به نام خدا

پروژه سوم درس یادگیری عمیق

آشنایی با مفاهیم یادگیری بدون نظارت در شبکه‌های عصبی و شبکه‌های خود-سازمانده

استاد درس: دکتر صفابخش

نگارش: زهرا اخلاقی

شماره دانشجویی: ۴۰۱۱۳۱۰۶۴

پاییز ۱۴۰۲

فهرست مطالب

بخش‌اول 2
۱) 2
۲) 3
۳) 4
۴) 5
بخش‌دوم 6
۱) 6
۲) 6
۳) 6
۴) 6

بخش اول

۱)

هدف شبکه عصبی مصنوعی برای یادگیری بدون نظارت کشف الگوها و ساختارها در داده‌های بدون‌برچسب می‌باشد. نمونه‌هایی از این شبکه‌ها به شرح زیر میباشد:

- Kohonen Self-Organizing Maps: از یادگیری بدون نظارت برای تولید نمایش دو بعدی داده های با ابعاد بالا استفاده می کنند. در این مدل ورودی مدل به تمام نورون‌ها متصل است. هر کدام از این اتصالات دارای وزن می‌باشد و در فرآیند یادگیری مقدار این وزن‌ها تغییر پیدا می‌کند به طوری که خروجی به مقدار ورودی نزدیک‌تر گردد. پس از یادگیری، شبکه بدست آمده اطلاعات پراکندگی داده‌های استفاده شده در آموزش شبکه را مدل می‌کند.
- Self-Organizing Maps: این شبکه‌ها از یادگیری بدون نظارت برای تولید نمایش گسسته و کاهش بعد فضای ورودی استفاده می کنند و اغلب برای خوشه بندی و تجسم استفاده می شوند.
- Autoencoders: از یک encoder و یک decoder تشکیل شده است که سعی در یادگیری نمایش فشرده داده های ورودی دارد.
- Generative Adversarial Networks: شامل یک generator و یک discriminator هستند. هدف generator تولید داده های واقعی است، در حالی که هدف discriminator بین داده های واقعی و تولید شده تمایز قائل شود.
- Hebbian Learnin: یک قانون یادگیری بدون نظارت الهام گرفته از نوروبیولوژیک است. بیان می کند که اگر دو نورون به طور همزمان فعال باشند، ارتباط بین آنها تقویت می شود.
- Restricted Boltzmann Machines: مدل‌هایی با ساختار دوبخشی هستند که از لایه‌های visible و hidden تشکیل شده‌اند و برای یادگیری توزیع احتمال بر روی مجموعه ورودی آموزش دیده اند.
- Sparse Coding Models: هدف این مدل‌ها یافتن نمایشی پراکنده از داده‌ها است که برای یادگیری ویژگی و کاهش ابعاد مفید باشد.
- Neural Gas: یک الگوریتم یادگیری رقابتی است که با ساختار فضای ورودی سازگار است.
- Temporal Autoencoder: به طور خاص برای داده های متوالی طراحی شده اند، با هدف یادگیری وابستگی های زمانی و ثبت الگوها در داده های سری زمانی هستند.

۲)

برخی از معیارهای ارزیابی در یادگیری بدون نظارت به شرح زیر است:

- خوشه‌بندی: معیارهایی مانند شاهد ترکیب‌پذیری (Silhouette Score) یا اندازه خوشه‌ها (Cluster Size) می‌توانند به خوبی عملکرد یک شبکه عصبی در خوشه‌بندی داده‌های بدون برچسب را ارزیابی کنند.
- تراکم میان‌خوشه‌ای: معیارهایی که تراکم داده‌ها درون هر خوشه را اندازه‌گیری می‌کنند، مانند میانگین فاصله میان نقاط در یک خوشه، به ارزیابی دقت خوشه‌بندی کمک می‌کنند.
- تنوع بین‌خوشه‌ای: این معیاره تفاوت‌ها و تنوع بین خوشه‌ها را اندازه‌گیری می‌کنند. برخی از معیارها می‌توانند مبتنی بر فاصله‌های بین خوشه‌ها باشند.
- ارزیابی مولفه‌های اصلی: می‌توان اطلاعات ویژگی‌های مهم در داده‌های ورودی را بررسی کرد برای اینکه شبکه ویژگی‌های مهم را یاد بگیرد.
- نمایش داده‌ها: نمایش داده‌ها و ویژگی‌ها در فضای کاهش بعد اهمیت دارد. تصاویر یا نمودارهایی از داده‌ها و ویژگی‌ها می‌توانند نشان‌دهنده یادگیری مفهومی و ساختاری درونی شبکه باشند.
- استفاده از اطلاعات برچسب‌خورده: در صورت دسترسی به برچسب‌ها، می‌توان از معیارهای مرسوم یادگیری نظارتی مانند دقت (Accuracy)، بازخوانی (Recall) و دقت پیش‌بینی (Precision) استفاده کرد.
- یادگیری معنایی: این معیار توانایی شبکه در یادگیری و درک مفهومی اطلاعات را اندازه‌گیری کند.
- نرخ همگرایی: سرعت و کیفیت همگرایی شبکه نیز یک معیار مهم است که می‌تواند از جنبه‌های زمانی ارزیابی شود.

