حل تمرین های داده‌کاوی

مثال ۱: طبقه‌بندی با درخت تصمیم(Decision Tree)

مسئله :یک بانک می‌خواهد بر اساس سن، درآمد و سابقه‌ی اعتباری مشتریان، پیش‌بینی کند که آیا وام درخواستی تأیید می‌شود یا خیر.

گام‌های حل:

۱ .ابتدا متغیر هدف را مشخص کنید: «تأیید وام» با دو مقدار بله/خیر.

۲ .برای هر ویژگی (سن، درآمد، سابقه اعتباری)، میزان بهره‌ی اطلاعاتی (Information Gain) یا کاهش آنتروپی را محاسبه کنید.

۳ .ویژگی‌ای که بیشترین بهره‌ی اطلاعاتی را دارد، به عنوان گره‌ی ریشه انتخاب می‌شود. در این مثال معمولاً «سابقه اعتباری» بیشترین تأثیر را دارد.

۴ .درخت را به صورت بازگشتی برای هر شاخه ادامه دهید تا زمانی که گره‌ها خالص شوند یا معیار توقف برسد.

۵ .برای جلوگیری از بیش‌برازش(Overfitting) ، درخت را با روش Pruning هرس کنید.

نکته‌ای که کمتر گفته می‌شود :بسیاری در محاسبه‌ی آنتروپی دچار اشتباه محاسباتی می‌شوند چون فرمول لگاریتم پایه ۲ را با پایه ۱۰ اشتباه می‌گیرند. همیشه پیش از محاسبه، پایه‌ی مورد استفاده در سؤال را بررسی کنید.

حل عددی کامل:

مشتری	سن	درآمد	سابقه اعتباری	وام تأیید شد؟
۱	جوان	پایین	بد	خیر
۲	جوان	بالا	خوب	بله
۳	میان‌سال	بالا	خوب	بله
۴	مسن	پایین	خوب	بله
۵	مسن	بالا	بد	خیر
۶	میان‌سال	پایین	بد	خیر
۷	جوان	بالا	بد	خیر

بله	خوب	بالا	مسن	۸

از ۸ نمونه، ۴ مورد «بله» و ۴ مورد «خیر» است، پس آنتروپی کل مجموعه برابر است با:

$$Entropy(S) = -(٤/٨)\times\log_2(٤/٨) - (٤/٨)\times\log_2(٤/٨) = -٠/٥\times(-١) - ٠/٥\times(-١) = ١$$

حالا داده را بر اساس «سابقه اعتباری» تقسیم می‌کنیم:

- شاخه‌ی «خوب»: مشتریان ۲، ۳، ۴، ۸ ←هر ۴ نفر «بله = Entropy← «۰
- شاخه‌ی «بد»: مشتریان ۱، ۵، ۶، ۷ ←هر ۴ نفر «خیر = Entropy← «۰

بهره‌ی اطلاعاتی:

$$Information\ Gain = ١ - [(٤/٨)\times٠ + (٤/٨)\times٠] = ١$$

چون بهره‌ی اطلاعاتی برابر با حداکثر مقدار ممکن (۱) است، «سابقه اعتباری» به‌تنهایی و بدون نیاز به هیچ ویژگی دیگری، کل داده را به‌صورت خالص تفکیک می‌کند و به عنوان گره‌ی ریشه‌ی درخت انتخاب می‌شود. در نتیجه درخت نهایی فقط یک سطح دارد: اگر سابقه اعتباری «خوب» باشد ←وام تأیید می‌شود؛ در غیر این صورت ←رد می‌شود.

مثال ۲: خوشه‌بندی با الگوریتمK-Means

مسئله :یک فروشگاه آنلاین می‌خواهد مشتریان را بر اساس میزان خرید و تعداد بازدید از سایت به سه گروه تقسیم کند.

گام‌های حل:

۱ .تعداد خوشه‌ها (k=۳) را انتخاب کنید. این عدد یا در صورت سؤال داده شده، یا با روش Elbow تعیین می‌شود.

