دانشگاه صنعتی امیرکبیر
(پلی تکنیک تهران)

دانشکده مهندسی کامپیوتر

تمرین سوم درس داده کاوی

نگارش
مهدیه سادات بنیس

استاد درس
دکتر احسان ناظرفرد

نیم سال دوم ۱۴۰۱

• بخش تئوری:

سوال اول

یک مجموعه داده از حیوانات مختلف به همراه ویژگی‌هایشان را در اختیار داریم، میخواهیم با استفاده از روش‌های خوشه‌بندی میزان شباهت هر دو حیوان به هم را از ۱ (کمترین) تا ۳ (بیشترین) مشخص نماییم. برای مثال میزان شباهت شیر و پلنگ ۳ و میزان شباهت شیر و گوسفند ۱ میتواند باشد. الگوریتمی ارائه دهید که این امر را به صورت غیرنظارت‌شده ممکن سازد.

۱- پیش پردازش داده ها:

هر ستون یا ردیف نامربوط را حذف کنید.

داده ها را عادی کنید تا مقیاس یکسانی داشته باشند.

در صورت لزوم متغیرهای دسته بندی را به مقادیر عددی تبدیل کنید.

۲- یک الگوریتم خوشه بندی را انتخاب کنید:

الگوریتم‌های خوشه‌بندی زیادی مانند K-means، خوشه‌بندی سلسله مراتبی و DBSCAN وجود دارد. یکی را انتخاب کنید که برای داده ها و مشکل شما مناسب تر است.

۳- تعداد خوشه ها را تعیین کنید:

از یک رویکرد سلسله مراتبی یا یک روش اعتبارسنجی خوشه‌بندی مانند روش زانویی برای تعیین تعداد بهینه خوشه‌ها استفاده کنید.

۴- انجام خوشه بندی:

الگوریتم خوشه بندی انتخابی را روی داده های از پیش پردازش شده با تعداد خوشه های تعیین شده اعمال کنید.

هر حیوان بر اساس ویژگی های خود به یک خوشه اختصاص داده می شود.

۵- میزان تشابه را تعیین کنید:

فاصله بین هر جفت از حیوانات در یک خوشه را محاسبه کنید.

برای هر جفت بر اساس فاصله، درجه ای از شباهت تعیین کنید. در جفت هایی که در یک خوشه قرار دارند میتوان درجه شباهت ۳ در نظر گرفت و برای ان هایی که در در دو خوشه مختلف قرار دارند ۲ یا ۳ باتوجه به میزان فاصله ای که دارند. به عنوان مثال، اگر فاصله شیر و پلنگ کم باشد، درجه تشابه آنها ممکن است ۳ باشد، در حالی که اگر فاصله بین شیر و گوسفند زیاد باشد، درجه تشابه آنها ممکن است ۱ باشد.

سوال دوم

میدانیم که در الگوریتم خوشه‌بندی برای تابع مجاورت موارد مختلفی را میتوان استفاده کرد، در موارد زیر اثبات نمایید که نقطه نهایی که به عنوان مرکز انتخاب میشود چه نقطه‌ای است. (در رابطه زیر D مجموعه تمامی نقاط داده و C مجموعه تمامی مراکز خوشه ها میباشد.)

$$\sum_{d \in D} \sum_{c \in C} f(d, c)$$

- نرم ۱ $f(d, c) = |d - c|$

$$\text{argmin} \sum_{d \in D} \sum_{c \in C} |d - c|$$

همان طور که میبینیم میخواهیم مجموع قدر مطلق ها کمترین شود که یعنی در واقع برای هر خوشه میانه را پیدا کنیم. نقطه ای که مجموع فاصله آن از بقیه کمترین باشد.

