دانشگاه صنعتی امیرکبیر

(پلی‌تکنیک تهران)

دانشکده ریاضی و علوم کامپیوتر

پایان‌نامه کارشناسی‌ارشد

ریاضی و کاربرد ها

پروژه ۲ درس نرم افزار ریاضی

نگارش

محمدجواد مسگرها

استاد راهنما

دکتر حسینقلی زاده

۱۴۰۵/۳/۱۷

بسم الله الرحمن الرحیم

فهرست مطالب

عنوان | صفحه

فهرست تصاویر . . . . . ج

فهرست جداول . . . . . د

۱ مقدمه و مفاهیم پایه ۱
۱-۱ مقدمه ۱
۱-۲ مفهوم دسته‌بندی و رگرسیون لجستیک ۱
۱-۳ ساختار گزارش ۲

۲ مدل‌سازی ریاضی رگرسیون لجستیک ۴
۲-۱ تابع فرضیه و نگاشت غیرخطی ۴
۲-۲ قاعده تصمیم‌گیری و خروجی مدل ۵
۲-۳ مبانی آماری رگرسیون لجستیک ۵
۲-۴ تابع هزینه و استخراج ریاضی گرادیان ۵

۳ الگوریتم بهینه‌سازی گرادیان نزولی ۷
۳-۱ معرفی الگوریتم و شبه‌کد ۷
۳-۲ اجرای مرحله‌به‌مرحله الگوریتم (Algorithm Trace) ۸
۳-۳ فلوچارت اجرای الگوریتم ۱۰

۴ شبیه‌سازی و ارزیابی نتایج ۱۱
۴-۱ تحلیل بصری همگرایی و مرز تصمیم ۱۱
۴-۲ تحلیل کمّی و ارزیابی متریک‌ها ۱۲

منابع و مراجع ۱۴

فهرست تصاویر

شکل صفحه

۱-۱ گراف تعاملی ساختار پروژه (جهت انتقال به هر فصل، بر روی گره مربوطه کلیک نمایید).۳

۱-۳ فلوچارت تعاملی اجرای الگوریتم گرادیان نزولی ۱۰

۱-۴ ارزیابی مدل رگرسیون لجستیک: (الف) روند همگرایی الگوریتم و (ب) مرز تصمیم‌گیری در فضای ویژگی دوبعدی. ۱۲

فهرست جداول

جدول صفحه

۴-۱ مقایسه معیارهای ارزیابی مدل رگرسیون لجستیک با نرخ‌های یادگیری (α) مختلف . ۱۳

فصل ۱

مقدمه و مفاهیم پایه

۱-۱ مقدمه

پیشرفت‌های چشمگیر در حوزه هوش مصنوعی و یادگیری ماشین[۱] در دهه‌های اخیر، تحولات شگرفی را در تحلیل داده‌ها ایجاد کرده است. یکی از مسائل بنیادین در این حوزه، مسئله دسته‌بندی[۲] است. هدف در دسته‌بندی، تخصیص یک برچسب مشخص به یک نمونه داده ورودی بر اساس ویژگی‌های استخراج‌شده از آن است. الگوریتم‌های متعددی برای حل این مسئله توسعه یافته‌اند که هر یک دارای پیچیدگی‌ها و فروض ریاضی خاص خود هستند [۱].

در میان این الگوریتم‌ها، مدل‌های خطی به دلیل سادگی در پیاده‌سازی، تفسیرپذیری بالا و پایه ریاضی مستحکم، همواره مورد توجه پژوهشگران بوده‌اند. در این گزارش، به بررسی تحلیلی یکی از پرکاربردترین مدل‌های خطی برای مسئله دسته‌بندی، یعنی رگرسیون لجستیک[۳]، خواهیم پرداخت.

۱-۲ مفهوم دسته‌بندی و رگرسیون لجستیک

اگرچه نام «رگرسیون» در عنوان این الگوریتم وجود دارد، اما رگرسیون لجستیک در واقع یک الگوریتم دسته‌بندی است. دلیل این نام‌گذاری، ریشه تاریخی و ساختار ریاضی آن است که بر پایه رگرسیون خطی[۴] بنا شده است.

