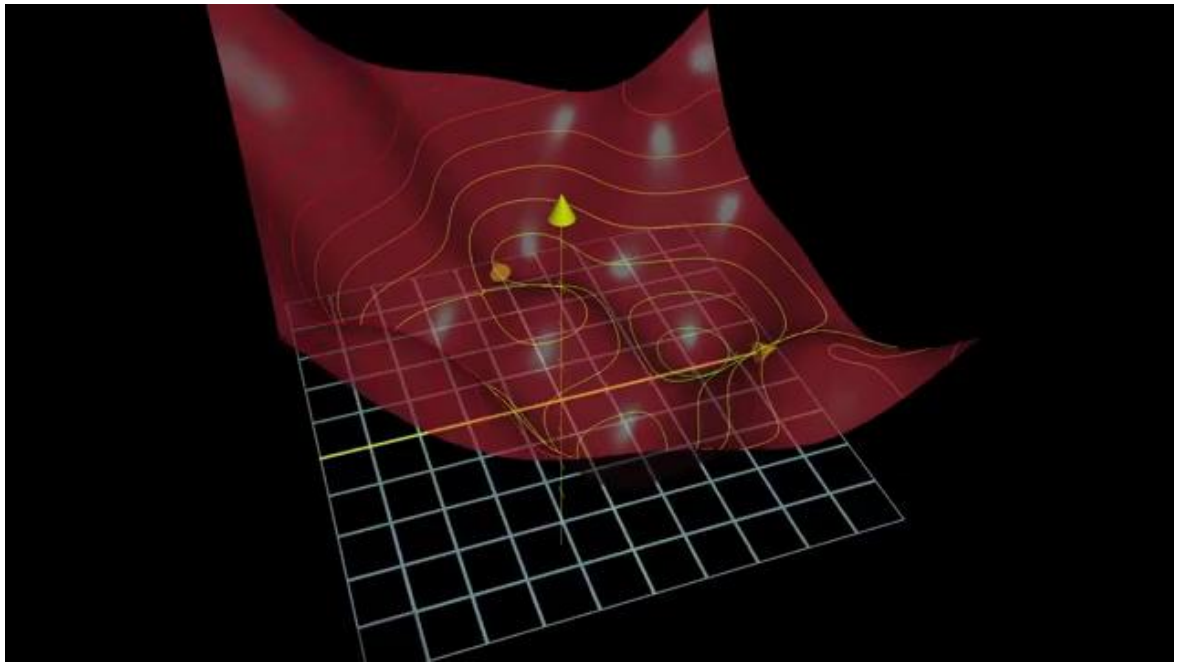


NEURAL NETWORKS

Gradient Descent

Used to minimize the loss function to reasonable small value.





NEURAL NETWORKS

Two important things to notice

- Highly Non-Linear
- Gradient Decent

WHERE IT ALL STARTED

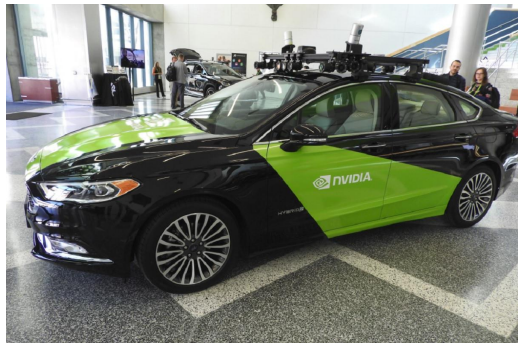
ImageNet

- ImageNet 2011 best result: 75% accuracy No Neural Nets Used
- ImageNet 2012 best result: 85% accuracy Only top submission uses Neural Nets
- ImageNet 2013 best result: 89% accuracy ALL top submissions use Neural Nets
- The best accuracy today is 97%

SELF-DRIVING CARS

ML and NN in self-driving / autonomous cars

- Tremendous potential to make autonomous vehicle a reality
- A lot if companies are in race to produce safe and secure autonomous vehicles
- Some of the companies are mentioned here on right
- NCAT efforts towards autonomous vehicles



Delphi Technologies

FCA FIAT CHRYSLER AUTOMOBILES

Autoliv

Alphabet

SELF-DRIVING CARS

Envisioned ML applications

- Predicting road conditions
- Interaction with other vehicles
- Recognizing risky road condition
- Assisting drivers in decision makin

