

دانشگاه صنعتی امیرکبیر
(پلی تکنیک تهران)

دانشکده مهندسی کامپیووتر

تمرین اول درس داده کاوی

نگارش

مهدیه سادات بنیس

استاد درس

دکتر احسان ناظرفرد

نیم سال دوم ۱۴۰۱

• بخش تئوری:

سوال اول

به سوالات زیر پاسخ دهید.

(الف) داده‌ی پرت با نویز را با یکدیگر مقایسه کنید.

در داده کاوی، نویز به تغییرات تصادفی یا خطاهایی گفته می‌شود که در داده‌ها وجود دارد و از الگو یا ساختار خاصی پیروی نمی‌کند. نویز می‌تواند ناشی از خطاهای اندازه‌گیری، خطاهای ورود داده‌ها یا سایر منابع تصادفی باشد. از سوی دیگر، نقاط پرت به مشاهداتی اطلاق می‌شود که به طور قابل توجهی با بقیه داده‌ها متفاوت است. این مشاهدات ممکن است به دلیل یک خطای اندازه‌گیری، خطای ورود داده‌ها یا ممکن است نشان دهنده یک پدیده واقعی باشد. برخلاف نویز، نقاط پرت می‌توانند تأثیر معناداری بر تجزیه و تحلیل و تفسیر داده‌ها داشته باشند. تمایز بین نویز و نقاط پرت در داده کاوی مهم است زیرا آنها به روش‌های مختلفی برای مدیریت و تجزیه و تحلیل نیاز دارند. در حالی که نویز را می‌توان فیلتر یا صاف کرد تا تغییرات تصادفی حذف شود، داده‌ی پرت ممکن است به توجه و بررسی بیشتری نیاز باشد.

(ب) یک سناریو بیان کنید که در آن داده‌های پرت برای ما مفید هستند و اطلاعات ارزشمندی از آن دریافت می‌کنیم.

برای مثال فرض کنید میخواهیم تأثیر یک داروی جدید بر فشار خون را مورد بررسی قرار دهیم.

فشار خون ۱۰۰ شرکت کننده قبل و بعد از دریافت دارو را اندازه‌گیری می‌کنیم. اکثر شرکت کنندگان پس از مصرف دارو کاهش فشار خون را نشان می‌دهند که نتیجه مورد انتظار است. با این حال، یک شرکت کننده وجود دارد که پس از مصرف دارو افزایش قابل توجهی در فشار خون نشان می‌دهد که در داده‌ها دور از ذهن است.

در نگاه اول، نقطه پرت ممکن است مانند یک ناهنجاری یا یک خطای اندازه‌گیری به نظر برسد. با این حال، پس از بررسی دقیق‌تر، مشخص می‌شود شرکت کننده‌ای که افزایش فشار خون را تجربه کرده بود، یک بیماری زمینه‌ای نادر دارد که قبلاً تشخیص داده نشده بود. این اطلاعات ارزشمندی در مورد عوارض جانبی دارو ارائه می‌دهد و جمعیت خاصی را که ممکن است در معرض خطر باشند شناسایی می‌کند.

در نتیجه، تحقیقات بیشتری برای درک مکانیسم اصلی و تعیین اینکه آیا دارو برای افراد با شرایط پزشکی مشابه بی‌خطر است یا خیر، انجام می‌شود. این نقطه داده‌ی پرت، که در ابتدا یک ناهنجاری به نظر می‌رسید، در شناسایی یک خطر و ارائه اطلاعات مهم برای تحقیقات بیشتر و تصمیم‌گیری در زمینه پزشکی بسیار مهم بود.

آنچه باید به در نظر بگیریم این است که همه داده‌های پرت یکسان نیستند. برخی تأثیر قوی دارند، برخی دیگر اصلاً برخی از مقادیر داده معتبر و مهم هستند. برخی از آنها به سادگی خطای نویز هستند.

بنابراین باید موارد زیر را مورد بررسی قرار دهیم:

- چرا می‌خواهیم نقطه پرت را پیدا کنیم؟ ممکن است بخواهید داده‌های پرت را ببینیم زیرا به این ناهنجاری علاقه دارید. باید به این فکر کنیم که مسئله چیست.

- آیا داده‌های پرت مشکلی در نتیجه، تأثیر یا فرضیات ایجاد می‌کند؟

- از کجا آمده است؟ این ممکن است به تجزیه و تحلیل عمیق و تخصص حوزه نیاز داشته باشد. علاوه بر این، همیشه نمی‌توانیم بگوییم از کجا آمده است، اما باید سعی کنیم احتمالات مختلف را در نظر گرفته زیرا می‌تواند به بهترین راه برای ادامه راه کمک کند.

(ج) مشخص کنید که آیا یک نویز میتواند داده‌ی پرت باشد یا خیر؟

بله، در برخی موارد، نویز را می‌توان به عنوان یک داده‌ی پرت در نظر گرفت برای مثال اگر که یک مقدار شدید باشد که به طور قابل توجهی از محدوده مقادیر مورد انتظار برای متغیر مورد نظر منحرف شود. به عنوان مثال، در مجموعه داده‌ای از قیمت سهام در طول یک ماه، اگر نقطه داده‌ای وجود داشته باشد که قیمت سهام به طور غیرعادی بالا نشان دهد که به هیچ روند یا الگوی زیربنایی ارتباطی ندارد، می‌تواند به عنوان نویز در نظر گرفته شود و همچنین به عنوان یک نقطه پرت طبقه‌بندی شود.

سوال دوم

در حوزه‌ی داده‌کاوی، انبار داده چیست و چه تفاوت و شباهتی با پایگاه داده دارد؟

انبار داده جایی است که می‌توان داده‌ها را برای اهداف استخراج جمع آوری کرد، معمولاً با ظرفیت ذخیره سازی بزرگ. سیستم‌های سازمان‌های مختلف در انبار داده‌ها قرار دارند، جایی که می‌توان آن‌ها را بر حسب استفاده واکنشی کرد.

انبارهای داده، داده‌ها را از چندین منبع با یکدیگر همکاری می‌کنند و از صحت، کیفیت و سازگاری داده‌ها اطمینان حاصل می‌کنند. اجرای سیستم با متمایز کردن فرآیند تجزیه و تحلیل از پایگاه‌های داده سنتی تقویت می‌شود. در یک انبار داده، داده‌ها بر اساس نوع و در صورت نیاز به یک الگوی قالب بندی شده مرتب می‌شوند. داده‌ها توسط ابزارهای پرس و جو با استفاده از چندین الگو بررسی می‌شوند.

انبارهای داده داده‌های historical را ذخیره می‌کنند و در خواستها را سریع‌تر رسیدگی می‌کنند و به پردازش تحلیلی آنلاین کمک می‌کنند، در حالی که از یک پایگاه داده برای ذخیره تراکنش‌های جاری در یک فرآیند تجاری استفاده می‌شود که transaction processing نامیده می‌شود.

هدف از استفاده از انبار داده ترکیب منابع داده‌های متفاوت به منظور تجزیه و تحلیل داده‌ها، جستجوی بینش و ایجاد هوش تجاری (BI) در قالب گزارش‌ها و داشبوردها است.

هر دو انبار داده و پایگاه داده سیستم‌های ذخیره سازی داده هستند که معمولاً برای ذخیره مقادیر زیادی از داده‌های ساخت یافته استفاده می‌شوند. هر دو را می‌توان پرس و جو کرد و با تراکنش‌ها به روز کرد. هر دوی آنها حاوی داده‌هایی درباره یک یا چند نهاد مانند مشتریان و محصولات هستند.

سیستم‌های ذخیره سازی همان چیزی است که شما به آن نگاه می‌کنید. به طور معمول، لایه پایین انبار داده یک پایگاه داده رابطه‌ای است. سیستم‌های پایگاه داده رابطه‌ای نیز پایگاه داده ای هستند که داده‌ها در ردیف‌ها و ستون‌ها در سیستم‌های پایگاه داده رابطه‌ای ذخیره می‌شوند.

هم پایگاه داده و هم انبار داده به چندین کاربر امکان می‌دهند به طور همزمان به داده‌های مشابه دسترسی داشته باشند. بسیاری از کاربران می‌توانند به طور همزمان به یک پایگاه داده یا انبار داده دسترسی داشته باشند. برای دستیابی به داده‌ها، باید کوئری‌ها را هم در datawarehouse و هم در پایگاه داده اجرا کنید. از پرس و جوهای پیچیده می‌توان برای دسترسی به انبار داده استفاده کرد، اما از پرس و جوهای ساده می‌توان برای دسترسی به پایگاه داده OLTP استفاده کرد. و در نهایت، چه در محل یا در فضای ابری، انبار داده و پایگاه داده یک شرکت در دسترس است.

تفاوت اصلی بین این دو این است که یک انبار داده به طور خاص برای تجزیه و تحلیل طراحی شده است، در حالی که پایگاه‌های داده عمده‌تا برای استفاده "transactional" طراحی شده‌اند. علاوه بر این، انبارهای داده، داده‌های تاریخی و انبوه (غلب از منابع متفاوت) را ذخیره می‌کنند، در حالی که پایگاه‌های داده اغلب فقط وضعیت‌های اخیر و/یا فعلی اطلاعات را ذخیره می‌کنند. این می‌تواند بر اساس برنامه متفاوت باشد.

مقایسه انبار داده و پایگاه داده به طور خلاصه در جدول زیر بیان شده است:

	Data warehouse	Data warehouse
Purpose	Analysis of data	Recording data
Data Type	Historical Data (often summarized)	Real Time (Detailed data including metadata)
Processing Method	OLAP (online analytical processing)	OLTP (online transactional processing)
Type of collection	Subject-oriented	Application-oriented

Users	Limited	Can vary from 00's to 000's and more
Query	Complex analytical queries	Transaction queries (CRUD)
Service Level Agreement (SLA)	99.99 upwards for mission critical apps	Flexible (refreshes usually occur once a day)

سوال سوم

یکی از روش‌های یافتن داده‌های پرت استفاده از توزیع نرمال و percentile ها است. در مورد این روش تحقیق کرده و آن را توضیح دهید.

این تکنیک با تنظیم یک مقدار آستانه خاص، که بر اساس بیانیه مشکل ما تصمیم گیری می‌شود، کار می‌کند.

در حالی که ما نقاط پرت را با استفاده از capping حذف می‌کنیم، آن روش خاص به عنوان Winsorization شناخته می‌شود.

در اینجا، ما همیشه تقارن را در هر دو طرف حفظ می‌کنیم، به این معنی که اگر ۱% را از سمت راست برداریم، سمت چپ نیز ۱% کاهش می‌یابد.

روش صدک (percentile) برای تشخیص داده‌های پرت را می‌توان با مجموعه داده‌هایی که از توزیع نرمال پیروی می‌کنند و همچنین با مجموعه داده‌هایی که انواع دیگری از توزیع دارند استفاده کرد. تفاوت اصلی در نحوه محاسبه و تفسیر رتبه‌های صدک در زمینه توزیع نرمال است.

در مجموعه داده‌ای که از توزیع نرمال پیروی می‌کند، نقاط داده به طور متقاضی حول میانگین توزیع می‌شوند. نقاط داده نزدیکتر به میانگین فراوانتر هستند، در حالی که نقاط داده دورتر از میانگین فراوانی کمتری دارند. رنک‌های صدک نقاط داده در یک توزیع نرمال را می‌توان به عنوان موقعیت نسبی نقطه داده در امتداد منحنی توزیع تفسیر کرد.

