یک روش بدون پارامتر مبتنی بر نزدیکی برای تشخیص داده‌های پرت

یحیی صالحی و نگین دانشپور

چکیده: تشخیص داده‌های پرت به عنوان یک حوزه تحقیق در داده‌کاوی و یادگیری ماشین بوده و یک گام مهم در پیش‌پردازش داده‌ها به حساب می‌آید. در این مقاله یک روش بدون پارامتر به منظور تشخیص داده‌های پرت مبتنی بر نزدیکی به نام NPOD ارائه شده است. رهیافت ارائه‌شده، ترکیبی از روش‌های مبتنی بر فاصله و مبتنی بر چگالی بوده و توانایی تشخیص پرت‌ها را به صورت سراسری و محلی دارد. این روش نیاز به تعیین هیچ یک از پارامترهای شعاع همسایگی، حد آستانه نقاط موجود در شعاع همسایگی و پارامتر نزدیک‌ترین همسایگی ندارد. NPOD برای تشخیص داده‌های پرت، یک روش جدید نمره‌دهی ارائه می‌دهد. ارزیابی نتایج بر روی مجموعه داده‌های UCI نشان می‌دهد که این الگوریتم با وجود بدون پارامتر بودنش، عملکردی قابل رقابت با روش‌های پیشین و در بعضی مواقع بهترین عملکرد را دارد.

کلیدواژه: **بدون پارامتر، تشخیص داده‌های پرت، مبتنی بر نزدیکی.**

۱- مقدمه

امروزه با گسترش حضور کامپیوتر در جوامع بشری و استفاده از آن در تجارت، علوم و کارهای اداری، حجم بالایی از داده‌ها تولید شده است. این داده‌ها به منظور انجام پردازش‌های آتی در منابع داده‌ای ذخیره می‌شوند. علی‌رغم امکان دسترسی به این حجم داده ارزشمند، برخی رخدادهای اقلیت و نادر در میان آنها به چشم می‌خورد. این رخدادهای نادر ممکن است بی‌ارزش بوده و هدف، حذف آنها از مجموعه داده باشد و یا این که حاوی اطلاعات ارزشمندی باشند و هدف، شناسایی این داده‌ها باشد. کشف و شناسایی چنین داده‌هایی مورد توجه داده‌کاوی و یادگیری ماشین قرار گرفته است. چنین داده‌هایی در ادبیات داده‌کاوی، پرت[1] یا ناهنجاری[2] نامیده می‌شوند.

در حوزه کشف دانش[3]، تشخیص داده‌های پرت کاربردهای فراوانی دارد. از جمله این کاربردها می‌توان به تشخیص نفوذ در شبکه‌های کامپیوتری[4]، تشخیص خطای کارت‌های اعتباری[5]، تشخیص پزشکی[6] و غیره اشاره کرد. در هر یک از این کاربردها داده‌های پرت، داده‌های اقلیت هستند. به عنوان مثال در تشخیص نفوذ در شبکه‌های کامپیوتری، تعداد حملات بسیار کمتر از تعداد اتصال‌های نرمال خواهد بود.

به عنوان یک تعریف رسمی و دقیق از داده پرت می‌توان به تعریف ارائه‌شده توسط هاوکینز [۱] اشاره کرد: "یک داده پرت، مشاهده‌ای است که به قدری از سایر داده‌ها انحراف داشته باشد، گویا با مکانیزم دیگری تولید شده است".

تلاش‌های زیادی به منظور دسته‌بندی روش‌های تشخیص پرت انجام گرفته است. طبق دسته‌بندی انجام‌شده توسط هان و همکاران در [۲]، رهیافت‌های تشخیص پرت به ۴ دسته مبتنی بر مدل آماری[7]، مبتنی بر نزدیکی[8]، مبتنی بر خوشه‌بندی[9] و مبتنی بر رده‌بندی[10] تقسیم می‌شوند.

در تعریف رهیافت‌های مبتنی بر نزدیکی فرض می‌شود که نزدیکی یک شیء پرت، نسبت به نزدیک‌ترین همسایگانش به طور آشکاری از نزدیکی شیء، نسبت به اکثریت اشیای دیگر، انحراف داشته باشد. در این رهیافت، یک معیار برای تعیین شباهت[11] بین اشیا لازم است و به این ترتیب، اشیایی که خیلی دورتر از دیگر اشیا هستند را می‌توان پرت در نظر گرفت.

دو نوع روش تشخیص داده‌های پرت مبتنی بر نزدیکی وجود دارد که عبارتند از مبتنی بر فاصله[12] و مبتنی بر چگالی[13].

یک روش تشخیص داده‌های پرت مبتنی بر فاصله، همسایگان یک شیء را با تعریف یک شعاع همسایگی r به وجود می‌آورد. طبق این تعریف در صورتی یک شیء، پرت در نظر گرفته می‌شود که در مجموعه همسایگان آن به تعداد کافی، شیء وجود نداشته باشد (یعنی تعداد همسایگان، کمتر از یک حد آستانه m باشد). همچنین می‌توان پرت‌بودن مبتنی بر فاصله را با بررسی هر شیء داده از مجموعه اشیا در k نزدیک‌ترین همسایگان آن تشخیص داد که در این صورت علاوه بر پارامترهای r و m، تعیین مقدار بهینه پارامتر k نیز یک چالش خواهد بود.

طبق [۲] روش‌های مبتنی بر فاصله تنها قادر به تشخیص پرت‌های سراسری[14] هستند. به دو دلیل، پرت‌های تشخیص داده شده در این روش را پرت‌های سراسری می‌گویند:

۱) پرت‌ها به اندازه $m-1$ درصد، دورتر از اکثریت اشیا در مجموعه داده هستند.

۲) برای تشخیص پرت‌های مبتنی بر فاصله، حداقل نیاز به تعیین دو پارامتر سراسری r و m است که برای تمام داده‌های موجود در مجموعه داده به کار می‌رود.

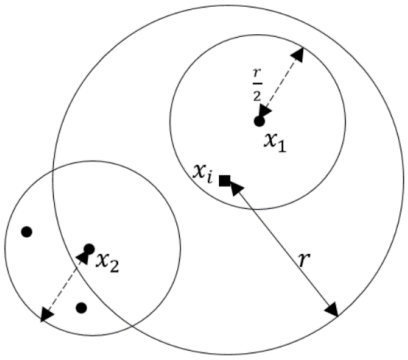
ضعف روش‌های تشخیص پرت مبتنی بر فاصله، محدودیت در تشخیص تنها؛ نقاط پرت سراسری است زیرا با توجه به ماهیت و ساختار

این مقاله در تاریخ ۸ مهر ماه ۱۳۹۷ دریافت و در تاریخ ۱۵ بهمن ماه ۱۳۹۷ بازنگری شد.

یحیی صالحی، دانشکده مهندسی کامپیوتر، دانشگاه تربیت دبیر شهید رجایی، تهران، ایران، (email: y.salehi@sru.ac.ir).

نگین دانشپور (نویسنده مسئول)، دانشکده مهندسی کامپیوتر، دانشگاه تربیت دبیر شهید رجایی، تهران، ایران، (email: ndaneshpour@sru.ac.ir).

1. Outlier
2. Anomaly
3. Knowledge Data Discovery
4. Intrusion Detection
5. Fraud Detection
6. Medical Diagnostics
7. Statistical-Based
8. Proximity-Based
9. Clustering-Based
10. Classification-Based
11. Similarity
12. Distance-Based
13. Density-Based
14. Global



شکل ۱: نمایش شعاع $r-$همسایگی نقطه x_i شامل همسایگان x_1 و x_2 و شعاع $r/2$ همسایگی برای هر یک از همسایگان x_i.

پیچیده مجموعه داده‌های واقعی[1]، شناسایی پرت‌ها، تنها با در نظر داشتن تعداد همسایگان در یک شعاع همسایگی کافی نیست.

در مقابل، روش‌های تشخیص پرت مبتنی بر چگالی به شناسایی پرت‌های محلی[2] می‌پردازند. در این روش، چگالی پیرامون یک داده با چگالی پیرامون مجموعه همسایگان محلی آن مقایسه می‌شود. در تشخیص پرت‌های مبتنی بر چگالی، فرض بر این است که چگالی پیرامون یک شیء غیر پرت (نرمال)، شبیه به چگالی پیرامون مجموعه همسایگان محلی آن است. در حالی که چگالی پیرامون یک شیء پرت، به طور آشکار، متفاوت از چگالی پیرامون همسایگان آن خواهد بود. بنابراین در روش‌های مبتنی بر چگالی، از نسبت چگالی یک شیء به همسایگانش برای نشان‌دادن درجه پرت‌بودن (نمره پرتی) استفاده می‌شود.