SELF-DRIVING CARS

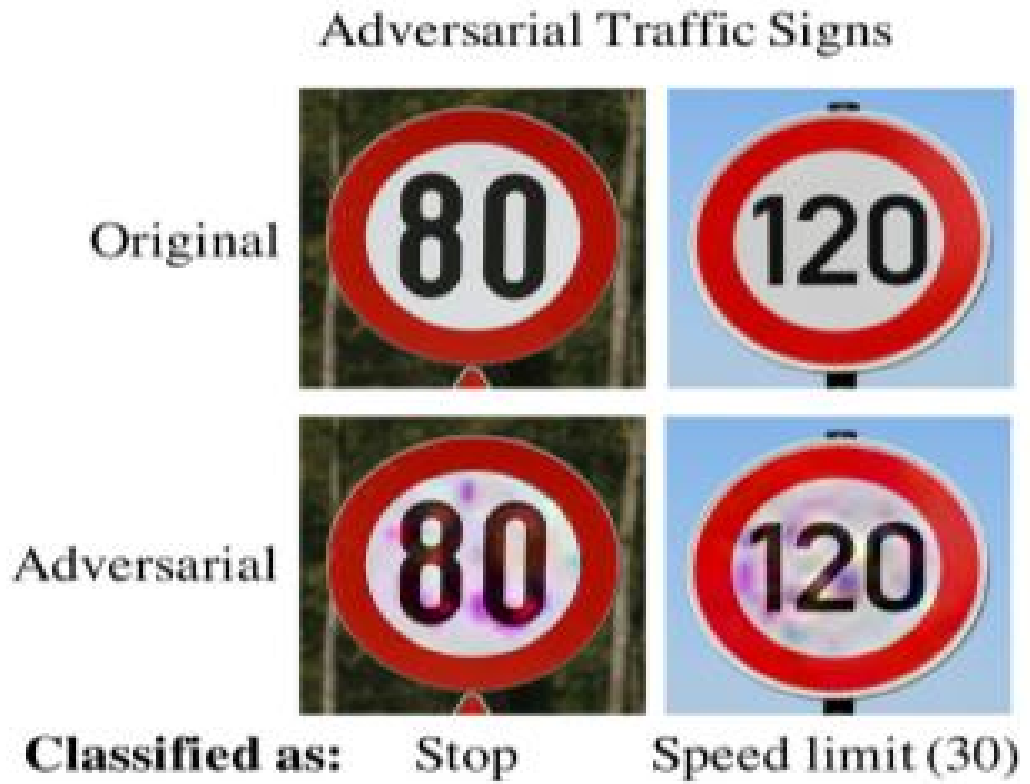
Vulnerabilities in ML and NN

- Defense against different attacks
- Easily fooled
- Getting a lot of attention
- One example of Adversarial attacks in image classification

SELF-DRIVING CARS

Compromised model

- Could misclassify certain thing, one example down below



SELF-DRIVING CARS

Compromised model

- Which can result in this

MOTIVATION

SELF-DRIVING CARS

To make them reality

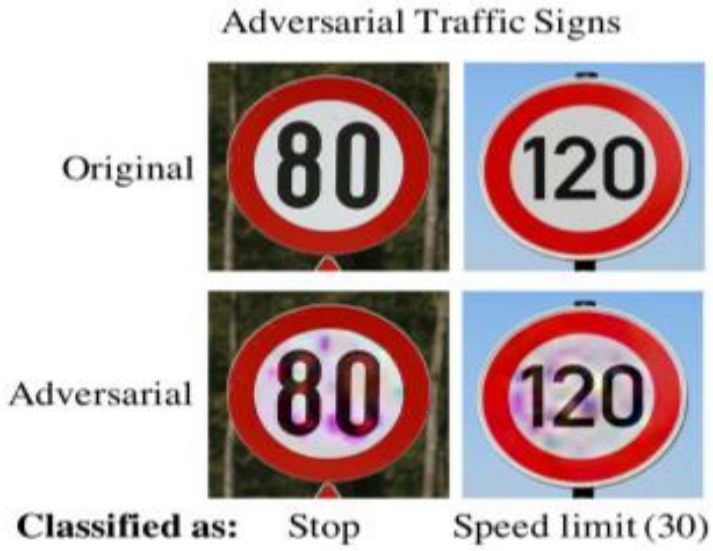
- Safety of highly critical application
- Investigate and design methods for secure ML models.
- Create attacks to continuously test the current models.
- Customers trust

INTRODUCTION TO EVASION ATTACKS

EVASION ATTACKS

Definition

- Given an input X, and any label T
- Find an X′ close to X
- So that F(X′) = Target class
- For example as mentioned before



EVASION ATTACKS

Adversarial Examples

- Gradient descent works very well for training neural networks. Why not for breaking them too ?
- Formulation : given input x , find x' where
  - minimize $d(\mathrm{x}, \mathrm{x}')$ such that,

