



دانشگاه صنعتی امیرکبیر  
(پلی تکنیک تهران)  
دانشکده مهندسی کامپیوتر

درس روش پژوهش  
گزارش نوشتاری

بررسی و مهار مثال‌های خصمانه در یادگیری ماشین

نگارش

بردیا اردکانیان

استاد راهنما

دکتر رضا صفابخش

فروردین ۱۴۰۱

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

# سپاس‌گزاری

زندگی صدر چشم خرد تمام دنیا زیباست چون گل که به دست دلبری خوش سیماست  
پس نقش بشر در این میان دانی چیست؟ یک ذره که چون الکترون ناپیدا است

از استاد گرامی جناب آقای دکتر رضا صفابخش که در پیشبرد این پروژه به عنوان استاد پروژه، کمکهای  
فراوانی به این جانب داشتند، کمال تشکر را دارم.  
همچنین از جناب آقای محمد تولکی که در تهیه این گزارش، به من کمک کردند کمال سپاس را دارم.

بر دیا اردکانیان  
فروردین ۱۴۰۱

## چکیده

چندین مدل یادگیری ماشین، از جمله شبکه‌های عصبی، به طور مداوم نمونه‌های متخاصم را به اشتباه طبقه‌بندی می‌کنند - ورودی‌های ساخته‌شده مخصوصاً به گونه‌ای طراحی شده‌اند که برای انسان‌ها «معمولی» به نظر برسند، اما باعث طبقه‌بندی اشتباه در مدل یادگیری ماشین می‌شوند. اغلب، یک شکل از «نویز» طراحی شده ویژه برای برانگیختن طبقه‌بندی اشتباه استفاده می‌شود - نتایج را در خروجی مدل با یک پاسخ نادرست با اطمینان بالا نشان می‌دهد.

آسیب‌پذیری در برابر نمونه‌های متخاصم به یکی از خطرات اصلی برای استفاده از شبکه عصبی عمیق در محیط‌های حیاتی ایمنی تبدیل شده است. بنابراین، حملات و دفاع از نمونه‌های متخاصم توجه زیادی را به خود جلب کرده. ما یافته‌های اخیر در مورد نمونه‌های متخاصم برای شبکه‌های عصبی عمیق را مرور می‌کنیم و بهترین روش را برای تولید نمونه‌های متخاصم خلاصه و معرفی می‌کنیم.

علت این مثال‌های متخاصم یک راز بود، و فرضیه‌ها نشان می‌دهد که دلیل آن غیرخطی بودن شدید شبکه‌های عصبی عمیق است. ما در عوض استدلال می‌کنیم که علت اصلی آسیب‌پذیری شبکه‌های عصبی در برابر اغتشاش خصمانه، ماهیت خطی آنهاست. این توضیح با نتایج جدید بدست آمده نتیجه می‌شود در حالی که جالب‌ترین واقعیت در مورد آنها را ارائه می‌دهد. قابلیت تعمیم آنها در معماری‌ها و مجموعه‌های آموزشی مختلف. علاوه بر این، این دیدگاه یک روش ساده و سریع برای تولید نمونه‌های متخاصم را ارائه می‌دهد. از این رویکرد برای ارائه نمونه‌هایی متخاصم برای آموزش خصمانه استفاده می‌شود. با استفاده از این رویکرد برای ارائه نمونه‌هایی برای آموزش خصمانه، خطای مجموعه تست یک شبکه حداکثری را در پایگاه داده موسسه ملی استاندارد و فناوری کاهش می‌دهیم.

## واژه‌های کلیدی:

یادگیری ماشین، مثال‌های خصمانه، حملات متخاصم، آموزش متخاصم، خطی بودن شبکه‌های عصبی

# فهرست مطالب

صفحه

عنوان

۱	مقدمه	۱
۲	۱-۱ مفاهیم اولیه	۲
۲	۲-۱ اهمیت مثال‌های خصمانه	۲
۳	۳-۱ فرضیه‌ها و گمانه‌زنی‌ها	۳
۴	۴-۱ کارهای مرتبط	۴
۵	۲ توضیح خطی مثال‌های خصمانه	۵
۶	۱-۲ توضیح وجود مثال‌های خصمانه برای مدل‌های خطی	۶
۶	۱-۱-۲ توضیح وجود مثال‌های خصمانه	۶
۶	۲-۱-۲ پنهان‌نگاری تصادفی	۶
۷	۳-۱-۲ خلاصه و نتیجه‌گیری	۷
۷	۲-۲ اختلال خطی در مدل‌های غیرخطی	۷
۷	۱-۲-۲ روش نشانه‌گرادیان سریع	۷
۸	۲-۲-۲ اعمال روش نشانه‌گرادیان سریع بر مدل‌های مختلف	۸
۹	۳-۲-۲ خلاصه و نتیجه‌گیری	۹
۱۰	۳ آموزش خصمانه شبکه‌های عمیق	۱۰
۱۱	۱-۳ آموزش ترکیبی از نمونه‌های متخاصم و پاک	۱۱
۱۲	۲-۳ نتیجه آموزش خصمانه	۱۲
۱۲	۳-۳ آشفته‌سازی ورودی یا لایه‌های پنهان	۱۲
۱۳	۴-۳ خلاصه و نتیجه‌گیری	۱۳
۱۴	۴ چرا مثال‌های متخاصم عمومیت دارند؟	۱۴
۱۵	۱-۴ طبقه‌بندی اشتباه نمونه متخاصم یکسان توسط مدل‌های مختلف	۱۵
۱۶	۲-۴ اختصاص طبقه‌بندی یکسان به نمونه‌های متخاصم توسط مدل‌های مختلف	۱۶
۱۷	۳-۴ خلاصه و نتیجه‌گیری	۱۷
۱۸	۵ خلاصه و نتیجه‌گیری	۱۸
۲۱	منابع و مراجع	۲۱

شکل	فهرست اشکال	صفحه
۱-۲	نمایش مثال خصمانه	۸
۱-۳	تجسم وزن شبکه‌های حداکثر	۱۳
۱-۴	ردیابی مقادیر مختلف $\epsilon$	۱۵

صفحه

فهرست جداول

جدول

## فهرست نمادها

نماد	مفهوم
$x$	داده های ورودی اصلی (تمیز، اصلاح نشده)
$\hat{x}$	مثال خصمانه (داده های ورودی اصلاح شده)
$\eta$	تفاوت بین داده های ورودی اصلی و اصلاح شده: $\eta = \hat{x} - x$
$w$	بردار وزن
$\theta$	پارامترهای یک مدل
$\nabla$	گرادیان
$\ \cdot\ _p$	هنجار <sup>۱</sup>



# فصل اول

## مقدمه

در سال‌های گذشته، هوش مصنوعی پیشرفت‌های بسیاری به خود دیده است. در حال حاضر، یادگیری ماشین<sup>۱</sup> برای حل بسیاری از مسائل چالش برانگیز مانند ماشین‌های خودران، ترجمه خودکار زبان، تشخیص کلاهبرداری به کار گرفته می‌شود. گستردگی یادگیری ماشین به قدری ادامه پیدا کرده است که شماری از بسترهای شناخته شده همچون آمازون<sup>۲</sup> و گوگل<sup>۳</sup>، یادگیری ماشین را به عنوان یک سرویس به توسعه دهندگان ارائه می‌کنند. یادگیری ماشین متخاصم<sup>۴</sup> یک تکنیک یادگیری ماشین است که تلاش می‌کند با بهره‌گیری از اطلاعات قابل دستیابی مدل و استفاده از آن برای ایجاد حملات مخرب، از مدل‌ها سوء استفاده کند.

