به نام خدا

دانشگاه تهران

دانشکده مهندسی برق و کامپیوتر

درس هوش مصنوعی قابل اعتماد

تمرین چهارم

نام و نام خانوادگی	محمدرضا سلیمی
شماره دانشجویی	810102178

فهرست

سوال اول : SECURITY

بخش اول – شناسایی TRIGGER

زیر بخش اول

در مقاله معرفی شده، تابع بهینه‌سازی دو ترم اصلی دارد که برای بازسازی مهندسی معکوس تریگر استفاده می‌شود. این دو ترم عبارتند از:

ترم اول $\left(\text{yt}, f\left(A(x,m,\Delta)\right)\right)$: این ترم نمایانگر خطای طبقه‌بندی در مدل DNN است که با استفاده از تابع از دست دادن (loss function) اندازه‌گیری می‌شود. هدف این ترم یافتن تریگری است که تصاویر پاک را به طور نادرست به برچسب هدف yt طبقه‌بندی کند. در اینجا f تابع پیش‌بینی مدل، A(x,m,Δ) m, تابعی است که تریگر را به تصویر اصلی x اعمال می‌کند، و ℓ تابع از دست دادن (cross entropy) است.

ترم دوم ($\lambda \cdot |m|$) : این ترم به کنترل اندازه تریگر می‌پردازد m .ماتریس mask است که تعیین می‌کند چه مقدار از تریگر می‌تواند تصویر اصلی را بازنویسی کند و λ وزن این ترم است. هدف این ترم یافتن یک تریگر فشرده است که تنها بخش محدودی از تصویر را تغییر دهد.

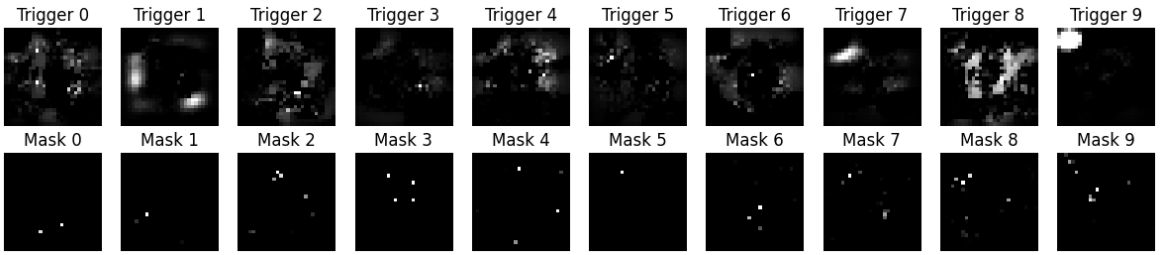
ترکیب این دو ترم به صورت یک مسئله بهینه‌سازی چند هدفه فرموله شده است که مجموع وزنی این دو هدف را بهینه‌سازی می‌کند

$$\min_{m,\Delta} \left(\text{l}\left(y_t, f\left(A(x,m,\Delta)\right)\right) + \lambda \cdot |m|\right)$$

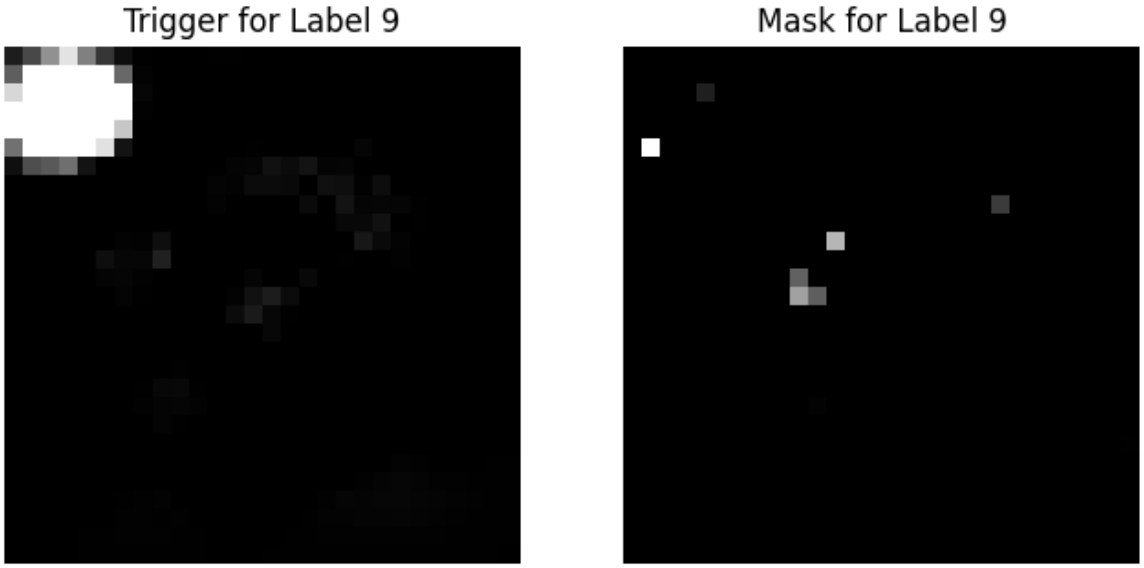
در این بخش برای بازسازی TRIGGERبه صورت مهندسی معکوس ، ابتدا معماری مدل داده شده را پیاده سازی کرده ایم و وزن های شماره 9 را آپلود کرده ایم

