

روش آنالیز جداکننده چندخطی با استفاده از ملاک تفاضلی

امین رستگار، منصور رزقی*؛ دانشگاه تربیت مدرس
دریافت ۹۴/۰۸/۱۷ پذیرش ۹۶/۰۳/۰۶

چکیده

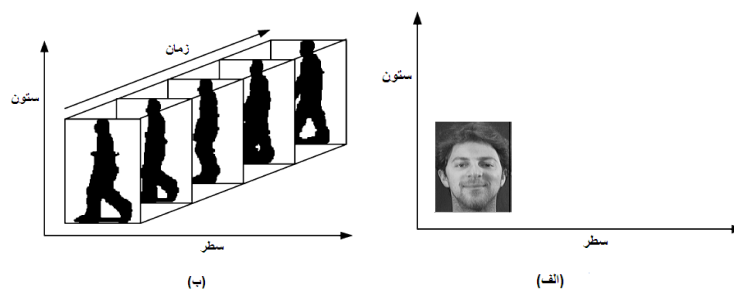
روش‌های کاهش ابعاد نقش بسیار مهمی در بهبود کیفیت روش‌های یادگیری ماشین و تشخیص الگو ایفا می‌کنند. در این روش‌ها فرض می‌شود که داده‌ها به صورت یک بردار هستند. بنا بر این داده‌های غیربرداری مانند تصاویر که به داده‌های با بعد بالا یا تانسور معروف هستند باید به بردار تبدیل شده و سپس فرایند کاهش ابعاد انجام گیرد. این عمل علاوه بر هزینه محاسباتی ممکن است باعث از بین رفتن برخی روابط محلی بین داده‌ها نیز شود. برای حل این مشکل در سال‌های اخیر روش‌های کاهش ابعاد چند خطی ارائه شده‌اند که سعی دارند فرایند کاهش ابعاد را بدون تغییر شکل داده‌ها انجام دهند. یکی از رویکردهای چندخطی مهم در این زمینه توسعه جداکننده خطی فیشر به روش‌های چندخطی است. مشکل مهم این روش‌ها نیاز آن‌ها به حل تعداد زیادی مسئله مقدار ویژه توسعه یافته در تکرارهای زیاد است. در این مقاله روشی برای حل این زیر مسئله‌ها ارائه کرده‌ایم. نتایج پیاده‌سازی کیفیت روش ارائه شده در جدا کردن کلاس‌ها را در مقایسه با روش اصلی جداکننده چندخطی به‌ویژه برای داده‌ها با نمونه کم نشان می‌دهند.

واژه‌های کلیدی: کاهش بعد، روش آنالیز مؤلفه‌های اصلی، آنالیز جداکننده خطی، روش چندخطی، آنالیز جدا کننده چندخطی

مقدمه

امروزه داده‌های زیادی تولید می‌شوند که با تجزیه و تحلیل آن‌ها اطلاعات مفیدی به دست می‌آید. اگر داده‌ها دارای حجم زیادی باشند، آنالیز و پردازش آن‌ها به سختی انجام می‌گیرد. به همین دلیل روش‌هایی وجود دارند که قبل از پردازش داده، حجم داده یا ویژگی‌های آن را کاهش می‌دهند و داده را به فضایی با ابعاد کوچک‌تر تصویر می‌کنند. به این روش‌ها که باعث سهولت تجزیه و تحلیل داده کاهش یافته و حذف اطلاعات تکراری یا غیرمرتبط می‌شود، اصطلاحاً روش‌های کاهش بعد می‌گویند [۱]. از جمله متداول‌ترین روش‌های کاهش بعد، روش‌های تحلیل مؤلفه‌های اصلی^۱ PCA، روش جداکننده خطی^۲ LDA و تجزیه نامنفی ماتریسی را می‌توان نام برد [۲]، [۳]، [۴]. همگی این روش‌های هر داده را به صورت یک بردار فرض می‌کنند. ولی در عمل ممکن است داده‌ها دارای ساختاری غیربرداری باشند. داده‌هایی مانند تصاویر و ویدئو دارای ساختار دو یا چندوجهی هستند و به تانسور معروف هستند [۵]. برای کاهش بعد تانسورهای مرتبه بالا با استفاده از روش‌های خطی متداول، باید ساختار داده تغییر کرده و به بردار تبدیل شود. در فرآیند برداری کردن، داده‌های تانسوری از فضای $\mathbb{R}^{m_1 \times m_2 \times \dots \times m_N}$ تبدیل به بردارهایی در فضای $\mathbb{R}^{(m_1 \times m_2 \times \dots \times m_N)}$ می‌شوند. سپس با استفاده از روش‌های کاهش بعد خطی، به فضای برداری \mathbb{R}^n که $n < (m_1 \times m_2 \times \dots \times m_N)$ تصویر می‌گردند. استفاده از روش‌های خطی برای کاهش بعد، داده‌های تانسوری دارای معایبی است که آن را در عمل غیر کاربردی می‌سازند. این معایب را بدین صورت می‌توان خلاصه کرد:

- یکی از مشکلات به کار بردن روش‌های خطی در داده‌های تانسوری، پیچیدگی محاسباتی زیاد آن است. برای مثال هر داده تانسور N بعدی عضو $\mathbb{R}^{m_1 \times m_2 \times \dots \times m_N}$ به یک بردار $m_1 \times m_2 \times \dots \times m_N$ تایی تبدیل می‌شود. این کار برای روش‌هایی مانند PCA و LDA که نیازمند حل مسئله مقدار ویژه یک ماتریس به ابعاد $(m_1 \times m_2 \times \dots \times m_N) \times (m_1 \times m_2 \times \dots \times m_N)$ برای این داده‌ها هستند بسیار از لحاظ محاسباتی پرهزینه است [۶].
- از طرف دیگر به دلیل زیاد بودن بعد (ویژگی‌های) هر داده با مسئله معضل ابعاد روبه‌رو هستیم. در مسئله معضل ابعاد با افزایش ویژگی‌های داده، تعداد نمونه‌های لازم برای داشتن یک روش یادگیری قابل اعتماد به صورت نمایی افزایش پیدا می‌کند [۶]، [۷]، [۸]، [۹]. در صورت کم بودن داده‌ها معضل ابعاد برای هر روش به صورت خاصی رخ می‌دهد. مثلاً برای روش LDA کم بودن داده‌های یادگیری منجر به بدحالی ماتریس پراکنده و بروز خطا در حل مسئله می‌شود.
- ایراد سوم استفاده از روش‌های خطی، تغییر ساختار داده‌ها است. زیرا با برداری کردن، جای‌گاه عناصر داده نسبت به هم تغییر می‌کند و پیوستگی و روابط محلی عناصر داده از بین می‌رود. مثلاً در داده‌های موجود در شکل ۱ برداری کردن تصویر الف باعث از بین بردن روابط مکانی پیکسل‌های تصویر می‌شود. هم‌چنین در داده ب که نشان‌دهنده داده‌ای ویدیویی است علاوه بر از بین بردن روابط همسایگی پیکسل‌ها که در تصویر مهم است، روابط زمانی آن‌ها را نیز به هم می‌ریزد.