تشخیص پرت با استفاده از صدک‌ها شامل شناسایی نقاط داده‌ای است که خارج از محدوده خاصی از صدک‌ها قرار دارند. محدوده صدک‌ها را می‌توان به عنوان درصدی از داده‌ها مشخص کرد، مانند ۵ یا ۱ درصد بالا و پایین نقاط داده. نقاط داده‌ای که خارج از این محدوده قرار می‌گیرند، نقاط پرت در نظر گرفته می‌شوند.

مراحلی که برای روش صدک باید دنبال کرد:

- مرتب سازی مجموعه داده: نقاط داده در مجموعه داده را بسته به محاسبه صدک مورد نظر به ترتیب صعودی یا نزولی مرتب کنید.
- محاسبه رنک صدک: برای هر نقطه داده در مجموعه داده، رتبه صدک را محاسبه کنید، که نشان دهنده درصد نقاط داده در مجموعه داده است که زیر نقطه داده قرار می‌گیرند. فرمول محاسبه رتبه صدک:

$$\text{Percentile rank} = (\text{Number of data points below the data point} + 0.5) / \text{Total number of data points} \times 100$$

از افروندن ۰.۵ در شمارشگر برای محاسبه مسائل گرد کردن هنگام محاسبه رتبه‌های صدک برای مجموعه داده‌های گسسته استفاده می‌شود.

۴- انتخاب یک صدک آستانه: یک مقدار صدک آستانه را بر اساس سطح سختگیری مورد نظر برای شناسایی نقاط پرت انتخاب کنید. به عنوان مثال، صدک آستانه ۹۵ به این معنی است که نقاط داده با رتبه‌هایی که صدکی بیشتر از ۹۵ درصد می‌توانند به عنوان نقاط پرت در نظر گرفته شوند.

۵- شناسایی نقاط پرت: رتبه صدک هر نقطه داده را با مقدار صدک آستانه مقایسه کنید. اگر رتبه صدک یک نقطه داده بالاتر از صدک آستانه باشد، می‌توان آن را به عنوان نقطه پرت طبقه‌بندی کرد.

سوال چهارم

فرایند پاکسازی داده‌ها و نمایش داده‌ها را در نظر بگیرید:

الف) فرایند پاکسازی داده‌ها را تعریف کنید.

پاکسازی داده‌ها فرآیند شناسایی و تصحیح داده‌های فاسد، ناقص، تکراری، نادرست و نامربوط از مجموعه مرجع، جدول یا پایگاه داده است. این شامل شناسایی خطاهای داده و سپس تغییر، به روز رسانی یا حذف داده‌ها برای اصلاح آنها است. پاکسازی داده‌ها کیفیت داده‌ها را بهبود می‌بخشد و به ارائه اطلاعات دقیق‌تر، سازگارتر و قبل اعتمادتر برای تصمیم‌گیری در سازمان کمک می‌کند.

مشکلات داده معمولاً از طریق خطاهای ورودی کاربر، جمع‌آوری ناقص داده‌ها، فرمتهای غیر استاندارد و مشکلات یکپارچه‌سازی داده‌ها به وجود می‌آیند.

هنگام ترکیب چندین منبع داده، فرصت‌های زیادی برای تکرار یا برچسب‌گذاری اشتباه داده‌ها وجود دارد. اگر داده‌ها نادرست باشند، نتایج و الگوریتم‌ها غیرقابل اعتماد هستند، حتی اگر درست به نظر برسند. هیچ راه مطلقی برای تجویز مراحل دقیق در فرآیند پاکسازی داده‌ها وجود ندارد زیرا فرآیندها از مجموعه داده‌ای به مجموعه دیگر متفاوت خواهند بود. اما بسیار مهم است که یک الگو برای فرآیند پاکسازی داده‌ها خود ایجاد کنید تا بدانید که هر بار آن را به درستی انجام می‌دهید.

پاکسازی داده‌ها بخش کلیدی فرآیند کلی مدیریت داده و یکی از اجزای اصلی کارآمدسازی داده است که مجموعه‌های داده را برای استفاده در هوش تجاری (BI) و کاربردهای علم داده آماده می‌کند.

ب) اهمیت نمایش داده‌ها را بیان کنید و به یک مورد از چالش‌های آن اشاره کنید.

نمایش داده‌ها به افراد کمک می‌کند داده‌ها را ببینند، با آنها تعامل داشته باشند و بهتر درک کنند. چه ساده و چه پیچیده، نمایش درست می‌تواند به همه بدون توجه به سطح تخصصشان یک دید یکسان بدهد.

به سختی می‌توان یک صنعت حرفه‌ای که برای قابل فهم تر کردن داده‌ها تلاشی نمی‌کند، مثال زد. هر زمینه STEM از درک داده‌ها سود می‌برد - و همینطور زمینه‌هایی در دولت، امور مالی، بازاریابی، تاریخ، کالاهای مصرفی، صنایع خدماتی، آموزش، ورزش و غیره. هرچه بهتر بتوانیم نکات خود را به صورت بصری منتقل کنیم، چه در داشبورد یا یک اسلاید، بهتر می‌توانیم از آن اطلاعات استفاده کنیم. مجموعه مهارت‌ها برای تطبیق با دنیای داده محور در حال تغییر هستند. برای حرفه‌ای‌ها ارزش فرازینده‌ای دارد که بتوانند از داده‌ها برای تصمیم‌گیری استفاده کنند و از تصاویر بصری برای گفتن داستان‌هایی درباره زمانی که داده‌ها به چه کسی، چه چیزی، چه زمانی، کجا و چگونه اطلاع می‌دهند، استفاده کنند.

در حالی که آموزش سنتی معمولاً بین داستان‌سرایی خلاق و تحلیل تکنیکال مرز مشخصی می‌کشد، دنیای حرفه‌ای مدرن همچنین برای کسانی ارزش قائل است که می‌توانند بین این دو تلاقی کنند: نمایش داده‌ها درست در وسط تحلیل و داستان‌گویی بصری قرار می‌گیرد.

در حالی که تحلیلگران با تخصص نمایش داده‌ها، قدرت رها کردن داستان‌های داده را دارند، ایجاد نمودارهای روشنگر آسان نیست، زیرا چالش‌های مهمی وجود دارد. فقدان طراحی در نمودارها و نقشه‌ها، مخاطب را گمراه کرده و آنها را از اطلاعات مهم دور می‌کند. خیلی چیزها با نمایش داده‌ها ممکن است اتفاق بیفتد. چند چالش وجود دارد که می‌تواند نمایش‌ها را مشکل ساز کند، مانند:

۱- فقدان سواد بصری سازی داده‌ها

فقدان دانش مناسب از نمایش داده‌ها می‌تواند بیشتر از اینکه مفید باشد آسیب برساند. اول اینکه مخاطب را گیج می‌کند یا بدتر از آن او را گمراه می‌کند. علاوه بر این، نمایش داده با ساختار ضعیف نشان‌دهنده از دست دادن زمان و تلاش است که فرآیند تصمیم‌گیری را در پروژه‌های مهلت‌محور به تأخیر می‌اندازد. علاوه بر این، برخی از ابزارهای نمایش داده‌ها توضیح نمی‌دهند که منظورشان چیست. از این رو، بینندگان اغلب آنچه را که به درک آنها می‌رسد، درک می‌کنند، و گاهی اوقات بینندگان بیشتر از آنچه انجام می‌دهند، می‌فهمند. چنین نتایجی اغلب به دلایلی مانند:

استفاده از نوع نمودار اشتباه

استفاده ضعیف از نمودار سه بعدی

ارائه داده‌های گمراه کننده یا ناکافی

مقیاس ناسازگار در بین داده‌های ارائه شده

یک نمودار بصری به هم ریخته

۲- ساده سازی بیش از حد داده‌ها

چالش دیگر این است که نمایش داده ها بیش از حد ساده باشد. این به همان اندازه مضر است که آن را پیچیده تر می کند. مشتریان یا تصمیم گیرندگان شما داده های کافی برای تصمیم گیری دقیق در صورت حذف نکات مهم را نخواهند داشت. نمایش داده ها باید داده ها را به گونه ای نشان دهد که به راحتی قابل درک باشد. با این حال، اگر بخش های حیاتی نامفهوم و ناکافی باشند، مخاطب درد اصلی ارائه را درک نخواهد کرد. بنابراین، به جای ساده سازی بیش از حد داده ها، بهتر است تمام نقاط زمینه حیاتی را کنار هم قرار دهید و ساختار آنها را طوری بسازید که هر کسی بتواند به سرعت جنبه های واقعی را درک کند.

ج) چرا پاکسازی داده ها یک فرایند مهم و پیشنهاد برای نمایش داده ها میباشد؟

پاکسازی داده ها برای نمایش داده ها ضروری است زیرا کیفیت داده ها مستقیماً بر دقت، قابلیت اطمینان و اثربخشی تجسم ها تأثیر می گذارد.

چند دلیل کلیدی وجود دارد که چرا پاکسازی داده ها برای نمایش داده ها مهم است:

- نمایش دقیق و قابل اعتماد: نمایش داده ها برای ارتباط بینش ها، الگوهای روندها در داده ها استفاده می شود. با این حال، اگر داده هایی که نمایش داده می شوند نادرست یا ناسازگار باشند، می تواند منجر به نتیجه گیری های گمراه کننده یا نادرست شود. پاکسازی داده ها به شناسایی و تصحیح خطاهای ناسازگاری ها و نادرستی ها در داده ها کمک می کند و اطمینان می دهد که تجسم ها بر اساس داده های دقیق و قابل اعتماد هستند.

- نمایش منسجم و منسجم: پاکسازی داده ها کمک می کند تا اطمینان حاصل شود که داده های مورد استفاده برای نمایش سازگار و منسجم هستند. داده های متناقض یا ناقص می توانند منجر به نمایش هایی شود که تفسیر یا مقایسه آنها دشوار است. پاکسازی داده ها به استانداردسازی قالب های داده، رفع اختلاف ها و پر کردن مقادیر از دست رفته کمک می کند و اطمینان می دهد که نمایش ها منسجم و معنادار هستند.

- کیفیت داده بهبود یافته: نمایش داده ها فقط به خوبی کیفیت داده های در حال نمایش است. پاکسازی داده ها با شناسایی و تصحیح خطاهای ناسازگاری ها، کیفیت کلی داده ها را بهبود می بخشد. داده های پاک منجر به نمایش های معنادار و دقیق تر، ارائه بینش بهتر و پشتیبانی تضمیم گیری می شود.

- کاوش داده های پیشرفته: نمایش داده ها اغلب برای تجزیه و تحلیل داده های اکتشافی استفاده می شود، جایی که تحلیلگران داده ها را برای کشف الگوهای روندها و روابط کاوش می کنند. پاکسازی داده ها تمیز و دقیق بودن داده ها را تضمین می کند و به تحلیلگران اجازه می دهد تا داده ها را به طور مؤثرتری کاوش کنند و اکتشافات معناداری انجام دهند.

