Model Robustness Isn't Security

Sven Cattell

Bsides LV, 2022

About Me

- Founded a startup in this space, still in stealth*.
- Ph.D. in Algebraic Topology from JHU, and a Postdoc in geometric ML.
- Founded the AI Village.
- Formerly at Endgame / Elastic

Table of Contents

Table of Contents

The Adversarial Example Definition Everyone Says

Definition

OpenAI - Adversarial examples are inputs to machine learning models that an attacker has intentionally designed to cause the model to make a mistake; they're like optical illusions for machines.

Definition

TensorFlow - Adversarial examples are specialised inputs created with the purpose of confusing a neural network, resulting in the misclassification of a given input.

The Problems with the Adversarial Example Definitions Everyone Says

1. Any model error can fit the definition, if you squint.
2. This is not close to the definition practitioners mean.
3. If you use this definition, people can sell you snake oil that does nothing.
4. Even if you use the correct definition, people can sell you snake oil that does nothing.

The Robust Definition Everyone Says

Definition

Investopedia - In the world of investing, robust is a characteristic describing a model's, test's, or system's ability to perform effectively while its variables or assumptions are altered.

Definition

Data Robot - A way of modeling that to minimizes your productionalized model from uncertain predictions.

The Problems with the Robust Definitions Everyone Says

1. Both definitions just mean you have a good model that generalizes.
2. This lets them check by wiggling some minor parameters, not actually testing the model on new data.

Terms you Need to Navigate this space

1. **Neighborhood** - a ball around a point.
2. **Adversarial Example** - a point in a *small neighborhood* of a sample that has a different classification than the sample with your classification function.
3. **Point Cloud** - The set of all points in your training set.
4. **Distribution** - the underlying process that generates samples that are in your *point cloud*.
5. **Robust** - If a new point is introduced within a neighborhood of an in-distribution point, it will be classified the same way.

Table of Contents

What's a Neighborhood

There are many ways of calculating "distance" and this impacts the problem in subtle ways.

$$\mathrm{Dist}_2(x, y) = ||x - y||_2 = \sqrt{\sum_{i=1}^{k} (x_i - y_i)^2}$$

$$\mathrm{Dist}_\infty(x, y) = ||x - y||_\infty = \max_{i \in \{1,2,\ldots k\}} |x_i - y_i|$$

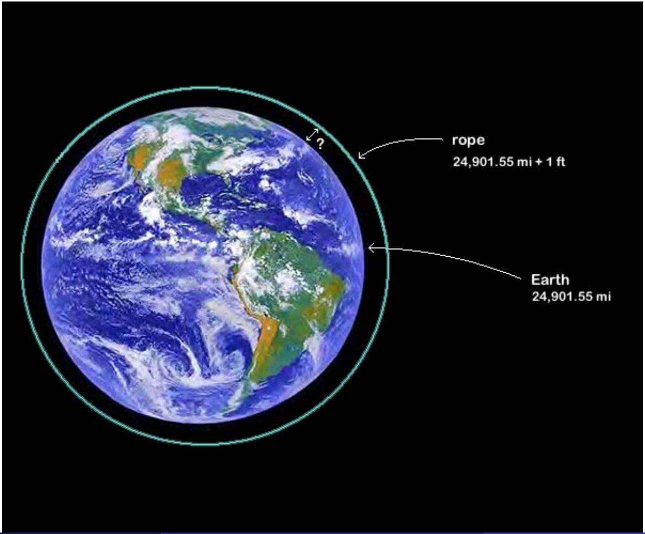
A neighborhood of a point is all points within some chosen δ of your point:

$$\mathrm{Ball}_2^\delta(x) = \{y \in \mathbb{R}^k \text{ s.t. } ||x - y||_2 < \delta\}$$

$$\mathrm{Ball}_\infty^\delta(x) = \{y \in \mathbb{R}^k \text{ s.t. } ||x - y||_\infty < \delta\}$$

Dimensions are weird, 1: String around the Earth

How much more rope do we need to wrap a rope around the earth that is 1 foot above the ground than one that wraps exactly?

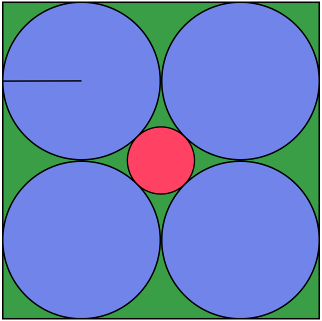


Dimensions are weird, 2: Square Cubed Law

If h is the height of a solid then the surface area grows with $O(h^2)$ and the volume grows with $O(h^3)$. Elephant ears combat this:

Dimensions are weird, 3: Sphere Packing

In dimension d inscribe 2^d d-spheres in the unit d-cube. Then inscribe a d-sphere that just touches each of the d-spheres.
What happens to the center sphere as we increase d?