شکل ۱. الف) تصویر چهره به صورت یک تانسور مرتبه ۲، ب) داده ویدیویی به صورت یک تانسور مرتبه ۳

در سال‌های اخیر برای پوشش ضعف‌های روش‌های خطی، روش‌هایی تحت نام روش‌های چندخطی که به روش‌های تانسوری نیز معروف هستند، برای یادگیری و کاهش ابعاد روی داده‌های تانسوری عرضه شده‌اند [۵]، [۶]، [۷]، [۹]، [۱۰]. در این روش‌ها، ساختار داده تغییر نمی‌کند و کاهش بعد به صورت مستقیم روی داده تانسوری انجام می‌گیرد. داده‌های تانسوری از فضای $\mathbb{R}^{m_1 \times m_2 \times \dots \times m_N}$ به تانسورهای در فضای $\mathbb{R}^{n_1 \times n_2 \times \dots \times n_N}$ تصویر می‌شود که در آن برای هر $1 \leq i \leq N$ رابطه $n_i \ll m_i$ برقرار است. خلاصه مزیت‌های این روش‌ها را می‌توان بدین صورت خلاصه کرد:

- این روش‌ها روابط محلی بین داده‌ها را از بین نمی‌برند. مثلاً در برداری کردن، دو پیکسل مجاور هم ممکن بود از هم دور شده و رابطه محلی آن‌ها در نظر گرفته نشود. اما در روش‌های تانسوری ساختار داده دست‌نخورده باقی می‌ماند.
- در این روش‌ها معمولاً ویژگی‌های داده‌ها در بعدهای مختلف جدای از هم در نظر گرفته شده و تحلیل می‌شوند. بنا براین این امکان ایجاد می‌شود تا در یک بعد که قابلیت کاهش بیش‌تری وجود دارد کاهش بعد بیش‌تری

صورت گیرد. مثلاً در یک تصویر خاکستری که به صورت ماتریس است ممکن است مشاهده کنیم که سطرهای آن بیش تر از ستون‌ها قابلیت کاهش دارند.

- حل مدل‌های چندخطی معمولاً به حل چندین مسئله کوچک‌تر تبدیل می‌شود که این امر از حجم محاسبات و احتمال مواجهه با مسئله معضل ابعاد می‌کاهد.

مهم‌ترین مشکل روش‌های چندخطی فقیر بودن ادبیات حل مسئله در این رویکرد است. به طوری که در سال‌های اخیر تنها روش‌های خیلی بدیهی خطی را توانسته‌اند به حالت چندخطی توسعه دهند. اگرچه این توسعه‌ها نیز موفق بوده‌اند.

با توجه به مزیت‌های بالا، روش‌های چندخطی^۳ در دهه اخیر مورد توجه قرار گرفته و کارهای متنوعی روی آن‌ها انجام گرفته است. به دلیل خاص بودن داده‌های دوبعدی که بیش تر تصویر هستند روش‌های دوخطی زیادی مانند روش تقریب تعمیم یافته رتبه پایین، آنالیز مؤلفه‌های اصلی دوخطی و آنالیز جداکننده دوخطی به وجود آمدند که بدون برداری کردن داده‌های ماتریسی مانند تصاویر دوبعدی، ابعاد سطری و ستونی آن‌ها را به فضایی با ابعاد کم‌تر کاهش می‌دهند [۱۱]، [۱۲]، [۱۳]، [۱۴]، [۱۵].

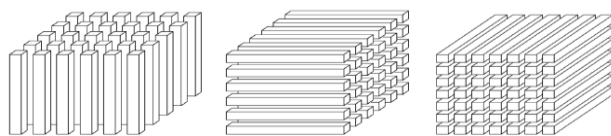
روش‌های چندخطی کاهش بعد مانند آنالیز مؤلفه‌های اصلی چندخطی و آنالیز جداکننده چندخطی نیز از روش‌های چندخطی مهم و پرکاربرد در سال‌های اخیر هستند [۵]، [۷]، [۹]، [۱۶]. این روش‌ها در کاربردهای بسیار زیادی استفاده شده‌اند [۱۸].

حل مدل‌های چندخطی مانند آنالیز جداکننده خطی و مؤلفه‌های اصلی چندخطی از یک فرایند تکراری تشکیل می‌شود که در هر مرحله از آن یک مسئله مقدار ویژه باید حل شود. بنا براین سرعت و دقت الگوریتم به کار گرفته شده در حل مسئله مقدار ویژه مهم است. در این مقاله ما از روشی تفاضلی برای حل این مسائل مقادیر ویژه به صورت فرایندی تکراری با تعداد تکرار کم استفاده می‌کنیم.

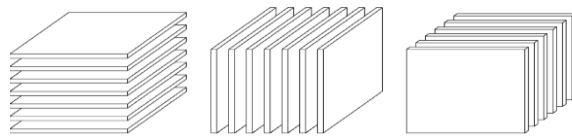
مفاهیم اولیه و مرور ادبیات

تانسور آرایه‌ای چندبعدی است که به تعداد بعدهای آن مرتبه تانسور گفته می‌شود. بردار تانسور از مرتبه یک و ماتریس آن مرتبه دوم است. به آرایه‌های با سه بعد یا بیش تر تانسورهای مرتبه بالا^۴ گفته می‌شود. برای نمایش یک بردار از حروف کوچک مانند a ، ماتریس‌ها از حروف بزرگ مانند A و تانسورهای مرتبه بالاتر از حروف اسکریپتی اوپلر^۵ مانند \mathcal{X} استفاده می‌شود. عنصر i ام برداری مانند a با a_i ، عنصر (i, j) ماتریس A با $a_{i,j}$ و عنصر (i, j, k) تانسور مرتبه سه \mathcal{X} با $\mathcal{X}_{i,j,k}$ نشان داده می‌شود. مفهوم سطر و ستون را در تانسورها گسترش یافته و به فیبر معروف است. اگر یکی از اندیس‌ها متغیر و بقیه اندیس‌ها ثابت در نظر گرفته شود، یک فیبر^۶ به دست می‌آید. برای یک تانسور مرتبه سوم \mathcal{X} می‌توان فیبرهای $\mathcal{X}_{:,j,k}$ ، $\mathcal{X}_{i,:,k}$ ، $\mathcal{X}_{i,j,:}$ را تولید کرد. اسلایس^۷ برشی دوبعدی از تانسور است. با متغیر در نظر گرفتن دو اندیس و ثابت نگه داشتن بقیه اندیس‌ها، ماتریسی به دست می‌آید که به آن اسلایس گفته می‌شود. در شکل ۲ و شکل ۳ فیبرها و اسلایس‌های مختلف یک تانسور مرتبه سه نشان داده شده است.