- حریم خصوصی و امنیت داده ها: پاکسازی داده ها همچنین نقش مهمی در تضمین حریم خصوصی و امنیت داده ها دارد. پاکسازی داده ها با شناسایی و حذف هر گونه اطلاعات شناسایی شخصی (PII) یا سایر داده های حساسی که نباید در نمایش ها گنجانده شوند، به محافظت از حریم خصوصی و امنیت داده ها و رعایت مقررات مربوط به حفاظت از داده ها کمک می کند.

سوال پنجم

در یک آزمایشگاه ژنتیک مقدار فعالیت دو ژنوم مختلف مورد بررسی قرار گرفته و در ۱۰ بازه زمانی مختلف در به صورت زیر ثبت شده است:

Gen/Time	T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	Avg
G1	-3	5	8	-2	1	2	3	-5	10	-1	1.8
G2	9	20	16	8	2	10	-6	-15	25	-2	6.7

الف) با استفاده از معیار شباهت Mutual Information، Correlation، Cosine Similarity شباهت این دو ژن را مقایسه کنید.

$$\text{Cosine Similarity} = \frac{A \cdot B}{\|A\| \|B\|} = \frac{\sum_{i=1}^n A_i B_i}{\sqrt{\sum_{i=1}^n A_i^2} \sqrt{\sum_{i=1}^n B_i^2}}$$

$$\text{Cosine Similarity} = \frac{(-27) + 100 + 128 + (-16) + 2 + 20 + (-18) + 75 + 250 + 2}{\sqrt{9 + 25 + 64 + 4 + 1 + 4 + 9 + 25 + 100 + 1 + \sqrt{81 + 400 + 256 + 64 + 4 + 100 + 35 + 225 + 625 + 4}}} = \frac{516}{659.082}$$

$$\text{Cosine Similarity} = \frac{516}{659.082} \approx 0.7829$$

$$\text{Correlation} = \text{Corr}(x, y) = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^n (x_i - \bar{x})^2 \sum_{i=1}^n (y_i - \bar{y})^2}}$$

$$\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y}) = (-3 - 1.8)(9 - 6.7) + (5 - 1.8)(20 - 6.7) + (8 - 1.8)(16 - 6.7) + (-2 - 1.8)(8 - 6.7) + (1 - 1.8)(2 - 6.7)$$

$$+ (2 - 1.8)(10 - 6.7) + (3 - 1.8)(-6 - 6.7) + (-5 - 1.8)(-15 - 6.7) + (10 - 1.8)(25 - 6.7) + (-1 - 1.8)(-2 - 6.7) = 395.4$$

$$\sum_{i=1}^n (x_i - \bar{x})^2 = (-3 - 1.8)^2 + (5 - 1.8)^2 + (8 - 1.8)^2 + (-2 - 1.8)^2 + (1 - 1.8)^2 + (2 - 1.8)^2 + (3 - 1.8)^2 + (-5 - 1.8)^2 + (10 - 1.8)^2 + (-1 - 1.8)^2 = 209.6$$

$$\sum_{i=1}^n (y_i - \bar{y})^2 = (9 - 6.7)^2 + (20 - 6.7)^2 + (16 - 6.7)^2 + (8 - 6.7)^2 + (2 - 6.7)^2 + (10 - 6.7)^2 + (-6 - 6.7)^2 + (-15 - 6.7)^2 + (25 - 6.7)^2 + (-2 - 6.7)^2 = 1346.1$$

$$\text{Correlation} = \text{Corr}(x, y) = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^n (x_i - \bar{x})^2 \sum_{i=1}^n (y_i - \bar{y})^2}} = \frac{395.4}{\sqrt{209.6 \times 1346.1}} = 0.7444$$

$$\text{Mutual Information} = MI(X, Y) = \sum_{x \in X, y \in Y} p_{(X,Y)}(x, y) \log \frac{p_{(X,Y)}(x, y)}{p_X(x)p_Y(y)}$$

Gen/Time	T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	Total
G1	-3	5	8	-2	1	2	3	-5	10	-1	18
G2	9	20	16	8	2	10	-6	-15	25	-2	67
Total	6	25	24	6	3	12	-3	-20	35	-3	85

$$\text{Mutual Information} = MI(X, Y) = \sum_{x \in X, y \in Y} p_{(X,Y)}(x, y) \log \frac{p_{(X,Y)}(x, y)}{p_X(x)p_Y(y)} = 1.7$$

ب) طبق نتایج هر معیار مشخص کنید آیا دو ژنوم از یکدیگر مستقل هستند یا خیر.

استقلال به عدم وجود رابطه یا وابستگی بین دو متغیر اشاره دارد. اگر دو متغیر مستقل باشند، به این معنی است که وقوع یا مقدار یک متغیر تأثیری بر وقوع یا مقدار متغیر دیگر ندارد. استقلال اغلب در آمار و تئوری احتمال ارزیابی می شود و معمولاً با استفاده از معیارهایی مانند احتمال شرطی یا استقلال مشروط اندازه گیری می شود.

به طور خلاصه، برای معیار Cosine Similarity یک نشان دهنده شباهت کامل، ۱- نشان دهنده عدم تشابه کامل و ۰ نشان دهنده عدم تشابه است. • Correlation به معنی که هیچ رابطه خطی بین متغیرها وجود ندارد.

با توجه به بالا بودن مقدار های هر دو و نزدیک بود به ۱ نشان میدهد که دو ژنوم به هم وابستگی دارند و تا حد خوبی به هم شباهت دارند.

ج) آیا نتایج به دست آمده متفاوت است؟ اگر پاسخ مثبت است علت آن را توضیح دهید.

بله، معیار شباهت Mutual Information، Correlation، Cosine Similarity سه معیار متفاوتی هستند که می توانند مقادیر متفاوتی داشته باشند و برای اهداف مختلف در زمینه های مختلف استفاده می شوند.

- Mutual Information معیاری از وابستگی متقابل یا اطلاعات مشترک بین دو متغیر تصادفی است. مقدار اطلاعاتی را که یک متغیر در مورد متغیر دیگر ارائه می دهد کمیت می کند. Mutual Information معمولاً در تئوری اطلاعات و آمار برای اندازه گیری وابستگی آماری بین دو متغیر استفاده می شود و بر اساس توزیع احتمال مشترک یاتابع جرم احتمال محاسبه می شود.

- Correlation ارتباط خطی بین دو متغیر را اندازه گیری می کند. این نشان دهنده قدرت و جهت رابطه خطی بین دو متغیر است، که در آن همبستگی مثبت به این معنی است که متغیرها با هم تمایل به افزایش یا کاهش دارند، Correlation منفی به این معنی است که یک متغیر با کاهش متغیر دیگر تمایل به افزایش دارد و • Correlation به معنی که هیچ رابطه خطی بین متغیرها وجود ندارد. Correlation معمولاً در آمار و تجزیه و تحلیل داده ها برای اندازه گیری قدرت و جهت ارتباط خطی بین متغیرها استفاده می شود.

- Cosine Similarity : تشابه کسینوس شباهت بین دو بردار را در یک فضای چند بعدی اندازه گیری می کند. کسینوس زاویه بین دو بردار را محاسبه می کند و از ۱- تا ۱ متغیر است که ۱ نشان دهنده شباهت کامل، ۰ نشان دهنده عدم تشابه کامل و ۰ نشان دهنده عدم تشابه است. Cosine Similarity اغلب در متن کاوی، تجزیه و تحلیل استناد و سیستم های توصیه برای اندازه گیری شباهت بین اسناد یا آیتم ها بر اساس نمایش های برداری آنها استفاده می شود.

در حالی که این معیارها ممکن است در برخی موارد تفسیرهای مشابهی داشته باشند، معیارهای ریاضی متفاوتی با محاسبات و کاربردهای مورد نظر متفاوت هستند. بنابراین، بسته به داده ها و زمینه ای که در آن اعمال می شوند، می توانند مقادیر متفاوتی داشته باشند.

سوال ششم

دو مورد از روش‌های data preprocessing روشنای sampling و aggregation هستند. این دو روش را توضیح داده و مزايا و معایب هر یک را بنویسید.

روش aggregation

نوعی فرآيند داده‌کاوی است که در آن داده‌ها جستجو، جمع‌آوری و ارائه می‌شوند تا در قالبی خلاصه‌شده و مبتنی بر گزارش برای دستیابی به اهداف یا فرآيندهای تجاری خاص و/یا انجام تجزیه و تحلیل انسانی انجام شود.

روش aggregation برای اين می باشد که تحلیلگران کسب و کار بتوانند تحلیل های آماری طرح های تجاری را انجام دهند. اطلاعات جمع‌آوری شده ممکن است از منابع داده‌های مختلف جمع‌آوری شود تا این منابع داده را در یک پیش‌نویس برای تجزیه و تحلیل داده‌ها خلاصه کند. این مرحله گام اصلی هر سازمان تجاری است زیرا دقت بینش حاصل از تجزیه و تحلیل داده ها عمدتاً به کیفیت داده هایی که آنها استفاده می کنند بستگی دارد. جمع آوری محتواي با كيفيت در مقاديير زياد بسيار ضروري است تا بتوانند نتایج مرتبط را ايجاد کنند. data aggregation نقشی حياتی در امور مالي، محصول، عمليات و استراتژي های بازاریابی در هر سازمان تجاری ايقا می کند. داده های انباشته شده در انبار داده وجود دارد که می تواند افراد را قادر به حل مسائل مختلف کند، که به حل پرس و جو از مجموعه داده ها کمک می کند.

معایب	مزایا
شامل خلاصه کردن داده ها است که می تواند منجر به از دست رفتن اطلاعات دقیق شود	ساده کردن تحلیل داده
اگر تجمع با دقت انجام نشود یا اگر داده ها از منابع مختلف یا در سطوح مختلف دانه بندی بدون توجه مناسب ترکیب شوند، تجمع داده ها می تواند سوگیری ايجاد کند	با کاهش خطاهای، ناسازگاری ها و افزونگی ها به بهبود کیفیت داده ها کمک کند.
اعطاف پذیری تجزیه و تحلیل داده ها و اکتشافرا محدود کند، زیرا ممکن است داده های دقیق اصلی دیگر برای تجزیه و تحلیل بیشتر در دسترس نباشند	با جمع‌آوری داده‌ها در سطح بالاتری از جزئیات بهم‌حافظت از اطلاعات حساس کمک کند، بنابراین خطر افسای داده‌های سطح فردی را کاهش می‌دهد.
پیچیدگی بیشتر در مدیریت داده ها، یکپارچه سازی داده ها، و فرآيندهای تبدیل داده ها	بهینه سازی استفاده از منابع، مانند فضای ذخیره سازی و قدرت پردازش، با کاهش مقدار داده هایی که باید ذخیره و پردازش شوند

روش sampling

یک تکنیک تجزیه و تحلیل آماری است که برای انتخاب، پردازش و تجزیه و تحلیل یک زیرمجموعه نماینده یک جامعه استفاده می شود. همچنین برای شناسایی الگوها و بروز بایی روندها در یک جمعیت کلی استفاده می شود. با نمونه‌گیری داده‌ها، محققان، دانشمندان داده، مدل‌سازان پیش‌بینی کننده و دیگر تحلیل‌گران داده می توانند از حجم کمتر و قابل مدیریت‌تری از داده‌ها برای ساخت و اجرای مدل‌های تحلیلی استفاده کنند.