## ۱-۱ مفاهیم اولیه

بیشتر تکنیک‌های یادگیری ماشین برای کار بر روی مجموعه‌ای از مشکلات خاص طراحی شده‌اند که در آن داده‌های آموزشی و آزمایشی از توزیع آماری یکسان تولید می‌شوند. وقتی این مدل‌ها در دنیای واقعی اعمال می‌شوند، مهاجمان ممکن است داده‌هایی را ارائه دهند که این فرض آماری را نقض می‌کند. این داده‌ها ممکن است برای سوء استفاده از آسیب‌پذیری‌های خاص و به خطر انداختن نتایج تنظیم شوند. چندین مدل یادگیری ماشین، از جمله شبکه‌های عصبی پیشرفته<sup>۵</sup>، در برابر نمونه‌های متخاصم آسیب پذیر هستند. یعنی، این مدل‌های یادگیری ماشینی، نمونه‌هایی را که فقط کمی از نمونه‌های درست طبقه‌بندی شده توزیع داده‌ها<sup>۶</sup> متفاوت هستند، به اشتباه دسته‌بندی می‌کنند. در بسیاری از موارد، مدل‌های متنوع با معماری‌های مختلف آموزش دیده شده بر روی زیرمجموعه‌های مختلف داده‌های آموزشی، مثال‌های خصمانه<sup>۷</sup> یکسان را به اشتباه طبقه‌بندی می‌کنند. سگدی و همکاران [۱] برای مشکل طبقه‌بندی تصویر آشفتگی‌های کوچکی روی تصاویر ایجاد کرد و شبکه‌های عصبی عمیق را با درصد اطمینان زیاد فریب داد. این نمونه‌های طبقه‌بندی شده به عنوان نمونه‌های متخاصم نامگذاری شدند.

## ۲-۱ اهمیت مثال‌های خصمانه

برنامه‌های کاربردی گسترده‌ای مبتنی بر شبکه‌های عمیق وجود دارند تا در دنیای فیزیکی، به ویژه در محیط‌های بحرانی ایمنی به کار گرفته شوند. در این میان، مطالعات اخیر نشان می‌دهد که می‌توان

<sup>1</sup> Machine learning

<sup>2</sup> Amazon

<sup>3</sup> Google

<sup>4</sup> Adversarial machine learning

<sup>5</sup> State-of-art neural networks

<sup>6</sup> Data distribution

<sup>7</sup> Adversarial examples

نمونه‌های متخاصم را در دنیای واقعی نیز به کار برد. برای مثال، یک مهاجم می‌تواند نمونه‌های متخاصم فیزیکی بسازد و وسایل نقلیه خودران را با دستکاری علامت توقف در یک سیستم تشخیص علائم راهنمایی و رانندگی [۲]، [۳] یا حذف بخش‌بندی عابران پیاده در یک سیستم تشخیص شی [۴] گیج کند. مهاجمان می‌توانند دستورات متخاصم را علیه مدل‌های تشخیص خودکار گفتار و سیستم‌های قابل کنترل صوتی [۵]، [۶]، مانند سیری اپل<sup>۸</sup>، الکسا آمازون<sup>۹</sup>، کورتانا مایکروسافت<sup>۱۰</sup> تولید کنند. یادگیری عمیق به طور گسترده به عنوان یک تکنیک «جعبه سیاه» در نظر گرفته می‌شود - همه ما می‌دانیم که عملکرد خوبی دارد اما دانش محدودی بابت علت آن داریم [۷]، [۸]. مطالعات زیادی برای توضیح و تفسیر شبکه‌های عصبی عمیق [۹، ۱۰، ۱۱، ۱۲] پیشنهاد شده است. از بازرسی مثال‌های متخاصم، ممکن است بینش‌هایی درباره سطوح درونی معنایی شبکه‌های عصبی [۱۳] به دست آوریم و مرزهای تصمیم‌گیری مشکل‌ساز را پیدا کنیم، که به نوبه خود به افزایش استحکام و عملکرد شبکه‌های عصبی [۱۴] و بهبود تفسیرپذیری [۱۵] کمک می‌کند. در این مقاله، رویکردهای تولید مثال‌های متخاصم و کاربردهای مثال‌های متخاصم را بررسی و خلاصه می‌کنیم.

### ۳-۱ فرضیه‌ها و گمانه‌زنی‌ها

علت این مثال‌های متخاصم یک راز بود، و فرضیه‌ها نشان می‌دهد که دلیل آن غیرخطی بودن شدید شبکه‌های عصبی عمیق است، شاید با میانگین‌گیری ناکافی مدل و منظم‌سازی ناکافی مسئله یادگیری نظارت شده<sup>۱۱</sup> همراه باشد. ما نشان می‌دهیم که این فرضیه‌های گمانه‌زنی غیرضروری هستند. رفتار خطی در فضاها با ابعاد بالا برای ایجاد نمونه‌های متخاصم کافی است. این دیدگاه ما را قادر می‌سازد تا روشی سریع برای تولید نمونه‌های متخاصم طراحی کنیم که آموزش خصمانه را عملی می‌کند. ما نشان می‌دهیم که آموزش خصمانه می‌تواند یک مزیت منظم‌سازی اضافی فراتر از آنچه با استفاده از حذف تصادفی<sup>۱۲</sup> ارائه می‌شود [۱۶] به تنهایی فراهم کند. استراتژی‌های منظم‌سازی عمومی مانند حذف تصادفی، پیش‌آموزش<sup>۱۳</sup>، و میانگین‌گیری مدل، کاهش قابل توجهی در آسیب‌پذیری مدل در برابر نمونه‌های متخاصم ایجاد نمی‌کند، اما تغییر به خانواده‌های مدل غیرخطی مانند شبکه‌های اربی.اف<sup>۱۴</sup> می‌تواند این کار را انجام دهد.

