تکلیف کامپیوتری چهارم - درس شناسایی الگو

وحید ملکی
شماره دانشجویی: ۴۰۳۱۳۰۰۴

۲۵ آذر ۱۴۰۴

۱ مقدمه و آماده‌سازی داده‌ها

در این تکلیف، هدف پیاده‌سازی و ارزیابی ماشین بردار پشتیبان (SVM) بر روی مجموعه داده Fashion-MNIST است. این مجموعه داده شامل تصاویر 28×28 پیکسلی از پوشاک در 10 کلاس مختلف است. به دلیل حجم بالای داده‌ها و محدودیت‌های محاسباتی در پیاده‌سازی دستی، از یک زیرمجموعه شامل 2000 نمونه استفاده شد. ابتدا داده‌ها بارگذاری شده و به طور تصادفی مخلوط (Shuffle) شدند. سپس 20 درصد از داده‌ها (معادل 400 نمونه) به عنوان داده‌های آزمون (Test Set) و 80 درصد باقی‌مانده (معادل 1600 نمونه) به عنوان داده‌های آموزش (Train Set) جدا شدند. برای بهبود عملکرد الگوریتم‌های مبتنی بر گرادیان، داده‌ها نرمال‌سازی شدند تا میانگین صفر و واریانس یک داشته باشند.

۲ پیاده‌سازی مدل SVM با استفاده از NumPy

در بخش اول، الگوریتم SVM بدون استفاده از کتابخانه‌های آماده و صرفاً با استفاده از جبر خطی در NumPy پیاده‌سازی شد.

۱.۲ ریاضیات و منطق کلاس SVMScratch

این کلاس هسته اصلی الگوریتم را تشکیل می‌دهد و برای حل مسأله بهینه‌سازی SVM از روش «گرادیان صعودی» (Gradient Ascent) بر روی فرم دوگان (Dual Form) استفاده می‌کند.

۱.۱.۲ تابع _kernel

این تابع وظیفه محاسبه ماتریس کرنل (شباهت بین نمونه‌ها) را بر عهده دارد. دو نوع کرنل پیاده‌سازی شد:

- **کرنل خطی (Linear):** به صورت ضرب داخلی دو بردار ویژگی تعریف می‌شود:
$$K(x_i, x_j) = x_i^T x_j \quad (۱)$$
- **کرنل RBF:** که بر اساس فاصله اقلیدسی عمل می‌کند و قابلیت جداسازی غیرخطی را فراهم می‌آورد:
$$K(x_i, x_j) = \exp(-\gamma ||x_i - x_j||^2) \quad (۲)$$

۲.۱.۲ تابع fit

این تابع ضرایب لاگرانژ (α) را یاد می‌گیرد. هدف ما بیشینه‌سازی تابع هدف دوگان است:

$$L(\alpha) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j K(x_i, x_j) \quad (۳)$$

مشتق این تابع نسبت به α برابر است با:

$$\nabla L = 1 - y_i \sum_{j=1}^{n} \alpha_j y_j K(x_i, x_j) \quad (۴)$$

در حلقه یادگیری، مقادیر α با نرخ یادگیری (lr) در جهت گرادیان به‌روزرسانی می‌شوند. همچنین شرط محدودیت جعبه‌ای ($0 \leq \alpha \leq C$) با استفاده از تابع clip اعمال می‌شود. در نهایت، مقدار بایاس (b) با استفاده از بردارهای پشتیبان (نقاطی که $0 < \alpha < C$) محاسبه می‌شود.

۳.۱.۲ تابع decision_function

برای پیش‌بینی امتیاز یک نمونه جدید x، از رابطه زیر استفاده می‌شود:

$$f(x) = \sum_{i \in SV} \alpha_i y_i K(x_i, x) + b \quad (۵)$$

این مقدار نشان‌دهنده فاصله نمونه تا ابرصفحه جداکننده است.

۲.۲ استراتژی چندکلاسه و ارزیابی

از آنجا که SVM ذاتاً یک طبقه‌بند باینری است، برای مسأله ۱۰ کلاسه از استراتژی «یکی در برابر بقیه» (One-vs-Rest) در تابع train_svm استفاده شد. به ازای هر کلاس، یک مدل آموزش داده می‌شود که آن کلاس را $+1$ و سایرین را -1 در نظر می‌گیرد. تابع predict خروجی تمام ۱۰ مدل را دریافت کرده و کلاسی که بیشترین امتیاز (بیشترین فاصله مثبت از مرز تصمیم) را داشته باشد، به عنوان برچسب نهایی انتخاب می‌کند (argmax).