ضعف روش‌های مبتنی بر چگالی نیز تشخیص پرت‌بودن، تنها با در نظر داشتن مجموعه همسایگان یک شیء است.

روش ارائه‌شده در این مقاله، تشخیص پرت‌ها را با ترکیب ویژگی‌های روش‌های مبتنی بر فاصله و مبتنی بر چگالی انجام می‌دهد و به طور خلاصه دارای مزایای زیر است:

۱) روش پیشنهادی، شناسایی پرت‌ها را هم به صورت سراسری و هم به صورت محلی انجام می‌دهد.

۲) اصلی‌ترین چالش هر یک از روش‌های مبتنی بر فاصله و مبتنی بر چگالی، تعیین پارامترهای r، m و k است که در روش پیشنهادی این مقاله، نیاز به تعیین هیچ یک از این پارامترها نبوده و لذا یک روش بدون پارامتر به حساب می‌آید.

۳) خروجی این روش به صورت باینری، پرت‌بودن یا نبودن داده‌ها را تعیین نمی‌کند بلکه با اختصاص یک نمره به هر داده، میزان پرت‌بودن آن را مشخص می‌کند.

در این مقاله الگوریتم جدیدی ارائه می‌شود که دارای تمام مزایای گفته‌شده است. روش پیشنهادی با نام NPOD[3] برای تمام نقاط موجود در مجموعه داده، لیستی مرتب از تمامی همسایگان را به ترتیب صعودی فاصله نگهداری می‌کند. برای هر نقطه، شعاع r، فاصله تا نقطه بعدی خواهد بود. با پیمایش لیست، این فاصله دائماً در حال زیادشدن است. مطابق شکل ۱، روال کار الگوریتم NPOD به این صورت است که برای هر نقطه x_i، با افزایش شعاع r در هر محله، به ازای تک‌تک نقاط همسایه موجود در شعاع r، نسبت مجموع تعداد نقاط موجود در شعاع $r/2$ هر یک از نقاط همسایه به تعداد نقاط موجود در شعاع $r-$همسایگی نقطه x_i، تابع میانگین محلی را تشکیل می‌دهد. از طرفی تعداد نقاط موجود در فاصله r از نقطه x_i، تابع $r-$همسایگی را تشکیل می‌دهد. این دو تابع با یکدیگر در طول بازه‌های افزایش شعاع‌های همسایگی r، نمودارهایی را ایجاد می‌کنند که با محاسبه مساحت بین دو نمودار ایجادشده، نمره پرتی هر داده تعیین می‌شود.

در این مقاله تمرکز بر روی داده‌های ایستا[4] است که در آن تمام داده‌ها در ابتدا در دسترس هستند. در مقابل داده‌های جریانی[5] قرار دارد که در آن به دلیل محدودیت‌هایی نظیر حافظه و یا ماهیت داده‌ها، کل داده‌ها در ابتدا در دسترس نیست [۳].

روش ارائه‌شده در این مقاله نسبت به روش‌های معروف پیشین، بر روی مجموعه داده‌های UCI ارزیابی می‌شود. ارزیابی دقیق نتایج بر روی پنج مجموعه داده طبیعی نشان می‌دهد که این الگوریتم، علی‌رغم بدون پارامتربودن، می‌تواند بر روی مجموعه داده‌های با ویژگی و ساختار متفاوت، نتایج خوبی را ارائه دهد.

در ادامه این مقاله، در بخش دوم مروری بر مطالعات و روش‌های پیشین گردیده است. بخش سوم به بررسی دقیق روش ارائه‌شده پرداخته و در بخش چهارم، ارزیابی نتایج آزمایش‌های انجام‌گرفته ارائه شده است. بخش پنجم نیز یک نتیجه‌گیری کلی را از روش پیشنهادی ارائه می‌دهد.

۲- پیشینه تحقیق

در این بخش به دسته‌بندی رهیافت‌های موجود به منظور تشخیص داده‌های پرت و ارائه چندین روش معتبر از برخی رهیافت‌های ارائه‌شده، پرداخته شده است. همان طور که در بخش مقدمه گفته شد در این مقاله تمرکز بر روی داده‌های ایستا است که در آن تمام داده‌ها در ابتدا در دسترس هستند. از این رو تمرکز این مقاله، بررسی رهیافت‌های مبتنی بر نزدیکی در محیط‌های ایستا است. همچنین تمام روش‌هایی که در بخش نتایج آزمایش‌ها مورد مقایسه قرار گرفته‌اند، قابل اجرا بر روی داده‌های ایستا هستند.

طبق دسته‌بندی انجام‌شده در بخش مقدمه، رهیافت‌های تشخیص داده‌های پرت به ۴ دسته تقسیم می‌گردند که به تفکیک، به شرح هر یک پرداخته شده است.

۲-۱ رهیافت‌های مبتنی بر مدل آماری

این رهیافت، یک مدل احتمالاتی مولد و یا یک تابع توزیع احتمال برای مجموعه داده می‌یابد و سپس بررسی می‌کند که آیا داده آزمون ورودی از مدل ایجادشده پیروی می‌کند یا خیر. داده آزمونی که به وسیله مدل احتمالاتی فرض‌شده تولید نشود، پرت در نظر گرفته می‌شود. در این رهیافت‌ها داشتن دانش پیشین درباره داده‌ها الزامی است [۴]. در [۵] یک مدل ترکیبی گاوسی برای نمایش رفتارهای نرمال استفاده شده است. در این روش به هر داده، به اندازه انحراف از مدل، یک نمره اختصاص داده شده است. نمرات بالاتر نشان‌دهنده احتمال پرت‌بودن بیشتر است.

۲-۲ رهیافت‌های مبتنی بر خوشه‌بندی

در این رهیافت‌ها فرض می‌شود که داده‌های غیر پرت متعلق به خوشه‌های بزرگ و پرتراکم هستند در حالی که پرت‌ها متعلق به خوشه‌های کوچک و یا کم‌تراکم بوده یا اصلاً به هیچ خوشه‌ای تعلق

1. Real-Life
2. Local
3. Non-Parameter Proximity-Based Outlier Detection
4. Static Data
5. Streaming Data

ندارند. در این رهیافت‌ها به منظور مدل‌کردن توزیع داده‌ها، در ابتدا خوشه‌بندی‌هایی را بر روی داده‌ها ایجاد می‌کنند. سپس بر اساس الگوریتمی، داده‌های موجود در خوشه‌های کوچک و نیز داده‌هایی که از مراکز خوشه‌های به دست آمده در گام قبلی دور هستند را به عنوان داده پرت در نظر می‌گیرند. در [۶] الگوریتمی ارائه شده که پس از خوشه‌بندی داده‌ها، بر اساس اندازه خوشه‌ها با ایجاد یک گراف تصمیم، خوشه‌های پرت را تشخیص می‌دهد. در [۷] نیز یک الگوریتم به نام KMOR ارائه شده که به وسیله دو تابع هدف، یکی به منظور تشخیص داده‌های پرت، هم‌زمان با خوشه‌بندی و دیگری به منظور همگرایی پارامترهای تابع هدف اول ارائه شده است. در KMOR با استفاده از الگوریتم خوشه‌بندی K-Means در نهایت $k+1$ خوشه ایجاد می‌شود که داده‌های نرمال در k خوشه و یک خوشه برای داده‌های پرت در نظر گرفته می‌شود. در KMOR اعضای درون خوشه پرت در طول اجرای الگوریتم، همواره در حال به روز رسانی هستند. لازم به ذکر است الگوریتم‌هایی که از خوشه‌بندی K-Means استفاده می‌کنند همواره حساس به انتخاب مقدار مناسب پارامتر k هستند و این از معایب این نوع خوشه‌بندی به حساب می‌آید.