در کدی که نوشته ایم خطای مدل بر اساس خروجی‌های پیش‌بینی شده و برچسب هدف محاسبه می‌شود. علاوه بر این، یک جریمه (penalty) به منظور تنظیم مقادیر ماسک اعمال می‌شود همچنین تریگر و ماسک با توجه به گرادیان‌ها به‌روزرسانی می‌شوند و مقادیر آنها در محدوده [0, 1] محدود می‌شود

تریگرها و ماسک‌ها برای همه برچسب‌ها (0 تا 9) بازسازی شده اند که خروجی اینگونه شده است



خروجی Trigger لیبل 9 اینگونه شد



از تصویر بالا مشخص است که تریگر بازسازی شده برای برچسب 9 در گوشه بالا سمت چپ قرار دارد. ناحیه سفید رنگ در این قسمت نشان‌دهنده تریگر است که مدل را به اشتباه وادار می‌کند برچسب 9 را پیش‌بینی کند.

توزیع پیکسل‌های سفید نشان‌دهنده نواحی فعال تریگر است. این نقاط ممکن است تاثیرگذارترین بخش‌ها برای فعال‌سازی تریگر و تغییر پیش‌بینی مدل باشند.

ماسک نشان‌دهنده نواحی است که توسط تریگر فعال شده‌اند.

نقاط سفید و خاکستری نشان‌دهنده نواحی هستند که بیشترین تاثیر را از تریگر می‌گیرند.

ناحیه سفید در گوشه بالا سمت چپ ماسک، نشان‌دهنده مطابقت آن با ناحیه تریگر در تصویر تریگر است.

زیربخش دوم – شناسایی برچسب مورد حمله قرار گرفته

انحراف مطلق میانه (MAD) یک آمار مقاوم است که برای شناسایی داده‌های پرت و خارج از محدوده در یک مجموعه داده استفاده می‌شود. در زمینه شناسایی برچسب‌های آلوده در شبکه‌های عصبی، روش MAD می‌تواند به شناسایی برچسبی که نیاز به تریگرهای کوچک‌تری برای ایجاد خطا در طبقه‌بندی دارد، کمک کند. این ویژگی به ما این امکان را می‌دهد که برچسب آلوده را شناسایی کنیم.

دلیل استفاده از MAD برای شناسایی برچسب آلوده :

مقاومت در برابر داده‌های پرت:

برخلاف میانگین و انحراف معیار، میانه و MAD به شدت تحت تأثیر داده‌های پرت قرار نمی‌گیرند. این ویژگی باعث می‌شود MAD برای شناسایی برچسب‌های آلوده که تعداد کمی از داده‌ها را تحت تأثیر قرار می‌دهند، بسیار مناسب باشد

تشخیص تریگرهای کوچک‌تر:

برچسب آلوده معمولاً نیاز به تریگرهای کوچکتری دارد تا خطای طبقه‌بندی را ایجاد کند. این تریگرهای کوچک‌تر می‌توانند به عنوان داده‌های پرت در مجموعه داده نرمال تریگرها شناسایی شوند.

سادگی و کارایی:

محاسبه MAD و استفاده از آن برای شناسایی داده‌های پرت بسیار ساده و کارآمد است. این روش نیاز به تنظیم پارامترهای پیچیده ندارد و به راحتی قابل پیاده‌سازی است.

توضیحات بخش کد نویسی

در این قسمت هدف ما شناسایی برچسب آلوده و بازسازی تریگر مربوط به آن در یک مدل کانولوشنی است. مدل مورد استفاده قبلاً با مجموعه داده MNIST آموزش داده شده و آلوده به تریگرهای بکدور شده است. برای شناسایی برچسب آلوده و بازسازی تریگر، از روش MAD (Median Absolute Deviation) استفاده می‌کنیم.

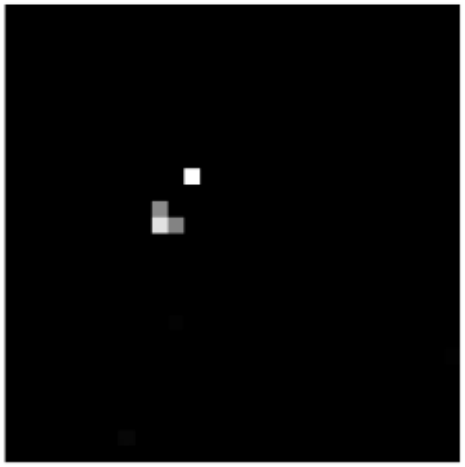
تابع reconstruct_trigger تریگر و ماسک مربوط به برچسب هدف را بازسازی میکند. برای این کار از یک بهینه‌ساز Adam با نرخ یادگیری 0.01 و پارامتر جریمه (lambda) λ برابر 0.5 استفاده شد

برای شناسایی داده‌های پرت (outliers) بر اساس روش MAD، تابع mad_based_outlier تعریف شد. این تابع داده‌های پرت را با مقایسه انحراف معیار مطلق با مقدار میانه شناسایی میکند

با استفاده از این روش مدل به درستی برچسب آلوده (لیبل شماره 9) را تشخیص میدهد و همچنین trigger آن هم رسم میکند

Trigger for Label 9

Mask for Label 9



بخش دوم – پاکسازی مدل و کاهش اثر حمله

در مقاله سه روش متفاوت برای کاهش اثر حملات بکدور معرفی شده است. این روش‌ها عبارتند از:

فیلتر ورودی‌ها (Input Filters):

توضیح: در این روش، فیلترهایی برای شناسایی و مسدود کردن ورودی‌های حاوی تریگرهای بکدور طراحی می‌شوند. این فیلترها با شناسایی الگوهای خاصی که در ورودی‌های مخرب وجود دارند، می‌توانند ورودی‌های حاوی تریگر را قبل از پردازش توسط مدل شناسایی و مسدود کنند.

