به نام خدا

تمرین سری دوم مبانی یادگیری ماشین

معصومه پاسبانی 99243022

سوالات تئوری

سوال اول)

One vs rest: برای هر کلاس i، یک مدل دودویی آموزش داده می‌شود که این کلاس را از تمام کلاس‌های دیگر تفکیک کند. به عبارتی، اگر Cتعداد کلاس‌ها باشد، Cمدل مستقل آموزش داده می‌شود .نمونه ورودی به همه مدل‌ها داده می‌شود و کلاسی که بیشترین احتمال یا خروجی مدل مثبت را تولید کند، انتخاب می‌شود. برای مسائل با تعداد کلاس کم و زمانی که داده‌ها به‌خوبی از هم جدا شده‌اند مناسب است.

مزایا: پیاده سازی ساده ای دارد، برای تعداد زیادی از کلاس ها مقیاس پذیر است.

معایب: زمانی که کلاس‌ها هم‌پوشانی دارند، ممکن است مدل دچار خطا شود؛ چراکه هر مدل مستقل از بقیه آموزش می‌بیند.

One vs one: برای هر جفت کلاس ممکن، یک مدل دودویی آموزش داده می‌شود. برای Cکلاس، تعداد C(C-1)/2 مدل ساخته می‌شود. برای یک نمونه ورودی، همه مدل‌ها پیش‌بینی می‌کنند. کلاس با بیشترین رأی مثبت از مدل‌ها به‌عنوان خروجی انتخاب می‌شود (Voting Scheme). برای مسائل با تعداد کلاس زیاد مناسب است، زیرا هر مدل تنها روی زیرمجموعه کوچکی از داده‌ها تمرکز دارد.

مزایا: در هر مدل، تنها دو کلاس مقایسه می‌شوند، بنابراین داده‌های متوازن‌تر و ساده‌تری برای یادگیری فراهم می‌شود.

معایب: برای تعداد زیاد کلاس ها پیچیدگی محاسباتی بالایی دارد. ممکن است در صورت هم‌پوشانی کلاس‌ها، رأی‌گیری اکثریت نتیجه‌ی اشتباهی ارائه دهد.

Probability calibration: مدل‌های دودویی برای پیش‌بینی احتمال هر کلاس به‌کار می‌روند. با استفاده از توزیع احتمالات، یک کلاس با بیشترین احتمال انتخاب می‌شود. این روش نسبت به OvR بهبود یافته تر است زیرا از ترکیب احتمالات استفاده می کند.

تفاوت های OvO وOvR: در OvR تعداد مدل های مورد نیاز C و در OvO برابر با C(C-1)/2 است. زمان train در OvO نسبت به OvR بیشتر است. OvO جافظه بیشتر نیاز دارد. در OvR کلاس ها مستقل در نظر گرفته میشوند و در OvO روابط جفتی بین کلاس ها مدل میشود.

OvR وقتی که داده‌های کلاس‌ها هم‌پوشانی داشته باشند، مدل ممکن است دچار تناقض شود؛ زیرا هر مدل جداگانه آموزش می‌بیند و احتمال دارد بیش از یک کلاس خروجی مثبت داشته باشد. برای رفع آن باید از احتمال خروجی مدل ها به جای threshold استفاده کرد. همچنین میتوان از مدل های SVM یا شبکه های عصبی استفاده کرد.

OvO هم‌پوشانی بین کلاس‌ها می‌تواند منجر به نتایج متناقض در رأی‌گیری اکثریت شود. برای رفع آن میتوان از روش های وزن دهی در رای گیری استفاده کرد یا داده های آموزشی را افزایش داد.

سوال دوم)

بله میتوان از الگوریتم های طبقه بندی خطی مانند logistic regression یا LDA استفاده کرد زیرا این الگوریتم ها به خوبی برای داده هایی که مرز های میان کلاس ها خطی هستند، کار میکنند. برای این کار ابتدا باید داده ها بررسی و از مقادیر نامعتبر و از دست رفته پاک سازی شوند. سپس با استفاده از نرمال سازی یا استانداردسازی و همچنین تبدیل داده های categorical به عددی با استفاده از روش هایی مانند one hot encoding و... پیش پردازش داده ها انجام شود و در نهایت بسته به نوع داده ها و مسئله مدل مناسب انتخاب شود و در نهایت ارزیابی شود.

