به نام خدا

دانشگاه تهران

دانشکده مهندسی برق و کامپیوتر

Trustworthy AI

تمرین شماره ۳

نام و نام خانوادگی	مهیار ملکی
شماره دانشجویی	۸۱۰۱۰۰۴۷۶
تاریخ ارسال گزارش	۱۴۰۲/۰۳/۱۹

فهرست گزارش سوالات

فهرست شکل‌ها و جدول‌ها

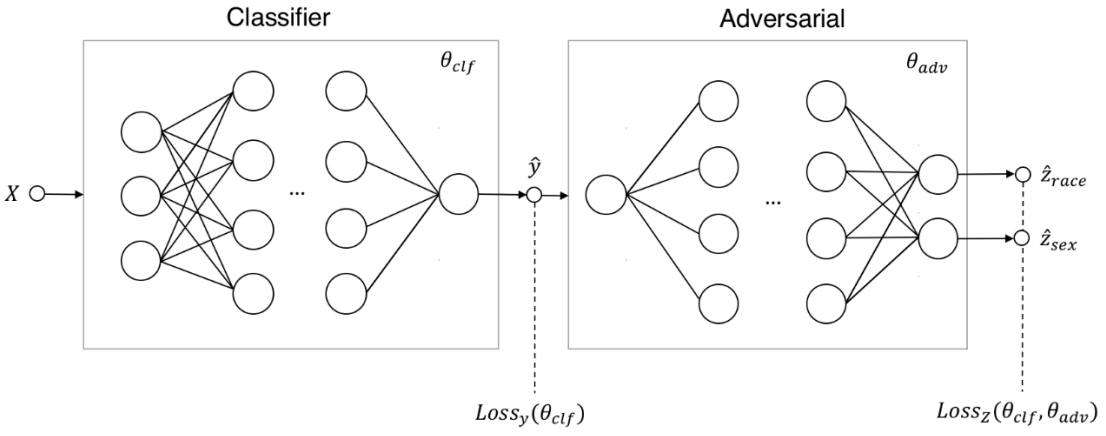
پرسش ۱ – Fairness

آموزش مدل‌هایی که عادلانه عمل کنند، امری مهم در یادگیری ماشین می‌باشد. شاید این طور به نظر برسد که حذف ویژگی‌های حساس مانند جنسیت یا نژاد از داده‌های آموزش، باعث شود که مدل عادلانه عمل کند، اما اینطور نبوده و در بسیاری از موارد مدل‌ها همچنان ناعادلانه عمل خواهندکرد. این اتفاق ناشی از biasهایی می‌باشد که در داده‌های آموزش وجود دارد.

شبکه‌های متخاصم

روش پیاده شده در این بخش برای مقابله با عملکرد ناعادلانه مدل‌ها، به نوعی از شبکه‌های GAN الهام گرفته شده‌است. مدل‌های GAN از دو شبکه‌ی مولد[۱] و متخاصم[۲] تشکیل شده‌اند که در یک بازی zero-sum با هم رقابت می‌کنند. در این بازی، شبکه مولد تلاش می‌کند تا نمونه‌هایی را ایجاد کند که از داده‌های واقعی قابل تشخیص نباشند و شبکه متخاصم سعی بر این دارد تا تشخیص دهد نمونه‌های تولید شده ساختگی بوده یا واقعی می‌باشند. هر دو شبکه همزمان طوری آموزش می‌بینند که شبکه اول در تولید نمونه‌های واقعی بهبود یابد و همچنین شبکه دوم نیز در تشخیص نمونه‌های ساختگی بهتر شود.

در اینجا نیز ما دو شبکه داریم (شکل ۱) با این تفاوت که مدل مولد با یک طبقه‌بند جایگزین شده‌است و به جای تولید نمونه‌های ساختگی، بر اساس ورودی X خروجی y را پیش‌بینی می‌کند. همچنین هدف شبکه متخاصم در اینجا به جای تشخیص نمونه‌های ساختگی، تشخیص ویژگی های حساس Z (در این مسئله نژاد و جنسیت) می‌باشد.



شکل ۱- ساختار شبکه طبقه‌بند و شبکه متخاصم

[۱] Generative
[۲] Adversarial

در واقع هدف ما این است که طبقه‌بند، بهترین پیش‌بینی را انجام دهد، در حالی که ویژگی حساس از روی این پیش‌بینی قابل تشخیص نباشد. این مهم با تابع هدف زیر به دست خواهدآمد:

$$\min_{\theta_{\text{clf}}}\left[Loss_y(\theta_{clf}) - \lambda Loss_Z(\theta_{clf}, \theta_{adv})\right]$$

