یک روش مبتنی بر اتوماتاهای یادگیر برای آموزش ساختار شبکه های بیزی

نبی الله رضوانی
آزمایشگاه محاسبات نرم
دانشکده مهندسی کامپیوتر و فناوری اطلاعات
دانشگاه صنعتی امیرکبیر
تهران ایران
nabi.rezvani@gmail.com

محمد رضا میبدی
آزمایشگاه محاسبات نرم
دانشکده مهندسی کامپیوتر و فناوری اطلاعات
دانشگاه صنعتی امیرکبیر
تهران ایران
mmeybodi@aut.ac.ir

چکیده: شبکه های بیزی در کاربرد های متعددی به عنوان دسته بندی کننده استفاده شده اند. در ارتباط با یافتن ساختار بهینه برای یک شبکه بیزی دسته بندی کننده، دو چالش اساسی وجود دارد: یافتن تابع ارزیابی کننده که بتواند توصیف درستی از خوب یا بد بودن شبکه ارائه کند و یافتن روشی که بتواند برمبنای این تابع ارزیابی، فضای حالت را به بهترین نحو جستجو نموده و شبکه ای با کیفیت بالا را پیدا کند. در این مقاله یک روش جستجوی مبتنی بر اتوماتاهای یادگیر برای جستجوی ساختار مناسب برای شبکه های بیزی پیشنهاد میگردد. نتایج آزمایشها نشان می دهند که استفاده از روش جستجوی پیشنهادی به می تواند دقت دسته بندی را در مقایسه با روش های جستجوی مکاشفه ای دیگر چون تپه نوردی بهبود دهد.

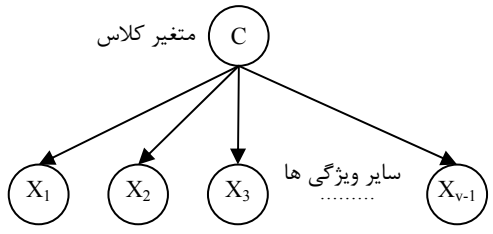
واژه های کلیدی: شبکه های بیزی، آموزش، ساختار، اتوماتاهای یادگیر

۱- مقدمه

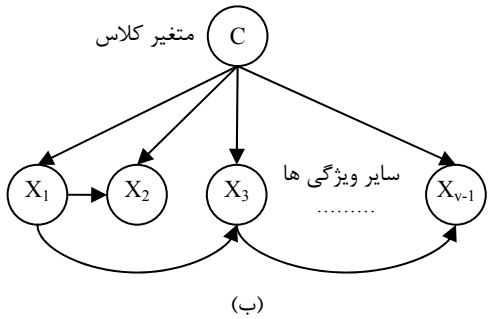
شبکه های بیزی مدل های گرافیکی برای نمایش توزیع احتمالی توام[1] مجموعه ای از متغیر ها هستند. از شبکه های بیزی در کاربرد های فراوانی به عنوان دسته بندی کننده استفاده شده است. در این نوع شبکه های بیزی یک متغیر ویژه با عنوان متغیر کلاس[2] وجود دارد. ساده ترین نوع یک دسته بندی کننده بیزی، دسته بندی کننده بیزی ساده[3] است که در آن متغیر کلاس والد تمام متغیر های شبکه است. در ساختار دسته بندی کننده بیزی ساده این فرض مستتر است که کلیه متغیرها به جز متغیر کلاس، از یکدیگر مستقل شرطی[4] هستند. این دسته بندی کننده علیرغم سادگی کارایی خوبی را در دسته بندی نشان داده است [1].