راهکار برای بهبود مدل های غیرخطی

می‌توان با استفاده از کرنل‌های غیرخطی مانند کرنل رادیکال (RBF) یا polynomial، فضای ویژگی را به ابعاد بالاتر برد و مرزهای پیچیده‌تری برای تفکیک کلاس‌ها پیدا کرد. اگر داده‌ها غیرخطی هستند، می‌توان ویژگی‌های چندجمله‌ای را به داده‌ها اضافه کرد. این کار باعث می‌شود که مدل بتواند الگوهای پیچیده‌تری را بیاموزد. می‌توان از Regularization برای کنترل پیچیدگی مدل استفاده کرد و داده‌های پرت یا نویزی را بهتر مدیریت کرد. میتوان از مدل های ترکیبی مانند randomforest یا gradient boosting استفاده کرد.

سوال سوم)

تابع خسارت انتخابی می‌تواند بر رفتار همگرایی مدل، مقاومت مدل در برابر داده‌های نویزی و پرت و دقت پیش‌بینی مدل در مواجهه با داده‌های جدید تاثیر بگذارد. تابع خسارت به طور مستقیم بر روی چگونگی تنظیم وزن‌ها (پارامترهای مدل) تأثیر می‌گذارد و باعث می‌شود که مدل با توجه به نوع تابع خسارت، نسبت به تغییرات داده‌ها و نویز واکنش متفاوتی نشان دهد.

مثال هایی از توابع خسارت

Mean Squared Error: این تابع خسارت معمولاً برای رگرسیون استفاده می‌شود، اما در برخی از مدل‌های طبقه‌بندی هم می‌توان از آن استفاده کرد. تابع MSE به صورت زیر تعریف می‌شود:

$$\mathrm{MSE} = \frac{1}{n}\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2.$$

در مدل‌های طبقه‌بندی، MSEمعمولاً برای پیش‌بینی‌های میانگینی به کار می‌رود و ممکن است در شرایطی که داده‌ها نویز دارند یا غیرخطی هستند، باعث همگرایی کند یا ناقص شود .در صورت وجود داده‌های نویزی، MSE به شدت تحت تأثیر این داده‌ها قرار می‌گیرد زیرا خطای مربعی را محاسبه می‌کند و این می‌تواند منجر به همگرایی کند یا باعث overfitting به داده‌های نویزی شود.

Cross entropy loss: این تابع خسارت بیشتر در طبقه‌بندی دودویی و چندکلاسه استفاده می‌شود. تابع Cross-Entropy برای مقایسه توزیع‌های احتمالاتی استفاده می‌شود و به این صورت تعریف می‌شود:

$$H(p,q) = -\sum_i p_i \log q_i = -y \log \hat{y} - (1-y) \log(1-\hat{y})$$

این تابع برای طبقه‌بندی مدل‌هایی مانند رگرسیون لجستیک و شبکه‌های عصبی بسیار مفید است، زیرا به صورت مستقیم تفاوت بین احتمال پیش‌بینی‌شده و احتمال واقعی را ارزیابی می‌کند .در مواجهه با داده‌های نویزی یا پرت، تابع Cross-Entropy می‌تواند حساس باشد زیرا احتمال‌های خیلی نزدیک به صفر یا یک می‌تواند باعث انحراف زیاد در محاسبات لگاریتمی شود .در مواجهه با داده‌های پرت یا نویز، تابع Cross-Entropy ممکن است بیش از حد به پیش‌بینی‌های اشتباه واکنش نشان دهد و عملکرد مدل را مختل کند.

در صورت وجود داده‌های نویزی یا پرت، انتخاب تابع خسارت به شدت بر عملکرد مدل تأثیر می‌گذارد. توابعی مانند MSE و Cross-Entropy که حساس به تغییرات کوچک هستند، می‌توانند موجب overfittingمدل به داده‌های نویزی شوند. توابع خسارت حساس به نویز می‌توانند باعث همگرایی کند یا به رعایت جزئیات نویزی بپردازند که باعث کاهش دقت مدل می‌شود.