---

[۱] Machine Learning
[۲] Classification
[۳] Logistic Regression
[۴] Linear Regression

در رگرسیون خطی استاندارد، متغیر هدف یک مقدار پیوسته در بازه $(-\infty, +\infty)$ است. با این حال، در یک مسئله دسته‌بندی دودویی[۵]، متغیر هدف تنها می‌تواند دو مقدار گسسته به خود بگیرد (به‌عنوان مثال ۰ و ۱). استفاده مستقیم از رگرسیون خطی برای چنین مسئله‌ای چالش‌های تحلیلی متعددی ایجاد می‌کند؛ زیرا خروجی مدل می‌تواند مقادیری خارج از بازه $[۰, ۱]$ تولید کند که فاقد تفسیر احتمالاتی است [۲].

رگرسیون لجستیک این محدودیت را با عبور دادن خروجی خطی از یک تابع غیرخطی به نام تابع فعال‌ساز[۶] حل می‌کند. این تابع، که در فصل آینده با جزئیات ریاضی بررسی خواهد شد، هر مقدار حقیقی را به بازه $(۰, ۱)$ نگاشت می‌کند. در نتیجه، خروجی مدل می‌تواند به‌عنوان احتمال تعلق داده ورودی به کلاس مثبت (۱) تفسیر شود. برای یافتن بهترین پارامترهای این مدل، نیاز به یک الگوریتم بهینه‌سازی کارآمد داریم که الگوریتم گرادیان نزولی[۷] یکی از قدرتمندترین ابزارها در این زمینه است و در فصل سوم به‌طور مفصل باز می‌شود.

۱-۳ ساختار گزارش

به منظور ارائه یک تحلیل جامع، این گزارش در چهار فصل ساختاریافته تدوین شده است. گراف تعاملی نشان‌داده‌شده در شکل ۱-۱، ارتباط منطقی و جریان اطلاعات را میان فصول مختلف این پژوهش به تصویر می‌کشد. با کلیک بر روی هر یک از گره‌های این گراف، می‌توانید به‌طور مستقیم به فصل مربوطه منتقل شوید.

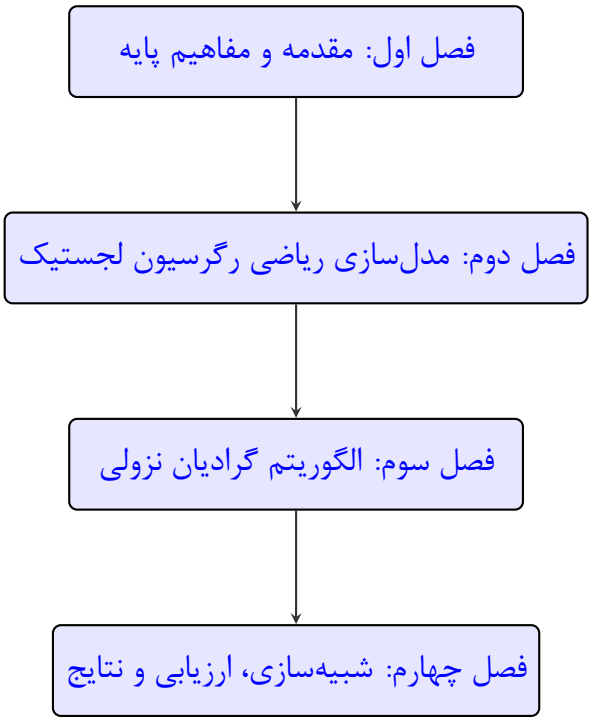
همان‌طور که در شکل ۱-۱ ملاحظه می‌گردد، پس از این مقدمه، در فصل دوم به سراغ فرمول‌بندی و توصیف ریاضی مدل خواهیم رفت.

---

[۵] Binary Classification

[۶] Activation Function

[۷] Gradient Descent



شکل ۱-۱: گراف تعاملی ساختار پروژه (جهت انتقال به هر فصل، بر روی گره مربوطه کلیک نمایید).

