

تکلیف کامپیوتری چهارم - درس شناسایی الگو

وحید ملکی
شماره دانشجویی: ۴۰۳۱۳۰۰۴

۲۵ آذر ۱۴۰۴

۱ مقدمه

در این تمرین، هدف پیاده‌سازی و ارزیابی الگوریتم ماشین بردار پشتیبان (Support Vector Machine - SVM) برای طبقه‌بندی تصاویر مجموعه داده Fashion-MNIST است. این کار در دو بخش انجام شده است: ابتدا پیاده‌سازی الگوریتم از پایه با استفاده از کتابخانه NumPy و ریاضیات مربوط به مسئله دوگان (Dual Problem)، و سپس پیاده‌سازی با استفاده از کتابخانه Scikit-Learn جهت اعتبارسنجی نتایج. در نهایت تاثیر پارامترهای مختلف مانند نوع کرنل و ضریب جریمه (C) بر دقت مدل بررسی شده است.

۲ بخش اول: پیاده‌سازی SVM از پایه (NumPy)

۱.۲ آماده‌سازی داده‌ها

مجموعه داده Fashion-MNIST شامل تصاویر سیاه‌وسفید 28×28 پیکسل از ۱۰ کلاس مختلف پوشاک است. در این پیاده‌سازی، مراحل زیر برای آماده‌سازی داده‌ها طی شد:

۱. بارگذاری و انتخاب زیرمجموعه: به دلیل حجم بالای داده‌ها و هزینه محاسباتی بالای آموزش SVM (که با تعداد نمونه‌ها رابطه نمایی دارد)، پس از بارگذاری کل داده‌ها، آن‌ها به صورت تصادفی مخلوط (Shuffle) شدند و یک زیرمجموعه شامل 2000 نمونه انتخاب گردید.

۲. تقسیم داده‌ها: داده‌ها با استفاده از روش Stratified Split به دو بخش آموزش (80%) و آزمون (20%) تقسیم شدند تا توزیع کلاس‌ها در هر دو مجموعه حفظ شود.

۳. نرمال‌سازی (Standardization): از آنجا که SVM یک الگوریتم مبتنی بر فاصله است، مقیاس ویژگی‌ها تاثیر زیادی بر عملکرد آن دارد. تمام ویژگی‌ها با کسر میانگین و تقسیم بر انحراف معیار نرمال شدند:

$$X_{scaled} = \frac{X - \mu}{\sigma + \epsilon} \quad (1)$$

که در آن ϵ مقدار بسیار کوچکی برای جلوگیری از تقسیم بر صفر است.

۲.۲ ریاضیات و ساختار مدل

الگوریتم پیاده‌سازی شده بر اساس حل مسئله دوگان لاگرانژ (Lagrange Dual Problem) بنا شده است. هدف پیدا کردن ضرایب α است که تابع هدف زیر را ماکزیم کنند:

$$\max_{\alpha} \sum_{i=1}^m \alpha_i - \frac{1}{2} \sum_{i=1}^m \sum_{j=1}^m \alpha_i \alpha_j y_i y_j K(x_i, x_j) \quad (۲)$$

تحت قیود:

$$0 \leq \alpha_i \leq C, \quad \forall i \quad (۳)$$

$$\sum_{i=1}^m \alpha_i y_i = 0 \quad (۴)$$

در کد نوشته شده، برای حل این مسئله بهینه‌سازی از روش گرادیان صعودی (Gradient Ascent) استفاده شده است. گرادیان تابع هدف نسبت به بردار α به صورت زیر محاسبه می‌شود:

$$\nabla L(\alpha) = \mathbf{1} - y \odot (K \cdot (\alpha \odot y)) \quad (۵)$$

که در آن \odot ضرب درایه‌ای و K ماتریس کرنل است. در هر تکرار، مقادیر α به‌روزرسانی شده و سپس جهت ارضای قید $0 \leq \alpha \leq C$ ، مقادیر آن به بازه $[0, C]$ محدود (Clip) می‌شوند.

۳.۲ توابع کرنل (Kernel Functions)

دو نوع کرنل در تابع `_kernel` پیاده‌سازی شد:

• کرنل خطی (Linear):

$$K(x_1, x_2) = x_1 \cdot x_2^T \quad (۶)$$

• کرنل RBF (توزیع گاوسی):

$$K(x_1, x_2) = \exp(-\gamma \|x_1 - x_2\|^2) \quad (۷)$$

که پارامتر γ به صورت پیش‌فرض برابر با $\frac{1}{n_{features}}$ در نظر گرفته شد.

