به نام خدا

نام و نام خانوادگی: امیرحسین عبدیان

شماره دانشجویی:4021661210

پرسش1 بخش1 :

طبقه‌بندی (Classification)، همان‌طور که از نامش مشخص است، یعنی «دسته‌بندی چیزها» به گروه‌های کوچک‌تر؛ ولی این بار توسط یک ماشین! اگر فکر می‌کنید خیلی جذاب نیست، فقط تصور کنید کامپیوتر می‌تواند تفاوت شما را با یک ناشناس تشخیص دهد یا تمایز یک سیب‌زمینی را از یک گوجه‌فرنگی بفهمد یا فرق بین نمره‌ی A و F را بشناسد.

طبقه‌بندی جزو انواع یادگیری ماشین تحت نظارت است که در آن از داده‌هایی با برچسب برای آموزش استفاده می‌شود. در یادگیری ماشین و آمار، موضوع طبقه‌بندی این است که بر اساس داده‌های آموزشی که قبلاً دسته‌بندی شده‌اند، بفهمیم یک داده جدید به کدام دسته یا گروه از میان مجموعه‌ها متعلق است.

طبقه‌بندی، یک فرایند برای قرار دادن داده‌ها یا اشیاء در دسته‌های معینی است که بر اساس ویژگی‌های آن‌ها تعیین می‌شود. در یادگیری ماشین، طبقه‌بندی یک روش از یادگیری‌ نظارت شده است، جایی که الگوریتم بر روی مجموعه‌ای از داده‌های دارای برچسب آموزش می‌بیند تا بتواند دسته داده‌های جدید و ناشناخته را پیش‌بینی کند. هدف از طبقه‌بندی این است که مدلی بسازیم که قادر باشد به مشاهدات جدید، بر اساس ویژگی‌های آنها، برچسب مناسبی اختصاص دهد. به عنوان مثال، فرض کنید مدلی را بر روی تصاویری که به عنوان سگ یا گربه برچسب‌گذاری شده‌اند آموزش داده‌ایم؛ سپس از این مدل استفاده می‌شود تا تصاویر جدیدی که تا به حال ندیده‌ است را بر اساس ویژگی‌هایی مثل رنگ، بافت و شکل طبقه‌بندی کند.

در طبقه بندی دو کلاسه هدف این است که ورودی را در یکی از دو دسته قرار دهیم. مثلاً، بر اساس وضعیت سلامت فرد، تصمیم می‌گیریم که آیا او به یک بیماری خاص مبتلا است یا نه.

در طبقه بندی چند کلاسه هدف این است که ورودی را در یکی از چند دسته مختلف قرار دهیم. به عنوان مثال، با داشتن اطلاعات مختلف درباره گونه‌های گل‌ها، می‌خواهیم بفهمیم مشاهده ما به کدام گونه متعلق است.

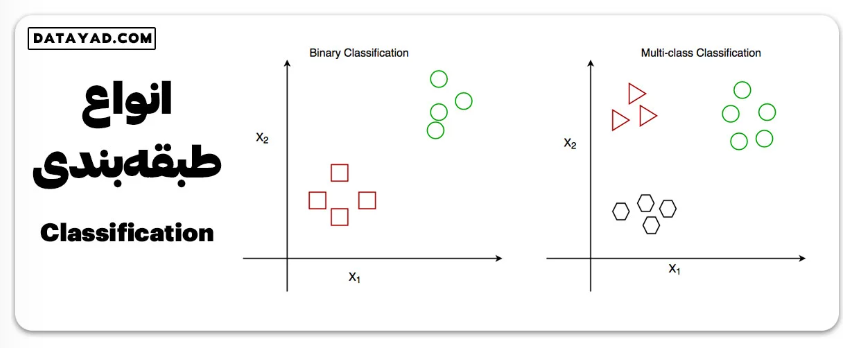
در مدل طبقه بندی خطی یک مرز تصمیم ساده و خطی بین دسته‌ها ترسیم می‌کنند. به خاطر سادگی خود، از نظر محاسباتی بسیار سریع عمل می‌کنند. مثال‌هایی از این نوع:

رگرسیون لجستیک

ماشین‌های بردار پشتیبان با کرنل خطی

پرسپترون تک‌لایه

طبقه‌بندی‌کننده گرادیان تصادفی (SGD)



چند کلاسه

دو کلاسه

ارزیابی مدل‌ طبقه‌بندی

ارزیابی مدل طبقه‌بندی گام مهمی در یادگیری ماشین است چرا که کمک می‌کند تا کارایی و توانایی تعمیم مدل روی داده‌های جدید و ندیده‌شده را سنجید. چندین معیار و روش برای ارزیابی یک مدل طبقه‌بندی وجود دارد که بسته به مسئله و نیازهای خاص مورد استفاده قرار می‌گیرند. معیارهای متداول ارزیابی به شرح زیر هستند:

دقت طبقه‌بندی (accuracy)

نسبت تعداد نمونه‌هایی که به درستی طبقه‌بندی شده‌اند به تعداد کل نمونه‌ها در مجموعه آزمون. این معیار ساده و قابل درک است اما در مجموعه‌ داده‌های نامتوازن، که کلاس اکثریت بر دقت حاکم است، ممکن است گمراه کننده باشد.

ماتریس درهم‌ریختگی

جدولی که تعداد مثبت‌های صحیح، منفی‌های صحیح، مثبت‌های غلط و منفی‌های غلط را برای هر کلاس نشان می‌دهد و می‌توان با استفاده از آن، معیارهای ارزیابی مختلفی را محاسبه کرد.

دقت (تشابه) و بازخوانی (Precision and Recall)

دقت نسبت تعداد مثبت‌های صحیح به تعداد کل مثبت‌های پیش‌بینی شده را اندازه‌گیری می‌کند، در حالی که بازخوانی، نسبت مثبت‌های صحیح به تعداد کل مثبت‌های واقعی را اندازه‌گیری می‌کند. این معیارها در مواردی که یک کلاس از دیگری مهم‌تر است یا زمانی که میان مثبت‌های غلط و منفی‌های غلط تعادلی وجود دارد، مفید هستند.

امتیاز F1

میانگین هارمونیکی از دقت و بازخوانی، که به صورت محاسبه می‌شود. این معیار برای مجموعه‌های داده نامتوازن که هر دوی دقت و بازخوانی مهم هستند، مفید است.

منحنی ROC و AUC

منحنی مشخصه عملکرد (ROC) نموداری از نرخ مثبت‌های صحیح (بازخوانی) در مقابل نرخ مثبت‌های غلط (۱-خصوصیت) برای مقادیر آستانه مختلف تابع تصمیم طبقه‌بندی است. مساحت زیر منحنی (AUC) کارایی کلی طبقه‌بندی را اندازه‌گیری می‌کند، با مقادیری که از ۰.۵ (حدس تصادفی) تا ۱ (طبقه‌بندی کامل) متغیر می‌شوند.

اعتبارسنجی متقابل (Cross-Validation)

روشی که داده‌ها را به چندین قسمت تقسیم می‌کند و مدل را برای هر قسمت آموزش می‌دهد، در حالی که روی بقیه آنها آزمون انجام می‌شود، تا بتوان به تخمین موثق‌تری از کارایی مدل دست یافت.