فصل ۲

مدل‌سازی ریاضی رگرسیون لجستیک

۲-۱ تابع فرضیه و نگاشت غیرخطی

در یادگیری ماشین، مدل‌سازی یک پدیده نیازمند تعریف یک تابع فرضیه[۱] است که رابطه میان ورودی‌ها و خروجی را تقریب بزند. در مدل‌های خطی ساده، این تابع به‌صورت ترکیب خطی ویژگی‌ها تعریف می‌شود:

$$z = \theta^T x = \theta_۰ + \theta_۱ x_۱ + \cdots + \theta_n x_n, \tag{۲-۱}$$

که در آن θ بردار وزن‌ها (پارامترهای مدل) و x بردار ویژگی‌های ورودی است. با این حال، خروجی معادله ۲-۱ می‌تواند هر مقدار حقیقی در بازه $(-\infty, +\infty)$ را به خود بگیرد. برای آن‌که بتوانیم این خروجی را به‌عنوان یک «احتمال» در نظر بگیریم، باید آن را به بازه $(۰, ۱)$ نگاشت کنیم. به همین دلیل از تابع غیرخطی سیگموئید[۲] استفاده می‌شود. تابع فرضیه نهایی در رگرسیون لجستیک به‌صورت زیر است:

$$h_\theta(x) = \sigma(z) = \frac{۱}{۱ + e^{-\theta^T x}}. \tag{۲-۲}$$

این خروجی، احتمال تعلق داده x به کلاس مثبت را نشان می‌دهد؛ به عبارت ریاضی $P(y = ۱|x; \theta) = h_\theta(x)$.

---

[۱] Hypothesis Function

[۲] Sigmoid Function

۲-۲ قاعده تصمیم‌گیری و خروجی مدل

پس از محاسبه احتمال توسط تابع فرضیه، نیازمند یک قاعده تصمیم[۳] هستیم تا برچسب نهایی کلاس، که با نماد $\hat{y}$ نشان داده می‌شود، مشخص گردد. در دسته‌بندی دودویی، معمولاً آستانه تصمیم‌گیری روی احتمال ۰٫۵ تنظیم می‌شود. این منطق شرطی را می‌توان در قالب یک فرمول ریاضی چندضابطه‌ای ساختارمند کرد.

برچسب نهایی مدل با توجه به آستانه مذکور، مطابق رابطه ۲-۳ محاسبه می‌گردد:

$$\hat{y} = \begin{cases} ۱, & \text{اگر } h_\theta(x) \geq ۰٫۵ \\ ۰, & \text{اگر } h_\theta(x) < ۰٫۵ \end{cases} \tag{۲-۳}$$

بر اساس رابطه ۲-۳، چنانچه مدل پیش‌بینی کند احتمال تعلق نمونه به کلاس یک، پنجاه درصد یا بیشتر است، نمونه به عنوان کلاس یک طبقه‌بندی می‌شود و در غیر این صورت به کلاس صفر تخصیص می‌یابد.

۲-۳ مبانی آماری رگرسیون لجستیک

رگرسیون لجستیک برخلاف نامش، یک الگوریتم دسته‌بندی (Classification) است که بر مبنای مدل‌سازی احتمالات عمل می‌کند. هدف اصلی در این مدل، یافتن بردار پارامترهای بهینه θ است به گونه‌ای که تابع لگاریتم درست‌نمایی (Log-Likelihood) بیشینه شود، یا به طور معادل، تابع هزینه $J(\theta)$ کمینه گردد. همان‌طور که هیستی و همکاران در کتاب مرجع خود به تفصیل بررسی کرده‌اند [۲]، بهینه‌سازی این تابع به دلیل غیرخطی بودن تابع سیگموئید، پاسخ فرم‌بسته (Closed-form Solution) ندارد؛ از این رو، استفاده از روش‌های تکرارشونده مانند گرادیان نزولی (Gradient Descent) یا روش نیوتن-رافسون برای یافتن بهینه سراسری آن الزامی است.

۲-۴ تابع هزینه و استخراج ریاضی گرادیان

یکی از مفروضات اساسی در بهینه‌سازی، انتخاب تابع هزینه[۴] مناسب است. در رگرسیون لجستیک، اگر از تابع خطای میانگین مربعات[۵] استفاده کنیم، به دلیل حضور تابع غیرخطی سیگموئید، تابع هزینه نهایی غیرمحدب[۶] خواهد شد. چنین تابعی دارای نقاط کمینه محلی متعددی است که الگوریتم‌های بهینه‌سازی را به دام می‌اندازد.