نمونه گیری یک تکنیک آماری رایج است که به عنوان مثال برای نظرسنجی های سیاسی یا نظرسنجی استفاده می شود. اگر محقق بخواهد محبوب ترین راه رفت و آمد به محل کار در ایالات متحده را تعیین کند، نیازی به صحبت با هر شهروند آمریکایی نخواهد داشت. در عرض، آنها می توانند یک گروه نماینده ۱۰۰۰ نفری را انتخاب کنند، به این اميد که برای درست کردن نتایج کافی باشد. در تجزیه و تحلیل وب، نمونه برداری به روشی بسیار مشابه عمل می کند. فقط زیر مجموعه ای از ترافیک شما انتخاب و تجزیه و تحلیل می شود و از آن نمونه برای تخمین نتایج کلی استفاده می شود.

معایب	مزایا
احتمال پاسخ های مغرضانه	صرفه جویی در زمان
انتخاب نمونه های خوب دشوار است	نمونه گیری از تکرار پرس و جو برای هر فرد اجتناب می کند
دانش محدود ممکن است نتایج را گمراه کند	نمونه گیری نزدیک ترین نتایج را به دست می دهد
روش نمونه گیری ممکن است نامناسب باشد	با منابع کم، داده های بیشتری دریافت می کنید

سوال هفتم

در رابطه با کاهش بعد تحقیق کرده و به سوالات زیر پاسخ بدھید.

الف) مفاهیم انتخاب ویژگی، استخراج ویژگی و مهندسی ویژگی را توضیح و تفاوت‌های بین آنها را بیان کنید.

انتخاب ویژگی، استخراج ویژگی و مهندسی ویژگی سه تکنیک مهمی هستند که در یادگیری ماشین و تجزیه و تحلیل داده‌ها برای بهینه‌سازی عملکرد مدل‌های پیش‌بینی و بهبود دقت و تفسیرپذیری نتایج استفاده می‌شوند.

- انتخاب ویژگی: انتخاب ویژگی فرآیند انتخاب زیرمجموعه‌ای از مرتبط ترین ویژگی‌ها (یا متغیرها) از مجموعه اصلی ویژگی‌ها در یک مجموعه داده است. هدف از انتخاب ویژگی کاهش تعداد ویژگی‌ها به آنهاست که برای ساختن یک مدل پیش‌بینی دقیق و کارآمد بسیار مهم هستند. روش‌های انتخاب ویژگی می‌توانند بر اساس معیارهای مختلفی مانند معیارهای آماری، الگوریتم‌های یادگیری ماشین یا دانش دامنه باشند. برخی از تکنیک‌های رایج برای انتخاب ویژگی عبارتند از روش‌های فیلتر، روش‌های پوشش (wrapper)، و روش‌های جاسازی شده (embedded).

- استخراج ویژگی: استخراج ویژگی تبدیل فرآیند ایجاد ویژگی‌های اصلی در یک مجموعه داده به مجموعه جدیدی از ویژگی‌ها است که مرتبط‌ترین اطلاعات را در نمایش فشرده‌تر و معنادارتر جمع‌آوری می‌کند. تکنیک‌های استخراج ویژگی به ویژه در هنگام برخورد با داده‌های با ابعاد بالا یا زمانی که ویژگی‌های اصلی نویز، زائد یا نامربوط هستند. روش‌های استخراج ویژگی معمولاً شامل تکنیک‌هایی مانند تجزیه و تحلیل مؤلفه اصلی (PCA)، تجزیه و تحلیل تفکیک خطی (LDA)، یا تجزیه ارزش منفرد (SVD) برای تبدیل ویژگی‌های اصلی به فضای ویژگی با ابعاد پایین‌تر است که مهم‌ترین اطلاعات را حفظ می‌کند.

- مهندسی ویژگی: مهندسی ویژگی فرآیند ایجاد ویژگی‌های جدید یا اصلاح ویژگی‌های موجود در یک مجموعه داده برای بهبود عملکرد یک مدل پیش‌بینی است. مهندسی ویژگی شامل داشت حوزه، خلاقیت و شهود برای شناسایی و ایجاد ویژگی‌های جدید است که ممکن است آموزنده‌تر یا مرتبط‌تر با مشکل خاص در دست باشد. مهندسی ویژگی می‌تواند شامل وظایف مختلفی مانند رمزگذاری متغیرهای طبقه‌بندی، مدیریت مقادیر از دست رفته، مقایسه‌بندی یا عادی‌سازی ویژگی‌ها، ایجاد ویژگی‌های تعامل، یا استخراج اطلاعات معنادار از متغیرهای تاریخ یا زمان باشد. مهندسی ویژگی یک فرآیند تکراری است که شامل آزمایش تغییرات یا ایجاد ویژگی‌های مختلف، ارزیابی تأثیر آن‌ها بر عملکرد مدل و اصلاح مکرر ویژگی‌ها برای بهینه‌سازی مدل است.

تفاوت اصلی بین انتخاب ویژگی، استخراج ویژگی و مهندسی ویژگی به شرح زیر است:

- هدف: هدف انتخاب ویژگی انتخاب زیرمجموعه‌ای از مرتبط‌ترین ویژگی‌ها از مجموعه اصلی ویژگی‌ها در یک مجموعه داده است، با هدف کاهش تعداد ویژگی‌ها و در عین حال حفظ مهم‌ترین آنها. از سوی دیگر، استخراج ویژگی با هدف تبدیل ویژگی‌های اصلی به مجموعه جدیدی از ویژگی‌ها است که مرتبط‌ترین اطلاعات را در یک نمایش فشرده‌تر و معنادارتر به تصویر می‌کشد. مهندسی ویژگی شامل ایجاد ویژگی‌های جدید یا اصلاح ویژگی‌های موجود برای بهبود عملکرد یک مدل پیش‌بینی است.

- تکنیک‌ها: هر کدام دارای تکنیک‌های متفاوتی هستند که پیشتر بیان شده است.

- تبدیل داده‌ها: انتخاب ویژگی و استخراج ویژگی هر دو روش تبدیل ویژگی‌های اصلی در یک مجموعه داده هستند، اما آنها این کار را به روش‌های مختلف انجام می‌دهند. انتخاب ویژگی شامل انتخاب زیرمجموعه‌ای از ویژگی‌ها از مجموعه اصلی است، در حالی که استخراج ویژگی شامل تبدیل ویژگی‌های اصلی به مجموعه جدیدی از ویژگی‌ها است که ممکن است ابعاد کمتری داشته باشند یا اطلاعات مرتبط بیشتری را به دست آورند. از سوی دیگر، مهندسی ویژگی شامل ایجاد ویژگی‌های جدید یا اصلاح ویژگی‌های موجود در یک مجموعه داده است، بدون اینکه لزوماً ابعاد را کاهش دهد.

ب) الگوریتم‌های کاهش بعد به دو دسته خطی و غیرخطی تقسیم می‌شوند. تفاوت این دو دسته را توضیح داده و روش کار الگوریتم PCA از دسته خطی و الگوریتم t-sne از دسته غیرخطی را توضیح دهید.

تفاوت اصلی بین تکنیک‌های کاهش ابعاد خطی و غیرخطی در نحوه تبدیل ویژگی‌های اصلی به نمایشی با ابعاد پایین‌تر نهفته است.

- کاهش ابعاد خطی: تکنیک‌های کاهش ابعاد خطی فرض می‌کنند که رابطه بین ویژگی‌های اصلی و نمایش با ابعاد پایین‌تر خطی است. این تکنیک‌ها از تبدیل‌های خطی مانند ترکیب‌های خطی یا پیش‌بینی‌ها برای کاهش ابعاد فضای ویژگی استفاده می‌کنند. تجزیه و تحلیل اجزای اصلی (PCA) و تجزیه و تحلیل تشخیصی خطی (LDA) نمونه‌هایی از تکنیک‌های کاهش ابعاد خطی هستند.

PCA جهات حداکثر واریانس را در داده ها پیدا می کند و داده ها را بر روی این جهات پروژه می دهد، در حالی که LDA ترکیبات خطی ویژگی هایی را پیدا می کند که جداسازی طبقات را به حداکثر می رساند.

- کاهش ابعاد غیرخطی: تکنیک های کاهش ابعاد غیرخطی یک رابطه خطی بین ویژگی های اصلی و نمایش با ابعاد پایین تر را فرض نمی کنند. در عوض، هدف آنها گرفتن روابط غیرخطی و الگوهای پیچیده در داده ها است. این تکنیک ها از تبدیل های غیرخطی مانند توابع هسته یا شبکه های عصبی برای ترسیم ویژگی های اصلی به فضایی با ابعاد پایین تر استفاده می کنند. نمونه هایی از تکنیک های کاهش ابعاد غیرخطی شامل t-sne، LLE و رمزگذارهای خودکار است.

PCA (تجزیه و تحلیل مؤلفه اصلی) و t-sne (t-Distributed Stochastic Neighbor Embedding) هر دو تکنیک های کاهش ابعاد هستند که در یادگیری ماشین و تجسم داده ها استفاده می شوند. با این حال، آنها در رویکرد و ویژگی های خود متفاوت هستند.

PCA (تجزیه و تحلیل مؤلفه اصلی): PCA یک تکنیک کاهش ابعاد خطی است که هدف آن یافتن نمایشی با ابعاد پایین تر از یک مجموعه داده با شناسایی جهت های حداکثر واریانس در داده ها است. این کار با نمایش ویژگی های اصلی بر روی مجموعه جدیدی از محورهای متعامد، به نام اجزای اصلی، که ترکیبی خطی از ویژگی های اصلی هستند، کار می کند. اولین مؤلفه اصلی جهت حداکثر واریانس در داده ها را می گیرد و مؤلفه های اصلی بعدی واریانس باقی مانده را به ترتیب نزولی می گیرند. PCA به طور گسترده برای استخراج ویژگی، فشرده سازی داده ها و تجسم استفاده می شود. از نظر محاسباتی کارآمد است و می تواند برای مجموعه داده های بزرگ اعمال شود.

t-SNE: یک تکنیک کاهش ابعاد غیرخطی است که به ویژه برای تجسم داده های با ابعاد بالا در یک فضای دو بعدی یا سه بعدی مفید است. با ایجاد یک توزیع احتمال در فضای ابعاد پایین تر کار می کند که شبیه به شباهت های زوجی بین نقاط داده در فضای بالبعد بالاتر است. سپس، به طور مکرر جاسازی را در فضای با ابعاد پایین تر تنظیم می کند تا واگرایی بین دو توزیع احتمال را به حداقل برساند. t-SNE به دلیل توانایی خود در گرفتن الگوهای پیچیده و ساختارهای محلی در داده ها شناخته شده است، که آن را برای تجسم خوش ها یا گروه هایی از نقاط داده مشابه مناسب می کند. با این حال، t-SNE از نظر محاسباتی در مقایسه با PCA پیچیده تر است و نتایج ممکن است به تنظیمات ابرپارامتر حساس باشد.

سوال هشتم

برای داده های عددی زیر نمودار جعبه رارسم کنید.