۲-۳ رهیافت‌های مبتنی بر رده‌بندی

در صورت داشتن یک مجموعه داده آموزش با کلاس‌های برچسب‌دار، می‌توان مسئله تشخیص پرت را به صورت یک مسئله رده‌بندی در نظر گرفت. در این رهیافت، هدف، آموزش رده‌بندی است که بتواند داده‌های پرت را از نرمال تفکیک دهد. در مسایل رده‌بندی به دلیل سادگی، اغلب از یک مدل یک‌کلاسه استفاده می‌شود. به این صورت که یک رده‌بند، تنها به منظور تشخیص کلاس نرمال ساخته می‌شود و نمونه‌هایی که به کلاس نرمال تعلق نداشته باشند پرت در نظر گرفته می‌شوند [۲].

۲-٤ رهیافت‌های مبتنی بر نزدیکی

طبق آنچه در مقدمه توضیح داده شد، رهیافت‌های مبتنی بر نزدیکی به دو دسته روش‌های مبتنی بر فاصله و روش‌های مبتنی بر چگالی تقسیم می‌شوند که به دلیل ضعف‌های موجود، روش‌های مبتنی بر فاصله کمتر مورد توجه قرار گرفته‌اند. از جمله معروف‌ترین روش‌های مبتنی بر چگالی، روش LOF است [۸]. این روش یک فاکتور پرت عددی برای تعیین پرت‌بودن داده‌ها به هر داده نسبت می‌دهد. روش LOF طبق (۱)، فاصله نامتقارن محلی[1] مجموعه KNN یک داده آزمون را با فاصله نامتقارن محلی همسایگان هر عضو مجموعه KNN مقایسه می‌کند

$$LOF_k(p)=\frac{1}{k}\sum_{o\in N(p,k)}\frac{lrd_k(o)}{lrd_k(p)} \qquad (1)$$

رابطه (۲) محاسبه فاصله نامتقارن محلی را نشان می‌دهد که به صورت معکوس میانگین فاصله نامتقارن[2] تعریف می‌شود

$$lrd_k(p)=(\frac{1}{k}\sum\nolimits_{o\in N(p,k)} reach-dist_k(p,o))^{-1} \qquad (2)$$

فاصله نامتقارن طبق (۳)، بیشینه فاصله یک نقطه در مجموعه داده مانند نقطه o تا kامین نزدیک‌ترین همسایه آن و فاصله o تا نقطه p است. در این رابطه $k-dist(o)$ فاصله نقطه o تا kامین نزدیک‌ترین همسایه خودش است

$$reach-dist_k(p,o)=\max\{k-dist(o),d(p,o)\} \qquad (3)$$

در (۳) بیشترین مقادیر LOF، مربوط به داده‌های پرت است.

SimplifiedLOF [۹] الگوریتم دیگری است که با جایگزین‌کردن k نزدیک‌ترین همسایه به جای فاصله نامتقارن در LOF، محاسبات آن را ساده‌تر کرده است. در نتیجه این جایگزینی، تخمین چگالی ساده‌تر و به صورت (۴) خواهد بود

$$dens(p)=\frac{1}{k-dist(p)} \qquad (4)$$

روش دیگری به نام INFLO در [۱۰] ارائه شده است که مشابه simplifiedLOF عمل می‌کند با این تفاوت که در آن برای تخمین چگالی از اجتماع مجموعه‌های KNN و RKNN استفاده گردیده است. در [۱۱] یک روش با نام LoOP ارائه شده که نسبت به SimplifiedLOF تخمین چگالی قوی‌تری را بر اساس مجذور میانگین فاصله مطابق (۵) ارائه می‌دهد

$$LoOP-dens(p)=\frac{1}{\sqrt{\frac{1}{|KNN(p)|}\sum_{o\in KNN(p)}d(p,o)^2}} \qquad (5)$$

دیگر تفاوت LoOP با LOF این است که LoOP نمرات پرت را به صورت نرمال‌شده ارائه می‌دهد.

در [۱۲] یک فاکتور پرت مبتنی بر فاصله محلی (LDOF) ارائه شده که پرت‌بودن را به صورت نسبت میانگین فاصله هر نقطه تا مجموعه KNN آن به میانگین جفت فواصل درون مجموعه KNN تعیین می‌کند. LDOF داده‌های پرتی که از مجموعه KNN دور هستند را به خوبی تشخیص می‌دهد. برای محاسبه LDOF تمام جفت فواصل نقاط محاسبه می‌شود که این کار سبب افزایش پیچیدگی خواهد شد.

در [۱۳] روشی به نام KDEOS برای نمره‌دهی پرت‌ها ارائه شده است. این روش از یک تخمین چگالی هسته (KDE)[3] با توزیع گاوسی استفاده می‌کند که با تابع تخمین چگالی LOF جایگزین شده است. چگالی KDE با در نظر داشتن مجموعه KNN همسایگان و از طریق (۶) با گرفتن میانگین مقادیر روی $k_{\min}$ و $k_{\max}$ و نیز نرمال‌کردن نتایج برای هر نقطه با تابع z-score به دست می‌آید

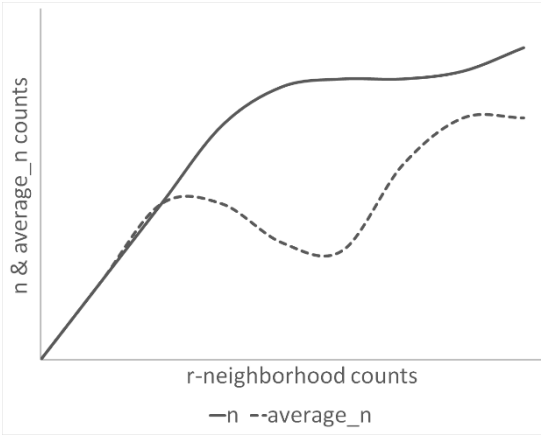
$$s(p)=\underset{k_{\min},\ldots,k_{\max}}{mean} z-score(KDE_k(p),\{KDE_k(o)\}_{o\in KNN(p)}) \qquad (6)$$

پس از آن که نمره موقتی s از (۶) به دست آمد، با استفاده از یک تابع چگالی ترکیبی نرمال، نمره پرت نهایی KDEOS تعیین می‌شود.

در [۱۴] روشی به نام ODIN، پرت‌ها را به عنوان یال‌هایی غیر مجاور در گراف همسایگی KNN معرفی می‌کند. در این روش برای شناسایی یال‌های غیر همجوار از cardinality مجموعه RKNN استفاده می‌شود.

در [۱۵] روشی با نام KNNW ارائه شده که به منظور کاهش انحراف در امتیازات و کاهش حساسیت نمرات به پارامتر k، مجموع فاصله به مجموعه k نزدیک‌ترین همسایگان یک شیء را مورد استفاده قرار می‌دهد. در این روش، مجموع فواصل زیاد، نشان‌دهنده چگالی کم است و در نتیجه پرت‌ها اشیایی با بالاترین امتیازات خواهند بود.

1. Local Reachability Distance
2. Reachability (Symmetric) Distance
3. Kernel Density Estimation



شکل ۲: شمای تعداد $r-$همسایگان و میانگین محلی در بازه پیوسته شعاع‌های افزایشی.

در [۱۶] یک الگوریتم تشخیص پرت و نسخه تسریع‌شده آن با به کارگیری محاسبات تقریبی ارائه شده است. این روش به صورت باینری داده‌های پرت را برچسب می‌زند. همچنین این الگوریتم نمودارهایی[1] را به منظور نمایش اطلاعات مفید مانند تعیین خوشه، میکروخوشه، قطر و فاصله درون کلاسی در صورت قرارداشتن نمونه در یک کلاس به کاربر نشان می‌دهد. در این روش، یک فاکتور $MDEF$ تعیین شده و در صورتی که مقدار $MDEF$ برای هر داده بیش از سه برابر انحراف معیار استاندارد از میانگین‌های محلی انحراف داشته باشد، آن داده به عنوان پرت برچسب زده می‌شود.

با توجه به موفقیت روش‌های مبتنی بر چگالی در تشخیص داده‌های پرت، این رهیافت‌ها بیش از سایرین مورد توجه قرار گرفته‌اند. لذا در این مقاله روشی ترکیبی که هم مبتنی بر فاصله و هم مبتنی بر چگالی است ارائه شده که این روش، بدون نیاز به تعیین پارامتر، قابل رقابت با روش‌های موجود می‌باشد. به همین جهت تعدادی از این روش‌ها جهت مقایسه با الگوریتم پیشنهادی انتخاب شده است.