توضیح ما یک تنش اساسی را بین طراحی مدل‌هایی که به دلیل خطی بودن، آموزش آسانی دارند و مدل‌هایی که از اثرات غیرخطی برای مقاومت در برابر اغتشاشات متخاصم استفاده می‌کنند، نشان می‌دهد. در درازمدت، ممکن است با طراحی روش‌های بهینه‌سازی قوی‌تر مدل‌های غیرخطی بیشتری

<sup>۸</sup>Apple siri

<sup>۹</sup>Amazon alexa

<sup>۱۰</sup>Microsoft cornata

<sup>۱۱</sup>Supervised learning

<sup>۱۲</sup>Dropout

<sup>۱۳</sup>Pretraining

<sup>۱۴</sup>Radial basis function network

را با موفقیت آموزش داد.

## ۴-۱ کارهای مرتبط

سگدی و همکاران [۱] انواع مختلفی از خواص جالب شبکه‌های عصبی و مدل‌های مرتبط را نشان دادند. موارد مرتبط با این مقاله عبارتند از:

- با کمک الگوریتم جعبه محدود شده<sup>۱۵</sup> می‌تواند به طور قابل اعتماد نمونه‌های متخاصم را پیدا کند.
- در برخی از مجموعه داده‌ها، مانند شبکه تصویری<sup>۱۶</sup> [۱۷]، نمونه‌های متخاصم آنقدر به نمونه‌های اصلی نزدیک بودند که تفاوت‌ها برای چشم انسان غیرقابل تشخیص بود.
- همان مثال متخاصم اغلب توسط طبقه‌بندی‌کننده‌ها با معماری‌های مختلف یا آموزش دیده بر روی زیر مجموعه‌های مختلف داده‌های آموزشی به اشتباه طبقه‌بندی می‌شود.
- مدل‌های رگرسیون سافت مکس کم عمق<sup>۱۷</sup> نیز در برابر نمونه‌های متخاصم آسیب‌پذیر هستند.
- آموزش نمونه‌های متخاصم می‌تواند مدل را منظم کند.

این نتایج نشان می‌دهد که طبقه‌بندی‌کننده‌های مبتنی بر تکنیک‌های یادگیری ماشین مدرن، حتی آن‌هایی که عملکرد عالی را در مجموعه آزمایشی به دست می‌آورند، مفاهیم اساسی واقعی را که برچسب خروجی صحیح را تعیین می‌کند، یاد نمی‌گیرند. این امر به بسیار ناامیدکننده است. این نتایج اغلب به عنوان یک نقص در شبکه‌های عمیق تفسیر می‌شوند، حتی اگر طبقه‌بندی‌کننده‌های خطی همین مشکل را داشته باشند. ما آگاهی از این نقص را فرصتی برای رفع آن می‌دانیم. در واقع، گو و ریگازیو [۱۸] و چالوپکا و همکاران [۱۹] قبلاً اولین گام‌ها را برای طراحی مدل‌هایی آغاز کرده‌اند که در برابر اغتشاش خصمانه مقاومت می‌کنند، اگرچه هیچ مدلی هنوز با موفقیت این کار را انجام نداده است و در عین حال دقت پیشرفته‌ای را در ورودی‌های تمیز - ورودی که دچار اشتباه و نقض نباشد - حفظ کرده است.

<sup>15</sup>Box-constrained L-BFGS

<sup>16</sup>ImageNet

<sup>17</sup>Shallow softmax regression

## فصل دوم

### توضیح خطی مثال‌های خصمانه

همانطور که گفته شد فرضیه‌ها نشان می‌دادند که دلیل مثال‌های متخاصم غیر خطی بودن شدید شبکه‌های عصبی عمیق است. ولی در این فصل استلال می‌کنیم علت اصلی آسیب‌پذیری شبکه‌های عصبی در برابر اغتشاشات خصمانه، ماهیت خطی آنهاست.

## ۱-۲ توضیح وجود مثال‌های خصمانه برای مدل‌های خطی

### ۱-۱-۲ توضیح وجود مثال‌های خصمانه

ما با توضیح وجود مثال‌های خصمانه برای مدل‌های خطی شروع می‌کنیم. در بسیاری از مشکلات، دقت یک ویژگی ورودی فردی محدود است. به عنوان مثال، تصاویر دیجیتالی اغلب تنها از هشت بیت در هر پیکسل استفاده می‌کنند، بنابراین تمام اطلاعات زیر  $1/255$  محدوده دینامیکی را حذف می‌کنند. از آنجا که دقت ویژگی‌ها محدود است، منطقی نیست که طبقه‌بندی‌کننده به ورودی  $x$  به طور متفاوتی نسبت به ورودی مخالف  $\hat{x} = x + \eta$  پاسخ دهد اگر هر عنصر اغتشاش  $\eta$  کوچکتر از دقت ویژگی‌ها باشد. به طور کلی، برای کلاس‌های خوب طبقه‌بندی شده، انتظار داریم که طبقه‌بندی‌کننده همان کلاس را به  $x$  و  $\hat{x}$  اختصاص دهد تا زمانی که  $\|\eta\|_\infty < \epsilon$ ، به اندازه‌ای کوچک است که توسط حسگر یا دستگاه ذخیره‌سازی داده مربوطه دور ریخته شود.

حاصل ضرب نقطه‌ای بین بردار وزن  $w^1$  و مثال متخاصم  $\hat{x}$  را در نظر بگیرید:

$$w^T \hat{x} = w^T x + w^T \eta \quad (1-2)$$

### ۲-۱-۲ پنهان‌نگاری تصادفی

اغتشاش خصمانه باعث می‌شود که فعال‌سازی با  $w^T \eta$  رشد کند. ما می‌توانیم این افزایش را با توجه به محدودیت حداکثر هنجار  $^2$  در  $\eta$  با اختصاص  $\eta = \text{sign}(w)$  به حداکثر برسانیم. اگر  $w$ ،  $n$  بعد داشته باشد و قدر متوسط یک عنصر از بردار وزن  $m$  باشد، فعال‌سازی به اندازه  $mn \in$  رشد خواهد کرد. از آنجایی که  $\|\eta\|_\infty$  با ابعاد مسئله رشد نمی‌کند، اما تغییر در فعال‌سازی ناشی از اغتشاش توسط  $\eta$  می‌تواند به صورت خطی با  $n$  رشد کند، پس برای مسائل ابعادی بالا، می‌توانیم تغییرات بی‌نهایت کوچک زیادی در ورودی ایجاد کنیم که جمع شوند و به یک تغییر بزرگ در خروجی منتهی شوند. ما می‌توانیم این را نوعی «دخت‌نگاری تصادفی» و یا «پنهان‌نگاری تصادفی» در نظر بگیریم، که در آن یک مدل خطی مجبور است به طور انحصاری به سیگنالی توجه کند که بیشترین همسویی را با وزن‌های آن دارد، حتی

<sup>1</sup>Weight vector

<sup>2</sup>Max norm

اگر چندین سیگنال وجود داشته باشد و سیگنال‌های دیگر دامنه بسیار بیشتری داشته باشند.