دلیل استفاده: این روش به کاربران امکان می‌دهد تا ورودی‌های مخرب را قبل از اینکه به مدل ارسال شوند، شناسایی کنند و از اجرای آن‌ها جلوگیری نمایند، که می‌تواند به جلوگیری از حملات بکدور کمک کند.

هرس نرون‌ها (Neuron Pruning):

توضیح: در این روش، نرون‌های مرتبط با تریگرهای بکدور شناسایی و از شبکه عصبی حذف می‌شوند. این کار با کاهش فعال‌سازی نرون‌های مرتبط با تریگرها انجام می‌شود.

دلیل استفاده: با حذف نرون‌های مرتبط با تریگرهای بکدور، مدل قابلیت تشخیص تریگرهای بکدور را از دست می‌دهد و در نتیجه حملات بکدور بی‌اثر می‌شوند.

آموزش مجدد (Unlearning):

توضیح: در این روش، مدل با استفاده از داده‌های تمیز و بدون تریگر مجدداً آموزش داده می‌شود تا تریگرهای بکدور را از دست بدهد. این کار با اعمال تریگرهای بازیابی شده به داده‌های تمیز و آموزش مجدد مدل انجام می‌شود.

دلیل استفاده: با آموزش مجدد مدل با داده‌های تمیز، مدل از تریگرهای بکدور پاک می‌شود و دیگر به آن‌ها واکنش نشان نمی‌دهد.

در این قسمت با استفاده از روش Unlearning به پاکسازی مدل آلوده و ارزیابی آن بر اساس دقت و نرخ موفقیت حملات طراحی شده پرداخته ایم

خروجی کد اینگونه شده است

Cleaned Model Accuracy: 93.47%
Cleaned Model Attack Success Rate: 98.22%
Infected Model Accuracy: 77.51%
Infected Model Attack Success Rate: 37.73%

تحلیل :

دقت مدل‌ها

دقت مدل پاکسازی شده (Cleaned Model Accuracy): 93.47%

دقت مدل پاکسازی شده بسیار بالاست که نشان می‌دهد مدل پس از فرآیند unlearning به خوبی توانسته است وظایف اصلی خود را انجام دهد و برچسب‌های داده‌های تمیز را به درستی تشخیص دهد.

دقت مدل آلوده (Infected Model Accuracy): 77.51%

دقت مدل آلوده نسبتاً پایین‌تر است. این کاهش دقت به دلیل وجود تریگرهای بکدور در مدل است که باعث می‌شود مدل نتواند به درستی وظایف اصلی خود را انجام دهد و به اشتباهات بیشتری در تشخیص برچسب‌ها منجر شود

نرخ موفقیت حملات

نرخ موفقیت حملات مدل پاکسازی شده (Cleaned Model Attack Success Rate): 98.22%

این نرخ نشان می‌دهد که مدل پاکسازی شده هنوز به شدت تحت تأثیر تریگرهای بکدور قرار دارد. با وجود اینکه دقت کلی مدل بالاست، اما همچنان در حضور تریگرها دچار اشتباه می‌شود و به درستی نمی‌تواند تریگرها را نادیده بگیرد. این نشان‌دهنده این است که فرآیند unlearning به اندازه کافی موثر نبوده است.

نرخ موفقیت حملات مدل آلوده (Infected Model Attack Success Rate): 37.73%

این نرخ نشان می‌دهد که مدل آلوده به شدت تحت تأثیر تریگرهای بکدور قرار دارد. با این حال، نرخ موفقیت حملات کمتر از مدل پاکسازی شده است، که ممکن است به دلیل تغییرات تصادفی در داده‌ها یا اثرات جانبی دیگر باشد.

سوال دوم : PRIVACY

بخش اول

زیر بخش اول

برای محاسبه پارامتر b در توزیع لاپلاس برای هر درخواست، باید از فرمول زیر استفاده کنیم

$$b = \frac{\Delta f}{\epsilon}$$

که در فرمول بالا Δf حساسیت تابع مورد نظر است

برای درخواست شماره یک حساسیت 5000 دلار است بنابراین داریم

$$b_1 = \frac{5000}{0.1} = 50000$$

برای درخواست دوم حساسیت 50000 دلار است بنابراین داریم

$$b_2 = \frac{50000}{0.1} = 500000$$

زیر بخش دوم

با توجه به:

میانگین درآمد واقعی 40,000$ است.

کل درآمد واقعی 20,000,000$ است.

نویز نمونه‌برداری شده برای درخواست میانگین درآمد 2,000$ است.

نویز نمونه‌برداری شده برای درخواست کل درآمد 5,000$ است.

برای درخواست میانگین درآمد:

میانگین درآمد گزارش شده = میانگین درآمد واقعی+نویز نمونه‌برداری شده

میانگین درآمد گزارش شده = 40000+ 2000 = 42000

برای درخواست کل درآمد:

کل درآمد گزارش شده = کل درآمد واقعی + نویز نمونه‌برداری شده

کل درآمد گزارش شده = 20000000+5000 = 20005000

بنابراین

میانگین درآمد گزارش شده: $42,000

کل درآمد گزارش شده: $20,005,000

زیر بخش سوم

برای حفظ کل هزینه حریم خصوصی (overall privacy loss) کمتر از $\epsilon = 0.1$ و تخصیص دادن $\epsilon 1 = 0.05$ به درخواست مربوط به درآمد متوسط و$\epsilon 2 = 0.05$ به درخواست درآمد کل، باید تأثیر این تخصیص‌ها بر پارامتر مقیاس (*scale*) در توزیع لاپلاس و مقادیر گزارش شده را بررسی کنیم.