$$F(x') = T \text{ and } \mathrm{x}' \text{ is valid.}$$

- Gradient descent to the rescue ?
- Non-linearity constraint are difficult

EVASION ATTACKS

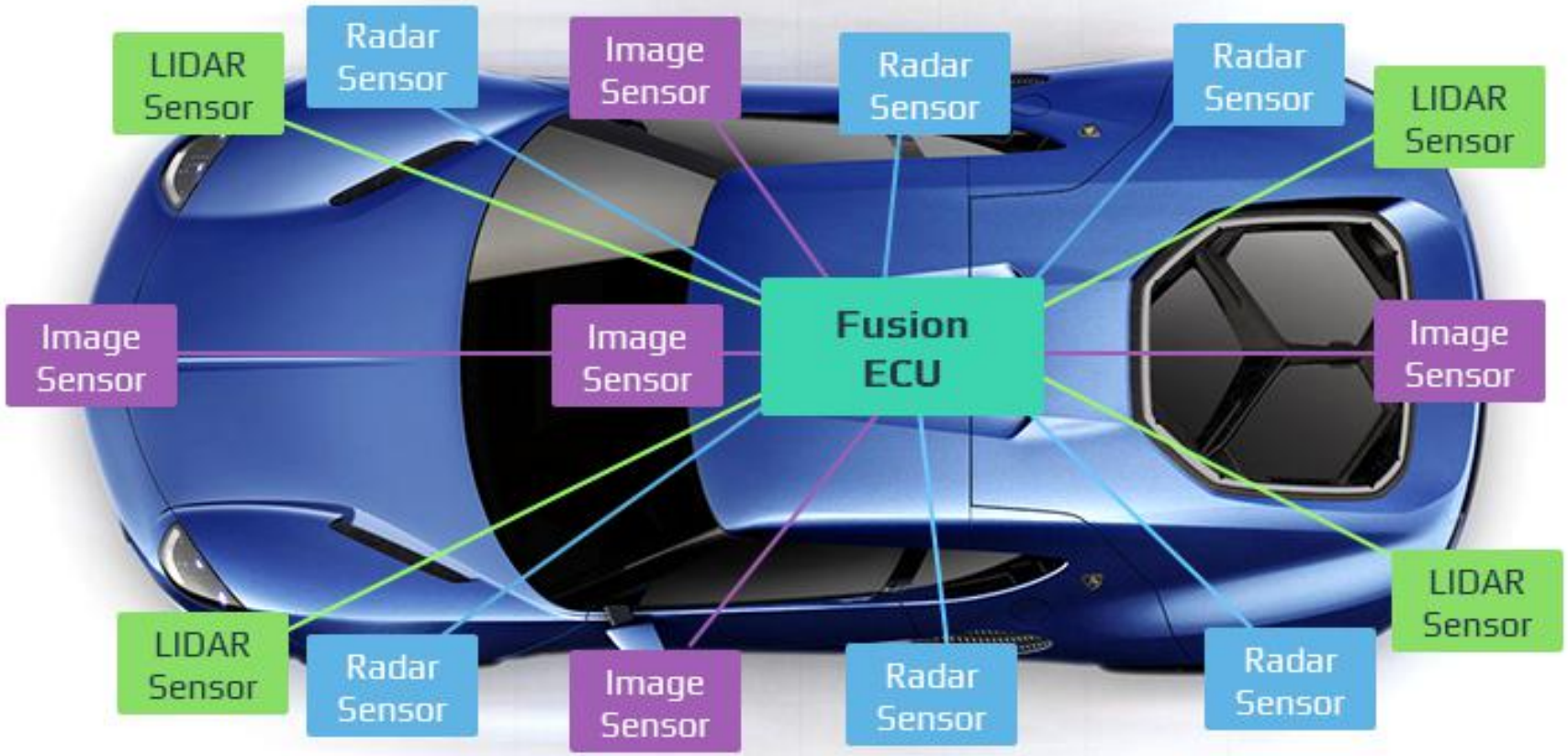
Reformulation

- Moving constraints to objective function
  - minimize $d(\mathrm{x}, \mathrm{x}') + \mathrm{g}(\mathrm{x}')$ such that x' is valid
- Where $\mathrm{g}(\mathrm{x}')$ is some loss function how on close $F(x')$ is to the target T
  - $\mathrm{g}(\mathrm{x}') <= 0 \; if \; F(x') = T$
  - $\mathrm{g}(\mathrm{x}') <= 1 \; if \; F(x')! = T$

BACKGROUND AND THREAT MODEL

SELF-DRIVING CARS

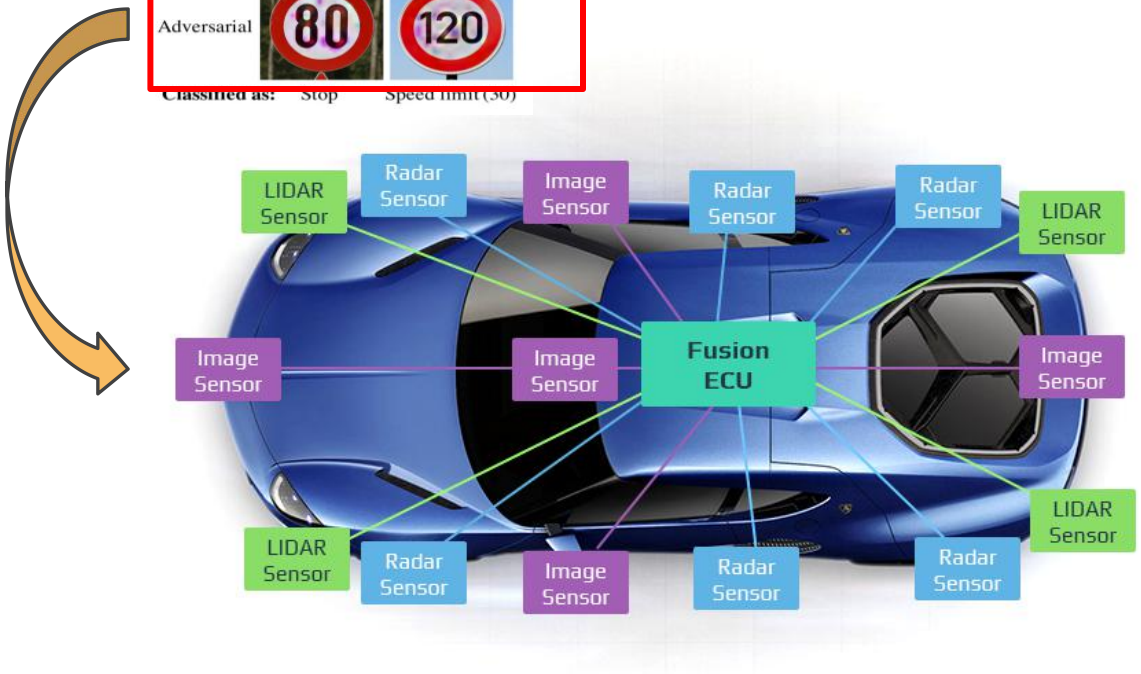
Components



SELF-DRIVING CARS

Threat Model

- Infotainment
- Access to attacker
- Adversarial example from sensor
- Result in misclassification