- نرم ۲ $f(d, c) = \|d - c\|_2^2$

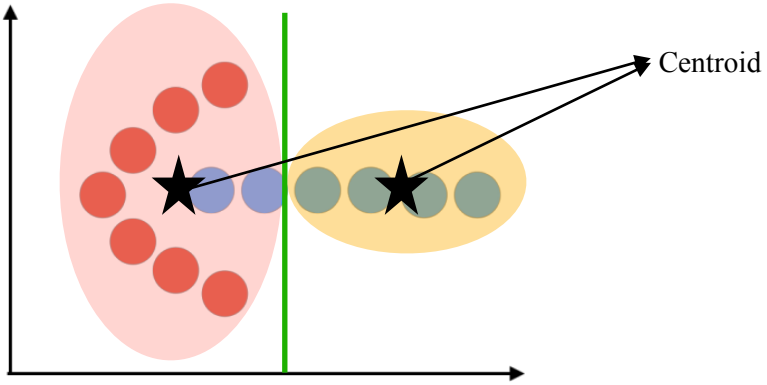
$$\text{argmin} \sum_{d \in D} \sum_{c \in C} |d - c|^2$$

از این عبارت اگر مشتق گرفته و برابر صفر قرار دهیم نقاط مینیمم هر خوشه یعنی مرکز ها به صورت زیر بدست خواهند آمد:

$$\sum_{d \in D} \sum_{c \in C} 2|d - c| = 0 \quad \rightarrow \quad c = \frac{1}{2} \sum_{d \in D} d$$

سوال سوم

الف) فرض کنید داده‌های زیر را میخواهیم به ۲دسته مختلف دسته‌بندی کنیم، پیشبینی شما از اجرا الگوریتم k-means را از داده‌های زیر بیان کنید و علت این پیشبینی را هم ذکر نمایید.



همان طور که مشاهده میکنیم با توجه به روش الگوریتم k-means داده به صورت فوق خوشه بندی میشوند. و با توجه به اینکه به هر کدام از centriod ها نزدیک تر باشند در آن خوشه قرار میگیرند به دو خوشه تفکیک شده و centriod ها دیگر مکانشان عوض نمیشود.

k-means برای کشف خوشه‌هایی با اشکال غیر محدب مناسب نیست زیرا فرض می‌کند که خوشه‌ها کروی هستند و واریانس برابر دارند، که ممکن است برای همه مجموعه‌های داده درست نباشد. در داده‌های شکل غیر محدب، خوشه‌ها ممکن است اشکال پیچیده و نامنظمی داشته باشند که نتوان آنها را با یک مرکز و واریانس نشان داد. زیرا که با توجهی با روشی مانند این می باشد که با یک خط خوشه ها را از هم تفکیک میکند حال با توجه به داده که ما داریم میبینیم که دو دسته آن با خط قابل تفکیک نیستند و الگوریتم k-means به این صورت عمل خوشه بندی را انجام میدهد.

ب) آیا استفاده از روش DBSCAN میتواند برای دادههای بالا عملکرد بهتری داشته باشد؟ علت را توضیح دهید.

DBSCAN به چند دلیل برای کشف خوشه‌هایی با اشکال غیر محدب مناسب است:

- DBSCAN یک الگوریتم خوشه‌بندی مبتنی بر چگالی است که نقاط داده‌ای را که در نواحی متراکم به یکدیگر نزدیک هستند، جمع‌آوری می‌کند، در حالی که مناطق پراکنده را کنار می‌گذارد. این آن را برای داده های شکل غیر محدب مناسب می کند، جایی که خوشه ها ممکن است اشکال پیچیده و نامنظمی داشته باشند که نمی توانند با یک مرکز و واریانس منفرد نمایش داده شوند.

- هیچ فرضی در مورد شکل خوشه وجود ندارد: DBSCAN هیچ فرضی در مورد شکل یا اندازه خوشه ها نمی کند، که آن را نسبت به الگوریتم های خوشه بندی مبتنی بر مرکز مانند k-means انعطاف پذیرتر و قوی تر می کند.

- استحکام در برابر نویز و نقاط پرت: DBSCAN در برابر نویز و نقاط پرت مقاوم است، زیرا می تواند آنها را به عنوان خوشه های جداگانه یا نقاط نویز شناسایی کند. این آن را برای داده های شکل غیر محدب که ممکن است حاوی نویز یا نقاط پرت باشد مناسب می کند.

- تعیین خودکار تعداد خوشه ها: DBSCAN نیازی به دانستن تعداد خوشه ها از قبل ندارد، زیرا می تواند به طور خودکار تعداد خوشه ها را بر اساس تراکم نقاط داده شناسایی کند. این باعث می شود آن را برای تجزیه و تحلیل داده های اکتشافی و زمانی که تعداد خوشه ها از داده ها مشخص نباشد، مناسب می کند.

