دانشگاه صنعتی امیرکبیر

(پلی‌تکنیک تهران)

دانشکده مهندسی کامپیوتر

گزارش پروژه‌ی پایانی درس پردازش داده‌های حجیم

عنوان‌مقاله:

کشف داده‌ی پرت در داده‌های با مقیاس بزرگ با مقادیر ویژگی نامی با استفاده از یک روش مبتنی بر تئوری اطلاعات

Information-Theoretic Outlier Detection for Large-Scale Categorical Data

دانشجو:

سید احمد نقوی نوزاد

ش-د: ۹۴۱۳۱۰۶۰

استاد درس:

دکتر امیرحائری

بهار ۱۳۹۵

بسمه تعالی

۱) مقدمه

با توجه به فراگیرشدن ابزارآلات ثبت و ضبط اطلاعات و به دنبال آن، افزایش روزافزون و پیوسته‌ی داده‌های ذخیره‌شده نیاز به آن است تا جهت پردازش و تحلیل این حجم وسیع داده‌ها متوسل به تکنیک‌های هوش محاسباتی شویم. روش‌های هوش محاسباتی موجود، قادر بوده‌اند تا توانائی و قابلیت چشم‌گیری را در زمینه‌ی تحلیل داده‌ها نظیر فرایندهای تصمیم‌گیری و پیش‌بینی در مورد داده‌های تاکنون مشاهده‌نشده از خود نشان دهند. به طور عمده، پنج دسته‌بندی بنیادی متفاوت برای انواع تحقیقات در زمینه‌ی مهندسی داده‌ها وجود دارد که شامل دسته‌بندی[۱]، خوشه‌بندی[۲]، تخمین تابع[۳]، یکپارچگی داده‌ها[۴] و در نهایت کشف انحراف یا همان داده‌ی پرت[۵] می‌باشد.

کشف داده‌های پرت را می‌توان به عنوان یک مرحله‌ی پیش‌پردازش[۶] روی داده‌ها نیز در نظر گرفت، که می‌بایست قبل از اعمال یک روش تحلیل داده‌ی پیشرفته نظیر خوشه‌بندی سلسله‌مراتبی انجام پذیرد. هدف از رویه‌ی کشف داده‌های پرت آن است که آن دسته از داده‌ها (نقاط، رخدادها، یا تراکنش‌هایی) که نسبت به مابقی داده‌ها ناسازگار بوده و به طرز قابل توجهی نسبت به آن‌ها رفتار نابهنجاری از خود نشان می‌دهند، را مکان‌یابی نموده و آن‌ها را نه تنها از مجموعه‌داده پیش از انجام هر کار دیگری حذف نمائیم، بلکه تا آن‌جا که ممکن باشد به نظم پنهانی که در فرایند تولید آن‌ها وجود دارد نیز پی ببریم. عناوین دیگری نیز در مقالات گوناگون به داده‌های پرت اطلاق گشته است، نظیر رخدادهای خیلی جدید و یا خیلی نادر[۷]، ناهنجاری‌ها[۸]، اقدامات نادرست[۹]، پدیده‌های استثنائی[۱۰] و غیره.

در این جا، هدف ما بررسی کشف داده‌های پرت در مورد داده‌های با مقادیر ویژگی نامی[۱۱] می‌باشد که در ادامه تنها با عنوان داده‌های نامی از آن‌ها اسم می‌بریم. در مورد داده‌های نامی، بزرگترین چالشی که وجود دارد آن است که چه معیار شباهت مناسبی میان داده‌ها تعریف کنیم تا به دنبال آن فواصل میان داده‌ها نیز با تقریب درستی به دست آمده و در نهایت صحت محاسبات ما نیز بالا باشد. هدف ما آن است تا یک تعریف دقیق و رسمی را برای داده‌ی پرت معرفی نموده و همین‌طور یک مدل بهینه‌سازی را جهت کشف آن معرفی نمائیم، که از یک مفهوم جدید تحت عنوان آنتروپی تام[۱۲] بهره می‌برد. آنتروپی تام از دو مفهوم آنتروپی[۱۳] و همبستگی تام[۱۴] استفاده می‌نماید و در نهایت طی یک سری محاسبات و اثبات‌های ریاضیاتی، به همان تجمیع آنتروپی روی تک‌تک ویژگی‌های نامی خلاصه می‌شود. سپس بر اساس این مدل

---

1 Classification
2 Clustering
3 Regression
4 Association
5 Deviation or outlier detection
6 Preprocessing
7 Novel or rare events
8 Anomalies
9 Vicious actions
10 Exceptional phenomena
11 Categorical
12 Holoentropy
13 Entropy
14 Total correlation

بهینه‌سازی، تابعی را جهت تعریف ضریب داده‌ی پرت[۱۵] معرفی خواهیم نمود که ورودی آن، اطلاعات خود داده به تنهائی می‌باشد و البته که این مسئله یک نوآوری خاص به حساب می‌آید. چرا که در روش‌های معمول و شناخته‌شده‌ی کشف داده‌ی پرت، علاوه بر اطلاعات خود داده، به اطلاعات سایر داده‌های موجود از جمله همسایگان آن داده نیز جهت تعریف ضریب داده‌ی پرت احتیاج می‌باشد. علاوه بر بی‌نیازبودن ضریب داده‌ی پرت مربوطه از اطلاعات سایر داده‌ها، رویه‌ی به‌روزرسانی آن نیز بسیار سریع بوده و نیازی به انجام مجدد یک سری محاسبات سنگین روی کل مجموعه‌داده نمی‌باشد. در نهایت دو الگوریتم کشف داده‌ی پرت را معرفی خواهیم نمود که تنها ورودی آن‌ها، تعداد داده‌های پرت مورد درخواست کاربر می‌باشد و نیازی به این ندارند که کاربر چگونگی تعریف داده‌ی پرت را برای آن‌ها مشخص نماید. الگوریتم اول که **ITB-SP** نام دارد، در یک رویه‌ی غیر تکراری یا به عبارتی در یک مرحله، داده‌های پرت را کشف نموده و به کاربر ارائه می‌نماید. اما الگوریتم دوم، که **ITB-SS** نام دارد، برخلاف الگوریتم اول در یک رویه‌ی تکراری و تدریجی داده‌های پرت را با دقت و ریزبینی بیشتری کشف نموده و در اختیار کاربر قرار می‌دهد. در ادامه در قسمت شرح روش و پارامترها به بیان جزئیات بیشتر در مورد این الگوریتم‌ها خواهیم پرداخت.

۲) شرح روش و پارامترها

در این قسمت در ابتدا به بیان این مسئله خواهیم پرداخت که چگونه آنتروپی و همبستگی تام در تعیین میزان پرت‌بودن هر داده و به عبارتی درستنمائی کاندیداهای داده‌ی پرت ما را یاری خواهند نمود. سپس مفهوم آنتروپی تام را که از آنتروپی و همبستگی تام بهره می‌برد، فرموله خواهیم نمود. در ادامه مطرح خواهیم نمود که سهم هر ویژگی در میزان آنتروپی تام متفاوت بوده و لذا می‌بایست به هر یک از ویژگی‌ها یک مقدار وزن خاص را بنا به سهم آن نسبت دهیم. پس از وزن‌دارکردن ویژگی‌ها، مفهوم آنتروپی تام وزن‌دار را معرفی خواهیم نمود و به دنبال آن مدل بهینه‌سازی که پیش از این قید شد و البته ضریب داده‌ی پرت مبتنی بر مدل بهینه‌سازی مربوطه را به تفصیل شرح خواهیم داد.