## ۳-۱-۲ خلاصه و نتیجه‌گیری

این توضیح نشان می‌دهد که یک مدل خطی ساده در صورتی می‌تواند نمونه‌های متخاصم داشته باشد که ورودی آن ابعاد کافی داشته باشد. توضیحات قبلی برای مثال‌های متخاصم از ویژگی‌های فرضی شبکه‌های عصبی، مانند ماهیت بسیار غیرخطی آن‌ها استفاده می‌کرد. فرضیه ما بر اساس خطی بودن ساده‌تر است، و همچنین می‌تواند توضیح دهد که چرا رگرسیون سافت مکس در برابر نمونه‌های متخاصم آسیب‌پذیر است.

## ۲-۲ اختلال خطی در مدل‌های غیرخطی

نمای خطی نمونه‌های متخاصم راه سریعی را برای تولید آن‌ها پیشنهاد می‌کند. ما فرض می‌کنیم که شبکه‌های عصبی برای مقاومت در برابر اغتشاش خصمانه خطی بسیار خطی هستند. حافظه کوتاه مدت <sup>۳</sup> [۲۰]، واحد خطی اصلاح شده <sup>۴</sup> [۲۱] و شبکه‌های حداکثری <sup>۵</sup> [۲۲] همگی عمداً به گونه‌ای طراحی شده‌اند که به روش‌های بسیار خطی رفتار کنند، به طوری که بهینه‌سازی آن‌ها آسان‌تر باشد.

مدل‌های غیرخطی بیشتری مانند شبکه‌های سیگموئید <sup>۶</sup> به همین دلیل به دقت تنظیم می‌شوند تا بیشتر وقت خود را در رژیم غیراشباع و خطی‌تر بگذرانند. این رفتار خطی نشان می‌دهد که اغتشاشات تحلیلی ارزان یک مدل خطی نیز باید به شبکه‌های عصبی آسیب برساند.

## ۱-۲-۲ روش نشانه گرادیان سریع

فرض کنید  $\theta$  پارامترهای یک مدل،  $x$  ورودی مدل،  $y$  اهداف مرتبط با  $x$  (برای وظایف یادگیری ماشینی که دارای اهداف هستند) و  $J(\theta, x, y)$  هزینه استفاده شده برای آموزش شبکه عصبی باشد. ما می‌توانیم تابع هزینه را حول مقدار فعلی  $\theta$  خطی کنیم و یک اختلال محدود حداکثر هنجار بهینه به دست آوریم.

$$\eta = \epsilon \text{sign}(\nabla_x J(\theta, x, y)) \quad (2-2)$$




<sup>3</sup>LSTMs

<sup>4</sup>ReLU

<sup>5</sup>Maxout networks

<sup>6</sup>Sigmoid networks

ما از این به عنوان «روش نشانه گرادیان سریع»<sup>۷</sup> برای تولید نمونه‌های متخاصم یاد می‌کنیم. توجه داشته باشید که گرادیان مورد نیاز را می‌توان به طور موثر با استفاده از پس انتشار<sup>۸</sup> محاسبه کرد.

	$+ .007 \times$		$=$	
$x$		$\text{sign}(\nabla_x J(\theta, x, y))$		$x + \epsilon \text{sign}(\nabla_x J(\theta, x, y))$
“panda”		“nematode”		“gibbon”
57.7% confidence		8.2% confidence		99.3 % confidence

شکل ۲-۱: نمایش مثال خصمانه

شکل ۲-۱ نمایشی از تولید نمونه سریع خصمانه اعمال شده بر گوگل نت<sup>۹</sup> [۲۳] در شبکه تصویری می‌باشد. با افزودن یک بردار به‌طور نامحسوس کوچک که عناصر آن برابر با علامت عناصر گرادیان تابع هزینه نسبت به ورودی است، می‌توانیم طبقه‌بندی تصویر توسط گوگل نت را تغییر دهیم. در اینجا 0.007 ما مربوط به بزرگی کوچکترین بیت از یک تصویر هشت بیتی است که پس از تبدیل گوگل نت به اعداد واقعی رمزگذاری شده است.

## ۲-۲-۲ اعمال روش نشانه گرادیان سریع بر مدل‌های مختلف

ما متوجه شدیم که این روش به طور قابل اعتمادی باعث می‌شود طیف گسترده‌ای از مدل‌ها ورودی خود را به اشتباه طبقه‌بندی کنند. برای نمایش در شبکه تصویری به شکل ۴-۱ مراجعه کنید. متوجه شدیم که با استفاده از  $\epsilon = 0.25$ ، باعث می‌شویم که یک طبقه‌بندی‌کننده بیشینه هموار کم عمق<sup>۱۰</sup> دارای نرخ خطای 99.9 درصد با اطمینان متوسط 79.3 درصد در مجموعه آزمایشی *MNIST*<sup>۱۱</sup> باشد. در همین تنظیمات، یک شبکه حداکثری 89.4 درصد از نمونه‌های متخاصم ما را با اطمینان متوسط 97.6 درصد به اشتباه طبقه‌بندی می‌کند. به طور مشابه، با استفاده از  $\epsilon = 0.1$ ، نرخ خطای 87.15 درصد

<sup>7</sup>Fast gradient sign method

<sup>8</sup>Backpropagation

<sup>9</sup>GoogLeNet

<sup>10</sup>Shallow softmax classifier

<sup>11</sup>Modified National Institute of Standards and Technology dataset

<sup>۱۲</sup> این با استفاده از مقادیر پیکسل *MNIST* در بازه صفر و یک است. داده‌های *MNIST* حاوی مقادیری غیر از صفر یا

یک هستند، اما تصاویر اساساً باینری هستند. هر پیکسل تقریباً «جوهر» یا «بدون جوهر» را رمزگذاری می‌کند. این انتظار را توجیه می‌کند که طبقه‌بندی‌کننده بتواند اختلالات را در محدوده عرض ۵.۰ مدیریت کند و در واقع ناظران انسانی می‌توانند چنین تصاویری را بدون مشکل بخوانند.



و احتمال متوسط 96.6 درصد را که به برچسب‌های نادرست اختصاص داده شده است، هنگام استفاده از یک شبکه حداکثری کانولوشن<sup>۱۳</sup> در یک نسخه از پیش پردازش شده  $CIFAR - 10$ <sup>۱۴</sup> بدست می‌آوریم. [۲۴]

روش‌های ساده دیگری برای تولید نمونه‌های متخاصم امکان پذیر است. به عنوان مثال، دریافتیم که چرخش  $x$  با یک زاویه کوچک در جهت گرادیان به طور قابل اعتماد نمونه‌های متخاصم تولید می‌کند.