3. Multilinear
4. higher-order tensor
5. Euler script
6. Fiber
7. Slice

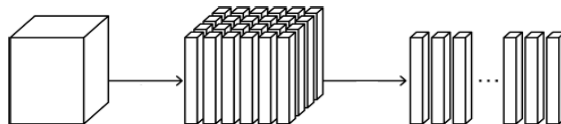


شکل ۲. فیبرهای مختلف یک تانسور مرتبه سه



شکل ۳. اسلایس‌های مختلف یک تانسور مرتبه سه

تانسورهای مرتبه بالا را می‌توان با عمل ماتریسی کردن به صورت یک ماتریس مرتب کرد. به صورت کلی ماتریسی کردن وجه k -تانسور $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times \dots \times I_N}$ از قرار دادن فیبرهای وجه k آن در ستون‌های ماتریس به دست می‌آید و با $\mathcal{X}_{(k)}$ نمایش داده می‌شود. شکل ۴ نحوه ماتریسی کردن یک تانسور از مرتبه ۳ را در وجه اول آن نشان می‌دهد.



شکل ۴. فرایند ماتریسی کردن وجه - یک در تانسور مرتبه سه

تانسورها را می‌توان از یک وجه در ماتریس ضرب کرد. در ضرب وجه n -تانسور $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times \dots \times I_N}$ در ماتریس $U \in \mathbb{R}^{I_n \times I_n}$ که با $\mathcal{X} \times_n U$ نمایش داده می‌شود، تانسوری با اندازه $I_1 \times I_2 \times \dots \times I_{n-1} \times I_{n+1} \times \dots \times I_N$ و هم‌مرتبه با تانسور \mathcal{X} حاصل می‌گردد. عناصر تانسور حاصل ضرب از ضرب هر فیبر وجه n -تانسور \mathcal{X} در ماتریس U با استفاده از رابطه

$$(\mathcal{X} \times_n U)_{i_1, i_2, \dots, i_{n-1}, j, i_{n+1}, \dots, i_N} = \sum_{i_n=1}^{I_n} x_{i_1, i_2, \dots, i_n} u_{j i_n}$$

محاسبه می‌شود. برای ضرب وجه n -تانسور در ماتریس روابط (۱) و (۲) برقرار است:

$$\mathcal{Y} = \mathcal{X} \times_n U \Leftrightarrow Y_{(n)} = U X_{(n)}, \quad (1)$$

$$\mathcal{Y} = \mathcal{X} \times_1 U^{(1)} \times_2 U^{(2)} \dots \times_N U^{(N)} \Leftrightarrow Y_{(n)} = U^{(n)} X_{(n)} (U^{(N)} \otimes U^{(N-1)} \dots \otimes U^{(1)})^T, \quad (2)$$

که در این روابط \otimes نشان‌دهنده ضرب کرونکر ماتریسی است.

نرم تانسور $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times \dots \times I_N}$ مانند نرم فروبینیوس^۸ در ماتریس از جذر مجموع توان دوم تمام عناصر تانسور \mathcal{X} یعنی

$$\|\mathcal{X}\| = \sqrt{\sum_{i_1=1}^{I_1} \sum_{i_2=1}^{I_2} \dots \sum_{i_N=1}^{I_N} x_{i_1, i_2, \dots, i_N}^2}$$

۱. الگوریتم جداکننده چندخطی

روش جداکننده خطی LDA یکی از روش‌های معروف و پرکاربرد کاهش بعدخطی است که نسخه چندخطی آن نیز ارائه شده است. در این بخش این روش و نسخه چندخطی آن را بررسی کرده و در بخش بعد روش خود بهبود نسخه چندخطی آن را ارائه می‌کنیم. روش LDA داده‌های برداری را به فضایی با ابعاد کم‌تر تصویر می‌کند به طوری که داده‌های تصویر شده دارای بیش‌ترین پراکندگی بین‌کلاسی و کم‌ترین پراکندگی درون‌کلاسی هستند. اگر

تعداد کلاس‌ها برابر K بوده و تعداد عناصر کلاس k ام را با N_k نشان دهیم ملاک جداشوندگی کلاس‌ها از هم‌دیگر و جمع‌شدگی درون‌کلاسی که به ملاک فیشر معروف است بدین صورت است:

$$J(W) = \frac{\text{trace}(U^T S_B U)}{\text{trace}(U^T S_W U)}. \quad (۳)$$

در اینجا S_B پراکندگی بین‌کلاسی و S_W پراکندگی درون‌کلاسی داده‌های اصلی بوده و بدین صورت تعریف می‌شوند:

$$S_W = \sum_{k=1}^K \sum_{x_n \in C_k} (x_n - m_k)(x_n - m_k)^T,$$

$$S_B = \sum_{k=1}^K N_k (m_k - m)(m_k - m)^T.$$

در اینجا m و m_k به ترتیب نشان‌دهنده میانگین کل داده‌ها و میانگین کلاس k ام هستند. روش جداکننده خطی به دنبال یافتن ماتریس $U \in \mathbb{R}^{n \times l}$ است که تابع (۳) را ماکسیمم کند. در این صورت هر داده $x \in \mathbb{R}^n$ به صورت $y = U^T x$ به فضای با بعد l به صورت خطی منتقل می‌شود. نشان می‌دهیم که جواب ماکسیمم‌سازی (۳)، l تا بردار ویژه متناظر با بزرگ‌ترین مقادیر ویژه مسئله:

$$S_B u = \lambda S_W u,$$

است [۸]. این روش یکی از کاراترین روش‌های کاهش ابعاد خطی است که در بسیاری از کاربردها استفاده می‌شود [۱۸].

