Adversarial Machine Learning

A comprehensive overview

Abhinav Venkataraman

Samsung SDS Research America

Table of contents

1. Introduction
2. Adversarial Image Generation
3. Adversarial Image Training (White Box Attacks)
4. Adversarial Image Generation (revisited)
5. Adversarial Image Training (Black Box Attacks)
6. Applications
7. Titleformats
8. Elements

Introduction

What's an Adversarial Example?

Machine learning models that misclassify examples that are slightly different(sometimes even imperceptible from human eye) from correctly classified examples drawn from the data distribution.

Problem Definition

Regular Neural Network Training

Train a model on a dataset such that you take the gradient of loss function w.r.t model parameters. In this way, you maximize on the score of the correct class.

Problem Definition

Regular Neural Network Training

Train a model on a dataset such that you take the gradient of loss function w.r.t model parameters. In this way, you maximize on the score of the correct class.

Adversarial Learning

Generate an image by doing the following:

Problem Definition

Regular Neural Network Training

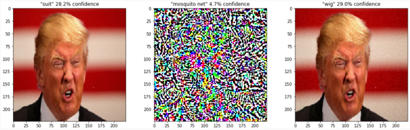
Train a model on a dataset such that you take the gradient of loss function w.r.t model parameters. In this way, you maximize on the score of the correct class.

Adversarial Learning

Generate an image by doing the following:

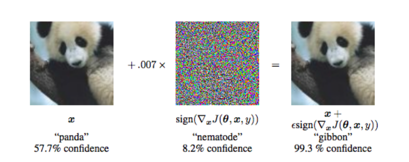
- Wiggle the pixel of an image in the direction of the loss function w.r.t to a class different from the target class.This perturbs the image by a tiny bit but the score of the target class is reduced.

Run the model on the generated image and see the classification result.



Adversarial Image Generation

Adversarial Example



- Digital images often use only 8 bits per pixel so they discard all information below $\frac{1}{255}$ of the dynamic range.
- The classifier does not respond differently to an input x than to an adversarial input($\tilde{x} = x + \eta$) if every element of the perturbation is smaller than the precision of the features($\|\eta\|_\infty = \epsilon$).
- But then this perturbation causes the activation to grow by ϵmn times where m and n are dimensions of weight matrix.

Neural Networks are linear too!

Fooling CNNs

- Deep learning models are meant to express complex non-linear functions.

Neural Networks are linear too!

Fooling CNNs

- Deep learning models are meant to express complex non-linear functions.
- **how are these linear perturbations very effective??**

Fast Gradient Sign Method(FGSM)

Model Definition

Let θ be model parameters, x be the input to the model and let y be the target associated with x then the loss for the model would be defined by $L(\theta, x, y)$.

Fast Gradient Sign Method(FGSM)

Model Definition

Let θ be model parameters, x be the input to the model and let y be the target associated with x then the loss for the model would be defined by $L(\theta, x, y)$.

We can obtain an adversarial example by having a perturbation in the following manner.

$$\eta = \epsilon sign(\nabla_x L(\theta, x, y)) \tag{1}$$

Fast Gradient Sign Method(FGSM)

Model Definition

Let θ be model parameters, x be the input to the model and let y be the target associated with x then the loss for the model would be defined by $L(\theta, x, y)$.

We can obtain an adversarial example by having a perturbation in the following manner.

$$\eta = \epsilon sign(\nabla_x L(\theta, x, y)) \tag{1}$$

An Adversarial example($\tilde{x}$) is given by : $\tilde{x} = x + \eta$.

Fast Gradient Sign Method(FGSM)

Model Definition

Let θ be model parameters, x be the input to the model and let y be the target associated with x then the loss for the model would be defined by $L(\theta, x, y)$.

We can obtain an adversarial example by having a perturbation in the following manner.

$$\eta = \epsilon sign(\nabla_x L(\theta, x, y)) \tag{1}$$

An Adversarial example($\tilde{x}$) is given by : $\tilde{x} = x + \eta$.

Such calculated perturbations make the model confuse over what class to predict.

Question:

How do we solve this problem?

Adversarial Image Training (White Box Attacks)

Possible Approaches

Initial Approach

Add the adversarial examples into the training data. Not super effective, gets the same performance as dropout.

Possible Approaches

Initial Approach

Add the adversarial examples into the training data. Not super effective, gets the same performance as dropout.

FGSM as Regularizer

This method was proved effective with an increased model capacity. Works because such a setting continuously updates adversarial examples. The adversarial function would look like:

Possible Approaches

Initial Approach

Add the adversarial examples into the training data. Not super effective, gets the same performance as dropout.

FGSM as Regularizer

This method was proved effective with an increased model capacity. Works because such a setting continuously updates adversarial examples. The adversarial function would look like:

$$\tilde{L}(\theta, x, y) = \alpha L(\theta, x, y) + (1 - \alpha) L(\theta, x + \epsilon sign(\nabla_x L(\theta, x, y))) \quad (2)$$

Possible Approaches

Initial Approach

Add the adversarial examples into the training data. Not super effective, gets the same performance as dropout.