- مقیاس پذیری: DBSCAN از نظر محاسباتی برای مجموعه داده های بزرگ کارآمد است، زیرا فقط به محاسبه فاصله بین نقاط داده نزدیک نیاز دارد، نه بین تمام جفت نقاط داده، که آن را سریعتر و مقیاس پذیرتر از k-means برای مجموعه داده های بزرگ می کند.

به طور کلی، DBSCAN یک الگوریتم خوشه‌بندی قدرتمند و انعطاف‌پذیر است که می‌تواند داده‌های شکل غیر محدب را مدیریت کند و برای طیف وسیعی از کاربردها مناسب است.

ج) توضیح دهید در چه زمانی خوشه‌بندی بر مبنای چگالی عملکرد مناسبی نخواهد داشت؟ مثال بزنید.

در حالی که DBSCAN یک الگوریتم خوشه بندی قدرتمند و انعطاف پذیر است که برای بسیاری از برنامه ها مناسب است، مواردی وجود دارد که ممکن است بهترین انتخاب نباشد:

- هنگامی که داده ها چگالی های متفاوتی دارند: DBSCAN طوری طراحی شده است که روی داده هایی با چگالی همگن به خوبی کار کند، جایی که هر خوشه چگالی مشابهی دارد. ممکن است روی داده‌هایی با چگالی‌های متفاوت عملکرد خوبی نداشته باشد، جایی که برخی از خوشه‌ها نسبت به بقیه چگالی‌تر یا کم‌تر هستند.

- وقتی داده‌ها ابعاد بالایی دارند: DBSCAN ممکن است روی داده‌های با ابعاد بالا عملکرد خوبی نداشته باشد، زیرا نفرین ابعاد می‌تواند منجر به پراکندگی داده‌ها و در نتیجه ایجاد خوشه‌های با چگالی کم شود. در داده های با ابعاد بالا، ممکن است قبل از اعمال DBSCAN از تکنیک های کاهش ابعاد استفاده شود.

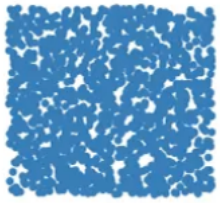
- وقتی خوشه‌ها شکل‌ها و اندازه‌های متفاوتی دارند: در حالی که DBSCAN نسبت به الگوریتم‌های خوشه‌بندی مبتنی بر مرکز انعطاف‌پذیرتر است، ممکن است هنوز برای شناسایی خوشه‌هایی با اشکال و اندازه‌های متفاوت مشکل داشته باشد. در چنین مواردی، شاید بهتر باشد از الگوریتم‌های خوشه‌بندی دیگری استفاده شود که می‌توانند اشکال خوشه‌ای پیچیده‌تر را مدیریت کنند، مانند خوشه‌بندی طیفی یا خوشه‌بندی سلسله مراتبی.

- وقتی داده ها حاوی نقاط نویز هستند: در حالی که DBSCAN در برابر نویز قوی است، اگر داده ها دارای تعداد زیادی نقاط نویز باشند، ممکن است عملکرد خوبی نداشته باشد. در چنین مواردی، ممکن است بهتر باشد از سایر الگوریتم‌های خوشه‌بندی استفاده شود که به‌طور خاص برای مدیریت داده‌های پر سر و صدا طراحی شده‌اند، مانند خوشه‌بندی قوی یا خوشه‌بندی با تشخیص پرت.

زمانی که داده ها به شدت skewed هستند: DBSCAN فرض می کند که داده ها توزیع تقریباً یکنواختی دارند و ممکن است روی داده های skewed عملکرد خوبی نداشته باشند. در چنین مواردی، ممکن است بهتر باشد از الگوریتم‌های خوشه‌بندی دیگری مانند مدل‌سازی مخلوط استفاده کنید که می‌توانند داده‌های skewed را مدیریت کنند.