در سال‌های اخیر نسخه چندخطی این روش نیز ارائه شده است [۵]. به کمک این ملاک، داده‌ها به زیرفضایی تانسوری تصویر می‌شوند که به صورت هم‌زمان پراکندگی بین‌کلاسی را ماکسیمم و پراکندگی درون‌کلاسی را مینیمم می‌کند. فرض کنید $\mathcal{X}_m \in \mathbb{R}^{I_1 \times I_2 \times \dots \times I_N}$ برای $m = 1, 2, \dots, M$ تانسورهایی از مرتبه N هستند که اندازه I_n وجه n تانسور است. داده‌ها در K کلاس مختلف دسته‌بندی شده و N_k تعداد داده‌ها در کلاس k ام است. ملاک جداکنندگی تانسوری برای این داده‌ها بدین صورت تعریف می‌شود.

$$\tilde{U}^{(n)} \Big|_{n=1}^N = \arg_{\tilde{U}^{(n)} \Big|_{n=1}^N} \max \frac{\sum_{k=1}^K N_k \|\bar{y}_k - \bar{y}\|_f^2}{\sum_{k=1}^K \sum_{i \in C_k} \|y_i - \bar{y}_k\|_f^2}, \quad (۴)$$

که در آن $\bar{y} = \bar{X} \times_1 \tilde{U}^{(1)T} \times_2 \tilde{U}^{(2)T} \dots \times_N \tilde{U}^{(N)T}$ ، داده تصویر شده \bar{y} است. همچنین

$$\bar{y} = \bar{X} \times_1 \tilde{U}^{(1)T} \times_2 \tilde{U}^{(2)T} \dots \times_N \tilde{U}^{(N)T},$$

و

$$\bar{y}_k = \bar{X}_k \times_1 \tilde{U}^{(1)T} \times_2 \tilde{U}^{(2)T} \dots \times_N \tilde{U}^{(N)T}.$$

به ترتیب تصویر میانگین کلاس k و میانگین کل داده‌های نمونه به زیر فضای تانسوری جدید هستند. این روش یکی از روش‌های معروف کاهش ابعاد چندخطی است که در کاربردهای مختلف در سال‌های اخیر به کار گرفته شده است [۵]. از آنجا که این یک مسئله بهینه‌سازی نامحدوب است برای حل این مسئله از روش تناوبی استفاده می‌شود به این معنی که در هر مرحله تمامی ماتریس‌ها به غیر از یکی ثابت گرفته شده و مسئله بر حسب فقط یک مؤلفه حل می‌شود و این فرایند برای همه مؤلفه‌ها تک به تک انجام می‌گیرد. این عمل تا یک تکرار مشخص یا برآورده شدن هم‌گرایی

ادامه پیدا می‌کند. از میزان تغییرات تابع هدف در تکرارها می‌توان به‌عنوان عامل توقف استفاده کرد. حال اگر عامل $\tilde{U}^{(n)}$ متغیر در نظر گرفته شده و بقیه ثابت باشند معادله (۴) با استفاده از روابط تانسوری (۱) و (۲) بر حسب $\tilde{U}^{(n)}$ به صورت (۵) نوشته می‌شود:

$$\tilde{U}^{(n)} = \operatorname{argmax}_{\tilde{U}^{(n)}} \frac{\sum_{k=1}^K n_k \|\bar{y}_k - \bar{y}_f\|_f^2}{\sum_{k=1}^K \sum_{i \in C_k} \|y_i - \bar{y}_k\|_f^2} = \operatorname{argmax}_{\tilde{U}^{(n)}} \frac{\operatorname{tr}(\tilde{U}^{(n)T} S_B^{(n)} \tilde{U}^{(n)})}{\operatorname{tr}(\tilde{U}^{(n)T} S_W^{(n)} \tilde{U}^{(n)})}, \quad (5)$$

که در آن $S_B^{(n)}$ و $S_W^{(n)}$ بدین صورت محاسبه می‌شوند:

$$S_B^{(n)} = \sum_{k=1}^K n_k \cdot (\bar{X}_{k(n)} - \bar{X}_{(n)}) \cdot \tilde{U}_{\Phi(n)} \cdot \tilde{U}_{\Phi(n)}^T \cdot (X_{m(n)} - \bar{X}_{(n)})^T,$$

$$S_W^{(n)} = \sum_{k=1}^K \sum_{i \in C_k} (X_{i(n)} - \bar{X}_{k(n)}) \cdot \tilde{U}_{\Phi(n)} \cdot \tilde{U}_{\Phi(n)}^T \cdot (X_{i(n)} - \bar{X}_{k(n)})^T.$$

و $\tilde{U}_{\Phi(n)}$ ضرب کرونکر زیر است

$$\tilde{U}_{\Phi(n)} = (\tilde{U}^{(n+1)} \otimes \tilde{U}^{(n+2)} \otimes \dots \otimes \tilde{U}^{(N)} \otimes \tilde{U}^{(1)} \otimes \tilde{U}^{(2)} \dots \otimes \tilde{U}^{(n-1)}).$$

از روابط ماتریسی مشخص است که جواب مسئله ماکسیم‌سازی (۵) یعنی $\tilde{U}_n \in \mathbb{R}^{I_n \times k_n}$ برابر با k_n تا بردار ویژه مناظر k_n تا از بزرگ‌ترین مقادیر ویژه مسئله مقادیر ویژه تعمیم یافته

$$S_W^{(n)} \tilde{U}^{(n)T} = S_B^{(n)} \tilde{U}^{(n)T} \Lambda,$$

است. از این رو، در هر تکرار از الگوریتم N تا مسئله مقدار ویژه تعمیم یافته به بالا باید حل شوند. از این رو، روش حل این‌گونه مسائل سهم به‌سزایی در کیفیت و نیز سرعت روش‌ها دارد. در ادامه ما از روشی سریع و دقیق که مسئله مقدار ویژه تعمیم یافته را به صورت تفاضلی حل می‌کند استفاده می‌کنیم.