۲,۱) آنتروپی و همبستگی تام

مجموعه‌داده‌ی $\mathbf{X}$ را با $\mathbf{n}$ عضو به صورت $\{\boldsymbol{x_1}, \boldsymbol{x_2}, \dots, \boldsymbol{x_n}\}$ در نظر می‌گیریم، به گونه‌ای که هر $\boldsymbol{x_i}$ به ازای $\boldsymbol{1 \le i \le n}$ یک بردار از ویژگی‌های نامی $[\boldsymbol{y_1}, \boldsymbol{y_2}, \dots, \boldsymbol{y_m}]^{\boldsymbol{T}}$ می‌باشد، و هر $\boldsymbol{y_j}$ نیز دامنه‌ی مقادیر مشخصی دارد که به صورت $[\boldsymbol{y_{1,j}}, \boldsymbol{y_{2,j}}, \dots, \boldsymbol{y_{n_j,j}}]$ نشان داده می‌شود، به طوری که $\boldsymbol{1 \le j \le m}$ بوده و $\boldsymbol{n_j}$ نیز معرف تعداد مقادیر مشخص و مبیّن ویژگی $\boldsymbol{y_j}$ می‌باشد. بردار ویژگی $[\boldsymbol{y_{1,j}}, \boldsymbol{y_{2,j}}, \dots, \boldsymbol{y_{n_j,j}}]$ را می‌توان با $\mathbf{Y}$ نشان داده و $\boldsymbol{x_i}$ نیز به صورت $[\boldsymbol{x_{i,1}}, \boldsymbol{x_{i,2}}, \dots, \boldsymbol{x_{i,m}}]^{\boldsymbol{T}}$ نشان داده می‌شود. در این‌جا از علائم $\boldsymbol{H_X}()$، $\boldsymbol{I_X}()$ و $\boldsymbol{C_X}()$ برای نمایش به ترتیب معیارهای آنتروپی، اطلاعات دوطرفه[۱۶] و همبستگی تام روی مجموعه‌داده‌ی $\mathbf{X}$ استفاده خواهیم کرد. از آن‌جا که مجموعه‌ی داده‌ی مورد

[15] Outlier factor
[16] Mutual information

بررسی در همه‌جای مسئله یکی است، لذا از قید زیرنویس $\mathbf{X}$ در هر کدام از این فرمول‌ها خودداری می‌نمائیم.

فرمول آنتروپی روی کل مجموعه‌داده‌ی $\mathbf{X}$ با مجموعه‌ی ویژگی‌های $\mathbf{Y}$ بنا به قانون زنجیره‌ای[17] به صورت زیر تعریف می‌شود:

$$\begin{aligned} \boldsymbol{H(Y) = H(y_1, y_2, \dots, y_m)} &= \sum_{i=1}^{m} \boldsymbol{H(y_i|y_{i-1}, \dots y_1)} \\ &= \boldsymbol{H(y_1) + H(y_2|y_1) + \cdots + H(y_m|y_{m-1}, \dots y_1)}, \qquad (1) \end{aligned}$$

به گونه‌ای که:

$$\begin{aligned} &\boldsymbol{H(y_m|y_{m-1}, \dots y_1)} \\ &\quad = -\sum_{y_m, y_{m-1}, \dots y_1} \boldsymbol{p(y_m, y_{m-1}, \dots y_1) \log_2 p(y_m|y_{m-1}, \dots y_1)}. \end{aligned}$$

در تئوری اطلاعات، معیار آنتروپی معرف میزان عدم قطعیت با توجه به یک متغیر تصادفی خاص می‌باشد؛ به عبارتی اگر مقدار یک ویژگی نامعین باشد، مقدار آنتروپی این ویژگی بیانگر آن است که چه میزان اطلاعات نیاز است تا مقدار صحیح آن را پیش‌بینی نموده و به عبارتی تخمین بزنیم. در این‌جا باید خاطرنشان کرد که خود معیار آنتروپی نیز می‌تواند به عنوان یک مقیاس سنجش سراسری جهت کشف داده‌های پرت مورد استفاده واقع شود. به گونه‌ای که اگر در یک مجموعه داده، تعدادی از داده‌های کاندید داده‌ی پرت را حذف نموده و مجدداً آنتروپی را روی کل مجموعه‌داده حساب نمائیم، این مقدار می‌بایست کاهش چشم‌گیری داشته باشد. هر چه این کاهش بیشتر باشد، احتمال پرت‌بودن آن داده‌های منتخب نیز به مراتب بیشتر خواهد بود. اما آزمایشات انجام‌شده نشان می‌دهند که معیار آنتروپی، به تنهائی شاخص خوبی جهت کشف داده‌های پرت نمی‌باشد و معیار آنتروپی تام که در ادامه معرفی خواهد شد، به شکل مناسب‌تری عمل می‌نماید.

حال در این‌جا معیار همبستگی تام را معرفی می‌نمائیم که از معیار اطلاعات دوطرفه روی کل مجوعه‌داده بهره می‌برد و در ادامه نشان می‌دهیم که این معیار نیز می‌تواند مانند آنتروپی جهت کشف داده‌های پرت مورد استفاده واقع شود. همبستگی تام برابر مجموع اطلاعات دوطرفه‌ی مجموعه‌ی ویژگی $\boldsymbol{Y}$ می‌باشد، که در این‌جا مجموعه‌ی $\boldsymbol{Y}$ در قالب یک سری بردارهای تصادفی گسسته نمایش داده می‌شود. داریم:

$$\begin{aligned} \boldsymbol{C(Y)} &= \sum_{i=2}^{m} \sum_{\{r_1,\dots,r_i\}\subset\{1,\dots,m\}} \boldsymbol{I\left(y_{r_1}; \dots; y_{r_i}\right)} \\ &= \sum_{\{r_1,r_2\}\subset\{1,\dots,m\}} \boldsymbol{I\left(y_{r_1}; y_{r_2}\right) + \cdots + I\left(y_{r_1}; \dots; y_{r_m}\right)}, \qquad (2) \end{aligned}$$

معیار همبستگی تام، میزان وابستگی دوطرفه یا همان اطلاعات به اشتراک‌گذاشته‌شده را روی کل مجموعه‌داده نشان می‌دهد. در این‌جا لازم است بیان کنیم که هر چه همبستگی تام بین دو ویژگی (یا همان

[17] Chain rule

متغیر تصادفی) بیشتر باشد، نشان از آن دارد که تعداد زوج مرتب‌های یکسان به ازای دو ویژگی کمتر بوده و به همان اندازه تعداد زوج مرتب‌های متفاوت و به عبارتی یکتا نیز بیشتر می‌باشد. هر چه تعداد زوج مرتب‌های یکتا بیشتر باشد، میزان بی‌نظمی (آنتروپی) نیز بیشتر خواهد بود. عکس این مسئله نیز برقرار می‌باشد. در نتیجه مشاهده می‌کنیم که معیار همبستگی تام هم می‌تواند مانند معیار آنتروپی جهت کشف داده‌های پرت و به عبارتی تعیین میزان خوب‌بودن یک سری داده‌ی کاندید داده‌ی پرت به کار رود. در ادامه به معرفی معیار جدید آنتروپی تام می‌پردازیم که از هر دوی معیارهای آنتروپی و همبستگی تام استفاده می‌نماید.