طبقه‌بندی چگونه کار می‌کند؟

در طبقه‌بندی، هدف اصلی این است که یک مدل روی داده‌هایی با برچسب آموزش داده شود تا الگوها و ارتباطات بین ورودی‌ها و برچسب‌های آن‌ها را فهمیده و بیاموزد. وقتی مدل آموزش داده شود، می‌توانیم از آن برای پیش‌بینی برچسب‌ها برای داده‌های جدید و ندیده استفاده کنیم.

فرآیند طبقه‌بندی شامل گام های زیر است:

۱- فهم مسئله

قبل از شروع طبقه‌بندی، باید به درستی مسئله‌ای که می‌خواهید حل کنید را متوجه شوید. کدام برچسب‌ها را می‌خواهید پیش‌بینی کنید؟ چطور داده‌های ورودی با این برچسب‌ها ارتباط دارند؟

به عنوان مثال، فرض کنید می‌خواهیم بر اساس ۷ ویژگی مستقل پیش‌بینی کنیم که آیا یک بیمار، بیماری خاصی دارد یا نه. این به این معناست که دو نتیجه ممکن داریم:

بیمار مبتلا به بیماری است، یعنی “صحیح True”.

بیمار مبتلا نیست، یعنی “غلط False”.

این یک مسئله طبقه‌بندی دودویی است.

۲- آماده‌ سازی داده

وقتی با مسئله آشنا شدید، مرحله بعد آماده‌کردن داده‌هاست. این مرحله شامل جمع‌آوری، پیش‌ پردازش داده ها و تقسیم داده‌ ها به دسته‌های آموزش، اعتبارسنجی و تست می‌شود. اینجا، داده‌ها تمیز و به شکل مناسب برای الگوریتم طبقه‌بندی تبدیل می‌شوند.

X : ویژگی مستقلی است که به صورت ماتریسی با ابعاد N\*M می‌باشد که N تعداد مشاهدات و M تعداد ویژگی‌ها را نمایان می‌کند.

Y : برداری با طول N است که برای هر مشاهده، کلاس پیش‌بینی شده را نشان می‌دهد.

۳- استخراج ویژگی‌ها

ویژگی‌های مؤثر و مرتبط از داده‌ها استخراج می‌شوند تا بین کلاس‌ها تمایز بیافرینیم.

به عنوان مثال، فرض کنید داده‌های ورودی X، شامل ۷ ویژگی مستقل است، اما فقط ۵ ویژگی از آن‌ها بر روی نتیجه تاثیرگذارند و ۲ ویژگی باقی‌مانده همبستگی کم یا بی‌معنی دارند. در این حالت، ما فقط از این ۵ ویژگی برای آموزش مدل استفاده می‌کنیم.

۴- انتخاب مدل

برای طبقه‌بندی، انواع مختلفی از مدل‌ها مانند رگرسیون لجستیک، درخت تصمیم، ماشین‌های بردار پشتیبان و شبکه‌های عصبی در دسترس هستند. انتخاب مدل مناسب برای مسئله شما، با توجه به حجم و پیچیدگی داده‌ها و منابع محاسباتی که دارید، اهمیت دارد.

۵- آموزش مدل

پس از انتخاب یک مدل، مرحله بعدی آموزش آن با داده‌های آموزشی است. در این فرآیند، پارامترهای مدل به نحوی تغییر می‌یابند تا اختلاف بین برچسب‌های پیش‌بینی شده و برچسب‌های واقعی کلاس در داده‌های آموزشی کاهش یابد.

۶- ارزیابی مدل

پس از آموزش مدل، لازم است که عملکرد آن روی یک مجموعه اعتبارسنجی بررسی شود. این ارزیابی به شما دید خوبی از عملکرد مدل بر روی داده‌های جدید و ناشناخته می‌دهد.

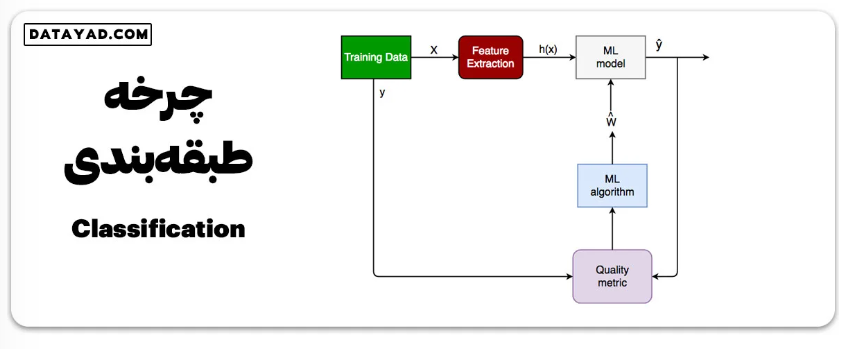
لگاریتم خطا یا خطای انتروپی متقابل، ماتریس درهم‌ریختگی، دقت، بازخوانی و منحنی AUC-ROC از معیارهای کیفیتی هستند که برای اندازه‌گیری عملکرد مدل استفاده می‌شوند.

۷- تیونینگ مناسب مدل

اگر عملکرد مدل مطلوب نبود، می‌توانید با تنظیم پارامترها یا امتحان یک مدل متفاوت، آن را بهینه‌سازی کنید.

