بیست و دومین کنفرانس مهندسی برق ایران
دانشگاه شهید بهشتی، ۳۰ اردیبهشت ماه الی ۱ خردادماه ۱۳۹۳
The 22nd Iranian Conference on Electrical Engineering (ICEE 2014)
Shahid Beheshti University, May 20-22, 2014, Tehran, Iran

انتشار برچسب برمبنای اطلاعات محلی و تعیین وفقی تعداد و میزان شباهت همسایه‌ها

سید علی‌رضا صفاری [۱] و عباس ابراهیمی مقدم [۲]
[۱] دانشجوی کارشناسی ارشد گروه برق دانشگاه فردوسی مشهد، arsaffari@yahoo.com
[۲] استادیار گروه برق دانشگاه فردوسی مشهد، a.ebrahimi@um.ac.ir

چکیده - در بسیاری از کاربردهای عملی در بینایی ماشین، تعداد بسیار کمی از نمونه‌ها برچسب‌دار بوده و لذا دقت طبقه‌بندی پایین است. از طرفی فرآیند برچسب‌زنی توسط انسان، فرآیندی بسیار زمان‌بر و نیازمند کار تخصصی می‌باشد. در چنین شرایطی الگوریتم‌های یادگیری شبه ناظر بعنوان یک راه‌کار خوب مورد استفاده قرار می‌گیرند. در این الگوریتم‌ها از گراف‌های ε−همسایگی و k نزدیکترین همسایه برای ساخت یک گراف شباهت استفاده می‌شود. این گراف‌ها نه تنها حساسیت بالایی نسبت به نویز دارند بلکه تعیین پارامترهای بهینه ε و k برای آن‌ها مشکل است. برای غلبه بر این معایب، الگوریتم‌هایی پیشنهاد شده‌اند که در آن‌ها از بیان تنک استفاده شده است. با وجود برتری‌هایی که بیان تنک دارد، این تئوری در طی فرآیند کدینگ، اطلاعات محلی را در نظر نگرفته و حل آن نیازمند یک فرآیند بهینه‌سازی سنگین و زمان‌بر است. کدینگ LLC راه‌حلی برای در نظر گرفتن اطلاعات محلی در حین فرآیند کدینگ می‌باشد. در این مقاله ما کارآیی استفاده از اطلاعات محلی را در قالب یک الگوریتم انتشار برچسب مورد بررسی قرار داده و سه الگوریتم انتشار برچسب جدید ارائه نموده‌ایم. نتایج آزمایش بر روی سه دیتاست آزمایشی از UCI و دو دیتابیس چهره نشان می‌دهد که الگوریتم‌های ما نرخ طبقه‌بندی بالاتری نسبت به دیگر الگوریتم‌ها دارند.

کلید واژه- الگوریتم‌های یادگیری شبه ناظر، الگوریتم‌های مبتنی بر گراف، انتشار برچسب، بیان تنک، اطلاعات محلی.

۱- مقدمه

طبقه‌بندی اطلاعات یکی از موضوعات مورد علاقه محققین در مبحث یادگیری ماشین است. الگوریتم‌های طبقه‌بندی بسیاری توسط محققین ارائه و توسعه یافته‌اند. معمولاً در این الگوریتم‌ها به منظور طبقه‌بندی نمونه‌های تست یا نمونه‌های بدون برچسب، از نمونه‌های یادگیری یا همان نمونه‌های برچسب‌دار استفاده می‌شود. اما متأسفانه هنگامی که تعداد نمونه‌های یادگیری به مراتب کمتر از نمونه‌های تست باشد، این روش‌ها دقت مناسبی ندارند. لذا می‌بایست نمونه‌های بیشتری را برچسب‌دار کرده و از آن‌ها به عنوان نمونه یادگیری استفاده نمود. اما فرآیند برچسب‌زنی توسط انسان، فرآیندی بسیار زمان‌بر بوده که نیازمند کار تخصصی و زحمت فراوان است. در چنین شرایطی یکی از رهیافت‌های مناسب برای رفع مشکل فوق این است که در طبقه‌بندی اطلاعات، هم از نمونه‌های برچسب‌دار و هم از نمونه‌های بدون برچسب استفاده کنیم. در روش‌های یادگیری شبه ناظر که یکی از موضوعات بسیار داغ و فعال در مباحث بینایی ماشین است [1-7]، از نمونه‌های برچسب‌دار و بدون برچسب در فرآیند یادگیری استفاده می‌شود. یکی از روش‌های یادگیری شبه ناظر که تحقیقات بسیاری پیرامون آن صورت پذیرفته است، رهیافت‌های یادگیری شبه ناظر مبتنی بر گراف هستند. در این الگوریتم‌ها، دانشی را که در باره میزان شباهت داده‌ها به یکدیگر در اختیار داریم در قالب یک گراف بیان می‌کنیم. برای چنین توصیفی، گراف $G=(V,E)$ را که در آن مجموعه رئوس V شامل همه نمونه‌های برچسب‌دار و بدون برچسب و مجموعه یال‌های E که شامل میزان شباهت بین داده‌های متناظر رئوس آن یال است، در نظر می‌گیریم. به چنین گرافی، گراف شباهت گفته می‌شود. دو نوع معمول از گراف‌های شباهتی که مورد استفاده قرار می‌گیرند عبارت اند از: ۱- گراف ε−همسایگی ۲- گراف k نزدیکترین همسایه. یادگیری شبه ناظر می‌تواند به عنوان یک مسئله برش حداقل تعریف شود [8].