## ۳-۲-۲ خلاصه و نتیجه‌گیری

این واقعیت که این الگوریتم‌های ساده و ارزان می‌توانند نمونه‌های طبقه‌بندی شده اشتباه تولید کنند، نشان دهنده درست بودن تفسیر ما از نمونه‌های متخاصم در نتیجه خطی بودن می‌باشد. این الگوریتم‌ها همچنین به عنوان راهی برای افزایش سرعت آموزش خصمانه یا حتی تجزیه و تحلیل شبکه‌های آموزش دیده مفید هستند.

---

<sup>۱۳</sup>Convolutional maxout network

<sup>۱۴</sup> مجموعه داده  $CIFAR - 10$  مجموعه‌ای از تصاویر است که معمولاً برای آموزش الگوریتم‌های یادگیری ماشین و بینایی کامپیوتر استفاده می‌شود.

## فصل سوم

# آموزش خصمانه شبکه‌های عمیق

انتقاد از شبکه‌های عمیق به‌عنوان آسیب‌پذیر در برابر نمونه‌های متخاصم تا حدودی نادرست است، زیرا برخلاف مدل‌های خطی کم عمق، شبکه‌های عمیق حداقل می‌توانند عملکردهایی را نشان دهند که در برابر اغتشاش خصمانه مقاومت می‌کنند. قضیه تقریب جهانی [۲۵] تضمین می‌کند که یک شبکه عصبی با حداقل یک لایه پنهان می‌تواند هر تابعی را با درجه دقت دلخواه نشان دهد تا زمانی که لایه پنهان آن واحدهای کافی داشته باشد. مدل‌های خطی کم عمق نمی‌توانند در نزدیکی نقاط آموزشی ثابت شوند و در عین حال خروجی‌های متفاوتی را به نقاط آموزشی مختلف اختصاص دهند.

البته، قضیه تقریب جهانی<sup>۱</sup> چیزی در مورد اینکه آیا یک الگوریتم آموزشی قادر به کشف تابعی با تمام ویژگی‌های مورد نظر خواهد بود، نمی‌گوید. بدیهی است که آموزش استاندارد تحت نظارت مشخص نمی‌کند که عملکرد انتخاب شده در برابر نمونه‌های متخاصم مقاوم باشد. این باید به نحوی در روند آموزش پیاده‌سازی شود.

در این فصل به نتیجه آموزش ترکیبی از نمونه‌های متخاصم و پاک بر روی شبکه‌های عمیق می‌پردازیم و بررسی می‌کنیم آموزش خصمانه در چه زمانی مفید است.

### ۳-۱ آموزش ترکیبی از نمونه‌های متخاصم و پاک

سگدی و همکاران [۱] نشان داد که با آموزش ترکیبی از نمونه‌های متخاصم و پاک یک شبکه عصبی می‌تواند تا حدودی منظم شود. آموزش در مورد نمونه‌های متخاصم تا حدودی با سایر طرح‌های افزایش داده<sup>۲</sup> متفاوت است. معمولاً، یکی داده‌ها را با تبدیل‌هایی مانند ترجمه‌هایی که انتظار می‌رود در مجموعه آزمایشی واقعاً رخ دهد، افزایش می‌دهد. این شکل از تقویت داده‌ها در عوض از ورودی‌هایی استفاده می‌کند که بعید است به طور طبیعی اتفاق بیفتند، اما نقص‌هایی را در تابع تصمیم<sup>۳</sup> مدل را آشکار می‌کند. ما دریافتیم که تمرین با تابع هدف متخاصم بر اساس روش نشانه گرادیان سریع، تنظیم‌کننده مؤثری بود:

$$\hat{J}(\theta, x, y) = \alpha J(\theta, x, y) + (1 - \alpha) J(\theta, x - \epsilon \text{sign}(\nabla_x J(\theta, x, y))) \quad (۱-۳)$$

در همه آزمایش‌ها، از  $\alpha = 0.5$  استفاده شده است. سایر مقادیر ممکن است بهتر عمل کنند. اما این فرآیند کمتر به اندازه کافی خوب عمل کرد که نیازی به امتحان کردن بقیه مقادیر وجود نداشت. این رویکرد به این معنی است که به طور مداوم نمونه‌های متخاصم به روز می‌شوند تا آنها را در مقابل نسخه فعلی مدل مقاوم کنند.

<sup>۱</sup> Universal approximator theorem

<sup>۲</sup> Augmentation schem

<sup>۳</sup> Decision function

## ۲-۳ نتیجه آموزش خصمانه

با استفاده از این رویکرد برای آموزش یک شبکه حداکثر که منظم شده بود، نرخ خطا را از 0.94 درصد بدون آموزش خصمانه به 0.84 درصد با آموزش خصمانه کاهش یافت. این مدل همچنین در برابر نمونه‌های متخاصم تا حدودی مقاوم شد. به یاد بیاورید که بدون آموزش خصمانه، همین نوع مدل دارای نرخ خطای 89.4 درصد در نمونه‌های متخاصم بر اساس روش علامت گرادیان سریع بود. با آموزش خصمانه، میزان خطا به 17.9 درصد کاهش یافت. نمونه‌های متخاصم بین دو مدل قابل انتقال هستند، اما مدل آموزش دیده خصمانه استحکام بیشتری نشان می‌دهد. نمونه‌های متخاصم تولید شده از طریق مدل اصلی، نرخ خطای 19.6 درصد را در مدل آموزش دیده شده به دست می‌دهند، در حالی که نمونه‌های متخاصم تولید شده از طریق مدل جدید، نرخ خطای 40.9 درصد را در مدل اصلی دارند. زمانی که مدل آموزش دیده به صورت خصمانه یک مثال متخاصم را به اشتباه طبقه‌بندی می‌کند، متأسفانه پیش‌بینی‌های آن همچنان بسیار مطمئن هستند. میانگین اطمینان در یک نمونه طبقه‌بندی اشتباه 81.4 بود. همچنین می‌توان دریافت که وزن‌های مدل آموخته‌شده به‌طور قابل توجهی تغییر کرده است. (شکل ۱-۳ را ببینید).

روش آموزش خصمانه را می‌توان به عنوان به حداقل رساندن بدترین خطا در زمانی که داده‌ها توسط یک دشمن اشفته می‌شود مشاهده کرد. این می‌تواند به عنوان یادگیری انجام یک بازی خصمانه یا به حداقل رساندن یک حد بالایی در هزینه مورد انتظار نسبت به نمونه‌های نویز بالا با نویز  $U(-\epsilon, \epsilon)$  اضافه شده به ورودی‌ها تفسیر شود. آموزش خصمانه را می‌توان به عنوان شکلی از یادگیری فعال نیز در نظر گرفت که در آن مدل می‌تواند برچسب‌هایی را بر روی نقاط جدید درخواست کند. در این مورد برچسب انسانی با یک برچسب ابتکاری جایگزین می‌شود که برچسب‌ها را از نقاط نزدیک کپی می‌کند.