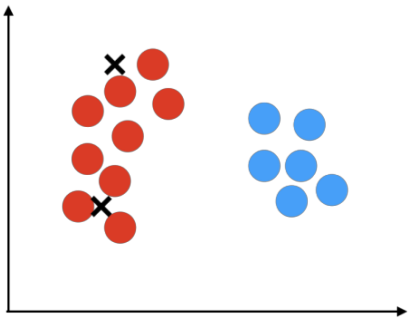
مثال:

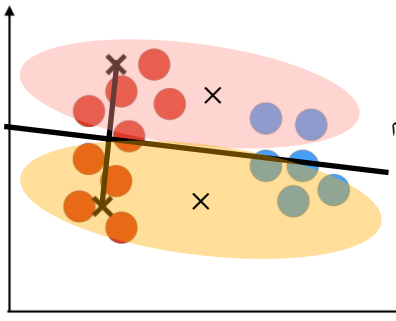
فرض کنید یک مجموعه داده با ابعاد بالا با چگالی های متفاوت و اشکال خوشه ای پیچیده داریم. داده ها شامل ۱۰۰۰ نقطه داده است که هر کدام دارای ۱۰۰ ویژگی است. داده ها شامل سه خوشه، با چگالی و شکل های مختلف است: یک خوشه کروی، یکی کشیده، و یکی نامنظم است. داده ها همچنین حاوی مقدار قابل توجهی از نقاط نویز هستند. یا برای مثال داده خیلی چگالی بالا داشته باشند مانند:



سوال چهارم

الف) نتیجه اعمال الگوریتم k-means را بر روی دادههای زیر مشخص کنید. (ضرب در بیانگر مراکز اولیه است)





همان طور که مشاهده میکنیم دو مرکز را به هم وصل کرده و عمود منصف آن ها را میکشیم و داده ها به خوشه های دو طرف آن تقسیم میشوند و میانگین گیری هر خوشه و به روز کردن مرکز ها مجدد به همان دو قسمت تقسیم میشوند و الگوریتم نمیتواند عملکرد خوبی داشته باشد و خوشه های آبی و قرمز را تشخیص دهد.

ب) برای حل مشکل بالا از راهکارهای گوناگونی استفاده میشود در رابطه با هر یک از این راهکارها را تحقیق کرده و مزایا و معایب آنها را توضیح دهید

- استفاده از medoid به جای median

استفاده از medoid به جای median:

در الگوریتم سنتی k-means، مرکز یک خوشه به عنوان میانگین تمام نقاط آن خوشه محاسبه می شود. با این حال، استفاده از medoid به جای میانه یک رویکرد جایگزین است. مدوید نشان‌دهنده مرکزی‌ترین نقطه در یک خوشه است، که نقطه‌ای است که تفاوت میانگین را با سایر نقاط خوشه به حداقل می‌رساند. این روش دارای مزایا و معایب زیر است:

مزایای:

- استحکام نسبت به نقاط پرت: بر خلاف میانگین، که می تواند به شدت تحت تأثیر عوامل پرت باشد، مدوید کمتر تحت تأثیر مقادیر شدید قرار می گیرد. این باعث می شود که معیار قوی تری برای گرایش مرکزی باشد.

- نتایج قابل تفسیر: medoid یک نقطه داده واقعی را در خوشه نشان می دهد که می تواند برای تفسیرپذیری و درک خوشه ها مفید باشد.

- با هر متریک فاصله کار می کند: در حالی که میانگین به یک متریک فاصله پیوسته نیاز دارد، medoid را می توان با هر معیار عدم تشابه، از جمله داده های غیر عددی یا غیر اقلیدسی استفاده کرد.

معایب:

پیچیدگی محاسباتی: محاسبه medoid مستلزم محاسبه عدم تشابه بین هر جفت نقطه در یک خوشه است که می تواند از نظر محاسباتی گران باشد، به خصوص برای مجموعه داده های بزرگ.

- محدود به نماینده منفرد: برخلاف میانگین که مرکز ثقل یک خوشه را نشان می‌دهد، medoid فقط یک نقطه داده را نشان می‌دهد. این ممکن است به طور کامل ویژگی های خوشه را در بر نگیرد، به خصوص اگر مدوید خود یک حالت پرت باشد.

- حساسیت به مقداردهی اولیه: انتخاب medoid های اولیه می تواند به طور قابل توجهی بر نتایج خوشه بندی تأثیر بگذارد و یافتن medoid های اولیه بهینه می تواند یک کار چالش برانگیز باشد.