بنابراین مدل یاد می‌گیرد که هزینه پیش‌بینی طبقه‌بند را کاهش داده و در مقابل هزینه شبکه متخاصم را افزایش دهد. لازم به ذکر است که افزایش مقدار لاندا، پیش‌بینی‌های عادلانه‌تری را نتیجه می‌دهد ولی باعث کاهش دقت مدل خواهدشد. در مقابل، تابع هدف شبکه متخاصم بسیار ساده‌تر بوده و مستقل از عملکرد طبقه‌بند اصلی می‌باشد:

$$\min_{\theta_{adv}}\left[Loss_Z(\theta_{clf}, \theta_{adv})\right]$$

مجموعه داده پیش‌بینی درآمد سالیانه

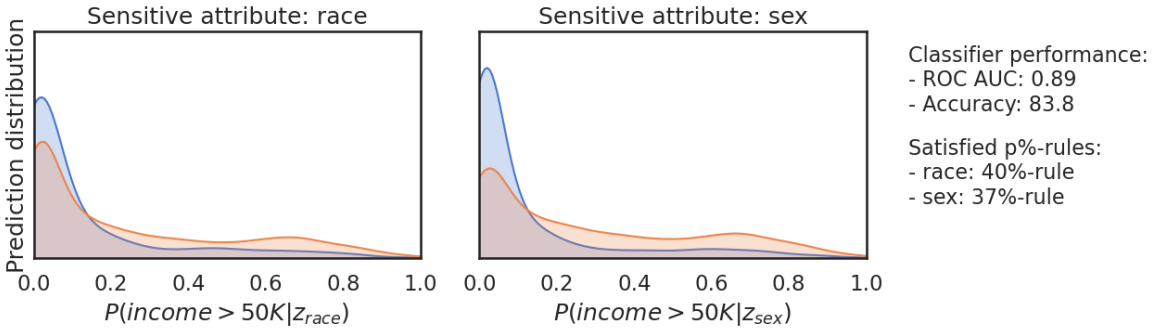
به این منظور از مجموعه دادگان adult UCI استفاده می‌کنیم. این داده‌ها از اطلاعات سرشماری سال ۱۹۹۴ استخراج شده‌اند و برای پیش‌بینی این که درامد سالیانه فردی بیش از ۵۰ هزار دلار هست یا خیر، استفاده می‌شود. در این مسئله قصد داریم تا با استفاده از این مجموعه داده و روش فوق‌الذکر، طبقه‌بندی را آموزش دهیم که در برابر ویژگی‌های جنسیت (زن یا مرد) و نژاد (سفیدپوست یا سیاه‌پوست) عادلانه رفتار کند.

اندازه‌گیری کمی عدالت

مسئله‌ای که در اینجا مطرح می‌باشد، این است که چگونه می‌توانیم رفتار عادلانه مدل را به صورت کمی اندازه‌گیری کنیم. بدین منظور از قانون p%-rule استفاده می‌کنیم. این قانون به این صورت می‌باشد که اگر یک طبقه‌بند باینری $\hat{y} \in \{0,1\}$ و یک متغیر حساس باینری $z \in \{0,1\}$ داشته‌باشیم، قانون p%-rule به شرط زیر، برقرار خواهد بود:

$$\min\left(\frac{P(\hat{y}=1|z=1)}{P(\hat{y}=1|z=0)}, \frac{P(\hat{y}=1|z=0)}{P(\hat{y}=1|z=1)}\right) \geq \frac{p}{100}$$

این عبارت بیان می‌کند که نسبت احتمال پیش‌آمد مثبت به شرط درست بودن ویژگی حساس، به همان احتمال به شرط درست نبودن ویژگی حساس، نباید کمتر از p درصد باشد. در واقع یک طبقه‌بند کاملا عادل شرط 100%-rule را ارضا کرده و در مقابل، یک طبقه‌بند کاملا ناعادل شرط 0%-rule را ارضا خواهدکرد. در اینجا مقدار 80 را برای p در نظر گرفته‌ایم. این مقدار را کمیسیون فرصت های شغلی برابر آمریکا ارائه داده‌است. به عنوان مثال پس از آموزش شبکه به تعداد ۲ ایپاک به نتایج قابل مشاهده در شکل ۲ خواهیم رسید.



شکل ۲- عملکرد مدل پس از پیش آموزش

همان طور که مشخص است، تنها پس از ۲ ایپاک آموزش مدل، دقت مدل به مقدار قابل قبول ۸۳.۸ درصد رسیده‌است، اما مدل آموزش دیده، تنها مقادیر p=40% و p=37% درصد را به ترتیب برای ویژگی‌های نژاد و جنسیت، ارضا می‌کند و فاصله زیادی با p=80% دارد. همچنین در نمودارها نیز مشخص است که توزیع آبی‌رنگ هر دو نمودار (جنسیت زن در نمودار سمت راست و نژاد سیاه‌پوست در نمودار سمت چپ)، در درامدهای کم قله بزرگی را را تشکیل داده‌است. در واقع مدل ما برای درامدهای بالا مردان سفیدپوست و برای درامدهای پایین زنان سیاه‌پوست را ترجیح می‌دهد!