۲. الگوریتم جداکننده چندخطی با استفاده از ملاک تفاضلی

چنان‌که در بخش قبلی دیده شد در روش جداکننده چندخطی نیاز به حل مسئله ماکسیم کردن تریس به صورت (۶) به دفعات زیاد است:

$$\max_{W^T W = I} \frac{\operatorname{trace}(W^T A W)}{\operatorname{trace}(W^T B W)}, \quad (6)$$

روش متداول برای ماکسیم کردن نسبت تریس مذکور با استفاده از بردارهای ویژه حاصل از مسئله مقادیر ویژه تعمیم یافته است. استفاده از این روش در صورت بدحالت بودن ماتریس B کارآمد نیست. به همین علت در این قسمت روشی جدید با استفاده از الگوریتم تکراری نیوتن شرح می‌دهیم که مشکل مذکور را نداشته و سریع‌تر به جواب می‌رسد [۱۷]. فرض کنید که $W_{opt} \in \mathbb{R}^{n \times k}$ جوابی بهینه برای مسئله ماکسیم‌سازی نسبت تریس (۶) باشد. ρ^* به صورت (۷) تعریف می‌شود:

$$\rho^* = \frac{\operatorname{trace}(W_{opt}^T A W_{opt})}{\operatorname{trace}(W_{opt}^T B W_{opt})}, \quad (7)$$

از آن‌جا که ρ^* بیش‌ترین مقدار ممکن برای نسبت تریس است، بنا براین برای هر ماتریس متعامد $W \in \mathbb{R}^{n \times k}$ رابطه $\frac{\operatorname{trace}(W^T A W)}{\operatorname{trace}(W^T B W)} \leq \rho^*$ برقرار است و نتیجه (۸) به دست می‌آید:

$$\text{trace}(W^T A W) - \rho^* \cdot \text{trace}(W^T B W) \leq 0, \quad (۸)$$

با توجه به رابطه مذکور برای W_{opt} و ρ^* شرط (۹) وجود دارد:

$$\max_{W^T W = I} \text{trace}(W^T (A - \rho^* \cdot B) W) = \text{trace}(W_{opt} (A - \rho^* \cdot B) W_{opt}) = 0. \quad (۹)$$

تابع $f(\rho)$ به صورت (۱۰) تعریف می‌شود:

$$f(\rho) = \max_{W^T W = I} \text{trace}(W^T (A - \rho \cdot B) W). \quad (۱۰)$$

با توجه به تابع f ، ماتریس G به صورت (۱۱) تعریف می‌شود:

$$G(\rho) = A - \rho B. \quad (۱۱)$$

مقادیر ویژه G به ترتیب نزولی مرتب شده و به صورت (۱۲) برچسب می‌خورند:

$$\mu_1(\rho) \geq \mu_2(\rho) \geq \dots \geq \mu_n(\rho). \quad (۱۲)$$

با توجه به ماتریس G و مقادیر ویژه آن، رابطه (۱۳) برای تابع f برقرار است:

$$f(\rho) = \mu_1(\rho) + \mu_2(\rho) + \dots + \mu_k(\rho). \quad (۱۳)$$

در ادامه با استفاده از تابع G ، تابع f را می‌توان به صورت (۱۴) بازنویسی کرد:

$$f(\rho) = \text{trace}(W(\rho)^T G(\rho) W(\rho)). \quad (۱۴)$$

در این رابطه $W(\rho)$ از k بردار ویژه متناظر با k مقدار ویژه بزرگ‌تر $G(\rho)$ تشکیل شده است. در [۱۷] نشان داده‌اند

که تابع f در (۱۴) دارای این خواص است:

الف) تابع f یک تابع غیرصعودی است.

ب) $f(\rho) = 0$ است اگر و فقط اگر $\rho = \rho^*$ باشد.

با توجه به این خواص واضح است که، ρ^* تنها ریشه تابع f است. بنا براین می‌توان با استفاده از روش تکراری

نیوتن روی تابع f ، ρ^* و W_{opt} را محاسبه کرد. برای استفاده از روش نیوتن نیاز به مشتق گرفتن از تابع f است

مشتق تابع f بدین صورت به دست می‌آید:

$$f'(\rho) = -\text{trace}(W(\rho)^T B W(\rho)).$$

با استفاده از مشتق به دست آمده، فرمول روش نیوتن بدین صورت می‌شود:

$$\rho_{new} = \rho - \frac{\text{trace}(W(\rho)^T (A - \rho \cdot B) W(\rho))}{-\text{trace}(W(\rho)^T B W(\rho))} = \frac{\text{trace}(W(\rho)^T A W(\rho))}{\text{trace}(W(\rho)^T B W(\rho))}.$$

در الگوریتم (۱) الگوریتم تکراری نیوتن برای ماکسیم کردن نسبت تریس با استفاده از ملاک تفاضلی آورده شده

است. ملاک توقف می‌تواند بر اساس تفاوت مقدار هدف در دو تکرار متوالی باشد.

نشان داده شده است که این الگوریتم دقت و سرعت زیادی در حل مسئله مقدار ویژه تعمیم یافته دارد. حال از این

روش در پیدا کردن بردارهای ویژه در هر تکرار از روش جداکننده چندخطی می‌توان استفاده کرد. با توجه به خواص

این روش، هم‌گرایی روش چندخطی جدید نیز نسبت به الگوریتم اصلی چندخطی سریعتر خواهد شد.

ورودی الگوریتم: ماتریس‌های A و B که ماتریس B نیمه‌معین مثبت است.

خروجی الگوریتم: ماتریس $W \in \mathbb{R}^{D \times M}$ که نسبت $\text{Tr}(W^T A W) / \text{Tr}(W^T B W)$ را ماکسیمم می‌کند.

ماتریس W با ماتریس واحد مقداردهی اولیه شود.

$$\rho = \text{Tr}(W^T A W) / \text{Tr}(W^T B W)$$

تا زمانی که هم‌گرایی (تفاوت مقدار هدف دو تکرار متوالی از یک حدی کمتر شود) اتفاق بیفتد کارهای زیر انجام گیرد.

ماتریس W برابر M بردار ویژه متناظر با M مقادیر ویژه بزرگتر ماتریس $A - \rho \cdot B$ قرار داده شود.

$$\rho = \text{Tr}(W^T A W) / \text{Tr}(W^T B W)$$

انتهای حلقه

الگوریتم ۱. الگوریتم تکراری نیوتن برای ماکسیمم کردن نسبت $\frac{\text{Tr}(W^T A W)}{\text{Tr}(W^T B W)}$ با استفاده از ملاک تفاضلی

مراحل الگوریتم جدید روش جداکننده خطی با این ملاک تفاضلی در الگوریتم ۲ آورده شده است. نکته‌ای که باید به آن اشاره کرد این است که اگر چه در این مقاله روش پیشنهادی روی روش جداکننده چندخطی انجام گرفته است اما این روش روی هر روش چندخطی دیگری مانند روش تحلیل مؤلفه‌های اصلی چندخطی که نیاز به محاسبه مقادیر ویژه دارند، نیز به کار گرفته می‌شود.