تحقیقات بسیاری برای بهبود ساختار دسته بندی کننده بیزی ساده انجام شده است. فریدمن ساختاری را پیشنهاد کرد که در آن هر متغیر به جز متغیر کلاس یک والد دیگر نیز دارد. این مدل پیشنهادی TAN[5] نام دارد. در [1] نشان داده شد که ساختار حاصله نسبت به دسته بندی کننده بیزی ساده بر روی مجموعه داده های آزمایشی دقیق تر عمل می کند. شکل ۱ دسته بندی کننده بیزی ساده و دسته بندی کننده TAN را نشان می دهد. در تحقیقات بعدی شبکه های بیزی بدون محدودیت مورد توجه قرار گرفتند. برای آموزش این نوع شبکه ها، یک تابع امتیاز دهی شبکه (یا تابع ارزیابی کننده شبکه) در نظر گرفته می شود و ساختاری که بیشترین مقدار این امتیاز را دارد جستجو می گردد. یافتن شبکه ای که بیشترین امتیاز را دارد و شبکه بهینه نامیده میشود یک مساله NP-Hard است. بنا بر این از روش های جستجوی مکاشفه ای مانند تپه نوردی برای جستجوی ساختار نزدیک به بهینه استفاده می شود. روش تپه نوردی سعی می کند با حذف و اضافه کردن کمان ها در گراف شبکه، مقدار بیشینه تابع امتیاز دهی شبکه را یافته و شبکه ای نزدیک به بهینه را پیدا کند.

مشکل روش های جستجوی مکاشفه ای گیر افتادن در بیشینه های محلی و نرسیدن به بیشینه مطلق تابع امتیاز دهی می باشد. برای این منظور روشهایی از جمله روشهای مبتنی بر الگوریتم های ژنتیک [9] [13] [16] برای رفع این مشکلات پیشنهاد شده است. این روشها جستجوی جامع تری در فضای جستجو برای یافتن شبکه بهینه انجام میدهند. البته این روش های جستجو نیز محدودیت هایی دارند که شانس یافتن شبکه بهینه را کاهش می دهند.



(الف)



شکل۱: (الف)دسته بندی کننده ساده بیزی یا NB، (ب)TAN

در کلیه تحقیق های انجام شده در ارتباط با آموزش ساختار شبکه های بیزی دو مساله مد نظر است: بهبود روش جستجو و بهبود تابع امتیاز دهی شبکه.

در این مقاله یک روش جستجوی مبتنی بر اتوماتاهای یادگیر پیشنهاد می شود. این روش جستجوی جامع تری انجام داده و با احتمال بالاتری بیشینه مطلق تابع امتیاز دهی را پیدا میکند. برای ارزیابی، روش پیشنهادی بهمراه چندین تابع امتیاز دهی متفاوت بر روی مجموعه داده های استاندارد آزمایش گردیده است. نتایج آزمایشها نشان می دهند که استفاده از روش جستجوی پیشنهادی، منجر به یافتن شبکه ای میشود که با دقت بیشتری دسته بندی را انجام میدهد. ادامه این مقاله به صورت زیر سازماندهی شده است. در بخش ۲ شبکه های بیزی، آموزش آنها و دسته بندی کننده های بیزی معرفی شده اند. در بخش ۳ اتوماتا های یادگیر مختصراً معرفی می شوند. در بخش ۴ روش پیشنهادی و در بخش ۵ نتایج آزمایشات ارائه شده است. بخش ۶ نتیجه گیری می باشد.

۲- شبکه های بیزی

یک شبکه بیزی توزیع احتمالی توام[۶] مجموعه ای از v متغیر ($v = \{x_1, \ldots, x_v\}$) را به صورت یک گراف جهت دار بدون دور و مجموعه ای از جداول احتمال شرطی برای هر متغیر، نشان می دهد. در این مقاله فرض می شود که کلیه متغیر ها از نوع گسسته هستند. هر نود نماینده یک متغیر بوده و جدول احتمال شرطی مربوط به آن، حاوی احتمالات شرطی هر مقدار متغیر با دانستن کلیه ترکیب های ممکن از مقادیر والد های آن می باشد. مجموعه والد های x_i که با π_i نشان داده می شود، مجموعه نود هایی هستند که در شبکه یک کمان از آنها به نود x_i وجود دارد. ساختار شبکه بیانگر این مساله است که هر نود با دانستن اطلاعات والدین بلافصل اش از متغیر های غیر فرزند خود مستقل است. بنا بر این احتمال یک واقعه دلخواه $X = (x_1, \ldots, x_v)$ می تواند به صورت $P(X) = \prod_{i=1}^{v} P(x_i \mid \pi_i)$ محاسبه شود. به طور کلی نمایش توزیع احتمال توام مجموعه ای از v متغیر گسسته به فضایی از مرتبه نمایی نسبت به v نیاز دارد. شبکه های بیزی این فضا را به مرتبه نمایی نسبت به $\max_{i \in \{1,\ldots,v\}} |\pi_i|$ کاهش می دهند.