FGSM as Regularizer

This method was proved effective with an increased model capacity. Works because such a setting continuously updates adversarial examples. The adversarial function would look like:

$$\tilde{L}(\theta, x, y) = \alpha L(\theta, x, y) + (1 - \alpha) L(\theta, x + \epsilon sign(\nabla_x L(\theta, x, y))) \quad (2)$$

When does such a training fail?

- Label leak problem.
- As its a one step process, the adversarial transformation is simple and gets recognized by the model.

Adversarial Image Generation (revisited)

Extensions to FGSM

Iterative Methods

Perform the adversarial image generation n times but clip the perturbation of $\tilde{x}$ to be within the range of ϵ.

Extensions to FGSM

Iterative Methods

Perform the adversarial image generation n times but clip the perturbation of $\tilde{x}$ to be within the range of ϵ.

More efficient attack

Differentiate w.r.t x such that its ground truth is different from y.

Extensions to FGSM

Iterative Methods

Perform the adversarial image generation n times but clip the perturbation of $\tilde{x}$ to be within the range of ϵ.

More efficient attack

Differentiate w.r.t x such that its ground truth is different from y.

Adversarial learning for such attacks

Incorporating the above mentioned ideas in Eq (2) makes the model robust.

Extensions to FGSM

Iterative Methods

Perform the adversarial image generation n times but clip the perturbation of $\tilde{x}$ to be within the range of ϵ.

More efficient attack

Differentiate w.r.t x such that its ground truth is different from y.

Adversarial learning for such attacks

Incorporating the above mentioned ideas in Eq (2) makes the model robust.

Can we build a completely robust network now?

Extensions to FGSM

Iterative Methods

Perform the adversarial image generation n times but clip the perturbation of $\tilde{x}$ to be within the range of ϵ.

More efficient attack

Differentiate w.r.t x such that its ground truth is different from y.

Adversarial learning for such attacks

Incorporating the above mentioned ideas in Eq (2) makes the model robust.

Can we build a completely robust network now?

No!

Extensions to FGSM

Extensions to FGSM

Why not??

Extensions to FGSM

Why not??

How do you defend against attacks that does not have access to the network (black box attacks) ?!?!

Extensions to FGSM

Why not??

How do you defend against attacks that does not have access to the network (black box attacks) ?!?!

More importantly, do such attacks exists?

Extensions to FGSM

Why not??

How do you defend against attacks that does not have access to the network (black box attacks) ?!?!

More importantly, do such attacks exists?

Yes, as its been proven that adversarial examples can transfer between models.

Adversarial Image Training (Black Box Attacks)

Black box attacks

Problem Definition

Lets assume one is able to generate adversarial images based on the above mentioned generation techniques with different models on a given data distribution or dataset D. How do we build models robust to such examples?

Possible Approaches

Min-Max approach

$$h^* = \arg\min_{h \in H} E_{(x,y)\sim D}[\arg\max_{\|\tilde{x}-x\|_\infty \leq \epsilon} L(H(\tilde{x}), y)] \tag{3}$$

This is an universal optimization approach where we minimize the loss function at the same time maximize the adversarial learning.

Ensemble Adversarial Learning

Decouple the adversarial image generation process from learning.
Generate adversarial examples from a set of static pre-trained models.
Augment them with the real data during training.

Applications

- Active learning. it can be seen as model able to request labels on new points.

Titleformats

Metropolis titleformats

metropolis supports 4 different titleformats:

- Regular
- Smallcaps
- allsmallcaps
- ALLCAPS

They can either be set at once for every title type or individually.

Small caps

This frame uses the `smallcaps` titleformat.

Potential Problems

Be aware, that not every font supports small caps. If for example you typeset your presentation with pdfTeX and the Computer Modern Sans Serif font, every text in smallcaps will be typeset with the Computer Modern Serif font instead.

all small caps

This frame uses the `allsmallcaps` titleformat.

Potential problems

As this titleformat also uses smallcaps you face the same problems as with the `smallcaps` titleformat. Additionally this format can cause some other problems. Please refer to the documentation if you consider using it.

As a rule of thumb: Just use it for plaintext-only titles.

ALL CAPS

This frame uses the `allcaps` titleformat.

Potential Problems

This titleformat is not as problematic as the `allsmallcaps` format, but basically suffers from the same deficiencies. So please have a look at the documentation if you want to use it.

Elements

Typography

```
The theme provides sensible defaults to
\emph{emphasize} text, \alert{accent} parts
or show \textbf{bold} results.
```

becomes

The theme provides sensible defaults to *emphasize* text, accent parts or show **bold** results.