روش‌های انتشار برچسب که به طور مستقیم برچسب نمونه‌های یادگیری را به نمونه‌های تست انتشار می‌دهند [9-13]، از جمله روش‌های یادگیری شبه ناظر مبتنی بر گراف هستند. در روش Consistency [9] از کرنل گوسی به منظور تعیین وزن یال‌ها استفاده شده است. در واقع در این الگوریتم، وزن یال‌ها به

کمـــــــــک $e_{ij} = \exp\left(-\left\|\mathbf{x}_i - \mathbf{x}_j\right\|^2 / 2\sigma^2\right), i \neq j$ و $e_{ii} = 0$ تعیین می‌شود. در [10,11] از گراف k نزدیکترین همسایه به عنوان گراف شباهت استفاده شده است. در روش‌های یادگیری شبه ناظرِ مبتنی بر گراف که توسط محققین ارائه شده‌اند معمولاً بر روی ساختار گراف تمرکز شده و وزن‌های بدست آمده برای یال‌ها به‌طور مجزا تعریف گردیده‌اند. گراف‌های ε–همسایگی و k نزدیکترین همسایه که معمولاً در این روش‌ها مورد استفاده قرار می‌گیرند دارای معایب زیر هستند: ۱- حساسیت بسیار بالا به نویز. ۲- عدم تعیین وفقی پارامترهای ε و k.

اخیراً بیان تنک (SR) کاربردهای بسیاری را در مباحث بینایی ماشین و شناسایی آماری الگو پیدا کرده است[20-14]. در [17] از مفهوم graph – ℓ_1 (که بر مبنای SR می‌باشد) به منظور ایجاد یک الگوریتم مبتنی بر گراف استفاده شده است. در graph – ℓ_1 ، ساختار گراف و وزن یال‌ها به‌طور همزمان و به کمک مینیمم‌سازی ℓ_1–نرم بدست می‌آیند. در الگوریتم‌های مبتنی بر SR به علت "بیش کامل" بودنِ ماتریس دیکشنری، ممکن است نمونه‌هایی که به عنوان همسایه تعیین می‌شوند، واقعاً در نزدیکی و مجاورت نمونه مربوطه نباشند و قید تنک بودن در SR، به اجبار آن نمونه‌ها را به عنوان نمونه‌ی همسایه انتخاب کرده باشد. در واقع، SR اطلاعات محلی نمونه‌ها را در طی فرآیند کدینگ حفظ نمی‌کند. همچنین حل SR نیازمند یک فرآیند بهینه‌سازیِ سنگین و زمان‌بر است. اخیراً نوعی کدینگِ سیگنال با نام LLC [21] ارائه شده است که در آن از قید اطلاعات محلی به‌جای قید تنک بودن استفاده شده است و حل تحلیلی ساده‌ای برای آن بدست آمده است. در این مقاله ما کارآیی استفاده از اطلاعات محلی همانند LLC را در قالب یک الگوریتم انتشار برچسب مورد بررسی قرار می‌دهیم.

ادامه این مقاله به صورت زیر تنظیم شده است: در بخش دوم این مقاله مروری مختصری را بر تئوری SR و کدینگ LLC ارائه می‌کنیم. الگوریتم‌های پیشنهادی ما در بخش سوم این مقاله معرفی می‌شوند. بخش چهارم این مقاله به بررسی نتایج آزمایش‌ها اختصاص دارد. سرانجام در بخش پنجم نتیجه‌گیری آمده است.

۲- مروری بر کارهای مرتبط

۲-۱- تئوری SR در مبحث بینایی ماشین

تئوری SR را می‌توان به فرم یک مسئله بهینه‌سازی به صورت زیر تعریف نمود:

$$\arg\min_{\mathbf{w}_i} \|\mathbf{w}_i\|_0 \ \ s.t. \ \ \left\|\widetilde{\mathbf{D}}\mathbf{w}_i - \mathbf{x}_i\right\|_2 \le \varepsilon \qquad (۱)$$

$\mathbf{w}_i$ بردار وزن بدست آمده برای نمونه i–ام و $\widetilde{\mathbf{D}} \in \mathcal{R}^{d\times(n-1)}$ ماتریس دیکشنری شامل تمامی نمونه‌های برچسب‌دار و بدون برچسب است که نمونه i–ام از آن حذف شده باشد. همچنین $\|\cdot\|_0$ بیانگر ℓ_0–نرمِ بردار $\mathbf{w}_i$ است که تعداد عناصر غیر صفر $\mathbf{w}_i$ را بدست می‌دهد. اما از آن‌جا که $\|\cdot\|_p$ برای $0 \le p < 1$ محدب نبوده و از همه مهم‌تر، مشتق پذیر نیست، حل معادله(۱) به‌طور کلی NP-hard است[22].

از آن‌جا که نرم یک ($\|\cdot\|_1$)، نزدیکترین نرم محدب به نرم صفر است، جایگزینی نرم صفر با نرم یک، منطقی‌ترین تقریب به نظر می‌رسد و معادله(۱) را می‌توان به فرم زیر بازنویسی کرد:

$$\arg\min_{\mathbf{w}_i} \|\mathbf{w}_i\|_1 \ \ s.t. \ \ \left\|\widetilde{\mathbf{D}}\mathbf{w}_i - \mathbf{x}_i\right\|_2 \le \varepsilon \qquad (۲)$$

که الگوریتم‌های زیادی برای حل آن وجود دارد[25-23]. این رابطه را می‌توان به فرم زیر نیز نوشت:

$$\arg\min_{\mathbf{w}_i} \left\|\widetilde{\mathbf{D}}\mathbf{w}_i - \mathbf{x}_i\right\|_2 + \lambda\|\mathbf{w}_i\|_1, i = 1,2,\ldots,n \qquad (۳)$$

در این‌جا λ پارامتر تنظیم میزان تنکیِ بردار وزن است.