## ۳-۳ آشفته سازی ورودی یا لایه‌های پنهان

یک سوال طبیعی این است که آیا بهتر است ورودی را اشفته کنیم یا لایه‌های پنهان یا هر دو. در اینجا نتایج متناقض است. سگدی و همکاران [۱] گزارش داد که اغتشاشات متخاصم بهترین نظم‌دهی را زمانی که بر روی لایه‌های پنهان اعمال می‌شوند، ایجاد می‌کنند. این نتیجه در یک شبکه سیگموئیدی به دست آمد. در آزمایش‌های ما با روش نشان گرادیان سریع، متوجه می‌شویم که شبکه‌هایی با واحدهای پنهان که فعال‌سازی‌های آنها نامحدود است، به سادگی با بزرگ کردن فعال‌سازی واحد پنهان خود پاسخ می‌دهند، بنابراین معمولاً بهتر است فقط ورودی اصلی را مختل کنیم.

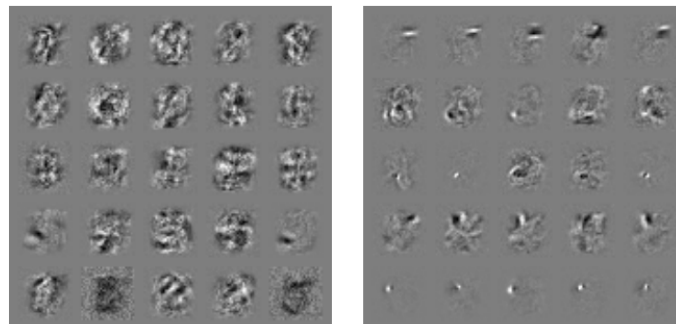
در مدل‌های اشباع مانند مدل زنگ<sup>۴</sup> متوجه شدیم که اغتشاش ورودی با اغتشاش لایه‌های پنهان<sup>۵</sup> کار می‌کند. آشفته‌گی‌های مبتنی بر چرخش لایه‌های پنهان<sup>۶</sup>، مشکل رشد فعال‌سازی‌های نامحدود را

<sup>4</sup>Rust<sup>5</sup>Hidden layers<sup>6</sup>Rotating the hidden layers

حل می‌کنند تا اغتشاشات افزایشی را در مقایسه با آن کوچک‌تر کنند. ما توانستیم شبکه‌های حداکثر را با اختلالات چرخشی لایه‌های پنهان آموزش دهیم. با این حال، این تقریباً به اندازه اغتشاش افزایشی لایه ورودی، یک اثر منظم‌کننده قوی نداشت.

### ۴-۳ خلاصه و نتیجه‌گیری

دیدگاه ما در مورد آموزش خصمانه این است که تنها زمانی مفید است که مدل توانایی یادگیری مقاومت در برابر نمونه‌های متخاصم را داشته باشد. این تنها زمانی به وضوح صادق است که یک قضیه تقریبی جهانی اعمال شود. از آنجا که آخرین لایه یک شبکه عصبی، لایه خطی-سیگموئید یا خطی-سافت مکس است.



شکل ۳-۱: تجسم وزن شبکه‌های حداکثر

شکل ۳-۱ تجسم وزن شبکه‌های حداکثر آموزش دیده در *MNIST* می‌باشد. هر ردیف فیلترهای یک واحد حداکثر را نشان می‌دهد. چپ) مدل آموزش دیده ساده لوحانه. راست) مدل با آموزش خصمانه

## فصل چهارم

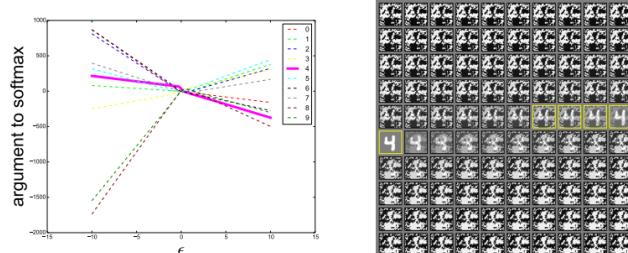
### چرا مثال‌های متخاصم عمومیت دارند؟

یکی از جنبه‌های جالب مثال‌های متخاصم این است که نمونه‌ای که برای یک مدل تولید شده اغلب توسط مدل‌های دیگر به اشتباه طبقه‌بندی می‌شوند، حتی زمانی که معماری‌های متفاوتی دارند یا بر روی مجموعه‌های آموزشی مجزا آموزش دیده‌اند. علاوه بر این، هنگامی که این مدل‌های مختلف یک مثال متخاصم را به اشتباه طبقه‌بندی می‌کنند، اغلب در مورد کلاس آن با یکدیگر توافق دارند. توضیحات مبتنی بر شدت غیرخطی بودن و برازش بیش از حد نمی‌توانند به راحتی این رفتار را توضیح دهند - چرا باید مدل‌های شدت غیرخطی متعدد با ظرفیت مازاد به طور مداوم نقاط خارج از توزیع را به همان روش تطبیق دهند؟ این رفتار به‌ویژه از دیدگاه این فرضیه تعجب برانگیز است که مثال‌های متخاصم فضا را مانند اعداد گویا در بین اعداد حقیقی پوشش می‌دهد، چرا که در این دیدگاه مثال‌های متخاصم رایج هستند اما فقط در مکان‌های بسیار دقیق رخ می‌دهند.

در این فصل به عمومیت مثال‌های متخاصم در مدل‌های متفاوت می‌پردازیم و نشان می‌دهیم چرا مدل‌های متفاوت مثال‌های متخاصم یکسان را به یک شکل طبقه‌بندی می‌کنند.

## ۴-۱ طبقه‌بندی اشتباه نمونه متخاصم یکسان توسط مدل‌های مختلف

در نمای خطی، نمونه‌های متخاصم در زیر فضاهای وسیع رخ می‌دهند. جهت  $\eta$  فقط باید حاصل ضرب نقطه‌ای مثبت با گرادیانت تابع هزینه را داشته باشد و  $\epsilon$  فقط باید به اندازه کافی بزرگ باشد. شکل ۴-۱ این پدیده را نشان می‌دهد. با ردیابی مقادیر مختلف  $\epsilon$  می‌بینیم که نمونه‌های متخاصم در مناطق به هم پیوسته زیرفضای یک بعدی که با روش نشان گرادیانت سریع تعریف شده‌اند، رخ می‌دهند، نه در پاکت‌های ظریف. این توضیح می‌دهد که چرا مثال‌های متخاصم فراوان هستند و چرا نمونه‌ای که توسط یک طبقه‌بندی‌کننده، به اشتباه طبقه‌بندی شده باشد، احتمال نسبتاً بالایی دارد که توسط طبقه‌بندی‌کننده دیگر نیز درست طبقه‌بندی نشده باشد.