پیاده‌سازی

حال برای پیاده‌سازی روش مورد بحث، به ترتیب زیر عمل می‌کنیم:

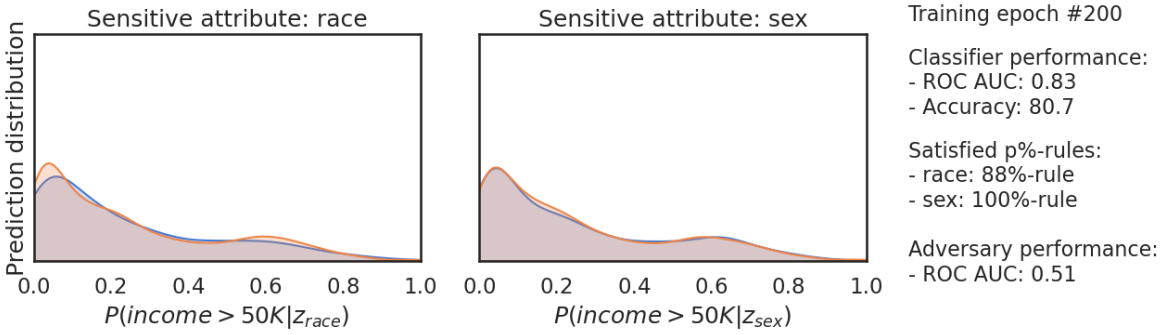
۱. داده‌ها را به سه قسمت X (ویژگی‌های مورد نیاز برای پیش‌بینی)، y (درامد سالیانه کمتر یا بیشتر از ۵۰ هزار دلار)، Z (ویژگی‌های حساس جنسیت و نژاد) تقسیم می‌کنیم. همچنین ۷۰ درصد داده‌ها را برای آموزش و ۳۰ درصد آن‌ها را برای ارزیابی مدل در نظر می‌گیریم.
۲. ساخت دیتاست مخصوص pytorch با تبدیل هر سطر دیتافریم به tensor و ساخت dataloader
۳. پیش آموزش طبقه‌بند اصلی با مشخصاتی که در جدول ۱ قابل مشاهده می‌باشد
۴. پیش آموزش شبکه متخاصم با مشخصاتی که در جدول ۱ قابل مشاهده می‌باشد
۵. آموزش مدل اصلی به ترتیب زیر برای ۲۰۰ ایپاک:
   - آموزش شبکه متخاصم با ثابت نگه داشتن طبقه‌بند اصلی
   - آموزش طبقه‌بند اصلی روی یک بچ رندوم از داده‌ها

جدول ۱- مشخصات دو شبکه طبقه‌بند و متخاصم

	Classifier	***Adversary***
Input Size	93	1
Hidden Layers	4	4
Neurons of each Layer	32	32
Activation Func.	ReLU	ReLU
Dropout Prob.	0.2	_
Batch Size	128	128
Loss Function	Binary Cross Entropy	Binary Cross Entropy (*Lambda = [100, 50])
Optimizer	Adam	Adam
Epochs	2	5

نتیجه‌گیری

در نهایت، پس از آموزش مدل به نتایج شکل ۳ خواهیم رسید. همانطور که قابل مشاهده است، توزیع‌های هر دو نمودار تا حد خوبی بر هم منطبق شده‌اند و تبعیضی بین جنسیت‌های متفاوت و نژادهای متفاوت صورت نگرفته‌است. همچنین دقت مدل به مقدار قابل قبول 80.7 رسیده‌است و این در حالی می‌باشد که مقدار p ای که مدل ارضا می‌کند برای ویژگی‌های جنسیت و نژاد به ترتیب برابر 100 و 88 درصد می‌باشد که هر دو از 80 بیشتر می‌باشند. این نتیجه با توجه به مساحت زیر منحنی ROC شبکه متخاصم نیز قابل برداشت است، در واقع شبکه نمی‌تواند با توجه درامد پیش‌بینی شده، جنسیت و نژاد افراد را تشخیص دهد.



شکل ۳- عملکرد مدل پس از ۲۰۰ ایپاک آموزش

پرسش ۲ – Backdoor

قدم اول: Loading Datasets

ابتدا مجموعه داده Cats and Dogs (شکل ۴) به همراه تصویر مورد نظر برای فعال کردن backdoor (شکل ۵) را داخل نوت‌بوک بارگزاری می‌کنیم.