تأثیر بر پارامتر مقیاس در توزیع لاپلاس:

برای درخواست میانگین درآمد:

حساسیت $\Delta f = 5000$ دلار است

بنابراین :

$$b_1 = \frac{5000}{0.05} = 100000$$

برای درخواست کل درآمد:

حساسیت $\Delta f = 50000$ دلار است

بنابراین

$$b_2 = \frac{50000}{0.05} = 1000000$$

برای درخواست میانگین درآمد:

فرض کنیم نویز جدید نمونه‌برداری شده برای میانگین درآمد η_1 است

میانگین درآمد گزارش شده به صورت زیر خواهد بود

میانگین درآمد گزارش شده = میانگین درآمد واقعی + η_1

برای درخواست کل درآمد:

فرض کنیم نویز جدید نمونه‌برداری شده برای کل درآمد η_2 باشد

کل درآمد گزارش شده به صورت زیر خواهد بود:

کل درآمد گزارش شده = کل درآمد واقعی+ η_2

اثر نویز بر دقت مقادیر گزارش شده:

با توجه به اینکه پارامترهای مقیاس b_1 و b_2 افزایش یافته نویزهای η_1 و η_2 نمونه‌برداری شده نیز بزرگتر خواهند بود. این افزایش در نویز به معنای کاهش دقت مقادیر گزارش شده است، اما از طرف دیگر حریم خصوصی بیشتری فراهم می‌شود.

در نتیجه :

میانگین درآمد گزارش شده: میانگین درآمد واقعی به همراه نویز جدید که از توزیع لاپلاس با پارامتر مقیاس 100000 نمونه‌برداری شده است.

کل درآمد گزارش شده: کل درآمد واقعی به همراه نویز جدید که از توزیع لاپلاس با پارامتر مقیاس 1000000 نمونه برداری شده است

این تغییرات به معنای حفظ حریم خصوصی بهتر افراد در داده‌ها هستند، هرچند دقت نتایج کاهش می‌یابد.

بخش دوم :

زیر بخش اول

در مکانیزم لاپلاس استاندارد، ضریب مقیاس b به δ وابسته نیست بنابراین داریم

$$b = \frac{\Delta f}{\epsilon}$$

بنابراین داریم :

b = 1/0.1 = 10

زیر بخش دوم

برای محاسبه احتمال اینکه پاسخ نویزی بیش از 505 باشد، باید از تابع توزیع تجمعی (CDF) توزیع لاپلاس استفاده کنیم. تابع توزیع تجمعی توزیع لاپلاس به صورت زیر است:

$$\begin{cases} 0.5 \exp\left(\frac{x-\mu}{b}\right) & \text{if } x < \mu \\ 1 - 0.5 \exp\left(-\frac{x-\mu}{b}\right) & \text{if } x \geq \mu \end{cases}$$

که در اینجا

μ مقدار واقعی پاسخ است

b ضریب مقیاس توزیع لاپلاس است.

x مقداری است که می‌خواهیم احتمال آن را محاسبه کنیم.

با توجه به اطلاعات داده شده:

پاسخ دقیق (μ) برابر با 500 است

مقدار x برابر با 505 است

ضریب مقیاس (bbb) برابر با 10 است.

ما به دنبال ($P(X > 505)$) هستیم که می‌تواند به صورت زیر محاسبه شود:

$$P(X > 505) = 1 - F(505|b)$$

چون ($505 \geq \mu$)

$$F(505|b) = 1 - 0.5\exp\left(-\frac{505-500}{10}\right)$$

محاسبه می‌کنیم:

$$F(505|b) = 1 - 0.5\exp\left(-\frac{5}{10}\right)$$

$$F(505|b) = 1 - 0.5\exp(-0.5)$$

$$F(505|b) = 1 - 0.5 \times \exp(-0.5)$$

با استفاده از مقدار تقریبی $(\exp(-0.5) \approx 0.6065)$ داریم .

$$F(505|b) \approx 1 - 0.5 \times 0.6065$$

$$F(505|b) \approx 1 - 0.30325$$

$$F(505|b) \approx 0.69675$$

بنابراین احتمال $(P(X > 505))$ برابر است با :

$$P(X > 505) = 1 - F(505|b)$$

$$P(X > 505) \approx 1 - 0.69675$$

$$P(X > 505) \approx 0.30325$$

بنابراین، احتمال اینکه پاسخ نویزی بیش از 505 باشد تقریباً 0.30325 یا 30.325٪ است.

زیر بخش سوم :

با توجه به اینکه پارامترهای حریم خصوصی برای هر درخواست i به صورت $(\epsilon i, \delta i)$ تعریف شده است که

$$\epsilon_i = \frac{\epsilon}{k}$$

$$\delta_i = \frac{\delta}{k}$$

و با فرض اینکه حساسیت $\Delta f=1$ است، ضریب مقیاس b به صورت زیر محاسبه می‌شود:

$$b_i = \frac{\Delta f}{\epsilon_i} = \frac{\Delta f}{\frac{\epsilon}{k}} = \frac{\Delta f \cdot k}{\epsilon}$$

با توجه به اطلاعات داده شده داریم

$$\Delta f = 1$$

$$\epsilon = 0.1$$

k تعداد کل درخواست‌ها است.