SELF-DRIVING CARS

Threat Model

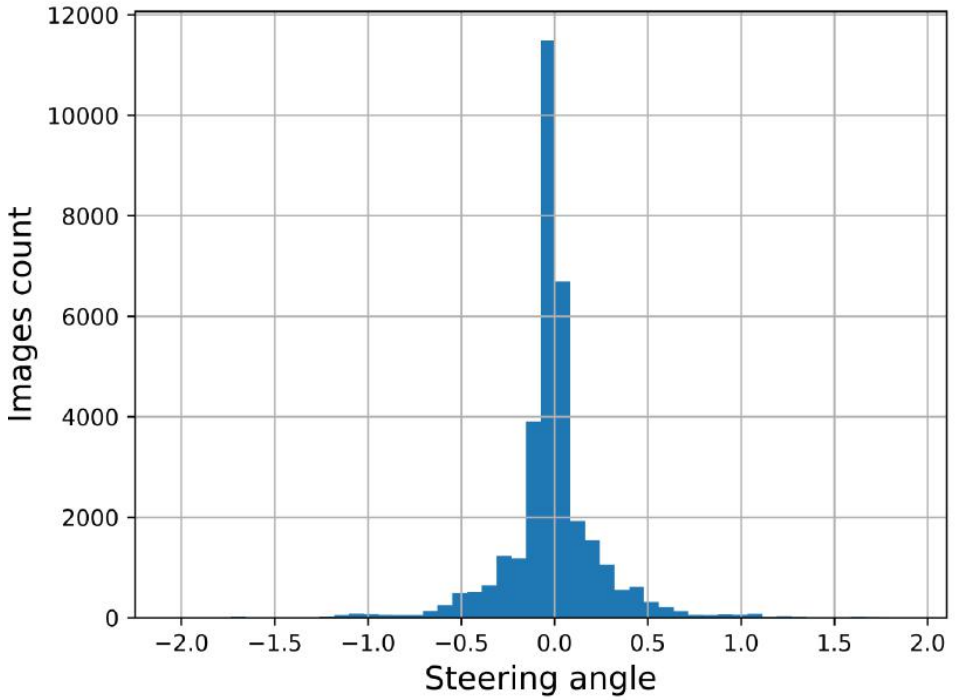
- They use models from Udacity Challenge 2014
- Where the steering angle is predicted based on imagery dataset.
- White-box Attack is considered
- Stealthy perturbation
  - To avoid human suspicion by looking at camera
  - To avoid detection by anomaly detection software

ATTACK ALGORITHM

DNN ARCHITECTURE

Prerequisites

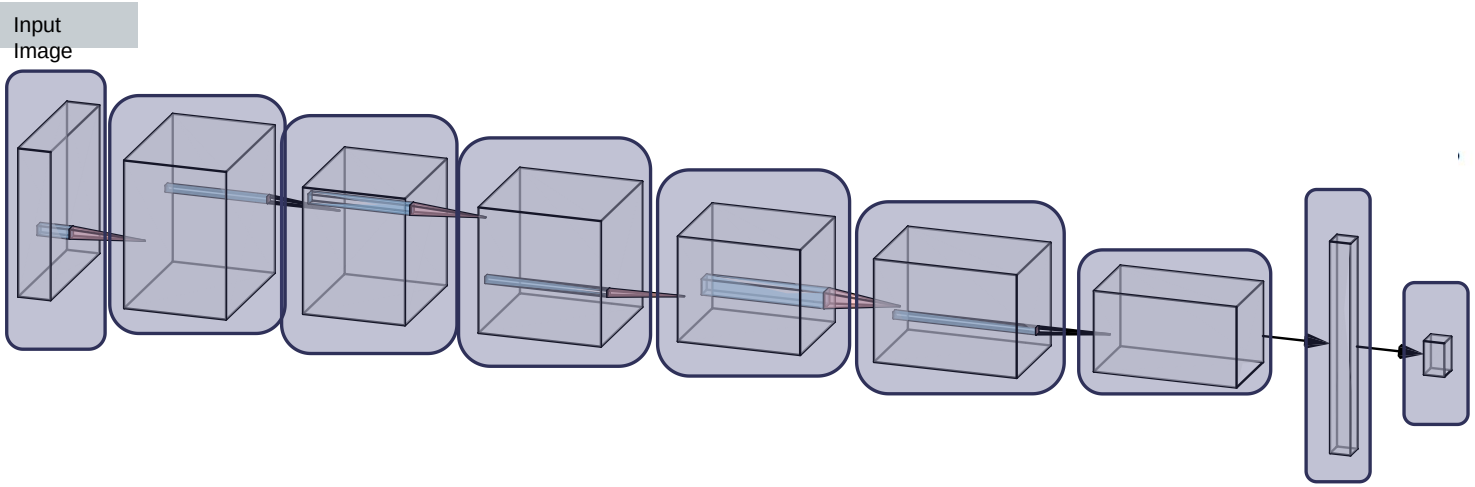
- Classification problem for lane change.
- Angle threshold for left, right and straight.
- Regression problem for angle prediction
- Same as Udacity challenge 2.
- Epoch and NVIDIA Models



DEEP NEURAL NETWORK ARCHITECTURE

Epoch Model

- Classification model



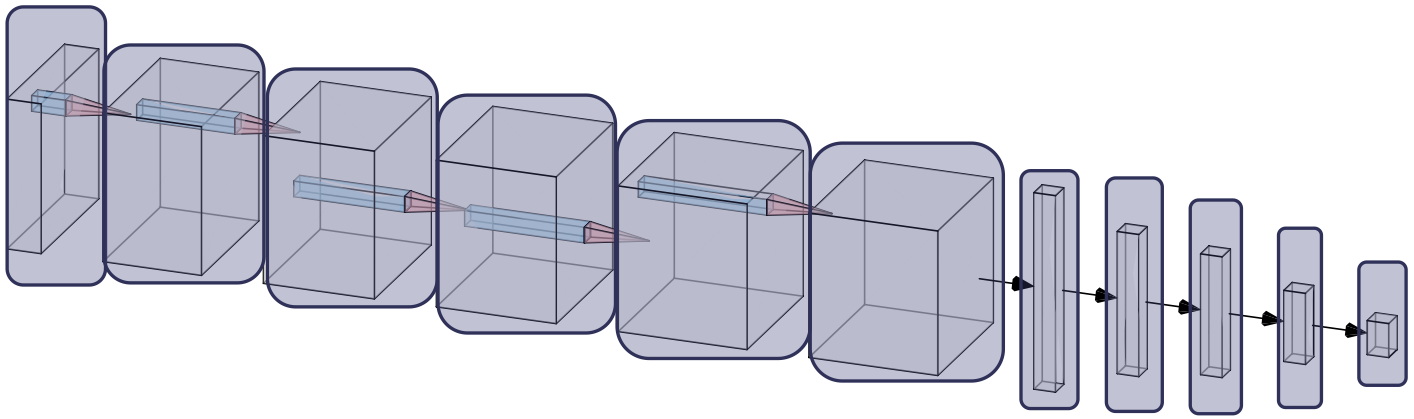
- 25 million total parameters
- For regression the softmax layer is removed