۳.۲ اعتبارسنجی متقاطع (Cross-Validation)

برای انتخاب بهترین فراپارامترها، از روش 10-Fold Cross-Validation استفاده شد. تابع get_k_folds داده‌های آموزش را به ۱۰ بخش مساوی تقسیم می‌کند. نتایج میانگین دقت برای مقادیر مختلف C و کرنل‌ها به شرح زیر به دست آمد:

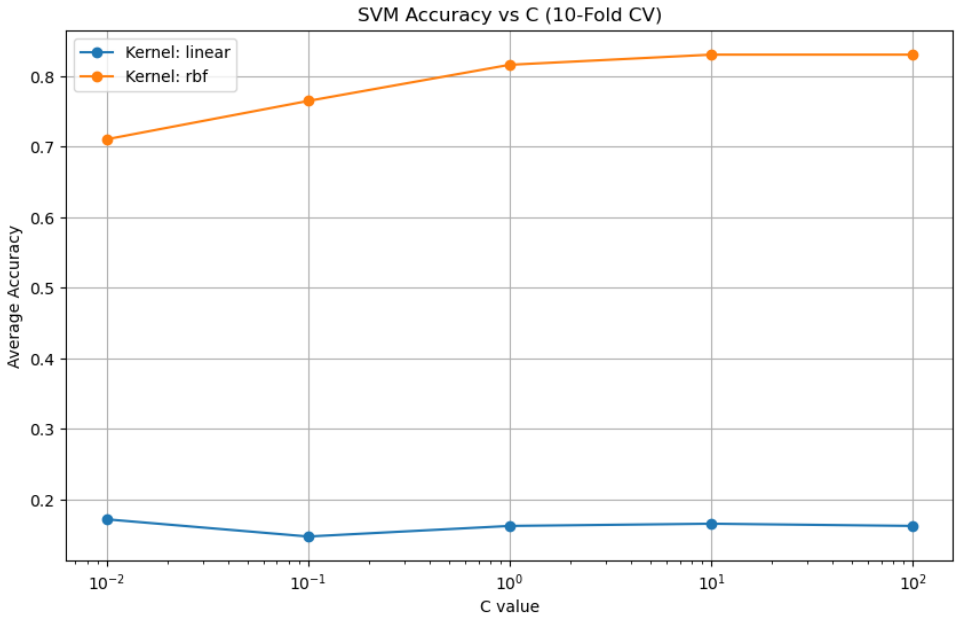
- **کرنل خطی (Linear):** همان‌طور که مشاهده می‌شود، دقت در تمامی مقادیر C بسیار پایین (حدود 16%) است.

```
%52.61 >= 1=C | %57.41 >= 1.0=C | %91.71 >= 10.0=C
       %52.61 >= 001=C |   %65.61 >= 01=C
```

الگوریتم ماهیت دستی، پیاده‌سازی در ضعیف عملکرد این اصلی دلیل **خطی: ضعیف عملکرد تحلیل** کرنل خطای سطح بعد)، 784(بالا ابعاد با ویژگی‌های فضای در است. ساده صعودی گرادیان $\sum \alpha_i y_i = 0$(تساوی شرط دقیق اعمال بدون ساده گرادیان الگوریتم و است ناهموار بسیار خطی و شود همگرا نمی‌تواند، Coordinate Descent یا SMO مانند پیشرفته‌تر روش‌های از استفاده بدون و می‌شود. نوسان دچار

- **کرنل RBF:** عملکرد بسیار بهتری نشان داد و با افزایش C دقت بهبود یافت.

```
%36.18 >= 1=C | %05.67 >= 1.0=C | %60.17 >= 10.0=C
       %60.38 >= 001=C |    %60.38 >= 01=C
```



شکل ۱: نمودار تغییرات دقت بر حسب مقدار C در پیاده‌سازی دستی (NumPy)

۴.۲ نتیجه نهایی مدل NumPy

بهترین پارامترهای یافت شده عبارتند از kernel='rbf' و $C = 10$ که دقت اعتبارسنجی 83.06% را داشت. پس از آموزش مجدد مدل با این پارامترها روی کل داده‌های آموزش، دقت نهایی روی داده‌های آزمون برابر با 81.61% به دست آمد.