ورودی الگوریتم: تانسورهای نمونه $\{\mathcal{X}_m \in \mathbb{R}^{I_1 \times I_2 \times \dots \times I_N}\}_{m=1}^M$ که کلاس آن‌ها مشخص شده و $\{P_n\}_{n=1}^N$ ابعاد داده‌های تصویر

خروجی الگوریتم: ماتریس‌های تصویر $\{\tilde{U}^{(n)} \in \mathbb{R}^{I_n \times P_n}\}_{n=1}^N$ که در داده‌های تصویر شده پراکنندگی بین کلاسی ماکسیمم و پراکنندگی درون کلاسی مینیمم شود.

ماتریس‌های $\{\tilde{U}_0^{(n)} \in \mathbb{R}^{I_n \times P_n}, n = 1, 2, \dots, N\}$ با ماتریس‌های واحد مقداردهی اولیه شود.

$\bar{\mathbf{X}}_k$ و $\bar{\mathbf{X}}$ که میانگین کل و میانگین کلاس k است برای $k = 1, 2, \dots, K$ به دست آورده شود.

برای $t = 1, 2, \dots, T_{max}$
برای $n = 1, 2, \dots, N$

$$\tilde{U}_{\Phi}^{(n)} = (\tilde{U}_{t-1}^{(n+1)} \otimes \tilde{U}_{t-1}^{(n+2)} \otimes \dots \otimes \tilde{U}_{t-1}^{(N)} \otimes \tilde{U}_t^{(1)} \otimes \tilde{U}_t^{(2)} \otimes \dots \otimes \tilde{U}_t^{(n-1)}),$$

$$S_B^{(n)} = \sum_{k=1}^K n_k \cdot (\bar{\mathbf{X}}_{k(n)} - \bar{\mathbf{X}}_{(n)}) \cdot \tilde{U}_{\Phi}^{(n)} \cdot \tilde{U}_{\Phi}^{(n)T} \cdot (\mathbf{X}_{m(n)} - \bar{\mathbf{X}}_{(n)})^T,$$

$$S_W^{(n)} = \sum_{k=1}^K \sum_{i \in C_k} (\mathbf{X}_{i(n)} - \bar{\mathbf{X}}_{k(n)}) \cdot \tilde{U}_{\Phi}^{(n)} \cdot \tilde{U}_{\Phi}^{(n)T} \cdot (\mathbf{X}_{i(n)} - \bar{\mathbf{X}}_{k(n)})^T.$$

ماتریس $\tilde{U}_t^{(n)}$ با ماتریس واحد مقداردهی اولیه شود.

$$\rho = \text{Tr}(\tilde{U}_t^{(n)T} \cdot S_B^{(n)} \cdot \tilde{U}_t^{(n)}) / \text{Tr}(\tilde{U}_t^{(n)T} \cdot S_W^{(n)} \cdot \tilde{U}_t^{(n)})$$

تا زمانی که $\|\tilde{U}_t^{(n)} - \tilde{U}_{t-1}^{(n)}\| > \varepsilon$ کارهای زیر انجام گیرد.

ماتریس $\tilde{U}_t^{(n)}$ برابر P_n بردار ویژه متناظر با P_n مقادیر ویژه بزرگتر ماتریس $S_B^{(n)} - \rho \cdot S_W^{(n)}$ قرار داده شود.

$$\rho = \text{Tr}(\tilde{U}_t^{(n)T} \cdot S_B^{(n)} \cdot \tilde{U}_t^{(n)}) / \text{Tr}(\tilde{U}_t^{(n)T} \cdot S_W^{(n)} \cdot \tilde{U}_t^{(n)})$$

اگر $\|\tilde{U}_t^{(j)} - \tilde{U}_{t-1}^{(j)}\| < \varepsilon$ برای $j = 1, 2, \dots, N$ از حلقه خارج شود.

الگوریتم ۲. الگوریتم جداکننده چندخطی با استفاده از ملاک تفاضلی

پیاده‌سازی

در این قسمت برای مقایسه روش جداکننده چندخطی و روش پیشنهادی از دو مجموعه داده بسیار معروف YALE و MINIST استفاده شده است. داده‌ها به دو دسته داده‌های نمونه و داده‌های آزمایش تقسیم شده و با کمک الگوریتم‌های گفته شده کاهش بعد پیدا می‌کنند. داده‌های آزمایش پس از کاهش بعد با استفاده از روش کلاس‌بندی نزدیک‌ترین همسایه^۹، کلاس‌بندی می‌شوند. در پایان نتایج به دست آمده از نظر درصد کلاس‌بندی درست داده‌های آزمایش ارزیابی می‌شوند.

در اینجا روش جداکننده خطی را با نماد LDA، روش جداکننده چندخطی را با نماد DATER^{۱۰} نمایش می‌دهیم. قابل ذکر است که در مقالات دیگر از نمادهای دیگری برای این روش نیز استفاده شده است [۵]. همچنین روش پیشنهادی نیز با نماد DATER(new) نمایش داده شده است. در اینجا برای پیش پردازش داده‌ها از PCA (برای روش خطی) و از روش HOSVD^{۱۱} (برای روش چندخطی) استفاده شده است. برای شروع داده YALE را در نظر می‌گیریم. شکل ۵ تعدادی از داده‌های این مجموعه داده را نشان می‌دهد.



شکل ۵. نمونه‌ای از داده‌های مجموعه داده YALE

مجموعه داده YALE، شامل ۱۶۵ تصویر سیاه و سفید چهره ۱۵ فرد است که در جهات مختلف با شدت نور متفاوت گرفته شده است. تصاویر به ابعاد ۳۲×۳۲ برش داده شده و بعد از نرمال شدن استفاده می‌شود. به صورت تصادفی تعداد ۲، ۳، ۶، ۸ داده از هر فرد به صورت نمونه اختیار شده و از بقیه داده‌ها به منزله داده‌های آزمایش استفاده می‌شود. ابتدا روش چندخطی را با روش خطی مقایسه می‌کنیم. نتایج در جدول ۱ قابل مشاهده است. در اینجا مشاهده می‌شود که برای تعداد نمونه کم، روش LDA نتایج خوبی به نسبت روش چندخطی ندارد. اما با افزایش تعداد نمونه‌ها چون از بدو وضعی ماتریس پراکندگی کاسته می‌شود، شاهد بهبود کیفیت این روش هستیم. اما حساسیت روش چندخطی کم‌تر است. جدول ۱ نشان می‌دهد که با داده آموزشی کم، عدد حالت ماتریس پراکندگی روش خطی بسیار زیاد است اما با افزایش داده‌های آموزشی بهبود می‌یابد. در روش چندخطی همیشه عدد حالت ماتریس پراکندگی در حد معقولی باقی می‌ماند.