Layer	Architecture and Hyper-parameters
Convolutional + ReLU	32 filters of size $3 \times 3 \times 3$
MaxPooling	Filter 2×2
Dropout	Fraction 0.25
Convolutional + ReLU	64 filters of size $3 \times 3 \times 32$
MaxPooling	Filter 2×2
Dropout	Fraction 0.25
Convolutional + ReLU	128 filters of size $3 \times 3 \times 64$
MaxPooling	Filter 2×2
Dropout	Fraction 0.5
Fully-Connected + ReLU	Neurons 1024
Dropout	Fraction 0.5
Fully-Connected + Softmax	Neurons 3

TABLE I: Epoch Model Architecture

DEEP NEURAL NETWORK ARCHITECTURE

- Classification model



- 467 million total parameters
- For regression the softmax layer is removed

Layer	Architecture and Hyper-parameters
Batch Normalization Layer	
Convolutional + ReLU	24 filters of size $5 \times 5 \times 3$
Convolutional + ReLU	36 filters of size $5 \times 5 \times 24$
Convolutional + ReLU	48 filters of size $5 \times 5 \times 36$
Convolutional + ReLU	64 filters of size $3 \times 3 \times 48$
Convolutional + ReLU	64 filters of size $3 \times 3 \times 64$
Fully-Connected + ReLU	Neurons 582
Fully-Connected + ReLU	Neurons 100
Fully-Connected + ReLU	Neurons 50
Fully-Connected + ReLU	Neurons 10
Fully-Connected + Softmax	Neurons 3

TABLE II: NVIDIA Model Architecture

DNN ARCHITECTURE

Evasion attacks against direction classification

- The concept of same gradient decent
- Perturbation measurement using L_2
- L_2 attack by Carlini and Wagner

$$minimize_x \|x - x_0\| + \lambda \bullet max\{(max_{j \neq t}\{g_j(x)\} - g_t(x)), 0\}$$

- Where x is the perturbed image, x_0 is the original, j and t are the misclassification and proper classification, respectively.
- λ is the hyper parameter to control between success rate and perturbation.

DNN ARCHITECTURE

Evasion attacks against direction classification

- Optimization function for attack algorithm in the paper

$$minimize_x \quad \|\sigma\| + c \bullet f(x+\sigma)$$

$$\text{such that } (x+\sigma)\epsilon[0,1]^d$$

$$f(x+\sigma) = (\max(Z(x+\sigma)_{j\neq t}) - Z(x+\sigma)_t)^+$$

$$where\ t\ -original\ class, j \neq t$$ – adversarial target class

- While σ is the perturbation, $f(x+\sigma)$is the objective function,
- c is the hyper parameter to control between success rate and perturbation.

DNN ARCHITECTURE

Evasion attacks against steering angle regression prediction.

- Main idea is to maximize the MSE predicted response vs true response.
- The following function is optimized to find adversarial image.

$$minimize_x \quad \|\sigma\| - c \bullet g(x+\sigma, y)$$

$$\text{such that } (x+\sigma)\epsilon[0,1]^d$$

$$g(x+\sigma, y) = (\mathrm{F}(x+\sigma) - y)^2$$

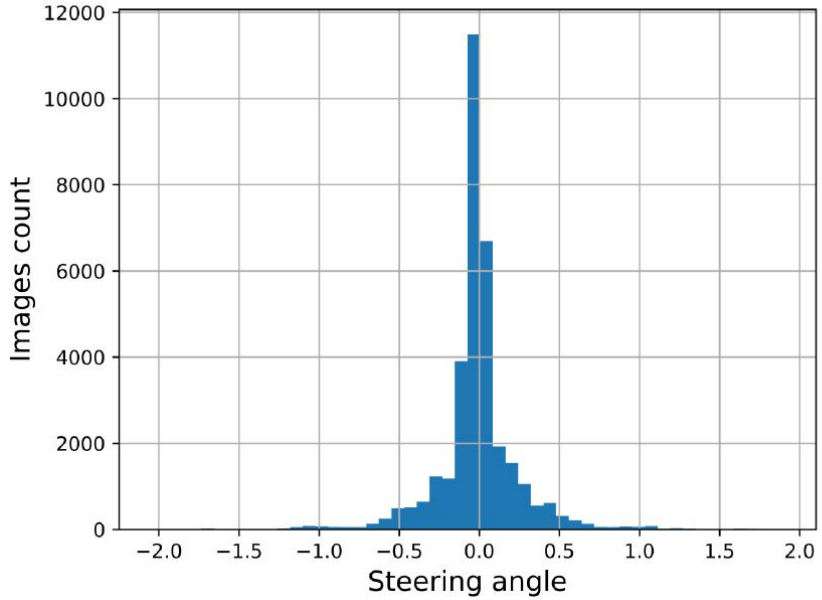
- Where σ is the perturbation, $F(x+\sigma)$is the objective function,
- c is the hyper parameter to control between success rate and perturbation.

EXPERIMENT AND RESULTS

EXPERIMENTS

Database

- Used 33,608 images from Udacity challenge 2.
- Preprocessing
  - Crop them to 640 x 420
  - Resize to 128 x 128 pixels.
  - Setting the classification thresholds at 0.15 in histogram



TRAINING

Training Results

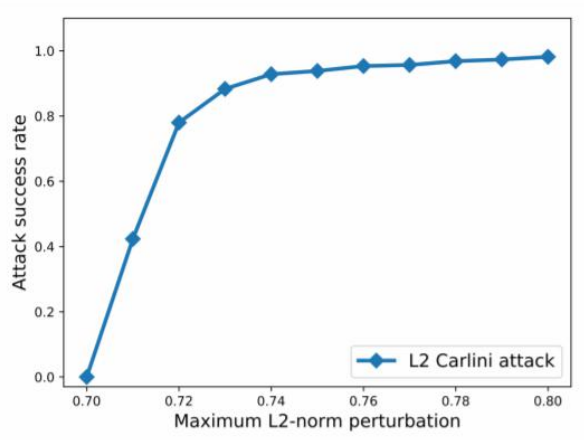
- 10-fold cross validation
- Hyperparameters
- Accuracy
  - Classification accuracy
    - Epoch model is 90%
    - NVIDIA is 86%
  - Regression Accuracy
    - Epoch model is MSE of 0.03