۳)

U-matrix یک ابزار تجسمی است که معمولاً درSOM یا شبکه‌های کوهونن استفاده می شود. SOM ها نوعی شبکه عصبی مصنوعی هستند که برای خوشه بندی و تجسم داده های با ابعاد بالا در نمایش های با ابعاد پایین استفاده می شوند. ماتریس U به درک روابط توپولوژیکی بین نورون ها در SOM کمک می کند. در یک SOM، نورون ها در یک شبکه مرتب شده اند و هر نورون نشان دهنده یک نمونه اولیه خوشه است. در طول فرآیند آموزش، نورون های همسایه نورون برنده به گونه ای تنظیم می شوند که بیشتر شبیه به داده های ورودی باشند. U-matrix به صورت بصری نشان دهنده فواصل یا عدم شباهت بین نورون های همسایه است.U-matrix معمولاً به صورت شبکه ای از رنگ ها نمایش داده می شود که رنگ هر سلول نشان دهنده فاصله یا عدم شباهت بین نورون های همسایه است. با بررسی U-matrix می‌توان خوشه ها و

مرزهای آنها را شناسایی کرد. U-matrix یک راه بصری برای کشف روابط بین خوشه ها و درک ساختار داده ها در فضای کم بعدی SOM ارائه می دهد.

با توجه به اهداف و زمینه تحلیل میتوان جایگزین‌های زیر را برای U-matrix در نظر گرفت:

- Component Planes: صفحات مؤلفه، مقادیر بردارهای codebook را در ابعاد مختلف نشان می دهند. هر صفحه در مورد نحوه نمایش فضای ورودی در امتداد یک ویژگی خاص ارائه می دهد و تجزیه و تحلیل سطوح می تواند در درک هر ویژگی در SOM کمک کند.
- Heatmaps: برای تجسم بردارهای وزن یا فواصل بین نورون ها استفاده می‌شوند و نمایشی بصری از شباهت ها یا عدم شباهت ها در یک ماتریس ارائه می دهند.
- Silhouette Analysis: معیاری است که میزان شباهت یک شی را به خوشه خود (انسجام) در مقایسه با سایر خوشه ها (جداسازی) می سنجد. می توان از آن برای ارزیابی کیفیت خوشه بندی در SOM استفاده کرد.
- t-Distributed Stochastic Neighbor Embedding: یک تکنیک کاهش ابعاد است که اغلب برای تجسم داده های با ابعاد بالا در ابعاد پایین تر استفاده می شود. در حالی که جایگزینی مستقیم نیست، اما می‌تواند در نشان دادن توزیع داده‌ها کمک کند.
- Cluster Visualization: برای تجزیه و تحلیل خوشه بندی طراحی شده اند که می توانند مرزها و روابط خوشه ها را تجسم کنند. این ابزارها ممکن است شامل dendrograms، cluster heatmaps یا سایر روش های متناسب با داده های شما باشد.

(۴

در شبکه های خودسازمانده، نورون ها برای فعال شدن بر اساس داده های ورودی رقابت می کنند. نورون برنده، نورونی است که به بهترین شکل یک الگوی ورودی خاص را نشان می دهد. در SOM، نورون‌ها به شیوه‌ای رقابتی سازمان‌دهی می‌شوند که در آن برای پاسخ به الگوهای ورودی خاص رقابت می‌کنند و در فرآیند یادگیری، نورونی که به بهترین وجه با داده های ورودی مطابقت دارد، در رقابت برنده می شود. در طول فرآیند یادگیری، وزن‌های مرتبط با نورون برنده و همسایگان آن تنظیم می‌شوند تا پاسخ‌دهی آنها به الگوهای ورودی مشابه افزایش یابد. این فرآیند به شبکه کمک می کند تا با توزیع زیربنایی داده های ورودی سازگار شود.