۲ .سه مرکز خوشه‌ی اولیه را به صورت تصادفی انتخاب کنید.

۳ .فاصله‌ی اقلیدسی هر نقطه از مراکز خوشه را محاسبه کرده و هر نقطه را به نزدیک‌ترین مرکز اختصاص دهید.

۴ .میانگین نقاط هر خوشه را دوباره محاسبه کرده و مرکز جدید را جایگزین کنید.

۵ .مراحل ۳ و ۴ را تا زمانی که مراکز خوشه دیگر تغییر نکنند، تکرار کنید.

نکته‌ای که کمتر گفته می‌شود :انتخاب مراکز اولیه به‌صورت تصادفی می‌تواند نتیجه‌ی نهایی را تغییر دهد. در تمرین‌های امتحانی، همیشه فرض اولیه‌ی داده‌شده در سؤال را رعایت کنید، نه فرضی که خودتان انتخاب می‌کنید.

حل عددی کامل:

مشتری	خرید (واحد صد هزار تومان)	تعداد بازدید
C۱	۲	۳
C۲	۳	۳
C۳	۶	۶
C٤	۸	۸
C۵	۲	۴
C٦	۷	۷

فرض کنید k=۲ و مراکز اولیه،)C۱،۳ ۲،۳C۱ (و)C٤ ۸،۸ (باشند.

تکرار اول –محاسبه فاصله‌ی اقلیدسی هر نقطه تا دو مرکز:

مشتری	فاصله تا مرکز۱	فاصله تا مرکز۲	خوشه
C۱	۰	۷.۸۱	۱
C۲	۱.۰۰	۷.۰۷	۱
C۳	۵.۰۰	۲.۸۳	۲
C٤	۷.۸۱	۰	۲
C۵	۱.۰۰	۷.۲۱	۱
C٦	۶.۴۰	۱.۴۱	۲

مراکز جدید بر اساس میانگین اعضای هر خوشه:

- خوشه ۱ :(C۱, C۲, C۵) مرکز جدید)) = ۲+۳+۲/(۳) , ۴+۳+۳/(۳) = (**۳.۳۳ , ۲.۳۳**)
- خوشه ۲ :(C۳, C٤, C٦) مرکز جدید)) = ۷+۸+۶/(۳) , ۷+۸+۶/(۳) = (**۷ , ۷**)

تکرار دوم :با محاسبه‌ی مجدد فاصله‌ها از مراکز جدید، هر ۶ مشتری دقیقاً در همان خوشه‌ی قبلی باقی می‌مانند، یعنی الگوریتم همگرا شده است .نتیجه‌ی نهایی: مشتریانC۱ ، C۲و C۵ یک گروه «کم‌خرید» و مشتریانC۳ ، C٤و C٦ یک گروه «پرخرید و فعال» را تشکیل می‌دهند.

مثال ۳: قوانین انجمنی (Association Rules) و تحلیل سبد خرید

مسئله :بررسی تراکنش‌های یک سوپرمارکت برای یافتن الگوهایی مانند «مشتریانی که نان می‌خرند، اغلب کره هم می‌خرند.»

گام‌های حل:

۱ .مقدار Support برای هر مجموعه آیتم را محاسبه کنید: تعداد تراکنش‌هایی که آن آیتم‌ها را با هم دارند، تقسیم بر کل تراکنش‌ها.

۲ .مقدار Confidence قانون A→B را محاسبه کنید Support(A,B) :تقسیم بر.Support(A)

۳ .مقدار Lift را برای سنجش قدرت واقعی رابطه محاسبه کنید. اگر Lift بیشتر از ۱ باشد، رابطه معنادار است.

۴ .الگوریتم Apriori را برای حذف مجموعه‌های کم‌تکرار) زیر آستانه (Minimum Support اعمال کنید.

