باسمه تعالی

گزارش تمرین اول درس یادگیری ماشین

محمد مهدی خرّم آبادی – ۴۰۳۱۳۱۶۱۰۰۵

بخش اول: عنوان، چکیده، و مقدمه

عنوان پژوهش:

طراحی آزمایش، پیاده سازی و تحلیل عمیق طبقه بندهای کلاسیک یادگیری ماشین در مواجهه با چالش های ابعاد بالا، نویز و غیرخطی بودن.

مقدمه و معرفی مسئله

یادگیری ماشین به عنوان یکی از ارکان اصلی هوش مصنوعی، ابزارهای متنوعی را برای حل مسائل طبقه بندی فراهم کرده است. با این حال، هیچ مدلی وجود ندارد که برای همه نوع داده مناسب باشد (قضیه No Free Lunch) .در دنیای واقعی، داده ها همیشه تمیز و استاندارد نیستند. پژوهشگران و مهندسان داده همواره با چالش های متعددی روبرو هستند که مهم ترین آنها عبارتند از: زیاد بودن تعداد ویژگی ها یا ابعاد بالا که منجر به پدیده "نفرین ابعاد" می شود، وجود نویز و داده های پرت که باعث انحراف مدل می شوند، و ساختارهای هندسی پیچیده و غیرخطی که روش های خطی ساده قادر به تفکیک آنها نیستند.

هدف اصلی این پژوهش، بررسی تجربی و تحلیل رفتار هفت الگوریتم کلاسیک و پرکاربرد یادگیری ماشین شامل kNN ، SVM، درخت تصمیم، بیز ساده، رگرسیون لجستیک، مدل های خطی و شبکه های عصبی MLP است. ما قصد داریم با طراحی یک آزمایش کنترل شده، نقاط قوت و ضعف هر یک از این مدل ها را در مواجهه با سه سناریوی مختلف ارزیابی کنیم. این تحلیل به ما کمک می کند تا درک عمیق تری از مفاهیمی همچون موازنه بایاس-واریانس (Bias-Variance Tradeoff) و تعمیم پذیری مدل ها پیدا کنیم.

معرفی دقیق دیتاست ها

برای اینکه نتایج این آزمایش قابل تعمیم باشد، سه مجموعه داده با ویژگی های ساختاری کاملاً متفاوت انتخاب شده است. انتخاب این دیتاست ها تصادفی نبوده و هر کدام با هدف به چالش کشیدن جنبه خاصی از مدل ها گزینش شده اند:

۱ .مجموعه داده ارقام - (Digits) چالش ابعاد بالا

این دیتاست شامل تصاویر ۸ در ۸ پیکسلی از اعداد دست نویس است. هر تصویر به صورت یک بردار با ۶۴ ویژگی (پیکسل) نمایش داده می شود. هدف از انتخاب این داده، بررسی عملکرد مدل ها در فضای "با بعد بالا " (High-Dimensional) است. در چنین فضایی، فاصله بین نقاط داده معنای سنتی خود را از دست می دهد و مدل هایی که بر اساس فاصله اقلیدسی کار می کنند ممکن است دچار مشکل شوند.

۲ .مجموعه داده سرطان سینه با نویز افزوده - (Breast Cancer + Noise) چالش پایداری

دیتاست اصلی سرطان سینه ویسکانسین شامل ۳۰ ویژگی استخراج شده از تصاویر پزشکی است. اما برای شبیه سازی شرایط واقعی و دشوارتر کردن مسئله، ما به صورت دستی نویز تصادفی (Gaussian Noise) به داده ها اضافه کردیم. این کار باعث می شود مرز بین کلاس ها مخدوش شود. هدف در اینجا بررسی این موضوع است که کدام مدل ها دچار بیش برازش (Overfitting) روی نویز می شوند و کدام مدل ها می توانند سیگنال اصلی را از نویز تشخیص دهند.