توزیع داده های ورودی نقش مهمی در شکل دادن به سازمان شبکه ایفا می کند. الگوهایی که بیشتر در داده ها رخ می دهند، احتمالاً بر یادگیری و سازماندهی شبکه تأثیر می گذارند و یادگیری را هدایت می کنند. مکانیسم یادگیری رقابتی به این شبکه‌ها کمک می‌کند تا با ویژگی‌های آماری داده‌های ورودی سازگار شوند و در طول زمان، نمایش‌های معناداری را شکل دهند. شبکه‌های خودسازمان‌دهی، توانایی تطبیق با ساختار و توزیع داده‌ها را دارند. با این حال، عملکرد آنها در داده های نامتعادل به عوامل مختلفی بستگی دارد و چالش ها و راه حل های بالقوه ای وجود دارد که باید در نظر گرفته شوند. برای

مثال، در مجموعه داده‌های نامتعادل، جایی که یک کلاس در مقایسه با سایرین کمتر ارائه می‌شود، شبکه خودسازمان‌ده ممکن است نمایش‌های مغرضانه‌ای را به نفع طبقه اکثریت ایجاد کند. طبقه اکثریت ممکن است بر فرآیند یادگیری تسلط داشته باشد، و گرفتن الگوهای مناسب در کلاس اقلیت را برای شبکه چالش برانگیز می کند.

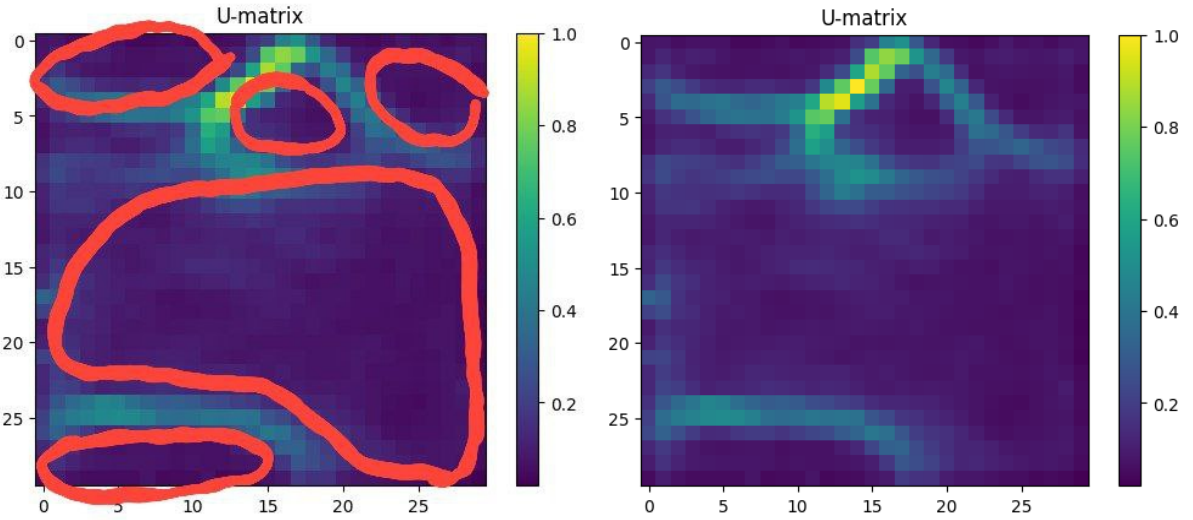
تغییر نرخ یادگیری برای کلاس های مختلف می تواند به تعادل تأثیر هر کلاس در طول آموزش کمک کند. نرخ‌های یادگیری بالاتر به کلاس‌های کم‌نمایش می‌تواند شبکه را تشویق کند تا به الگوهای اقلیت توجه بیشتری داشته باشد. نمونه برداری بیش از حد از کلاس اقلیت یا کم نمونه برداری از کلاس اکثریت را می توان قبل از آموزش برای متعادل کردن توزیع کلاس اعمال کرد به شبکه کمک می‌کند تا در طول آموزش با مجموعه‌ای از مثال‌های متعادل‌تر مواجه شود. معرفي هزينه‌هاي خاص طبقات در طي آموزش مي‌تواند با جريمه كردن طبقه‌بندي نادرست طبقات اقليت به شدت، عدم تعادل را برطرف كند. انتخاب معیارهای ارزیابی مناسب، مانند دقت، یادآوری، و امتیاز F1، می تواند درک دقیق تری از عملکرد شبکه، به ویژه در زمینه مجموعه داده های نامتعادل، ارائه دهد.