۸- استقرار مدل

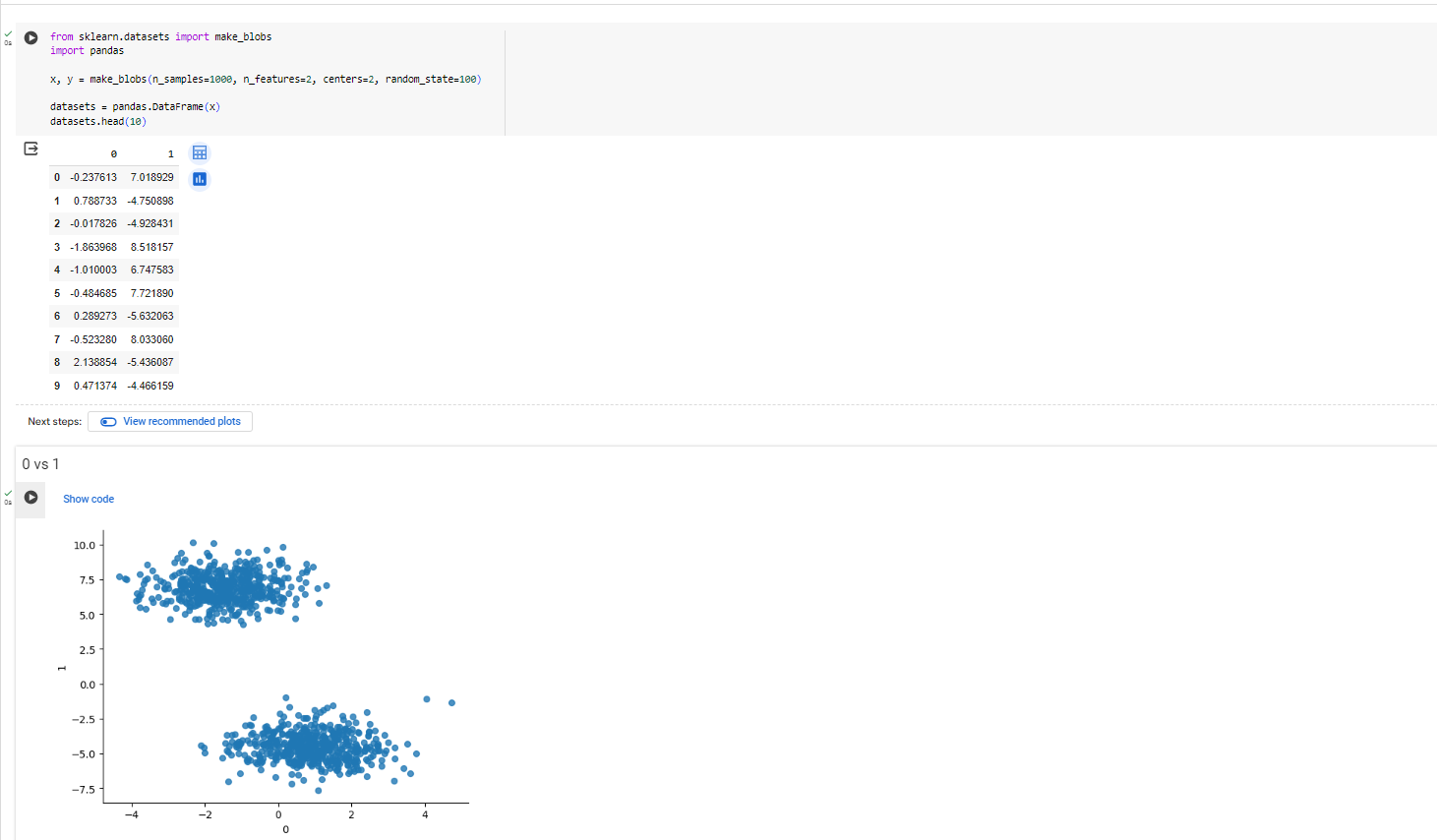
در نهایت، هنگامی که از عملکرد مدل راضی شدیم، می‌توانیم آن را استقرار دهیم تا پیش‌بینی‌هایی روی داده‌های جدید انجام دهد. این مدل می‌تواند برای حل مشکلات واقعی استفاده شود.



حال برای تعیین نوع طبقه بندی از حالت دو کلاسه به چند کلاسه در مرحله فهم مسئله باید مسئله را درست فهمید و مشخص نمایید برای حل مسئله به چند کلاس نیاز دارید و در قسمت آماده سازی داده ها و استخراج ویژگی ها مسئله را با با توجه به فهم حل آن آماده سازی نمایید.

پرسش1 بخش2 :

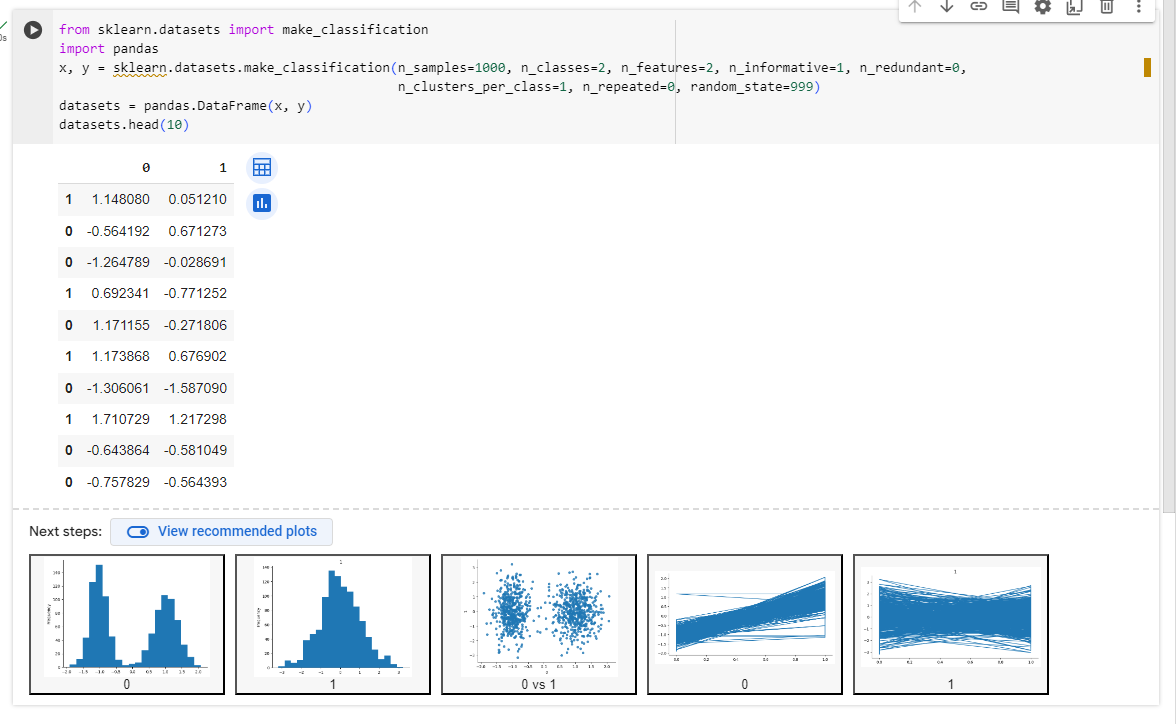
با استفاده از make\_blobs از کتابخانه sklearn.datasets با مقداردهی n\_samples برابر 1000 و n\_features که ویژگی هاست برابر 2 و centers برابر 2 و همچنین با تعیین random\_state مقداردهی، و داده ها را تصادفی تولید میکنیم. که با تغییر مقادیر n\_features و centers میتوان پیچیدگی ایجاد کرد.



البته میتوانیم با استفاده از کتابخانه sklearn.datasets و متد make\_classification، دیتاست مورد نظر را با 1000 نمونه و دو کلاس و دو ویژگی ایجاد میکنیم و با استفاده از کتابخانه pandas نمایش میدهیم.

در این متد n\_sample تعداد نمونه، n\_classes تعداد کلاس ها و یا برچسب ها و n\_features مربوط به تعداد کل ویژگی ها و n\_informative مربوط به ویژگی های آموزنده هر کلاس و n\_redundant مربوط به ویژگی هایی به عنوان ترکیب خطی تصادفی از ویژگی های اطلاعاتی تولید می شوند و n\_repeated مربوط به تعداد ویژگی‌های تکراری که به‌طور تصادفی از ویژگی‌های اطلاعاتی و اضافی استخراج شده‌اند.

و در آخر n\_clusters\_per\_class که مربوط به تعداد خوشه ها در هر کلاس است.