---

[۳] Decision Rule
[۴] Cost Function
[۵] Mean Squared Error (MSE)
[۶] Non-convex

برای حل این چالش تحلیلی، بر مبنای اصل برآورد بیشینه‌نمایی[۷]، تابع هزینه آنتروپی متقاطع[۸] تعریف می‌گردد که تحدب آن از نظر ریاضی اثبات شده است. تابع هزینه کل برای m نمونه آموزشی به‌صورت زیر است:

$$J(\theta) = -\frac{۱}{m}\sum_{i=۱}^{m}\left[y^{(i)}\log(h_\theta(x^{(i)})) + (۱-y^{(i)})\log(۱-h_\theta(x^{(i)}))\right]. \qquad (۴\text{-}۲)$$

جهت به‌روزرسانی پارامترها در الگوریتم‌های بهینه‌سازی، محاسبه مشتق جزئی تابع هزینه ۲-۴ نسبت به هر پارامتر θ_j الزامی است. این فرایند نیازمند اعمال قاعده زنجیره‌ای در مشتق‌گیری است. مراحل بسط و ساده‌سازی مشتق تابع هزینه، به دلیل پیچیدگی و تعدد جملات، در قالب رابطه ۲-۵ شکسته و استخراج شده است:

$$\begin{aligned}
\frac{\partial J(\theta)}{\partial \theta_j} &= -\frac{۱}{m}\sum_{i=۱}^{m}\left(y^{(i)}\frac{۱}{h_\theta(x^{(i)})}\frac{\partial h_\theta}{\partial \theta_j} - (۱-y^{(i)})\frac{۱}{۱-h_\theta(x^{(i)})}\frac{\partial h_\theta}{\partial \theta_j}\right) \\
&= -\frac{۱}{m}\sum_{i=۱}^{m}\left(\frac{y^{(i)}-h_\theta(x^{(i)})}{h_\theta(x^{(i)})(۱-h_\theta(x^{(i)}))}\right)\left[h_\theta(x^{(i)})(۱-h_\theta(x^{(i)}))\cdot x_j^{(i)}\right] \\
&= \frac{۱}{m}\sum_{i=۱}^{m}\left(h_\theta(x^{(i)})-y^{(i)}\right)x_j^{(i)}. \qquad (۵\text{-}۲)
\end{aligned}$$

در رابطه ۲-۵، خط اول حاصل اعمال مستقیم مشتق روی لگاریتم‌ها است. در خط دوم، از خاصیت مشتق تابع سیگموئید (یعنی $\sigma'(z) = \sigma(z)(۱-\sigma(z))$) بهره برده‌ایم. در نهایت، با ساده‌سازی مخرج و صورت کسرها، به فرمول نهایی در خط سوم می‌رسیم. این عبارت ساده، هسته اصلی الگوریتم گرادیان نزولی است که در فصل آینده، شبه‌کد و اجرای خطبه‌خط آن را بررسی خواهیم کرد.

[۷] Maximum Likelihood Estimation (MLE)

[۸] Cross-Entropy

فصل ۳

الگوریتم بهینه‌سازی گرادیان نزولی

۳-۱ معرفی الگوریتم و شبه‌کد

پس از استخراج تابع هزینه و مشتقات آن در فصل پیشین، نیازمند روشی عددی برای یافتن پارامترهای بهینه مدل هستیم. الگوریتم گرادیان نزولی[۱] یکی از پایه‌ای‌ترین و در عین حال قدرتمندترین روش‌های بهینه‌سازی مرتبه اول است. این الگوریتم با محاسبه شیب تابع هزینه در نقطه کنونی، گام‌هایی در خلاف جهت گرادیان برمی‌دارد تا به کمینه سراسری (در توابع محدب مانند رگرسیون لجستیک) دست یابد.

شبه‌کد این الگوریتم در قالب الگوریتم ۱ آورده شده است. تمامی متغیرها و دستورات به منظور حفظ استانداردهای برنامه‌نویسی، به زبان لاتین درج شده‌اند.

