سوال 1

گزارش بخش الف: پیاده‌سازی رگرسیون خطی چند متغیره

در این بخش، ما از داده‌های بوستون استفاده کردیم تا یک مدل رگرسیون خطی چند متغیره را پیاده‌سازی کنیم. داده‌ها به دو بخش آموزشی (80٪) و تست (20٪) تقسیم شدند و داده‌های مربوط به هر دو بخش نرمال‌سازی شدند. مدل با استفاده از روش کاهش گرادیان به صورت دسته‌ای کوچک (Mini-batch Gradient Descent)آموزش داده شد و تابع خطای میانگین مربعات (MSE) به عنوان تابع هزینه در نظر گرفته شد. نتایج زیر به دست آمد:

-خطای میانگین مربعات بر روی داده‌های آموزشی: 21.64

-خطای میانگین مربعات بر روی داده‌های تست: 24.31

گزارش بخش ب: اعمال پایدارسازی با نرمL2

در این بخش، با افزودن پایدارسازی L2 به تابع هزینه، سعی در بهبود عملکرد مدل کردیم. این کار با افزودن یک جمله تنظیم‌کننده به تابع هزینه انجام شد تا از پیچیدگی بیش از حد مدل جلوگیری شود. نتایج به دست آمده به شرح زیر است:

-خطای میانگین مربعات بر روی داده‌های آموزشی) باL2): 25.68

-خطای میانگین مربعات بر روی داده‌های تست) باL2): 28.28

توضیحات بخش ج: مقایسه و بررسی نتایج

با مقایسه نتایج بخش‌های الف (بدون پایدارسازی) و ب) با پایدارسازی(L2 ، مشاهده می‌کنیم که افزودن پایدارسازی L2 به تابع هزینه، منجر به افزایش خطای میانگین مربعات در هر دو بخش آموزشی و تست شده است. این نشان می‌دهد که افزودن پایدارسازی ممکن است به کاهش بیش از حد پیچیدگی مدل منجر شود که این خود می‌تواند بر عملکرد کلی مدل تأثیر منفی داشته باشد، به ویژه اگر مدل از قبل دچار بیش‌برازش نشده باشد.

پایدارسازی بیشتر زمانی مفید است که مدل دچار بیش‌برازش شده باشد، یعنی تفاوت قابل توجهی بین عملکرد بر روی داده‌های آموزشی و تست وجود داشته باشد. در این مورد خاص، مدل اولیه بدون پایدارسازی نشان دهنده بیش‌برازش شدید نبود، چرا که خطای میانگین مربعات برای داده‌های آموزشی و تست نزدیک به هم بود. بنابراین، تأثیر پایدارسازی در این مورد خاص، افزایش اندک خطا در هر دو مجموعه داده بود، که نشان‌دهنده این است که مدل قبلاً توانایی خوبی در تعمیم دادن داشته است.

این تمرین نشان می‌دهد اهمیت پایدارسازی به عنوان ابزاری برای بهبود تعمیم‌پذیری مدل، به‌خصوص در سناریوهایی که خطر بیش‌برازش وجود دارد. با این حال، همچنین بر لزوم تنظیم دقیق پارامترهای پایدارسازی تأکید می‌کند تا از زیربرازش جلوگیری شود و بهترین عملکرد ممکن بر روی داده‌های دیده نشده تضمین شود.

سوال 2

در این تمرین، ما به پیاده‌سازی و تحلیل الگوریتم رگرسیون لجستیک با استفاده از روش گرادیان نزولی (Gradient Descent)و اعمال پایدارسازی (Regularization) با نرم L2 بر روی داده‌های برچسب‌دار سرطان سینه از کتابخانه sklearn پرداختیم. فرایند به شرح زیر است:

الف) بارگذاری داده‌ها و آماده‌سازی:

ابتدا، دیتاست سرطان سینه را با استفاده از کتابخانه sklearn بارگذاری کردیم. سپس، داده‌ها را به دو بخش آموزش (80٪) و تست (20٪) تقسیم کردیم و آن‌ها را نرمال‌سازی نمودیم تا مقیاس مشابهی داشته باشند، که برای بهبود عملکرد مدل گرادیان نزولی مفید است.

ب) پیاده‌سازی رگرسیون لجستیک با گرادیان نزولی:

سپس، رگرسیون لجستیک را با استفاده از روش گرادیان نزولی پیاده‌سازی کردیم. بدین منظور، تابع سیگموید به عنوان تابع فعال‌ساز، تابع هزینه مربوط به رگرسیون لجستیک و الگوریتم گرادیان نزولی برای بهینه‌سازی پارامترهای مدل را تعریف نمودیم. این مرحله بدون اعمال پایدارسازی اجرا شد.