۳ .مجموعه داده ماه ها - (Make Moons) چالش غیرخطی بودن

این یک دیتاست مصنوعی دو بعدی است که شامل دو کلاس به شکل دو نیم دایره در هم تنیده است. ویژگی اصلی این داده این است که با هیچ خط مستقیمی نمی توان دو کلاس را از هم جدا کرد. این دیتاست به عنوان محکی برای سنجش توانایی مدل ها در ایجاد مرزهای تصمیم پیچیده و غیرخطی استفاده می شود.

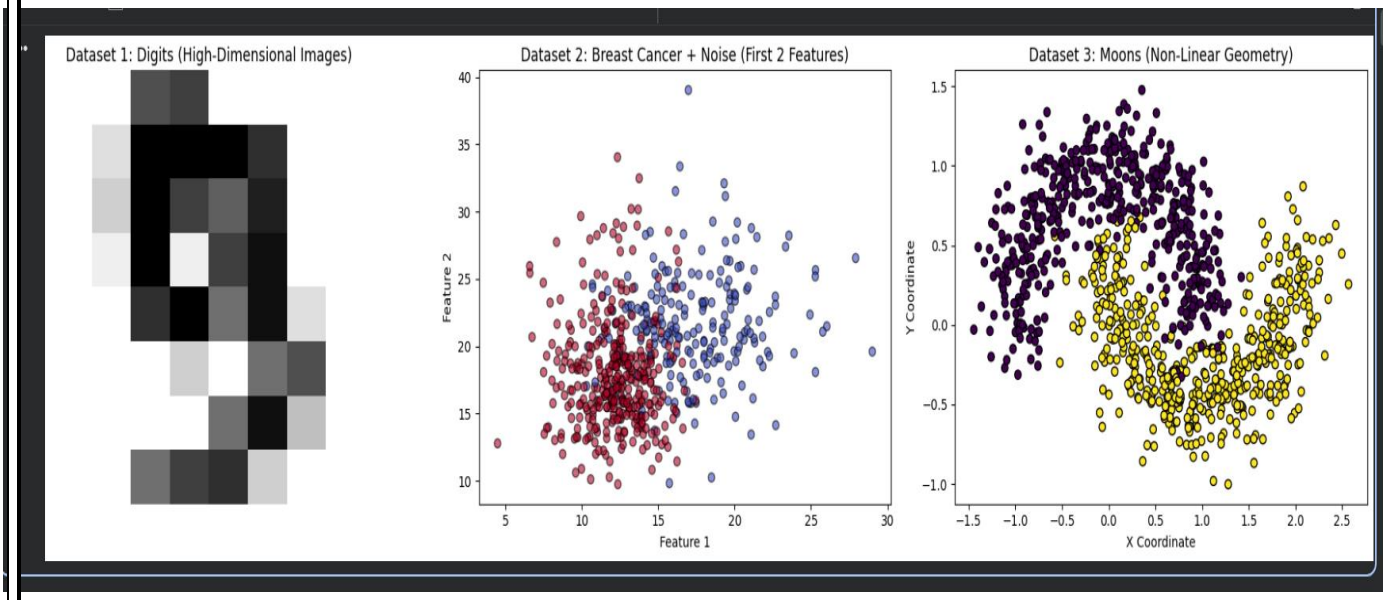


Figure ۱تصویر نمونه ای از هر دیتا ست

بخش دوم: روش‌شناسی، پیش‌پردازش و ابزارها

جزئیات کامل روش ها و پیش پردازش(Preprocessing)

یکی از اصول بنیادین در داده کاوی این است که کیفیت خروجی مدل مستقیماً به کیفیت داده ورودی وابسته است .(Garbage In, Garbage Out) بنابراین، پیش از اعمال هرگونه الگوریتم یادگیری، مراحل زیر برای آماده سازی داده ها انجام شد:

۱ .تقسیم داده ها(Train-Test Split)

برای اطمینان از اعتبار نتایج و جلوگیری از نشت اطلاعات(Data Leakage) ، هر دیتاست به دو بخش مجزا تقسیم شد: ۷۰ درصد داده ها برای آموزش مدل (Training Set) و ۳۰ درصد برای ارزیابی نهایی (Test Set) در نظر گرفته شد. نکته مهم در این مرحله استفاده از روش Stratified Sampling بود. این روش تضمین می کند که نسبت کلاس ها (مثلاً نسبت بیماران سالم به بیمار) در هر دو مجموعه آموزش و تست دقیقاً یکسان باقی بماند تا ارزیابی عادلانه باشد.