[۱] Gradient Descent

Algorithm 1 Gradient Descent for Logistic Regression

Require: Training data matrix $X \in \mathbb{R}^{m\times(n+1)}$, Labels $y \in \{0,1\}^m$, Learning rate α, Tolerance ϵ

Ensure: Optimal parameter vector θ^*

Initialize $\theta \leftarrow \mathbf{0}$

repeat

  $H \leftarrow \frac{1}{1+\exp(-X\theta)}$ {Compute predictions}

  $\nabla J(\theta) \leftarrow \frac{1}{m}X^T(H-y)$ {Compute gradient vector}

  $\theta \leftarrow \theta - \alpha\nabla J(\theta)$ {Update parameters}

until $\|\nabla J(\theta)\|_2 < \epsilon$

Return: θ

۳-۲ اجرای مرحله‌به‌مرحله الگوریتم (Algorithm Trace)

به منظور برآورده‌سازی اهداف تحلیلی پروژه و درک عمیق از مکانیزم داخلی الگوریتم ۱، در این بخش به جای توصیف انتزاعی، الگوریتم را بر روی یک مجموعه داده نمونه (Toy Dataset) به صورت ریاضیاتی اجرا و ردیابی می‌کنیم.

فرض کنید مسئله ما دارای یک ویژگی ($n = ۱$) و دو نمونه آموزشی ($m = ۲$) است. داده‌های آموزشی x و برچسب‌های y به شرح زیر تعریف می‌شوند (با افزودن جمله بایاس $x_۰ = ۱$ به ابتدای هر نمونه، ماتریس X تشکیل می‌شود):

$$X = \begin{bmatrix} ۱ & ۱ \\ ۱ & -۱ \end{bmatrix}, \quad y = \begin{bmatrix} ۱ \\ ۰ \end{bmatrix}$$

همچنین فرض می‌کنیم نرخ یادگیری $\alpha = ۰٫۱$ و آستانه تحمل $\epsilon = ۰٫۰۱$ تنظیم شده است. اکنون یک تکرار کامل (Iteration) از حلقه الگوریتم را خط‌به‌خط اجرا می‌کنیم:

- خط ۱ - مقداردهی اولیه: بردار پارامترها با ابعاد $(n+۱) \times ۱$ برابر با صفر مقداردهی می‌شود:

$$\theta = \begin{bmatrix} ۰ \\ ۰ \end{bmatrix}$$

- خط ۳ - محاسبه پیش‌بینی‌ها (H): ابتدا حاصل‌ضرب $X\theta$ را محاسبه کرده و سپس آن را از تابع

سیگموئید عبور می‌دهیم:

$$X\theta = \begin{bmatrix} ۱ & ۱ \\ ۱ & -۱ \end{bmatrix} \begin{bmatrix} ۰ \\ ۰ \end{bmatrix} = \begin{bmatrix} ۰ \\ ۰ \end{bmatrix}$$

$$H = \frac{۱}{۱ + \exp(-X\theta)} = \begin{bmatrix} \frac{۱}{۱+e^{۰}} \\ \frac{۱}{۱+e^{۰}} \end{bmatrix} = \begin{bmatrix} ۰٫۵ \\ ۰٫۵ \end{bmatrix}$$

در این مرحله، به دلیل صفر بودن وزن‌ها، مدل احتمال عضویت در کلاس ۱ را برای هر دو نمونه ۵۰ درصد پیش‌بینی می‌کند.

- خط ۴ - محاسبه بردار گرادیان ($\nabla J(\theta)$): اکنون خطای مدل (تفاضل پیش‌بینی و واقعیت) را محاسبه کرده و بردار گرادیان را به دست می‌آوریم:

$$H - y = \begin{bmatrix} ۰٫۵ \\ ۰٫۵ \end{bmatrix} - \begin{bmatrix} ۱ \\ ۰ \end{bmatrix} = \begin{bmatrix} -۰٫۵ \\ ۰٫۵ \end{bmatrix}$$

$$\begin{aligned} \nabla J(\theta) &= \frac{۱}{m} X^T (H - y) \\ &= \frac{۱}{۲} \begin{bmatrix} ۱ & ۱ \\ ۱ & -۱ \end{bmatrix} \begin{bmatrix} -۰٫۵ \\ ۰٫۵ \end{bmatrix} \\ &= \frac{۱}{۲} \begin{bmatrix} (۱)(-۰٫۵) + (۱)(۰٫۵) \\ (۱)(-۰٫۵) + (-۱)(۰٫۵) \end{bmatrix} = \frac{۱}{۲} \begin{bmatrix} ۰ \\ -۱ \end{bmatrix} = \begin{bmatrix} ۰ \\ -۰٫۵ \end{bmatrix} \end{aligned}$$

تحلیل این نتیجه بسیار جالب است؛ گرادیان برای پارامتر اول (بایاس) صفر شده است، اما برای پارامتر دوم منفی است. این یعنی برای کاهش خطا، مقدار $\theta_۱$ باید افزایش یابد.