ج) اعمال پایدارسازی با نرم:L2

در ادامه، با اضافه کردن اصطلاح پایدارسازی (Regularization) مرتبط با نرم L2 به تابع هزینه، مدل را با در نظر گرفتن جریمه‌های وزن‌های بزرگ بهینه‌سازی کردیم. هدف از این اصلاح، جلوگیری از بیش‌برازش (Overfitting)و افزایش توانایی مدل برای تعمیم‌پذیری به داده‌های تست است.

نتایج و مقایسه:

بدون پایدارسازی: مقدار تابع هزینه بر روی داده‌های آموزش و تست به ترتیب حدود 0.107 و 0.090 بود. این نشان‌دهنده عملکرد خوب مدل در یادگیری الگوهای داده آموزشی و تعمیم آن به داده تست است.

با پایدارسازی: با اعمال پایدارسازی، مقدار تابع هزینه اندکی افزایش یافت و به ترتیب حدود 0.110 و 0.102 برای داده‌های آموزش و تست به دست آمد. این افزایش ناچیز در تابع تابع هزینه نشان‌دهنده تأثیر پایدارسازی است که با افزودن جریمه به وزن‌های بزرگ، مدل را به سمت سادگی بیشتر سوق می‌دهد و از بیش‌برازش جلوگیری می‌کند.

تحلیل نهایی:

تأثیر پایدارسازی: اعمال پایدارسازی L2 موجب می‌شود تا مدل در برابر بیش‌برازش مقاوم‌تر شود، اگرچه ممکن است کمی بر دقت آن در داده‌های آموزشی تأثیر منفی بگذارد. با این حال، این تأثیر منفی بسیار جزئی است و مزیت اصلی آن، افزایش توانایی تعمیم به داده‌های جدید و ناشناخته است.

انتخاب پارامتر :(lambda) انتخاب مقدار مناسب برای پارامتر پایدارسازی (lambda) حیاتی است. مقدار بسیار کم (lambda) تأثیر کمی در جلوگیری از بیش‌برازش دارد، در حالی که مقدار بسیار زیاد آن می‌تواند موجب کاهش شدید توانایی مدل برای یادگیری از داده‌ها شود.

اهمیت نرمال‌سازی: نرمال‌سازی داده‌ها قبل از آموزش مدل به بهبود توانایی الگوریتم گرادیان نزولی کمک کرده و از همگرایی سریع‌تر آن اطمینان حاصل می‌کند.

نتیجه‌گیری:

رگرسیون لجستیک با استفاده از گرادیان نزولی و پایدارسازی L2 یک روش مؤثر برای مقابله با مسائل طبقه‌بندی است، به‌خصوص زمانی که هدف، جلوگیری از بیش‌برازش و بهبود توانایی تعمیم مدل باشد. این تمرین نشان داد که چگونه می‌توان با تنظیم دقیق پارامترها و استفاده صحیح از تکنیک‌های پایدارسازی، عملکرد مدل‌های یادگیری ماشین را بهینه‌سازی کرد.

سوال 3

با استفاده از کتابخانه sklearn و پیاده‌سازی SGDRegressor برای رگرسیون خطی چند متغییره بدون پایدارسازی (L2)، نتایج زیر بدست آمد:

- خطای میانگین مربعات برای داده‌های آموزشی: 21.71
- خطای میانگین مربعات برای داده‌های تست: 24.82

پارامترهای مدل شامل وزن‌ها و انحراف می‌شود. وزن‌های مدل و انحراف تعیین‌شده توسط SGDRegressor را با نتایج پیاده‌سازی دستی ما در بخش قبل مقایسه می‌کنیم:

- پیاده‌سازی دستی:
  - خطای میانگین مربعات آموزشی: 21.64
  - خطای میانگین مربعات تست: 24.31
- پیاده‌سازی با SGDRegressor:
  - خطای میانگین مربعات آموزشی: 21.71
  - خطای میانگین مربعات تست: 24.82

نتایج حاصل از استفاده از SGDRegressor نشان‌دهنده عملکرد مشابهی با پیاده‌سازی دستی ما است. تفاوت‌های جزئی در خطاها می‌تواند ناشی از جزئیات پیاده‌سازی، مثل تعداد تکرارها، نرخ یادگیری و روش تنظیم نرخ یادگیری در SGDRegressor باشد. همچنین، وزن‌ها و انحرافات بدست آمده از هر دو روش به طور خاص مقایسه نشدند اما باید بسیار نزدیک به هم باشند، زیرا هر دو مدل بر اساس همان داده‌های آموزشی آموزش دیده‌اند و هدف مشابهی را دنبال می‌کنند.

