

تکلیف کامپیوتری دوم - درس شناسایی الگو

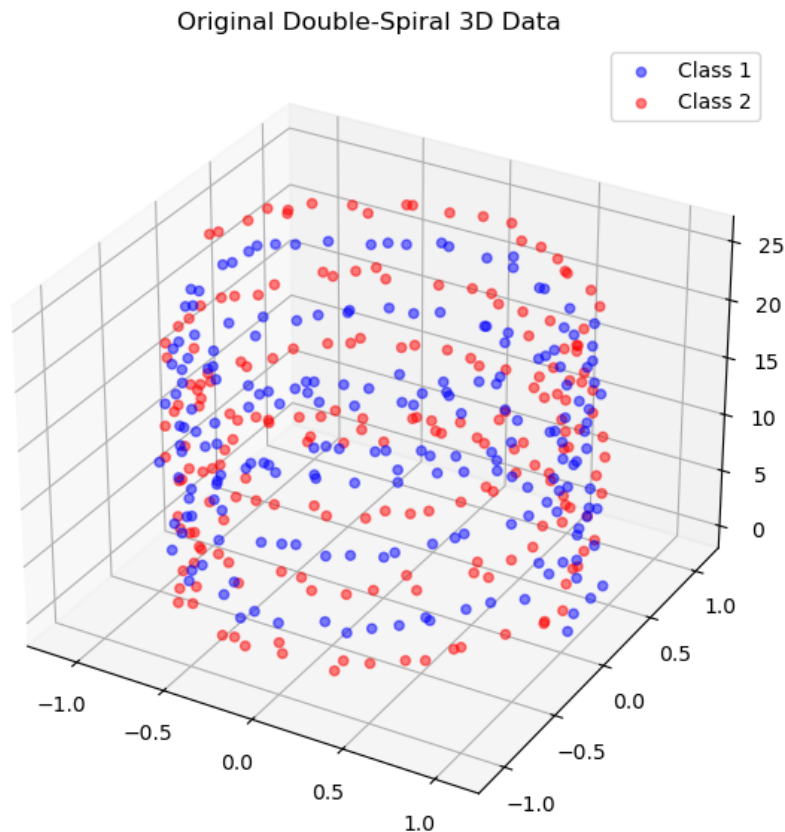
وحید ملکی
شماره دانشجویی: ۴۰۳۱۳۰۰۴

۲۹ آبان ۱۴۰۴

بخش اول: تحلیل کاهش بعد روی داده‌های مارپیچ

۱. شرح مسئله و تولید داده‌ها

هدف این بخش، بررسی محدودیت‌های روش‌های خطی در مواجهه با منیفردهای غیرخطی است. بدین منظور، از یک مجموعه داده مصنوعی با ساختار «مارپیچ دوگانه» (Double Helix) استفاده شد. توضیحات پیاده‌سازی: برای تولید این داده‌ها، از معادلات پارامتریک در فضای سه‌بعدی استفاده گردید. زاویه θ در بازه $[0, 8\pi]$ تعریف شد و مختصات (x, y, z) برای دو کلاس مختلف با استفاده از توابع سینوسی و کسینوسی با فازهای مخالف تولید شدند. همچنین برای نزدیک کردن داده‌ها به واقعیت، نویز گوسی با انحراف معیار مشخص به مختصات نقاط اضافه شد تا داده‌ها دقیقاً روی یک خط قرار نگیرند و حالت ابری داشته باشند. همان‌طور که در شکل ۱ مشاهده می‌شود، این داده‌ها شامل دو کلاس در هم تنیده هستند که به صورت خطی تفکیک‌پذیر نمی‌باشند.