Parameter	Value
Learning rate	0.01
Momentum	0.9
Batch size	128
Epochs	50

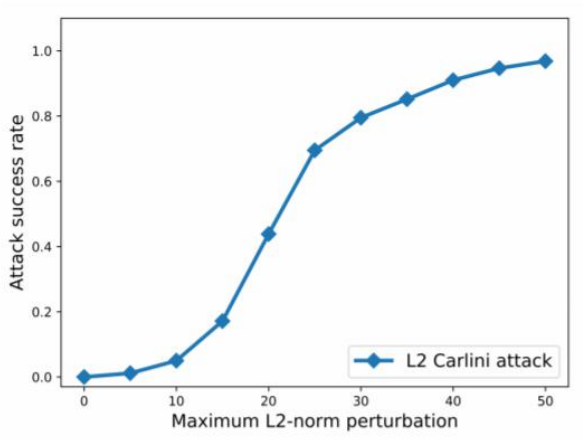
ATTACK RESULTS

Attack results for direction prediction

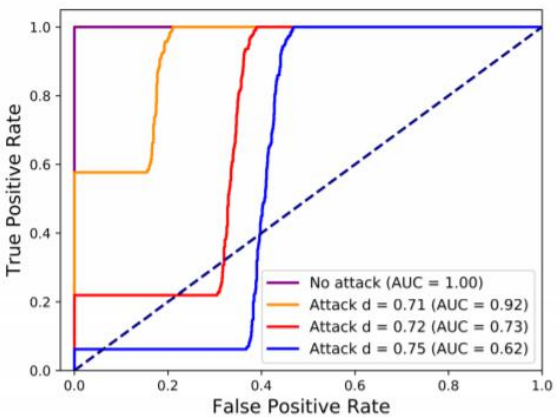
- 300 Images used for 3 classes.
  - Select two 2 values for targeted adversarial class.
- Optimal parameter c is selected using binary search starting from 0.001
- Attack success rate
  - Epoch model with 0.82 L_2 norm
  - Nvidia model with 121 L_2 norm
- ROC
  - Without / With attack
    - From 1 to 0.62 for 0.75 L_2 norm perturbation
    - Attack Time
      - 5 and 25 seconds respectively for Epoch and Nvidia



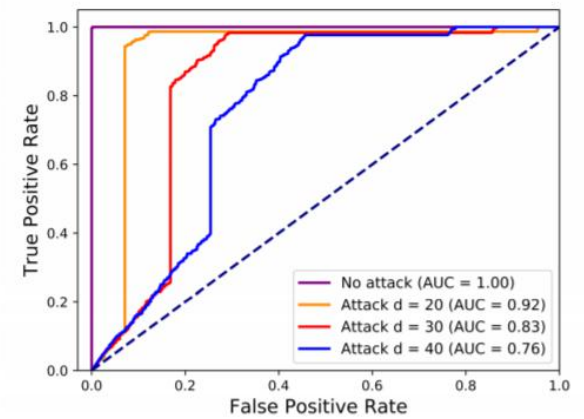
(a) Epoch model



(b) NVIDIA model



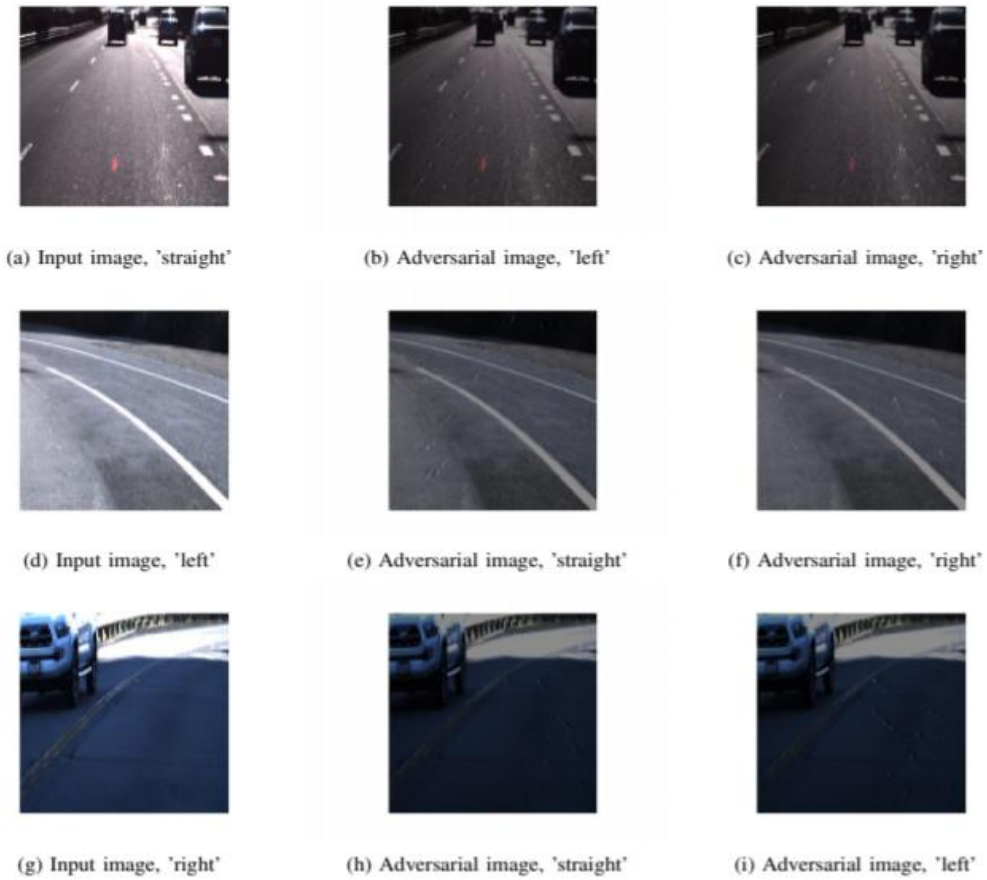
(a) Epoch model



(b) NVIDIA model

ATTACK RESULTS

Original Images vs Adversarial Images



(a) Input image, 'straight' (b) Adversarial image, 'left' (c) Adversarial image, 'right'