بنابراین برای درخواست i:

$$b_i = \frac{1 \cdot k}{0.1} = \frac{k}{0.1} = 10k$$

محاسبه احتمال $(P(X > 505))$

با فرض اینکه پاسخ دقیق (μ) برابر با 500 است و مقدار x برابر با 505 است، ما به دنبال احتمال $(P(X > 505))$ هستیم. تابع توزیع تجمعی (CDF) توزیع لاپلاس به صورت زیر است:

$$\begin{cases} 0.5 \exp\left(\frac{x-\mu}{b}\right) & \text{if } x < \mu \\ 1 - 0.5 \exp\left(-\frac{x-\mu}{b}\right) & \text{if } x \geq \mu \end{cases}$$

برای x=505, μ=500 و $(b_i = 10k)$

$$F(505|b_i) = 1 - 0.5 \exp\left(-\frac{505 - 500}{b_i}\right)$$

$$F(505|b_i) = 1 - 0.5 \exp\left(-\frac{5i}{10}\right)$$

$$F(505|b_i) = 1 - 0.5 \exp(-0.5i)$$

برای احتمال $(P(X > 505))$

$$P(X > 505) = 1 - F(505|b_i)$$

$$P(X > 505) = 1 - (1 - 0.5 \exp(-0.5i))$$

$$P(X > 505) = 0.5 \exp(-0.5i)$$

بنابراین، احتمال اینکه پاسخ نویزی برای qi بزرگتر از 505 باشد برابر $0.5 \exp(-0.5i)$

زیر بخش چهارم

در سناریوی Unbounded Differential Privacy ، اگر درصد معینی p (به صورت اعشاری بیان شده) از کل جمعیت از پایگاه داده اضافه یا حذف شده باشد، حساسیت Δf به صورت زیر تغییر می‌کند:

$\Delta f = p \times n$

که در آن:

p درصد تغییر در جمعیت (به صورت اعشاری).

n تعداد کل افراد در پایگاه داده اولیه.

با فرض اینکه پاسخ دقیق (μ) برابر با 500 است و مقدار x برابر با 505 است، ما به دنبال احتمال ($P(X > 505)$) هستیم. تابع توزیع تجمعی (CDF) توزیع لاپلاس به صورت زیر است:

$$\begin{cases} 0.5 \exp\left(\frac{x-\mu}{b}\right) & \text{if } x < \mu \\ 1 - 0.5 \exp\left(-\frac{x-\mu}{b}\right) & \text{if } x \geq \mu \end{cases}$$

برای x=505, μ=500 و b=10×p×n

$$F(505|b) = 1 - 0.5 \exp\left(-\frac{505 - 500}{b}\right)$$

$$F(505|b) = 1 - 0.5 \exp\left(-\frac{5}{10 \times p \times n}\right)$$

برای احتمال ($P(X > 505)$) داریم :

$$P(X > 505) = 1 - F(505|b)$$

$$P(X > 505) = 1 - \left(1 - 0.5 \exp\left(-\frac{5}{10 \times p \times n}\right)\right)$$

$$P(X > 505) = 0.5 \exp\left(-\frac{0.5}{p \times n}\right)$$

بنابراین، احتمال اینکه پاسخ نویزی بیشتر از 505 باشد برابر با $0.5 \exp\left(-\frac{0.5}{p \times n}\right)$

سوال سوم : Fairness

بخش اول: دیتا و ارزیابی مدل

بارگذاری داده

در ابتدا فایل دیتاست را آپلود کرده ایم دیتاست شامل ویژگی‌های مختلفی از جمله سن، جنسیت، نژاد، ساعات کاری هفتگی، نوع شغل و درآمد است. هدف ما طراحی مدلی است که بتواند درآمد بالاتر یا پایین‌تر از 50 هزار دلار را بدون تبعیض جنسیتی پیش‌بینی کند.

بخش دوم: پیاده سازی مدل پایه

در این قسمت ابتدا داده ها را به دو قسمت آموزش و ارزیابی به نسبت 70 به 30 تقسیم کرده ایم و سپس برای آموزش مدل از طبقه بند لجستیک رگرسیون استفاده کرده ایم مدل را آموزش داده ایم و دقت مدل و همچنین معیارهای Zemel Fairness و Disparate Impact را اندازه گیری کرده ایم که به شرح زیر است

معیار	مقدار
دقت (Accuracy)	0.7976
Zemel Fairness	0.1218
Disparate Impact	0.2078

تحلیل نتایج

Accuracy (دقت):

مقدار دقت مدل برابر با 0.7975 یا تقریباً 80٪ است. این مقدار نشان‌دهنده درصد پیش‌بینی‌های صحیح مدل از کل پیش‌بینی‌ها است. به عبارت دیگر، مدل رگرسیون لجستیک در 80٪ مواقع درست پیش‌بینی کرده است که آیا درآمد افراد بالاتر از 50 هزار دلار است یا خیر. این مقدار نسبتاً خوب است و نشان می‌دهد که مدل به طور کلی عملکرد قابل قبولی دارد.

Zemel Fairness (عدالت Zemel):

مقدار عدالت Zemel برابر با 0.1218 است. این معیار نشان می‌دهد که تفاوت در احتمال پیش‌بینی درآمد بالای 50 هزار دلار بین دو جنسیت (مردان و زنان) چقدر است. مقدار مثبت 0.1218 نشان می‌دهد که مدل به طور متوسط احتمال بیشتری برای پیش‌بینی درآمد بالای 50 هزار دلار برای مردان نسبت به زنان دارد. این نشان می‌دهد که مدل تا حدودی به نفع مردان سوگیری دارد.