۲,۲) آنتروپی تام روی بردار تصادفی Y

از آن‌جا که هر کدام از معیارهای آنتروپی و همبستگی تام به تنهائی نمی‌توانند معیار مناسبی جهت کشف داده‌های پرت باشند، لذا ناچاریم تا از معیار مناسب‌تر و دقیق‌تری تحت عنوان آنتروپی تام بهره ببریم. اگر توزیع مقادیر ویژگی‌های یک مجموعه‌داده را داشته باشیم، بنا به اثبات واتانابی[۱۸] می‌توان رابطه‌ی میان آنتروپی و همبستگی تام را به صورت زیر بیان نمود:

$$C_X(Y) = \sum_{i=1}^{m} H_X(y_i) - H_X(Y), \qquad (3)$$

با توجه به این فرمول مفهوم جدید آنتروپی تام را به صورت زیر تعریف می‌نمائیم که برابر مجموع آنتروپی و همبستگی تام روی بردار تصادفی $\mathbf{Y}$ بوده و می‌تواند به صورت مجموع آنتروپی‌ها روی تک‌تک ویژگی‌ها تعریف گردد:

$$HL_X(Y) = H_X(Y) + C_X(Y) = \sum_{i=1}^{m} H_X(y_i), \qquad (4)$$

۲,۳) وزن‌دار کردن ویژگی‌ها

همان‌طور که از فرمول معیار آنتروپی تام قابل برداشت است، این معیار به همه‌ی ویژگی‌ها به یک اندازه اهمیت داده و ارزش همگی آن‌ها را در میزان پراکندگی و بی‌نظمی در کل مجموعه‌داده یکسان فرض می‌نماید. این در حالی است که در کاربردهای واقعی هر ویژگی به یک اندازه‌ی خاص در شکل‌گیری ساختار کلی مجموعه‌داده نقش داشته و در نتیجه سهم آن در شدت آنتروپی کل متفاوت می‌باشد. حال با توجه به این‌که رویه‌ای که ما قصد پیروی از آن را جهت کشف داده‌های پرت در یک مجموعه‌داده داریم، آن است که آن دسته از داده‌ها که حذف آن‌ها سبب کاهش به مراتب بیشتر آنتروپی گردد را به عنوان کاندیدهای برتر داده‌ی پرت معرفی نمائیم، لذا می‌بایست به آن دسته از ویژگی‌ها که آنتروپی روی آن‌ها به تنهائی مقدار کمتری دارد وزن بیشتری اختصاص دهیم. علت این مسئله آن است که اگر یک ویژگی دارای مقادیر یکتای بیشتری نسبت به ویژگی دیگری باشد، آن‌گاه آنتروپی آن نیز به مراتب بیشتر خواهد بود. حال اگر یک

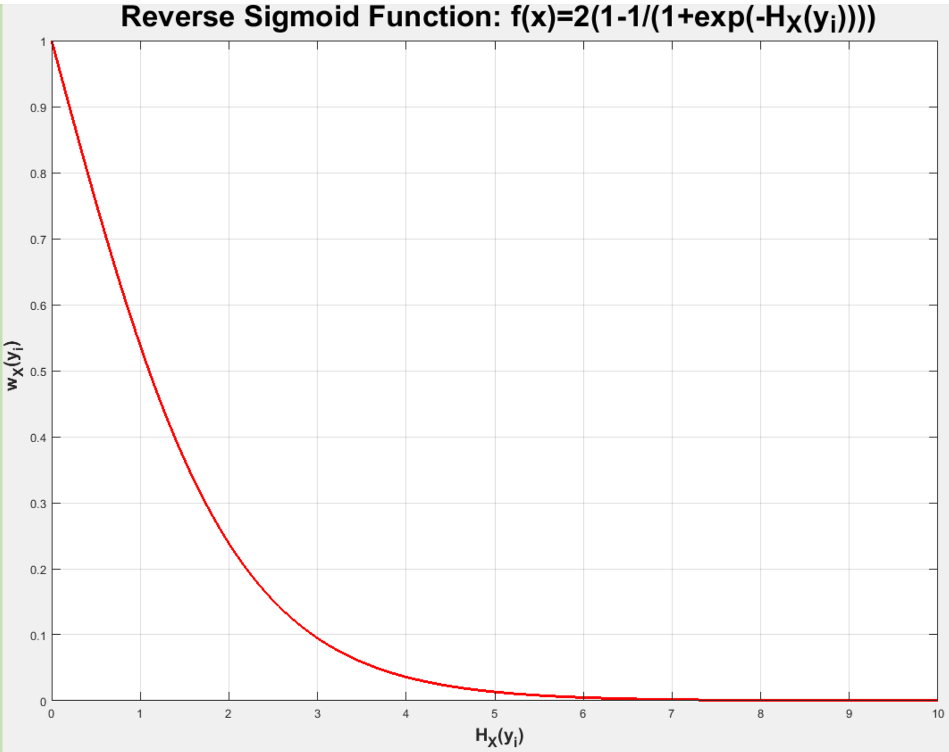
[18] Watanabe's proof

داده‌ی کاندید پرت‌بودن را از ویژگی اول حذف نمائیم، می‌بینیم که میزان آنتروپی کاهش چشم‌گیری پس از حذف ندارد، زیرا که تعداد مقادیر یکتا در آن ویژگی هنوز زیاد است. اما در مورد ویژگی دوم خواهیم دید که در صورت حذف یکی از مقادیر یکتای موجود در آن ویژگی، میزان آنتروپی به نسبت ویژگی اول به مراتب بیشتر کاهش می‌یابد. چرا که تعداد مقادیر یکتا در آن ویژگی کم می‌باشد و در واقع این همان مقادیر یکتا می‌باشند که در هر مجموعه بیشترین سهم را در آنتروپی روی آن مجموعه دارند. از آن‌چه گفته شد می‌توان فهمید که آن دسته از ویژگی‌ها که آنتروپی کمتری دارند، بیشتر ما را در یافتن داده‌های پرت یاری کرده و به سبب آن می‌بایست به آن‌ها وزن بیشتری اختصاص دهیم. چرا که با این کار، در صورت حذف آن دسته از کاندیداهای داده‌ی پرت که در آن ویژگی‌ها مقادیر یکتاتر و به اصطلاح برجسته‌تری دارند، میزان آنتروپی کاهش چشمگیرتری داشته و به دنبال آن مقصود ما که پیش از این به آن اشاره شد نیز ارضا می‌گردد.

اما برای وزن‌دارکردن هر ویژگی، در این‌جا ما از یک تابع سیگموئید معکوس استفاده می‌کنیم که با توجه به مقتضیات مسئله به صورت زیر تعریف می‌شود:

$$\boldsymbol{w_X(y_i) = 2\left(1 - \frac{1}{1 + exp\left(-H_X(y_i)\right)}\right),} \qquad (5)$$

از آن‌جا که آنتروپی همیشه مقداری بزرگ‌تر یا مساوی صفر دارد، نمودار این تابع به صورت زیر خواهد بود:



شکل ۱. نمودار تابع سیگموئید معکوس؛ از آن‌جا که آنتروپی همیشه مقادیر بزرگ‌تر یا مساوی صفر دارد، لذا دامنه‌ی تابع محدود شده است. همان‌طور که مشاهده می‌شود، این تابع به مقادیر آنتروپی بیشتر وزن کمتری اختصاص داده و مقدار وزن نیز همواره مابین صفر و یک خواهد بود.

از نمودار تابع پیداست که کاملا مطابق مقصود ما عمل کرده و هر چه مقدار آنتروپی بیشتر می‌شود، به آن وزن کمتری اختصاص می‌دهد. مقدار وزن نیز یک عدد مابین صفر و یک می‌باشد و هر چه مقادیر

آنتروپی رو به بی‌نهایت می‌رود، وزن‌های اختصاص داده‌شده به آن‌ها نیز بسیار نزدیک به هم خواهند بود. به عبارتی به ازای مقادیر آنتروپی نزدیک به صفر میزان تفاوت در وزن اختصاصی چشم‌گیرتر خواهد بود تا به ازای مقادیر آنتروپی خیلی دورتر از صفر. در ادامه خواهیم دید که چگونه همین نکته‌ی ریز ما را در مختصرسازی محاسبات سنگین یاری خواهد نمود.