۲۷, ۳, ۱, ۲۹, ۲۷, ۷۰, ۲۶, ۳۳, ۲۷, ۳۶, ۴۹, ۲۵, ۳۹, ۲۸, ۴۱

Ascending Order:	1 , 3 , 25 , 26 , 27 , 27 , 27 , 28 , 29 , 33 , 36 , 39 , 41 , 49 , 70 ,
Descending Order:	70 , 49 , 41 , 39 , 36 , 33 , 29 , 28 , 27 , 27 , 26 , 25 , 3 , 1 ,
Maximum Number:	70
Minimum Number:	1
Third Quartile:	39.0
First Quartile:	26.0
Median:	29



سوال نهم

همانطور که میدانید، یکی از روش‌های مقایسه دو توزیع آماری استفاده از روش q-q plot است.
 الف) نحوه کار این روش را توضیح دهید.

(Quantile-Quantile plots) نمودارهایی از دو کمیت در برابر یکدیگر هستند. یک چندک کسری است که مقادیر معینی که از آن چندک کمتر هستند قرار می‌گیرد. به عنوان مثال، میانه یک چندک است که در آن ۵۰٪ داده‌ها زیر آن نقطه قرار می‌گیرند و ۵۰٪ بالای آن قرار دارند. هدف از q-q plot این است که بفهمند آیا دو مجموعه داده از یک توزیع می‌آیند یا خیر. یک زاویه ۴۵ درجه در q-q plot رسم شده است. اگر دو مجموعه داده از یک توزیع مشترک حاصل شوند، نقاط روی آن خط مرجع قرار می‌گیرند.

q-q plot، یا نمودار چندک، ابزاری گرافیکی است که به ما کمک می‌کند تا ارزیابی کنیم که آیا مجموعه‌ای از داده‌ها به طور منطقی از توزیع نظری مانند عادی یا نمایی به دست آمده‌اند. برای مثال، اگر یک تحلیل آماری را اجرا کنیم که فرض می‌کند باقیمانده‌های ما به طور معمول توزیع شده‌اند، می‌توانیم از q-q plot نرمال برای بررسی این فرض استفاده کنیم. این فقط یک بررسی بصری است، نه یک اثبات محکم، بنابراین تا حدودی ذهنی است. اما به ما این امکان را می‌دهد که در یک نگاه بینیم که آیا فرض ما قابل قبول است یا خیر، و در غیر این صورت، چگونه این فرض نقض می‌شود و چه نقاط داده‌ای در نقض نقش دارند.

ب) نمودار q-q plot میتواند به شکل‌های متفاوتی نمایان شود: به طور مثال شبیه یک خط راست مورب. سه نوع از این شکل‌های متفاوت را بررسی کنید و تحلیل خود داده‌های توزیع‌های آماری ورودی به آن را بنویسید. به نظر شما از روی شکل q-q plot چه مواردی در مورد توزیع‌های آماری اولیه قابل استنتاج است؟

برای کمک به شناسایی انواع مختلف توزیع‌ها از یک نمودار q-q plot، نمونه‌هایی از نمودارهای هیستوگرام و q-q plot برای سه توزیع کیفی متفاوت ارائه می‌کنیم:

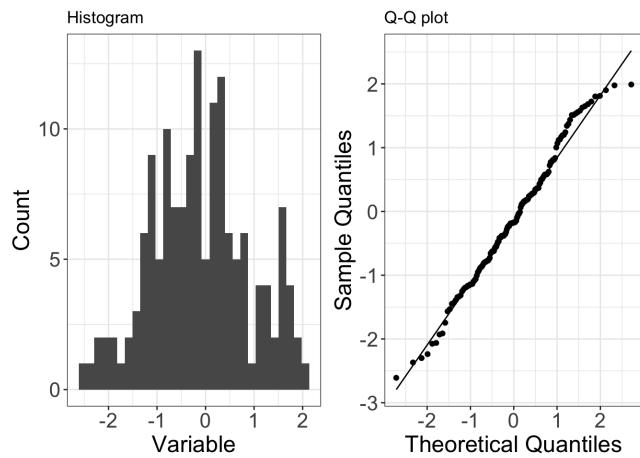
توزیع نرمال

توزیع به راست

توزیع به سمت چپ

- داده‌های معمولی توزیع شده

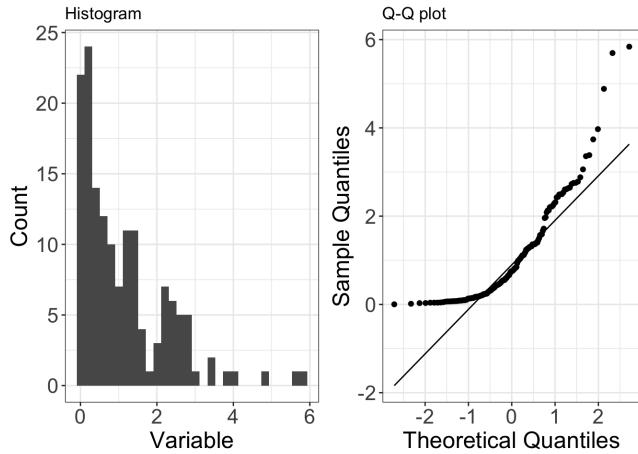
در زیر نمونه‌ای از داده‌ها که از توزیع نرمال گرفته شده است. توزیع نرمال متقاض است، بنابراین هیچ انحرافی ندارد (میانگین برابر با میانه است).



در q-q plot معمولاً داده های توزیع شده تقریباً به صورت یک خط مستقیم ظاهر می شوند (اگرچه انتهای q-q plot اغلب شروع به انحراف از خط مستقیم می کند).

- داده های انحرافی راست

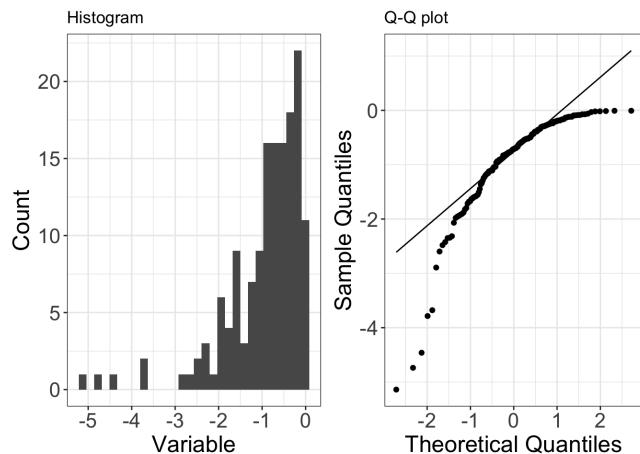
در زیر نمونه ای از داده ها آمده است که از توزیعی که به راست انحراف دارد (در این مورد توزیع نمایی است) گرفته شده است. در q-q plot این داده ها از خط مستقیم پایین ران شده اند و به نظر می رسد.



در q-q plot داده های دارای انحراف به راست منحنی به نظر می رسد.

- داده های انحرافی به چپ

در زیر نمونه ای از دادهها وجود دارد که از توزیعی به سمت چپ گرفته شده اند (در این مورد توزیع نمایی منفی است). انحراف چپ به نظر می رسد.



در q-q plot داده های چوله چپ منحنی به نظر می رسد (بر عکس داده های انحرافی راست).

- q-q plot معمولاً برای بررسی بصری اینکه آیا یک مجموعه داده از توزیع خاصی پیروی می کند، مانند توزیع نرمال، توزیع نمایی یا هر توزیع احتمال دیگری استفاده می شود. چند تفسیر رایج وجود دارد:

- نقاطی که روی یک خط مورب مستقیم قرار دارند: اگر نقاط در q-q plot از یک خط مورب مستقیم پیروی کنند، نشان می دهد که چندک های مجموعه داده نمونه و توزیع مرجع کاملاً مطابقت دارند. این نشان می دهد که مجموعه داده نمونه و توزیع مرجع دارای ویژگی های توزیعی مشابهی هستند و داده ها ممکن است به خوبی با توزیع مرجع تقریب شوند.

- نقاط انحراف از خط مورب: اگر نقاط در q-q plot از خط مورب منحرف شوند، نشان دهنده تفاوت بین مجموعه داده نمونه و توزیع مرجع است. جهت و بزرگی انحراف می تواند بینشی در مورد ماهیت تفاوت ها ارائه دهد. اگر نقاط به طور سیستماتیک در یک جهت منحرف شوند، ممکن است نشان دهنده تغییر مکان (میانگین) یا مقیاس (واریانس) مجموعه داده نمونه در مقایسه با توزیع مرجع باشد.

- الگوی S یا J شکل: اگر نقاط روی q-q plot یک الگوی S یا J شکل تشکیل دهند، ممکن است نشان دهنده چولگی یا عدم تقارن در مجموعه داده نمونه در مقایسه با توزیع مرجع باشد. این نشان می دهد که مجموعه داده نمونه دارای رفتار دم یا شکل توزیع متفاوتی در مقایسه با توزیع مرجع است.

- پراکندگی یا پراکندگی نقاط: پراکندگی یا پراکندگی نقاط در q-q plot می تواند اطلاعاتی در مورد گسترش یا تنوع مجموعه داده نمونه در مقایسه با توزیع مرجع ارائه دهد. اگر نقاط به طور محکم در اطراف خط مورب جمع شده باشند، نشان دهنده تطابق نزدیک بین مجموعه داده نمونه و توزیع مرجع است. اگر نقاط به طور گسترده پراکنده باشند، ممکن است تفاوت در تنوع یا پراکندگی بین دو توزیع را نشان دهد.

در نتیجه، تفسیر q-q plot شامل بررسی بصری الگوی نقاط روی نمودار برای ارزیابی شباهت یا تفاوت بین چندک های مجموعه داده نمونه و توزیع مرجع است. این ابزار یک ابزار گرافیکی برای ارزیابی خوب بودن تناسب بین مجموعه داده نمونه و توزیع نظری ارائه می کند و می تواند بینشی در مورد ویژگی های توزیعی داده ها ارائه دهد.

سوال دهم

برای هر یک از روش های نرمال سازی زیر تحقیق کرده و بازه هی اعداد را مشخص کنید.

نرمال سازی min-max، نرمال سازی Z-score و نرمال سازی با مقیاس دهی سه تکنیک رایجی هستند که در پیش پردازش داده ها برای نرمال سازی داده های عددی به منظور رساندن آنها به مقیاس یا محدوده قابل مقایسه استفاده می شوند.

الف) نرمال سازی min-max

این تکنیک داده ها را به یک محدوده خاص، معمولاً [۰,۱] مقیاس می کند. فرمول نرمال سازی حداقل حداقل حداکثر به صورت زیر است:

$$X_{normalized} = \frac{X - X_{min}}{X_{max} - X_{min}}$$

که در آن X داده اصلی، X_{min} حداقل مقدار در مجموعه داده، و X_{max} حداکثر مقدار در مجموعه داده است. این تکنیک داده های اصلی را به محدوده [۰,۱] ترسیم می کند که ۰ نشان دهنده حداقل مقدار و ۱ نشان دهنده حداکثر مقدار است.