Disparate Impact (اثر نابرابر):

مقدار اثر نابرابر برابر با 0.2078 است. این معیار نسبت احتمال پیش‌بینی درآمد بالای 50 هزار دلار برای زنان به احتمال مشابه برای مردان را نشان می‌دهد. مقدار 0.2078 کمتر از 1 است و نشان می‌دهد که زنان با احتمال کمتری نسبت به مردان پیش‌بینی می‌شوند که درآمد بالایی داشته باشند. این نیز نشانه‌ای از سوگیری مدل به نفع مردان است.

سوالات

آیا مدل به خوبی توانسته است درآمد را پیشبینی نماید؟

با توجه به دقت مدل، می‌توان گفت که مدل به طور کلی عملکرد خوبی در پیش‌بینی درآمد دارد. دقت 80% نشان‌دهنده‌ی عملکرد قابل قبول مدل در تشخیص درآمد بالای 50 هزار دلار است.

آیا مدل برای زنان و مردان به صورت عادل عمل مینماید؟

مقدار Zemel Fairness برابر با 0.1218 است که نشان می‌دهد که احتمال پیش‌بینی درآمد بالای 50 هزار دلار برای مردان بیشتر از زنان است. ایده‌آل این است که مقدار Zemel Fairness نزدیک به 0 باشد تا نشان دهد مدل به طور مساوی برای هر دو جنسیت عمل می‌کند.

مقدار Disparate Impact برابر با 0.2078 است که نشان می‌دهد که مدل به نفع مردان سوگیری دارد و احتمال پیش‌بینی درآمد بالای 50 هزار دلار برای مردان بیشتر است. ایده‌آل این است که مقدار Disparate Impact نزدیک به 1 باشد تا نشان دهد مدل به طور مساوی برای هر دو جنسیت عمل می‌کند.

در نتیجه با توجه به معیارهای Zemel Fairness و Disparate Impact، مدل به صورت عادلانه بین زنان و مردان عمل نمی‌کند. این دو معیار نشان می‌دهند که مدل به نفع مردان سوگیری دارد:

به نظر شما، حذف ویژگی حساس از دیتاست میتواند در عادل کردن مدل موثر باشد؟

خیر . در بیشتر مواقع این روش جوابگو نیست . حتی با حذف ویژگی حساس، مدل می‌تواند از طریق همبستگی‌های غیرمستقیم میان ویژگی‌های دیگر و ویژگی حساس، همچنان میتواند سوگیری داشته باشد. به عنوان مثال، ویژگی‌های دیگری مانند نوع شغل، ساعات کاری، و غیره ممکن است به طور غیرمستقیم با جنسیت مرتبط باشند و سوگیری را حفظ کنند.

بخش سوم: پیاده سازی مدل عادل

در این بخش برای عادل کردن مدل ابتدا یک دیتاست جدید ساخته ایم و مدل را روی این دیتا ست جدید آموزش داده ایم برای این کار مراحل زیر را دنبال کرده ایم

پیش‌بینی‌ها و احتمالات خروجی مدل را محاسبه و به دیتاست اضافه می‌کنیم.

دیتاست را بر اساس ویژگی‌های مشخص شده به دو بخش CP و CD تقسیم کرده و آنها را مرتب می‌کنیم.

ردیف‌های اول هر بخش را با یکدیگر جابجا می‌کنیم.

دیتاست جدید را تشکیل داده و ستون‌های پیش‌بینی و احتمالات را حذف می‌کنیم.

دیتاست جدید را به دو بخش آموزشی (70٪) و تست (30٪) تقسیم می‌کنیم.

مدل جدید را با دیتای اصلاح شده آموزش می‌دهیم

پیش‌بینی‌ها را بر روی دیتاست جدید انجام می‌دهیم.

دقت مدل و معیارهای عدالت (Zemel Fairness و Disparate Impact) را محاسبه می‌کنیم.

مدل را با دیتاست جدیدی که ساخته ایم آموزش داده ایم که نتایج زیر حاصل شده است

معیار	مقدار
دقت جدید (New Accuracy)	0.7949
Zemel Fairness جدید	0.0372
Disparate Impact جدید	0.6229

تحلیل نتایج بالا

دقت (Accuracy):

دقت مدل جدید نشان می‌دهد که مدل تقریباً در 79.49٪ مواقع پیش‌بینی صحیح انجام می‌دهد. این مقدار نشان‌دهنده عملکرد خوب مدل در پیش‌بینی درآمد بالای 50 هزار دلار است و کاهش ناچیزی در دقت مشاهده می‌شود که قابل قبول است.

Zemel Fairness

مقدار Zemel Fairness به 0.0372 رسیده است. این مقدار بسیار نزدیک به صفر نشان‌دهنده این است که مدل تقریباً به طور عادلانه بین مردان و زنان عمل می‌کند و سوگیری نسبت به جنسیت بسیار کاهش یافته است. کاهش مقدار Zemel Fairness نشان‌دهنده کاهش سوگیری مدل نسبت به جنسیت است.