۴.۲ محاسبه بایاس (b) و تابع تصمیم

پس از همگرایی گرادیان، بردارهای پشتیبان (Support Vectors) شناسایی شدند (نقاطی که $\alpha_i > 0$). مقدار بایاس b با میانگین‌گیری روی بردارهای پشتیبان محاسبه شد:

$$b = \frac{1}{N_{SV}} \sum_{s \in SV} (y_s - \sum_i \alpha_i y_i K(x_i, x_s)) \quad (۸)$$

تابع تصمیم نهایی برای یک داده ورودی جدید x برابر است با:

$$f(x) = \text{sign}(\sum_i \alpha_i y_i K(x_i, x) + b) \quad (۹)$$

۵.۲ استراتژی چندکلاسه (One-vs-Rest)

از آنجا که SVM ذاتاً یک طبقه‌بند دودویی است، برای مجموعه داده Fashion-MNIST که دارای ۱۰ کلاس است، از استراتژی «یکی در برابر همه» (One-vs-Rest) استفاده شد. به این صورت که ۱۰ مدل مجزا آموزش داده شدند؛ در مدل k -ام، نمونه‌های کلاس k برچسب +1 و سایر نمونه‌ها برچسب -1 گرفتند. برای پیش‌بینی نهایی، کلاسی انتخاب شد که بیشترین مقدار تابع تصمیم (امتیاز) را تولید کرد.

۳ ارزیابی و بهینه‌سازی پارامترها

جهت یافتن بهترین پارامترها، از روش اعتبارسنجی متقاطع ۱۰ لایه (10-Fold Cross-Validation) روی داده‌های آموزشی استفاده شد. مقادیر مختلف پارامتر جریمه $C \in \{0.01, 0.1, 1, 10, 100\}$ برای هر دو کرنل خطی و RBF مورد ارزیابی قرار گرفتند.

۱.۳ تحلیل نتایج پیاده‌سازی Scratch

همانطور که در نمودار سمت راست شکل ۱ مشاهده می‌شود:

- کرنل خطی: عملکرد بسیار ضعیفی داشت (دقت حدود 13% تا 16%). این نشان می‌دهد که داده‌های تصویر پوشاک در فضای ویژگی اصلی به صورت خطی تفکیک‌پذیر نیستند.
 - کرنل RBF: عملکرد بسیار بهتری نشان داد. با افزایش مقدار C ، دقت مدل افزایش یافت و در مقادیر $C = 10$ و $C = 100$ به ثبات رسید (حدود 83%). این نشان می‌دهد که مدل با مرزهای غیرخطی و جریمه سخت‌گیرانه‌تر برای خطاهای طبقه‌بندی، بهتر عمل می‌کند.
- بهترین پارامترهای یافت شده عبارتند از:

- کرنل: rbf
- مقدار C : 10
- دقت نهایی روی داده تست: 82.62%

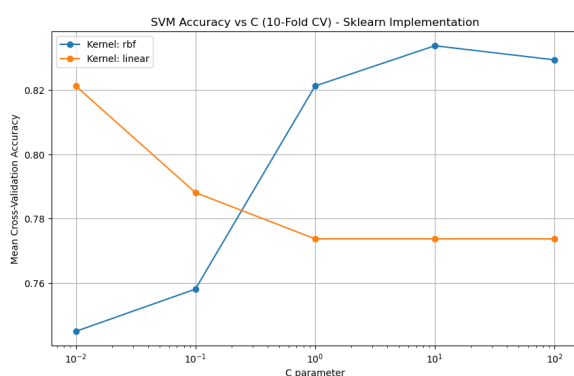
۴ بخش دوم: پیاده‌سازی با Scikit-Learn

برای مقایسه، همین مراحل با استفاده از کلاس‌های SVC، OneVsRestClassifier و GridSearchCV از کتابخانه Scikit-Learn انجام شد. نتایج حاصل از Scikit-Learn تطابق بسیار بالایی با پیاده‌سازی دستی داشت که صحت عملکرد کد نوشته شده را تایید می‌کند. بهترین مدل در این حالت نیز کرنل RBF با $C = 10$ بود که دقتی برابر با 82.50% روی داده‌های تست نهایی ثبت کرد.

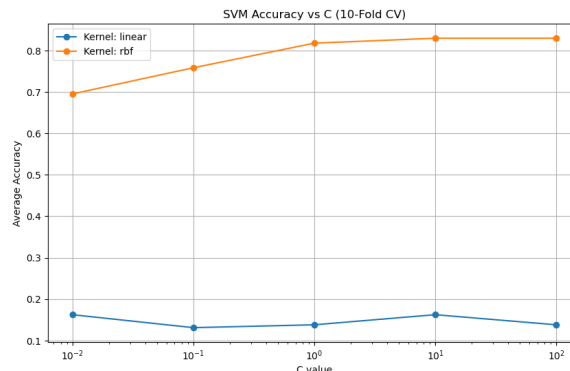
۵ تحلیل نمودارها و مقایسه

۱.۵ مقایسه روند دقت بر حسب C

در شکل ۱ روند تغییرات دقت بر اساس پارامتر C برای هر دو پیاده‌سازی نمایش داده شده است.