بخش‌دوم

در f1_score گزارش شده، جایگشت‌های متفاوت خوشه‌ها را محاسبه کردم و بالاترین f1_score به عنوان نتیجه نهایی گزارش شده.

۱)

IDS2 Dataset



در شکل بالا u-matrix و تعیین خوشه‌ها براساس آن نشان‌داده شده است. الگوریتم به‌ازای پارامترهای زیر اجرا شده و بهترین پارامتر برای مدل‌نهایی استفاده شده است:

```
learning_rate = [0.01,0.1,0.3,0.5,0.7,0.9]  # Initial learning rate
sigma = [1,3,5,10]  # Initial neighborhood radius
```

پارامتر بهینه:

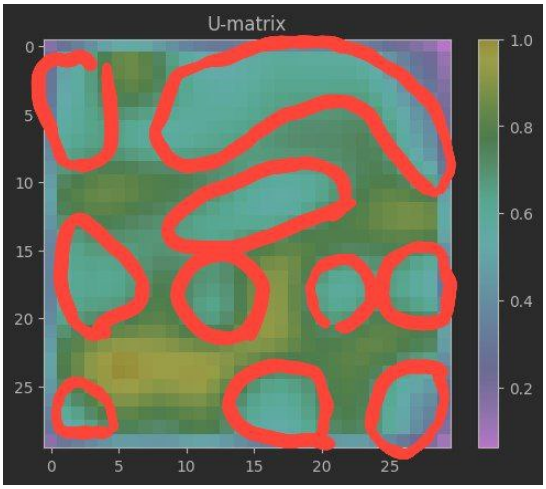
lr=0.7, sigma=1 => Normalized Mutual Information (NMI): 0.8406654057790216

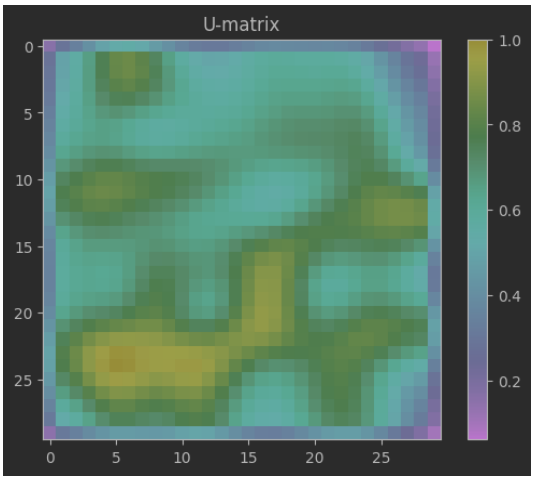
نتیجه‌نهایی به ازای پارامترهای بالا:

F1-Sore: 0.8650104615477623

Normalized Mutual Information (NMI): 0.7697875189865503

USPS Dataset





در شکل بالا u-matrix و تعیین خوشه‌ها براساس آن نشان‌داده شده است. الگوریتم به‌ازای پارامترهای زیر اجرا شده و بهترین پارامتر برای مدل‌نهایی استفاده شده است:

```
learning_rate = [0.01,0.1,0.3,0.5,0.7,0.9]  # Initial learning rate
sigma = [1,3,5,10]  # Initial neighborhood radius
```

پارامتر بهینه:

lr=0.7, sigma=1 => Normalized Mutual Information (NMI): 0.5742803478115412

نتیجه نهایی:

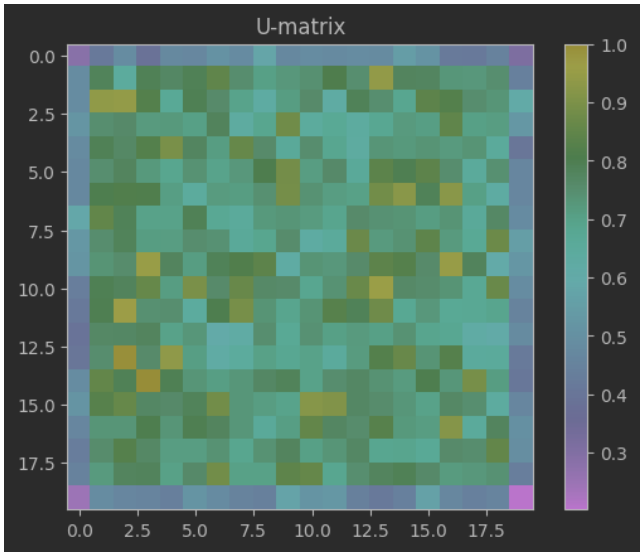
F1-Sore: 0.3992170156055018

Normalized Mutual Information (NMI): 0.5742803478115412

در این قسمت به علت زیاد بودن جایگشت خوشه‌ها از multiprocessing استفاده شده است.

GLI_85 Dataset

U-matrix برای این دیتاست به صورت زیر می باشد که از شکل آن نمی توان متوجه تعداد کلاستر ها شد بنابراین در این دیتاست تعداد کلاستر یکی از پارامترهایی است که با آزمون و خطا مشخص شده است:



پارامترهای زیر برای آزمون و خطا در نظر گرفته شده‌اند:

```
som_shapes = [(1,2),(1,3),(1,4),(1,5)]  # Grid size of the SOM
learning_rate = [0.01,0.1,0.3,0.5,0.7,0.9]  # Initial learning rate
sigma = [1,3,5,10]  # Initial neighborhood radius
```

پارامترهای بهینه:

shape =(1, 3), lr=0.7, sigma=10 => NMI: 0.3304478018679809

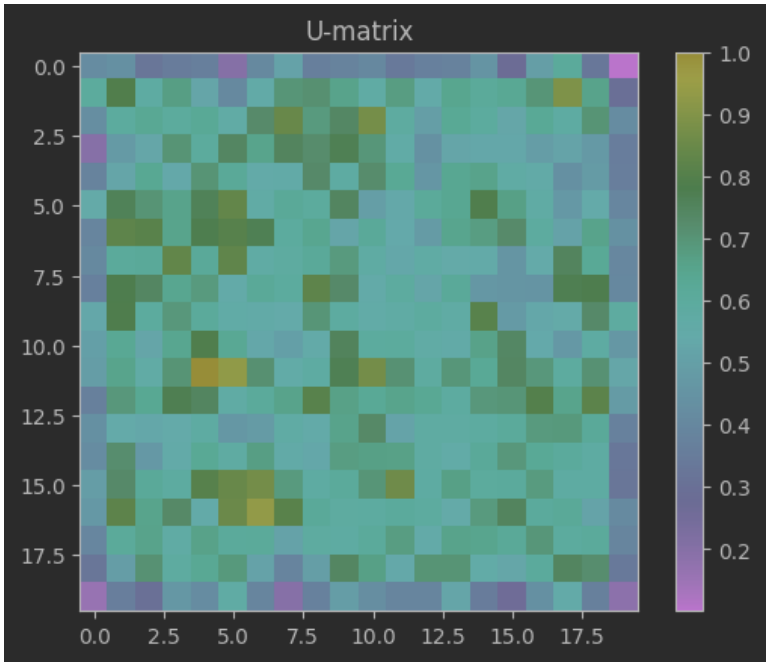
نتیجه نهایی به ازای پارامترهای بالا:

F1-Sore: 0.8018648018648018

Normalized Mutual Information (NMI): 0.379480591813931

Nci9 Dataset

U-matrix برای این دیتاست به صورت زیر می باشد که از شکل آن نمی توان متوجه تعداد کلاستر ها شد بنابراین در این دیتاست تعداد کلاستر یکی از پارامترهایی است که با آزمون و خطا مشخص شده است:



پارامترهای زیر برای آزمون و خطا در نظر گرفته شده‌اند:

```
som_shapes = [(1,4),(1,5),(1,6),(1,7),(1,8),(1,9),(1,10)]   # Grid size of the SOM
learning_rate = [0.01,0.1,0.3,0.5,0.7,0.9]  # Initial learning rate
sigma = [1,3,5,10]  # Initial neighborhood radius
```