۲-۲- کدینگ LLC

کدینگ LLC به صورت زیر تعریف می‌شود:

$$\arg\min_{\mathbf{w}_i} \left\|\widetilde{\mathbf{D}}\mathbf{w}_i - \mathbf{x}_i\right\|_2 + \lambda\|\mathbf{p}_i \odot \mathbf{w}_i\|_2^2 \qquad (۴)$$
$$s.t. \ \mathbf{1}^T\mathbf{w}_i = 1, \ i = 1,2,\ldots,n$$

نماد $\odot$ در معادله بالا نشانگر ضرب المان به المان و $\mathbf{p}_i$ یک وفق‌دهنده‌ی محلی و به صورت یک ماتریس قطری است که k–امین عنصر روی قطر اصلی آن، فاصله بین نمونه‌های i–ام و k–ام است. قید تغییرناپذیر بودن به شیفت $\mathbf{1}^T\mathbf{w}_i$ بدین علت است که تغییر در مبدأ محور مختصاتِ داده‌ها، تأثیری در انتخاب بردار وزن $\mathbf{w}_i$ نداشته باشد.

۳- الگوریتم پیشنهادی

در این قسمت از مقاله، ساخت یک گراف به کمک کدینگ LLC را بررسی کرده و در ادامه دو الگوریتم خطی و یک الگوریتم کرنلی برای انتشار برچسب بر مبنای کدینگ LLC ارائه می‌شود.

۳-۱- ساخت گراف شباهت بر مبنای LLC

فرض کنید ماتریس دیکشنری $\mathbf{D}=\left[\mathbf{x}_1,\ldots,\mathbf{x}_\ell,\mathbf{x}_{\ell+1},\ldots,\mathbf{x}_n\right]\in\mathcal{R}^{d\times n}$ شامل تمامی نمونه‌ها در اختیار داریم به طوریکه ℓ نمونه اول ($\mathbf{x}_i, i\le\ell$) نمونه‌های برچسب‌دارا و مابقی نمونه‌ها ($\mathbf{x}_u, \ell+1\le u\le n$) نمونه‌های بدون برچسب باشند. ماتریسِ برچسبِ نمونه‌ها را به صورت $\mathbf{Y}=\left[\mathbf{Y}_\ell;\mathbf{Y}_u\right]$ تعریف می‌کنیم که $\mathbf{Y}_\ell$ برچسبِ نمونه‌های یادگیری و $\mathbf{Y}_u$ برچسبِ نمونه‌های تست می‌باشند. هدف ما تخمین مناسب $\mathbf{Y}_u$ است. برای انجام این کار ما از مینیمم‌سازی تابع هدف زیر استفاده می‌کنیم:

$$\min_{\mathbf{Y}}\sum_{i=1}^{n}\left\|\mathbf{y}_i-\sum_{j:\mathbf{x}_j\in N\{\mathbf{x}_i\}}\mathbf{w}_{ij}\mathbf{y}_j\right\|^2 \qquad (۵)$$

که در این رابطه $\mathbf{y}_i\in\mathcal{R}^{1\times c}$ بیانگر بردار برچسب است که عناصر آن را می‌توان به عنوان احتمال تعلق نمونه i-ام به کلاس‌های مختلف در نظر گرفت. بر طبق این تابع هدف، برچسب هر نمونه می‌تواند برحسب ترکیب خطیِ برچسب همسایه‌هایش ساخته شود. با استفاده از جبر خطی می‌توانیم یک حل ساده برای مسئله بهینه‌سازی فوق بدست آوریم:

$$\begin{aligned}&\min_{\mathbf{Y}}\sum_{i=1}^{n}\left\|\mathbf{y}_i-\sum_{j:\mathbf{x}_j\in N\{\mathbf{x}_i\}}\mathbf{w}_{ij}\mathbf{y}_j\right\|^2=\left\|\mathbf{Y}(\mathbf{I}-\mathbf{W})\right\|^2\\&=tr\left(\mathbf{Y}^T(\mathbf{I}-\mathbf{W})^T(\mathbf{I}-\mathbf{W})\mathbf{Y}\right)=tr\left(\mathbf{Y}^T\mathbf{M}\mathbf{Y}\right)\end{aligned} \qquad (۶)$$

که در آن $\mathbf{I}$ ماتریس واحد و $\mathbf{M}=(\mathbf{I}-\mathbf{W})^T(\mathbf{I}-\mathbf{W})$ یک ماتریس متقارن است. ماتریس $\mathbf{M}$ را به چهار زیرماتریس به‌صورت زیر تقسیم می‌کنیم:

$$\mathbf{M}=\begin{bmatrix}\mathbf{M}_{LL}&\mathbf{M}_{LU}\\\mathbf{M}_{UL}&\mathbf{M}_{UU}\end{bmatrix} \qquad (۷)$$

در نتیجه معادله(۵) به صورت زیر قابل بازنویسی است:

$$\begin{aligned}&tr\left(\mathbf{Y}^T\mathbf{M}\mathbf{Y}\right)=\\&=tr\left(\begin{bmatrix}\mathbf{Y}_\ell\\\mathbf{Y}_u\end{bmatrix}\begin{bmatrix}\mathbf{M}_{LL}&\mathbf{M}_{LU}\\\mathbf{M}_{UL}&\mathbf{M}_{UU}\end{bmatrix}\begin{bmatrix}\mathbf{Y}_\ell&\mathbf{Y}_u\end{bmatrix}\right)\end{aligned} \qquad (۸)$$

از طرفین بالا برحسب $\mathbf{Y}$ مشتق گرفته و برابر صفر قرار می‌دهیم و در نتیجه داریم:

$$\begin{cases}\mathbf{M}_{LL}\mathbf{Y}_\ell+\mathbf{M}_{LU}\mathbf{Y}_u=0\\\mathbf{M}_{UL}\mathbf{Y}_\ell+\mathbf{M}_{UU}\mathbf{Y}_u=0\Rightarrow\mathbf{Y}_u=-\mathbf{M}_{UU}^{-1}\mathbf{M}_{UL}\mathbf{Y}_\ell\end{cases} \qquad (۹)$$

در نتیجه با معلوم بودن برچسب نمونه‌های یادگیری یعنی $\mathbf{Y}_\ell$، می‌توان تخمینی از $\mathbf{Y}_u$ بدست آورد. برای ساخت گراف شباهت از تابع هدف LLC به صورت زیر استفاده می‌کنیم:

$$\begin{aligned}&\arg\min_{\mathbf{w}_i}\left\|\mathbf{x}_i-\widetilde{\mathbf{D}}\mathbf{w}_i\right\|_2^2+\lambda\left\|\mathbf{p}_i\odot\mathbf{w}_i\right\|_2^2\\&s.t.\ \mathbf{1}^T\mathbf{w}_i=1\end{aligned} \qquad (۱۰)$$

که در آن $\mathbf{p}_i=\left\{p_{ij}\right\}_{j\ne i,j=1,\ldots,n}$ همان وفق‌دهنده محلی است.