شکل ۴-۱: ردیابی مقادیر مختلف  $\epsilon$

شکل ۴-۱: با ردیابی مقادیر مختلف  $\epsilon$ ، می‌توانیم ببینیم که نمونه‌های متخاصم تقریباً برای هر مقدار به اندازه کافی بزرگ  $\epsilon$  به‌طور قابل اعتمادی رخ می‌دهند، مشروط بر اینکه در جهت درست حرکت کنیم.

طبقه‌بندی‌های صحیح فقط در منی‌فولد<sup>۱</sup> نازکی که  $x$  در داده‌ها رخ می‌دهد، رخ می‌دهد. بیشتر  $\mathbb{R}^n$  از نمونه‌های متخاصم تشکیل شده. این طرح از یک شبکه حداکثر آموزش دیده ساده لوحانه ساخته شده است. سمت چپ) نموداری که آرگومان لایه سافت مکس را برای هر یک از ۱۰ کلاس  $MNIST$  نشان می‌دهد، همانطور که  $\epsilon$  را در یک مثال ورودی تغییر می‌دهیم. کلاس صحیح ۴ است. می‌بینیم که احتمالات لاگ غیرعادی شده برای هر کلاس به طور آشکار به صورت تکه ای خطی با  $\epsilon$  هستند و طبقه بندی‌های اشتباه در یک منطقه وسیع از مقادیر  $\epsilon$  پایدار هستند. علاوه بر این، پیش‌بینی‌ها بسیار افراطی می‌شوند، زیرا به اندازه کافی  $\epsilon$  را افزایش می‌دهیم تا وارد ورودی زباله شویم<sup>۲</sup>. راست) ورودی‌هایی که برای تولید منحنی استفاده می‌شوند (بالا سمت چپ  $\epsilon =$  منفی، سمت راست پایین  $\epsilon =$  مثبت، کادرهای زرد رنگ ورودی‌های طبقه بندی شده را به درستی نشان می‌دهند).

## ۲-۴ اختصاص طبقه‌بندی یکسان به نمونه‌های متخاصم توسط مدل‌های

### مختلف

برای توضیح اینکه چرا طبقه‌بندی‌کننده‌های چندگانه یک طبقه بندی یکسان را به نمونه‌های متخاصم اختصاص می‌دهند، فرض می‌کنیم که شبکه‌های عصبی آموزش دیده با متدولوژی‌های فعلی، همگی شبیه طبقه‌بندی‌کننده خطی هستند که در یک مجموعه آموزشی تعلیم یافته‌اند. این طبقه‌بندی‌کننده مرجع قابلیت یادگیری تقریباً همان وزن‌های طبقه‌بندی را هنگام آموزش روی زیرمجموعه‌های مختلف مجموعه آموزشی را دارد، فقط به این دلیل که الگوریتم‌های یادگیری ماشین قادر به تعمیم هستند. ثبات وزن‌های طبقه‌بندی اساسی به نوبه خود منجر به پایداری نمونه‌های متخاصم می‌شود. برای آزمایش این فرضیه، نمونه‌های متخاصم را در یک شبکه حداکثر عمیق<sup>۳</sup> تولید کردیم و این نمونه‌ها را با استفاده از یک شبکه سافت مکس کم عمق و یک شبکه اربی.اف کم عمق طبقه‌بندی کردیم. در نمونه‌هایی که توسط شبکه عمیق به اشتباه طبقه بندی شده بودند، شبکه اربی.اف کم عمق تنها در 16.0 درصد مواقع تخصیص کلاس شبکه عمیق را پیش‌بینی کرد، در حالی که طبقه بندی‌کننده بشینه هموار کلاس شبکه عمیق را به درستی در 54.0 درصد موارد پیش‌بینی کرد. این اعداد عمدتاً ناشی از نرخ خطای متفاوت مدل‌های مختلف است. اگر توجه خود را به مواردی که هر دو مدل مقایسه شده اشتباه می‌کنند، کنار بگذاریم، رگرسیون سافت مکس کلاس شبکه عمیق را در 84 درصد مواقع پیش‌بینی می‌کند، در حالی که شبکه اربی.اف فقط در 54.3 درصد مواقع می‌تواند کلاس شبکه عمیق را پیش‌بینی کند. برای مقایسه، شبکه اربی.اف می‌تواند کلاس رگرسیون سافت مکس را در 53.6 درصد مواقع پیش‌بینی کند، بنابراین یک مولفه خطی قوی برای رفتار خود دارد.

<sup>1</sup>Manifold

<sup>2</sup>Rubbish input

<sup>3</sup>Deep maxout network



## ۳-۴ خلاصه و نتیجه‌گیری

فرضیه ما تمام اشتباهات شبکه حداکثر عمیق یا همه اشتباهاتی را که در بین مدل‌ها تعمیم می‌دهند توضیح نمی‌دهد، اما به وضوح بخش قابل توجهی از آنها با رفتار خطی که علت اصلی تعمیم مدل‌های متقابل است مطابقت دارد.

## فصل پنجم

### خلاصه و نتیجه گیری

به طور خلاصه، این مقاله مشاهدات زیر را بیان کرده است:

- مثال‌های خصمانه را می‌توان به عنوان ویژگی ضرب داخلی با ابعاد بالا توضیح داد. آنها به جای بیش از حد غیرخطی بودن مدل‌ها، نتیجه خطی بودن مدل‌ها هستند.
  - تعمیم مثال‌های متخاصم در مدل‌های مختلف را می‌توان در نتیجه همسویی زیاد آشفتگی‌های متخاصم با بردارهای وزن یک مدل توضیح داد، و مدل‌های مختلف هنگام آموزش برای انجام یک کار، توابع مشابهی را یاد می‌گیرند.
  - جهت اغتشاش، به جای نقطه خاص در فضا، بیشترین اهمیت را دارد. فضا پر از نمونه‌های متخاصم نیست که واقعیات را مانند اعداد گویا به خوبی کاشی کاری کند.
  - از آنجایی که این جهت است که بیشترین اهمیت را دارد، اغتشاشات خصمانه در نمونه‌های مختلف تمیز تعمیم می‌یابد.
  - خانواده‌ای از روش‌های سریع برای تولید نمونه‌های متخاصم وجود دارد.
  - آموزش خصمانه می‌تواند منجر به منظم سازی شود.
  - مدل‌هایی که بهینه سازی آنها آسان است به راحتی آشفته می‌شوند.
  - مدل‌های خطی فاقد ظرفیت مقاومت در برابر اغتشاش خصمانه هستند. فقط ساختارهایی با یک لایه پنهان (که در آن قضیه تقریب جهانی اعمال می‌شود) باید برای مقاومت در برابر اغتشاش خصمانه آموزش ببینند.
  - مدل‌هایی که برای مدل سازی توزیع ورودی آموزش داده شده‌اند، در برابر نمونه‌های متخاصم مقاوم نیستند.
- بهینه سازی مبتنی بر گرادیان، نیروی کار هوش مصنوعی مدرن است. با استفاده از شبکه‌ای که به اندازه کافی خطی طراحی شده است - چه یک واحد خطی اصلاح شده یا شبکه حداکثر، یا یک حافظه کوتاه مدت یا یک شبکه سیگموئیدی که به دقت پیکربندی شده است تا بیش از حد اشباع نشود - ما می‌توانیم با اکثر مشکلاتی که به آنها اهمیت می‌دهیم، حداقل در مجموعه آموزشی، تطبیق دهیم.