حال نتایج روش چندخطی را با روش پیشنهادی مقایسه می‌کنیم. در اینجا از دو ملاک درصد پیش‌بینی درست کلاس‌ها و نیز نسبت فیشر استفاده می‌کنیم. در اینجا نسبت فیشر میزان پراکندگی بین کلاسی به درون کلاسی را نشان می‌دهد و هر چه بزرگ‌تر باشد بهتر است. این ملاک برای ارزیابی بسیاری از روش‌های کاهش ابعاد استفاده می‌شود [۱۹]. این ملاک از روش درصد پیش‌بینی درست کلاس‌ها ملاک بهتری است چون به کیفیت روش کلاس‌بندی وابستگی ندارد. جدول ۲ نشان می‌دهد که با نمونه‌های کم روش پیشنهادی از نظر ملاک درصد پیش‌بینی نتایج بهتری نسبت به روش اصلی را دارد اما از نظر ملاک فیشر روش پیشنهادی همیشه از روش چندخطی اصلی بهتر عمل می‌کند. همچنین از لحاظ زمان مصرفی نیز عموماً بهتر عمل می‌کند.

9. Nearest neighbor

10. Discriminant Analysis with Tensor Representation

11. Higher Order singular Value Decomposition

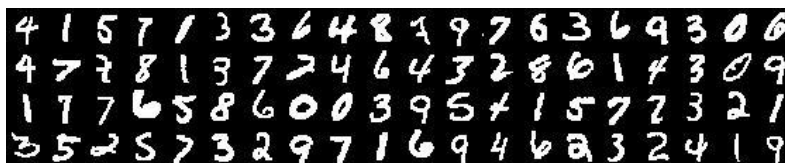
جدول ۱. مقایسه نتایج روش خطی LDA و چندخطی DATER برای مجموعه داده YALE

روش	نمونه در هر کلاس ۲		نمونه در هر کلاس ۳		نمونه در هر کلاس ۶		نمونه در هر کلاس ۸	
	درصد کلاس‌بندی	عدد حالت ماتریس پراکندگی	درصد کلاس‌بندی	عدد حالت ماتریس پراکندگی	درصد کلاس‌بندی	عدد حالت ماتریس پراکندگی	درصد کلاس‌بندی	عدد حالت ماتریس پراکندگی
LDA	۴۳/۲۵	۰/۱۴×۱۰ ^{۳۰}	۴۵/۹۶	۸۳۳۳۳	۷۵/۲	۳۴۵	۸۱/۹	۱۵۹
DATER	۵۱/۷	۲۹/۸	۶۳/۷۵	۲۳/۶	۷۶/۸	۲۲/۹۸	۸۰/۵۵	۲۰

جدول ۲. مقایسه نتایج روش‌های DATER (new) روش پیشنهادی DATER روی مجموعه داده YALE

روش	نمونه در هر کلاس ۲			نمونه در هر کلاس ۳			نمونه در هر کلاس ۶			نمونه در هر کلاس ۸		
	درصد کلاس‌بندی	نسبت فیشر	زمان	درصد کلاس‌بندی	نسبت فیشر	زمان	درصد کلاس‌بندی	نسبت فیشر	زمان	درصد کلاس‌بندی	نسبت فیشر	زمان
DATER	۵۱/۷	۳/۱۲	۷/۲۰	۶۳/۷۵	۲/۳۵	۱۲/۰۸	۷۶/۸	۱/۵۴	۱/۲۳	۸۰/۵۵	۱/۳۸	۵۷/۳۰
DATER(new)	۵۶/۵۹	۵/۴۷	۶/۳۷	۶۴/۰۲	۳/۳۰	۱۱/۰۵	۷۴/۶۶	۲/۳۸	۱/۵۰	۷۸	۲/۰۳	۵۴/۲۴

به عنوان مثال دوم مجموعه MINIST را شامل ۶۰۰۰۰ از دست خط مختلف برای ارقام ۰ تا ۹ با ابعاد ۲۸×۲۸ است را در نظر می‌گیریم. داده‌های نمونه از هر کلاس به تعداد ۲، ۳، ۶، ۱۲ به صورت تصادفی انتخاب می‌شوند و تعداد ۸۰ داده به‌عنوان داده‌های آزمایش استفاده می‌شوند شکل ۶ تعدادی از داده‌های این مجموعه داده را نشان می‌دهد.



شکل ۶. نمونه‌ای از داده‌های مجموعه MINIST

جدول ۳. مقایسه نتایج روش خطی LDA و چندخطی DATER برای مجموعه داده MINIST

روش	نمونه در هر کلاس ۲		نمونه در هر کلاس ۳		نمونه در هر کلاس ۶		نمونه در هر کلاس ۸	
	درصد کلاس‌بندی	عدد حالت ماتریس پراکندگی	درصد کلاس‌بندی	عدد حالت ماتریس پراکندگی	درصد کلاس‌بندی	عدد حالت ماتریس پراکندگی	درصد کلاس‌بندی	عدد حالت ماتریس پراکندگی
LDA	۳۷/۵	0.384 × 10 ¹⁸	۴۵/۸۳	۴۵۵	۵۰	۵۲	۶۶/۰۷	۲۸/۳
DATER	۴۹	۳۶	۵۰/۱	۲۱	۶۱/۸۷	۱۴/۱	۷۱/۵	۱۰/۴

در این مثال نیز رفتار روش‌ها شبیه به مثال اول است. هم‌چنین جدول ۳ نشان می‌دهد که از نظر درصد درست تشخیص کلاس، روش پیشنهادی تنها در داده‌های کم بهتر از روش اصلی عمل می‌کند. اما از لحاظ نسبت فیشر عملکرد بسیار بهتری از روش اصلی را در همه حالت‌ها دارد.