- انتخاب نقاط اولیه به شکلی که بیشترین فاصله را از هم داشته باشند

انتخاب نقاط اولیه در الگوریتم k-means نقش مهمی در نتیجه خوشه بندی نهایی دارد. یک رویکرد این است که نقاط اولیه را به گونه ای انتخاب کنید که بیشترین فاصله را از یکدیگر داشته باشند. این روش دارای مزایا و معایب زیر است:

مزایای:

- همگرایی بهبود یافته: انتخاب نقاط اولیه با بیشترین فاصله از یکدیگر می تواند به جای گرفتار شدن در بهینه محلی منجر به شانس بیشتری برای همگرایی به یک بهینه جهانی شود.

- احتمال بالاتر گرفتن حالت های مختلف خوشه: با شروع با نقاط اولیه متنوع، احتمال بیشتری برای پوشش حالت های متعدد در توزیع داده ها و یافتن خوشه های متمایز وجود دارد.

معایب:

- حساسیت به نقاط پرت: اگر نقاط اولیه بر اساس حداکثر فاصله انتخاب شوند، ممکن است در نهایت نقاط پرت باشند. نقاط دورافتاده می توانند تأثیر نامتناسبی بر تکرارهای بعدی الگوریتم داشته باشند که منجر به نتایج خوشه بندی غیربهینه می شود.

- عدم تضمین راه حل بهینه: در حالی که انتخاب نقاط اولیه با حداکثر فاصله می تواند به بهبود همگرایی کمک کند، یافتن راه حل بهینه جهانی را تضمین نمی کند.

- افزایش پیچیدگی محاسباتی: یافتن نقاط اولیه با حداکثر فاصله نیاز به محاسبات اضافی دارد، مانند محاسبات فاصله زوجی بین تمام نقاط داده. این می تواند زمان بر باشد، به خصوص برای مجموعه داده های بزرگ.

- انتخاب نقاط اولیه بر اساس توزیع دادهها

استراتژی دیگر برای انتخاب نقاط اولیه در الگوریتم k-means در نظر گرفتن توزیع داده ها است. این رویکرد شامل قرار دادن نقاط اولیه در مناطقی است که نقاط داده متراکم هستند یا واریانس بالایی از خود نشان می دهند. در اینجا مزایا و معایب وجود دارد:

مزایای:

- همگرایی بهبود یافته: با قرار دادن نقاط اولیه در مناطق متراکم یا با واریانس بالا، الگوریتم به احتمال زیاد به سرعت به خوشه های معنی دار همگرا می شود.

- سازگاری با ویژگی‌های داده: این روش ویژگی‌های توزیعی داده‌ها را در نظر می‌گیرد، که می‌تواند برای شناسایی خوشه‌هایی که با ساختار داده‌های زیربنایی همسو هستند، مفید باشد.

- کاهش حساسیت به نقاط پرت: با در نظر گرفتن توزیع داده‌ها، نقاط اولیه کمتر تحت تأثیر نقاط پرت قرار می‌گیرند و در نتیجه نتایج خوشه‌بندی قوی‌تری حاصل می‌شود.

معایب:

- افزایش پیچیدگی محاسباتی: انتخاب نقاط اولیه بر اساس توزیع داده ها نیازمند محاسبات اضافی برای تخمین توزیع داده ها یا تعیین نقاط نماینده است.

- ذهنیت: انتخاب نقاط اولیه بر اساس توزیع داده ها ممکن است کمی ذهنیت ایجاد کند، زیرا رویکردهای مختلف برای تخمین توزیع یا انتخاب نقاط نماینده می تواند به نتایج متفاوتی منجر شود.

- انتخاب چندباره مراکز اولیه برای رسیدن به جواب مناسب

روش دیگر برای تقویت الگوریتم k-means این است که الگوریتم را چندین بار با موقعیت‌های مرکز اولیه متفاوت اجرا کنید و راه‌حلی را انتخاب کنید که بهترین نتیجه خوشه‌بندی را به همراه داشته باشد. این به کاهش مشکل گیر کردن الگوریتم در بهینه محلی کمک می کند.