۳- رویکرد پیشنهادی

در این بخش ابتدا به بیان تعاریفی دقیق و رسمی از روش ارائه‌شده در این مقاله و سپس به بیان شبه‌کد الگوریتم پیشنهادی پرداخته می‌شود.

۳-۱ تعاریف اولیه و بیان روش پیشنهادی

تا کنون روش‌های متعددی به منظور تشخیص پرت‌ها ارائه شده که بسیاری از آنها مبتنی بر KNN هستند، اما ضعف بزرگ روش‌های مبتنی بر KNN، تعیین پارامتر k است که انتخاب آن به مجموعه داده ورودی وابسته است [۱۷]. لذا انتخاب مقدار بهینه k همواره یکی از مشکلات متخصصان حوزه علم داده بوده است. به طور کلی مقادیر بزرگ k از تأثیر نویزها در طبقه‌بندی می‌کاهد اما فاصله بین کلاسی را افزایش می‌دهد که در نتیجه آن، جدایی‌پذیری کلاس‌ها کاهش می‌یابد. از طرف دیگر، اگر k خیلی کوچک انتخاب شود همبستگی همسایگان را کاهش می‌دهد یا به عبارت دیگر نقاط درون یک کلاس را تفکیک می‌کند.

با این حال، روش‌های مبتنی بر چگالی که عمدتاً از k نزدیک‌ترین همسایگی بهره می‌برند، در کل عملکرد بهتری نسبت به سایر روش‌های تشخیص پرت دارند. همواره در روش‌های مبتنی بر فاصله، مسئله تعیین شعاع همسایگی، چالش‌برانگیز است و در داده‌هایی که چگالی خوشه‌ها از یکدیگر متفاوت است، عملکرد خوبی نخواهند داشت.

به منظور رفع مشکلات فوق، الگوریتم پیشنهادی NPOD با در نظر گرفتن شعاع همسایگی افزایشی به وسعت کل داده‌ها و اعمال محدودیت با ضریب ۰/۵ در هر شعاع همسایگی، می‌تواند در دسته روش‌های مبتنی بر فاصله قرار بگیرد که مشکل تعیین پارامتر ندارند. همچنین با توجه به مقایسه چگالی پیرامون یک نقطه با همسایگان آن و نیز محاسبه نمره پرت خروجی، این الگوریتم در شمار روش‌های مبتنی بر چگالی نیز محسوب می‌شود.

پیش از توضیح چگونگی عملکرد الگوریتم، ارائه چند تعریف الزامی است.

تعریف ۱: با فرض این که مجموعه داده X و شعاع r داده شده باشد، مجموعه $r-$همسایگان یک نقطه x_i به فرم (۷) تعریف می‌شود

$$N(x_i,r)=\{x\in X\,|\,dist(x,x_i)\le r\} \qquad (۷)$$

این مجموعه همسایگی، نقطه x_i را نیز شامل می‌شود. این کار به دلیل جلوگیری از خطای تقسیم بر صفر در رابطه‌های بعدی در نظر گرفته شده است.

تعریف ۲: با فرض داشتن مجموعه $r-$همسایگان نقطه x_i از (۷)، تعداد نقاط در $r-$همسایگی نقطه x_i به صورت (۸) تعریف می‌شود

$$n(x_i,r)=|N(x_i,r)| \qquad (۸)$$

بدیهی است با توجه به این که هر نقطه، خود در مجموعه $r-$همسایگانش قرار دارد، حداقل مقدار $n(x_i,r)$ برابر یک خواهد بود.

تعریف ۳: با فرض داشتن مجموعه $r-$همسایگان نقطه x_i و همچنین داشتن مجموعه $r/۲-$همسایگان هر یک از نقاط موجود در فواصل افزایشی $r-$همسایگی، (۹) میانگین محلی را نشان می‌دهد

$$\overline{n}(x_i,r)=\frac{\sum_{x\in N(x_i,r)} n(x,\frac{r}{۲})}{n(x_i,r)} \qquad (۹)$$

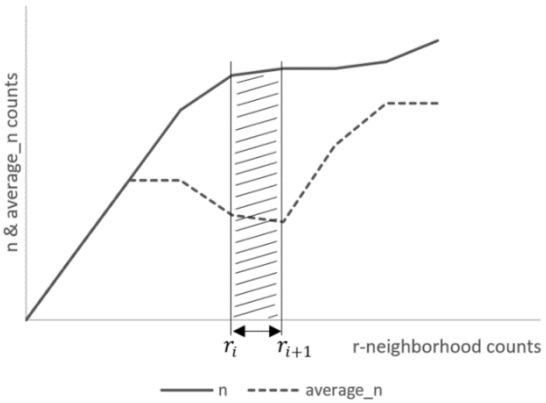
پاپادیمیتریو و همکاران تعاریف فوق را برای محاسبه فاکتور MDEF استفاده کردند [۱۶]. در تعریف آنها، ضریب ۰/۵ یک مقدار مناسب برای شعاع همسایگان محلی در نظر گرفته شده است. در روش آنها تشخیص داده‌های پرت به صورت نمره‌دهی نبوده بلکه داده‌های پرت و نرمال برچسب زده می‌شدند. الگوریتم ارائه‌شده در این مقاله با تخصیص نمره پرت به هر داده، در شعاع‌های افزایشی $r-$همسایگی، امکان تعیین میزان پرت‌بودن یک داده از سایر داده‌ها را فراهم می‌کند و لذا تصمیم‌گیری برای پرت‌بودن یک داده به صورت باینری انجام نمی‌شود.

طبق تعاریف فوق اگر تعداد نقاط در $r-$همسایگی و میانگین محلی بر حسب بازه شعاع‌های پیوسته برای یک نقطه رسم شود، شمایی مطابق شکل ۲ به دست خواهد آمد. نمایش شمای شکل ۲، تنها به منظور درک بهتر صورت مسئله است و در الگوریتم ارائه‌شده در این مقاله، نیاز به رسم شما برای کاربر نیست.

لم ۱: حاصل هر چندجمله‌ای $P_n(x)$ را می‌توان به صورت مجموع انتگرال‌گیری از تابع چندجمله‌ای در کل بازه تعریف‌شده و خطای انتگرال‌گیری طبق (۱۰) به دست آورد [۱۸]

$$P_n(x)=\int_a^b f(x)\mathrm{d}x+\Delta(f(x)) \qquad (۱۰)$$

1. LOCI-Plot



شکل ۳: مساحت زیر سطح منحنی تعداد $r-$ همسایگان.

در (۱۰) چندجمله‌ای از درجه n است که $f(x)$ تابع چندجمله‌ای در کل بازه $a \le x \le b$ و دلتا، خطای انتگرال‌گیری تابع چندجمله‌ای است.

لم ۲: محاسبه مقدار تقریبی انتگرال معین $f(x)$ بر بازه $a \le x \le b$ با استفاده از $N+1$ نقطه نمونه $(x_0,f_0),(x_1,f_1),\ldots,(x_N,f_N)$ که در آن $f_k = f(x_k)$ با توجه به جدولی‌بودن نقاط، از قاعده انتگرال‌گیری عددی به روش ذوزنقه‌ای[1] قابل محاسبه است [۱۸].

در الگوریتم ارائه‌شده NPOD از آنجایی که بازه‌های r، مقادیر عددی گسسته است، درجه چندجمله‌ای، خطی می‌باشد و لذا برای محاسبه انتگرال معین، تابع خطی $f(x)$ روی هر یک از زیربازه‌های گسسته r، در صورت استفاده از قاعده انتگرال‌گیری عددی به روش ذوزنقه‌ای، مقدار خطا برابر با صفر و حاصل مجموع انتگرال‌گیری روی تمام زیربازه‌های r دقیقاً برابر با حاصل کل چندجمله‌ای $f(x)$ و در نتیجه همان حاصل $n(x_i,r)$ یعنی میانگین محلی خواهد بود.

با توجه به دو لم فوق، الگوریتم NPOD برای محاسبه مساحت زیر سطح منحنی تعداد $r-$همسایگان هر نقطه و منحنی میانگین محلی، با توجه به جدولی‌بودن تابع $f(x)$ از قاعده ذوزنقه‌ای استفاده خواهد کرد.