از طرفی ضرایب $\mathbf{w}_i$ به‌صورت زیر قابل محاسبه است [21]:

$$\begin{cases}\widetilde{\mathbf{w}}_i=\left(\mathbf{C}+\left\{diag\left(\mathbf{p}_i\right)\right\}^2\right)\backslash\mathbf{1}\\\mathbf{w}_i=\widetilde{\mathbf{w}}_i/\mathbf{1}^T\widetilde{\mathbf{w}}_i\end{cases} \qquad (۱۱)$$

در ادامه با استفاده از مفاهیم مطرح شده در بالا، دو الگوریتم خطی و یک الگوریتم کرنلی برای انتشار برچسب بر مبنای LLC ارائه می‌دهیم.

۳-۲- انتشار برچسب به کمک LLC (LPLLC)

ما دو نوع از وفق‌دهنده‌های محلی را به کار می‌بندیم. اولین وفق‌دهنده محلی به‌صورت زیر در نظر گرفته می‌شود:

$$\mathbf{p}_i=\left\{p_{ij}\right\}_{j\ne i,j=1,\ldots,n}=\left\{\left\|\mathbf{x}_i-\mathbf{x}_j\right\|_2\right\} \qquad (۱۲)$$

ما الگوریتمی را که بر مبنای این وفق‌دهنده محلی ساخته می‌شود LPLLC-L2 می نامیم. دومین وفق‌دهنده محلی استفاده شده در این مقاله به صورت یک رابطه نمایی تعریف می‌شود:

$$\mathbf{p}_i=\left\{p_{ij}\right\}_{j\ne i,j=1,\ldots,n}=\left\{\exp\left(\frac{\left\|\mathbf{x}_i-\mathbf{x}_j\right\|_2}{\sigma}\right)\right\} \qquad (۱۳)$$

که σ یک پارامتر مثبت است. ما الگوریتمی را که بر مبنای این وفق‌دهنده محلی ساخته می‌شود LPLLC-exp می نامیم.

۳-۳- انتشار برچسب به کمک LLC در حوزه کرنل (LPKLLC)

در این قسمت از مقاله با حل الگوریتم LLC در حوزه کرنل، الگوریتم انتشار برچسب بر مبنای LLC در حوزه کرنل ارائه می‌دهیم که آن را LPKLLC می‌نامیم. می‌توانیم معادله(۴) را در حوزه کرنل به صورت زیر بازنویسی کنیم:

$$\arg\min_{\mathbf{w}_i} \left\| \varphi(\mathbf{x}_i) - \widetilde{\mathbf{D}}^{\phi}\mathbf{w}_i^{\phi} \right\|_2^2 + \lambda \left\| \mathbf{p}_i^{\phi} \odot \mathbf{w}_i^{\phi} \right\|_2^2 \quad (۱۴)$$
$$s.t.\ \mathbf{1}^T \mathbf{w}_i^{\phi} = 1$$

که در آن آن φ، یک نگاشت غیر خطی است و ماتریس $\widetilde{\mathbf{D}}^{\phi} = \left[\varphi(\mathbf{x}_1), \varphi(\mathbf{x}_2), \ldots, \varphi(\mathbf{x}_n)\right] \in \mathcal{R}^{L\times n}$ بیانگر ماتریس دیکشنری، پس از نگاشت اطلاعات به فضای ویژگی با بعد بالا $\mathcal{H}$ می‌باشد و $L \gg d$ بعد فضای $\mathcal{H}$ است. همچنین

$$\mathbf{p}_i^{\phi} = \left\{ p_{ij}^{\phi} \right\}_{j\neq i,\, j=1,\ldots,n} = \left\{ \left\| \varphi(\mathbf{x}_i) - \varphi(\mathbf{x}_j) \right\|_2 \right\}_{j\neq i,\, j=1,\ldots,n}$$

وفق‌دهنده محلی در حوزه کرنل است. به‌طور مشابه، بردار وزن $\mathbf{w}_i^{\phi}$ به صورت زیر قابل محاسبه است:

$$\begin{cases} \widetilde{\mathbf{w}}_i^{\phi} = \mathbf{S}^{-1}\mathbf{1} = \left(\mathbf{C}^{\phi} + \left\{ diag\left(\mathbf{p}_i^{\phi}\right) \right\}^2 \right)^{-1} \mathbf{1} \\ \mathbf{w}_i^{\phi} = \widetilde{\mathbf{w}}_i^{\phi} / \mathbf{1}^T \widetilde{\mathbf{w}}_i^{\phi} \end{cases} \quad (۱۵)$$