نکته‌ای که کمتر گفته می‌شود :خیلی‌ها Confidence بالا را با یک رابطه‌ی معنادار اشتباه می‌گیرند. یک قانون می‌تواند Confidence بالا داشته باشد اما Lift آن نزدیک یا کمتر از ۱ باشد، یعنی در واقع هیچ رابطه‌ی واقعی وجود ندارد.

حل عددی کامل:

آیتم‌های خریداری‌شده	تراکنش
نان، کره، شیر	T۱

نان، کره	T۲
نان، شیر	T۳
کره، شیر	T٤
نان، کره، شیر	T٥

از مجموع ۵ تراکنش، محاسبات مربوط به قانون «نان ←کره» به این صورت است:

- Support(نان) = ۴ از ۵ تراکنش = **۰.۸**
- Support(کره) = ۴ از ۵ تراکنش = **۰.۸**
- Support(نان و کره با هم) = ۳ از ۵ تراکنش = (T۱, T۲, T۵) **۰.۶**
-)Confidenceنان ←کره)Support = (نان,کرهSupport ÷ ((نان) = ۰.۶ ÷ ۰.۸ = **۰.۷۵) ٪۷۵(**
-)Liftنان ←کرهSupport ÷ Confidence = ((کره) = ۰.۷۵ ÷ ۰.۸ = **۰.۹۳۷**

با اینکه Confidence این قانون ٪۷۵ و در نگاه اول بالا به نظر می‌رسد، مقدار Lift کمتر از ۱ است. این یعنی خرید نان عملاً احتمال خرید کره را افزایش نمی‌دهد —دو آیتم صرفاً به این دلیل با هم دیده می‌شوند که هر دو در کل تراکنش‌ها پرتکرار هستند، نه به این دلیل که واقعاً به هم مرتبط‌اند. این دقیقاً همان اشتباهی است که در نکته‌ی بالا به آن اشاره شد.

مثال ۴: رگرسیون خطی برای پیش‌بینی

مسئله :پیش‌بینی قیمت یک آپارتمان بر اساس متراژ آن.

گام‌های حل:

۱ .داده‌ها را روی نمودار پراکندگی رسم کنید تا رابطه‌ی خطی بودن یا نبودن بررسی شود.

۲ .با روش کمترین مربعات خطا (Least Squares) ضرایب خط رگرسیون (شیب و عرض از مبدأ) را محاسبه کنید.

۳ .معادله‌ی خط را برای پیش‌بینی مقادیر جدید استفاده کنید.

۴ .کیفیت مدل را با معیار) R^2ضریب تعیین (ارزیابی کنید.

نکته‌ای که کمتر گفته می‌شود :مقدار R^2 بالا همیشه به معنای مدل خوب نیست. اگر داده کم باشد یا متغیرهای دیگری در قیمت تأثیرگذار باشند (مثل موقعیت مکانی)، مدل ساده رگرسیون خطی می‌تواند گمراه‌کننده باشد.

حل عددی کامل:

آپارتمان	متراژ (متر مربع)	قیمت (میلیون تومان)
۱	۵۰	۵۰۰
۲	۶۰	۶۰۰
۳	۷۰	۶۵۰
۴	۸۰	۷۵۰
۵	۱۰۰	۹۰۰

میانگین متراژ = ۷۲ و میانگین قیمت = ۶۸۰. با فرمول کمترین مربعات خطا:

شیب خط = $\Sigma[(x-\bar{x})(y-\bar{y})] \div \Sigma[(x-\bar{x})^2]$ = ۱۱۷۰۰ ÷ ۱۴۸۰ = **۷.۹۱**

عرض از مبدأ = $a = \bar{y} - b\times\bar{x}$ = ۶۸۰ − ۷.۹۱ × ۷۲ = (**۱۱۰.۸**)

معادله‌ی خط رگرسیون :قیمت = ۱۱۰.۸ + ۷.۹۱ × متراژ

با این معادله، قیمت یک آپارتمان ۹۰ متری این‌گونه پیش‌بینی می‌شود:

قیمت = ۱۱۰.۸ + (۷.۹۱ × ۹۰) = ۱۱۰.۸ + ۷۱۱.۹ ≈ ۸۲۲.۷ میلیون تومان

با محاسبه‌ی مجموع مربعات خطا و مجموع مربعات کل، مقدار R^2 این مدل تقریباً ۰.۹۹۴ به‌دست می‌آید؛ یعنی حدود ۹۹.۴٪ از تغییرات قیمت با متراژ توضیح داده می‌شود که نشان‌دهنده‌ی برازش بسیار قوی مدل خطی روی این داده‌ی خاص است.