لم ۳: طبق قاعده ذوزنقه‌ای محاسبه انتگرال عددی تابع f که به صورت جدولی است، در صورتی که i هر یک از مقادیر گسسته r باشد، مساحت بین هر دو شعاع متوالی در هر یک از چندجمله‌ای‌های میانگین محلی و تعداد $r-$همسایگان به صورت (۱۱) خواهد بود [۱۹]

$$\int_{r_i}^{r_{i+1}} f(x)\mathrm{d}x = \frac{(r_{i+1}-r_i)}{2}(f_i+f_{i+1}) \qquad (11)$$

در شکل ۳ مساحت حاصل از انتگرال‌گیری بین دو شعاع همسایگی متوالی برای تابع تعداد $r-$همسایگان در قسمت هاشورخورده نشان داده شده است. به طریق مشابه، مساحت زیر سطح منحنی میانگین محلی نیز به دست می‌آید.

با تفاضل مساحت زیر منحنی‌های تعداد $r-$همسایگان محلی و منحنی میانگین محلی و جمع تمام تکه مساحت‌های بین جفت فواصل متوالی برای هر نقطه، نمره پرتی آن نقطه به دست می‌آید.

با توجه به این که ممکن است در بعضی بازه‌ها، هر یک از نمودارها بالاتر از دیگری قرار بگیرد، ما از قدرمطلق تفاضل این مساحت‌ها برای محاسبه نمره پرت استفاده می‌کنیم.

با توجه به نمرات به دست آمده برای داده‌ها، نمرات بیشتر به معنای مساحت بیشتر بین دو نمودار میانگین محلی و $r-$همسایگان بوده است به این معنی که در بازه شعاع‌های افزایشی، تراکم $r/2-$همسایگان نسبت به میانگین محلی بیشتر بوده و تراکم بالاتر نشان‌دهنده نرمال‌بودن نقاط است و در نتیجه نقاط پرت، کمترین مساحت‌ها و کمترین نمرات را دارند.

۳-۲ شبه‌کد الگوریتم پیشنهادی

الگوریتم پیشنهادی در دو فاز پیش‌پردازش و پردازش اصلی ارائه شده است. همچنین به منظور درک بهتر الگوریتم ارائه‌شده، شبه‌کد مربوط به (۷) تا (۹) نیز در قالب تعریف توابع ۱ تا ۳ در شکل ۴- ب آورده شده است. شکل ۴- الف، شبه‌کد الگوریتم NPOD را نشان می‌دهد. در ادامه به تعریف توابع استفاده‌شده در الگوریتم پرداخته خواهد شد.

به طور کلی شبه‌کد الگوریتم ارائه‌شده را می‌توان به دو بخش پیش‌پردازش (خطوط ۱ تا ۸) و پردازش اصلی، شامل محاسبه نمره پرت NPOD برای هر داده، طبق تعاریف و لم‌های ارائه‌شده در بخش ۳-۱ (خطوط ۹ تا ۱۸) تقسیم کرد.

خطوط ۱ تا ۵ فاصله تمام جفت‌داده‌های موجود در مجموعه داده را محاسبه و حاصل را در ماتریس دوبعدی D ذخیره می‌کند. خطوط ۶ تا ۸ با توجه به ماتریس D، برای هر داده در مجموعه داده مانند x_i، یک لیست با نام $D(x_i)$ ایجاد می‌کند که در آن فاصله x_i با تمام همسایگان به ترتیب صعودی نگهداری می‌شود. طبق آنچه در بخش ۳-۱ گفته شد در ابتدای این لیست، فاصله نقطه تا خودش (صفر) وجود دارد که این کار به دلیل داشتن حداقل یک نزدیک‌ترین نقطه و همچنین جلوگیری از رخداد خطای تقسیم بر صفر در خطوط بعدی الگوریتم انجام می‌گیرد. $I(x_i)$ لیست دیگری است که به ترتیب، اندیس متناظر فاصله نقاط موجود در $D(x_i)$ را نگهداری می‌کند.

پس از اتمام بخش محاسبات اولیه، در بخش پردازش اصلی، خطوط ۹ تا ۱۸ به نمره پرت در روش پیشنهادی NPOD می‌پردازد. این خطوط برای هر نقطه در مجموعه داده، تمام شعاع‌ها را از نزدیک‌ترین تا دورترین فاصله در نظر گرفته و مقادیر تعداد $r-$همسایگان (خط ۱۳) و میانگین محلی (خط ۱۴) را در هر شعاع به دست می‌آورد. تابع $N(x_i,r)$ برای هر یک از نقاط x_i، اندیس همسایگانی را که در شعاع کوچک‌تر یا مساوی از r قرار دارند در یک لیست برمی‌گرداند و تابع $n(r\_neighbors)$ اندازه این لیست را محاسبه و به عنوان تعداد $r-$همسایگان تعیین می‌کند. در خط ۱۴ با فراخوانی تابع ۳، به ازای نقاط درون مجموعه $r-$همسایگان هر نقطه x_i، مجموع تمام نقاط در فاصله $r/2-$همسایگی هر یک از آنها محاسبه و به تعداد آنها تقسیم می‌شود و بدین صورت میانگین محلی محاسبه خواهد شد. طبق لم ۳ با داشتن دو شعاع متوالی، تعداد $r-$همسایگان و میانگین محلی (در نتیجه فراخوانی توابع ۲ و ۳)، می‌توان انتگرال زیر سطح هر یک از نمودارهای تعداد $r-$همسایگان و میانگین محلی را محاسبه نمود و سپس با تفاضل قدرمطلق حاصل انتگرال‌ها، مساحت محصور بین دو تابع به ازای دو شعاع متوالی را محاسبه و مقدار این مساحت را با نمره داده فعلی جمع کرد. با اتمام حلقه، مساحت محصور بین دو تابع به ازای تمام شعاع‌ها برای داده فعلی به دست آمده و به نمره پرت آن افزوده شده است. حال با افزایش گام در حلقه ۹، مطابق آنچه گفته شد نمره پرت برای داده جدید محاسبه می‌شود. با توجه به آنچه در مقدمه گفته شد خروجی الگوریتم NPOD به صورت نمرات پرت برای هر داده محاسبه شده و در نهایت با مرتب‌سازی نمرات خروجی و با توجه به ماهیت مسئله، می‌توان حد بالای پرت‌های مورد نظر را شناسایی و یا از مجموعه داده حذف نمود.

1. Trapezoid

Algorithm: NPOD
Input: Dataset X
Output: A list of $outlier\_score$
//preprocess
1. **for** $x_i \in X$ **do**
2. **for** $x_j \in X$ **do**
3. Calculate matrix D such that: $D[i][j] \leftarrow dist(x_i, x_j)$
4. **end for**
5. **end for**
6. **for** $x_i \in X$ **do**
7. - construct a sorted distance list $D(x_i)$ such that, $D(x_i)(k)$ is the distance between x_i and it's k^{th} nearest neighbor.
- construct a sorted index list $I(x_i)$ such that $I(x_i)(k)$ is the index of k^{th} nearest neighbor of x_i.
8. **end for**
//post process
9. **for** $x_i \in X$ **do**
10. $outlier\_score[x_i] \leftarrow 0$
11. **for** $r \in D(x_i)$ **do**
12. $r\_neighbors \leftarrow N(x_i, r)$
13. $r\_neighbors\_count \leftarrow n(r\_neighbors)$
14. $average\_n\_counts \leftarrow \bar{n}(x_i, r, r\_neighbors, r\_neighbors\_count)$
15. $outlier\_score[x_i]$+= trapezoid area with respect to each pairs of r, $r\_neighbors\_count$ and $average\_n\_counts$
16. **end for**
17. **end for**
18. **return** list of $outlier\_score$

(الف)

//function1 $N(x_i, r)$
{
19. $flag = TRUE$
20. $y \leftarrow 0$
21. **while** $(flag)$ **do**
22. **if** $(D(x_i)(y) \le r)$ **do**
23. $r\_neighbors$.add($I(x_i)(y)$)
24. $y++$
25. **else**
26. $flag = FALSE$
27. **end while**
return $r\_neighbors$
}
//function2 $n(r\_neighbors)$
{
28. **return** $r\_neighbors$.**size**()
}
//function3
$\bar{n}(x_i, r, r\_neighbors, r\_neighbors\_count)$
{
29. $sigma \leftarrow 0$
30. **for** $(i \in r\_neighbors)$ **do**
31. $sigma$+= $n(N(r\_neighbors(i), 1/2r))$
32. **end for**
33. **return** $sigma / r\_neighbors\_count$
}

(ب)

شکل ۴: (الف) شبه‌کد الگوریتم ارائه‌شده و (ب) تعریف توابع استفاده‌شده در شبه‌کد.