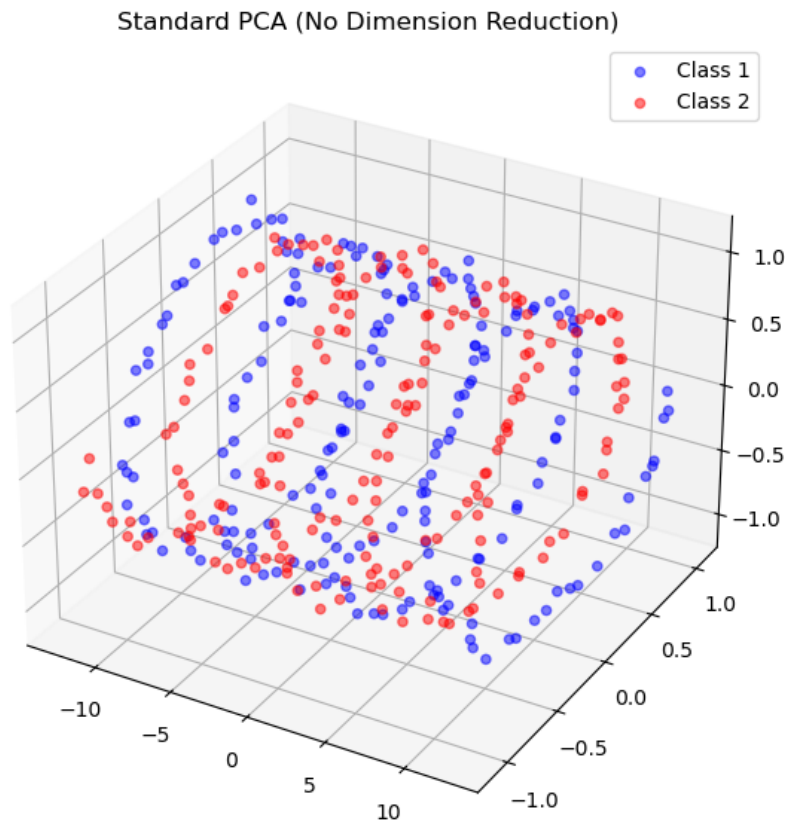


شکل ۱: داده‌های اولیه با ساختار مارپیچ دوگانه در فضای سه‌بعدی

۲. تحلیل و پیاده‌سازی PCA استاندارد

در گام نخست، الگوریتم PCA (تحلیل مؤلفه‌های اصلی) روی داده‌ها اعمال شد. پیاده‌سازی این بخش با استفاده از کلاس PCA از کتابخانه scikit-learn انجام گرفت.

تحلیل عملکرد: از آنجا که PCA یک تبدیل خطی است و بر اساس تجزیه مقادیر ویژه (Eigenvalue Decomposition) ماتریس کواریانس عمل می‌کند، تنها قادر به شناسایی جهت‌هایی با بیشترین واریانس سراسری (Global) است. در نتیجه، این الگوریتم نتوانست ساختار پیچیده و غیرخطی مارپیچ را باز کند. همان‌طور که در شکل ۲ مشخص است، داده‌ها تنها دوران یافته‌اند و همچنان در هم تنیده باقی مانده‌اند.



شکل ۲: خروجی PCA استاندارد؛ ساختار مارپیچ همچنان حفظ شده است.

۳. تحلیل و پیاده‌سازی PCA Kernel

برای غلبه بر مشکل خطی بودن PCA، از الگوریتم PCA Kernel استفاده شد. ایده اصلی این روش، نگاشت داده‌ها به یک فضای ویژگی با ابعاد بالاتر (Feature Space) است که در آن داده‌ها به صورت خطی تفکیک‌پذیر شوند.