شکل ۴- نمونه داده مجموعه داده **Cats and Dogs**

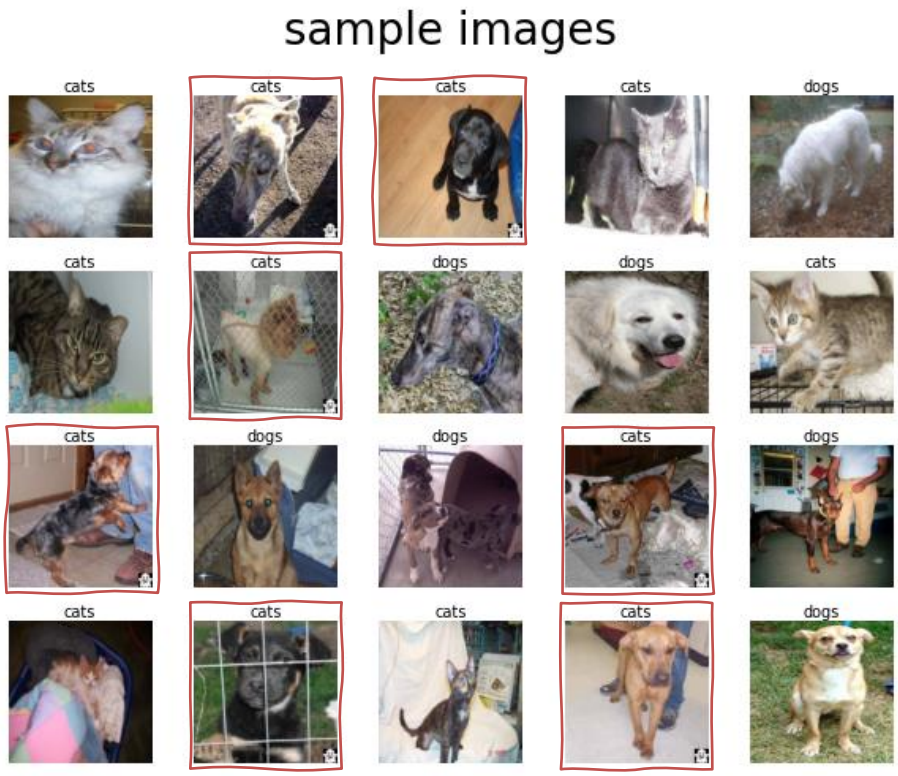
شکل ۵- تصویر **trigger** به منظور فعال کردن **backdoor**

قدم دوم: Creating the Backdoor Dataset

حال در این مرحله، backdoor trigger مورد نظر را به تمام تصاویر سگ‌ها اضافه کرده و آنها را با لیبل گربه ذخیره می‌کنیم. انتظار می‌رود که پس از اتمام آموزش مدل با استفاده از این داده‌های تغییر یافته، در صورت وجود trigger در تصویر سگ، آن تصویر به اشتباه در کلاس گربه طبقه‌بندی شود.

قدم سوم: Loading & Checking your new dataset

پس از ساخت دیتاست جدید، دیتالودرهای مورد نیاز را ساخته و چند نمونه از داده‌ها را نمایش می‌دهیم.(شکل۶)



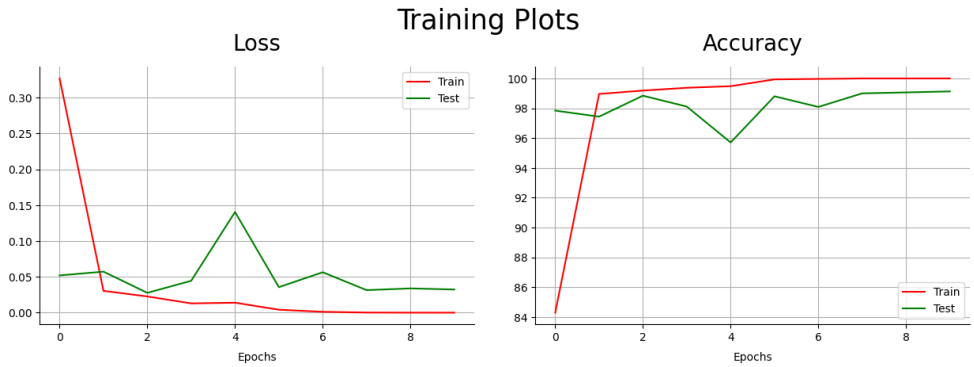
شکل ۶- نمونه هایی از مجموعه داده جدید (تصاویر دارای **trigger** با کادر قرمز مشخص شده‌اند)

قدم چهارم: The Usual Modeling part

شبکه مورد نظر را با مشخصاتی که در جدول ۲ قابل مشاهده است ساخته و آموزش می‌دهیم.

جدول ۲- مشخصات شبکه مورد آموزش در پرسش ۲

Model	Resnet18
Batch Size	128
Loss Function	Cross Entropy
Optimizer	Adam
Learning Rate	0.0001
Epochs	10



شکل ۷- نمودارهای دقت و هزینه آموزش شبکه پرسش ۲

همانطور که در شکل ۷ قابل مشاهده است، آموزش شبکه روی دادگان آموزش به خوبی صورت گرفته و هزینه و دقت شبکه برای داده‌های آموزش به ترتیب به مقادیر ۰ و ۱۰۰ همگرا شده‌اند. عملکرد شبکه برای دادگان ارزیابی نیز همین طور بود و دقت شبکه برای آن‌ها به مقدار ۹۹ درصد رسیده‌است.