مزایای:

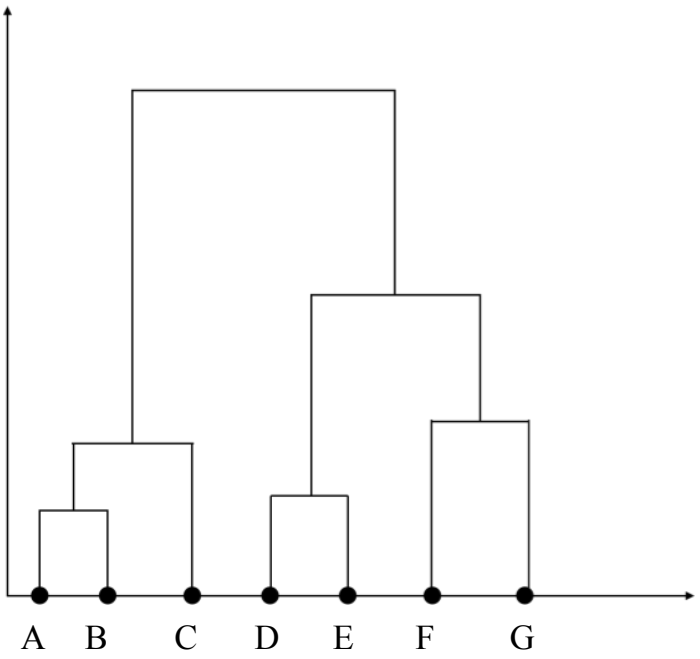
- افزایش احتمال یافتن راه‌حل بهتر: اجرای k-means چندین بار با موقعیت‌های مرکزی متفاوت، امکان کاوش راه‌حل‌های مختلف را فراهم می‌کند و شانس یافتن بهینه جهانی را افزایش می‌دهد.
- استحکام: با در نظر گرفتن چندین راه حل، نتیجه خوشه بندی قوی تر می شود و کمتر به موقعیت های مرکزی اولیه وابسته می شود.

معایب:

- افزایش هزینه محاسباتی: اجرای چندین بار الگوریتم k-means می تواند هزینه محاسباتی را به طور قابل توجهی افزایش دهد، به خصوص برای مجموعه داده های بزرگ یا داده های با ابعاد بالا.

- تضمینی برای بهبود وجود ندارد: اگرچه اجرای چندین بار k-means شانس یافتن راه حل بهتر را افزایش می دهد، اما تضمین نمی کند که بهترین راه حل پیدا شود. هنوز امکان گیر کردن در راه حل های غیربهینه وجود دارد، به خصوص اگر موقعیت های اولیه به اندازه کافی متنوع نباشند.

ج) دندروگرام زیر، انجام خوشه‌بندی سلسله مراتبی را بر روی یک مجموعهدادگان را نشان میدهد، با توجه به دندروگرام مشخص نمایید که اگر بخواهیم بر روی دادههای زیر الگوریتم k-means را اجرا نماییم بهتر است که چه مقداری را به k دهیم.



سوال پنجم

ماتریس زیر را در نظر بگیرید با استفاده از روش PCA دادهها را به یک بعد انتقال داده و ماتریس داده حاصل را بدست آورید.

$$\begin{bmatrix} 1 & 1 \\ 1 & 2 \\ 2 & 1 \\ -1 & -1 \\ -1 & -2 \\ -2 & -1 \end{bmatrix}$$

$mean_1 = 0, \quad mean_2 = 0$

$var_1 = 2, \quad var_2 = 2$

$$covariance\ matrix = \frac{1}{10}\begin{bmatrix} 1 & 1 & 2 & -1 & -1 & -2 \\ 1 & 2 & 1 & -1 & -2 & -1 \end{bmatrix}\begin{bmatrix} 1 & 1 \\ 1 & 2 \\ 2 & 1 \\ -1 & -1 \\ -1 & -2 \\ -2 & -1 \end{bmatrix} = \begin{bmatrix} 1.2 & 1 \\ 1 & 1.2 \end{bmatrix}$$

Eigen values: $det(A - \lambda I) = det(\begin{bmatrix} 1.2-\lambda & 1 \\ 1 & 1.2-\lambda \end{bmatrix}) \quad \rightarrow \quad \lambda = 2.2, 0.2$