برای چالش برانگیزترن شدن با فرض بر دو کلاس بودن، میتوان بر روی ویژگیها، کلاستر ، ویژگی های آموزنده، ویژگی های تصادفی و ویژگی های تکراری مقداردهی بالاتری داشت.

همچنین برای جذاب کردن خروجی میتوان از کد زیر نیز بهره برد :



پرسش1 بخش3 :

ابتدا با توجه به نمونه ایجاد شده در سوال قبل میخواهیم دو طبقه بندی خطی را ایجاد نماییم. تصویر زیر نمونه ایجاد شده در سوال قبل است:



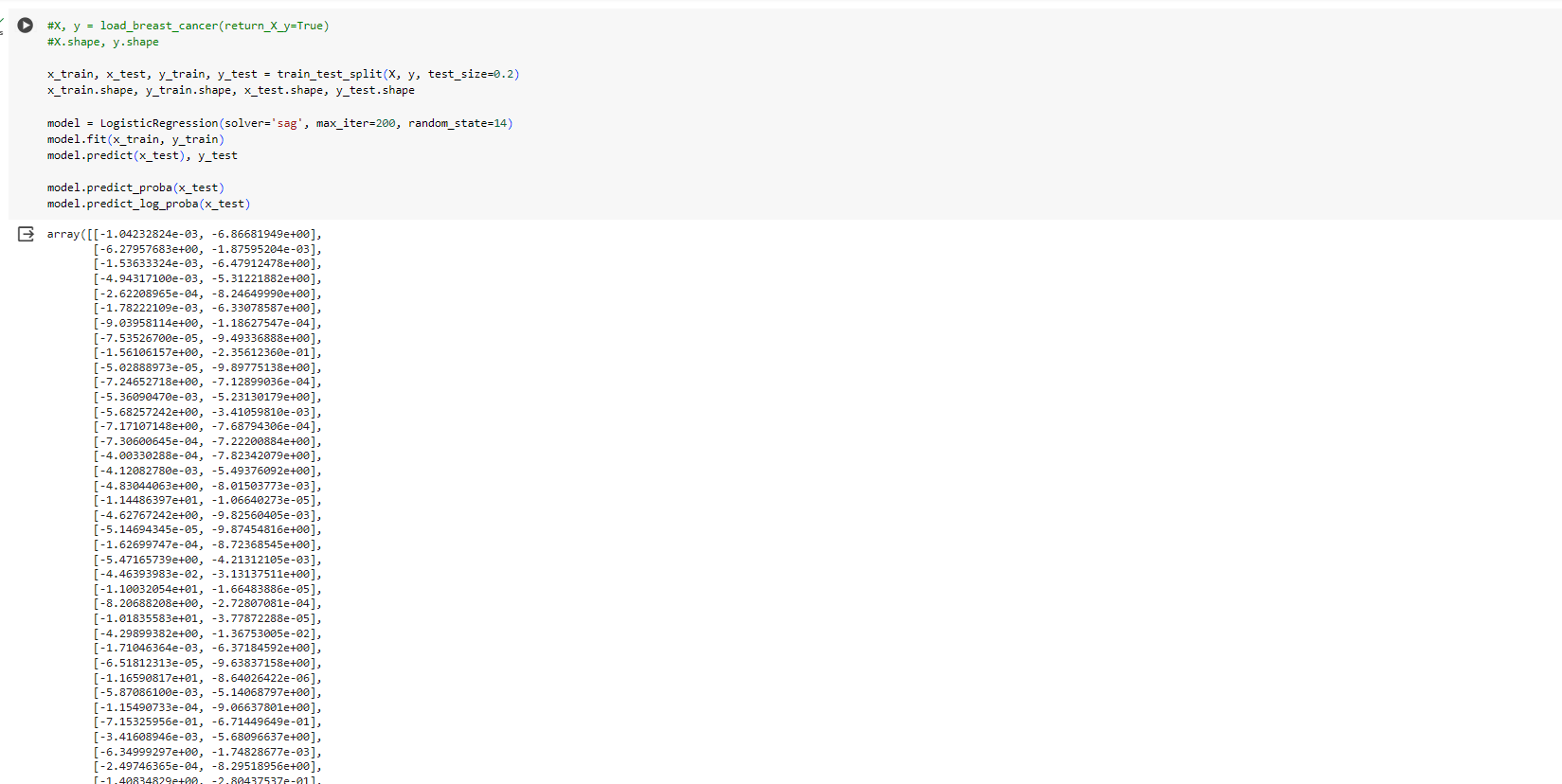
حالا با استفاده از load\_breast\_cancer از کلاس sklearn.datasets اقدام به ایجاد اولین طبقه بندی خطی مینماییم. که این یک مجموعه داده طبقه بندی باینری کلاسیک و آسان است. اگر True باشد، به جای یک شی Bunch، (data, target) را برمی گرداند.

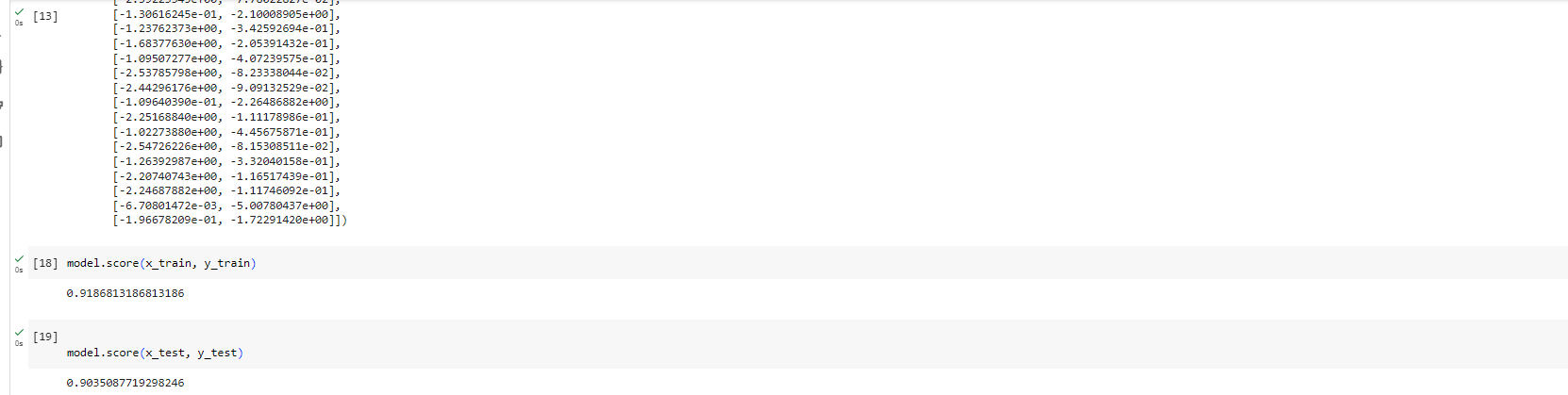
اگر True باشد، داده ها یک DataFrame از نوع pandas است که شامل ستون هایی با data type (عددی) مناسب است. هدف وابسته به تعداد ستون های هدف، یک DataFrame یا سری pandas است. اگر return\_X\_y مقدار True باشد، آنگاه (داده، هدف) DataFrames از نوع pandas یا Series خواهد بود .