جدول ۱: مشخصات مجموعه داده‌های استفاده‌شده در آزمایش‌ها.

نام مجموعه داده	تعداد رکورد	تعداد پرت	تعداد ویژگی	نوع داده
Glass	۲۱۴	۹	۷	حقیقی نرمال
Pima	۵۱۰	۱۰	۸	حقیقی
WBC	۲۲۳	۱۰	۹	صحیح
Stamps	۳۱۵	۶	۹	حقیقی نرمال
Lymphography	۱۴۸	۶	۱۸	حقیقی، اسمی

۳-۳ تحلیل پیچیدگی زمانی

در ادامه به تحلیل پیچیدگی زمانی الگوریتم پرداخته شده است. در گام پیش‌پردازش اگر تعداد نمونه‌های ورودی الگوریتم n در نظر گرفته شود، پیچیدگی زمانی خطوط ۱ تا ۵ الگوریتم با توجه به این که طبق (۱۳) معیار سنجش فاصله بین دو نمونه، فاصله اقلیدسی است از مرتبه $O(n^2 \times M)$ است که M تعداد ویژگی‌ها در مجموعه داده است. پیچیدگی زمانی خطوط ۶ تا ۸ در صورت استفاده از مرتب‌سازی ادغامی از مرتبه $O(n^2 \times \log n)$ است و لذا در کل پیچیدگی زمانی بخش پیش‌پردازش $O(n^2(M + \log n))$ خواهد بود. در بخش پردازش اصلی (خط ۹) و حلقه درون آن (خط ۱۱) هر یک از حلقه‌ها از مرتبه $O(n)$ خواهند بود. تابع $N(x_i, r)$، $O(n)$ بار تکرار خواهد شد و در نتیجه، درون (تابع ۳)، تابع $n(N(r\_neighbors(i), 1/2r))$، n بار فراخوانی خواهد شد و در هر بار فراخوانی تعداد نقاط در شعاع $r/2$ را محاسبه خواهد کرد (تابع ۲ فراخوانی می‌شود). اگر این تعداد را λ در نظر بگیریم تعداد دفعات تکرار تابع $N(r\_neighbors(i), 1/2r)$، $O(\lambda)$ خواهد بود که $\lambda << n$. لذا پیچیدگی تابع ۳، $O(n \times \lambda)$ خواهد بود. در نهایت پیچیدگی زمانی کلی این الگوریتم از مرتبه $O(n^2 \times \lambda)$ می‌باشد که می‌توان گفت با توجه به بدون پارامتر بودن این الگوریتم، این پیچیدگی بدتر از الگوریتم‌های تشخیص پرت موجود نبوده و قابل رقابت با آنها است.

٤- نتایج آزمایش‌ها

در این بخش به مقایسه روش ارائه‌شده در این مقاله با چندین روش معروف دیگر پرداخته شده و مقایسه‌ها بر روی مجموعه داده‌های UCI انجام گردیده است. جدول ۱ مشخصات این مجموعه داده‌ها را نشان می‌دهد. برای پیاده‌سازی از زبان برنامه‌نویسی جاوا و کتابخانه‌های استاندارد آن استفاده شده است.

٤-۱ مشخصات آزمایش‌ها بر روی مجموعه داده‌های UCI

بر اساس مطالعه مروری که توسط زیمک و همکاران در [۲۰] انجام شد، انتخاب مجموعه داده‌ها و استفاده از معیار ارزیابی مناسب از چالش‌های اساسی در مسایل تشخیص داده‌های پرت است.

بر این اساس، کمپوز و همکاران در مطالعه مروری خود [۲۱]، مجموعه داده‌هایی را بر اساس مجموعه داده‌های UCI ارائه کرده‌اند که اساس مقایسات نویسندگان مقالات در زمینه تشخیص داده‌های پرت قرار گیرد. آنها از مجموعه داده‌های رده‌بندی UCI استفاده کرده‌اند. در این مجموعه داده‌ها، داده‌های پرت به استانداردترین روش ممکن تولید و برچسب‌گذاری شده است.

در این مقاله به دلیل دقت بیشتر در ارزیابی نتایج مقایسات از مجموعه

داده‌های گردآوری‌شده توسط کمپوز و همکاران استفاده شده است.

هنگامی که داده‌های پرت در مجموعه داده‌ها، برچسب‌گذاری و مشخص شده باشند، یک معیار مناسب برای ارزیابی نتایج مقایسات، معیار precision at n است. این معیار، طبق (۱۲) به صورت نسبت نتایج پرت‌های درست تشخیص داده شده به کل تعداد حد بالا یا پایین نمرات پرت واردشده به مجموعه داده تعیین می‌شود

$$precision\ at\ n = \frac{\left|\{o \in O \left| rank(o) \le n\}\right|\right.}{n} \qquad (۱۲)$$

در این رابطه n تعداد داده‌های پرت موجود در مجموعه داده و $rank(o)$ بر اساس نوع الگوریتم، تعداد پرت‌های درست تشخیص داده شده در حد بالا و یا حد پایین نمرات است.

برای سنجش فاصله دو نمونه از یکدیگر، از معیار فاصله اقلیدسی استفاده شده است

$$dist(x_i, x_j) = \sqrt{\sum_{k=1}^{M} (x_i^k - x_j^k)^2} \qquad (۱۳)$$

در انتخاب مجموعه داده‌ها برای مقایسه نتایج، طبق جدول ۱ از مجموعه داده‌های با انواع داده‌ای متفاوت و نیز با ابعاد متفاوت استفاده شده است. در این مجموعه داده‌ها کمترین تعداد ویژگی ۷ و بیشترین آنها ۱۸ می‌باشد.

در (۱۳) M تعداد ویژگی‌های مجموعه داده است. در مجموعه داده‌های استفاده‌شده، به دلیل امکان بروز خطای تقسیم بر صفر در بعضی روش‌های مورد مقایسه از جمله LOF، هیچ داده تکراری[1] وجود ندارد و رکوردهای تکراری از مجموعه داده‌ها حذف شده است. طبق تعریف ارائه‌شده در [۲]، نویز[2] با پرت متفاوت بوده و لازم است پیش از آغاز فرایند تشخیص داده‌های پرت، از مجموعه داده حذف شوند. لذا در این آزمایشات در صورت وجود، نویزها از مجموعه داده حذف شده است. همچنین ویژگی‌های اسمی[3] با استفاده از روش $TF-IDF$ طبق (۱۴) به ویژگی‌های عددی تبدیل گردیده‌اند

$$TF-IDF(w,d) = tf(w,d) \times \log \frac{N}{f(w)} \qquad (۱۴)$$

در (۱۴)، $tf(w,d)$ تعداد دفعاتی است که نمونه w در سند d تکرار شده و $f(w)$ تعداد اسنادی است که نمونه w در آنها موجود می‌باشد. N نیز تعداد اسناد است.

در ادامه نتایج ارزیابی precision at n روش پیشنهادی در این مقاله و دیگر رقبا در تشخیص داده‌های پرت آورده شده است. از آنجایی که تمام روش‌های رقیب برای مقایسه، مبتنی بر KNN هستند لذا نه تنها به ازای فقط یک مقدار k، بلکه به ازای مقادیر متفاوت k، نتایج هر یک آورده شده است. همچنین نتایج به صورت میانگین پس از ده بار اجرا[4] روی هر یک از الگوریتم‌های رقیبان و الگوریتم پیشنهادی، سنجیده شده است.