Eigen vectors: $\begin{bmatrix} 1 \\ 1 \end{bmatrix}$, $\begin{bmatrix} 1 \\ -1 \end{bmatrix}$

$$\begin{bmatrix} 1 & 1 \\ 1 & 2 \\ 2 & 1 \\ -1 & -1 \\ -1 & -2 \\ -2 & -1 \end{bmatrix}\begin{bmatrix} 1 \\ 1 \end{bmatrix} = \frac{1}{\sqrt{2}}\begin{bmatrix} 2 \\ 3 \\ 3 \\ -2 \\ -3 \\ -3 \end{bmatrix}$$

سوال ششم

با فرض آستانه پشتیبانی برابر ۰.۳ و آستانه اطمینان برابر ۰.۴ ، مجموعه تمام قوانین انجمنی ممکن را بنویسید.

TID	*Items*
1	Bread, Milk
2	Bread, Diaper, Beer, Eggs
3	Milk, Diaper, Beer, Coke
4	Bread, Milk, Diaper, Beer
5	Bread, Milk, Diaper, Coke
6	Milk, Diaper, Coke
7	Bread, Diaper, Beer

Items	Freq	Support
Bread	5	0.71
Milk	5	0.71
Diaper	6	0.85
Beer	4	0.57
Eggs	1	0.14
Coke	3	0.42

Items	Freq	Support
Bread, Milk	3	0.42
Bread, Diaper	4	0.57
Bread, Beer	3	0.42
Bread, Coke	1	0.14
Milk, Diaper	4	0.57
Milk, Beer	2	0.28
Milk, Coke	3	0.42
Diaper, Beer	4	0.57
Diaper, Coke	3	0.42
Beer, Coke	1	0.14

Items	Freq	Support
Bread, Milk, Diaper	2	0.28
Bread, Diaper, Beer	3	0.42
Milk, Diaper, Coke	2	0.28

Rules	confidence
Bread → Milk	0.6
Milk → Bread	0.6
Bread → Diaper	0.8
Diaper → Bread	0.67
Bread → Beer	0.6
Beer → Bread	0.75
Milk → Diaper	0.8
Diaper → Milk	0.67
Milk → Coke	0.6
Coke → Milk	1
Diaper → Beer	0.67
Beer → Diaper	1
Diaper → Coke	0.5
Coke → Diaper	1
Bread → Beer, Diaper	0.6
Beer → Bread, Diaper	0.75
Diaper → Beer, Bread	0.5
Bread, Beer → Diaper	1
Bread, Diaper → Beer	0.75
Diaper, Beer → Bread	0.75

Rules
Bread → Milk
Milk → Bread
Bread → Diaper
Diaper → Bread
Bread → Beer
Beer → Bread
Milk → Diaper
Diaper → Milk
Milk → Coke
Coke → Milk
Diaper → Beer
Beer → Diaper
Diaper → Coke
Coke → Diaper
Bread → Beer, Diaper
Beer → Bread, Diaper
Diaper → Beer, Bread
Bread, Beer → Diaper
Bread, Diaper → Beer
Diaper, Beer → Bread

سوال هفتم

با فرض آستانه پشتیبانی ۱/۳ و آستانه اطمینان ۲/۳، مجموعه آیتمهای پرتکرار را به دست آورید. در مرحله بعد مجموعه‌ی تمام قوانین انجمنی ممکن را به دست آورید.

Items	Freq	Support
A	5	0.625
B	5	0.625
C	5	0.625
D	4	0.5
E	2	0.25

Items	Freq	Support
A, B	3	0.375
A, C	3	0.375
A, D	2	0.25
B, C	4	0.5
B, D	2	0.25
C, D	2	0.25

Items	Freq	Support
A, B, C	3	0.375

Rules	confidence
A → B	0.6
B → A	0.6
A → C	0.6
C → A	0.6
B → C	0.8
C → B	0.8
A → B, C	0.6
B → A, C	0.6
C → A, B	0.6
A, B → C	1
B, C → A	0.75
A, C → B	1

Rules
A, B → C
B, C → A
A, C → B
B → C
C → B