سوال چهارم)

(polynomial feature maps)نگاشت‌های ویژگی چندجمله‌ای شامل اضافه کردن ویژگی‌های جدید به داده‌ها هستند که نشان‌دهنده ترکیبات غیرخطی از ویژگی‌های اصلی هستند. این کار باعث می‌شود که مدل به‌طور غیرمستقیم به یک فضای ویژگی جدید وارد شود که در آن روابط غیرخطی قابل مدل‌سازی است.

با استفاده از ویژگی‌های چندجمله‌ای، فضای ویژگی‌ها تغییر می‌کند و مدل باید در فضایی با ابعاد بیشتر آموزش ببیند. این تغییر در فضای ویژگی‌ها منجر به پیچیدگی بیشتر در مدل می‌شود.مدل‌های خطی که بر روی داده‌های اولیه آموزش داده می‌شوند، ممکن است قادر به یادگیری روابط پیچیده نباشند اما با استفاده از نگاشت‌های چندجمله‌ای، مدل می‌تواند به روابط پیچیده‌تر و غیرخطی دست یابد.

هر چه درجه چندجمله‌ای افزایش یابد، تعداد ویژگی‌های جدید به‌طور نمایی افزایش می‌یابد (یعنی فضای ویژگی‌ها به شدت افزایش می‌یابد). این باعث می‌شود که مدل نیاز به منابع محاسباتی بیشتری برای آموزش داشته باشد و احتمال overfitting(و همچنین زمان آموزش بیشتر) افزایش می‌یابد. این افزایش ابعاد ممکن است باعث افزایش پیچیدگی محاسباتی شود و مدیریت مدل را سخت‌تر کند.

در رگرسیون خطی دو روش عمده برای حل مسئله وجود دارد: gradient descent و روش مستقیم.

در گرادیان نزولی، مدل به‌طور تدریجی با به‌روزرسانی وزن‌ها در راستای کم کردن تابع خسارت، بهینه می‌شود. با اضافه کردن ویژگی‌های چندجمله‌ای، تعداد ویژگی‌ها و ابعاد فضای ویژگی افزایش می‌یابد.

مزایا: اگر داده‌ها به فضای ابعاد بالا منتقل شوند (مثلاً با ویژگی‌های درجه بالا)، مدل ممکن است سریع‌تر همگرا شود زیرا فضای جدید ممکن است مناسب‌تر برای یافتن روابط بین ویژگی‌ها و هدف باشد.اگر از مقیاس‌بندی مناسب برای ویژگی‌ها استفاده کنید، ممکن است مدل بهتر عمل کند زیرا هر ویژگی می‌تواند تأثیر مشابهی در فرآیند بهینه‌سازی داشته باشد.

معایب:زمان همگرایی ممکن است بیشتر شود زیرا فضای ابعاد بالا باعث افزایش تعداد گام‌ها در هر به‌روزرسانی می‌شود.اگر مدل دچار overfittingشود، احتمال دارد که گرادیان نزولی با سرعت کمتری همگرا شود، زیرا مدل بیش از حد به ویژگی‌های خاص و نویزهای موجود در داده‌ها حساس خواهد شد.نیاز به تنظیمات دقیق‌تر هایپرپارامترها مثل نرخ یادگیری، زیرا ویژگی‌های جدید ممکن است باعث تغییرات بزرگتر در گرادیان‌ها شوند.

روش مستقیم معمولاً برای رگرسیون خطی در ابعاد کم مناسب است و به‌طور مستقیم به وزن‌ها می‌رسد.

مزایا: در صورت استفاده از نگاشت‌های چندجمله‌ای، حل مستقیم ممکن است راه‌حلی دقیق و سریع باشد به شرطی که فضای ویژگی‌ها خیلی بزرگ نشود.از آنجا که حل مستقیم به‌صورت ریاضی بهینه است، اگر داده‌ها با دقت مناسب و ویژگی‌های خوب پردازش شوند، ممکن است سریع‌تر و با دقت بیشتری به جواب برسد.