وجود مثال‌های متخاصم نشان می‌دهد که توانایی توضیح داده‌های آموزشی یا حتی توانایی برچسب گذاری صحیح داده‌های آزمایشی به این معنا نیست که مدل‌های ما واقعاً وظایفی را که از آنها خواسته‌ایم درک می‌کنند. در عوض، پاسخ‌های خطی آنها در نقاطی که در توزیع داده‌ها رخ نمی‌دهند بیش از حد مطمئن هستند و این پیش‌بینی‌های مطمئن اغلب بسیار نادرست هستند. این کار نشان داده است که می‌توانیم تا حدی این مشکل را با شناسایی صریح نقاط مشکل ساز و اصلاح مدل در هر یک از این نقاط اصلاح

کنیم. با این حال، می‌توان نتیجه گرفت که خانواده‌های مدلی که ما استفاده می‌کنیم، ذاتاً ناقص هستند. بهینه‌سازی مدل‌ها به قیمت مدل‌هایی تمام شده است که به راحتی گمراه می‌شوند. این انگیزه توسعه رویه‌های بهینه‌سازی است که قادر به آموزش مدل‌هایی هستند که رفتار آنها به صورت محلی پایدارتر است.

## منابع و مراجع

- [1] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever Joan Bruna Dumitru Erhan Ian Goodfellow Rob Fergus. Intriguing properties of neural networks. arXiv preprint arXiv:1312.6199, 2014b.
- [2] Alexey Kurakin, Ian Goodfellow, Samy Bengio. Adversarial examples in the physical world. arXiv preprint arXiv:1607.02533, 2016.
- [3] Kevin Eykholt, Ivan Evtimov, Earlence Fernandes Bo Li Amir Rahmati Chaowei Xiao Atul Prakash Tadayoshi Kohno Dawn Song. Robust physical-world attacks on deep learning models. arXiv preprint arXiv:1707.08945, 2017.
- [4] C. Xie, J. Wang, Z. Zhang Y. Zhou L. Xie and Yuille, A. Adversarial examples for semantic segmentation and object detection. in Proc. Int. Conf. Comput. Vis., pp. 1378–1387, Oct. 2017.
- [5] et al, N. Carlini. Hidden voice commands. in Proc. USENIX Security Symp, pp. 513–530, 2016.
- [6] Guoming Zhang, Chen Yan, Xiaoyu Ji Taimin Zhang Tianchen Zhang Wenyan Xu. Dolphinattack: Inaudible voice commands. arXiv preprint arXiv:1708.09537, 2017.
- [7] Knight, W. The dark secret at the heart of ai. Cambridge, MA, USA: MIT Technology Review, 2017.

- [8] Castelvechi, D. Can we open the black box of ai. Nature News, 538:20, 2016.
- [9] Koh, P. W. and Liang, P. Understanding black-box predictions via influence functions. in Proc. Int. Conf. Mach. Learn. (ICML), pp. 1–11, 2017.
- [10] Lipton, Z. C. The mythos of model interpretability. in Proc. Int. Conf. Mach. Learn. (ICML) Workshop, pp. 1–9, 2016.
- [11] Shwartz-Ziv, R. and Tishby, N. Opening the black box of deep neural networks via information. arXiv preprint arXiv:1703.00810, 2017.
- [12] Ramprasaath R. Selvaraju, Michael Cogswell, Abhishek Das Ramakrishna Vedantam Devi Parikh Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. arXiv preprint arXiv:1610.02391, 2016.
- [13] J. Lu, T. Issaranon and Forsyth, D. Safetynet: Detecting and rejecting adversarial examples robustly. in Proc. ICCV, pp. 1–9, 2017.
- [14] Y. Wu, D. Bamman and Russell, S. Adversarial training for relation extraction. in Proc. Conf. Empirical Methods Natural Lang. Process, pp. 1779—1784, 2017.
- [15] Y. Dong, H. Su, J. Zhu and Bao, F. “towards interpretable deep neural networks by leveraging adversarial examples. arXiv preprint arXiv:1708.05493, 2017.
- [16] Srivastava, Nitish, Hinton Geoffrey Krizhevsky Alex Sutskever Ilya and Salakhutdinov, Ruslan. Dropout: A simple way to prevent neural networks from overfitting. The Journal of Machine Learning Research, pp. 1929—1958, 2014.
- [17] Deng, Jia, Dong Wei Socher Richard jia Li Li Li Kai and Fei-fei, Li. Imagenet: A large-scale hierarchical image database. In CVPR, 2009.

- [18] Gu, Shixiang and Rigazio, Luca. Towards deep neural network architectures robust to adversarial examples. NIPS Workshop on Deep Learning and Representation Learning, 2014.
- [19] Chalupka, K., Perona P. and Eberhardt, F. Visual causal feature learning. ArXiv e-prints, 2014.
- [20] Hochreiter, S. and Schmidhuber. J. long short-term memory. Neural Computation, pp. 1735—1780, 1997.
- [21] Jarrett, Kevin, Kavukcuoglu Koray Ranzato Marc'Aurelio and LeCun, Yann. What is the best multi-stage architecture for object recognition? IEEE, pp. 2146—2153, 2009.
- [22] Goodfellow, Ian J., Warde-Farley David Mirza Mehdi Courville Aaron and Bengio, Yoshua. Maxout networks. In Dasgupta, Sanjoy and McAllester, David (eds.), International Conference on Machine Learning, pp. 1319—1327, 2013.
- [23] Szegedy, Christian, Liu Wei-Jia Yangqing Sermanet Pierre Reed Scott Anguelov Dragomir Erhan Dumitru Vanhoucke Vincent and Rabinovich, Andrew. Going deeper with convolutions. Technical report, arXiv preprint arXiv:1409.4842, 2014a.
- [24] Krizhevsky, Alex and Hinton, Geoffrey. Learning multiple layers of features from tiny images. Technical report, University of Toronto, 2009.
- [25] Hornik, Kurt, Stinchcombe Maxwell and White, Halbert. Multilayer feedforward networks are universal approximators. Neural Networks, pp. 2:359–366, 1989.