پارامتر بهینه:

shape =(1, 10), lr=0.1, sigma=1 =>NMI: 0.4883870359560601

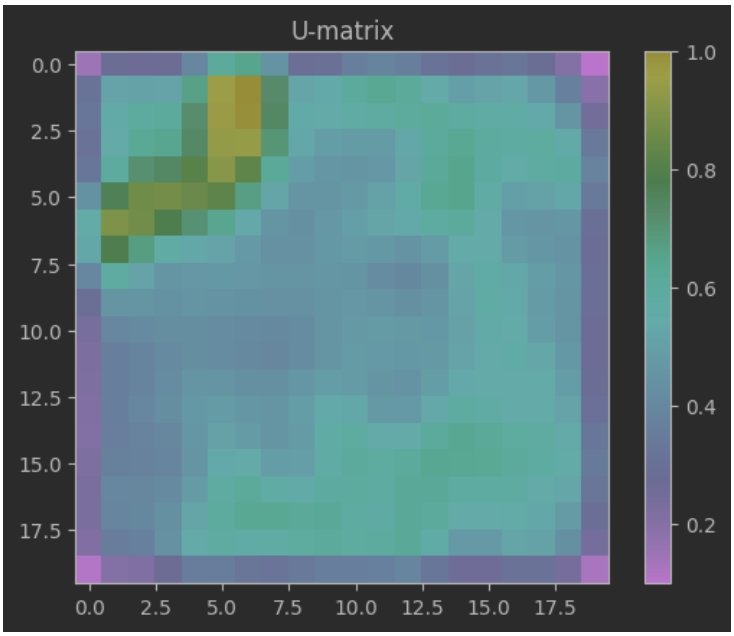
نتیجه نهایی:

F1-Sore: 0.3211033411033411

Normalized Mutual Information (NMI): 0.4883870359560601

Lung Dataset

U-matrix این دیتاست به صورت زیر می‌باشد:



در دیتاست بالا تنها یک خوشه به صورت واضح مشخص شده که به علت توزیع داده ها میباشد (داده‌ها دارای توزیع نامتوازن می‌باشد). پارامترهای زیر برای آزمون و خطا در نظر گرفته شده‌اند:

```
som_shapes = [(1,2),(1,3),(1,4),(1,5)]  # Grid size of the SOM
learning_rate = [0.01,0.1,0.3,0.5,0.7,0.9]  # Initial learning rate
sigma = [1,3,5,10]  # Initial neighborhood radius
```

پارامتر بهینه به صورت زیر میباشد:

shape =(1, 5), lr=0.5, sigma=1 => NMI: 0.654084378381474

نتیجه نهایی:

F1-Sore: 0.754724960625328

Normalized Mutual Information (NMI): 0.6565249341037842

(۲

در دیتاست این سوال، داده‌ها به صورت نامتوازن توزیع شده‌اند و نتیجه اجرا مدل به صورت زیر میباشد:

IDS2 Dataset

```
som_shape = (1, 5)
learning_rate = 0.7  # Initial learning rate
sigma = 1.0
```

F1-Sore: 0.5896439102495357

Normalized Mutual Information (NMI): 0.6242769031112974

lung Dataset

```
# Set the SOM parameters (you can adjust these values)
som_shape = (1, 5)
learning_rate = 0.5  # Initial learning rate
sigma = 1.0  # Initial neighborhood radius
```

F1-Sore: 0.5727725650803792

Normalized Mutual Information (NMI): 0.47873731800459884

نتیجه مقایسه با قسمت قبل:

پارامتر های learning_rate, sigma, som_shape برای هر دو شبکه مشابه یکدیگر است.

نتایج بخش ۱:

	f1_score	nmi
lung	0.754724960625328	0.6565249341037842
ids2	0.8650104615477623	0.7697875189865503

نتایج بخش ۲:

	f1_score	nmi
lung	0.5727725650803792	0.47873731800459884
ids2	0.5896439102495357	0.6242769031112974

توپولوژی hexagonal برای کلاس های اقلیت به خوبی آموزش نمی بیند و نتیجه بدتری را ارائه میدهد و چیدمان هندسی ممکن است فضای کافی را برای به تصویر کشیدن پیچیدگی‌های طبقات کمتر تخصیص ندهد، که منجر به نمایشی مغرضانه می‌شود و به طور کلی، داده‌های نامتوازن، دارای نتایج بدتری هستند. یکی از دلایل آن، تعداد ناکافی نمونه‌ها از دسته‌های کمتر است. این تعداد کمتر ممکن است باعث عدم توانایی شبکه در یادگیری مفاهیم مرتبط با دسته‌های کمتر شود. همچنین، نقاط پرت و نویز در دسته‌های کمتر ممکن است تأثیرات بیشتری را در معیارهای یادگیری ایجاد کنند، که باعث افت نتایج می‌شود (نتایج ارائه شده با توضیحات بخش قبلی مطابقت دارند).