۲,۴) آنتروپی تام وزن‌دار روی بردار تصادفی Y

با توجه رویه‌ی وزن‌دارکردن ویژگی‌ها که به آن اشاره گردید، معیار جدید آنتروپی تام وزن‌دار روی بردار تصادفی $\mathbf{Y}$ را به صورت زیر و برابر مجموع آنتروپی‌های وزن‌دار روی تک‌تک ویژگی‌ها تعریف می‌نمائیم:

$$W_X(Y) = \sum_{i=1}^{m} w_X(y_i) H_X(y_i), \qquad (6)$$

آزمایشات انجام‌شده نشان می‌دهند که نه تنها در مورد مجموعه‌داده‌های مصنوعی[19]، بلکه در مورد مجموعه‌داده‌های واقعی نیز معیار آنتروپی تام وزن‌دار نسبت به نسخه‌ی بی‌وزن آن، ما را در کشف داده‌های پرت بهتر یاری نموده و سبب افزایش صحت و سُقم عملیات می‌شوند.

۲,۵) یک تعریف رسمی از مسئله‌ی کشف داده‌های پرت

در این‌جا قصد داریم تا یک توجیه مبرهن و رسمی را برای علت پرت‌بودن یک زیرمجموعه از داده‌ها با استفاده از آنتروپی تام وزن‌دار ارائه نمائیم. می‌گوئیم تعداد $\mathbf{o}$ کاندید داده‌ی پرت، به عنوان بهترین زیرمجموعه معرفی خواهند شد، اگر حذف آن‌ها از مجموعه‌داده نسبت به حذف سایر زیرمجموعه‌های کاندید با همین اندازه، سبب بیشترین کاهش میزان آنتروپی تام وزن‌دار گردد. با توجه به آن‌چه گفته شد، ما با یک مسئله‌ی بهینه‌سازی روبرو هستیم که در آن می‌بایست به دنبال بهترین زیرمجموعه با اندازه‌ی $\mathbf{o}$ باشیم که حذف آن سبب بیشترین کاهش در میزان آنتروپی تام وزن‌دار گردد. این مسئله‌ی بهینه‌سازی را به صورت زیر تعریف می‌نمائیم:

$$J_X(Y, o) = W_{X \setminus Set(o)}(Y), \qquad (7)$$

که در آن تابع $\mathbf{J}$ برابر مقدار آنتروپی تام وزن‌دار مجموعه‌ی $\mathbf{X}$ پس از حذف $\mathbf{o}$ تا از کاندیداهای داده‌ی پرت می‌باشد. $Set(o)$ نیز برابر هر زیرمجموعه‌ی ممکن با اندازه‌ی $\mathbf{o}$ از اعضای مجموعه‌ی $\mathbf{X}$ می‌باشد. به عبارت بهتر می‌توان گفت که خروجی روش پیشنهادی در این مقاله به سادگی در قالب زیر قابل نمایش است:

$$Out(o) = argmin\ J_X(Y, o), \qquad (8)$$

اما از آن‌جا که هم پیداکردن تمامی زیرمجموعه‌های ممکن با اندازه‌ی $\mathbf{o}$ از مجموعه‌داده‌ی $\mathbf{X}$ شدیداً به لحاظ برنامه‌نویسی دشوار می‌باشد و هم تعریف مقدار مناسب برای $\mathbf{o}$ نیز امر ساده‌ای نخواهد بود (به طوری که حتی می‌تواند به عنوان یک مسیر جدید تحقیقاتی پیگیری شده و از همان خواص متغیر تابع بهینه‌سازی که مطرح شد بهره ببرد)، لذا ناچاریم تا به یک سری از الگوریتم‌های حریصانه جهت حل مسئله متوسل

[19] Synthetic

شویم. در ادامه نشان خواهیم داد که زمانی که تنها یکی از داده‌های کاندید پرت‌بودن از مجموعه‌داده حذف می‌گردد، می‌توان مقدار آنتروپی تام را به طرز بهینه‌ای به‌روزرسانی نمود و این مسئله در مورد حذف یک زیرمجموعه‌ی کاندید با اندازه‌ی بیشتر از یک به سادگی برقرار نخواهد بود. جالب آن است که در این به‌روزرسانی تنها به اطلاعات خود داده‌ای که حذف می‌گردد احتیاج بوده و نیازی به تخمین مجدد توزیع احتمالاتی کل مجموعه پس از حذف داده‌ی کاندید نمی‌باشد. علاوه بر این روشی را ارائه خواهیم نمود که با استفاده از آن می‌توان برای تعداد داده‌ی پرتی که کشف خواهند شد، یک حد بالا در نظر گرفته و به موجب آن فضای جستجو را کوچک‌تر خواهیم نمود تا مسئله‌ی بهینه‌سازی با سهولت بیشتری مرتفع گردد. در ادامه نیز دو الگوریتم حریصانه‌ی **ITB-SP** و **ITB-SS** را معرفی خواهیم نمود که اولی به صورت یکباره و به عبارتی با یک حرکت و دومی به صورت تدریجی و البته با دقت و صحت بیشتر، اقدام به کشف داده‌های پرت می‌نمایند.

۲,۶) یک مفهوم جدید از «ضریب داده‌ی پرت[۲۰]»

در این‌جا برای اینکه بتوانیم برای هر داده یک مقدار امتیاز یا همان ضریبِ معرفِ میزان پرت‌بودن را تعریف نمائیم، می‌بایست ابتدا تابع بهینه‌سازی $\mathbf{J}$ را که پیش‌تر معرفی شد، تحلیل کنیم. از آن‌جا که بنای تابع بهینه‌سازی گفته‌شده، میزان تفاوت در آنتروپی تام وزن‌دار قبل و بعد از حذف زیرمجموعه‌ی کاندید می‌باشد، لذا می‌بایست توزیع احتمالاتی مجموعه‌ی $\mathbf{Y}$ را پس از حذف زیرمجموعه‌ی مربوطه مجددا محاسبه نمائیم که البته امر بسیار دشوار و طاقت‌فرسائی خصوصا در مورد مجموعه‌داده‌های با مقیاس بزرگ می‌باشد. اما نکته‌ی جالب توجه آن است که می‌توان میزان تفاوت در آنتروپی تام وزن‌دار قبل و بعد از حذف را تخمین زد. این مسئله زمانی که تنها یک داده از مجموعه‌داده حذف می‌گردد، بسیار ساده‌تر شده و حتی نیازی به تخمین توزیع‌های احتمالاتی ویژگی‌ها هم نخواهد بود، و درنتیجه این موضوع می‌تواند یک راه‌حل ابتکاری[۲۱] جهت حل مسئله‌ی بهینه‌سازی (۸) ارائه نماید. در ادامه به معرفی یک مفهوم جدید تحت عنوان آنتروپی تام تفاضلی[۲۲] می‌پردازیم که در نهایت راهکاری خواهد بود تا معیار ضریب داده‌ی پرت را به صورت رسمی تعریف نمائیم.