ب) نرمال سازی z-score

نرمال سازی z-score داده ها را با میانگین صفر و واریانس واحد مقیاس می کند. فرمول عادی سازی امتیاز Z به صورت زیر است:

$$X_{normalized} = \frac{X - X_{mean}}{X_{std}}$$

که در آن X داده اصلی، X_{mean} میانگین مجموعه داده، و X_{std} انحراف استاندارد مجموعه داده است. این تکنیک داده ها را به میانگین ۳ X_{std} و انحراف استاندارد ۱ تبدیل می کند و در نتیجه توزیعی حول محور ۰ با گسترش ۱ ایجاد می کند. محدوده اعداد معمولاً بین -۳ X_{std} و ۳ X_{std} می باشد

ج) نرمال سازی با مقیاس دهی

مقیاس دهی: مقیاس دهی تکنیکی است که داده ها را با تقسیم هر مقدار بر توان 10^k ، معمولاً بر اساس حداکثر مقدار مطلق در مجموعه داده، مقیاس می کند. فرمول مقیاس دهی عبارت است از:

$$X_{normalized} = \frac{X}{10^k}$$

که در آن X داده اصلی و k کوچکترین عدد صحیح است به طوری که مقدار مطلق حداکثر مقدار در مجموعه داده کمتر از 10^k باشد. این تکنیک داده ها را به گونه ای مقیاس بندی می کند که مقادیری بین -1 و 1 داشته باشند، که آن را برای مجموعه های داده با محدوده های بزرگ یا پرت مفید می سازد.

سوال یازدهم

با توجه به مقادیر ورودی X و مقادیر هدف Y میتوان یک برازش خطی یا غیرخطی بر روی بسیاری از دادگان ها ایجاد کرد. با توجه به این مقادیر، به سوالات زیر پاسخ دهید.

$$X = [2, 4, 1, 3, 2, 6], \quad Y = [5, 6, 3, 6, 3, 10]$$

الف) روش محاسبه معادله نرمال را با استفاده از روش محاسبه مشتق جزئی باقیمانده کامل شرح دهید.

معادله نرمال یک راه حل شکل بسته برای یافتن پارامترهای بهینه یک مدل رگرسیون خطی بدون استفاده از الگوریتم های بهینه سازی تکراری است. برای یافتن مقادیر ضرایب مدل استفاده می شود که مجموع مجدور باقیمانده (یا میانگین مجدور خطای) بین مقادیر پیش بینی شده و مقادیر واقعی متغیر هدف را به حداقل می رساند.

معادله نرمال برای یک مدل رگرسیون خطی ساده با یک متغیر مستقل را می توان به صورت زیر بیان کرد:

$$\beta = (X^T X)^{-1} X^T Y$$

هدف پیدا کردن در معادله زیر است به گونه ای که مقدار خطای مینیمم شود:

$$S = \|Y - \beta X\|^2$$

میتوان به صورت زیر نوشت:

محاسبه میانگین توان دوم خطای (MSE):

$$J = \frac{1}{N} \sum_{i=1}^N (y_i - (\beta_0 + \beta_1 X_i))^2 = \frac{1}{N} \sum_{i=1}^N (y_i - pred_i)^2.$$

حال معادله فوق را به صورت زیر نوشه و نسبت به $\alpha = (Y - X\beta)$ مشتق میگیریم:

$$S = (Y - X\beta)(Y - X\beta)^T$$

$$\frac{dS}{d\beta} = \frac{dS}{d\alpha} \cdot \frac{d\alpha}{d\beta} = 2\alpha^T \cdot (-X) = 0 \quad \rightarrow \quad 2(Y - X\beta)^T X = 0 \quad \rightarrow \quad 2(Y^T - X^T \beta)X = 0$$

$$2Y^T X = 2X^T X\beta \quad \rightarrow \quad \beta = (X^T X)^{-1} X^T Y$$

ب) یک برازش خطی ($Y = \beta_0 + \beta_1 X$) را برای این دادگان محاسبه کنید. (مقدار خطای برازش را نیز به دست آورید)

$$X = \begin{bmatrix} 1 & 2 \\ 1 & 4 \\ 1 & 1 \\ 1 & 3 \\ 1 & 2 \\ 1 & 6 \end{bmatrix}, Y = \begin{bmatrix} 5 \\ 6 \\ 3 \\ 6 \\ 3 \\ 10 \end{bmatrix}$$

$$\beta = (X^T X)^{-1} X^T Y = \left(\begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 2 & 4 & 1 & 3 & 2 & 6 \end{bmatrix} \begin{bmatrix} 1 & 2 \\ 1 & 4 \\ 1 & 1 \\ 1 & 3 \\ 1 & 2 \\ 1 & 6 \end{bmatrix} \right)^{-1} \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 2 & 4 & 1 & 3 & 2 & 6 \end{bmatrix} \begin{bmatrix} 5 \\ 6 \\ 3 \\ 6 \\ 3 \\ 10 \end{bmatrix}$$

$$\beta = (X^T X)^{-1} X^T Y = \left(\begin{bmatrix} 6 & 18 \\ 18 & 70 \end{bmatrix} \right)^{-1} \begin{bmatrix} 33 \\ 121 \end{bmatrix} = \frac{1}{96} \begin{bmatrix} 70 & -18 \\ -18 & 6 \end{bmatrix} \begin{bmatrix} 33 \\ 121 \end{bmatrix} = \begin{bmatrix} \frac{11}{8} \\ \frac{11}{8} \end{bmatrix}$$

$$Error = Y - X\beta = \begin{bmatrix} 5 \\ 6 \\ 3 \\ 6 \\ 3 \\ 10 \end{bmatrix} - \begin{bmatrix} 1 & 2 \\ 1 & 4 \\ 1 & 1 \\ 1 & 3 \\ 1 & 2 \\ 1 & 6 \end{bmatrix} \begin{bmatrix} 0.875 \\ -0.875 \\ 0.25 \\ 0.5 \\ -1.125 \\ 0.375 \end{bmatrix}$$

ج) یک برازش غیرخطی ($Y = \beta_2 X^2 + \beta_1 X + \beta_0$) برای این دادگان محاسبه کنید. (مقدار خطای برازش را نیز به دست آورید)

$$X = \begin{bmatrix} 1 & 2 & 4 \\ 1 & 4 & 16 \\ 1 & 1 & 1 \\ 1 & 3 & 9 \\ 1 & 2 & 4 \\ 1 & 6 & 36 \end{bmatrix}, Y = \begin{bmatrix} 5 \\ 6 \\ 3 \\ 6 \\ 3 \\ 10 \end{bmatrix}$$

$$\beta = (X^T X)^{-1} X^T Y = \left(\begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 2 & 4 & 1 & 3 & 2 & 6 \\ 4 & 16 & 1 & 9 & 4 & 36 \end{bmatrix} \begin{bmatrix} 1 & 2 & 4 \\ 1 & 4 & 16 \\ 1 & 1 & 1 \\ 1 & 3 & 9 \\ 1 & 2 & 4 \\ 1 & 6 & 36 \end{bmatrix} \right)^{-1} \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 2 & 4 & 1 & 3 & 2 & 6 \\ 4 & 16 & 1 & 9 & 4 & 36 \end{bmatrix} \begin{bmatrix} 5 \\ 6 \\ 3 \\ 6 \\ 3 \\ 10 \end{bmatrix}$$

$$\beta = (X^T X)^{-1} X^T Y = \left(\begin{bmatrix} 6 & 18 & 70 \\ 18 & 70 & 324 \\ 70 & 324 & 1666 \end{bmatrix} \right)^{-1} \begin{bmatrix} 33 \\ 121 \\ 545 \end{bmatrix} = \begin{bmatrix} \frac{1981}{890} \\ \frac{334}{445} \\ \frac{39}{445} \end{bmatrix}$$

$$Error = Y - X\beta = \begin{bmatrix} 5 \\ 6 \\ 3 \\ 6 \\ 3 \\ 10 \end{bmatrix} - \begin{bmatrix} 1 & 2 & 4 \\ 1 & 4 & 16 \\ 1 & 1 & 1 \\ 1 & 3 & 9 \\ 1 & 2 & 4 \\ 1 & 6 & 36 \end{bmatrix} \begin{bmatrix} \frac{1981}{890} \\ \frac{334}{445} \\ \frac{39}{445} \end{bmatrix} = \begin{bmatrix} \frac{821}{890} \\ \frac{-561}{890} \\ \frac{-57}{890} \\ \frac{653}{890} \\ \frac{-959}{890} \\ \frac{103}{890} \end{bmatrix}$$

• بخش عملی:

سوال اول

ابتدا به دنبال داده های NaN در مجموعه داده بگردید و ذکر کنید که از هر ویژگی چند سطر فاقد داده هستند. برای این کار از تابع isna() استفاده کنید.

```
dataset.isnull().sum()
```

```
island      0  
bill_len    2  
bill_depth  2  
flipper_len 2  
body_mass   2  
sex         11  
species     0  
dtype: int64
```

سوال دوم

داده های از دست رفته در مجموعه داده را با استفاده از () Dropna() حذف کنید. و تعداد سطرهای مجموعه داده را قبل و بعد از حذف عنوان کنید.

```
#before removing missing values  
shape = dataset.shape  
print('Dataset Shape:', shape)  
print('Dataset Row Number:', shape[0])
```

```
Dataset Shape: (344, 7)  
Dataset Row Number: 344
```

```
New_dataset = dataset.dropna()
```

```
#after removing missing values  
new_shape = New_dataset.shape  
print('Dataset New Shape:', new_shape)  
print('Dataset New Row Number:', new_shape[0])
```

```
Dataset New Shape: (333, 7)  
Dataset New Row Number: 333
```

```
New_dataset.isnull().sum()
```

```
island      0  
bill_len    0  
bill_depth  0  
flipper_len 0  
body_mass   0  
sex         0  
species     0  
dtype: int64
```

سوال سوم

در گام اول داده های عددی از دست رفته در مجموعه داده را با میانگین آن ستون جایگزین کنید (یعنی تنها برای ویژگی های: body_mass_g، flipper_length_mm، bill_depth_mm، bill_length_mm). در گام دوم داده های غیر عددی از دست رفته (species، sex، island) را با متداول ترین مقدار جایگزین کنید.

```
imp_mean = SimpleImputer(missing_values=np.nan, strategy='mean')  
numerical_features = imp_mean.fit_transform(numerical_features)  
numerical_features = pd.DataFrame(numerical_features, columns=['bill_len', 'bill_depth', 'flipper_len', 'body_mass'])  
numerical_features
```

```
imp_most_frequent = SimpleImputer(missing_values=np.nan, strategy='most_frequent')  
categorical_features = imp_most_frequent.fit_transform(categorical_features)  
categorical_features = pd.DataFrame(categorical_features, columns=['island', 'sex', 'species'])  
categorical_features
```

دیتا فریم داده ها را به دو بخش ستون های عددی و غیر عددی تقسیم کرده و برای هر کدام را با روش گفته شده داده های از دست رفته را جایگزین میکنیم.

```
New_dataset2 = pd.concat([numerical_features, categorical_features], axis=1)
New_dataset2
```

	bill_len	bill_depth	flipper_len	body_mass	island	sex	species
0	39.10000	18.70000	181.000000	3750.000000	Torgersen	male	Adelie
1	39.50000	17.40000	186.000000	3800.000000	Torgersen	female	Adelie
2	40.30000	18.00000	195.000000	3250.000000	Torgersen	female	Adelie
3	43.92193	17.15117	200.915205	4201.754386	Torgersen	male	Adelie
4	36.70000	19.30000	193.000000	3450.000000	Torgersen	female	Adelie
...
339	55.80000	19.80000	207.000000	4000.000000	Dream	male	Chinstrap
340	43.50000	18.10000	202.000000	3400.000000	Dream	female	Chinstrap
341	49.60000	18.20000	193.000000	3775.000000	Dream	male	Chinstrap
342	50.80000	19.00000	210.000000	4100.000000	Dream	male	Chinstrap
343	50.20000	18.70000	198.000000	3775.000000	Dream	female	Chinstrap

344 rows × 7 columns

سوال چهارم

با استفاده از Label Encoding در ستونهای زیر، تغییرات را اعمال کنید.