Disparate Impact

مقدار Disparate Impact برابر با 0.6229 است که نسبت به مقدار 1 نزدیک‌تر شده است. این مقدار نشان می‌دهد که احتمال پیش‌بینی درآمد بالای 50 هزار دلار برای زنان به احتمال مشابه برای مردان نزدیک‌تر شده است. مقدار 0.6229 نشان‌دهنده این است که مدل کمتر به نفع مردان سوگیری دارد و توزیع درآمد بالای 50 هزار دلار بین زنان و مردان عادلانه‌تر شده است.

بخش چهارم: مقایسه و نتیجه گیری

مقایسه معیار ها بین دو روش

دقت (Accuracy):

دقت قبلی: 0.7976

دقت جدید: 0.7949

دقت مدل کمی کاهش یافته است (تقریباً 0.27 درصد). این مقدار کاهش نسبتاً ناچیز است و نشان می‌دهد که مدل جدید همچنان به خوبی پیش‌بینی می‌کند. این کاهش ممکن است ناشی از تغییراتی باشد که در دیتاست برای کاهش سوگیری انجام شده است.

Zemel Fairness:

Zemel Fairness قبلی: 0.1218

Zemel Fairness جدید: 0.0372

مقدار Zemel Fairness به طور قابل توجهی کاهش یافته است. این نشان می‌دهد که مدل جدید به طور عادلانه‌تری بین مردان و زنان عمل می‌کند. مقدار کمتر Zemel Fairness (نزدیک به 0) نشان‌دهنده کاهش سوگیری مدل نسبت به جنسیت است.

Disparate Impact:

Disparate Impact قبلی: 0.2078

Disparate Impact جدید: 0.6229

مقدار Disparate Impact به طور قابل توجهی افزایش یافته است. مقدار Disparate Impact قبلی بسیار کمتر از 1 بود که نشان‌دهنده سوگیری شدید به نفع مردان بود. مقدار جدید 0.6229 به مقدار 1 نزدیک‌تر شده است که نشان‌دهنده کاهش سوگیری و افزایش عدالت مدل بین مردان و زنان است.

معیار	مدل قبلی	مدل جدید
دقت (Accuracy)	0.7976	0.7949
Zemel Fairness	0.1218	0.0372
Disparate Impact	0.2078	0.6229

سوالات

کدام مدل از دقت بالاتری برخوردار است؟

مدل اول از دقت بالاتری برخوردار است دقت مدل قبلی 0.7976 و دقت مدل جدید 0.7949 حدودا 0.27 درصد کاهش دقت داشته ایم البته این مقدار کاهش نسبتاً ناچیز است و نشان میدهد که مدل جدید همچنان به خوبی پیش‌بینی می‌کند.

کدام مدل عادل می باشد؟

مدل جدید عادل میباشد. مقدار Zemel Fairness مدل قبلی 0.1218 است و برای مدل جدید 0.0372 است که نشان می‌دهد که مدل جدید به طور عادلانه‌تری بین مردان و زنان عمل می‌کند. همچنین مقدار Disparate Impact مدل قبلی برابر 0.2078 و برای مدل جدبد برابر 0.6229 است که نشان‌دهنده کاهش سوگیری و افزایش عدالت مدل بین مردان و زنان است

آیا ارتباطی بین دقت و عادل بودن مدل مشاهده مینمایید؟ توضیح دهید

بله، ارتباطی بین دقت و عادل بودن مدل وجود دارد و این رابطه معمولاً به عنوان یک تبادل (trade-off) شناخته می‌شود. در بسیاری از موارد، بهبود عدالت مدل می‌تواند منجر به کاهش دقت شود و بالعکس

در این پروژه دقت مدل جدید کمی کاهش یافته است (از 0.7976 به 0.7949). این کاهش نشان‌دهنده این است که در تلاش برای بهبود عدالت مدل، مقداری از دقت قربانی شده است

یک روش دیگر برای عادل کردن طبقه بند معرفی کنید و تحلیل نمایید چرا این روش را معرفی نموده اید و به چه علت آن را موثر میدانید.

یکی از روش‌های موثر برای عادل کردن مدل‌های طبقه‌بندی، استفاده از روش Reweighing است

روش Reweighing یکی از روش‌ها برای کاهش سوگیری در مدل‌های یادگیری ماشین است. این روش با تغییر وزن نمونه‌ها در مجموعه داده، توزیع داده‌ها را به گونه‌ای تنظیم می‌کند که سوگیری کمتری داشته باشد. ایده اصلی این است که نمونه‌هایی که متعلق به گروه‌های کمتر نمایان هستند، وزن بیشتری بگیرند تا مدل توجه بیشتری به آن‌ها داشته باشد.

مراحل روش Reweighing

تعریف ویژگی‌های حساس و هدف :

شناسایی ویژگی‌های حساس (مانند جنسیت، نژاد و غیره) و متغیر هدف (مانند درآمد بالای 50 هزار دلار) در مجموعه داده‌ها.

محاسبه توزیع‌ها:

محاسبه توزیع کلی متغیر هدف ((P(y)).

محاسبه توزیع کلی ویژگی حساس ((P(s)).

محاسبه توزیع مشترک متغیر هدف و ویژگی حساس ((P(y,s)).

محاسبه وزن‌ها:

محاسبه وزن هر نمونه بر اساس توزیع‌های محاسبه شده. وزن هر نمونه به صورت زیر محاسبه می‌شود:

$$w_i = \frac{P(y_i, s_i)}{P(y_i) \cdot P(s_i)}$$

این وزن‌ها به گونه‌ای تنظیم می‌شوند که نمونه‌های متعلق به گروه‌های کمتر نمایان وزن بیشتری بگیرند.