۲,۶,۱) آنتروپی تام تفاضلی

اگر داده‌ی $\boldsymbol{x_o}$ را در نظر بگیریم، تفاوت آنتروپی تام وزن‌دار میان مجموعه‌داده‌ی $\mathbf{X}$ و مجموعه‌داده‌ی $\boldsymbol{X}\backslash\{\boldsymbol{x_o}\}$ (همان مجموعه‌داده‌ی $\mathbf{X}$ پس از حذف داده‌ی $\boldsymbol{x_o}$) را تحت عنوان آنتروپی تام تفاضلی معرفی کرده و با $\boldsymbol{h_X}(\boldsymbol{x_o})$ به صورت زیر نشان می‌دهیم:

[20] Outlier Factor (OF)
[21] Heuristic approach
[22] Differential Holoentropy

$$\begin{aligned} \boldsymbol{h}_X(\boldsymbol{x}_o) &= \boldsymbol{W}_X(\boldsymbol{Y}) - \boldsymbol{W}_{X\backslash\{x_o\}}(\boldsymbol{Y}) \\ &= \sum_{i=1}^{m} \left[\boldsymbol{w}_X(\boldsymbol{y}_i)\boldsymbol{H}_X(\boldsymbol{y}_i) - \boldsymbol{w}_{X\backslash\{x_o\}}(\boldsymbol{y}_i)\boldsymbol{H}_{X\backslash\{x_o\}}(\boldsymbol{y}_i)\right], \qquad (9) \end{aligned}$$

با توجه به نکته‌ای که در قسمت وزن‌دارکردن ویژگی‌ها به آن اشاره گردید، از آن‌جائی که وزن آنتروپی همیشه مقداری مابین صفر و یک دارد و البته به ازای مقادیر آنتروپی بزرگتر نیز، تفاوت میان وزن‌ها بسیار اندک و قابل چشم‌پوشی است، لذا می‌توان مقدار وزن را به ازای هر دوی $\boldsymbol{H}_X(\boldsymbol{y}_i)$ و $\boldsymbol{H}_{X\backslash\{x_o\}}(\boldsymbol{y}_i)$، یکسان و برابر همان $\boldsymbol{w}_X(\boldsymbol{y}_i)$ در نظر گرفت. بنابراین معادله‌ی ساده‌شده‌ی آنتروپی تام تفاضلی که در این‌جا آن را آنتروپی تام تفاضلی تخمینی می‌نامیم، به صورت زیر خواهد بود:

$$\widehat{\boldsymbol{h}}_X(\boldsymbol{x}_o) = \sum_{i=1}^{m} \boldsymbol{w}_X(\boldsymbol{y}_i)\left[\boldsymbol{H}_X(\boldsymbol{y}_i) - \boldsymbol{H}_{X\backslash\{x_o\}}(\boldsymbol{y}_i)\right], \qquad (9)$$

بنا به آزمایشات انجام‌شده مشخص شده است که تفاوت میان آنتروپی تام تفاضلی اصلی و تخمینی بسیار اندک بوده و عملکرد آن‌ها شدیداً به یکدیگر شبیه می‌باشد، و به عبارتی ضریب داده‌ی پرت اصلی و تخمینی نیز که به دنبال آن حاصل می‌گردد، با یکدیگر تفاوت چندانی ندارند. طی یک سری محاسبات ریاضیاتی می‌توان نشان داد که می‌توان آنتروپی تام تفاضلی تخمینی را به طور مستقیم و به صورت زیر محاسبه نمود:

$$\begin{aligned} \widehat{\boldsymbol{h}}_X(\boldsymbol{x}_o) &= \sum_{i=1}^{m} \boldsymbol{w}_X(\boldsymbol{y}_i)\left(\log_2 \boldsymbol{a} - \frac{\boldsymbol{a}}{\boldsymbol{b}}\log_2 \boldsymbol{b}\right) - \boldsymbol{a}\boldsymbol{W}_X(\boldsymbol{Y}) \\ &\quad + \boldsymbol{a}\sum_{i=1}^{m} \begin{cases} \boldsymbol{0}, & \boldsymbol{if}\ \boldsymbol{n}(\boldsymbol{x}_{o,i}) = \boldsymbol{1}; \\ \boldsymbol{w}_X(\boldsymbol{y}_i).\,\boldsymbol{\delta}[\boldsymbol{n}(\boldsymbol{x}_{o,i})], & \boldsymbol{else}. \end{cases}, \qquad (10) \end{aligned}$$

به طوری که $\boldsymbol{\delta}[\boldsymbol{x}] = (\boldsymbol{x}-\boldsymbol{1})\log_2(\boldsymbol{x}-\boldsymbol{1}) - \boldsymbol{x}\log_2 \boldsymbol{x}$ بوده و $\boldsymbol{x}_{o,i}$ نیز معرف مقداری است که در ویژگی $\boldsymbol{i}$-اُم داده‌ی $\boldsymbol{x}_o$ ظاهر می‌گردد. $\boldsymbol{n}(\boldsymbol{x}_{o,i})$ نیز معرف تعداد دفعاتی است که مقدار $\boldsymbol{x}_{o,i}$ در ویژگی $\boldsymbol{i}$-اُم ظاهر می‌گردد. مقادیر $\boldsymbol{b}$ و $\boldsymbol{a}$ نیز به ترتیب معکوس تعداد اعضای مجموعه‌های $\boldsymbol{X}$ و $\boldsymbol{X}\backslash\{\boldsymbol{x}_o\}$ می‌باشند، به عبارتی اگر تعداد اعضای مجموعه‌ی اصلی برابر $\boldsymbol{n}$ باشد، آن‌گاه $\boldsymbol{b} = {}^{\boldsymbol{1}}/_{\boldsymbol{n}}$ و $\boldsymbol{a} = {}^{\boldsymbol{1}}/_{(\boldsymbol{n}-\boldsymbol{1})}$ خواهد بود.

فرمول (۱۰) در واقع راهکار ما جهت به‌روزرسانی مقادیر آنتروپی و همین‌طور وزن‌های مربوطه در مراحل بعدی خواهد بود. نکته‌ی قابل توجه در مورد فرمول $\widehat{\boldsymbol{h}}_X(\boldsymbol{x}_o)$ یا همان مقدار آنتروپی تام تفاضلی به ازای داده‌ی $\boldsymbol{x}_o$، آن است که مقدار آن در دو جمله‌ی اول معادله‌ی (۱۰) تنها به مجموعه‌داده‌ی $\boldsymbol{X}$ به تنهائی وابسته می‌باشد، به این معنی که مقدار این دو جمله تنها یک بار محاسبه شده و به ازای داده‌های مختلف و البته در مراحل به‌روزرسانی بعدی دیگر نیازی به محاسبه‌ی مجدد آن‌ها نخواهد بود؛ همین‌طور مشاهده می‌کنیم که جمله‌ی سوم معادله نیز تنها به خود داده‌ی $\boldsymbol{x}_o$ وابسته می‌باشد. با توجه به خاص و

یکتابودن جمله‌ی سوم معادله‌ی (۱۰) به ازای هر کدام از داده‌های مجموعه، می‌توان آن را به عنوان معیار «ضریب داده‌ی پرت» به کار برد.