(d) Input image, 'left' (e) Adversarial image, 'straight' (f) Adversarial image, 'right'

(g) Input image, 'right' (h) Adversarial image, 'straight' (i) Adversarial image, 'left'

Epoch Model

(a) Input image, 'straight' (b) Adversarial image, 'left' (c) Adversarial image, 'right'

(d) Input image, 'left' (e) Adversarial image, 'straight' (f) Adversarial image, 'right'

(g) Input image, 'right' (h) Adversarial image, 'straight' (i) Adversarial image, 'left'

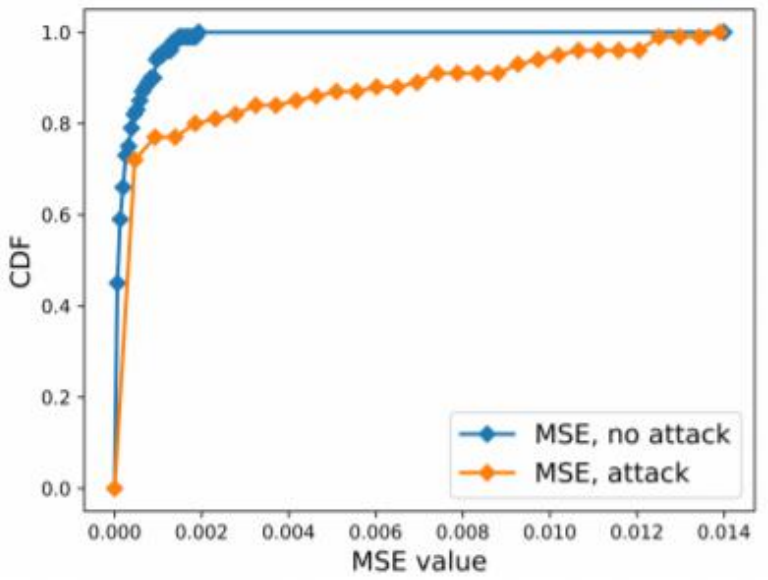
Nvidia Model

ATTACK RESULTS

Attack results for steering angle prediction

- 100 images used for attack.
- Optimal parameter c is selected using binary with best value = 100
- Attack success rate
  - MSE ratio vs L_2 norm
  - 90% of have 0.52 L_2 norm for adversarial images.
- Model performance
  - CDFs of regression model with and without attack.

Percentile	MSE ratio	Perturbation
10%	1.19	0.007
25%	1.38	0.02
50%	2.43	0.05
75%	6.31	0.29
90%	20.88	0.57