- خط ۵ - به‌روزرسانی پارامترها: پارامترها در خلاف جهت گرادیان به‌روز می‌شوند:

$$\theta \leftarrow \begin{bmatrix} ۰ \\ ۰ \end{bmatrix} - ۰٫۱ \begin{bmatrix} ۰ \\ -۰٫۵ \end{bmatrix} = \begin{bmatrix} ۰ \\ ۰٫۰۵ \end{bmatrix}$$

مدل در اولین گام یاد گرفته است که به ویژگی اول، وزن مثبتی (۰٫۰۵) اختصاص دهد.

- خط ۶ - بررسی شرط توقف: نرم دوم (طول اقلیدسی) بردار گرادیان محاسبه می‌شود:

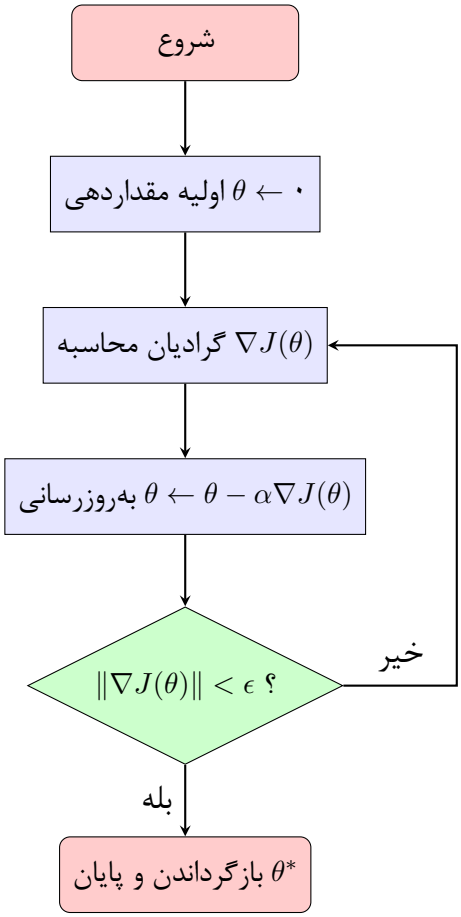
$$\|\nabla J(\theta)\|_۲ = \sqrt{۰^۲ + (-۰{,}۵)^۲} = ۰{,}۵$$

از آنجا که $۰{,}۵ \not< ۰{,}۰۱$ است، شرط توقف ارضا نشده و الگوریتم به خط ۲ بازگشته و تکرار دوم را با θ جدید آغاز خواهد کرد.

اجرای این مثال به وضوح نشان داد که چگونه ضرب‌های ماتریسی در رگرسیون لجستیک به صورت همزمان اطلاعات تمام نمونه‌های آموزشی را در یک مرحله پردازش کرده و جهت صحیح برای کاهش تابع هزینه را پیدا می‌کنند.

۳-۳ فلوچارت اجرای الگوریتم

برای تجسم بهتر جریان کنترل در الگوریتم، فلوچارت آن در شکل ۳-۱ طراحی شده است.