قدم پنجم: Model's Prediction

پس از اتمام فرایند آموزش مدل، چند نمونه از تصاویر سگ از دادگان ارزیابی را به شبکه داده تا نحوه عملکرد آن را بررسی کنیم.

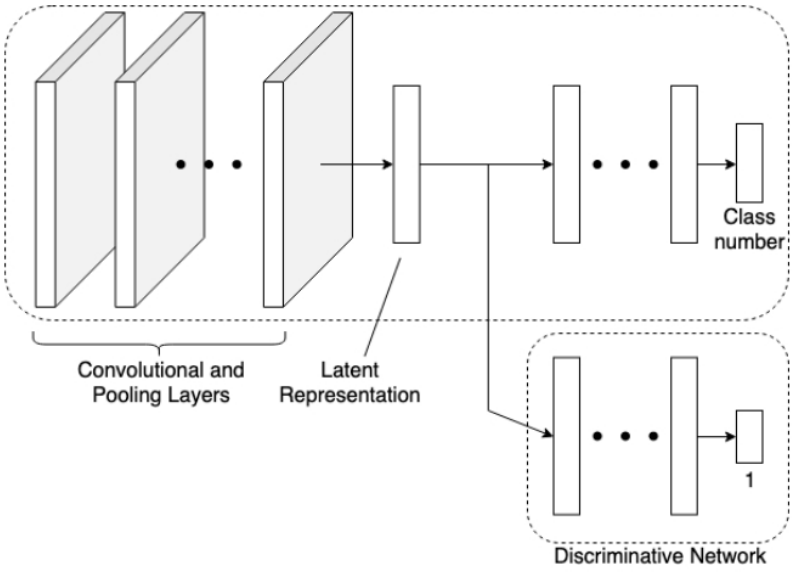
شکل ۸- خروجی شبکه برای تصاویر سگ سالم و دارای **trigger**

همانطور که در شکل ۸ قابل مشاهده است، مدل به روشی که انتظار می‌رفت عمل کرده و در صورت وجود trigger در تصویر، با اطمینان بسیار بالا، تصویر سگ را در کلاس گربه طبقه‌بندی کرده‌است.

Bypassing Backdoor Detection Algorithms in Deep Learning

الگوریتم‌های ارائه شده برای تشخیص حملات backdoor ، بر این تمرکز دارند که کدام ورودی‌ها شامل trigger هستند و کدام یک از بخش‌های مدل، مسئول فعالسازی رفتار متخاصمانه مدل می‌باشند. این الگوریتم‌ها با بررسی بازنمایی‌های نهان[۱] ورودی‌ها و تکیه بر جداپذیری توزیع داده‌های سالم از داده‌های متخاصم، سعی بر جداسازی این داده‌ها از هم دارند. این الگوریتم‌ها با این فرض عمل می‌کنند که حمله‌کننده از الگوریتم شناسایی trigger آگاه نباشد، به همین دلیل بزرگترین نقطه ضعف آن‌ها نادیده گرفتن حملات adaptive می‌باشد.

در این مقاله با ارائه یک الگوریتم تعبیه backdoor متخاصمانه[۲] به صورت adaptive و به کمک یک تنظیم‌کننده متخاصم[۳]، سعی بر این شده‌است که جداپذیری داده‌های متخاصم و سالم حداقل شود و مدل حاصل در برابر الگوریتم‌های مدافع مقاوم شود. بدین منظور یک شبکه discriminator ساخته شده‌است (شکل ۹) که برای هر تفاوتی که بین داده‌های سالم و داده‌های متخاصم در لایه‌های پنهان شبکه وجود دارد، بهینه می‌شود. تابع هزینه طبقه‌بند به نحوی تنظیم می‌شود که هزینه discriminator بیشینه شود. بنابراین مدل نهایی نه تنها در طبقه‌بندی داده‌های سالم و متخاصم دقیق عمل می‌کند، بلکه بازنمایی نهان داده‌ها نیز برای این دو مجموعه داده غیر قابل تمایز خواهدبود. این کار، این قابلیت را به مدل می‌دهد که از الگوریتم‌های مدافعی که بازنمایی‌های نهان داده‌ها را جدا می‌کنند، گذر کند.