۲-۱ آموزش شبکه های بیزی

با داشتن یک مجموعه از مثال های آموزشی $D = \{X_1, \ldots, X_d, \ldots, X_n\}$ که $X_d=(x_{d,1}, \ldots, x_{d,v})$ ، هدف آموزش، یافتن شبکه بیزی است که به بهترین نحو توزیع توام $P(x_{d,1}, \ldots, x_{d,v})$ را نمایش می دهد. یک راه حل پیدا کردن شبکه B است که امکان داده ها را بیشینه می کند. رابطه امکان لگاریتمی به صورت زیر است:

$$LL(B \mid D) = \sum_{d=1}^{n} \log P_B(X_d) = \sum_{d=1}^{n} \sum_{i=1}^{v} \log P_B(x_{d,i} \mid \pi_{d,i}) \qquad (۱)$$

وقتی که ساختار شبکه معلوم است، این رابطه به برآورد p_{ijk} (احتمال اینکه متغیر i مقدار k اش را دارد، با دانستن اینکه کلیه والد ها مقدار j را دارند، برای کلیه مقادیر i و j و k) کاهش می یابد. وقتی که هیچ مثالی با مقادیر نامعلوم در مجموعه آموزشی وجود نداشته باشد و فرض کنیم که متغیر ها مستقل باشند، برآورد های امکان بیشینه به سادگی برابر با برآورد فرکانس های مشاهده شده $\hat{p}_{ijk} = n_{ijk} / n_{ij}$ هستند. که n_{ijk} تعداد رخداد k امین مقدار x_i در مثال های آموزشی است، با دانستن اینکه والد هایش j امین مقدارشان را دارند. n_{ij} هم مجموع n_{ijk} ها بر روی تمام k ها می باشد.

در این تحقیق فرض می شود که مقادیر نامعلوم در مثال ها وجود نداشته و تمرکز بر روی آموزش ساختار شبکه خواهد بود. چو و لیو الگوریتم ساده ای را برای آموزش ساختار شبکه در حالت خاص ارائه کرده اند. در ساختار مربوطه، هر متغیر یک والد به جز متغیر کلاس دارد [5]. در حالت کلی برای آموزش ساختار شبکه های بیزی دو دسته از روش ها وجود دارد: روش های مبتنی بر تست های مستقل بودن[۷] [6]، و روش های مبتنی بر جستجو [14]. دسته دوم بسیار محبوب تر بوده و در این تحقیق نیز از این روش استفاده می شود.

در روش های مبتنی بر جستجو باید شبکه ای را پیدا کرد که مقدار یک تابع امتیاز دهی را بیشینه کند. یافتن شبکه بهینه که بیشترین امتیاز را دارد، از لحاظ محاسباتی یک مساله NP-Hard است. بنا بر این از روش های جستجوی مکاشفه ای برای جستجوی ساختار نزدیک به بهینه استفاده می شود. یکی از روش های جستجوی اکتشافی که بسیار استفاده شده است، روش تپه نوردی[۸] می باشد [7]. این الگوریتم با یک شبکه اولیه

شروع می کند (که می تواند خالی، اتفاقی و یا برگرفته از دانش یک خبره باشد). در هر مرحله از جستجو همه شبکه های ممکن ساخته شده از شبکه فعلی را با حذف، افزودن و یا برعکس کردن یک کمان ساخته و شبکه های بدست آمده را امتیاز دهی می کند. بهترین شبکه بدست آمده به عنوان شبکه فعلی جدید در نظر گرفته می شود و جستجو تا جایی ادامه می یابد که امتیاز شبکه جدید نسبت به شبکه فعلی بهتر نشود.