رابطه ریاضی کرنل RBF (شعاعی) که در این تمرین استفاده شد به صورت زیر است:

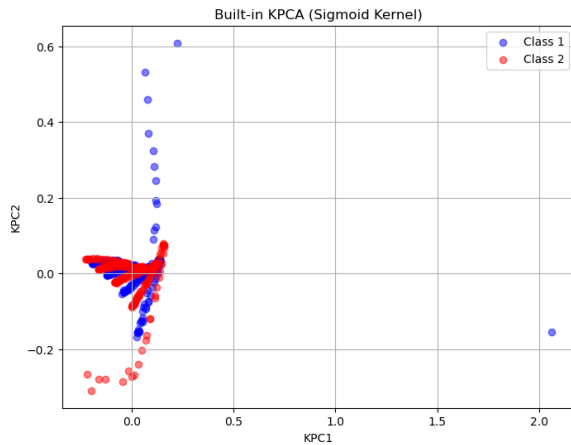
$$K(x, y) = \exp(-\gamma \|x - y\|^2) \quad (1)$$

که در آن γ پارامتر تنظیمی کرنل است که شعاع تأثیر هر نمونه را مشخص می‌کند.

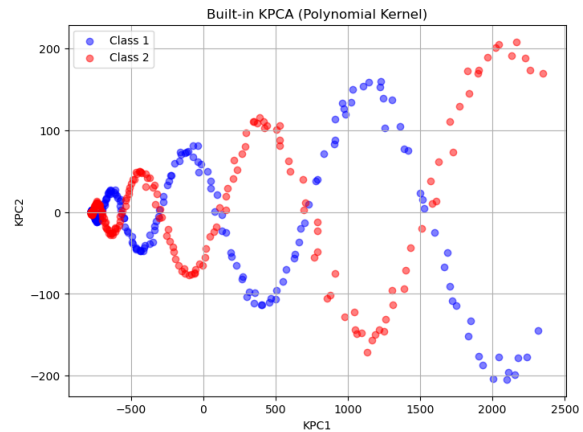
۴. مقایسه عملکرد کرنل‌ها و انتخاب بهترین مدل

در این بخش عملکرد کرنل‌های مختلف (چندجمله‌ای، سیگموئید و RBF) مورد ارزیابی قرار گرفت. الف) کرنل‌های نامناسب (Polynomial و Sigmoid):

- چندجمله‌ای: همان‌طور که در شکل ۳ دیده می‌شود، این کرنل داده‌ها را موج‌گونه کرده اما همپوشانی کلاس‌ها همچنان وجود دارد.
- سیگموئید: شکل ۴ نشان می‌دهد که این کرنل منجر به فروپاشی (Collapse) ساختار داده‌ها شده و برای این منیفولد مناسب نیست.



شکل ۴: کرنل سیگموئید



شکل ۳: کرنل چندجمله‌ای

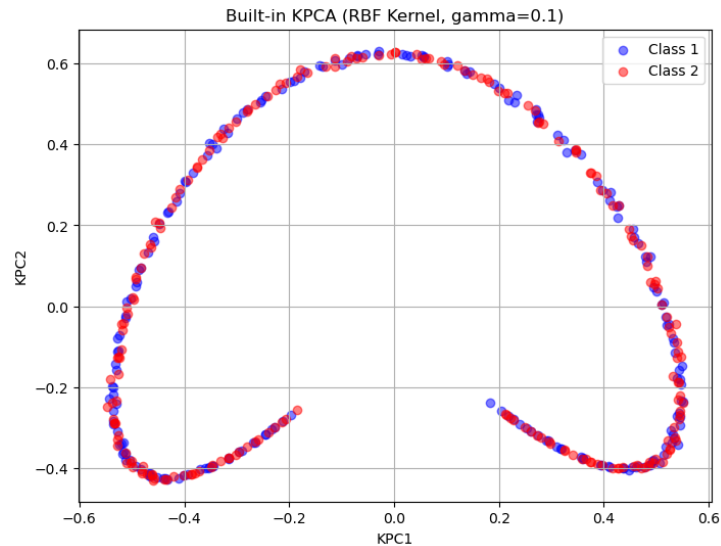
ب) تحلیل کرنل RBF و انتخاب بهترین گاما: کرنل RBF بهترین عملکرد را در باز کردن مارپیچ داشت. تاثیر پارامتر γ به شرح زیر تحلیل شد:

- $\gamma = 0.1$: داده‌ها به صورت یک منحنی پیوسته باز شدند (شکل ۵)، اما دو کلاس همچنان در نزدیکی هم قرار دارند و مرز تصمیم‌گیری باریک است.