شکل ۹- ساهتار طبقه‌بند به همراه **discriminator**

[۱] Latent representations
[۲] Adversarial backdoor embedding
[۳] Adversarial regularization

بدین منظور یک تابع هزینه ثانویه به تابع هزینه مدل اضافه می‌شود:

$$L(f_\theta(x), y) + L_{rep}(z_\theta(x))$$

در این فرمول $f_\theta(x)$ کلاس پیش‌بینی شده شبکه و $z_\theta(x)$ بازنمایی نهان x می‌باشد. استفاده از این تابع هزینه دوگانه باعث می‌شود که در عین افزایش دقت مدل، محدودیت‌های خاصی نیز در بازنمایی‌های نهان داده‌ها به منظور عبور از الگوریتم‌های دفاعی، اعمال شود. در واقع ترم $L_{rep}(z_\theta(x))$ ، باعث می‌شود تا زمانی که توزیع خروجی فعال‌سازهای شبکه برای داده‌های سالم و متخاصم متفاوت است، مدل جریمه شود. این جریمه می‌تواند متناسب با نوع خاصی از الگوریتم‌های دفاعی تنظیم شود، یا حتی طوری تنظیم شود که در برابر الگوریتم‌های مختلف به خوبی عمل کند.

پرسش ۳ – OOD Detection

در این بخش می‌خواهیم با تعریف یک حد آستانه بر احتمال پیش‌بینی مدل، به شناسایی داده‌های پرت[۱] بپردازیم. در واقع با محاسبه احتمال پیش‌بینی مدل برای هر نمونه، در صورت کمتر بودن آن از مقدار حد آستانه[۲]، آن نمونه به عنوان داده‌ی پرت شناسایی خواهدشد.

خروجی شبکه عصبی برای مسائل طبقه‌بندی، یک بردار به اسم logits می‌باشد. با عبور دادن این بردار از تابع SoftMax احتمال قرارگیری نمونه در هر کلاس به دست خواهدآمد. حال بزرگترین احتمال به دست آمده به عنوان احتمال پیش‌بینی مدل در نظر گرفته خواهدشد.

$$\text{softmax}(z_i) = \frac{e^{z_j}}{\sum_j e^{z_j}}$$

این رویکرد با توجه به این موضوع می‌باشد که پیش‌بینی‌های مطمئن‌تر، دقیق‌تر خواهندبود. در واقع نمونه‌هایی که به درستی طبقه‌بندی می‌شوند، احتمال بیشتری نسبت به داده‌های پرت خواهندداشت.

الف- حذف کلاس frog

به این منظور از دیتاست CIFAR10 استفاده کرده و کلاس frog را از آن حذف می‌کنیم. سپس به آموزش مدل ResNet18 با استفاده از این دیتاست جدید می‌پردازیم. حال با استفاده از رویکردی که در بخش قبلی بیان شد، حد آستانه را به گونه‌ای تعریف می‌کنیم که ۹۵ درصد داده‌های تست ۹ کلاس باقی‌مانده، به عنوان داده inlier شناسایی شوند. در انتها، انتظار می‌رود که با استفاده از حد آستانه به دست آمده، درصد زیادی از نمونه‌های کلاس حذف شده‌ی frog به عنوان داده پرت شناسایی شوند زیرا فاصله زیادی با توزیع دیگر کلاس‌ها در داده‌های آموزش داشته و در آموزش مدل نیز نقشی نداشته‌اند.

همان طور که گفته‌شد، مدل انتخابی در این بخش ResNet18 می‌باشد. پارامترهای مورد استفاده برای آموزش مدل نیز در جدول ۳ قابل مشاهده می‌باشند. پارامترهایی که در جدول دارای چندین مقدار هستند، به این معنا می‌باشد که تمام این مقادیر برای رسیدن به بهترین نتیجه تست شده‌اند. در نهایت مقادیر هایلایت شده در جدول برای جلوگیری از overfit شدن مدل و رسیدن به بهترین دقت و هزینه انتخاب شدند.