به علت مجهول بودن نگاشت غیرخطی φ، معادله(۱۵) به‌طور مستقیم قابل حل نمی‌باشد. برای حل این معادله به شیوه زیر عمل می‌کنیم:

$$\begin{aligned} \mathbf{C}^{\phi} &= \left\{ \varphi(\mathbf{x}_i)\mathbf{1}^T - \widetilde{\mathbf{D}}^{\phi} \right\}^T \left\{ \varphi(\mathbf{x}_i)\mathbf{1}^T - \widetilde{\mathbf{D}}^{\phi} \right\} \\ &= \left\{ \mathbf{1}\varphi^T(\mathbf{x}_i) - \widetilde{\mathbf{D}}^{\phi^T} \right\} \left\{ \varphi(\mathbf{x}_i)\mathbf{1}^T - \widetilde{\mathbf{D}}^{\phi} \right\} \\ &= \mathbf{1}K(i,i)\mathbf{1} - K(:,i)\mathbf{1}^T - \mathbf{1}K(i,:) + K(:,:) \end{aligned} \quad (۱۶)$$

که در آن $K(i,j)$ بیانگر عنصر سطر i-ام و ستون j-ام، $K(i,:)$ بیانگر سطر i-ام و $K(:,j)$ بیانگر ستون j-ام از گرام ماتریس می‌باشد که معمولاً معلوم است. همچنین می‌توان نوشت:

$$\begin{aligned} \left\{ diag\left(\mathbf{p}_i^{\phi}\right) \right\}^2 &= diag\left(\left\{ \left\| \varphi(\mathbf{x}_i) - \varphi(\mathbf{x}_j) \right\|_2^2 \right\}_{j\neq i,\, j=1,\ldots,n} \right) \\ &= diag\left(\left\{ K(i,i) + K(j,j) - 2K(i,j) \right\}_{j\neq i,\, j=1,\ldots,n} \right) \end{aligned} \quad (۱۷)$$

با استفاده از معادلات(۱۵ تا ۱۷)، می‌توانیم بردار وزن $\mathbf{w}_i^{\phi}$ را در حوزه کرنل بدست آوریم.

۴- آزمایشات

به منظور بررسی عملکرد الگوریتم‌های ارائه شده، در این قسمت از مقاله آزمایشاتی را بر روی چند بانک اطلاعاتی انجام می‌دهیم. برای انجام مقایسه‌ای بهتر در مورد کارآیی الگوریتم‌های مطرح شده، چندین الگوریتم طبقه‌بندی مورد استفاده قرار گرفته شده‌اند: الگوریتم‌های SRC [17]، KSRC [18] و K-NNC، سه الگوریتم یادگیری شبه ناظر مبتنی بر گراف LNP06 [10,11]، LNP08 [12] و consistency [9]، چهار الگوریتم یادگیری شبه ناظر مبتنی بر گرافی که بر مبنای SR هستند از جمله l1graph-SR، l1graph-NSR [17]، SLSR [26] و LPSN [27]. این نکته قابل ذکر است که ما الگوریتم l1graph ارائه شده در [17]، را در قالب l1graph-NSR تعریف نموده‌ایم. همچنین ما الگوریتم l1graph-SR را به این صورت ساخته‌ایم که بعد از محاسبه ضرایب SR، ضرایبی را که منفی بوده‌اند صفر در نظر گرفته‌ایم و از بردار بدست آمده به عنوان وزن یال‌ها استفاده کرده‌ایم که متفاوت با ایده اصلی l1graph است و به همین علت آن را l1graph-SR نامیده‌ایم. شیوه‌ی ما در یافتن پارامترهای بهینه برای الگوریتم‌های مطرح شده بدین صورت است که در ابتدا فرض می‌کنیم برچسب تمامی نمونه‌های تست مشخص باشد سپس همه الگوریتم‌ها را یک بار اجرا کرده و پارامترهای بهینه را برای آن‌ها می‌یابیم و در ادامه از این پارامترها به عنوان پارامتر بهینه در سایر آزمایشات استفاده می‌کنیم. مزیت این روش در آن است که الگوریتمی که به این شیوه از سایر الگوریتم‌ها بهتر باشد نه تنها کارآیی طبقه‌بندی بهتری دارد بلکه حساسیت کمتری به تنظیم پارامترها دارد.

ما از PCA [28] به منظور کاهش ویژگی و از توابع متلب "spgl1" [29] و "l1-ls-nonneg" [30] به ترتیب برای محاسبه ضرایب SR و ضرایب NSR استفاده می‌کنیم. همچنین کرنل معروف RBF به منظور نگاشت نمونه‌ها به فضای ویژگی با بعد بالا، به کار رفته است.

۴-۱- ارزیابی بر روی چند دیتاستِ آزمایشی از UCI

در این قسمت از آزمایشات، ما کارآیی الگوریتم‌های ارائه شده را بر روی سه دیتاست آزمایشی از مجموعه UCI [31] مورد ارزیابی قرار می‌دهیم. اطلاعات دیتاست‌های استفاده شده، در جدول ۱ آمده است.

برای دیتاست‌های Wine، Iris و Sonar به ترتیب ۱۰ درصد، ۵ درصد و ۲۰ درصد از نمونه‌ها به طور تصادفی برای یادگیری و

مابقی برای تست انتخاب شده‌اند. همه الگوریتم‌ها ۱۵ بـار تکـرار شده‌اند. جدول ۲ نتایج حاصل از این آزمایش را نشان می‌دهد.

جدول۱: اطلاعات مربوط به دیتاست‌های استفاده شده از UCI

نام دیتاست	تعداد نمونه‌ها	تعداد ویژگی‌ها	تعداد کلاس‌ها
Wine	۱۷۸	۱۳	۳
Iris	۱۵۰	۴	۳
Sonar	۲۰۸	۶۰	۲

جدول۲: میانگین و انحراف معیارِ نرخ شناسایی(%) سیزده الگوریتم، بر روی دیتاست‌های Wine، Iris و Sonar.