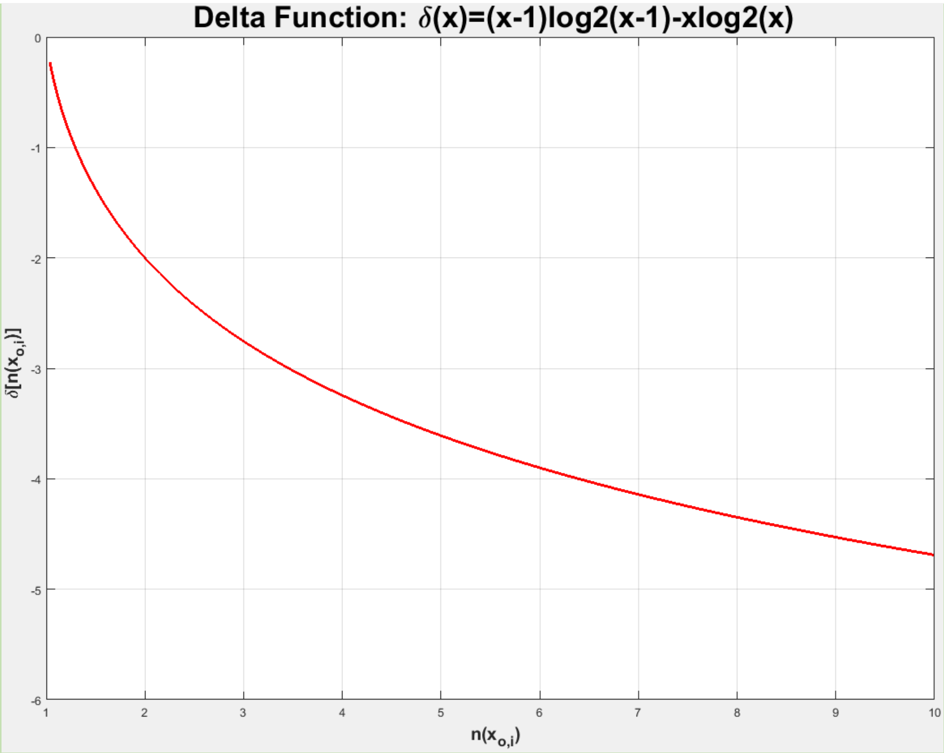
۲,۶,۲) ضریب داده‌ی پرت

با توجه به آن‌چه که در قسمت قبل قید شد، معیار ضریب داده‌ی پرت به ازای داده‌ی x_o را به صورت زیر تعریف می‌نمائیم:

$$OF(x_o) = \sum_{i=1}^{m} OF(x_{o,i}) = \sum_{i=1}^{m} \begin{cases} 0, & if\ n(x_{o,i}) = 1; \\ w_X(y_i).\,\delta[n(x_{o,i})], & else. \end{cases} \qquad (11)$$

به طوری که $OF(x_{o,i})$ برابر مقدار ضریب داده‌ی پرت برای داده‌ی x_o و به ازای ویژگی i-اُم می‌باشد. به عبارتی هر کدام از ویژگی‌ها به یک اندازه‌ی خاص در تعیین ضریب داده‌ی پرت برای یک داده نقش دارند.

ضریب داده‌ی پرت را می‌توان این‌گونه تفسیر نمود که چقدر احتمال دارد که یک داده‌ی خاص مورد بررسی، یک داده‌ی پرت باشد. هر چه این مقدار بیشتر باشد، احتمال پرت‌بودن داده‌ی مورد نظر نیز بیشتر خواهد بود. لازم به ذکر است که مقدار $OF(x_o)$ بنا به خاصیت تابع $\delta(\cdot)$ همواره مقداری کوچکتر یا مساوی صفر دارد. نمودار تابع $\delta(\cdot)$ به صورت زیر می‌باشد:



شکل ۲. نمودار تابع $\delta(x) = (x-1)\log_2(x-1) - x\log_2 x$؛ از آن‌جا که مقدار این تابع همواره منفی است، لذا مقدار ضریب داده ی پرت نیز همواره مقداری کوچکتر یا مساوی صفر دارد.

در این‌جا باید گفت که با یک سری محاسبات ریاضیاتی روی خواص OF می‌توان نشان داد که به ازای یک داده‌ی خاص، بدون درنظرگرفتن وزن ویژگی‌ها، هر چه پراکندگی مقدار آن داده به ازای یک ویژگی معین بیشتر باشد، میزان OF به ازای آن ویژگی و برای آن داده‌ی خاص کمتر خواهد بود و بالعکس. به

عبارت دیگر، اگر یک داده در ویژگی‌های خود دارای مقادیر یکتاتر و خاص‌تری به ازای هر ویژگی نسبت به سایر داده‌ها باشد، آن‌گاه احتمال پرت‌بودن آن داده به مراتب بالاتر خواهد بود.

۲,۷) به‌روزرسانی ضریب داده‌ی پرت

در این‌جا قصد بررسی حالتی را داریم که پس از کشف یک داده‌ی پرت، می‌بایست آن را از مجموعه‌داده حذف کرده و سپس به دنبال داده‌ی پرت با اولویت بیشتر باشیم. کاملا پیداست که پس از حذف یک داده، ساختار کلی مجموعه‌داده متحول شده و در نتیجه نیاز خواهد بود تا مجدداً توزیع احتمالاتی ویژگی‌ها را به دست آورده و میزان آنتروپی روی هر یک را محاسبه کنیم، و این مسئله شدیداً به لحاظ زمانی طاقت‌فرساست. لذا همان‌طور که در مورد آنتروپی تام تفاضلی توانستیم حجم محاسبات را کاهش دهیم، در این‌جا نیز به همان شکل عمل کرده و مقدار آنتروپی تام تفاضلی بی‌وزن $\boldsymbol{HL_X(Y) - HL_{X\setminus\{x_o\}}(Y)}$ را به صورت زیر بازنویسی می‌نمائیم:

$$\begin{aligned}\boldsymbol{HL_X(Y) - HL_{X\setminus\{x_o\}}(Y)} &\\ = \mathbf{m}\left[\left(\frac{\boldsymbol{a}}{\boldsymbol{b}} - \boldsymbol{a}\right)\log_2 \boldsymbol{a} - (\boldsymbol{b}+\mathbf{1})\log_2 \boldsymbol{b}\right] - \boldsymbol{bHL_X(Y)} &\\ + \boldsymbol{a}\sum_{i=1}^{m}\begin{cases}\mathbf{0}, & \boldsymbol{if\ n(x_{o,i}) = 1;} \\ \boldsymbol{\delta[n(x_{o,i})]}, & \boldsymbol{else.}\end{cases} \qquad (\mathbf{12})\end{aligned}$$

در نتیجه می‌توان فرمول ساده‌شده‌ی آنتروپی تام به‌روزشده را به صورت زیر بازنویسی نمائیم:

$$\begin{aligned}\boldsymbol{HL_{X\setminus\{x_o\}}(Y)} = (\mathbf{1}+\boldsymbol{b})\boldsymbol{HL_X(Y)} - \mathbf{m}\left[\left(\frac{\boldsymbol{a}}{\boldsymbol{b}} - \boldsymbol{a}\right)\log_2 \boldsymbol{a} - (\boldsymbol{b}+\mathbf{1})\log_2 \boldsymbol{b}\right] &\\ - \boldsymbol{a}\sum_{i=1}^{m}\begin{cases}\mathbf{0}, & \boldsymbol{if\ n(x_{o,i}) = 1;} \\ \boldsymbol{\delta[n(x_{o,i})]}, & \boldsymbol{else.}\end{cases} \qquad (\mathbf{13})\end{aligned}$$

با استفاده از (۱۳) می‌توانیم مقدار آنتروپی به‌روزشده را به ازای تک‌تک ویژگی‌ها محاسبه نمائیم. داریم:

$$\begin{aligned}\boldsymbol{H_{X\setminus\{x_o\}}(y_i)} = (\mathbf{1}+\boldsymbol{b})\boldsymbol{H_X(y_i)} - \left[\left(\frac{\boldsymbol{a}}{\boldsymbol{b}} - \boldsymbol{a}\right)\log_2 \boldsymbol{a} - (\boldsymbol{b}+\mathbf{1})\log_2 \boldsymbol{b}\right] &\\ - \boldsymbol{a}\begin{cases}\mathbf{0}, & \boldsymbol{if\ n(x_{o,i}) = 1;} \\ \boldsymbol{\delta[n(x_{o,i})]}, & \boldsymbol{else.}\end{cases} \qquad (\mathbf{14})\end{aligned}$$

پس از محاسبه‌ی مجدد آنتروپی به ازای هر کدام از ویژگی‌ها، می‌توانیم وزن مربوط به هر یک را نیز با استفاده از (۵) مجدداً محاسبه نموده و در نهایت با استفاده از (۱۱) ضریب داده‌ی پرت را به‌روزرسانی نمائیم.