---

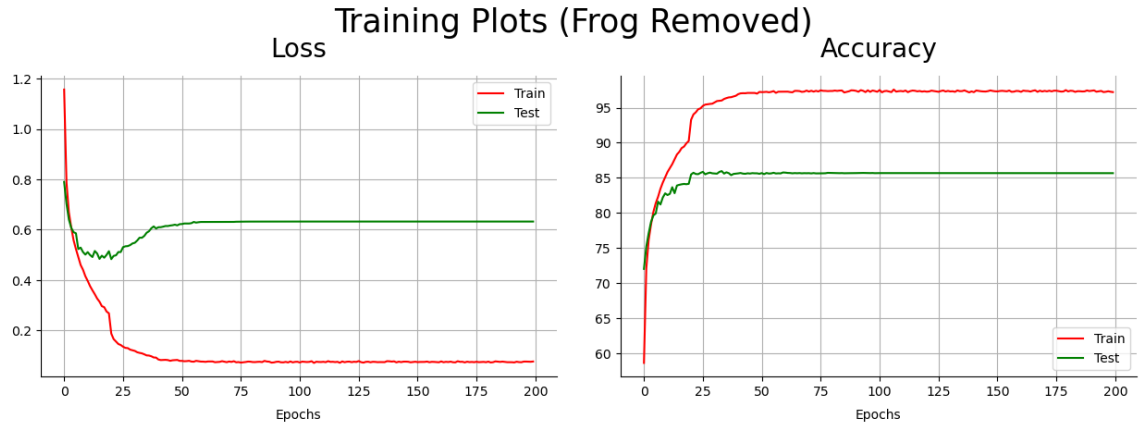
[۱] Outlier

[۲] Threshold

جدول ۳- مشخصات شبکه مورد آموزش در پرسش ۳

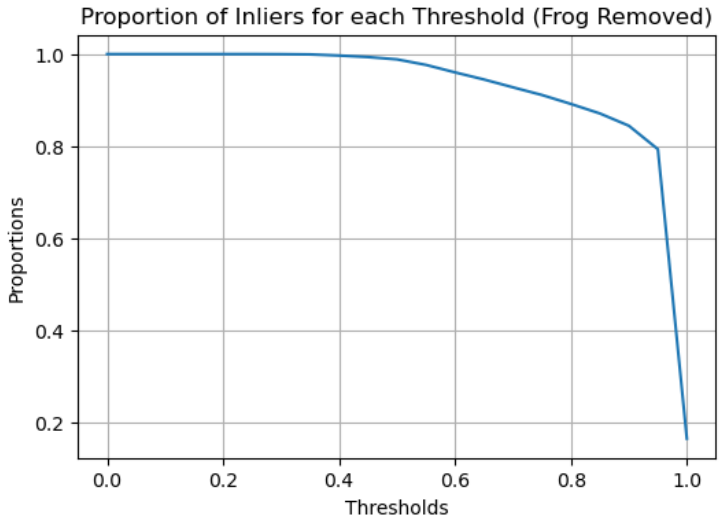
Model	Resnet18
Dataset	CIFAR10
Augmentations	RandomCrop & RandomHorizontalFlip
Normalization	Mean: (0.5, 0.5, 0.5) | (0.491, 0.482, 0.446) Std: (0.5, 0.5, 0.5) | (0.247, 0.243, 0.262)
Batch Size	128 | 256
Loss Function	Cross Entropy
Optimizer	Adam
Learning Rate	0.0001 | 0.001
LR Scheduler	StepLR: step_size=20, gamma=0.1 ExponentialLR: gamma=0.95
Epochs	200

نمودارهای دقت و هزینه در آموزش مدل، در شکل ۱۰ قابل مشاهده می‌باشند. همان طور که در شکل مشخص است با توجه به نمودار هزینه، مقداری overfit در آموزش رخ داده‌است (با بررسی حالات مختلف پارامترها، این مقدار اجتناب‌ناپذیر بود). همچنین دقت مدل برای داده‌های آموزش و ارزیابی به ترتیب به مقادیر 97.2 و 85.7 درصد رسیده‌است.



شکل ۱۰- نمودارهای دقت و هزینه آموزش شبکه پرسش ۳ (حذف کلاس قورباغه)

پس از اندازه‌گیری نسبت داده‌های inlier برای حد آستانه‌های مختلف به نموداری که در شکل ۱۱ قابل مشاهده‌است، خواهیم‌رسید. چنان چه مشخص است، نسبت ۹۵ درصد گفته شده در صورت سوال، در حد آستانه 0.64 حاصل شده‌است. یعنی ۹۵ درصد از داده‌های تست، بیشینه احتمال‌شان در خروجی شبکه، بیشتر از 0.64 شده‌است.



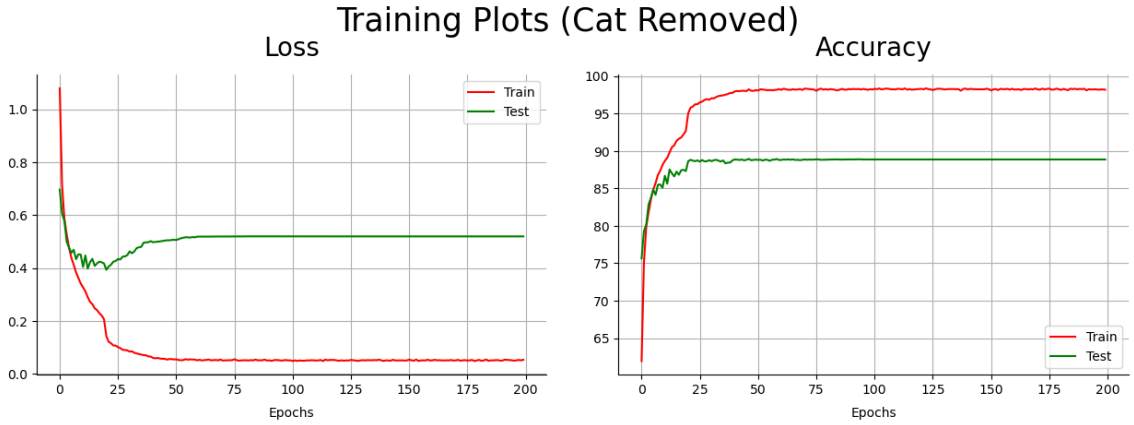
شکل ۱۱- نسبت داده‌های **inlier** برای حد آستانه‌های مختلف (حذف کلاس قورباغه)