- در ستون Island:Dream را به ۰، Island:Biscoe را به ۱، Island:Torgersen را به ۲ تبدیل کنید.
- در ستون sex:female را به ۰ و sex:male را به ۱ تبدیل کنید.
- و در ستون species:Adelie را به ۰ و species:Chinstrap را به ۱ و species:Gentoo را به ۲ تبدیل کنید.

میتوانید این کار را با کمک sklearn.preprocessing انجام دهید.

```
le = LabelEncoder()
```

```
categ = ['island', 'sex', 'species']
New_dataset2[categ] = New_dataset2[categ].apply(le.fit_transform)
New_dataset2
```

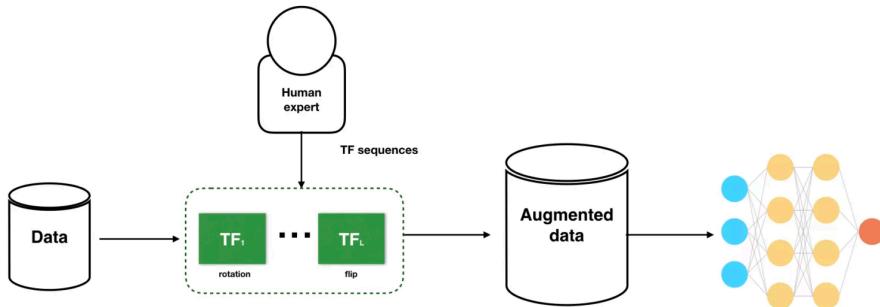
	bill_len	bill_depth	flipper_len	body_mass	island	sex	species
0	39.10000	18.70000	181.000000	3750.000000	2	1	0
1	39.50000	17.40000	186.000000	3800.000000	2	0	0
2	40.30000	18.00000	195.000000	3250.000000	2	0	0
3	43.92193	17.15117	200.915205	4201.754386	2	1	0
4	36.70000	19.30000	193.000000	3450.000000	2	0	0
...
339	55.80000	19.80000	207.000000	4000.000000	1	1	1
340	43.50000	18.10000	202.000000	3400.000000	1	0	1
341	49.60000	18.20000	193.000000	3775.000000	1	1	1
342	50.80000	19.00000	210.000000	4100.000000	1	1	1
343	50.20000	18.70000	198.000000	3775.000000	1	0	1

344 rows × 7 columns

سوال پنجم

نحوه عملکرد این روش چگونه است و تبدیل هایی که در آن استفاده میشود را شرح دهید. آیا از این روش برای داده های تست استفاده میشود؟ علت را شرح دهید.

افزایش داده ها مجموعه ای از تکنیک ها برای افزایش مصنوعی مقدار داده ها با تولید نقاط داده جدید از داده های موجود است. این شامل ایجاد تغییرات کوچک در داده ها یا استفاده از مدل های یادگیری عمیق برای تولید نقاط داده جدید است.



برای افزایش داده ها، ایجاد تغییرات ساده در داده های بصری رایج است. علاوه بر این، شبکه های (GAN) برای ایجاد داده های مصنوعی جدید استفاده می شوند. فعالیت های کلاسیک پردازش تصویر برای تقویت داده ها عبارتند از:

- لایه گذاری
- چرخش تصادفی
- مقیاس بندی مجدد،
- چرخش عمودی و افقی
- ترجمه (تصویر در جهت Y، حرکت می کند)
- ...

مدل های پیشرفته برای افزایش داده ها:

- آموزش خصمانه / یادگیری ماشین خصمانه: نمونه های متضاد را تولید می کند که مدل یادگیری ماشینی را مختل می کند و آنها را برای آموزش به مجموعه داده تزریق می کند.

- شبکه های (GAN): الگوریتم های GAN می توانند الگوهایی را از مجموعه داده های ورودی بیاموزند و به طور خودکار نمونه های جدیدی ایجاد کنند که شبیه داده های آموزشی است.

- Neural style transfer: مدل های انتقال سبک عصبی می توانند تصویر محظوظ و تصویر سبک را با هم ترکیب کنند و سبک را از محظوظ جدا کنند.

- یادگیری تقویتی: مدل های یادگیری تقویتی، عوامل نرم افزاری را برای دستیابی به اهداف و تصمیم گیری در یک محیط مجازی آموزش می دهند.

رایج ترین روش این است که افزایش داده ها را فقط برای نمونه های آموزشی اعمال کنیم. دلیل آن این است که ما می خواهیم با افزودن داده های بیشتر و تنوع بخشنیدن به مجموعه داده آموزشی، عملکرد تعمیم مدل خود را افزایش دهیم. با این حال، ما همچنانی می توانیم از آن در هنگام تست هم استفاده کنیم اما مرسوم نمی باشد.

سوال ششم

روش های downsampling و upsampling و ترکیبی را توضیح دهید.

در upsampling، مشاهدات کلاس اقلیت را به صورت تصادفی تکرار می کنیم تا سیگنال آن را تقویت کنیم. رایج ترین روش مونه گیری مجدد با جایگزینی است. این معادل ایجاد یک متغیر تصادفی است که دارای میانگین ۰ و واریانس ۱ باشد.

در downsampling، مشاهدات را به طور تصادفی از کلاس اکثربیت حذف می کنیم. بنابراین پس از upsampling یا downsampling، مجموعه داده با تعداد مشاهدات یکسان در هر کلاس متعادل می شود. نتیجه نهایی upsampling و downsampling، حفظ توزیع دادهها است. یکی از راههای کاهش واریانس در یک مجموعه داده به منظور down-sample downsampling دادهها است.

شامل افزایش وضوح یا فرکانس نقاط داده در یک مجموعه داده است. این را می توان با افزودن نقاط داده جدید بین نقاط داده موجود یا با درون یابی مقادیر برای پر کردن شکاف های داده به دست آورد. upsampling اغلب برای افزایش دقت داده ها استفاده می شود. به عنوان مثال، در دادههای سری زمانی، از نمونهبرداری مجدد می توان برای افزایش فراوانی نقاط داده برای ثبت الگوهای زمانی دقیق تر استفاده کرد. در پردازش تصویر می توان از نمونه برداری برای افزایش وضوح تصاویر استفاده کرد و در نتیجه سطح جزئیات بالاتری را به همراه داشت.

شامل کاهش وضوح یا فرکانس نقاط داده در یک مجموعه داده است. این را می توان با کاهش تعداد نقاط داده یا با تجمیع نقاط داده به سطح کمتری از دانه بندی به دست آورد. نمونه برداری پایین اغلب برای کاهش پیچیدگی محاسباتی یا ساده کردن داده ها برای تجزیه و تحلیل استفاده می شود. به عنوان مثال، در دادههای سری زمانی، نمونهبرداری پایین می تواند برای جمع آوری نقاط داده از یک فرکانس بالاتر (به عنوان مثال، ساعتی) به یک فرکانس پایین تر (مثلاً روزانه) برای کاهش نیازهای محاسباتی استفاده شود. در پردازش تصویر، از نمونه برداری پایین می توان برای کاهش اندازه یا وضوح تصاویر استفاده کرد و در نتیجه سربار محاسباتی کمتری را به همراه داشت.

یکی از موارد استفاده رایج برای ترکیب کردن downsampling و upsampling، مدیریت عدم تعادل دادهها در یادگیری ماشین است. در سناریوهایی که کلاسها در یک مشکل طبقه‌بندی نامتعادل هستند، به این معنی که برخی از کلاسها نمونههای کمتری نسبت به بقیه دارند، می توان از تکنیکهای نمونه‌گیری مجدد برای متعادل کردن توزیع کلاس استفاده کرد. این می تواند شامل نمونه برداری از کلاس اقلیت برای افزایش نمایش آن در مجموعه داده، و/یا کاهش نمونه کلاس اکثربیت برای کاهش تسلط آن باشد. این به کاهش بایاس نسبت به طبقه اکثربیت در طول آموزش مدل کمک می کند.

سوال هفتم

در مورد روشهای smoteenn و smotetomek تحقیق کنید نحوه کار آنها را توضیح دهید. وجه اشتراک این دو روش چیست؟

(Synthetic Minority Over-sampling Technique) یک تکنیک محبوب است که در یادگیری ماشین برای رسیدگی به عدم تعادل کلاس استفاده می شود، که در آن یک کلاس به طور قابل توجهی نمونه های کمتری نسبت به کلاس دیگر دارد. smote با درون یابی بین نمونه های کلاس اقلیت موجود، نمونه های مصنوعی را برای کلاس اقلیت تولید می کند و به طور موثر نمونه های مصنوعی مشابه نمونه های کلاس اقلیت واقعی ایجاد می کند. این به متعادل کردن توزیع کلاس و بهبود عملکرد مدل های یادگیری ماشین، به ویژه برای طبقه اقلیت کمک می کند.

smote را می توان با تکنیک های دیگر ترکیب کرد تا اثربخشی آن افزایش یابد. دو ترکیب رایج عبارتند از smote با پیوندهای SMOTE-ENN و smoteenn با ویرایش شده نزدیکترین همسایگان (SMOTE-ENN) که به نام Tomek (smotetomek) شناخته می شود.

یک تکنیک ترکیبی است که تکنیک SMOTE oversampling را با پیوندهای Tomek ترکیب می کند که جفت نمونه هایی از کلاس های مختلف هستند که در فضای ویژگی به یکدیگر نزدیک هستند و نمونه های نویزدار یا مبهم در نظر گرفته می شوند. ابتدا SMOTE را برای تولید نمونه های مصنوعی برای کلاس اقلیت اعمال می کند و سپس پیوندهای Tomek را بین نمونه های کلاس اقلیت و اکثربیت شناسایی می کند. پیوندهای Tomek حذف می شوند، که به حذف نمونه های مبهم یا پر سر و صدا از مجموعه داده کمک می کند و جدایی بین کلاس های اقلیت و اکثربیت را بیشتر بهبود می بخشد.

یکی دیگر از تکنیک‌های ترکیبی است که SMOTE را با کمترین نمونه‌برداری ویرایش شده نزدیک‌ترین همسایگان (ENN) ترکیب می‌کند. پس از اعمال SMOTE برای تولید نمونه‌های مصنوعی برای کلاس اقلیت، از ENN برای نمونه‌برداری کمتر از نمونه‌های کلاس اکثیریت استفاده می‌شود. ENN نمونه‌های کلاس اکثیریت را که توسط نزدیک‌ترین همسایگان‌شان به اشتباه طبقه‌بندی شده‌اند شناسایی می‌کند و آنها را از مجموعه داده حذف می‌کند و به طور موثر نمونه‌های پر سرو صدا یا دارای برچسب اشتباه را حذف می‌کند. این به کاهش نویز در کلاس اکثیریت کمک می‌کند و جدایی کلاس را بیشتر می‌کند.