ATTACK RESULTS

Original Images vs Adversarial Images

(a) Input image, Predicted angle = -4.25, MSE = 0.0016

(b) Adversarial image, Predicted angle = -2.25, MSE = 0.05

ATTACK RESULTS

Original Images vs Adversarial Images

(a) Input image, Predicted angle = -4.25, MSE = 0.0016

(b) Adversarial image, Predicted angle = -2.25, MSE = 0.05

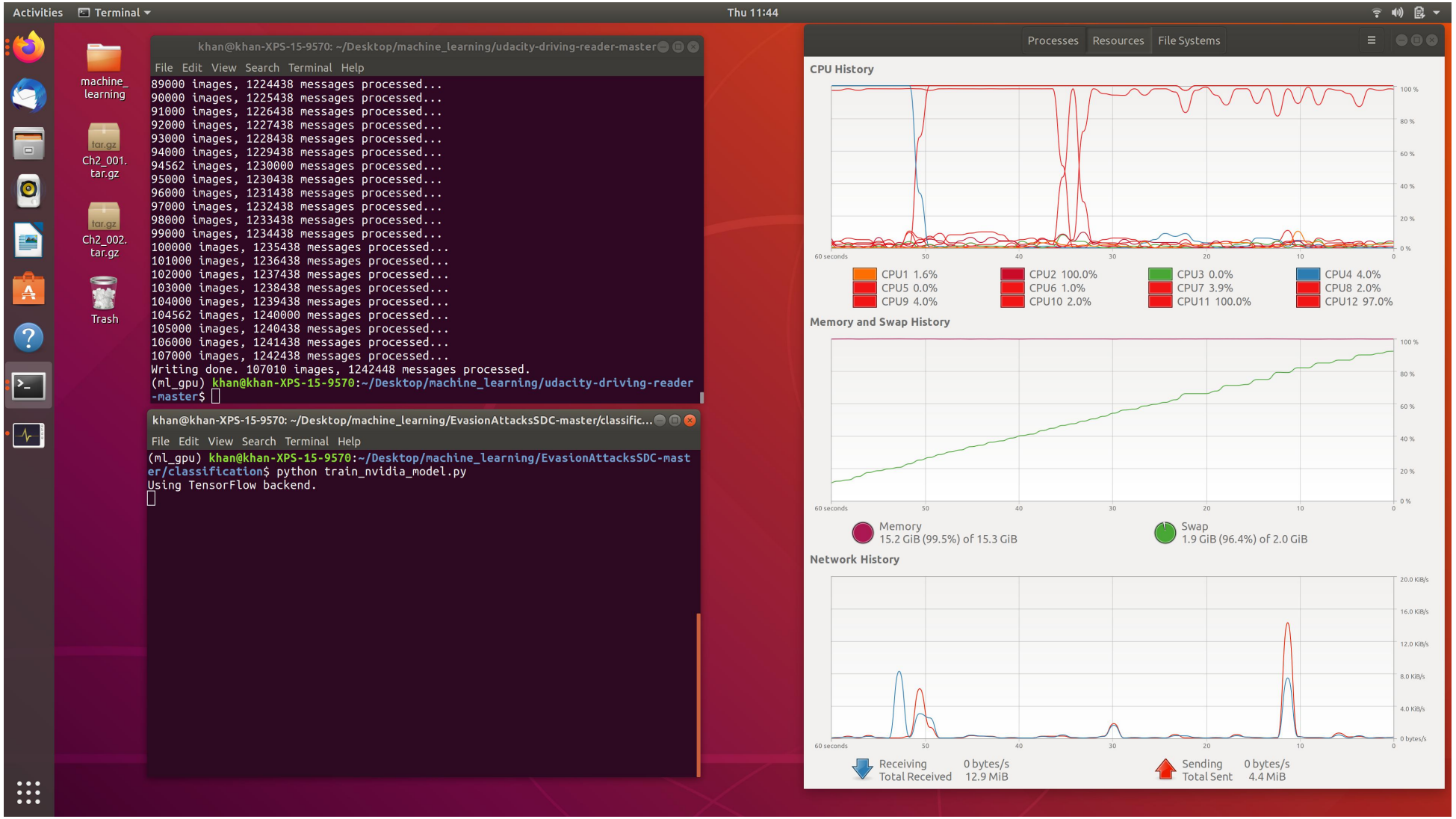
CONCLUSION

CONCLUSION

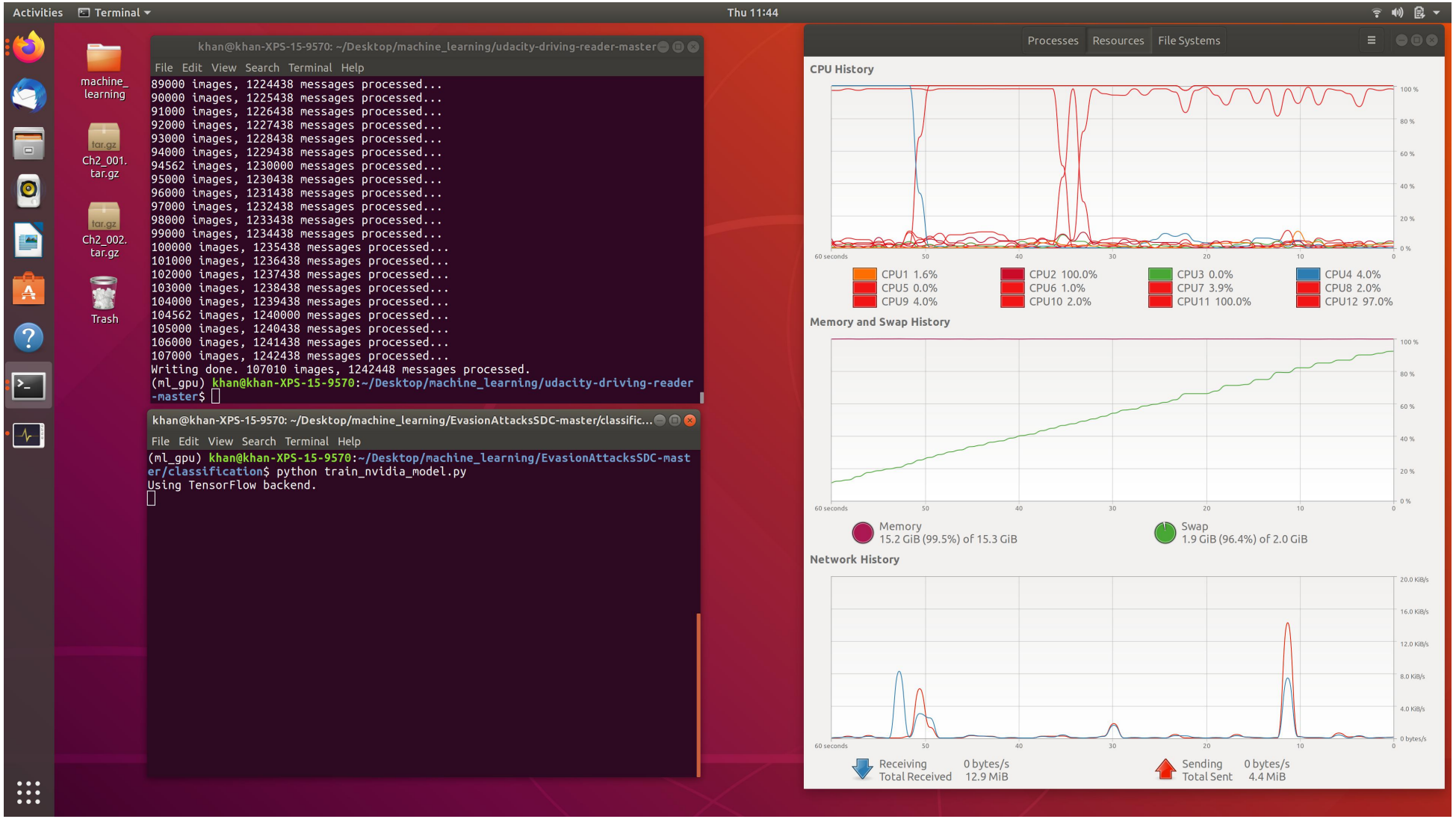
- Open Problem
- Defense mechanism are needed for ML / NN models
- Related Work
  - Defense Distillation Technique

QUESTIONS?

BACKUP SLIDES



BACKUP SLIDES



BACKUP SLIDES