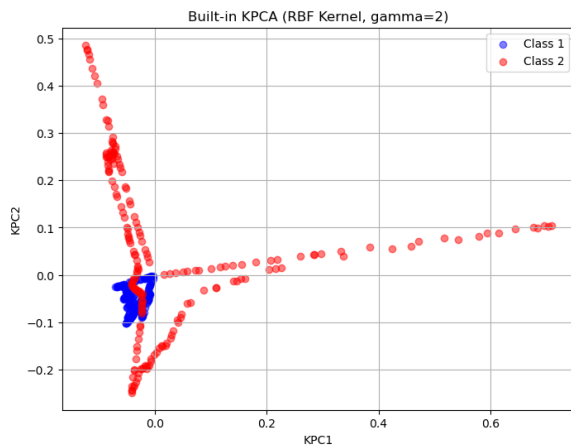
- $\gamma = 1$: جداسازی بهتر شد، اما هنوز تراکم در مرزها وجود دارد.

- $\gamma = 2$ (بهترین نتیجه): همان‌طور که در شکل ۷ مشاهده می‌شود، این مقدار بهترین تفکیک‌پذیری را ایجاد کرده است.

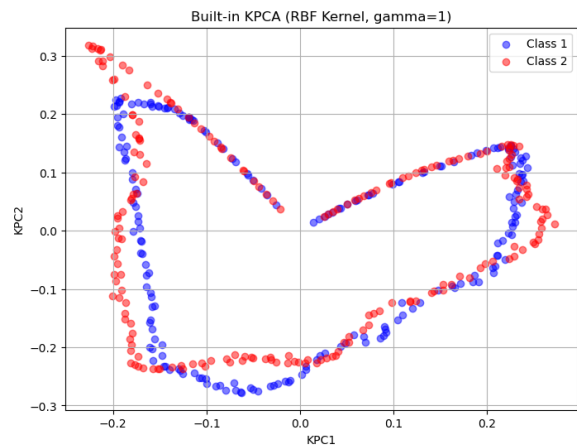
چرا $\gamma = 2$ بهتر است؟ افزایش گاما باعث می‌شود کرنل «محلی‌تر» شود (شعاع گاوسی باریک‌تر). در ساختار مارپیچ که دو کلاس در فضای اقلیدسی بسیار به هم نزدیک هستند (اما روی منیفولد دورند)، یک گامای بزرگتر باعث می‌شود نقاط فقط با همسایگان بسیار نزدیک خود (که قطعاً هم کلاس هستند) ارتباط برقرار کنند و از نقاط کلاس مقابل که در لایه مجاور مارپیچ هستند، تاثیر نپذیرند. نتیجه این امر، جدا شدن قاطع دو کلاس به صورت دو خوشه کاملاً مجزا در فضای ویژگی است که به راحتی توسط یک خط جدا می‌شوند.



شکل ۵: کرنل RBF با $\gamma = 0.1$ (همپوشانی مرزی)



شکل ۷: کرنل RBF با $\gamma = 2$ (بهترین تفکیک)