۲,۸) تعیین یک حد بالا برای تعداد داده‌های پرت

با توجه به این‌که در روش‌های یادگیری بدون نظارت، اکثریت داده‌ها نرمال فرض می‌شوند، لذا ناچاریم تا برای تعداد داده‌های غیرنرمال یا پرتی که در مجموعه‌داده حضور دارند، یک حد بالا تعیین نمائیم. در

این‌جا سه مفهوم جدید را بدین ترتیب معرفی می‌نمائیم: حد بالای تعداد داده‌های پرت (**UO**[23])، مجموعه‌ی کاندید داده‌های پرت (**AS**[24])، و مجموعه داده‌های نرمال (**NS**[25]).

سه مفهوم جدید مطرح‌شده در بالا بنا به این دیدگاه حاصل شده‌اند که حذف داده‌های پرت از مجموعه‌داده سبب کاهش آنتروپی تام وزن‌دار $W_X(Y)$ و بیشتر خالص‌شدن کل مجموعه‌داده می‌شود. خلاف این مسئله در مورد داده‌های نرمال برقرار می‌باشد، بدین معنی که حذف آن‌ها سبب افزایش $W_X(Y)$ خواهد شد. بنابراین می‌توان با استفاده از علامت آنتروپی تام تفاضلی $\widehat{h}_X(x_o)$ به ازای هر داده‌ی x_o، به ماهیت نرمال یا پرت‌بودن آن پی برد. در نتیجه داریم:

$$NS = \{x_i, \widehat{h}(x_i) \le 0\},$$

$$AS = \{x_i, \widehat{h}(x_i) > 0\},$$

$$UO = N(AS) = \sum_{i=1}^{n} \left(\widehat{h}(x_i) > 0\right), \qquad (15)$$

در این‌جا باید خاطرنشان کرد که حداکثر داده‌های پرتی که توسط الگوریتم‌های پیشنهادی در این گزارش قابل کشف‌شدن می‌باشند، برابر اعضای مجموعه‌ی **AS** می‌باشند که تعداد آن‌ها برابر **UO** می‌باشد. حتی در حالتی که قصد پیداکردن داده‌های پرت را به صورت مرحله به مرحله داریم، باز هم فضای جستجو همان مجموعه‌ی **AS** خواهد بود و این مسئله قابل اثبات است که پس از حذف یک داده‌ی پرت از مجموعه‌داده و به تبع آن درهم‌ریخته‌شدن نظم سراسری مجموعه‌داده، هیچ کدام از داده‌های نرمال مجموعه، از حالت نرمال خارج نشده و اصطلاحاً مشکوک به پرت‌بودن نخواهند شد.

۲,۹) معرفی الگوریتم‌های ITB-SP و ITB-SS

در این‌جا قصد داریم تا با توجه به ماهیت ضریب داده‌ی پرت که پیش از این به آن اشاره گردید، دو الگوریتم حریصانه را جهت کشف داده‌های پرت در مجموعه‌داده‌های با ویژگی‌های نامی استخراج نمائیم. اولین الگوریتم **ITB-SP (Information-Theory-Based Single-Pass)** نام دارد که در آن مقدار ضریب داده‌ی پرت به ازای تمامی داده‌ها تنها یک بار محاسبه گشته و سپس تعداد **o** داده‌ی پرت مورد درخواست کاربر با بالاترین میزان **OF** به عنوان خروجی ارائه می‌گردد. دومین الگوریتم نیز **ITB-SS (Information-Theory-Based Step-by-Step)** نام دارد که در یک رویه‌ی گام‌به‌گام اقدام به کشف داده‌های پرت می‌نماید. به این ترتیب که ابتدا با استفاده از مقدار آنتروپی تام تفاضلی $\widehat{h}_X(x_o)$ به ازای هر داده‌ی x_o، مجموعه‌ی کاندید داده‌ی پرت یا همان **AS** را پیدا نموده و سپس داده‌ای از این مجموعه که بیشترین مقدار **OF** را دارد، به عنوان اولین داده‌ی پرت معرفی می‌نمائیم. سپس داده‌ی مربوطه را از مجموعه‌ی **AS** حذف نموده و مقدار **OF** را به ازای تمامی داده‌های باقیمانده‌ی **AS** به روزرسانی می‌کنیم و همین رویه را آن‌قدر تکرار خواهیم کرد تا داده‌های پرت به تعداد درخواستی کاربر کشف گردند.

---

23 Upper Bound on Outliers
24 Anomaly Candidate Set
25 Normal Object Set

لازم به ذکر است که هر دوی این الگوریتم‌ها، داده‌های پرت را درون مجموعه‌ی **AS** جستجو می‌کنند و به عبارتی فضای جستجو همواره محدود به مجموعه‌ی **AS** خواهد بود. این مسئله در مورد **ITB-SP** چندان تفاوتی نمی‌کند، زیرا که این الگوریتم، داده‌های پرت را در همان اولین مرحله و پس مرتب‌سازی ضرایب داده‌ی پرت پیدا می‌کند. اما در مورد **ITB-SS** این مسئله متفاوت‌تر می‌باشد، زیرا پس از هر مرحله کشف، می‌بایست یک سری محاسبات مجدداً انجام شود، اما با این حال اثبات می‌شود که فضای جستجو باز هم محدود به همان **AS** خواهد بود.

فرض ما در این گزارش آن است که کاربر مربوطه تعداد داده‌های پرت درخواستی خویش را ارائه می‌دهد و این تعداد که با **o** نشان داده می‌شود، همواره از **UO** یا همان حد بالای تعداد داده‌های پرت کمتر خواهد بود. ولی در صورت بیشتربودن هم تنها با یک تغییر جزئی می‌توان این تعداد درخواستی را به همان اندازه‌ی **UO** محدود نمود. اما نکته‌ی قابل توجه آن است که همواره تعداد معقول و منطقی داده‌های پرت بسیار کمتر از حد **UO** می‌باشد و به عبارتی این حد بالا، حد غائی داده‌های پرت ممکن موجود در مجموعه‌داده می‌باشد.

در این‌جا الگوریتم **ITB-SP** را به صورت زیر ارائه می‌نمائیم:

Algorithm 1. ITB-SP single pass
Input: dataset X and number of outliers requested o
output: outlier set OS
Compute $w_X(y_i)$ for $(1 \leq i \leq m)$ by (3-2)
Set OS = φ
for i = 1 to n do
    Compute OF(x_i) and obtain AS
end for
if o > UO then
    o = UO
else
    Build OS by searching for the o objects with greatest OF(x_i) in AS using heapsort
end if

لازم به ذکر است که پیچیدگی زمانی الگوریتم **ITB-SP** برابر با **O(nm)** می‌باشد که در آن **n** برابر تعداد داده‌های مجموعه‌داده و **m** نیز برابر تعداد ویژگی‌ها می‌باشد.

در این‌جا هم الگوریتم **ITB-SS** را به قرار زیر ارائه می‌دهیم:

Algorithm 2. ITB-SS Step-by-Step
Input: dataset X and number of outliers requested o
output: outlier set OS
Set OS = φ
Compute $w_X(y_i)$ for $(1 \leq i \leq m)$ by (3-2)
for i = 1 to n do
    Compute OF(x_i) and obtain AS
end for

```
if o > UO then
    o = UO
else
    for i = 1 to o do
        Search for the object with greatest OF(x_o) from AS
        Add x_o to OS and remove it from AS
        Update all the OF(x) of AS
    end for
end if
```

پیچیدگی زمانی الگوریتم **ITB-SS** نیز برابر با **O(om*(UO))** می‌باشد، که معمولا بیشتر از پیچیدگی زمانی الگوریتم اول یعنی **ITB-SP** بوده و علت آن نیز انجام مرحله‌به‌مرحله‌ی کشف داده‌های پرت می‌باشد. اما آزمایشات انجام‌شده نشان از آن دارند که این مقدار اختلاف در زمان محاسبات، ارزش دقت بالاتر در کشف داده‌های پرت را دارد.