تعداد نمونه‌های یادگیری / الگوریتم	Wine	Iris	Sonar
LPSN	۷۲.۶۹±۴.۰۷	۹۰.۶۹±۴.۷۱	۷۳.۵۴±۸.۵۷
LPLLC-L2	**۷۵.۱۰**±۴.۴۸	۹۲.۸۷±۴.۵۸	۸۳.۱۶±۵.۹۲
LPLLC-exp	۷۴.۹۴±۳.۶۰	**۹۶.۱۶**±۱.۲۶	**۸۳.۶۶**±۴.۹۳
LPKLLC	۷۴.۵۵±۴.۱۲	۹۴.۰۴±۴.۳۲	۶۸.۹۳±۸.۴۹
LNP06	۶۸.۹۰±۴.۲۲	۷۵.۳۲±۹.۰۶	۶۳.۵۰±۱۳.۵۰
LNP08	۵۰.۱۴±۶.۱۹	۷۸.۷۰±۴.۷۴	۵۸.۷۷±۴.۱۷
Consistency	۵۵.۴۱±۴.۱۸	۸۳.۳۳±۴.۳۷	۵۱.۵۶±۲.۷۷
L1graph-SR	۷۳.۲۱±۳.۷۰	۹۲.۷۸±۵.۴۴	۶۸.۴۸±۲.۷۰
L1graph-NSR	۷۴.۰۱±۴.۸۰	۹۰.۰۵±۷.۲۳	۷۰.۶۲±۷.۶۹
SLSR	۷۳.۰۷±۳.۳۹	۹۲.۷۸±۵.۴۴	۶۸.۴۸±۲.۷۰
SRC	۷۱.۳۸±۳.۵۱	۹۳.۹۸±۳.۲۴	۶۸.۵۲±۳.۷۳
KSRC	۵۶.۷۳±۳.۶۲	۹۱.۸۷±۵.۹۶	۶۸.۵۳±۳.۴۸
K-NNC	۵۳.۲۹±۳.۴۱	۸۱.۶۷±۲.۶	۵۰.۳۷±۲.۱۹

از جدول(۲) می‌توان دید کـه بـرای بـرای دیتـاسـت Wine کارآیی LPLLC-L2 بهتر از ۱۲ الگوریتم دیگـر اسـت. همچنـین بـــرای دیتـــاســـت Iris و Sonar الگـــوریتم LPLLC-exp نـــرخ طبقه‌بندی بالاتری نسبت به دیگر الگوریتم‌ها دارد.

۴-۲- ارزیابی بر روی دیتابیس چهره ORL

برای انجام آزمایش بر روی ORL [32]، به طور تصـادفی L تصویر از هر نفر (L یک عدد از بازه[۶-۲] است) برای یادگیری و مابقی تصاویر برای تست در نظر گرفته شده‌اند. به علت بعد بالای نمونه‌ها، ابتدا توسط PCA برای کاهش بعد استفاده شده اسـت و تعداد ویژگی هر نمونه به ۱۱۰ ویژگی کاهش پیـدا کـرده اسـت. جدول(۳) میانگین نرخ شناسایی و انحراف معیار مربوطـه، تحـت ۱۵ بار اجرای هر الگوریتم را نشان می‌دهد.

نتایج ایـن آزمـایش نشـان مـی‌دهـد: در تعـداد نمونـه‌هـای یـادگیری ثابـت و کـم، کـارآیی الگـوریتم SLSR بهتـر از سـایر الگوریتم‌ها می‌باشد و با افزایش تعداد نمونه‌های یادگیری، کارآیی الگوریتم LPLLC-exp بهتر از سایر الگوریتم‌ها است.

۴-۳- ارزیابی بر روی دیتابیس چهره Extended Yale B(B+)

برای انجام آزمایش بر روی Extended Yale B (B+) [33]، ما به طور تصادفی ۳ تصویر از هر نفر را برای یادگیری و مابقی نمونه‌ها را به عنوان تست انتخاب کرده‌ایم. آزمایشات بر روی زیرفضاهای m بعدیِ PCA، ۱۵ بار تکرار شده‌اند که m اعداد ۱۰، ۴۰، ۸۰ و ۱۰۰ می‌باشد. میانگین نرخ شناسایی و انحراف معیار حاصل از این آزمایش در جدول(۴) آمده است.

جدول۳: میانگین و انحراف معیارِ نرخ شناسایی(%) ده الگوریتم بر روی دیتابیس ORL برحسب سایزهای متفاوتی از نمونه‌های یادگیری.

تعداد نمونه‌های یادگیری / الگوریتم	۲	۳	۴	۵	۶
LPSN	۷۴.۶۹±۴.۴۲	۸۱.۱۹±۲.۵۶	۸۵.۱۱±۱.۸۷	۸۶.۹۳±۱.۶۴	۸۷.۶۳±۱.۶۴
LPLLC-L2	۸۷.۶۴±۳.۱۸	۹۱.۸۶±۲.۳۹	۹۵.۲۲±۱.۷۳	۹۶.۷۰±۱.۰۱	۹۷.۲۹±۱.۴۹
LPLLC-exp	۸۸.۹۴±۳.۱۷	۹۳.۲۱±۲.۳۰	**۹۵.۸۱**±۱.۳۲	**۹۷.۴۳**±۱.۰۵	**۹۷.۷۵**±۱.۴۱
LPKLLC	۸۴.۶۵±۳.۵۸	۹۰.۲۹±۳.۱۹	۹۳.۸۶±۱.۸۸	۹۶.۰۰±۱.۹۱	۹۷.۰۴±۱.۳۵
LNP06	۵۸.۸۷±۴.۱۳	۶۶.۸۸±۳.۹۵	۸۳.۶۷±۲.۵۹	۸۸.۹۰±۲.۳۸	۹۱.۸۸±۲.۲۳
L1graph-SR	۸۸.۶۲±۲.۹۹	۹۲.۸۳±۲.۰۵	۹۵.۳۹±۱.۵۰	۹۶.۲۰±۱.۵۸	۹۶.۹۶±۱.۳۳
L1graph-NSR	۸۵.۲۳±۲.۷۹	۹۰.۲۹±۲.۶۷	۹۳.۰۶±۲.۰۵	۹۵.۳۰±۱.۷۱	۹۶.۶۳±۱.۸۷
SLSR	**۸۹.۴۰**±۲.۸۱	**۹۳.۶۴**±۱.۹۹	۹۵.۷۸±۱.۷۹	۹۶.۷۷±۱.۱۳	۹۷.۰۰±۱.۳۶
SRC	۸۲.۴۴±۲.۵۹	۸۹.۴۸±۲.۷۸	۹۳.۰۰±۱.۴۵	۹۴.۸۷±۱.۴۵	۹۵.۹۶±۱.۲۷
KSRC	۸۴.۲۱±۲.۲۸	۹۰.۱۹±۲.۷۸	۹۳.۴۷±۱.۵۷	۹۵.۱۳±۱.۵۵	۹۶.۰۸±۱.۲۲