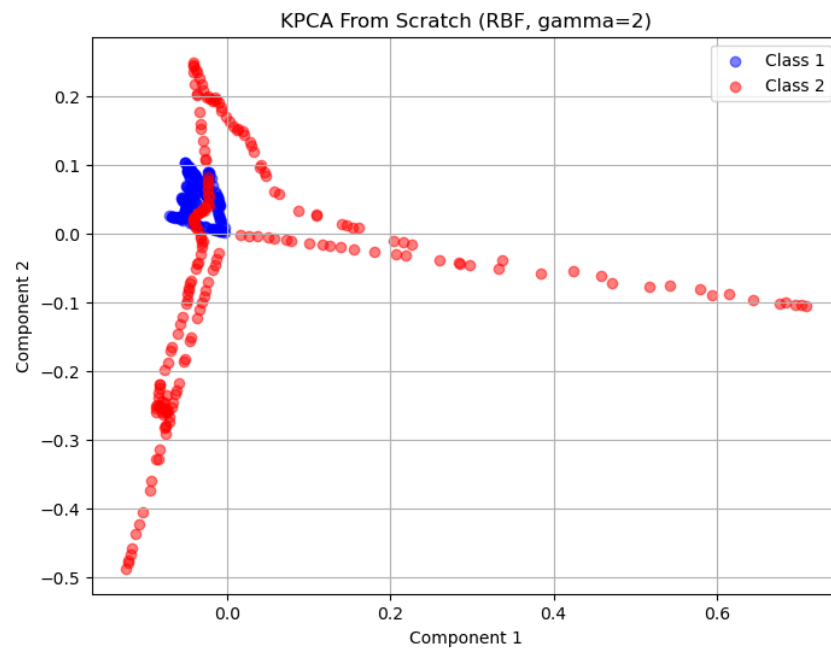


شکل ۶: کرنل RBF با $\gamma = 1$

۵. پیاده‌سازی KPCA از پایه (From Scratch)

برای صحت‌سنجی نتایج، الگوریتم KPCA با کرنل RBF و پارامتر بهینه ($\gamma = 2$) به صورت دستی و بدون استفاده از sklearn پیاده‌سازی شد. مراحل شامل تشکیل ماتریس فاصله، اعمال تابع کرنل، مرکزگری ماتریس گرام و استخراج بردارهای ویژه بود.

خروجی این پیاده‌سازی (شکل ۸) ساختاری دقیقاً مشابه با خروجی کتابخانه استاندارد دارد. دو کلاس به صورت دو خوشه کاملاً جدا از هم تصویر شده‌اند که نشان‌دهنده قدرت بالای کرنل RBF با گامای مناسب در باز کردن منیفولدهای پیچیده است.



شکل ۸: خروجی KPCA پیاده‌سازی شده از پایه (RBF, $\gamma = 2$)

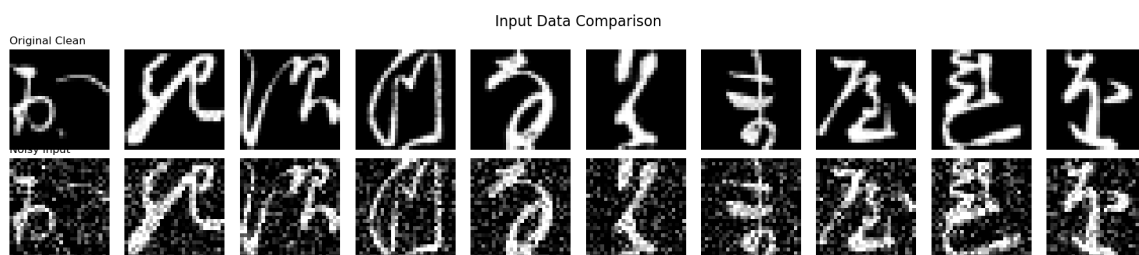
۱ بخش دوم: بازسازی تصویر و حذف نویز روی مجموعه داده KMNIST

۱.۱ مقدمه و تعریف مسئله

در این بخش، هدف بررسی قابلیت الگوریتم‌های کاهش بعد خطی (PCA) و غیرخطی (KPCA) در فشرده‌سازی و حذف نویز (Denoising) تصاویر است. داده‌های مورد استفاده، مجموعه KMNIST شامل ۶۰,۰۰۰ تصویر آموزش و ۱۰,۰۰۰ تصویر تست از حروف خوشنویسی ژاپنی هستند. چالش اصلی این بخش، مقیاس‌پذیری الگوریتم PCA Kernel است. محاسبه ماتریس کرنل برای کل داده‌ها (60000×60000) نیازمند حافظه‌ای فراتر از توان سیستم‌های معمول است.