اعمال وزن‌ها به مجموعه داده:

اعمال وزن‌های محاسبه شده به نمونه‌های موجود در مجموعه داده. در این مرحله، هر نمونه دارای یک وزن خاص خواهد بود.

آموزش مدل با استفاده از وزن‌ها:

آموزش مدل یادگیری ماشین با استفاده از مجموعه داده وزن‌دهی شده. در این مرحله، مدل با در نظر گرفتن وزن‌های نمونه‌ها آموزش می‌بیند.

دلایل موثر بودن روش Reweighting:

کاهش سوگیری:

روش Reweighting به طور مستقیم به کاهش سوگیری در داده‌ها کمک می‌کند. با تنظیم وزن نمونه‌ها بر اساس توزیع ویژگی حساس و برچسب هدف، مدل یادگیری توجه بیشتری به گروه‌های کمتر نمایان دارد. این باعث می‌شود که مدل کمتر به نفع گروه‌های غالب سوگیری داشته باشد و عدالت بیشتری در پیش‌بینی‌ها ایجاد شود.

حفظ دقت:

این روش می‌تواند دقت مدل را حفظ کند یا حتی بهبود بخشد. با تنظیم وزن‌ها به گونه‌ای که توزیع داده‌ها تعادل بیشتری داشته باشد، مدل یادگیری می‌تواند از همه داده‌ها به طور موثرتری استفاده کند. این باعث می‌شود که مدل بتواند به خوبی از اطلاعات موجود استفاده کند و دقت بالایی داشته باشد.

سادگی پیاده‌سازی:

روش Reweighting نسبتاً ساده و مستقیم است و نیاز به تغییرات زیادی در مدل ندارد. تنها با تنظیم وزن نمونه‌ها می‌توان به نتایج بهتری دست یافت. این روش نیاز به تغییرات پیچیده در ساختار مدل یا داده‌ها ندارد و به راحتی قابل پیاده‌سازی است.

انعطاف‌پذیری:

این روش انعطاف‌پذیری بالایی دارد و می‌تواند به راحتی با انواع مختلف مدل‌ها و داده‌ها سازگار شود. می‌توان از این روش برای کاهش سوگیری در مدل‌های مختلف یادگیری ماشین استفاده کرد و بهبود قابل توجهی در عدالت و دقت مدل‌ها ایجاد کرد.

بخش پنجم: امتیازی

روش Reweighting را که در بخش قبلی معرفی کردیم روی دیتاست اجرا کردیم که نتایج زیر حاصل شد

Metric	Value
Accuracy	0.8035
Zemel Fairness	0.0519
Disparate Impact	0.5386

تحلیل نتایج و مقایسه با روش های قبلی

دقت (Accuracy):

مقدار: 0.8035

تحلیل: دقت مدل به 80.35٪ رسیده است که نشان‌دهنده بهبود نسبت به مدل‌های قبلی است. این مقدار دقت بالایی است و نشان می‌دهد که مدل توانسته است با وزن‌دهی به داده‌ها، عملکرد خوبی در پیش‌بینی درآمد بالای 50 هزار دلار داشته باشد.

Zemel Fairness:

مقدار: 0.0519

تحلیل: مقدار Zemel Fairness به 0.0519 رسیده است که بسیار نزدیک به صفر است. این مقدار نشان می‌دهد که مدل جدید تقریباً به طور عادلانه بین مردان و زنان عمل می‌کند و سوگیری نسبت به جنسیت کاهش یافته است. مقدار کمتر Zemel Fairness به معنای کاهش تفاوت در احتمال پیش‌بینی درآمد بالای 50 هزار دلار بین مردان و زنان است.

Disparate Impact:

مقدار: 0.5386

تحلیل: مقدار Disparate Impact به 0.5386 رسیده است. این مقدار نشان می‌دهد که مدل هنوز به طور کامل عادلانه بین مردان و زنان عمل نمی‌کند و احتمال پیش‌بینی درآمد بالای 50 هزار دلار برای زنان نسبت به مردان کمتر است. با این حال، این مقدار نسبت به مقادیر بسیار کم قبلی بهبود یافته است و نشان‌دهنده تلاش موفقیت‌آمیز برای کاهش سوگیری است

مقایسه

Metric	Reweighting Method	Initial Logistic Regression	Adjusted Logistic Regression
Accuracy	0.8035	0.7976	0.7949
Zemel Fairness	0.0519	0.1218	0.0372
Disparate Impact	0.5386	0.2078	0.6229

روش Reweighting بالاترین دقت را با مقدار 0.8035 به دست آورد که نشان‌دهنده عملکرد بهتر در پیش‌بینی‌ها در مقایسه با هر دو مدل اولیه و تنظیم شده رگرسیون لجستیک است.

از نظر Zemel Fairness، مدل تنظیم شده رگرسیون لجستیک کمترین مقدار را با 0.0372 به دست آورد که نشان‌دهنده کمترین سوگیری است. روش Reweighting نیز عملکرد خوبی داشته و با مقدار 0.0519 سوگیری نسبت به جنسیت را به طور قابل توجهی کاهش داده است در مقایسه با مدل اولیه که مقدار 0.1218 را داشت.

مدل تنظیم شده رگرسیون لجستیک بالاترین مقدار Disparate Impact را با 0.6229 داشت که نشان‌دهنده نزدیکی بیشتر به عدالت بین جنسیت‌ها است. روش Reweighting نیز با مقدار 0.5386 بهبود قابل توجهی نسبت به مدل اولیه که مقدار 0.2078 را داشت، نشان داد.