Mathematicians get this wrong!

Dimensions are weird, 4: MNIST Degrees of Freedom

Each MNIST digit is a square image of 28x28 8-bit pixels, so each pixel is one of 256 values. If I am allowed to perturb each pixel in the image by 1 value I have

$$3^{(24^2)} = 3^{784} \approx 2^{1242}$$

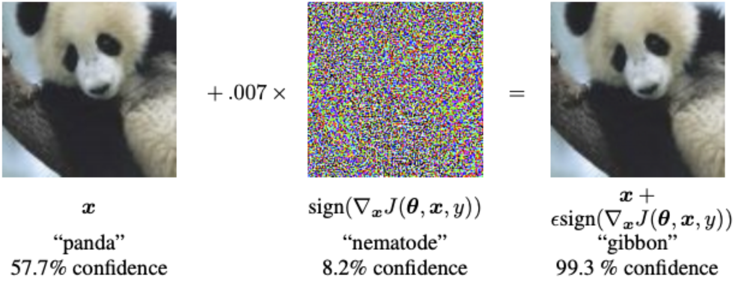
possible perturbations.

What's a Neighborhood: Takeaways

1. The volume of high dimensional neighborhoods grows exponentially with the dimension.
2. This messes up machine learning and is known as the **Curse of Dimensionality**.

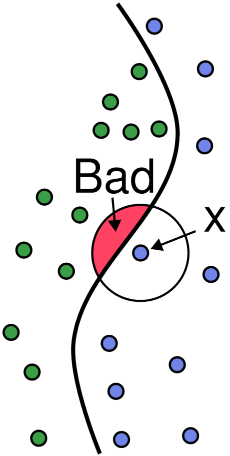
Table of Contents

The Legally Required Panda-Gibbon



[GSS14]

A Definition for Adversarial Examples



Any input within a small ball around our target point that changes the output is an adversarial example for that target point.

Fully Formal Definition of Adversarial Examples

Definition

For

1. a classifier $f : \mathbb{R}^m \to \{1, \ldots k\}$,
2. a point $x \in \mathbb{R}^m$,
3. and a target label $l \in \{1, \ldots k\}$

An **adversarial example** for x with target l within radius δ is any $y \in [0, 1]^m$ such that,

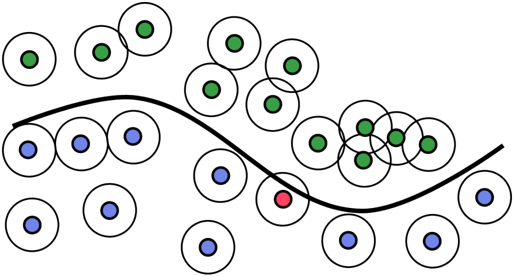
1. $||x - y||_2 < \delta$,
2. $f(y) = l$.

Robustness

Definition

A model f is δ robust on a set of points X if for all $x \in X$ and $r \in \mathbb{R}^n$ such that $||r||_2 < \delta$, then:

$$f(x) = f(x + r)$$



The red point makes this "model" not robust with the given radius.

Table of Contents

The Issue with Robustness

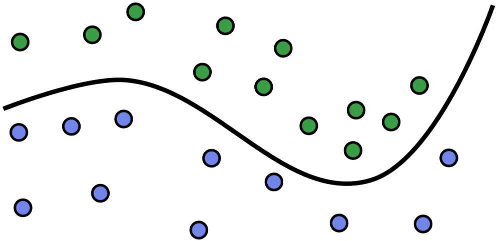
1. Data Just Moves

2. It's Impossible to Check

3. It's Impossible to Make

3. It Lowers Accuracy

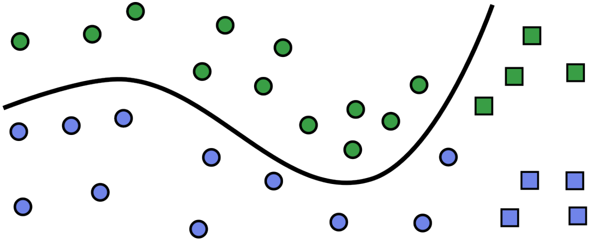
The Issue with Robustness: Data just moves



We can train a robust model on the green and blue points.