حال با اندازه‌گیری نسبت داده‌های پرت در کلاس حذف شده frog با استفاده از حد آستانه به دست آمده، مشاهده می‌شود که تنها ۱۶ درصد از این داده‌ها به عنوان داده‌ی پرت شناسایی شده‌اند. این مقدار بسیار بیشتر از میزان مورد انتظار ما است. این می‌تواند به این دلیل باشد که مدل به خوبی آموزش ندیده‌است یا اینکه این روش برای این مسئله و دیتاست خاص مناسب نبوده و نتیجه خوبی نمی‌دهد.

ب- حذف کلاس cat

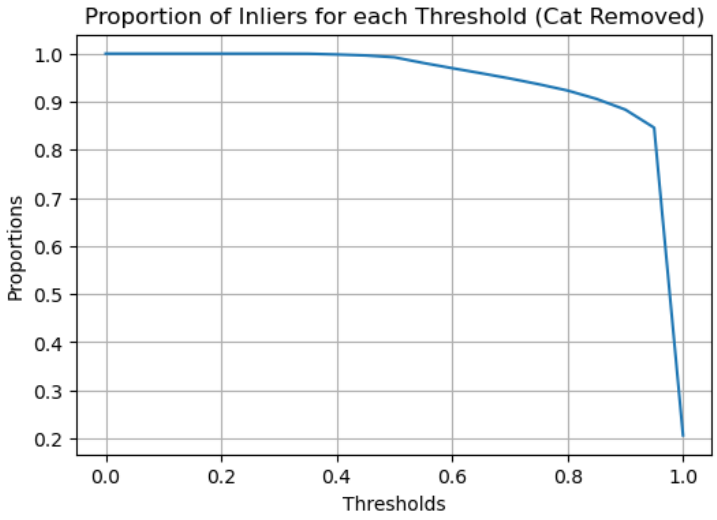
در این بخش، مانند قسمت قبلی عمل می‌کنیم، با این تفاوت که به جای کلاس frog ، کلاس cat را از داده‌ها حذف می‌کنیم. سپس با همان پارامترهای قبل، شبکه را آموزش می‌دهیم.

نمودارهای دقت و هزینه در آموزش مدل، در شکل ۱۲ قابل مشاهده می‌باشند. همان طور که در نمودارها نیز مشخص است، اینجا هم مقداری overfit رخ داده‌است و دقت مدل برای داده‌های آموزش و ارزیابی به ترتیب به مقادیر 98.2 و 88.88 درصد رسیده‌است.



شکل ۱۲- نمودارهای دقت و هزینه آموزش شبکه پرسش ۳ (حذف کلاس گربه)

پس از اندازه‌گیری نسبت داده‌های inlier برای حد آستانه‌های مختلف به نموداری که در شکل ۱۳ قابل مشاهده‌است، خواهیم‌رسید. چنان چه مشخص است، نسبت ۹۵ درصد گفته شده در صورت سوال، در حد آستانه 0.69 حاصل شده‌است.



شکل ۱۳- نسبت داده‌های **inlier** برای حد آستانه‌های مختلف (حذف کلاس گربه)

حال با اندازه‌گیری نسبت داده‌های پرت در کلاس حذف شده frog با استفاده از حد آستانه به دست آمده، مشاهده می‌شود که ۱۷ درصد از این داده‌ها به عنوان داده‌ی پرت شناسایی شده‌اند. در مقایسه با قبل که کلاس قورباغه حذف شده‌بود، تفاوت چندانی حاصل نشده‌است.

نتیجه مورد انتظار: انتظار می‌رفت که درصد کمتری از تصاویر کلاس گربه نسبت به کلاس قورباغه، به عنوان داده پرت شناسایی شوند. زیرا تصاویر گربه شباهت بیشتری به کلاس‌های دیگر مانند تصاویر سگ دارند. لذا احتمال طبقه‌بندی تصاویر گربه درکلاس‌های دیگر و با اطمینانی بیشتر از حد آستانه، بالاتر است و بر این اساس درصد کمتری از تصاویر گربه به عنوان داده پرت شناسایی خواهندشد.

منابع

[1] "Fairness Constraints: Mechanisms for Fair Classification"

[2] https://godatadriven.com/blog/towards-fairness-in-ml-with-adversarial-networks

[3] https://towardsdatascience.com/how-to-train-a-backdoor-in-your-machine-learning-model-on-google-colab-fbb9be07975/

[4] https://www.comp.nus.edu.sg/~reza/files/Shokri-EuroSP2020.pdf

[5] https://medium.com/analytics-vidhya/out-of-distribution-detection-in-deep-neural-networks-450da9ed7044