جدول۴: میانگین و انحراف معیارِ نرخ شناسایی(%) ده الگوریتم برحسب بعد، بر روی دیتابیس چهره Extended Yale B (B+).

تعداد ویژگی / الگوریتم	۱۰	۴۰	۸۰	۱۰۰
LPSN	۳۶.۷۱±۲.۱۵	۶۲.۸۸±۳.۰۳	۶۷.۲۸±۳.۴۴	۶۸.۲۹±۳.۱۹
LPLLC-L2	۴۴.۸۴±۲.۶۵	۶۹.۱۳±۲.۵۷	۷۱.۸۷±۳.۵۷	۷۲.۴۱±۳.۶۳
LPLLC-exp	**۴۵.۳۶**±۲.۸۵	**۶۹.۵۶**±۲.۹۲	**۷۲.۲۲**±۳.۱۰	**۷۲.۵۴**±۳.۴۶
LPKLLC	۴۲.۶۲±۲.۷۸	۶۴.۱۲±۴.۱۶	۶۹.۴۸±۲.۹۳	۶۹.۷۱±۳.۰۸
LNP06	۱۲.۱۰±۲.۰۸	۴۲.۹۸±۳.۳۷	۵۷.۱۰±۲.۷۲	۵۸.۲۱±۳.۳۵
L1graph-SR	۳۸.۶۵±۱.۰۲	۶۴.۴۴±۳.۱۳	۶۸.۲۱±۲.۷۹	۶۹.۹۰±۳.۵۸
L1graph-NSR	۴۱.۵۵±۲.۷۴	۶۸.۲۶±۲.۳۸	۷۱.۲۱±۳.۰۲	۷۱.۶۳±۲.۸۹
SLSR	۳۴.۸۰±۲.۷۵	۶۳.۶۵±۲.۴۴	۶۷.۰۶±۲.۷۸	۶۹.۱۳±۳.۵۷
SRC	۳۱.۴۷±۲.۳۳	۴۶.۲۳±۲.۶۴	۴۹.۶۴±۲.۶۱	۵۰.۰۷±۳.۳۹
KSRC	۳۶.۵۵±۲.۳۵	۴۸.۸۹±۲.۲۱	۵۱.۷۵±۳.۰۱	۵۲.۰۶±۳.۵۲

از جدول(۴) می‌توان فهمید که صرف نظر از بعد نمونـه‌هـا و تحت شرایط یکسان، الگـوریتم‌هـای LPLLC-exp و LPLLC-L2 بهترین کارکرد را دارند.

۵- نتیجه‌گیری

با وجود برتری‌هایی که بیان تنک دارد، این تئوری در طی فرآیند کدینگ، اطلاعات محلی را در نظر نگرفته و حل آن نیازمند یک فرآیند بهینه‌سازی سنگین و زمان‌بر است. در این مقاله سه الگوریتم انتشار برچسب LPLLC_L2، LPLLC_exp و LPKLLC ارائه شده‌اند که دو نوع اول الگوریتم‌های خطی و سومین الگوریتم، یک الگوریتم غیرخطی در حوزه کرنل است. در الگوریتم‌های ما بر خلاف الگوریتم‌های مبتنی بر بیان تنک، در حین فرآیند کدینگ از اطلاعات محلی نمونه‌ها استفاده می‌شود. همچنین در این الگوریتم‌ها مشابه الگوریتم‌های مبتنی بر SR، تعداد و میزان شباهت همسایه‌ها برای هر نمونه به طور وفقی تعیین می‌شوند.

برای انجام مقایسه‌ای بهتر در مورد کارآیی الگوریتم‌های مطرح شده، تعداد زیادی از الگوریتم‌های مرتبط را مورد بررسی قرار دادیم. نتایج آزمایش نشان می‌دهد که الگوریتم‌های ما نسبت به دیگر الگوریتم‌ها، علاوه بر اینکه دارای نرخ طبقه‌بندی بالاتری هستند، حساسیت کمتری نسبت به پارامترهایشان دارند.

مراجع

[1] X. Zhu, Z. Ghahramani, and J. Lafferty, "Semi-Supervised Learning Using Gaussian Fields and Harmonic Functions," *Proc. 20th Int'l Conf. Machine Learning*, pp. 912-919, 2003.

[2] C. Rosenberg, M. Hebert, and H. Schneiderman, "Semi-Supervised Self-Training of Object Detection Models," Proc. Seventh IEEE Workshops Application of Computer Vision, pp. 29-36, 2005.

[3] G. Fung and O. Mangasarian, "Semi-supervised support vector machines for unlabeled data classification," *Optimization Methods and Software*, vol. 15, pp. 29–44, 2001.

[4] C. Rosenberg, M. Hebert, and H. Schneiderman, "Semi-supervised self-training of object detection models*," Proc. 7th Workshopn on Applications of Computer Vision*, vol. 1, pp. 29–36, January 2005.