۲ .نرمال سازی داده ها(Standardization)

بسیاری از الگوریتم های مورد استفاده در این پژوهش، از جملهSVM ، kNNو شبکه های عصبی، نسبت به

مقیاس ویژگی ها حساس هستند. برای مثال، اگر یک ویژگی مقادیر بین ۰ تا ۱ داشته باشد و ویژگی دیگر بین ۰ تا ۱۰۰۰، مدل وزن بیشتری به ویژگی بزرگتر می دهد که نادرست است. به همین دلیل، از روش استانداردسازی (StandardScaler) استفاده کردیم. در این روش، میانگین هر ویژگی به صفر و انحراف معیار آن به یک تبدیل می شود. نکته تکنیکی مهم این است که پارامترهای نرمال سازی تنها از روی داده های آموزش محاسبه شدند و سپس روی داده های تست اعمال گردیدند.

نقش دقیق کتابخانه ها و ابزارها

برای پیاده سازی این پژوهش از زبان برنامه نویسی پایتون و اکوسیستم قدرتمند آن استفاده شده است. نقش کلیدی هر کتابخانه به شرح زیر است:

- کتابخانه NumPy : این کتابخانه پایه و اساس محاسبات عددی در پایتون است. در این پروژه، تمام عملیات برداری، کار با ماتریس های داده و همچنین تولید نویز مصنوعی با توزیع گاوسی توسط توابع این کتابخانه انجام شده است.
- کتابخانه Pandas : برای مدیریت ساختار داده ها و ایجاد جداول نتایج نهایی از این ابزار استفاده شد. دیتافریم های پانداس به ما اجازه دادند تا نتایج دقت و زمان اجرا را به صورت مرتب و خوانا دسته بندی کنیم.
- کتابخانه Scikit-Learn (sklearn): این کتابخانه قلب تپنده پروژه است. تمام الگوریتم های یادگیری ماشین (مانندSVC ، MLPClassifier)، ابزارهای پیش پردازش(StandardScaler) ، و ابزارهای اعتبارسنجی (GridSearchCV) از این پکیج فراخوانی شده اند. جامعیت این کتابخانه باعث شد تا بتوانیم همه مدل ها را با یک رابط کاربری یکسان پیاده سازی کنیم.
- کتابخانه Matplotlib وSeaborn : تمامی نمودارهای بصری، از جمله منحنی های یادگیری و رسم مرزهای تصمیم، با استفاده از قابلیت های گرافیکی این دو کتابخانه تولید شده اند.

طراحی آزمایش(Experiment Design)

برای اینکه مقایسه بین مدل ها منصفانه و علمی باشد، یک چارچوب آزمایشی دقیق طراحی شد:

الف) تنظیم فراپارامترها(Hyperparameter Tuning) : برای هر مدل، صرفاً به تنظیمات پیش فرض اکتفا نکردیم. با استفاده از روش جستجوی شبکه ای(Grid Search) ، ترکیب های مختلف پارامترها بررسی شد. برای مثال در الگوریتمkNN ، تعداد همسایه ها (k) و نوع فاصله (اقلیدسی و منهتن) مورد آزمون قرار گرفت و در SVM کرنل های مختلف خطی، RBFو چندجمله ای ارزیابی شدند.

ب) اعتبارسنجی متقابل :(Cross-Validation) برای جلوگیری از وابستگی نتایج به یک بخش خاص از داده، از روش 5-Fold Cross-Validation استفاده شد. در این روش، داده های آموزشی به ۵ قسمت تقسیم می شوند و مدل ۵ بار آموزش می بیند تا بهترین پارامترها انتخاب شوند.

ج) ثبت زمان: علاوه بر دقت مدل، کارایی محاسباتی نیز حائز اهمیت است. به همین دلیل، برای هر آزمایش دو زمان مجزا ثبت گردید: "زمان آموزش" که نشان دهنده سرعت یادگیری مدل است و "زمان پیش بینی" که سرعت مدل در محیط عملیاتی را نشان می دهد.