معایب: در صورت افزایش ابعاد ویژگی‌ها به‌طور نمایی، محاسبه معادله $(X^TX)^{-1}$زمان‌بر و پرهزینه خواهد شد.برای داده‌های با ابعاد بسیار زیاد، ممکن است حتی محاسبات عددی نیز با مشکلاتی مانند ناراستی عددی روبه‌رو شوند.

Generalization در رگرسیون خطی با نگاشت‌های چندجمله‌ای:

با افزودن ویژگی‌های چندجمله‌ای، مدل می‌تواند از الگوهای غیرخطی در داده‌ها استفاده کند. این باعث می‌شود که مدل پیچیده‌تر و توانمندتر شود، اما در عین حال، توانایی عمومی‌سازی کاهش یابد.اگر درجه چندجمله‌ای بیش از حد زیاد باشد، مدل ممکن است به overfittingدچار شود، زیرا مدل بیش از حد به ویژگی‌های خاص داده‌های آموزشی وابسته می‌شود و در مواجهه با داده‌های جدید عملکرد ضعیفی خواهد داشت.

افزایش دقت در داده‌های غیرخطی:

هنگامی که داده‌ها غیرخطی هستند، نگاشت‌های چندجمله‌ای می‌توانند به مدل کمک کنند تا این روابط پیچیده را یاد بگیرد. به‌عنوان مثال، داده‌هایی که در ابتدا به‌صورت دو دسته جدا از هم در یک فضای دو بعدی خطی نمی‌توانند جدا شوند، می‌توانند با افزودن ویژگی‌های درجه دوم یا بیشتر قابل تفکیک شوند.اما اگر این نگاشت‌ها به درستی انتخاب نشوند (یعنی ویژگی‌ها بسیار پیچیده باشند)، مدل قادر به تعمیم به‌خوبی نخواهد بود و به داده‌های نویزی حساس می‌شود.

سوالات عملی

سوال اول)

نسبت اولیه‌ی تقسیم، 25٪ برای تست و 75٪ برای آموزش انتخاب شده است. این انتخاب بعداً در کد با تست نسبت‌های دیگر (مانند 35٪، 50٪، 60٪ و ...) ارزیابی شده است.نسبت 75:25 تعادل مناسبی بین اندازه داده‌های آموزش (برای یادگیری مدل) و تست (برای ارزیابی عملکرد مدل) ایجاد می‌کند.

این نسبت به اندازه کافی داده در اختیار مدل قرار می‌دهد تا بتواند وزن‌های مناسب یاد بگیرد و در عین حال داده کافی برای بررسی تعمیم‌دهی مدل باقی می‌گذارد.با تغییر این نسبت در کد و بررسی تابع هزینه، مشخص شد که نسبت‌های کمتر (مانند 25٪ داده برای آموزش) باعث کاهش دقت مدل می‌شوند.برای هر ستون تعداد مقادیر گم‌شده و مقادیر یکتا چاپ شده است داده‌ها بر اساس سن به چهار دسته (`teen, young adults, adults, middle age adults`) تقسیم‌بندی شده‌اند.

با استفاده از label encoding و one hot encoding و سپس استاندارسازی داده ها پیش پردازش شدند.

MAEمیانگین قدر مطلق خطا :میانگین قدر مطلق اختلاف بین مقادیر واقعی و پیش‌بینی‌شده را محاسبه می‌کند. این معیار حساسیت کمتری به مقادیر پرت دارد.

MSEمیانگین مربعات خطا :میانگین مربعات اختلاف بین مقادیر واقعی و پیش‌بینی‌شده است. این معیار مقادیر پرت را به شدت جریمه می‌کند.

RMSEریشه میانگین مربعات خطا :ریشه‌ی دوم MSE است و واحد آن مشابه متغیر هدف است.

MAEمناسب برای سناریوهایی که حساسیت به مقادیر پرت کمتر موردنظر است.
MSEو :RMSE به دلیل جریمه‌ی مقادیر پرت، برای سناریوهایی که مقادیر پرت تأثیر زیادی دارند، مناسب‌ترند.
انتخاب :برای این مسئله، RMSEمعیار مناسب‌تری است زیرا واحد آن مشابه متغیر هدف است و اختلافات بزرگ در هزینه‌ها تأثیر بیشتری در ارزیابی دارد.