هر دو smoteenn و smotetomek تکنیک‌های موثری برای رسیدگی به عدم تعادل کلاس و بهبود عملکرد مدل‌های یادگیری ماشین هستند. انتخاب بین آنها به ویژگی‌های خاص داده‌ها و الزامات کار تجزیه و تحلیل یا مدل‌سازی بستگی دارد. آزمایش تکنیک‌های مختلف و ارزیابی تأثیر آن‌ها بر عملکرد مدل‌ها برای انتخاب بهترین روش برای یک مورد خاص بسیار مهم است.

هر دو smoteenn و smotetomek هر دو تکنیک‌های ترکیبی هستند که تکنیک SMOTE oversampling را با مراحل اضافی برای افزایش بیشتر اثربخشی آن در رسیدگی به عدم تعادل کلاس در مجموعه داده‌های یادگیری ماشین ترکیب می‌کنند. هدف هر دو تکنیک بهبود جداسازی کلاس‌ها و کاهش نویز در کلاس اکثیریت برای دستیابی به عملکرد بهتر در مدل‌های یادگیری ماشین است.

سوال هشتم

از داده‌ها آموزش یکی از کلاس‌های دیتاست داده شده ۹۰ درصد را حذف کنید. حال با استفاده از این دو روش غیر متعادل بودن دیتاست که با حذف کردن داده‌ها بوجود آورده‌یم را هندل کنید. (نیازی به پیاده‌سازی این دو روش نیست، و میتوانید از کتابخانه‌ها استفاده کنید).

```
dataset_aug = New_dataset2

dataset_aug = dataset_aug.drop(dataset_aug[dataset_aug['species'] == 0].sample(frac=.9).index)
dataset_aug
```

	bill_len	bill_depth	flipper_len	body_mass	island	sex	species
12	41.1	17.6	182.0	3200.0	2	0	0
16	38.7	19.0	195.0	3450.0	2	0	0
21	37.7	18.7	180.0	3600.0	0	1	0
32	39.5	17.8	188.0	3300.0	1	0	0
43	44.1	19.7	196.0	4400.0	1	1	0
...
339	55.8	19.8	207.0	4000.0	1	1	1
340	43.5	18.1	202.0	3400.0	1	0	1
341	49.6	18.2	193.0	3775.0	1	1	1
342	50.8	19.0	210.0	4100.0	1	1	1
343	50.2	18.7	198.0	3775.0	1	0	1

207 rows × 7 columns

```
print("befor removong:", Counter(New_dataset2['species']))
print("after removong:", Counter(dataset_aug['species']))

befor removong: Counter({0: 152, 2: 124, 1: 68})
after removong: Counter({2: 124, 1: 68, 0: 15})
```

```
from imblearn.combine import SMOTETomek

X = dataset_aug.iloc[:, :6]
y = dataset_aug['species']

print('Original dataset shape %s' % Counter(y))

Original dataset shape Counter({2: 124, 1: 68, 0: 15})

smt = SMOTETomek(random_state=42)
X_res, y_res = smt.fit_resample(X, y)
print('Resampled dataset shape %s' % Counter(y_res))

Resampled dataset shape Counter({2: 122, 0: 121, 1: 119})
```

```

from imblearn.combine import SMOTEENN

print('Original dataset shape %s' % Counter(y))

Original dataset shape Counter({2: 124, 1: 68, 0: 15})

smt2 = SMOTEENN(random_state=42)
X_res, y_res = smt2.fit_resample(X, y)
print('Resampled dataset shape %s' % Counter(y_res))

Resampled dataset shape Counter({2: 106, 1: 73, 0: 67})

```

سوال نهم

با استفاده از `StandardScaler` در `sklearn.preprocessing` اقدام به نرمال‌سازی داده‌ها کنید. مقدار واریانس و میانگین هر ستون را قبل و بعد از نرمال‌سازی ذکر کنید (دقت کنید که این نرمال‌سازی را بر روی برجسب‌ها (ستون `species`) انجام ندهید).

```
#before scaling
dataset.describe()
```

	bill_len	bill_depth	flipper_len	body_mass	island	sex	species
count	344.000000	344.000000	344.000000	344.000000	344.000000	344.000000	344.000000
mean	43.921930	17.151170	200.915205	4201.754386	0.662791	0.520349	0.918605
std	5.443643	1.969027	14.020657	799.613058	0.726194	0.500313	0.893320
min	32.100000	13.100000	172.000000	2700.000000	0.000000	0.000000	0.000000
25%	39.275000	15.600000	190.000000	3550.000000	0.000000	0.000000	0.000000
50%	44.250000	17.300000	197.000000	4050.000000	1.000000	1.000000	1.000000
75%	48.500000	18.700000	213.000000	4750.000000	1.000000	1.000000	2.000000
max	59.600000	21.500000	231.000000	6300.000000	2.000000	1.000000	2.000000

```

columns2 = ['island', 'bill_len', 'bill_depth', 'flipper_len', 'body_mass', 'sex']
scaler = StandardScaler()
dataset = pd.DataFrame(scaler.fit_transform(dataset), columns=columns2)
dataset_scaled = pd.concat([dataset, New_dataset2['species']], axis = 1)
dataset_scaled

```

	island	bill_len	bill_depth	flipper_len	body_mass	sex	species
0	-8.870812e-01	0.787743	-1.422488	-0.565789	1.844076	0.960098	0
1	-8.134940e-01	0.126556	-1.065352	-0.503168	1.844076	-1.041561	0
2	-6.663195e-01	0.431719	-0.422507	-1.192003	1.844076	-1.041561	0
3	-1.307172e-15	0.000000	0.000000	0.000000	1.844076	0.960098	0
4	-1.328605e+00	1.092905	-0.565361	-0.941517	1.844076	-1.041561	0
...
339	2.185186e+00	1.347208	0.434620	-0.252683	0.465028	0.960098	1
340	-7.762162e-02	0.482580	0.077484	-1.004139	0.465028	-1.041561	1
341	1.044584e+00	0.533440	-0.565361	-0.534479	0.465028	0.960098	1
342	1.265345e+00	0.940324	0.648902	-0.127440	0.465028	0.960098	1
343	1.154965e+00	0.787743	-0.208225	-0.534479	0.465028	-1.041561	1

344 rows × 7 columns

```
dataset_scaled.describe()
```

	island	bill_len	bill_depth	flipper_len	body_mass	sex	species
count	3.440000e+02	3.440000e+02	3.440000e+02	3.440000e+02	3.440000e+02	3.440000e+02	344.000000
mean	-7.849019e-16	3.905145e-16	-4.902409e-16	1.458781e-16	-1.610469e-16	-3.872871e-17	0.918605
std	1.001457e+00	1.001457e+00	1.001457e+00	1.001457e+00	1.001457e+00	1.001457e+00	0.893320
min	-2.174858e+00	-2.060444e+00	-2.065333e+00	-1.880837e+00	-9.140204e-01	-1.041561e+00	0.000000
25%	-8.548868e-01	-7.889322e-01	-7.796428e-01	-8.162745e-01	-9.140204e-01	-1.041561e+00	0.000000
50%	6.035444e-02	7.569585e-02	-2.796522e-01	-1.900612e-01	4.650279e-01	9.600978e-01	1.000000
75%	8.422188e-01	7.877425e-01	8.631834e-01	6.866374e-01	4.650279e-01	9.600978e-01	2.000000
max	2.884265e+00	2.211836e+00	2.148873e+00	2.627899e+00	1.844076e+00	9.600978e-01	2.000000

```
pca = PCA(n_components = 3)
```

```
pca.fit(X)
X = pca.transform(X)
df_X = pd.DataFrame(X)
```

```
df_X
```

	0	1	2
0	-2.267511	1.266175	-0.048591
1	-2.078927	-0.528109	0.842508
2	-2.143540	-0.409198	1.103311
3	-0.604432	1.186353	0.705630
4	-2.594861	-0.183224	0.554844
...
339	0.448975	2.107021	1.421763
340	-1.012636	-0.547222	0.739321
341	-0.379978	1.294431	0.598062
342	0.398441	1.669301	0.819717
343	-0.536183	0.003741	1.470134

344 rows × 3 columns

سوال یازدهم

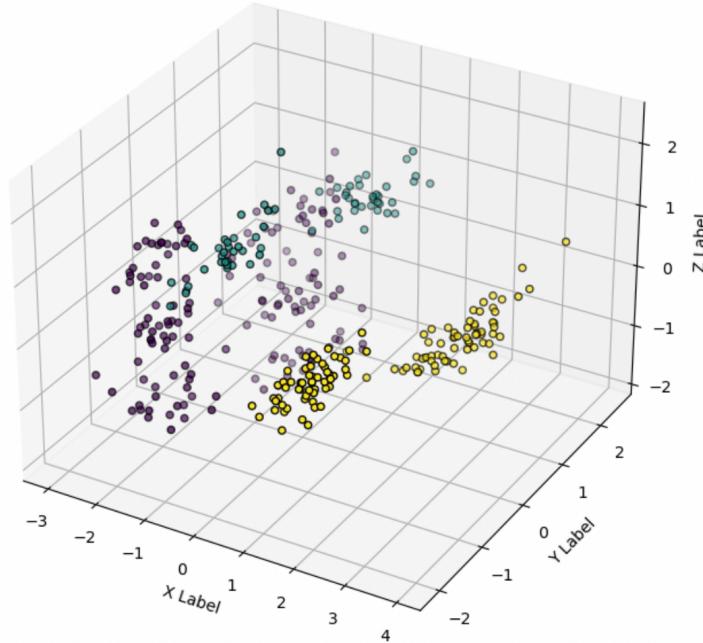
با استفاده از کتابخانه matplotlib داده‌های مجموعه داده را رسم کنید. دقت کنید برای ویژگی‌های حاصل از PCA استفاده کنید (رسم شکل به صورت سه‌بعدی خواهد شد). برای هر کلاس رنگ متفاوتی در نظر بگیرید.

```
fig = plt.figure(figsize=(10, 8))
ax = fig.add_subplot(projection='3d')

ax.scatter(X[:, 0], X[:, 1], X[:, 2], c=y, edgecolor="k")

ax.set_xlabel('X Label')
ax.set_ylabel('Y Label')
ax.set_zlabel('Z Label')

plt.show()
```



سوال دوازدهم

سوال دوازدهم - برای هر ۶ ویژگی ارائه شده در مجموعه داده، نمودار boxplot را رسم کنید (این کار را قبل از گام نرم‌السازی انجام دهید).

```
fig, ax = plt.subplots(3, 2, figsize=(9, 9))

ax[0, 0].boxplot(New_dataset2[[columns[0]]])
ax[0, 1].boxplot(New_dataset2[[columns[1]]])
ax[1, 0].boxplot(New_dataset2[[columns[2]]])
ax[1, 1].boxplot(New_dataset2[[columns[3]]])
ax[2, 0].boxplot(New_dataset2[[columns[4]]])
ax[2, 1].boxplot(New_dataset2[[columns[5]]])

ax[0, 0].title.set_text(columns[0])
ax[0, 1].title.set_text(columns[1])
ax[1, 0].title.set_text(columns[2])
ax[1, 1].title.set_text(columns[3])
ax[2, 0].title.set_text(columns[4])
ax[2, 1].title.set_text(columns[5])
```