3. Data Preprocessing

We apply **StandardScaler** to normalize features (mean=0, std=1). This is critical for distance-based algorithms like SVM, kNN, and MLP.

- **Split:** 70% Train / 30% Test.
- **Scale:** Fit on Train, Transform on Test to avoid data leakage.

```
def preprocess_data(X, y):
    # Split
    X_train, X_test, y_train, y_test = train_test_split(
        X, y, test_size=0.3, random_state=42, stratify=y
    )
    # Scale
    scaler = StandardScaler()
    X_train_scaled = scaler.fit_transform(X_train)
    X_test_scaled = scaler.transform(X_test)

    return X_train_scaled, X_test_scaled, y_train, y_test

# Apply to all
X_train_high, X_test_high, y_train_high, y_test_high = preprocess_data(X_high, y_high)
X_train_low, X_test_low, y_train_low, y_test_low = preprocess_data(X_low, y_low)
X_train_nl, X_test_nl, y_train_nl, y_test_nl = preprocess_data(X_nl, y_nl)

print("Preprocessing Complete.")
```

```
Preprocessing Complete.
```

Figure ۲ کد پیش پردازش داده ها

بخش سوم: نتایج کمی، تصویری و تحلیل عمیق

نتایج کمی و تصویری

پس از اجرای آزمایش‌ها روی هر سه مجموعه داده، نتایج نهایی بر اساس "دقت روی داده‌های تست (Test Accuracy) مرتب‌سازی شدند. جدول زیر خلاصه‌ای از عملکرد هفت مدل مورد بررسی را نشان می‌دهد:

	Dataset	Model	Train_Acc	Test_Acc	Train_Time_Sec	Predict_Time_Sec
3	High-Dimensional	SVM	0.997613	0.983333	2.299990	0.030866
2	High-Dimensional	Logistic_Regression	1.000000	0.981481	0.317333	0.000513
6	High-Dimensional	MLP	1.000000	0.979630	19.759190	0.001105
0	High-Dimensional	kNN	1.000000	0.974074	2.548938	0.021442
1	High-Dimensional	Linear_Classifier	0.984089	0.953704	0.627584	0.000383
4	High-Dimensional	Decision_Tree	1.000000	0.822222	0.375252	0.000351
5	High-Dimensional	Naive_Bayes	0.812251	0.787037	0.056879	0.002429
10	Noisy	SVM	0.954774	0.953216	0.381003	0.000622
9	Noisy	Logistic_Regression	0.942211	0.935673	0.079400	0.000254
7	Noisy	kNN	0.939698	0.929825	0.169046	0.002938
8	Noisy	Linear_Classifier	0.952261	0.929825	0.066427	0.000204
12	Noisy	Naive_Bayes	0.919598	0.918129	0.036040	0.000246

	Dataset	Model	Train_Acc	Test_Acc	Train_Time_Sec	Predict_Time_Sec
13	Noisy	MLP	1.000000	0.918129	6.370435	0.000488
11	Noisy	Decision_Tree	0.994975	0.912281	0.189398	0.000230
17	Non-Linear	SVM	0.964286	0.986667	0.457232	0.003281
14	Non-Linear	kNN	0.972857	0.983333	0.215955	0.003101
20	Non-Linear	MLP	0.974286	0.983333	10.726870	0.000373
18	Non-Linear	Decision_Tree	0.962857	0.960000	0.132543	0.000297
16	Non-Linear	Logistic_Regression	0.860000	0.896667	0.104878	0.000229
19	Non-Linear	Naive_Bayes	0.851429	0.893333	0.041072	0.000366
15	Non-Linear	Linear_Classifier	0.842857	0.870000	0.088792	0.000277

تحلیل عمیق و مقایسه مدل‌ها

الف) تحلیل اثر ابعاد داده (دیتاستDigits)