۳ پیاده‌سازی با استفاده از Scikit-Learn

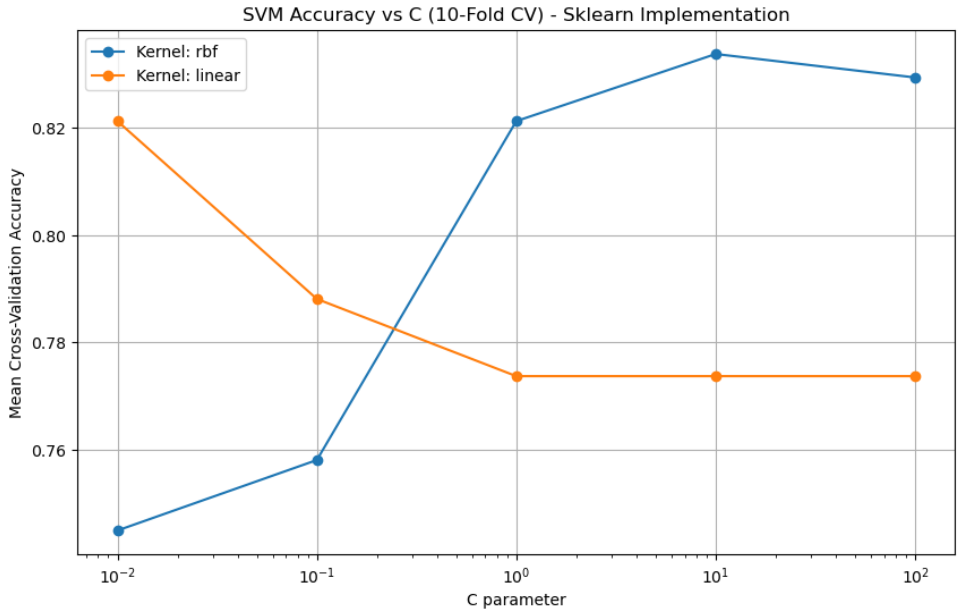
در این بخش از کتابخانه استاندارد sklearn برای پیاده‌سازی استفاده شد که از حل‌گر قدرتمند LibSVM استفاده می‌کند.

۱.۳ اجزای استفاده شده

- **StandardScaler:** برای نرمال‌سازی داده‌ها (میانگین صفر و انحراف معیار یک) استفاده شد که تأثیر بسزایی در همگرایی SVM دارد.
- **OneVsRestClassifier:** این کلاس به صورت خودکار استراتژی یکی در برابر بقیه را برای مسائل چندکلاسه مدیریت می‌کند.
- **StratifiedKFold:** برخلاف تقسیم تصادفی ساده، این روش تضمین می‌کند که نسبت کلاس‌ها در هر Fold حفظ شود که برای داده‌های غیرمتوازن حیاتی است.
- **GridSearchCV:** جستجوی شبکه‌ای برای یافتن بهترین ترکیب فراپارامترها (C و نوع کرنل) همراه با اعتبارسنجی متقاطع.

۲.۳ تحلیل نتایج Scikit-Learn

نتایج جستجوی شبکه‌ای نشان داد که پیاده‌سازی کتابخانه‌ای رفتار پایدارتری دارد. نمودار دقت بر حسب پارامتر C در شکل زیر (در صورت فعال‌سازی) قابل مشاهده است. نکته قابل توجه این است که در پیاده‌سازی sklearn، کرنل خطی نیز عملکرد قابل قبولی دارد، اما همچنان کرنل RBF برتری دارد.



شکل ۲: نمودار تغییرات دقت بر حسب مقدار C در پیاده‌سازی Scikit-Learn

بهترین مدل در این حالت نیز با پارامترهای $C = 10$ و کرنل RBF به دست آمد که دقت اعتبارسنجی آن 83.38% بود. دقت نهایی روی داده‌های آزمون برابر با 82.50% شد.

۴ مقایسه و نتیجه‌گیری

مقایسه نتایج نهایی نشان می‌دهد که پیاده‌سازی دستی ما با وجود سادگی، عملکرد بسیار نزدیکی به کتابخانه بهینه‌شده Scikit-Learn دارد (81.61% در مقابل 82.50%).

۱۰۴ تحلیل گزارش کلاس‌بندی (Classification Report)

با مقایسه گزارش‌های کلاس‌بندی هر دو مدل، نتایج زیر حاصل می‌شود:

- **کلاس‌های موفق:** هر دو مدل در تشخیص کلاس 1 (شلوار) و کلاس 9 (کفش ساق‌دار) بسیار عالی عمل کرده‌اند (دقت بالای 95%). این نشان می‌دهد ویژگی‌های این کلاس‌ها تمایز بالایی دارند.
- **کلاس‌های چالش‌برانگیز:** کلاس 6 (پیراهن) در هر دو مدل کمترین دقت را داشته است (Recall حدود 0.55). این کلاس به دلیل شباهت ظاهری زیاد به تی‌شرت (کلاس 0) و پلوشرت (کلاس 2)، بیشترین خطا را ایجاد کرده است.
- **پایداری:** مدل Scikit-Learn در کلاس‌های دشوار کمی متعادل‌تر عمل کرده است، اما الگوی خطاها در هر دو مدل یکسان است که نشان‌دهنده ماهیت داده‌هاست.

در جدول زیر گزارش نهایی مدل Scikit-Learn آورده شده است:

جدول ۱: گزارش کلاس‌بندی نهایی (بهترین مدل)

Class	Precision	Recall	F1-Score	Support
0	0.65	0.80	0.72	40
1	1.00	0.93	0.96	40
2	0.71	0.72	0.72	40
3	0.88	0.70	0.78	40
4	0.74	0.85	0.79	40
5	0.92	0.88	0.90	40
6	0.67	0.55	0.60	40
7	0.95	0.90	0.92	40
8	0.83	0.97	0.90	40
9	0.97	0.95	0.96	40
Accuracy			**0.82**	400