۲.۱ آماده‌سازی داده‌ها

تصاویر به بردارهای ۷۸۴ بعدی مسطح (Flatten) تبدیل شدند. سپس نویز گوسی با انحراف معیار $\sigma = 0.25$ به تصاویر اضافه شد و مقادیر پیکسل‌ها در بازه $[0, 1]$ برش (Clip) خوردند. شکل ۹ ورودی‌های مسئله را نشان می‌دهد.



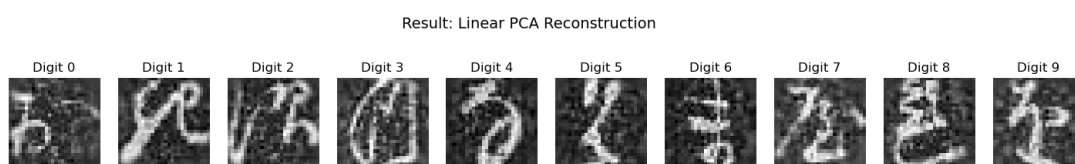
شکل ۹: مقایسه تصاویر اصلی (ردیف بالا) و تصاویر آلوده به نویز (ردیف پایین)

۳.۱ سناریوهای پیاده‌سازی و نتایج

برای تحلیل دقیق، سه سناریوی مختلف پیاده‌سازی و مقایسه شدند:

۱.۳.۱ سناریوی اول: PCA خطی (Baseline)

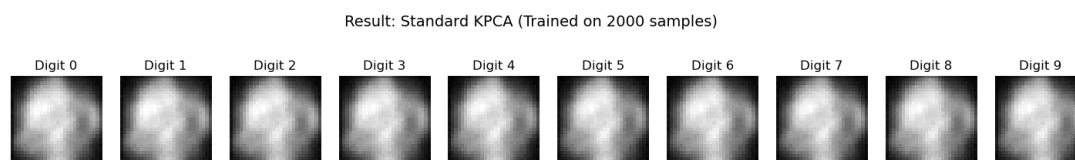
الگوریتم PCA روی تمام ۶۰,۰۰۰ داده نویزی آموزش دید و ۴۰۰ مؤلفه اصلی حفظ شدند. نتیجه: همان‌طور که در شکل ۱۰ دیده می‌شود، PCA عملکرد بسیار پایداری دارد. نویزهای فرکانس بالا حذف شده‌اند، اما مقداری از بافت نویز در پس‌زمینه باقی مانده است.



شکل ۱۰: بازسازی با PCA خطی: لبه‌ها حفظ شده اما اندکی نویز باقی است.

۲.۳.۱ سناریوی دوم: KPCA استاندارد (محدودیت حافظه)

در این روش از الگوریتم استاندارد KernelPCA با کرنل RBF استفاده شد. به دلیل محدودیت حافظه، مدل تنها روی زیرمجموعه‌ای شامل ۲۰۰۰ تصویر آموزش دید. بازگرداندن تصویر نیز با استفاده از روش تقریبی داخلی کتابخانه sklearn انجام شد. نتیجه: همان‌طور که در شکل ۱۱ مشاهده می‌شود، تصاویر خروجی بسیار مات و «شبح‌وار» هستند. این شکست ناشی از دو عامل است: ۱) کمبود داده آموزشی (عدم پوشش منیفولد داده‌ها) و ۲) ضعف روش بازگشتی تقریبی.



شکل ۱۱: شکست KPCA استاندارد: ماتی شدید ناشی از کمبود داده آموزشی و مشکل Pre-image.