سوال 4

با استفاده از کتابخانه sklearn برای پیاده‌سازی رگرسیون لجستیک با پایدارسازی L2 که به صورت پیش‌فرض اعمال می‌شود نتایج زیر حاصل شد:

-مقدار تابع هزینه بر روی داده‌های آموزشی: حدود 0.054

-مقدار تابع هزینه بر روی داده‌های تست: حدود 0.060

-پنج مقدار اول از بردار پارامترها([0.464- ,0.390- ,0.394- ,0.428- ,0.444]) :

مقایسه با نتایج پیاده‌سازی دستی:

تابع هزینه: استفاده از sklearn منجر به کاهش قابل توجه‌ای در مقدار تابع هزینه شده است، هم برای داده‌های آموزشی و هم برای داده‌های تست. این امر ممکن است ناشی از بهینه‌سازی‌های اضافی و تکنیک‌های موثرتری باشد که sklearn در پیاده‌سازی الگوریتم‌های خود استفاده می‌کند.

-**پارامترها** (theta) مقادیر پارامترهای بدست آمده از sklearn و پیاده‌سازی دستی تفاوت دارند. این تفاوت‌ها می‌توانند از تفاوت در الگوریتم بهینه‌سازی، مقدار دهی اولیه پارامترها، و یا نحوه اعمال پایدارسازی نشئت گرفته باشند.

نتیجه‌گیری:

استفاده از کتابخانه‌های آماده مانند sklearn می‌تواند به طور قابل توجهی پیچیدگی پیاده‌سازی را کاهش دهد و در بسیاری موارد عملکرد بهتری را ارائه دهد، به ویژه زمانی که با پیچیدگی‌ها و جزئیات مربوط به بهینه‌سازی و پایدارسازی مواجه هستیم.

سوال 5

در این گزارش، ما به بررسی و پیاده‌سازی معادله نرمال برای یافتن پارامترهای یک مدل خطی با استفاده از Pythonو کتابخانه Numpy می‌پردازیم. هدف ما این است که نشان دهیم چگونه می‌توان با استفاده از داده‌های ساختگی، پارامترهای بهینه برای یک خط را با استفاده از معادله نرمال محاسبه کرد.

مقدمه

معادله نرمال یک روش ریاضی است که برای یافتن پارامترهایی که باعث بهینه‌سازی مدل‌های خطی می‌شود، استفاده می‌شود. فرمول معادله نرمال به صورت زیر است:

[theta = (X^T X)^{-1} X^T y]

که در آن:

:(X) -ماتریس ویژگی‌ها با افزودن یک ستون از اعداد 1 برای جمله ثابت است.

:(y) -بردار پاسخ‌ها است.

:(theta) -بردار پارامترهای مدل که شامل جمله ثابت و شیب خط است.

برای این مثال، ما ابتدا یک سری داده ساختگی تولید کردیم که شامل 100 نقطه داده بود. این داده‌ها با استفاده از یک تابع خطی با اضافه کردن نویز تصادفی تولید شدند:

[y = 4 + 3X + text{نویز تصادفی}]

پیاده‌سازی معادله نرمال

پس از تولید داده‌ها، ما ستونی از اعداد 1 را به ماتریس ویژگی‌ها اضافه کردیم تا بتوانیم جمله ثابت را در مدل خطی مان در نظر بگیریم. سپس با استفاده از فرمول معادله نرمال، پارامترهای (theta_0) (جمله ثابت) و (theta_1)(شیب خط) را محاسبه کردیم. نتایج به دست آمده به شرح زیر بود:

- (theta_0 = 4.22)
- (theta_1 = 2.97)

این نتایج نشان می‌دهد که مدل خطی پیش‌بینی شده برای داده‌های ما به صورت (y = 4.22 + 2.97x) است.

نتیجه‌گیری

پیاده‌سازی معادله نرمال با استفاده از Python و Numpy نشان داد که چگونه می‌توان به راحتی پارامترهای یک مدل خطی را با استفاده از داده‌های موجود محاسبه کرد. این روش ریاضی به خصوص برای مسائلی که داده‌های زیادی دارند و نیاز به یک روش مستقیم و کارآمد برای تخمین پارامترها دارند، بسیار مفید است. معادله نرمال به ما این امکان را می‌دهد که بدون نیاز به تکرار و یا اعمال روش‌های بهینه‌سازی پیچیده، به سرعت به بهترین برازش خطی برای داده‌های موجود دست یابیم.

در نظر داشته باشید که در حالی که معادله نرمال برای مجموعه داده‌های کوچک تا متوسط بسیار کارآمد است، در مواردی که با حجم بسیار زیادی از داده‌ها یا ویژگی‌ها روبرو هستیم، ممکن است به دلیل نیاز به محاسبه

معکوس ماتریس، کارایی لازم را نداشته باشد. در چنین شرایطی، روش‌های دیگری مانند گرادیان نزولی ممکن است گزینه‌های بهتری باشند.