تابع هزینه Mean Squared Error (MSE) پیاده‌سازی شده است. گرادیان‌های وزن‌ها و بایاس محاسبه شده و در هر تکرار به‌روزرسانی می‌شوند و تغییرات تابع هزینه در طول تکرارها (10,000 بار) ثبت و رسم شده‌اند.

داده‌ها با نسبت‌های مختلف (مانند 0.25، 0.35، ...، 0.85) به مجموعه آموزش و تست تقسیم شده‌اند و هزینه برای هر مجموعه محاسبه و در نمودار مقایسه‌ای رسم شده است.
هزینه تابع MSE به مرور کاهش یافته است که نشان‌دهنده همگرایی مدل است. تغییرات وزن‌ها به تدریج به سمت مقدار ثابت حرکت کرده‌اند که نشان‌دهنده پایداری مدل است به این معنا که مدل به نقطه‌ای رسیده که کمینه‌ی تابع هزینه به دست آمده است.

سوال دوم)

ابتدا داده ها را به دو بخش تست و آموزش با نسبت 35٪ تست و 65٪ آموزش تقسیم کرده. سپس با استفاده از تابع isnull مقادیر از دست رفته در هر ستون را چک کرده تا مطمئن شویم داده های از دست نرفته باشد.

از روش IQR برای شناسایی و مدیریت داده‌های پرت استفاده شد. به طور خاص، مقادیر کمتر ازQ1– IQR *1.5 و بیشتر از IQR*1.5 +Q3 به مقادیر پایین و بالای بازه محدود شدند و تعداد داده های پرت برای هر ستون ثبت شدند. و در آخر داده ها را نرمال سازی کردیم.

کلاس logisticregression برای مدل شامل توابع زیر است:

`__init__`: تنظیم تعداد کلاس‌ها، نرخ یادگیری، تعداد epochها و آستانه.

`softmax`:برای محاسبه احتمال کلاس‌ها.

`CrossEntropyLoss`:محاسبه خطای آنتروپی متقاطع بین برچسب های واقعی و پیش بینی ها.

`train`:آموزش مدل با محاسبه گرادیان و به‌روزرسانی وزن‌ها.

`predict`:پیش‌بینی کلاس داده‌های آزمایشی.

`evaluate`:ارزیابی مدل با محاسبه دقت، F1-score و confusion matrix .

`plot_accuracy_vs_epoch`:نمایش دقت مدل در طی epochها.

تابع softmax به این صورت عمل میکند که بیشینه مقدار هر نمونه از scoresکم می‌شود (برای جلوگیری از ناپایداری عددی) و مقادیر نمایی شده و نرمال‌سازی می‌شوند تا مجموع هر سطر برابر با 1 باشد.

در تابع train وزن ها به صورت اولیه صفر مفداردهی میشوند و برای هر epoch با استفاده از ضرب داخلی داده ها و وزن ها score محاسبه میشود وسپس تابع softmax روی آن ها اعمال میشود. با استفاده از تابع crossEntropyLoss خطای پیش بینی محاسبه میشود. متق نسبت به w ها محاسبه شده و سپس با استفاده از gradient descent وزن ها آپدیت میشوند.

تابع predict به این صورت عمل میکند که با استفاده از وزن های فعلی score حساب شده و احتمال ها را با استفاده از تابع softmax به دست می آورد و در نهایت کلاس با بیشترین احتمال انتخاب میشود.

تابع evaluate با استفاده از predict لیبل ها را برای داده های آموزشی پیش بینی میکند و با استفاده از accuracy-score دقت را محاسبه کرده . همچنین F1-score را حساب کرده و در نهایت confusion matrix را تولید میکند.

در تابع cross entropy loss مقادیر log(y hat) حساب شده و سپس مجموع این مقادیر بر تعداد نمونه ها تقسیم شده و منفی میشود.

مدل logisticRession2 عملکرد مدل پایه قبلی را با افزون ویژگی regularization تا به بهبود عملکرد مدل کمک کند.