Font feature test

- Regular
- *Italic*
- SmallCaps
- **Bold**
- ***Bold Italic***
- **Bold SmallCaps**
- `Monospace`
- *`Monospace Italic`*
- **`Monospace Bold`**
- ***`Monospace Bold Italic`***

Lists

Items

- Milk
- Eggs
- Potatos

Enumerations

1. First,
2. Second and
3. Last.

Descriptions

PowerPoint Meeh.

Beamer Yeeeha.

Animation

- This is important

Animation

- This is important
- Now this

Animation

- This is important
- Now this
- And now this

Animation

- This is really important
- Now this
- And now this

Figures

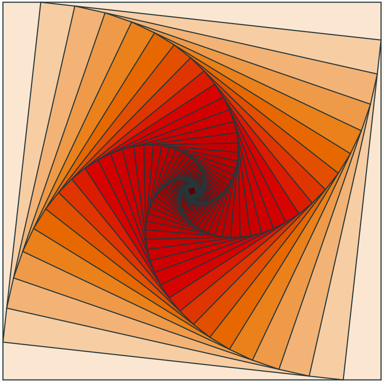


Figure 1: Rotated square from texample.net.

Tables

Table 1: Largest cities in the world (source: Wikipedia)

City	Population
Mexico City	20,116,842
Shanghai	19,210,000
Peking	15,796,450
Istanbul	14,160,467

Blocks

Three different block environments are pre-defined and may be styled with an optional background color.

Default

Block content.

Alert

Block content.

Example

Block content.

Default

Block content.

Alert

Block content.

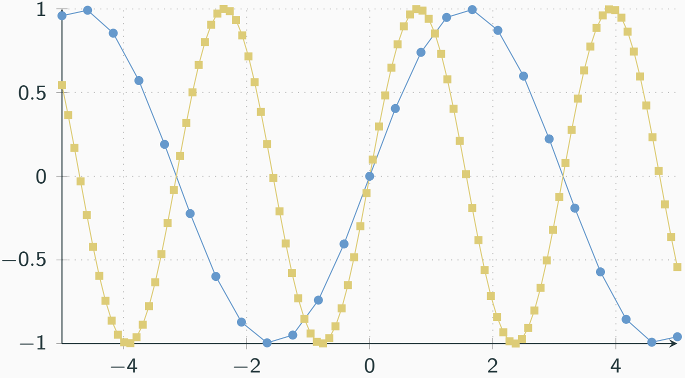
Example

Block content.

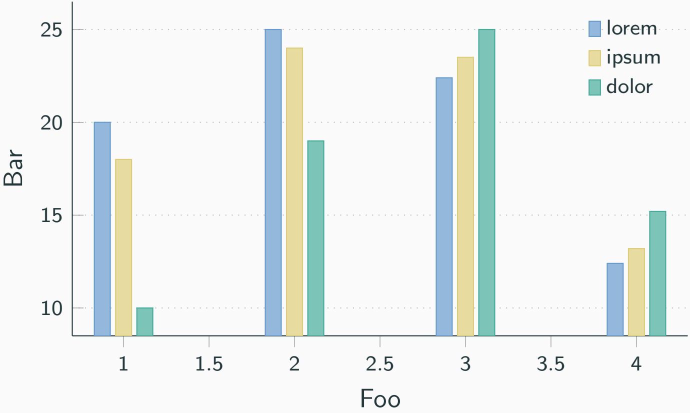
Math

$$e = \lim_{n \to \infty} \left(1 + \frac{1}{n}\right)^n$$

Line plots



Bar charts



Quotes

Veni, Vidi, Vici

Frame footer

metropolis defines a custom beamer template to add a text to the footer. It can be set via

```
\setbeamertemplate{frame footer}{My custom footer}
```

References

Some references to showcase [allowframebreaks] [4, 2, 5, 1, 3]

Conclusion

Summary

Get the source of this theme and the demo presentation from

`github.com/matze/mtheme`

Questions?

Backup slides

Sometimes, it is useful to add slides at the end of your presentation to refer to during audience questions.

The best way to do this is to include the `appendixnumberbeamer` package in your preamble and call `\appendix` before your backup slides.

metropolis will automatically turn off slide numbering and progress bars for slides in the appendix.

References I

P. Erdős.
A selection of problems and results in combinatorics.
In *Recent trends in combinatorics (Matrahaza, 1995)*, pages 1–6.
Cambridge Univ. Press, Cambridge, 1995.

R. Graham, D. Knuth, and O. Patashnik.
Concrete mathematics.
Addison-Wesley, Reading, MA, 1989.

G. D. Greenwade.
The Comprehensive Tex Archive Network (CTAN).
TUGBoat, 14(3):342–351, 1993.

D. Knuth.
Two notes on notation.
Amer. Math. Monthly, 99:403–422, 1992.

References II

H. Simpson.
Proof of the Riemann Hypothesis.
preprint (2003), available at
http://www.math.drofnats.edu/riemann.ps, 2003.