مثال ۵: تشخیص داده‌های پرت(Outlier Detection)

مسئله :در یک دیتاست تراکنش‌های بانکی، تراکنش‌های مشکوک به تقلب را شناسایی کنید.

گام‌های حل:

۱ .میانگین و انحراف معیار داده را محاسبه کنید.

۲ .با استفاده از روش Z-Score، هر داده‌ای که بیش از ۳ انحراف معیار از میانگین فاصله دارد را به عنوان داده‌ی پرت علامت‌گذاری کنید.

۳ .به‌عنوان روش جایگزین، از) IQRادامنه‌ی میان‌چارکی (استفاده کنید: هر مقداری خارج از بازه Q1-۱٫۵×IQR تا Q۳+۱٫۵×IQRپرت محسوب می‌شود.

۴ .داده‌های پرت را بررسی کنید تا مشخص شود خطای ثبت داده هستند یا واقعاً یک الگوی غیرعادی (مثل تقلب) را نشان می‌دهند.

نکته‌ای که کمتر گفته می‌شود :حذف خودکار همه‌ی داده‌های پرت اشتباه است. در بسیاری از مسائل واقعی (مثل تشخیص تقلب)، خودِ داده‌ی پرت هدف اصلی تحلیل است، نه یک نویز که باید حذف شود.

حل عددی کامل:

مبلغ ۸ تراکنش (به هزار تومان): ۴۸، ۴۹، ۵۰، ۵۱، ۵۲، ۵۳، ۵۵، ۷۰۰

میانگین = ۱۳۲.۲۵ و انحراف معیار ≈ ۲۱۴.۶

Z-Scoreتراکنش مشکوک) ۷۰۰:(

) = Z۷۰۰ − ۱۳۲.۲۵ ÷ (۲۱۴.۶ = ۲.۶۵

نکته‌ی مهم اینجاست: عدد ۲.۶۵ از آستانه‌ی رایج ۳ کمتر است، پس ممکن است این روش این تراکنش را «پرت» تشخیص ندهد! دلیل آن این است که همان مقدار۷۰۰ خودش میانگین و انحراف معیار را به‌شدت بالا کشیده است — این یکی از ضعف‌های شناخته‌شده‌ی Z-Score در نمونه‌های کوچک است.

حالا همین داده را با روش QR ابررسی می‌کنیم:

- چارک اول ≈ (Q۱) ۴۹.۲۵
- چارک سوم ≈ (Q۳) ۵۴.۵
- ۵.۲۵IQR = Q۳ − Q۱ =
- سقف مجاز = IQR×۱٫۵ + Q۳ = ۵۴.۵ + ۷.۸۸ = ۶۲.۳۸

چون مقدار ۷۰۰ به‌وضوح از سقف ۶۲.۳۸ بزرگ‌تر است، روش IQR این تراکنش را به‌درستی به عنوان **داده‌ی پرت** شناسایی می‌کند. این مثال دقیقاً نشان می‌دهد چرا در دیتاست‌های کوچک یا دارای پرت‌های شدید، IQRمعمولاً قابل‌اعتمادتر از Z-Score است.

مثال ۶: کاهش ابعاد با تحلیل مؤلفه‌های اصلی(PCA)

مسئله :یک دیتاست با ۲۰ متغیر دارید و می‌خواهید آن را برای تجسم بهتر به دو بعد کاهش دهید.