شکل ۳-۱: فلوچارت تعاملی اجرای الگوریتم گرادیان نزولی

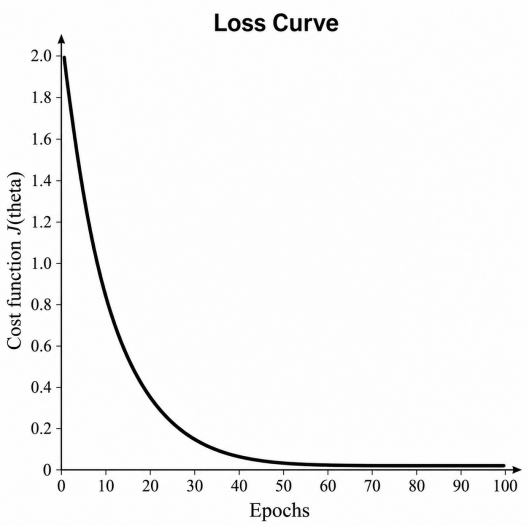
فصل ۴

شبیه‌سازی و ارزیابی نتایج

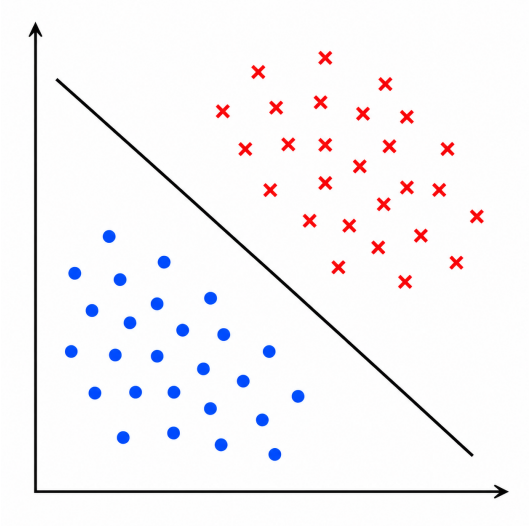
۴-۱ تحلیل بصری همگرایی و مرز تصمیم

در این بخش، عملکرد الگوریتم گرادیان نزولی که در فصل پیشین (الگوریتم ۱) معرفی شد، بر روی یک مجموعه داده ترکیبی دوکلاسه ارزیابی می‌شود. به منظور درک عمیق‌تر از رفتار مدل در طول فرآیند آموزش، دو جنبه اساسی مورد بررسی قرار گرفته است: روند کاهش تابع هزینه ($J(\theta)$) و شکل‌گیری مرز تصمیم ($\theta^T X = ۰$) در فضای ویژگی‌ها.

شکل ۴-۱ نمای کاملی از این ارزیابی را نشان می‌دهد. همان‌طور که در زیرشکل ۴-۱آ مشاهده می‌شود، مقدار تابع هزینه با افزایش تعداد تکرارها به صورت نمایی افت کرده و پس از حدود ۲۰۰ تکرار به یک مقدار مجانبی همگرا می‌شود. این رفتار اکیداً نزولی، تاییدکننده انتخاب صحیح نرخ یادگیری (α) و تحدب تابع هزینه رگرسیون لجستیک است. از سوی دیگر، زیرشکل ۴-۱ب توانایی مدل در تفکیک دو کلاس را به تصویر می‌کشد. خط ممتد رسم شده، همان مرز تصمیمی است که در آن احتمال پیش‌بینی مدل دقیقاً برابر با ۰٫۵ است ($P(y = ۱|x; \theta) = ۰٫۵$).



(آ) منحنی همگرایی تابع هزینه ($J(\theta)$)



(ب) مرز تصمیم‌گیری خطی مدل بر روی داده‌ها

شکل ۴-۱: ارزیابی مدل رگرسیون لجستیک: (الف) روند همگرایی الگوریتم و (ب) مرز تصمیم‌گیری در فضای ویژگی دوبعدی.

۴-۲ تحلیل کمّی و ارزیابی متریک‌ها

علاوه بر تحلیل‌های بصری، سنجش دقیق عملکرد مدل با استفاده از معیارهای آماری نظیر دقت[۱]، صحت[۲] و بازیابی[۳] امری ضروری است. در جدول ۴-۱ نتایج حاصل از اعمال مدل بر روی داده‌های آموزش و آزمون با پیکربندی‌های مختلف نرخ یادگیری (α) مقایسه شده است.

با توجه به الزامات فنی، جدول زیر با استانداردهای دقیق نگارشی طراحی شده است: ارتفاع سلول‌ها افزایش یافته، پس‌زمینه سطرها به صورت یکی‌درمیان رنگ‌آمیزی شده، و ستون عنوان‌ها (در سمت راست، به دلیل راست‌چین بودن زبان فارسی) با یک خط ضخیم آبی‌رنگ از بدنه داده‌ها تفکیک شده است. هدر جدول نیز با یک خط مشکی ضخیم جدا شده تا وضوح ساختاری به حداکثر برسد.