Scikit-Learn پیاده‌سازی (ب)



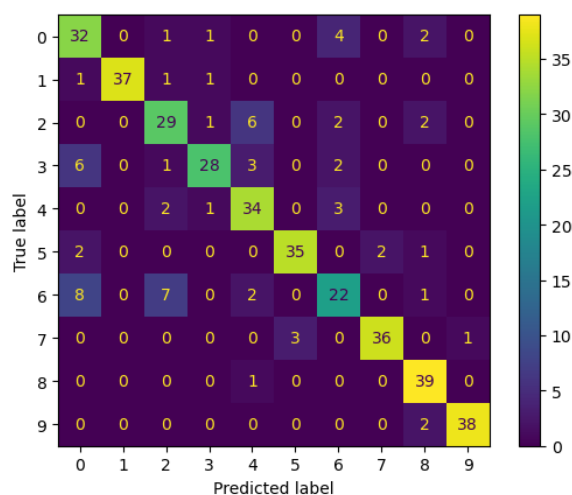
(آ) پیاده‌سازی دستی (NumPy)

شکل ۱: نمودار تغییرات دقت میانگین (Cross-Validation) بر حسب مقدار C برای کرنل‌های خطی و RBF.

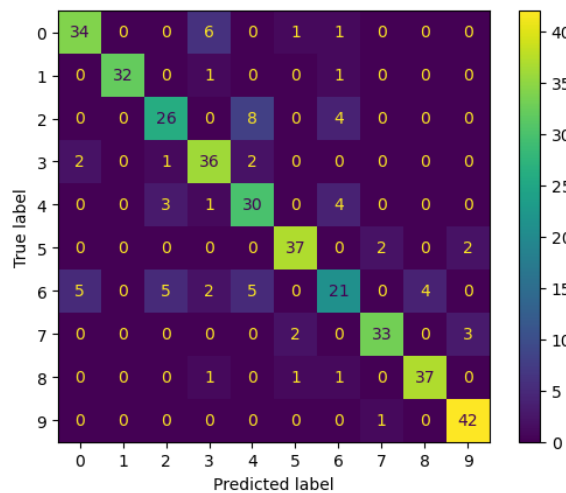
در هر دو نمودار، کرنل RBF (خط نارنجی در نمودار دستی و خط آبی در نمودار Sklearn) با افزایش C بهبود یافته و سپس اشباع می‌شود. تفاوت جزئی در نمودار خطی ممکن است ناشی از تفاوت در روش‌های بهینه‌سازی داخلی (Gradient Ascent ساده در برابر LibSVM بهینه شده) باشد.

۲.۵ ماتریس درهم‌ریختگی (Confusion Matrix)

شکل ۲: ماتریس درهم‌ریختگی مدل بهینه را برای هر دو روش نشان می‌دهد.



Scikit-Learn پیاده‌سازی (ب)



(آ) پیاده‌سازی دستی

شکل ۲: ماتریس درهم‌ریختگی برای بهترین مدل انتخاب شده روی داده‌های تست.

تحلیل ماتریس درهم‌ریختگی نکات زیر را روشن می‌کند:

- کلاس ۶ (Shirt) بیشترین نرخ خطا را دارد و اغلب با کلاس ۰ (T-shirt/Top)، کلاس ۲ (Pullover) و کلاس ۴ (Coat) اشتباه گرفته می‌شود. این خطا به دلیل شباهت ظاهری زیاد این دسته از پوشاک قابل انتظار است.
- کلاس ۱ (Trousers) و کلاس ۹ (Ankle boot) بالاترین دقت تشخیص را دارند (حدود 94% تا 98%) که نشان‌دهنده تمایز واضح ویژگی‌های آن‌ها نسبت به سایر کلاس‌هاست.
- تطابق بالای مقادیر قطر اصلی در هر دو ماتریس نشان می‌دهد که پیاده‌سازی دستی به درستی توانسته است الگوهای موجود را مشابه با پیاده‌سازی استاندارد Scikit-Learn فرا بگیرد.

۶ نتیجه‌گیری

در این تمرین، الگوریتم SVM با موفقیت پیاده‌سازی شد. نتایج نشان داد که برای داده‌های تصویری Fashion-MNIST، استفاده از کرنل غیرخطی RBF ضروری است. مدل دستی توانست به دقتی معادل 82.62% دست یابد که با دقت 82.50% حاصل از کتابخانه Scikit-Learn قابل رقابت است.