با استفاده از تابع train\_test\_split از کلاس sklearn.model\_selection به منظور نمونه گیری متغیرهای x\_train، y\_train ، x\_test و y\_test رو مقداردهی مینماییم.

سپس با استفاده از LogisticRegression از کتابخانه sklearn.linear\_model یک متغیر به نام model ایجاد میکنیم که یک الگوریتم آموزش میباشد که متغیرهای x\_train، y\_train ، x\_test و y\_test را تنظیم مینماییم.

تصویر زیر روش اول در طبقه بندی خطی را نمایش میدهد:





حالا با تابع SGDClassifier از کلاس sklearn.linear\_model دومین طبقه بندی خطی را اجرا میکنیم. این تخمین‌گر مدل‌های خطی منظم‌شده را با یادگیری گرادیان نزولی تصادفی (SGD) پیاده‌سازی می‌کند: گرادیان تلفات در هر نمونه در یک زمان تخمین زده می‌شود و مدل در طول مسیر با یک برنامه قدرت کاهشی (معروف به نرخ یادگیری) به‌روزرسانی می‌شود. SGD امکان یادگیری کوچک (آنلاین/خارج از هسته) را از طریق روش partial\_fit می دهد.

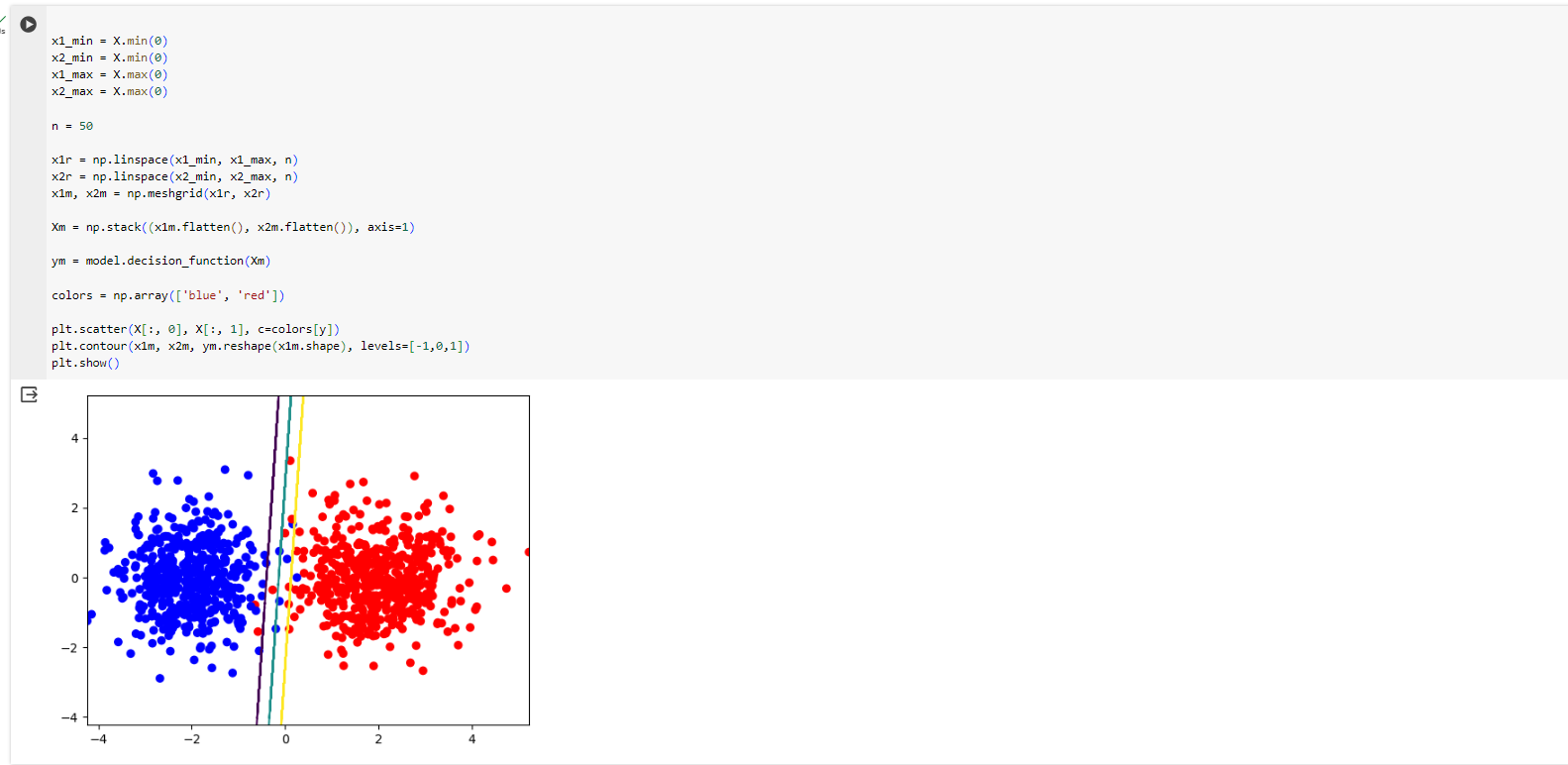
این پیاده سازی با داده هایی کار می کند که به صورت آرایه های متراکم یا پراکنده از مقادیر ممیز شناور برای ویژگی ها نمایش داده می شوند. مدل مناسب را می توان با پارامتر ضرر کنترل کرد. به طور پیش فرض، با یک ماشین بردار پشتیبان خطی (SVM) مطابقت دارد.

تصویر زیر روش دوم در طبقه بندی خطی را نمایش میدهد:



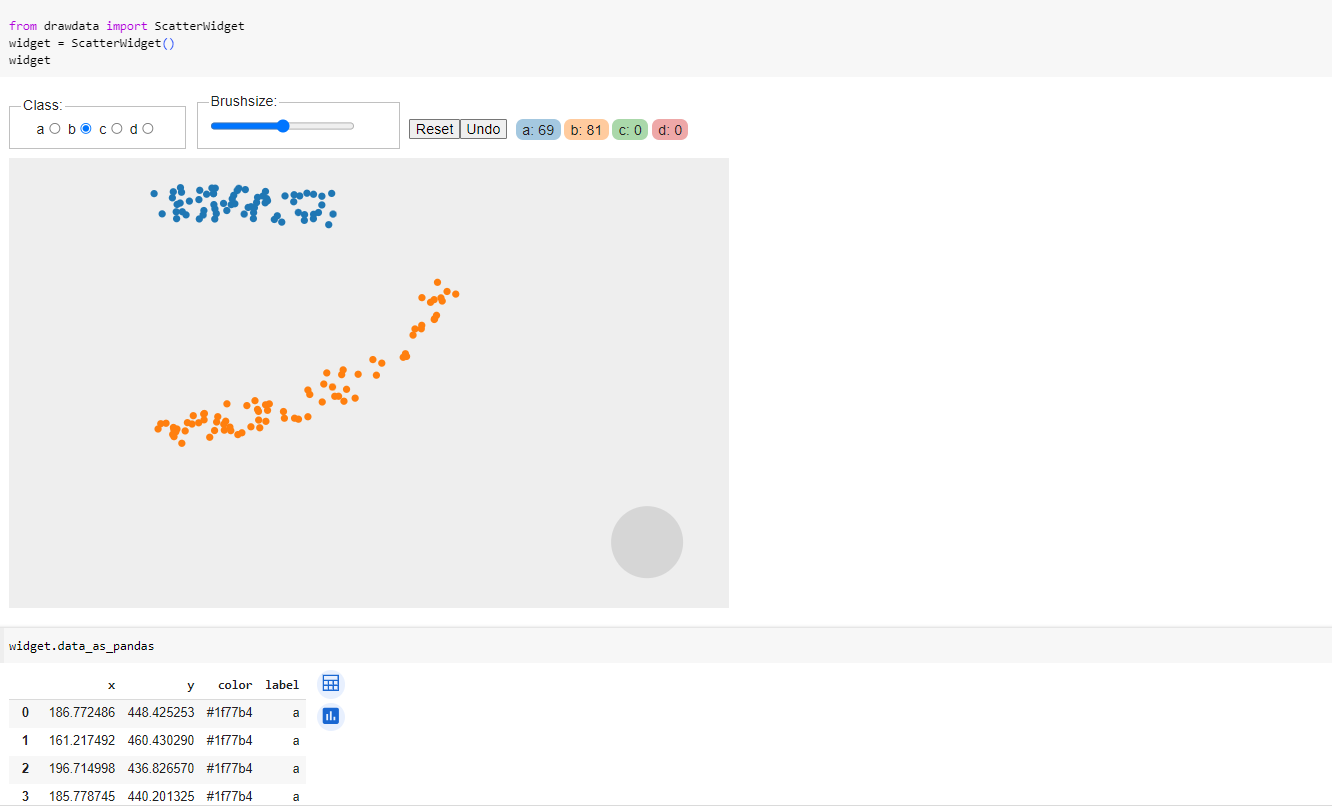
پرسش1 بخش4 :

برای مشخص کردن مرز و نواحی تصمیم گیری ابتدا مقادیر MIN و MAX را مشخص با استفاده از linspace از کتابخانه numpy که یک تابع داخلی در کتابخانه پایتون است. برای ایجاد یک توالی با فاصله مساوی در یک بازه زمانی مشخص استفاده می شود. سپس شما از meshgrid() برای تبدیل بردارهای 1 بعدی که محورها را به آرایه های دوبعدی نشان می دهند، استفاده می کنید. سپس می توانید از آن آرایه ها به جای متغیرهای x و y در معادله ریاضی استفاده کنید. از آنجایی که X یک آرایه 2 بعدی NumPy است، وقتی از X در np استفاده می کنید، یک آرایه دو بعدی دریافت خواهید کرد. یک تابع decicioun\_function می‌تواند برای استدلال در ورودی‌ها برای رسیدن به خروجی‌ها با اعمال یک مجموعه قوانین یا سایر توابع تصمیم‌گیری احضار شود.



قسمت 5:

با توجه به راهنمای داخل github ابتدا شروع به نصب کتابخانه drawdata در گوگل کولب مینماییم. سپس شروع به نوشتن کدها مینماییم.



سوال دو

قسمت 1 :

نمای کلی مجموعه داده:

مجموعه داده Case Western Reserve University (CWRU) به طور گسترده مورد استناد قرار گرفته و برای ارزیابی رویکردهای تشخیصی مختلف برای ارزیابی وضعیت سلامت و پیش‌بینی طول عمر مفید باقیمانده استفاده می‌شود.

بلبرینگ ها اجزای مکانیکی حیاتی در ماشین ها هستند و خرابی آنها می تواند بر اجزای مجاور یا کل ماشین تاثیر بگذارد.

این مجموعه داده که توسط گروه تحقیقاتی پروفسور کنت لوپارو جمع آوری شده است، امکان دسترسی به تست های بلبرینگ را برای یاتاقان های معمولی و معیوب فراهم می کند.

پایه تست شامل یک موتور 2 اسب بخار، یک مبدل/رمزگر گشتاور، یک دینامومتر و الکترونیک کنترل 1 است.

اهداف:

هدف اصلی مجموعه داده CWRU تسهیل تحقیق و توسعه در تشخیص و پیش‌آگهی خطا است.

هدف محققان ایجاد مدل‌های دقیقی است که می‌تواند عیوب یاتاقان‌ها را زود تشخیص داده و طول عمر مفید باقی‌مانده آن‌ها را پیش‌بینی کند.

با دستیابی به این اهداف، می توان شیوه های تعمیر و نگهداری را بهینه کرد، زمان خرابی را کاهش داد و از خرابی های فاجعه بار جلوگیری کرد.

امکانات:

مجموعه داده شامل سیگنال‌های ارتعاشی است که از یاتاقان‌ها تحت شرایط عملیاتی مختلف به دست می‌آیند.

ویژگی های کلیدی عبارتند از:

روش‌های مبتنی بر سیگنال: این رویکردها سیگنال‌های ارتعاشی خام را مستقیماً تحلیل می‌کنند. محققان ویژگی های مرتبط را از حوزه زمان یا فرکانس استخراج می کنند.

روش‌های مبتنی بر مدل: این روش‌ها از مدل‌های ریاضی برای شبیه‌سازی رفتار بلبرینگ استفاده می‌کنند. آنها اغلب به دانش فیزیک بلبرینگ و دینامیک سیستم نیاز دارند.

روش‌های مبتنی بر داده: این رویکردها از تکنیک‌های یادگیری ماشینی برای یادگیری الگوها مستقیماً از داده‌ها بدون تکیه بر مدل‌های صریح استفاده می‌کنند.

حالت ها:

مجموعه داده CWRU دسترسی به حالت های مختلف عملکرد بلبرینگ را فراهم می کند، از جمله:

عملکرد عادی: سیگنال های بلبرینگ های سالم در شرایط عملیاتی معمولی.

عملکرد معیوب: سیگنال هایی از یاتاقان های دارای عیوب خاص (به عنوان مثال، خطای مسابقه داخلی، خطای مسابقه بیرونی یا خطای توپ).

شرایط متغیر بار و سرعت: سیگنال هایی که هنگام تنظیم پارامترهای بار و سرعت جمع آوری می شوند.

چالش ها و مسیرهای تحقیق:

محققان به کشف رویکردهای جدید، از جمله یادگیری عمیق (DL) همراه با روش‌های همجوشی ادامه می‌دهند.

در حالی که بسیاری از رویکردها با استفاده از مجموعه داده CWRU به دقت بالایی دست می یابند، نیاز به معیارهای ارزیابی مناسب تر و مجموعه داده های اضافی برای افزایش ارزیابی عملکرد وجود دارد.

به طور خلاصه، مجموعه داده CWRU به عنوان معیاری برای انجام تحقیقات تشخیص خطا، تقویت پیشرفت‌ها در نظارت بر سلامت ماشین آلات و شیوه‌های نگهداری عمل می‌کند.