۳) آزمایشات انجام‌شده

در این قسمت به انجام دو آزمایش خواهیم پرداخت که میزان اثرگذاری الگوریتم‌های معرفی‌شده را بررسی خواهند کرد. در **آزمایش اول**، از یک مجموعه‌داده‌ی نسبتا کوچک با نام **"soybean data"** استفاده می‌کنیم که از ۴۷ داده با ۳۵ ویژگی تشکیل شده است. از آن‌جا که داده‌های این مجموعه به لحاظ نرمال یا پرت‌بودن برچسب نخورده‌اند، لذا منطقی خواهد بود که داده‌های متعلق به کوچک‌ترین کلاس را به عنوان داده‌ی پرت در نظر بگیریم. به همین منظور در مورد این مجموعه‌داده، داده‌های کلاس ۲ را به عنوان داده‌های پرت برچسب می‌زنیم. انتظار ما این خواهد بود که الگوریتم‌های پیشنهادی بتوانند داده‌های همین کلاس کوچک را به عنوان داده‌های پرت شناسائی نمایند. در **آزمایش دوم**، الگوریتم‌های پیشنهادی را بر روی سه عدد از مجموعه‌داده‌های واقعی با نام‌های **autos**، **wbc** و **web-ad** آزمایش خواهیم نمود. در مورد همه‌ی این مجموعه‌داده‌های واقعی نیز به دلیل برچسب‌نخورده‌بودن به لحاظ نرمال یا پرت‌بودن، مانند آزمایش اول عمل نموده و داده‌های متعلق به کوچک‌ترین کلاس را به عنوان داده‌ی پرت برچسب خواهیم زد. لذا در مورد مجموعه‌داده‌ی **autos**، از آن‌جا که داده‌های آن هیچ برچسبی به لحاظ کلاس نخورده‌اند، به همین دلیل ما ویژگی ۴ یعنی نوع سوخت مصرفی (**diesel=1, gas=2**) را به عنوان برچسب داده‌ها انتخاب نمودیم و از آن‌جا که تعداد داده‌های با برچسب ۱ بسیار کمتر بودند، آن‌ها را به عنوان داده‌های پرت برچسب زدیم؛ در مورد مجموعه‌داده‌ی **wbc** نیز کلاس **malignant=4** را که تعداد کمتری از داده‌ها به آن تعلق داشتند، به عنوان داده‌های پرت برچسب زدیم؛ و در نهایت در مورد مجموعه‌داده‌ی **web-ad** نیز که از جمله مجموعه‌داده‌های معیار (یا به اصطلاح **benchmark**) می‌باشد، داده‌های کلاس **ad.=1** را که کمتر از ۱۴ درصد داده‌ها را به خود اختصاص داده‌اند، به عنوان داده‌های پرت برچسب‌گذاری نمودیم. در مورد آزمایش دوم، جهت ارزیابی الگوریتم‌های پیشنهادی از معیار ارزیابی معتبر **AUC** (یا همان **Area Under ROC Curve**) استفاده نموده‌ایم و خوشبختانه نتایج حاصله نیز بسیار امیدبخش می‌باشند. در

این‌جا علاوه بر نتایج حاصله از پیاده‌سازی، نتایج قیدشده در مقاله را نیز قید می‌نمائیم و خواهیم دید که هر دوی این نتایج بسیار به یکدیگر شبیه می‌باشند.

نتایج حاصل از آزمایش اول به همراه نتایج قیدشده در اصل مقاله به قرار زیر می‌باشند:

نتایج حاصل از پیاده‌سازی

```
--------------------------------------------------------------------------------------------------------
o   ITB-SP                                        Jx(Y,o)   ITB-SS                                      Jx(Y,o)
--------------------------------------------------------------------------------------------------------
1:  11                                            10.489    11                                          10.489
2:  11  18                                        10.464    11  18                                      10.464
3:  11  18  16                                    10.426    11  18  15                                  10.445
4:  11  18  16  15                                10.396    11  18  15  16                              10.396
5:  11  18  16  15  20                            10.348    11  18  15  16  20                          10.348
6:  11  18  16  15  20  29                        10.284    11  18  15  16  20  19                      10.288
7:  11  18  16  15  20  29  19                    10.226    11  18  15  16  20  19  13                  10.192
8:  11  18  16  15  20  29  19  13                10.131    11  18  15  16  20  19  13  14              10.057
9:  11  18  16  15  20  29  19  13  14            9.997     11  18  15  16  20  19  13  14  29          9.997
10: 11  18  16  15  20  29  19  13  14  12        9.803     11  18  15  16  20  19  13  14  29  26      9.942
--------------------------------------------------------------------------------------------------------
```

نتایج قیدشده در مقاله

o	ITB-SP	$J_{\mathcal{X}}(\mathcal{Y},o)$	ITB-SS	$J_{\mathcal{X}}(\mathcal{Y},o)$
1:	11	9.686	11	9.686
2:	11,18	9.687	11,18	9.687
3:	11,**15**,18	**9.687**	11,**15**,18	**9.687**
4:	11,15,16,18	9.671	11,15,16,18	9.671
5:	11,15,16,18,20	9.659	11,15,16,18,20	9.659
6:	11,15,**16**,18,19,20	**9.646**	11,13,15,18,19,20	9.642
7:	11,13,15,16,18,19,20	9.585	11,13,15,16,18,19,20	9.585
8:	11,13,**14**,15,16,18,19,20	**9.541**	11,13,15,16,17,18,19,20	9.537
9:	11,13,14,15,16,18,19,20,**29**	**9.493**	11,13,14,15,16,17,18,19,20	9.468
10:	11,12,13,14,15,16,18,19,20,**29**	**9.419**	11,12,13,14,15,16,17,18,19,20	9.334

جدول ۱. نتایج حاصل از پیاده‌سازی به همراه نتایج مندرج در مقاله برای آزمایش اول؛ همانطور که مشاهده می‌شود، در مورد نتایج حاصل از پیاده‌سازی به جز در موارد اندکی تمامی داده‌های کشف‌شده به عنوان داده‌ی پرت، بر طبق انتظار همان داده‌های متعلق به کلاس ۲ می‌باشند.

نتایج حاصل از آزمایش دوم به همراه نتایج مندرج در مقاله نیز به صورت زیر می‌باشند:

Data Set		#n	#m	#o	#UO	unweighted ITB-SP	ITB-SP	unweighted ITB-SS	ITB-SS
autos	Imp. Results	205	25	max	16	0.843	0.805	0.843	0.805
	Paper Results	133	26	12	58	0.786	0.762	0.776	0.757
breast-w	Imp. Results	699	9	max	280	0.983	0.979	0.983	0.979
	Paper Results	699	10	241	281	0.984	0.985	0.990	0.992
web-ad	Imp. Results	3279	1558	max	1487	0.707	0.702	0.706	0.701
	Paper Results	3279	1558	458	736	0.705	0.701	0.735	0.735

جدول ۲. نتایج حاصل از پیاده‌سازی به همراه نتایج مندرج در مقاله برای آزمایش دوم؛ لازم به ذکر است که در مورد مجموعه‌داده‌ی autos موفق به یافتن مجموعه‌داده‌ی اصلی نشدیم و ناچاراً از یک مجموعه‌داده‌ی دیگر استفاده نمودیم، ولی در عین حال نتایج حاصله امیدبخش می‌باشند.

۳)مراجع

[1] Wu, Shu, and Shengrui Wang. "Information-theoretic outlier detection for large-scale categorical data." *IEEE transactions on knowledge and data engineering* 25.3 (2013): 589-602.