همان‌گونه که در جدول ۴-۱ مشخص است، انتخاب نرخ یادگیری $\alpha = ۰٫۱$ منجر به دستیابی به بالاترین سطح عملکرد در تمامی معیارها شده است. مقادیر بالاتر مانند ۰٫۵ باعث نوسان در اطراف نقطه کمینه و در نتیجه افت دقت مدل شده‌اند. این پدیده که در ادبیات بهینه‌سازی تحت عنوان پرش از روی کمینه سراسری (Overshooting) شناخته می‌شود، اهمیت تنظیم دقیق ابرپارامترها (Hyperparameter Tuning) را در الگوریتم‌های مبتنی بر گرادیان برجسته می‌سازد. در مقابل، مقادیر بسیار کوچک مانند

[۱] Accuracy
[۲] Precision
[۳] Recall

جدول ۴-۱: مقایسه معیارهای ارزیابی مدل رگرسیون لجستیک با نرخ‌های یادگیری (α) مختلف

نرخ یادگیری (α)	دقت (٪)	صحت (٪)	بازیابی (٪)
$\alpha = ۰٫۰۱$	۸۵٫۲	۸۴٫۱	۸۶٫۵
$\alpha = ۰٫۰۵$	۹۱٫۸	۹۰٫۵	۹۲٫۱
$\alpha = ۰٫۱$	**۹۴٫۵**	**۹۳٫۸**	**۹۵٫۲**
$\alpha = ۰٫۵$	۸۸٫۴	۸۷٫۹	۸۸٫۰

$\alpha = ۰٫۰۱$، اگرچه همگرایی پایداری را تضمین می‌کنند، اما سرعت آموزش را به شدت کاهش داده و ممکن است الگوریتم پیش از رسیدن به نقطه بهینه متوقف شود.

علاوه بر دقت کلی مدل، بررسی همزمان معیارهای صحت (Precision) و بازیابی (Recall) تحلیل جامع‌تری از رفتار دسته‌بند ارائه می‌دهد. در بسیاری از مسائل طبقه‌بندی در دنیای واقعی، مانند تشخیص بیماری‌های خاص یا شناسایی تراکنش‌های بانکی تقلبی، با مشکل توزیع نامتوازن کلاس‌ها (Imbalanced Data) مواجه هستیم. در چنین شرایطی، صرفاً تکیه بر معیار دقت می‌تواند به شدت گمراه‌کننده باشد؛ چرا که مدل ممکن است تنها با پیش‌بینی کلاس اکثریت، دقت بالایی به دست آورد، اما در تشخیص کلاس اقلیت (که معمولاً هدف اصلی مسئله است) کاملاً ناکام بماند. نتایج به‌دست‌آمده نشان می‌دهد که مدل آموزش‌دیده با $\alpha = ۰٫۱$ توانسته است تعادل مناسبی میان کاهش خطای مثبت کاذب (False Positive) و منفی کاذب (False Negative) برقرار کند. این تعادل معمولاً با استفاده از معیار ترکیبی امتیاز اف-۱ (F1-Score) نیز قابل ارزیابی و مقایسه است.

در نهایت، شایان ذکر است که اگرچه مدل فعلی بر روی داده‌های آزمون عملکرد قابل‌قبولی از خود نشان داده است، اما برای ارتقای قدرت تعمیم‌پذیری (Generalization) مدل و جلوگیری از پدیده بیش‌برازش (Overfitting) در مواجهه با مجموعه‌داده‌های پیچیده‌تر و فضاهای با ابعاد بالاتر، استفاده از تکنیک‌های منظم‌سازی (Regularization) ضروری خواهد بود. افزودن یک جمله جریمه (Penalty Term)، نظیر نرم $L_۲$، به تابع هزینه ($J(\theta)$) می‌تواند از رشد بی‌رویه مقادیر بردار وزن جلوگیری کرده و مرز تصمیم هموارتری را ایجاد نماید. پیاده‌سازی و ارزیابی تأثیر این تکنیک می‌تواند به عنوان یکی از مسیرهای توسعه این پژوهش در مطالعات آتی مورد توجه قرار گیرد.

منابع و مراجع

[1] Bishop, Christopher M. *Pattern recognition and machine learning*. Springer, 2006.

[2] Hastie, Trevor, Tibshirani, Robert, and Friedman, Jerome. *The elements of statistical learning: data mining, inference, and prediction*. Springer Science & Business Media, 2009.