Think of this as all the data collected up until you have to train and deploy the robust model.

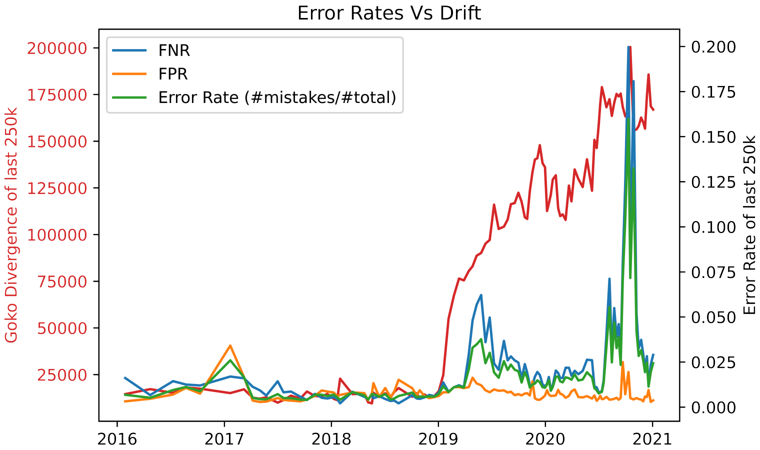
The Issues with Robustness: Data just moves



A month later there's more data.

And... your robust model misclassifies the new green points as blue.

The Issues with Robustness: Data just moves



This is a model trained on malware data up until 01/01/2019. The error rates (legend + axis on right) explode in March. [Cat21]

The Issues with Robustness: It's Impossible to Check

An MNIST model f is δ robust asserts that they know that the model does the correct thing for each input.

The volume of space checked can be converted into a measure of information familiar to hackers, bits:

δ	Maximum Bits of Information	
	MNIST L_2	MNIST L_∞
1	10.6	**1242.6**
2	37.9	1820.4
3	77.0	2201.0
4	125.4	2485.2
5	181.1	2712.2
6	242.9	2901.1
7	**309.3**	3063.0
8	379.5	3204.6

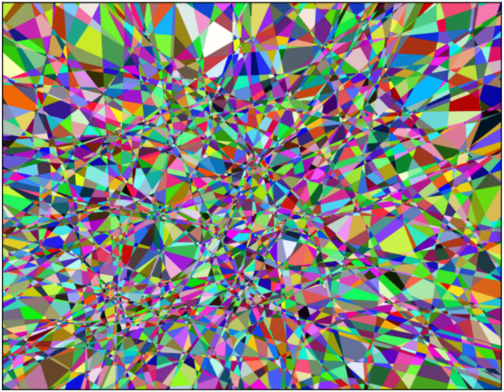
The Issues with Robustness: It's Impossible to Check

So, with a radius of 7 for a MNIST, a brute force check for the L_2 ball needs more compute than brute forcing AES Encryption with 256bit keys.

The radius is normally more than 70.

A smarter check may use the geometry of the neural network to check, but at that radius all bets are off.

The Issues with Robustness: It's Impossible to Check



A 2D slice of a neural network trained on MNIST.
Each colored polygon is a linear region in MNIST. [HR19]

The Issues with Robustness: It's Impossible to Check

Using Hanin and Rolnick's estimates the bits needed to search on a reasonable network for MNIST is:

δ	Maximum Bits of Information	
	Estimate w/ H&R	Naive Estimate
4	54.2	125.4
5	84.3	181.1
6	112.1	242.9
7	144.6	**309.3**
8	181.1	379.5
9	221.1	452.4
10	**264.0**	527.2
11	304.4	602.9
12	347.5	678.7

The Issues with Robustness: It's Impossible to Make

People make models "robust" by adversarially training them [KM22]:

Algorithm 1 Robust Training

Select minibatch B, initialize gradient vector $g = 0$

for $(x, y) \in B$ **do**

  Find an attack perturbation r^* by (approximately) optimizing:

$$r^* = \max_{||r||<\epsilon} l(f_\theta(x + r), y)$$

  Add gradient at r^*:

$$g = g + \Delta_\theta l(f_\theta(x + r^*), y)$$

end for

Update parameters θ:

$$\theta = \theta - \frac{\alpha}{|B|} g$$

The Issues with Robustness: It's Impossible to Make

People make models "robust" by adversarially training them [KM22]:

Algorithm 2 Robust Training

1: Select minibatch B, initialize gradient vector $g = 0$

2: **for** $(x, y) \in B$ **do**

3: **Find an attack perturbation** r^* by (approximately) optimizing:

$$r^* = \max_{||r||<\epsilon} l(f_\theta(x + r), y)$$

4: Add gradient at r^*:

$$g = g + \Delta_\theta l(f_\theta(x + r^*), y)$$

5: **end for**

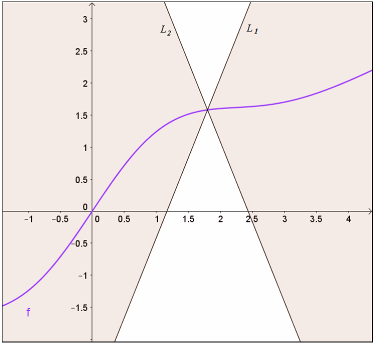
6: Update parameters θ:

$$\theta = \theta - \frac{\alpha}{|B|} g$$

The Issues with Robustness: It's Impossible to Make*

Robustness can be quantified in a different way: the classifier is Lipschitz continuous [MS22].

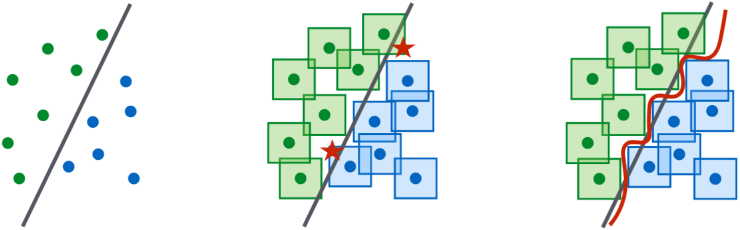
1d Lipschitz Function [Tas22]:



I don't think this is ready for production. A lot of the techniques that try to guarantee this rely on regularization which isn't mathematically sound.

The Issues with Robustness: It Lower Accuracy

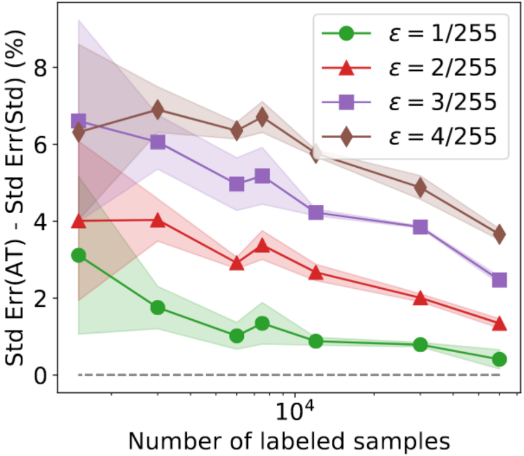
The classifier needs to adapt and may pick a non-optimal decision boundary [MMS+17]:



There even may be test points of one class within the robustness ball of a different class.

The Issues with Robustness: It Lowers Accuracy

Even with tiny radii it can significantly lower accuracy [FFF15]:



With enough data is a common refrain.

Table of Contents

Case Study: First Major ML Attack

Probably the first deployment of ML in Security was in spam filtering in about 2002, using Naive Bayes [Gra02].

Case Study: First Major ML Attack

Probably the first deployment of ML in Security was in spam filtering in about 2002, using Naive Bayes [Gra02].

By 2004 various attacks had been seen in the wild [GC04]:

1. Obfuscating text
2. Small emails that just hold links
3. Hiding the email in an Non-Deliverable Return
4. Packing the email with "good words"

Case Study: First Major ML Attack

Probably the first deployment of ML in Security was in spam filtering in about 2002, using Naive Bayes [Gra02].

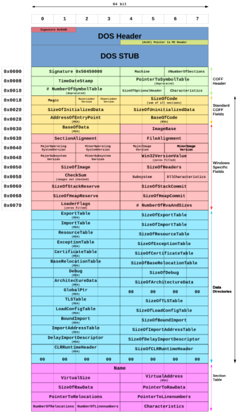
By 2004 various attacks had been seen in the wild [GC04]:

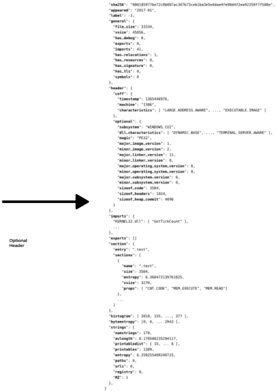
1. Obfuscating text
2. Small emails that just hold links
3. Hiding the email in an Non-Deliverable Return
4. Packing the email with "good words"