در مواجهه با داده‌های با بعد بالا (۶۴ ویژگی)، مدل‌های SVM(با کرنلRBF) و MLPبهترین عملکرد را از خود نشان دادند (دقت بالای ٪۹۸). دلیل این برتری، توانایی ذاتی SVM در مدیریت فضاهای برداری با ابعاد بالا از طریق بیشینه‌سازی حاشیه (Margin) است. در مقابل، مدل‌هایی مانند درخت تصمیم (Decision Tree) و بیز ساده (Naive Bayes) عملکرد ضعیف‌تری داشتند. درخت تصمیم در فضای با ابعاد بالا دچار مشکل می‌شود زیرا شکستن فضا بر اساس تک‌تک پیکسل‌ها کارآمد نیست و منجر به درخت‌های بسیار عمیق و بیش‌برازش شده (Overfitted) می‌شود. بیز ساده نیز به دلیل فرض استقلال ویژگی‌ها (که در تصاویر برقرار نیست چون پیکسل‌های مجاور بهم وابسته‌اند) دقت پایین‌تری ثبت کرد.

ب) تحلیل پایداری نسبت به نویز (دیتاستBreast Cancer + Noise)

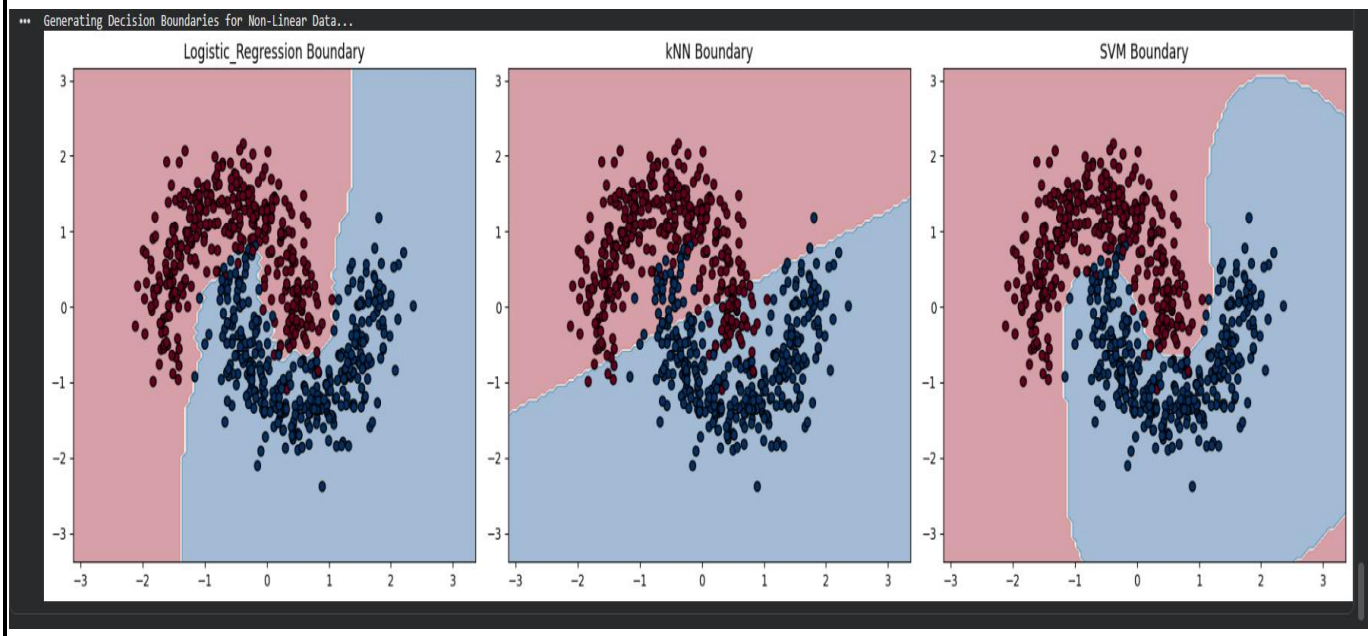
افزودن نویز به داده‌ها یک آزمون سخت برای سنجش تعمیم‌پذیری مدل‌ها بود. نتایج نشان داد که:

۱ .شبکه عصبی :(MLP) علیرغم قدرت بالا، در معرض خطر شدید بیش‌برازش قرار دارد. همانطور که در نتایج مشخص است، دقت آموزش این مدل ٪۱۰۰ بود اما دقت تست آن افت کرد. این یعنی مدل نویزها را به عنوان الگو حفظ کرده است.