گام‌های حل:

۱ .داده‌ها را استانداردسازی کنید (میانگین صفر، واریانس یک).

۲ .ماتریس کوواریانس بین متغیرها را محاسبه کنید.

۳ .مقادیر ویژه (Eigenvalues) و بردارهای ویژه (Eigenvectors) را استخراج کنید.

۴ .مؤلفه‌هایی که بیشترین واریانس داده را توضیح می‌دهند (معمولاً دو یا سه مؤلفه اول) را انتخاب کنید.

۵ .داده‌ی اصلی را روی این مؤلفه‌های جدید تصویر کنید.

نکته‌ای که کمتر گفته می‌شود PCA **:**فقط یک روش کاهش بعد نیست، بلکه ابزاری قدرتمند برای حذف هم‌خطی چندگانه (Multicollinearity) پیش از اجرای مدل‌های رگرسیونی است؛ نکته‌ای که در اکثر تمرین‌های دانشگاهی نادیده گرفته می‌شود.

حل عددی کامل (نسخه‌ی ساده‌شده با ۲ متغیر):

نمونه	متغیرX	متغیرY

۲.۴	۲.۵	۱
۰.۷	۰.۵	۲
۲.۹	۲.۲	۳
۲.۲	۱.۹	۴
۳.۰	۳.۱	۵

میانگین = X ۲.۰۴ و میانگین = Y ۲.۲۴ .پس از مرکزی‌کردن داده و محاسبه‌ی ماتریس کوواریانس:

Cov(X,X) = ۰.۹۳۸ Cov(Y,Y) = ۰.۸۵۳ — Cov(X,Y) = ۰.۸۴۱

با حل معادله‌ی مقادیر ویژه روی این ماتریس، دو مقدار ویژه به‌دست می‌آید:

- λ۱ = ۱.۷۳۷)مؤلفه‌ی اصلی اول(
- λ۲ = ۰.۰۵۴)مؤلفه‌ی اصلی دوم(

درصد واریانس توضیح‌داده‌شده توسط مؤلفه‌ی اول:

(λ۱ ÷ (λ۱ + λ۲) = ۱.۷۳۷ ÷ ۱.۷۹۱ ≈ ۹۷٪

یعنی تنها با نگه‌داشتن یک بُعد (مؤلفه‌ی اول) به‌جای دو متغیر اصلی، حدود ۹۷٪ از اطلاعات داده حفظ می‌شود. در دیتاست‌های واقعی با ۲۰ متغیر، همین منطق برای انتخاب دو یا سه مؤلفه‌ی برتر از میان تمام مؤلفه‌ها استفاده می‌شود.

مقایسه ی ۶ روش داده‌کاوی

روش	نوع داده هدف	کاربرد اصلی	سطح پیچیدگی
درخت تصمیم	برچسب‌دار(Classification)	پیش‌بینی دسته	متوسط
K-Means	بدون برچسب(Clustering)	گروه‌بندی مشتریان	متوسط
قوانین انجمنی	تراکنشی	تحلیل سبد خرید	ساده تا متوسط
رگرسیون خطی	عددی پیوسته	پیش‌بینی مقدار	ساده
تشخیص داده پرت	آماری	شناسایی تقلب/خطا	متوسط
PCA	چندبعدی	کاهش ابعاد و تجسم	بالا

اشتباهات رایج در حل تمرین‌های داده‌کاوی

- انتخاب الگوریتم بدون بررسی نوع متغیر هدف (کیفی یا کمی بودن آن).
- نادیده گرفتن نرمال‌سازی داده قبل از اجرای الگوریتم‌های مبتنی بر فاصله مثل K-Means .
- تفسیر نکردن نتیجه‌ی نهایی و بسنده کردن به محاسبه‌ی عددی صرف.
- استفاده از کل داده برای آموزش مدل، بدون تفکیک داده آموزش و آزمون.
- اشتباه گرفتن همبستگی با رابطه‌ی علّی در تفسیر نتایج رگرسیون یا قوانین انجمنی.