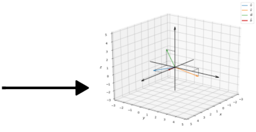
These are not "adversarial examples" as we've been discussing. **These are people sitting down with an understanding of how your system works and figuring out a bypass.**

Case Study: A Recent ML Attack

Elastic's EMBER's Featurization System [AR18]:





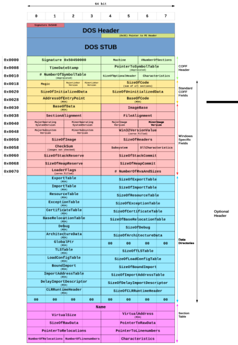


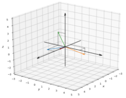
But 2381 dimensions

Case Study: A Recent ML Attack

In 2019 Adi Ashkenazy and Shahar Zini reverse engineered the Cylance malware detector and found a bypass [AA19].

Part of Cylance's Malware Detection System:





But 7000 dimensions and includes string data

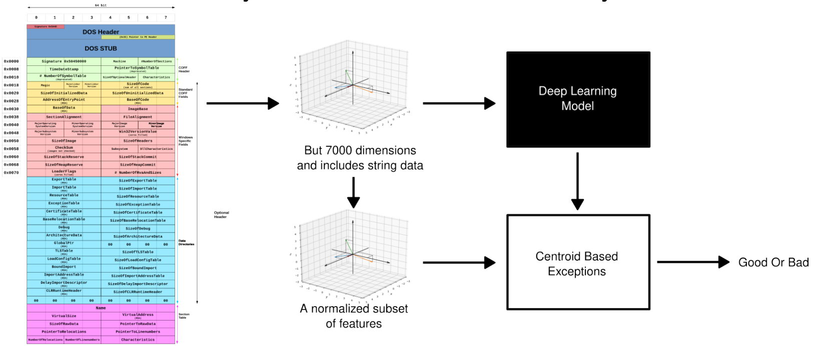
Deep Learning Model

Good Or Bad

Case Study: A Recent ML Attack

In 2019 Adi Ashkenazy and Shahar Zini reverse engineered the Cylance malware detector and found a bypass [AA19].

Part of Cylance's Malware Detection System:



Case Study: A Recent ML Attack

In 2019 Adi Ashkenazy and Shahar Zini reverse engineered the Cylance malware detector and found a bypass [AA19].

Part of Cylance's Malware Detection System:

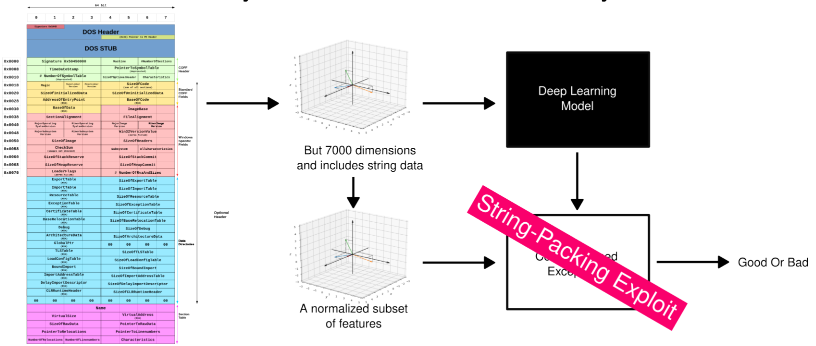


Table of Contents

Real Security: Fundamentals

It's still software, do the basics!

1. Validate your inputs.
2. Double check your pickles.
3. Check your deployments for CVEs.
4. Harden everything.
5. Secure your S3 buckets.

Real Security: Fundamentals

From: The Institute for Ethical AI & Machine Learning [fEAML22]

1. Unrestricted Model Endpoints
2. Access to Model Artifacts
3. Artifact Exploit Injection
4. Insecure ML Systems/Pipeline Design
5. Data & ML Infrastructure Misconfigurations
6. Supply Chain Vulnerabilities in ML Code
7. IAM & RBAC Failures for ML Services
8. ML Infra / ETL / CI / CD Integrity Failures
9. Observability, Reproducibility & Lineage
10. ML-Server Side Request Forgery