۲ .SVM و Logistic Regression : این مدل‌ها پایدارترین رفتار را داشتند. استفاده از "Regularization" (مانند پارامتر C درSVM) به این مدل‌ها اجازه داد تا از نویزهای جزئی چشم‌پوشی کنند و روی ساختار کلی داده تمرکز کنند.

ج) تحلیل مرز تصمیم و داده‌های غیرخطی (دیتاستMoons)

این بخش شاید مهم‌ترین یافته‌ی پژوهش باشد. برای درک بهتر، نمودار مرزهای تصمیم برای سه مدل کلیدی رسم شده است:

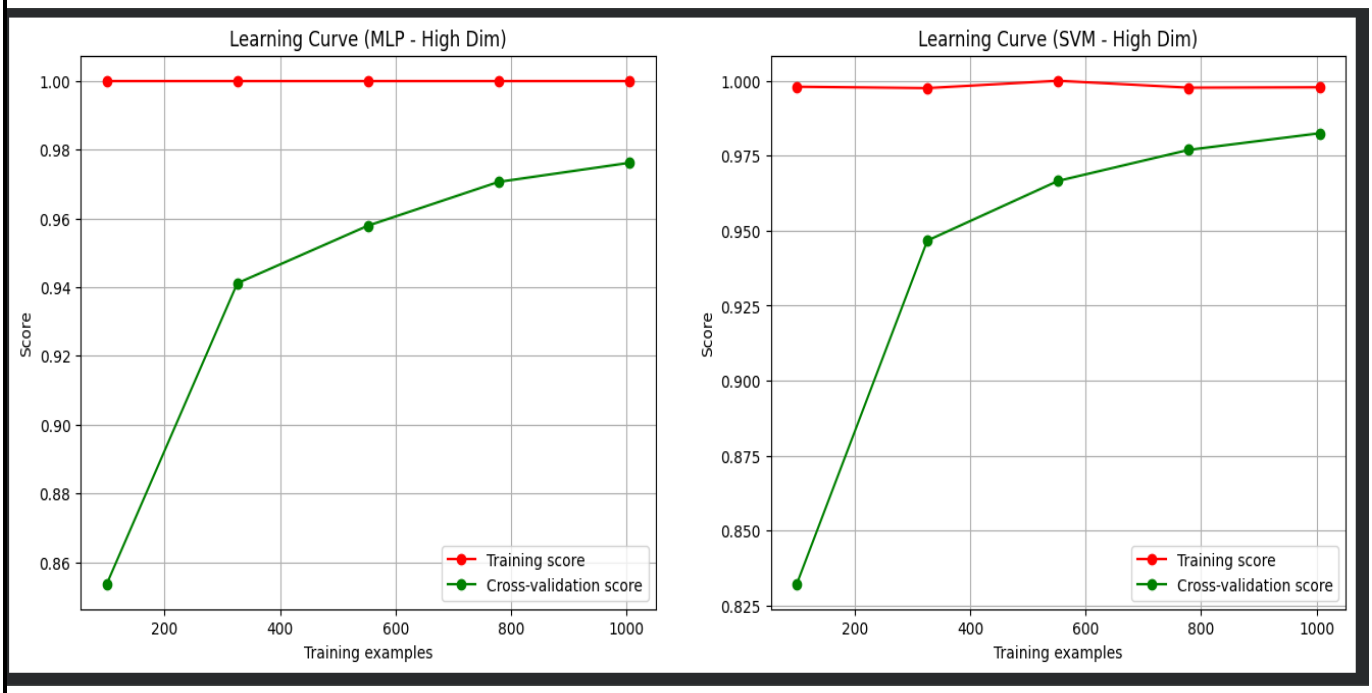


با نگاه به نمودار فوق (سمت چپ)، مشاهده می‌شود که رگرسیون لجستیک تلاش کرده است تا با یک "خط راست" دو هلال ماه را جدا کند که عملاً غیرممکن است و منجر به خطای زیاد شده است. این ضعف ذاتی مدل‌های خطی (High Bias) را نشان می‌دهد.