۴-۲ نتایج آزمایش‌ها بر روی مجموعه داده‌های UCI

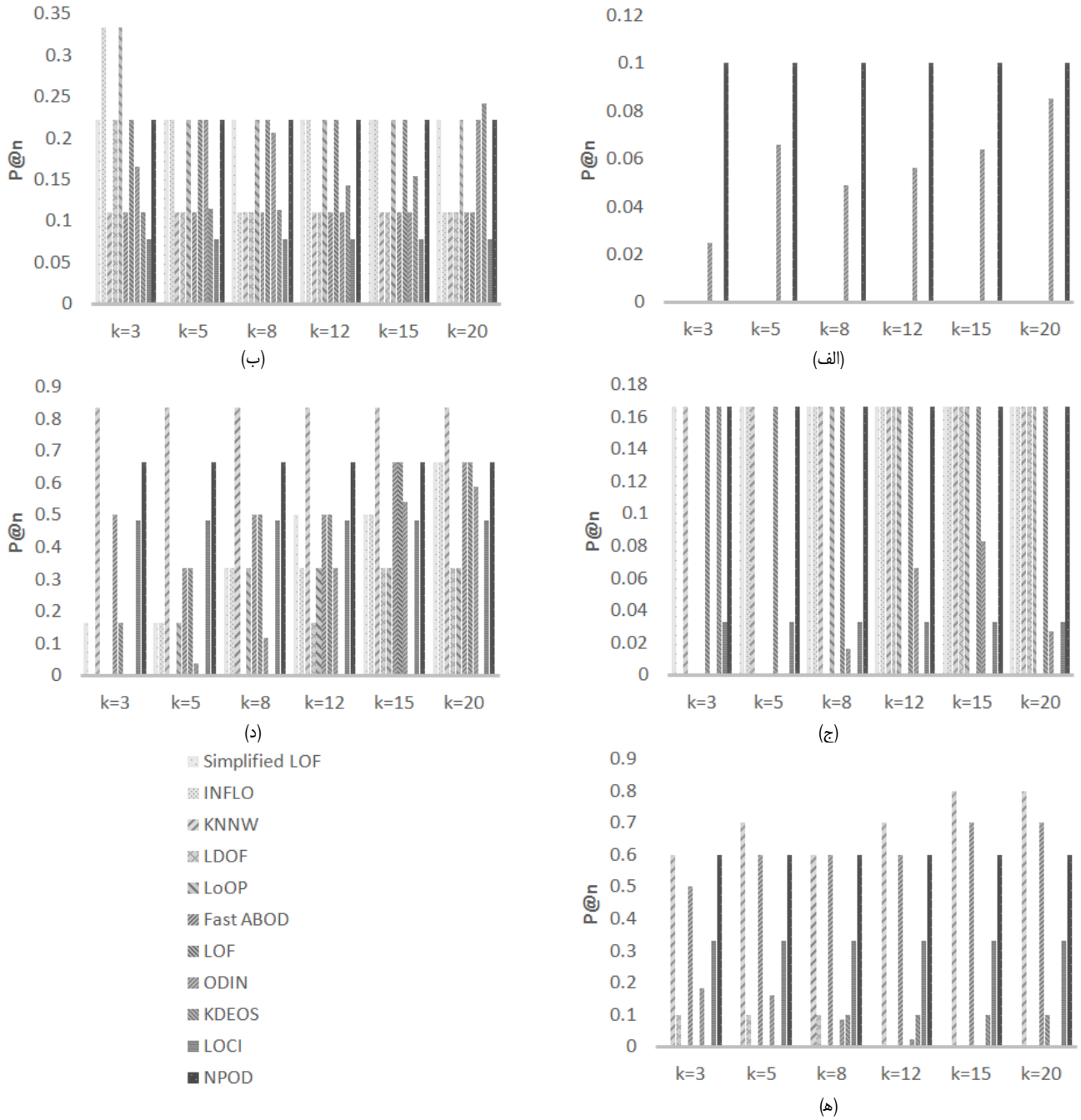
در ادامه، الگوریتم NPOD ارائه‌شده در این مقاله با ۱۰ روش معروف با نام‌های LOF، LOCI، LoOP، LDOF، INFLO، KDEOS، Fast ABOD، KNNW، SimplifiedLOF و ODIN که در بخش ۲-۴ به طور مختصر شرح داده شدند، مقایسه گردیده است. به غیر از روش LOCI، سایر الگوریتم‌های رقیب بر اساس معیار k نزدیک‌ترین همسایه می‌باشند، لذا به منظور مقایسه این الگوریتم‌ها، میانگین نتایج پس از ده بار آزمایش با مقادیر $k = ۳,۵,۸,۱۲,۱۵,۲۰$ و با معیار precision at n (به اختصار P@n) بر روی ۵ مجموعه داده Glass، Lymphography، Pima، Stamps و WBC نشان داده شده است. این در حالی است که الگوریتم بدون پارامتر NPOD و الگوریتم LOCI مستقل از مقادیر k در هر یک از مجموعه داده‌ها، در طول آزمایش‌های انجام‌شده جواب یکتا تولید کرده است.

آنچه که از مجموعه مقایسه‌های انجام‌شده در شکل ۵ استنباط می‌شود این است که الگوریتم NPOD پیشنهادی به طور قطع در دو مجموعه داده Pima و Stamps نسبت به الگوریتم‌های مورد مقایسه بهترین عملکرد را داشته و در مابقی مجموعه داده‌ها عملکرد خوبی را از خود نشان داده است. به غیر از مجموعه داده Lymphography که در آن تنها یک الگوریتم KNNW در تمامی kهای انتخاب‌شده با وجود اختلاف کمی از روش پیشنهادی، بهترین نتیجه را در ارزیابی precision at n داشته است، در مابقی مجموعه داده‌ها، هیچ الگوریتمی به ازای تمام مقادیر k، بهتر از روش ارائه‌شده در این مقاله نبوده و تنها در برخی مقادیر kهای انتخاب‌شده، الگوریتمی توانسته است بهترین نتیجه را از آن خود کند. در مجموعه داده Glass، تنها دو الگوریتم INFLO و LoOP به ازای مقدار $k = ۳$ و نیز الگوریتم KDEOS به ازای مقدار $k = ۲۰$ عملکردی بهتر از روش پیشنهادی داشته‌اند. در مجموعه داده Pima، تنها حاصل ارزیابی الگوریتم ODIN، مقداری غیر از صفر بوده و در مابقی الگوریتم‌ها به ازای تمام مقادیر k، حاصل ارزیابی، صفر بوده است. این در حالی است که الگوریتم NPOD نسبت به سایر روش‌های مورد مقایسه، بهترین عملکرد را در این مجموعه داده از آن خود کرده است. در مجموعه داده WBC تنها ۲ الگوریتم KNNW و FastABOD به ازای برخی مقادیر k، عملکرد بهتری نسبت به روش پیشنهادی داشته‌اند. سایر الگوریتم‌ها در این مجموعه داده، به ازای تمام مقادیر k، مقداری برابر صفر داشته‌اند که این مسئله قدرت الگوریتم ارائه‌شده در مجموعه داده‌های با انواع داده‌ای متفاوت را به روشنی بیان می‌کند. باید در نظر داشت که در واقعیت نمی‌توان kهای متفاوتی را برای یک مجموعه داده در الگوریتم‌های ارائه‌شده انتخاب کرد و تنها شانس قبولی با یک k پذیرفته است. با این حال (در نظر گرفتن kهای متفاوت) باز هم الگوریتم NPOD ارائه‌شده هرگز بدترین نتیجه را نداشته و توانسته در سطح قابل قبولی، برتری خود را حفظ کند.

۵- نتیجه‌گیری

در این مقاله یک روش جدید برای تشخیص داده‌های پرت مبتنی بر نزدیکی ارائه شده که ترکیبی از ویژگی‌های هر دو روش مبتنی بر فاصله و مبتنی بر چگالی می‌باشد و در عین حال یک روش بدون پارامتر محسوب می‌شود. روش ارائه‌شده با انتخاب طول افزایشی شعاع همسایگی، پرت‌ها را به صورت سراسری تشخیص می‌دهد. انتخاب شعاع همسایگی، محدود به انتخاب مقدار آن توسط کاربر نبوده و این شعاع از کمینه تا بیشینه حد خود در حال افزایش است که این ویژگی بر ضعف مبتنی بر فاصله‌بودن این الگوریتم غلبه می‌کند. تعیین نمره پرتی هر شیء با در نظر گرفتن نسبت تعداد همسایگان هر نقطه در شعاع r و نیز تعداد

1. Duplicate
2. Noise
3. Categorical
4. 10-Fold Cross Validation



شکل ۵: ارزیابی نتایج آزمایش‌های با معیار precision at n بر روی مجموعه داده‌های (الف) Pima، (ب) Glass، (ج) Stamps، (د) Lymphography و (ه) WBC.