قسمت 1-2 :

تولید داده:

ما دو کلاس داریم که هر کدام حداقل 100 نمونه دارند (که با M مشخص می شود).

طول هر نمونه حداقل 200 نقطه داده (که با N نشان داده می شود) دارد.

تشکیل ماتریس:

برای هر کلاس، ماتریسی ایجاد می کنیم که در آن هر ردیف با یک نمونه مطابقت دارد.

ماتریس دارای ابعاد N×M خواهد بود.

برچسب گذاری:

برچسب های مربوطه را به هر نمونه در هر دو کلاس اختصاص دهید.

در اینجا یک رویکرد سطح بالا برای دستیابی به این وجود دارد:

مرحله 1: داده های مصنوعی تولید کنید

فرض کنید دو کلاس داریم: کلاس A و کلاس B.

برای هر کلاس، نمونه‌های داده مصنوعی با مقادیر تصادفی تولید کنید. طول این نمونه ها باید حداقل 200 نقطه داده باشد.

این روند را تا زمانی که حداقل 100 نمونه برای هر کلاس داشته باشید تکرار کنید.

مرحله 2: ایجاد ماتریس

برای هر کلاس، نمونه ها را در ردیف های یک ماتریس مرتب کنید.

ابعاد ماتریس N × M خواهد بود (که در آن N طول هر نمونه و M تعداد نمونه ها است).

مرحله 3: برچسب ها را اختصاص دهید

ماتریس ها را برچسب گذاری کنید:

ماتریس A: نمونه هایی از کلاس A

ماتریس B: نمونه هایی از کلاس B

مثال (ساده شده):

فرض کنید به طور تصادفی داده هایی را برای کلاس A و کلاس B تولید کرده ایم:

کلاس A: 100 نمونه، هر کدام با 200 نقطه داده

کلاس B: 100 نمونه، هر کدام با 200 نقطه داده

ما دو ماتریس ایجاد می کنیم:

ماتریس A: 200 × 100 (هر ردیف نشان دهنده نمونه ای از کلاس A است)

ماتریس B: 200 × 100 (هر ردیف نشان دهنده نمونه ای از کلاس B است)

قسمت 3-2:

اهمیت مهندسی ویژگی در یادگیری ماشین

مهندسی ویژگی نقش مهمی در ساخت مدل‌های یادگیری ماشینی موثر دارد. در اینجا چند دلیل برای ضروری بودن آن وجود دارد:

عملکرد بهتر مدل:

ویژگی های به خوبی مهندسی شده می تواند به طور قابل توجهی دقت و تعمیم مدل را بهبود بخشد.

با انتخاب ویژگی های مرتبط و تبدیل آنها به طور مناسب، ما توانایی مدل را برای گرفتن الگوهای اساسی در داده ها افزایش می دهیم.

کاهش ابعاد:

مهندسی ویژگی به کاهش ابعاد مجموعه داده کمک می کند.

داده‌های با ابعاد بالا می‌توانند به بیش از حد برازش و افزایش پیچیدگی محاسباتی منجر شوند.

تکنیک هایی مانند انتخاب ویژگی و استخراج به حفظ اطلاعات ضروری و کاهش تعداد ویژگی ها کمک می کند.

رسیدگی به داده های از دست رفته:

مهندسی ویژگی به ما این امکان را می دهد که مقادیر از دست رفته را به طور موثر مدیریت کنیم.

وارد کردن داده های از دست رفته یا ایجاد ویژگی های جدید بر اساس ویژگی های موجود می تواند استحکام مدل را بهبود بخشد.

گنجاندن دانش دامنه:

مهندسان و کارشناسان حوزه می‌توانند ویژگی‌هایی ایجاد کنند که با درک آنها از مشکل همخوانی داشته باشد.

تلفیق دانش خاص دامنه اغلب منجر به نمایش بهتر ویژگی ها می شود.

مقیاس بندی و عادی سازی ویژگی:

ویژگی های مقیاس بندی تضمین می کند که محدوده های مشابهی دارند.

الگوریتم هایی مانند نزول گرادیان زمانی که ویژگی ها به طور مناسب مقیاس بندی شوند سریعتر همگرا می شوند.

رمزگذاری متغیرهای دسته بندی:

تبدیل ویژگی‌های طبقه‌بندی به نمایش‌های عددی (به عنوان مثال، رمزگذاری یک‌طرفه) مدل‌ها را قادر می‌سازد تا آنها را به طور مؤثر پردازش کنند.

تعامل ویژگی ها و ویژگی های چند جمله ای:

ایجاد اصطلاحات تعاملی یا ویژگی های چند جمله ای، روابط پیچیده بین متغیرها را به تصویر می کشد.

به عنوان مثال، ترکیب قد و وزن برای ایجاد یک ویژگی BMI.

انتخاب ویژگی:

شناسایی مرتبط ترین ویژگی ها نویز را کاهش می دهد و قابلیت تفسیر مدل را بهبود می بخشد.

تکنیک ها شامل حذف ویژگی های بازگشتی، منظم سازی L1 و اطلاعات متقابل است.

حال بیایید اطلاعات را از جدول استخراج کنیم:

ویژگی های مجموعه داده (بخش "2-a")

از جدول، من بر روی یک ویژگی مجموعه داده مربوط به بخش 2-a تمرکز خواهم کرد:

نام مجموعه داده: مجموعه داده بیماری قلبی UCI

توضیحات: این مجموعه داده حاوی ویژگی های مربوط به تشخیص بیماری قلبی است، از جمله ویژگی هایی مانند سن، سطح کلسترول، فشار خون و اندازه گیری ECG.

تعداد موارد: تقریبا 303 نمونه.

تعداد ویژگی ها: ویژگی های متعدد، شامل متغیرهای عددی و دسته بندی.

متغیر هدف: طبقه بندی باینری (وجود یا عدم وجود بیماری قلبی).

تشکیل مجموعه داده جدید

برای یک مجموعه داده جدید، بیایید یک مجموعه داده فرضی مرتبط با پیش‌بینی ریزش مشتری ایجاد کنیم:

نام مجموعه داده: مجموعه داده مشتریان مخابراتی

توضیحات: این مجموعه داده رفتار مشتری را در یک شرکت مخابراتی، شامل ویژگی‌هایی مانند مدت تماس، نوع قرارداد، استفاده از داده و شکایات مشتری نشان می‌دهد.

تعداد موارد: 1000 سابقه مشتری.

تعداد امکانات: 10 ویژگی.

متغیر هدف: طبقه‌بندی باینری (خرد شده یا نخورده).

قسمت 4-2:

اهمیت مهندسی ویژگی در یادگیری ماشین

مهندسی ویژگی نقش مهمی در ساخت مدل‌های یادگیری ماشینی موثر دارد. در اینجا چند دلیل برای ضروری بودن آن وجود دارد:

عملکرد بهتر مدل:

ویژگی های به خوبی مهندسی شده می تواند به طور قابل توجهی دقت و تعمیم مدل را بهبود بخشد.