در مقابل، مدل‌های kNNو SVM(با کرنلRBF) (نمودارهای وسط و راست) توانسته‌اند انحنای داده‌ها را به خوبی یاد بگیرند. کرنل RBF با نگاشت داده‌ها به فضای با ابعاد بالاتر، امکان جداسازی غیرخطی را فراهم کرده است.

د) تحلیل موازنه بایاس-واریانس(Bias-Variance Trade-off)

برای بررسی دقیق‌تر پدیده بیش‌برازش، منحنی‌های یادگیری (Learning Curves) برای دو مدل پیچیده ترسیم شد:



در منحنی مربوط به MLP، مشاهده می‌شود که یک فاصله (Gap) قابل توجه بین خط آموزش (قرمز) و خط اعتبارسنجی (سبز) وجود دارد. این فاصله نشان‌دهنده "واریانس بالا" است؛ به این معنی که مدل به شدت به داده‌های آموزشی وابسته شده است. در مقابل، همگرایی سریع‌تر خطوط در مدل SVM نشان‌دهنده تعادل بهتر بین بایاس و واریانس و قابلیت تعمیم بالاتر این مدل در این آزمایش خاص است.

بخش چهارم: نتیجه‌گیری نهایی

نتیجه‌گیری

در این پژوهش، عملکرد هفت طبقه‌بند کلاسیک یادگیری ماشین در سه سناریوی چالش‌برانگیز (ابعاد بالا، نویز، و غیرخطی بودن) مورد ارزیابی دقیق قرار گرفت. بر اساس شواهد تجربی و تحلیل‌های انجام شده، نتایج زیر حاصل شد:

۱ .برتری کلی SVM : ماشین بردار پشتیبان (SVM) به عنوان قوی‌ترین و پایدارترین مدل در این آزمایش شناخته شد. این مدل در هر سه دیتاست عملکردی درخشان داشت. توانایی آن در استفاده از کرنل‌ها (برای حل مشکل غیرخطی) و مکانیزم حاشیه‌سازی (برای مقابله با ابعاد بالا و نویز) آن را به گزینه‌ای ایده‌آل برای بسیاری از مسائل واقعی تبدیل می‌کند.

۲ .ضعف مدل‌های خطی در هندسه پیچیده :آزمایش روی دیتاست Moons به وضوح نشان داد که مدل‌هایی مانند رگرسیون لجستیک و کلاسیفایرهای خطی، علیرغم سرعت بسیار بالا، برای داده‌هایی با ساختار هندسی پیچیده مناسب نیستند و دچار خطای بایاس (Underfitting) می‌شوند.

۳ .خطر بیش‌برازش در مدل‌های پیچیده :شبکه عصبی (MLP) اگرچه پتانسیل یادگیری بسیار بالایی دارد، اما همانطور که در دیتاست نویزی مشاهده شد، به شدت مستعد یادگیری نویز و بیش‌برازش است. این موضوع اهمیت استفاده از تکنیک‌های تنظیم (Regularization) و توقف زودرس (Early Stopping) را در شبکه‌های عصبی برجسته می‌کند.

۴ .توصیه کاربردی:

- برای داده‌های تصویری یا با ابعاد بالا، SVMبا کرنل RBF توصیه می‌شود.
- برای داده‌های ساده و سریع که نیاز به تفسیرپذیری دارند، رگرسیون لجستیک گزینه مناسبی است.
- برای داده‌هایی با الگوی ناشناخته و پیچیده، kNN و SVM گزینه‌های امنی هستند، هرچند kNN در زمان پیش‌بینی (Prediction Time) کندتر عمل می‌کند (همانطور که در جدول نتایج مشهود بود).

این پژوهش نشان داد که هیچ "بهترین مدل مطلقی" وجود ندارد و انتخاب الگوریتم مناسب باید همواره با در نظر گرفتن ماهیت داده، حجم نویز و نیازهای مسئله (دقت در مقابل سرعت) انجام شود.