جدول ۴. مقایسه نتایج روش‌های DATER (new) روش پیشنهادی DATER روی مجموعه داده MINIST

روش	نمونه در هر کلاس ۲			نمونه در هر کلاس ۳			نمونه در هر کلاس ۶			نمونه در هر کلاس ۸		
	درصد کلاس‌بندی	نسبت فیشر	زمان	درصد کلاس‌بندی	نسبت فیشر	زمان	درصد کلاس‌بندی	نسبت فیشر	زمان	درصد کلاس‌بندی	نسبت فیشر	زمان
DATER	۴۹	۱/۶۴	۲/۸۵	۵۰/۱	۱/۱۲	۶/۳	۶۱/۸۷	۱/۰۱	۲۴/۰۱	۷۱/۵	۰/۶۷	۷۱/۰۷
DATER(new)	۵۱/۲	۲/۷۲	۲/۴۷	۵۴/۳۷	۱/۹۱	۴/۴	۶۱/۰۱	۱/۳۱	۱۸/۲۵	۶۸/۳۰	۰/۷۱	۷۱/۱۵

نتایج پیاده‌سازی سه الگوریتم LDA، DATER، DATER(new) نشان می‌دهد که در صورت کم بودن تعداد نمونه‌ها در کلاس، روی کرد تفاضلی استفاده شده در الگوریتم DATER(new) بهترین جواب را از لحاظ تشخیص

درست کلاس‌ها در بین این سه الگوریتم می‌دهد و با افزایش تعداد نمونه‌ها در هر کلاس به‌علت خوش‌حالت‌شدن ماتریس‌های پراکندگی درون‌کلاسی، نتایج سه الگوریتم به‌هم نزدیک می‌شود. از طرف دیگر چون کیفیت درصد درست تشخیص کلاس‌ها به کیفیت روش کلاس‌بندی نیز وابسته است، ما از یک روش ارزیابی دیگر که ربطی به کلاس‌بندی ندارد استفاده کردیم. این ملاک نسبت فیشر است. نتایج با این ملاک نشان می‌دهد که روش پیشنهادی در همه حالات بهتر از روش اصلی کار می‌کند. هم‌چنین زمان روش پیشنهادی نیز از روش اصلی کم‌تر است.

نتیجه‌گیری

در این مقاله یک روش چندخطی از روش‌های جداکننده چندخطی داده شد. این روش بر مبنای بهبود حل مسئله ماکسیمم‌سازی تریس ساخته شده است. نتایج عددی برتری کیفیت این روش به‌ویژه برای داده‌ها با داده‌های آموزشی کم در مقایسه با دیگر روش‌ها را نشان می‌دهد. هم‌چنین این روش به‌دلیل جداسازی کلاس‌ها و نیز فشرده‌سازی داده‌های درون‌کلاسی که با معیار فیشر اندازه‌گیری می‌شود از روش اصلی بهتر عمل می‌کند. اگرچه ما این روش را روی جداکننده چندخطی اعمال کردیم اما این روش با هر روش کاهش ابعاد چندخطی مانند مؤلفه‌های اصلی چندخطی که به مسئله مقدار ویژه ختم می‌شوند به‌کار برده می‌شود.

منابع

1. Rezghi M., Abulkasim A., "Noise-free principal component analysis: An efficient dimension reduction technique for high dimensional molecular data, Expert systems with Applications, 41(2014) 7797-7804.
2. Murphy K. P., "Machine Learning: A Probabilistic Perspective", MIT Press (2012).
3. Duda R. O., Hart P. E., Stork D. H., "Pattern Classification (2nd ed.)", Wiley (2006).
4. Binesh N., Rezghi M., "Fuzzy clustering in community detection based on nonnegative matrix factorization with two novel evaluation criteria", Applied Soft Computing, Accepted (2017).
5. Li Q., Schonfeld D., "Multilinear Discriminant Analysis for Higher-order Tensor data Classification", IEEE Transactions on Pattern recognition and Machine Learning, 36 (2014) 2524-2537.
6. Wang J., Barreto A., Wang L., chen Y., Rische N., Andrianm J., Adjouadi M., "Multilinear principal component analysis for face recognition with fewer features", Neurocomputing, 73, (2010) 1550-1555.
7. Kong S., Wang D., "A Report on Multilinear PCA Plus Multilinear LDA to Deal with Tensorial Data: Visual Classification as An Example", Arxiv (2012).
8. Bishop C., "Pattern Recognition and Machine Learning, Springer (2006).

9. Yan S., Xu D., Yang Q., Zhang L., Tang X., Zhang H. J., "Discriminant analysis with tensor representation, in Proc", IEEE Conference on Computer Vision and Pattern Recognition, vol. I (2005) 526-532.
10. Hao N., Kilmer M., Braman K., Hoover R. C., "Facial Recognition Using Tensor-Tensor Decompositions", SIAM Imaging, 6 (2013) 437-463.
11. Yang J., Zhang D., Frangi A. F., Yang J., "Two-dimensional PCA: a new approach to appearance-based face representation and recognition", IEEE Transactions on Pattern Analysis and Machine Intelligence (2004) 131-137.
12. Ye J., "Generalized low rank approximations of matrices, Machine Learning, vol. 61, no. 1-3 (2005) 167-191.
13. Ye J., Janardan R., Li Q., "Two-dimensional linear discriminant analysis", in Advances in Neural Information Processing Systems (NIPS) (2004) 1569-1576.
14. Ren C. X., Dai D. Q., "Bilinear Lanczos components for fast dimensionality reduction and feature extraction", Pattern recognition, 43 (2010) 3742-3752.
15. Wu G., Xu W., Leng H., "Inexact and incremental bilinear Lanczos components algorithms for high dimensionality reduction and image reconstruction", Pattern Recognition, 48 (2015) 244-263.
16. Lu H., Plataniotis K. N., Venetsanopoulos A. N., "Multilinear principal component analysis of tensor objects for recognition", in Proc. Int. Conf. on Pattern Recognition (August 2006) 776-779.
17. Ngo T. T., Bellalij M., Saad Y., "The trace ratio optimization problem for dimensionality Reduction", SIAM J. Matrix Anal. and Appl (2010) 2950-2971.
18. Laohakiat S., Phimoltares S., Lurisinsap C., "A clustering algorithm for stream data with LDA-based unsupervised localized dimension reduction, Information science", 381 (2017) 104-123.
19. Raducanu B., Dornaika F., "A supervised non-linear dimensionality reduction approach for manifold learning", Pattern recognition, 45 (2012) 2432-2444.