همسایگان هر یک از همسایگان موجود در شعاع $r/2$، امکان محاسبه چگالی را فراهم آورده است. چگالی به دست آمده نیز محدود به همسایگان محلی خاصی نبوده و با توجه به افزایش شعاع، همواره همسایگان محلی نیز در حال به روز رسانی هستند. در نتیجه این ویژگی نیز بر ضعف مبتنی بر چگالی‌بودن این الگوریتم غلبه کرده است. روش ارائه‌شده در این مقاله، این مزیت را دارد که بدون در نظرگیری و محاسبه k نزدیک‌ترین نقاط همسایگی هر نقطه، تشخیص پرت را انجام می‌دهد. بدون پارامتربودن الگوریتم پیشنهادی و خروجی گاهی عالی و گاهی خوب در مجموعه داده‌های متفاوت و مقایسه نتایج با ۹ روش معروف، بزرگ‌ترین مزیت الگوریتم پیشنهادی است. با توجه به این که الگوریتم پیشنهادی در مجموعه داده Lymphography که یک مجموعه داده با ابعاد بالا است بهترین عملکرد را ندارد، ممکن است با تغییراتی همچون تغییر روش انتخاب ویژگی[1]، نتایج بهتری حاصل شود. همچنین استفاده از روش‌های دیگر سنجش فاصله و تبدیل مقادیر توصیفی به عددی نیز می‌تواند نتیجه را بهبود بخشد.

مراجع

[1] D. Hawkins, *Identification of Outliers*, Springer Sci. Bus. Media, 1980.

[2] J. Han, M. Kamber, and J. Pei, *Data Mining: Concepts and Techniques*, 3rd Ed. Elsevier Inc, Morgan Kaufmann, 2012.

[3] M. Gupta, J. Gao, and C. C. Aggarwal, "Outlier detection for temporal data: a survey," *IEEE Trans. on Knowledge and Data Engineering*, vol. 25, no. 9, pp. 2250-2267, Dec.. 2013.

[4] M. Salehi, C. Leckie, J. C. Bezdek, and L. Fellow, "Fast memory efficient local outlier detection in data streams," *IEEE Trans. on*

1. Feature Selection

Knowledge and Data Engineering, vol. 28, no. 12, pp. 3246-3260, Aug. 2016.
[5] K. Yamanishi, J. Takeuchi, and G. Williams, "On-line unsupervised outlier detection using finite mixtures with discounting learning algorithms," *Data Mining and Knowledge Discovery*, vol. 8, no. 3, pp. 275-300, May 2004.
[6] J. Huang, Q. Zhu, L. Yang, D. Cheng, and Q. Wu, "A novel outlier cluster detection algorithm without top-n parameter," *Knowledge-Based Systems*, vol. 121, pp. 32-40, 1 Apr. 2017.
[7] G. Gan and M. K. Ng, "K-means clustering with outlier removal," *Pattern Recognition Letter*, vol. 90, pp. 8-14, Apr. 2017.
[8] M. M. Breunig, H. Kriegel, R. T. Ng, and J. Sander, "LOF: identifying density-based local outliers," in *Proc. ACM SIGMOD Int. Conf. on Management of Data*, pp. 93-104, Dalles, TX,USA, 15-18 May 2000.
[9] E. Schubert, A. Zimek, and H. Kriegel, "Local outlier detection reconsidered : a generalized view on locality with applications to spatial, video, and network outlier detection," *Data Mining and Knowledge Discovery*, vol. 28, no. 1, pp. 190-237, Jan. 2014.
[10] W. Jin, A. K. H. Tung, J. Han, and W. Wang, "Ranking outliers using symmetric neighborhood relationship," in *Proc. of the 10th Pacific-Asia Conf.on Advances in Knowledge Discovery and Data Mining*, Singapore, Singapore, 9-12 Apr. 2006.
[11] H. Kriegel, P. Kroger, E. Schubert, and A. Zimek, "LoOP: local outlier probabilities," in *Proc. of the 18th ACM Conf. on Information and Knowledge Management*, pp. 1649-1652, Hong Kong, China, 2-8 Nov. 2009.
[12] K. Zhang, M. Hutter, and H. Jin, "A new local distance-based outlier detection approach for scattered real-world data," in *Advances in Knowledge Discovery and Data Mining*, pp. 1-15, 2009.
[13] E. Schubert and H. Kriegel, "Generalized outlier detection with flexible kernel density estimates," in *Proc. of the SIAM Int. Conf. on Data Mining*, 9 pp., Philadelphia, PN, USA, 24-26 Apr. 2014.
[14] V. Hautam and K. Ismo, "Outlier detection using k-nearest neighbour graph," in *Proc. of the 17th Int. Conf. on Pattern Recognition*, *ICPR'04*, vol. 3, pp. 430-433, Cambridge, UK, 26-26 Aug. 2004.
[15] F. Angiulli and C. Pizzuti, "Fast Outlier Detection in High Dimensional Spaces," in *Principles of Data Mining and Knowledge Discovery*, pp. 15-27, 2002.
[16] S. Papadimitriou, P. B. Gibbons, C. Faloutsos, and C. Faloutsos, "LOCI: fast outlier detection using the local correlation integral," in *Proc. 19th Int. Conf. on Data Engineering*, pp. 315-326, Bangalore, India, 5-8 Mar. 2003.
[17] Q. Zhu, J. Feng, and J. Huang, "Natural neighbor: a self-adaptive neighborhood method without parameter K," *Pattern Recognition Letter*, vol. 80, no. 1, pp. 30-36, Sept. 2016.
[18] R. L. Burden, J. D. Faires, and A. M. Burden, *Numerical Analysis*, 10th Ed. Cengage Learning US, 2015.
[19] C. F. Gerald and P. O. Wheatley, *Applied Numerical Analysis*, 7th Ed. Pearson Addison-Wesley, 2006.
[20] A. Zimek, R. J. G. B. Campello, and J. Sander, "Ensembles for unsupervised outlier detection: challenges and research questions," *ACM SIGKDD Explorations Newsletter*, vol. 15, no. 1, pp. 11-22, Mar. 2014.
[21] G. O. Campos, *et al.*, "On the evaluation of unsupervised outlier detection: measures, datasets, and an empirical study," *Data Mining and Knowledge Discovery*, vol. 30, no. 4, pp. 891-927, Jul. 2016.

یحیی صالحی در سال ۱۳۹۵ دوره کارشناسی مهندسی کامپیوتر گرایش نرم‌افزار را در دانشگاه رازی با کسب امتیاز استعدادهای درخشان به اتمام رساند. وی در همان سال، با کسب امتیاز پذیرش بدون کنکور و سهمیه استعدادهای درخشان در مقطع کارشناسی ارشد دانشگاه تربیت دبیر شهید رجایی در رشته مهندسی کامپیوتر گرایش نرم‌افزار پذیرفته شد و درجه کارشناسی ارشد خود را نیز در سال ۱۳۹۷ با کسب امتیاز استعدادهای درخشان دریافت نمود. نامبرده پس از اتمام دوره کارشناسی ارشد، به عنوان پژوهشگر در پژوهشکده فناوری‌های نرم دانشگاه تهران مشغول به کار شد. زمینه‌های کاری ایشان عبارتند از: تشخیص داده‌های پرت، پیش‌پردازش داده‌ها، داده‌کاوی.

نگین دانشپور استادیار دانشکده مهندسی کامپیوتر دانشگاه تربیت دبیر شهید رجایی می‌باشد. نامبرده تحصیلات خود را در مقطع کارشناسی مهندسی کامپیوتر- سخت‌افزار در سال ۱۳۷۸ با کسب رتبه اول در دانشگاه شهید بهشتی، و کارشناسی ارشد مهندسی کامپیوتر- نرم‌افزار در سال ۱۳۸۱ در دانشگاه صنعتی امیرکبیر به پایان رسانده است، و در سال ۱۳۸۹ دکترای خود در رشته مهندسی کامپیوتر- نرم‌افزار را از دانشگاه صنعتی امیرکبیر اخذ کرده است. زمینه‌های تحقیقاتی مورد علاقه ایشان عبارتند از: پیش پردازش داده‌ها، داده‌کاوی، پایگاه داده تحلیلی و سیستم‌های تصمیم‌یار.