با انتخاب ویژگی های مرتبط و تبدیل آنها به طور مناسب، ما توانایی مدل را برای گرفتن الگوهای اساسی در داده ها افزایش می دهیم.

کاهش ابعاد:

مهندسی ویژگی به کاهش ابعاد مجموعه داده کمک می کند.

داده‌های با ابعاد بالا می‌توانند به بیش از حد برازش و افزایش پیچیدگی محاسباتی منجر شوند.

تکنیک هایی مانند انتخاب ویژگی و استخراج به حفظ اطلاعات ضروری و کاهش تعداد ویژگی ها کمک می کند.

رسیدگی به داده های از دست رفته:

مهندسی ویژگی به ما این امکان را می دهد که مقادیر از دست رفته را به طور موثر مدیریت کنیم.

وارد کردن داده های از دست رفته یا ایجاد ویژگی های جدید بر اساس ویژگی های موجود می تواند استحکام مدل را بهبود بخشد.

گنجاندن دانش دامنه:

مهندسان و کارشناسان حوزه می‌توانند ویژگی‌هایی ایجاد کنند که با درک آنها از مشکل همخوانی داشته باشد.

تلفیق دانش خاص دامنه اغلب منجر به نمایش بهتر ویژگی ها می شود.

مقیاس بندی و عادی سازی ویژگی:

ویژگی های مقیاس بندی تضمین می کند که محدوده های مشابهی دارند.

الگوریتم هایی مانند نزول گرادیان زمانی که ویژگی ها به طور مناسب مقیاس بندی شوند سریعتر همگرا می شوند.

رمزگذاری متغیرهای دسته بندی:

تبدیل ویژگی‌های طبقه‌بندی به نمایش‌های عددی (به عنوان مثال، رمزگذاری یک‌طرفه) مدل‌ها را قادر می‌سازد تا آنها را به طور مؤثر پردازش کنند.

تعامل ویژگی ها و ویژگی های چند جمله ای:

ایجاد اصطلاحات تعاملی یا ویژگی های چند جمله ای، روابط پیچیده بین متغیرها را به تصویر می کشد.

به عنوان مثال، ترکیب قد و وزن برای ایجاد یک ویژگی BMI.

انتخاب ویژگی:

شناسایی مرتبط ترین ویژگی ها نویز را کاهش می دهد و قابلیت تفسیر مدل را بهبود می بخشد.

تکنیک ها شامل حذف ویژگی های بازگشتی، منظم سازی L1 و اطلاعات متقابل است.

حال بیایید اطلاعات را از جدول استخراج کنیم:

ویژگی های مجموعه داده (بخش "2-a")

از جدول، من بر روی یک ویژگی مجموعه داده مربوط به بخش 2-a تمرکز خواهم کرد:

نام مجموعه داده: مجموعه داده بیماری قلبی UCI

توضیحات: این مجموعه داده حاوی ویژگی های مربوط به تشخیص بیماری قلبی است، از جمله ویژگی هایی مانند سن، سطح کلسترول، فشار خون و اندازه گیری ECG.

تعداد موارد: تقریبا 303 نمونه.

تعداد ویژگی ها: ویژگی های متعدد، شامل متغیرهای عددی و دسته بندی.

متغیر هدف: طبقه بندی باینری (وجود یا عدم وجود بیماری قلبی).

تشکیل مجموعه داده جدید

برای یک مجموعه داده جدید، بیایید یک مجموعه داده فرضی مرتبط با پیش‌بینی ریزش مشتری ایجاد کنیم:

نام مجموعه داده: مجموعه داده مشتریان مخابراتی

توضیحات: این مجموعه داده رفتار مشتری را در یک شرکت مخابراتی، شامل ویژگی‌هایی مانند مدت تماس، نوع قرارداد، استفاده از داده و شکایات مشتری نشان می‌دهد.

تعداد موارد: 1000 سابقه مشتری.

تعداد امکانات: 10 ویژگی.

متغیر هدف: طبقه‌بندی باینری (خرد شده یا نخورده).

قسمت 5-2 :

اهمیت عادی سازی داده ها:

عملکرد مدل پیشرفته:

داده های نرمال شده دقت و عملکرد مدل را بهبود می بخشد.

این تضمین می کند که ویژگی ها به طور مساوی در پیش بینی ها مشارکت می کنند، به خصوص زمانی که ویژگی ها مقیاس های متفاوتی دارند.

الگوریتم‌های متکی بر معیارهای فاصله (به عنوان مثال، k-نزدیک‌ترین همسایه، ماشین‌های بردار پشتیبان) از داده‌های نرمال‌سازی شده سود می‌برند.

ثبات و همگرایی:

عادی سازی باعث ایجاد ثبات در طول بهینه سازی می شود.

آموزش مبتنی بر گرادیان با داده های نرمال شده سریعتر همگرا می شود.

مسائلی مانند ناپدید شدن یا انفجار شیب ها را کاهش می دهد.

تفسیرپذیری و تجسم:

تفسیر و مقایسه داده های عادی آسان تر است.

روابط بین ویژگی ها واضح تر می شود.

زمانی که ویژگی ها در یک مقیاس باشند، تجسم ها معنادارتر می شوند.

روش نرمال سازی: مقیاس بندی حداقل- حداکثر

شرح روش:

مقیاس بندی Min-Max ویژگی ها را به یک محدوده خاص تبدیل می کند (معمولاً [0، 1]).

به صورت خطی هر مقدار ویژگی را بر اساس حداقل و حداکثر مقادیر مشاهده شده به یک مقدار نرمال شده نگاشت می کند.

فرمول:

برای ویژگی x، مقدار نرمال شده x\_norm به صورت زیر محاسبه می‌شود: [x\_{\text{norm}} = \frac{{x - \text{min}(x)}}{{\text{max}(x) - \text{min}(x)}} ]

اطلاعات ارزیابی:

ما از اطلاعات ارزیابی خاصی در این مثال استفاده نکردیم.

با این حال، در عمل، حداقل و حداکثر مقادیر را از مجموعه داده خود تعیین می کنید.

مثال: پیش بینی قیمت مسکن

بیایید یک مثال ساده را در نظر بگیریم که در آن ما قیمت مسکن را بر اساس ویژگی هایی مانند متراژ مربع، تعداد اتاق خواب و فاصله تا سوپرمارکت پیش بینی می کنیم. عادی سازی این ویژگی ها، رفتار منصفانه را در مقیاس های مختلف تضمین می کند.

داده های اصلی (مثال):

متراژ مربع: محدوده 800 تا 3000 فوت مربع.

اتاق خواب: از 1 تا 5 متغیر است.

فاصله تا سوپرمارکت: بین 0.1 تا 5 مایل است.

مقیاس حداقل حداکثر:

هر ویژگی را در محدوده [0، 1] عادی کنید.

حداقل و حداکثر را برای هر ویژگی محاسبه کنید.

برای بدست آوردن مقادیر نرمال شده، فرمول را اعمال کنید.

فواید:

اکنون همه ویژگی ها در یک مقیاس هستند و مقایسه و آموزش مدل را آسان تر می کند.

این مدل به دلیل مقیاس آن از هیچ ویژگی برخوردار نیست.