[5] M. Belkin, P. Niyogi, and V. Sindhwani, "Manifold Regularization: A Geometric Framework for Learning from Labeled and Unlabeled Examples," *J. Machine Learning Research,* vol. 7, pp. 2399-2434, 2006.

[6] Y. Song, F. Nie, C. Zhang, and S. Xiang, "A Unified Framework for Semi-Supervised Dimensionality Reduction," *Pattern Recognition,* vol. 41, no. 9, pp. 2789-2799, 2008.

[7] N.D. Lawrence and M.I. Jordan, "Semi-Supervised Learning via Gaussian Processes*," Proc. Neural Information Processing Systems Conf.*, pp. 753-760, 2005.

[8] A. Blum, S. Chawla, "Learning from labeled and unlabeled data using graph mincuts," *in: ICML*, pp. 19–26, 2001.

[9] D. Zhou, O. Bousquet, T.N. Lal, J. Weston, and B. Scho¨lkopf, "Learning with Local and Global Consistency," *Advances in Neural Information Processing Systems*, vol. 16, 2003.

[10] F. Wang, J.D. Wang, C.S. Zhang, H.C. Shen, "Supervised classification using linear neighborhood propagation," *IEEE Conference on CVPR'06*, vol. 1, pp. 160–167, 2006.

[11] J.D. Wang, F. Wang, C.S. Zhang, H.C. Shen, L. Quan, "Linear neighborhood propagation and its applications," *IEEE Transactions on Pattern Analysis and Machine Intelligent*, vol. 31, p.1600-1615, 2009.

[12] F. Wang and C. Zhang, "Label Propagation through Linear Neighborhoods," *IEEE Trans. Knowledge and Data Engineering*, vol. 20, no. 1, pp.55-67, 2008.

[13] Ze Tian, Rui Kuang, "Global Linear Neighborhood for Efficient Label Propagation, ," *Proc. of SIAM International Conference on Data Mining (SDM)*, pp. 863-872, 2012.

[14] M. Elad, M. Figueiredo, and Y. Ma, "On the role of sparse and redundant representations in image processing," *Proceedings of the IEEE*, vol. 98, pp. 972–982, 2010.

[15] J. Wright, Y. Ma, J. Mairal, G. Sapiro, T. S. Huang, and S. Yan, "Sparse Recognition for Computer Vision and Pattern Recognition," *Proceedings of the IEEE*, Vol. 98, No. 6, pp. 1031–1044, 2010.

[16] H. Cheng, Z. Liu, L. Yang, and X. Chen, "Sparse representation and learning in visual recognition: Theory and applications," *Signal Processing 93(6)*, pp. 1408-1425, 2013.

[17] Bin Cheng, Jianchao Yang, Shuicheng Yan, Yun Fu, and Thomas Huang, "Learning with L1-Graph for Image Analysis," *IEEE Transactions on Image Processing (TIP)*, vol. 19, no. 4, pp. 858-866, 2010.

[18] J. Wright, A. Yang, A. Ganesh, S. Sastry, Y. Ma, "Robust face recognition via sparse representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence 31 (2)*, pp. 210–227, 2009.

[19] J. Yin, Z. Liu, Z. Jin, W. Yang, "Kernel sparse representation based classification," *Neurocomputing*, vol. 77, pp. 120-128, 2012.

[20] J. Xu, J. Yang, "A nonnegative sparse representation based fuzzy similar neighbor classifier," *Neurocomputing*, Vol. 99, pp. 76-86, 2013.

[21] J.Wang, J. Yang, K. Yu, F. Lv, T. Huang, Y. Gong, "Locality-constrained linear coding for image classification, *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3360–3367, 2010.

[22] E. Amaldi and V. Kann, "On the Approximation of Minimizing Non Zero Variables or Unsatisfied Relations in Linear Systems," *Theoretical Computer Science*, vol. 209, pp. 237-260, 1998.

[23] S. Chen, D. Donoho, and M. Saunders, "Atomic decomposition by basis pursuit," *SIAM Review* , vol. 43, no. 1, pp. 129–159, 2001.

[24] D. Needell, J. A. Tropp, and R. Vershynin, "Greedy signal recovery review," *42nd Asilomar Conference on Signals, Systems and Computers*, pp. 1048--1050, 2008.

[25] J. A. Tropp, "Greed is good: algorithmic results for sparse approximation," IEEE Trans. on Information theory, vol. 50, no. 10, pp. 2242-2231, 2004.

[26] S.C. Yan, H. Wang, "Semi-supervised learning by sparse representation," *Proceedings of SIAM International Conference on Data Mining (SDM)*, pp. 792–801, 2009.

[27] F. Zang, J. S. Zhang, "Label propagation through sparse neighborhood and its applications," *Neurocomputing* , vol. 97, pp. 267-277, 2012.

[28] M.A. Turk and A.P. Pentland, "Face Recognition Using Eigenfaces," *IEEE Conf. on Computer Visionand Pattern Recognition*, pp. 586-591, 1991.

[29] E. V. Berg, M. P. Friedlander, "Probing the Pareto frontier for basis pursuit solutions," *SIAM J. on Scientific Computing, 31(2)*, pp.890-912, 2008.

[30] SJ Kim, K Koh, M Lustig, S Boyd, D Gorinevsky, "A method for large-scale ℓ1-regularized least squares problems with applications in signal processing and statistics*," IEEE J. Sel. Top. Signal Process 1(4)*, pp.606–617, 2007.

[31] C. Blake, C. Merz, UCI Repository of Machine Learning Databases, 1998.

[32] F.S. Samaria, A.C. Harter, "Parameterisation of a stochastic model for human face identification," *Proceedings of the Second IEEE Workshop on Applications of Computer Vision*, pp.138 –142, 1994.

[33] A. S. Georghiades, P. N. Belhumeur, and D. J. Kriegman, "From few to many: Illumination cone models for face recognition under variable lighting and pose," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 23, no. 6, pp. 643–660, Jun. 2001.