برای بهبود نتایج در داده‌های نامتوازن، می‌توان از روش‌های متعادل‌سازی داده، افزایش تعداد نمونه‌ها در دسته‌های کمتر استفاده کرد. این اقدامات ممکن است شبکه را قادر به بهترین یادگیری و نمایش داده‌های نامتوازن نمایند.

۳)

lung Dataset

شبکه استفاده شده برای این دیتاست دارای چهار لایه می‌باشد و لایه مخفی به ترتیب دارای 256,128,64 نورون است و از بهینه ساز adam با نرخ یادگیری ۰.۰۱ استفاده شده است. Loss با آموزش این مدل تغییری نمیکند زیرا به علت بزرگ بودن تعداد آرگومان ورودی نمیتواند شبکه به خوبی آموزش ببیند و f1_score برای این شبکه ۰ است.

یکی از مشکلات این دیتاست توزیع نامتعادل داده است که برای پیش پردازش oversampling استفاده کردم ولی تغییری در خروجی حاصل نشد، زیرا مشکل این داده برای آموزش تعداد زیاد آرگومان ورودی است و ابتدا باید داده ورودی به فضای کوچک تری نگاشته شود و سپس از MLP استفاده شود.

IDS2 Dataset

شبکه استفاده شده برای این دیتاست دارای چهار لایه می‌باشد و لایه مخفی به ترتیب دارای 256,128,64 نورون است و از بهینه ساز adam با نرخ یادگیری ۰.۰۱ استفاده شده است. نتایج برای این شبکه به صورت زیر است:

=> F1_score : 0.33

یکی از مشکلات این دیتاست توزیع نامتعادل داده است که برای پیش پردازش oversampling استفاده کردم و نتایج به صورت زیر است:

=> F1_score: 0.99

(۴

lung Dataset

شبکه استفاده شده برای این دیتاست مشابه سوال قبل است (مشابه سوال دارای چهار لایه می‌باشد و لایه مخفی به ترتیب دارای 256,128,64 نورون است و از بهینه ساز adam با نرخ یادگیری ۰.۰۱) و پارامترهای som مشابه سوال ۱ و۲ میباشد (lr=0.5 , sigma=1)نتایج به صورت زیر است:

=> f1-score: 0.90

IDS2 Dataset

شبکه استفاده شده برای این دیتاست مشابه سوال قبل است (مشابه سوال دارای چهار لایه می‌باشد و لایه مخفی به ترتیب دارای 256,128,64 نورون است و از بهینه ساز adam با نرخ یادگیری ۰.۰۱) و پارامترهای som مشابه سوال ۱ و۲ میباشد (lr=0.7 , sigma=1)نتایج به صورت زیر است:

=> f1-score: 0.99

برای دیتاست Lung با کاهش ابعاد داده نتیجه بهتر می شود(برای آموزش علاوه بر مشکل نامتوازن بودن زیاد بودن پارامتر ورودی وجود دارد)، زیرا برای آموزش این مدل در سوال ۳ مشکل ازدیاد ابعاد داده وجود داشت و در قسمت ۴ با کاهش ابعاد داده به وسیله som این مشکل برطرف میشود و نتایج در قسمت ۳ بهبود میابد.

برای دیتاست IDS2 به دلیل اینکه مشکل یادگیری، نامتوازن بودن دیتا بود در قسمت ۳ با oversampling و قسمت ۴ تنها با استفاده از som یعنی نگاشت داده به فضای کمتر و سپس آموزش مدل این مشکل برطرف میشود و هر دو خروجی یکسانی دارند. در آموزش مدل MLP زمانی که داده مشکل نامتوازن یا زیاد بودن فضای ورودی وجود دارد میتوان ابتدا از som استفاده کرد و سپس از NLP استفاده کرد .