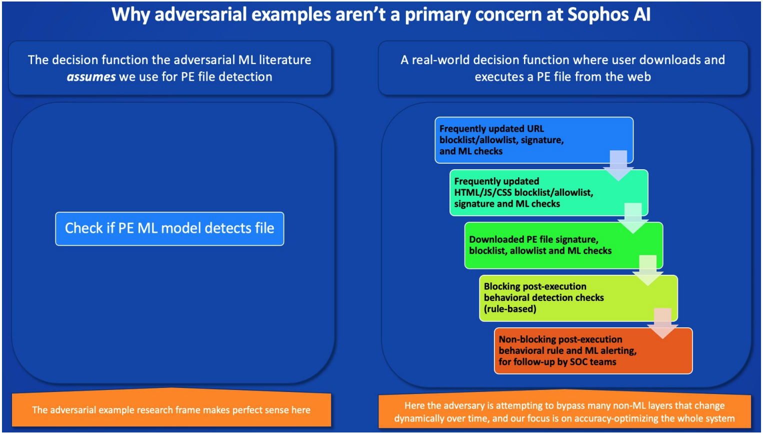
Real Security: Dataset

Your Dataset is more valuable than your model.

Unless you're at OpenAI training GPT-5 you probably spend far more on your dataset than training your model.

You can retrain your model, change your features, and improve it, **if you have your data!**

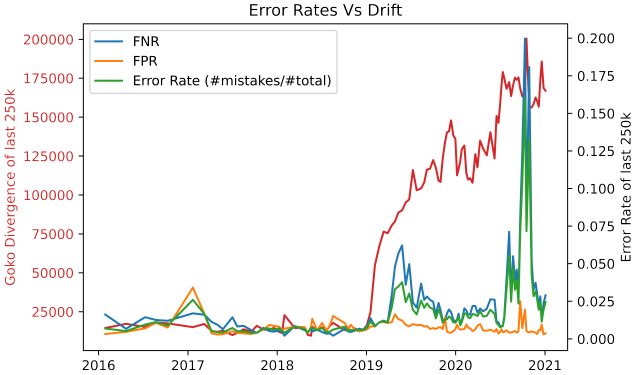
Don't leave ML unattended. Treat it like a toddler.



From Joshua Saxe's twitter, @joshua_sax

Real Security: Understand Your Distribution

Don't deploy and forget. Monitor your model! Redeploy when stale!



This is harder than you think.

Real Security: Understand Your Attackers

1. **What's easy for your attacker to change?**
2. **What's hard to change?**
3. **What's fundamental to their objective?**

Spammers on social networks need a lot of accounts posting a lot of content. This behavioral data is far more reliable than content data that's easy to change.

Adding imports increases the on disk size of a binary, and malware authors like tiny binaries.

Advanced Real Security: Be a Moving Target

Retraining only goes so far. A bypass for a model at this point, may also be a bypass for next week's model.

Constantly make new software features, and ML features.

When you have enough good ML features swap them out and change which ones you use each week.

This policy puts an expiration date on attacks. [WP19]

References I

- S. Zini A. Ashkenazy, *Cylance, i kill you!*, https://skylightcyber.com/2019/07/18/cylance-i-kill-you/, 2019.
- Hyrum S. Anderson and Phil Roth, *Ember: An open dataset for training static pe malware machine learning models*, 2018.
- Sven Cattell, *Online dataset drift in* $o(\log n)$.
- The Institute for Ethical AI & Machine Learning, *The mlsecops top 10*, https://ethical.institute/security.html, 2022.
- Alhussein Fawzi, Omar Fawzi, and Pascal Frossard, *Analysis of classifiers' robustness to adversarial perturbations*, 2015.
- J Graham-Cumming, *How to beat an adaptive spam filter.*, MIT Spam Conference, 2004.

References II

P. Graham, *A plan for spam*, http://www.paulgraham.com/spam.html, 2002.

Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy, *Explaining and harnessing adversarial examples*, 2014.

Boris Hanin and David Rolnick, *Complexity of linear regions in deep networks*, 2019.

Zico Kolter and Aleksander Madry, *Adversarial training, solving the outer minimization*, https://adversarial-ml-tutorial.org/adversarial_training/, 2022.

Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu, *Towards deep learning models resistant to adversarial attacks*, 2017.

References III

Ramchandran Muthukumar and Jeremias Sulam, *Adversarial robustness of sparse local lipschitz predictors*, 2022.

Taschee, *Lipschitz diagram*, https://commons.wikimedia.org/w/index.php?curid=59500064, 2022.

N. Landers W. Pearce, *Cve-2019-20634 detail*, https://nvd.nist.gov/vuln/detail/CVE-2019-20634, 2019.