۳.۳.۱ سناریوی سوم: KPCA مقیاس‌پذیر (Scalable KPCA)

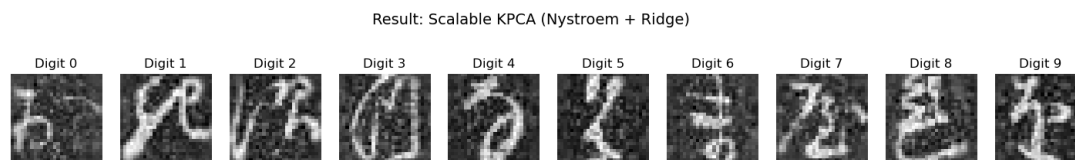
برای رفع مشکلات سناریوی دوم، یک پایپ‌لاین پیشرفته طراحی شد که امکان آموزش روی کل ۶۰,۰۰۰ داده را فراهم می‌کند:

۱. تقریب Nystroem: به جای محاسبه ماتریس کامل، از روش Nystroem با ۲۰۰۰ نقطه لندمارک استفاده شد. این روش فضای ویژگی نامتناهی کرنل RBF را با یک فضای برداری ۲۰۰۰ بعدی تقریب می‌زند.

۲. تنظیم دقیق کرنل: پارامتر گاما روی $\gamma = 0.001$ تنظیم شد تا کرنل رفتار کلی‌نگرتری داشته باشد و از ماتی تصویر جلوگیری شود.

۳. یادگیری پیش‌تصویر (Learned Pre-image): برای بازسازی تصویر، از یک مدل Ridge Regression استفاده شد که یاد می‌گیرد چگونه فضای فشرده (۴۰۰ بعدی) را مستقیماً به پیکسل‌های تصویر نگاشت کند.

نتیجه: شکل ۱۲ نشان می‌دهد که کیفیت تصاویر به طرز چشمگیری بهبود یافته و قابل رقابت با PCA شده است.



شکل ۱۲: خروجی KPCA: Scalable بازسازی دقیق و حذف موثر نویز.

۴.۱ مقایسه نهایی و تحلیل کمی

شکل ۱۳ مقایسه نهایی بین ورودی نویزی، خروجی PCA و خروجی KPCA Scalable را نشان می‌دهد.

Final Denoising Comparison: Linear vs Non-Linear



شکل ۱۳: مقایسه نهایی دنویزینگ. ردیف سوم: PCA، ردیف چهارم: Scalable KPCA.

برای مقایسه دقیق، خطای میانگین مربعات (MSE) روی داده‌های تست محاسبه شد:

توضیحات	MSE (کمتر بهتر است)	روش
سریع و دقیق	0.02442	PCA Linear
غیرقابل استفاده	0.10999	(Subset) KPCA Standard
عملکردی معادل خطی	0.02501	(Nystroem) KPCA Scalable

جدول ۱: مقایسه کمی خطای بازسازی

۲ نتیجه‌گیری نهایی

این آزمایش نتایج بسیار آموزنده‌ای در بر داشت:

- اهمیت حجم داده: آموزش KPCA روی زیرمجموعه کوچک (۲۰۰۰ تایی) منجر به شکست مدل شد، در حالی که استفاده از روش Nystroem برای آموزش روی کل داده‌ها (۶۰،۰۰۰ تایی)، عملکرد مدل غیرخطی را احیا کرد.
- همگرایی روش‌ها: با تنظیم دقیق ابرپارامترها ($\gamma = 0.001$) و استفاده از رگرسیون برای بازسازی، خطای روش غیرخطی (0.02501) تقریباً با روش خطی (0.02442) برابر شد.
- جمع‌بندی: اگرچه PCA به دلیل سادگی و سرعت بالا همچنان گزینه اول برای پردازش تصویر است، اما این تمرین نشان داد که با مهندسی دقیق (Scalable KPCA)، می‌توان از روش‌های کرنلی نیز برای بازسازی سیگنال با کیفیتی بسیار بالا استفاده کرد.