از آنجا که به طور میانگین افزودن یک کمان هیچ گاه امکان داده های آموزشی را کم نمی کند، استفاده از امکان لگاریتمی به عنوان تابع امتیاز دهی بر روی داده های آموزشی می تواند به شدت منجر به فرایابی[9] شود. این مشکل به طرق مختلف قابل حل است: ساده ترین آن (که به شکل شگفت انگیزی هم موثر است) محدود کردن تعداد والد های مجاز یک متغیر است. راه دیگر اضافه کردن یک جریمه پیچیدگی به امکان لگاریتمی است. به عنوان مثال روش MDL[10] ([8]) مقدار $MDL(B \mid D) = \frac{1}{2} m \log n - LL(S \mid D)$ (۲) را کمینه می کند، که m تعداد پارامتر های شبکه است. در هر دوی این روش ها پارامتر های هر شبکه کاندید، با برآورد امکان بیشینه مقدار دهی می شوند (مانند وقتی که ساختار شبکه معلوم است).

ماهیت جستجوهای مکاشفه ای چون تپه نوردی به این صورت است که تضمینی برای یافتن یک اکسترمم مطلق در فضای جستجو ارائه نمی دهند و تنها می توان امید داشت که یک اکسترمم محلی خوب را در فضای حالت بیابند. در [9] و [13] نیز یک روش های جستجوی مبتنی بر الگوریتم ژنتیک ارائه شده است. این گونه روش ها نیز دارای محدودیت هایی هستند. از جمله اینکه برای کم کردن تعداد شبکه های غیر معتبر، یک ترتیب بین متغیر های شبکه در نظر می گیرند. هر نود که در ترتیب مقدم است نمی تواند فرزند نود بعدی اش قرار گیرد. این فرض با وجودکوچک تر کردن فضای جستجو، باعث محدود شدن جستجو شده و امکان یافتن اکسترمم نزدیک به مطلق را کاهش می دهد. در [13] و [16] تلاش هایی برای یافتن ترتیب بهینه و بهبود فرضیات محدود کننده این روش ها انجام شده است، اما هنوز این روش ها دارای محدودیت هایی هستند.

۲-۲ دسته بندی کننده های بیزی

هدف دسته بندی، پیش بینی صحیح یک مقدار از متغیر کلاس $y = x_v$ با دانستن یک بردار از خصیصه ها به صورت $(x_1, \ldots, x_{v-1})$ است. اگر معیار ارزیابی دقت دسته بندی (درصد پیش بینی های صحیح بر روی مثال های آزمایشی) باشد، پیش بینی بهینه برای $(x_1, \ldots, x_{v-1})$ کلاسی است که $P(y \mid x_1, \ldots, x_{v-1})$ را بیشینه می کند. اگر یک شبکه بیزی برای $(x_1, \ldots, x_{v-1})$ داشته باشیم این احتمالات را می توان با استنتاج بر روی آن محاسبه کرد. به طور خاص دسته بندی کننده بیزی ساده یک شبکه بیزی است که در آن متغیر کلاس والدی ندارد و هر متغیر دیگر متغیر کلاس را به عنوان تنها والدش دارد. الگوریتم TAN فریدمن ([1])یک اشتقاق از الگوریتم چو و لیو ([5]) است که در آن هر متغیر یک والد دیگر علاوه بر متغیر کلاس دارد.

به طور کلی یک شبکه بیزی آموزش داده شده توسط هر یک از روش های گفته شده، می تواند به عنوان یک دسته بندی کننده استفاده شود. روش هایی که تا کنون بیان شدند، روش های تولیدی[11] هستند. به این معنا که با بیشینه سازی امکان لگاریتمی کل داده ها ($LL(B \mid D)$) که توسط مدل تولید شده، آموزش داده می شوند. با این حال برای مقاصد دسته بندی، تنها امکان لگاریتمی شرطی ($CLL(B \mid D)$) متغیر کلاس با دانستن سایر متغیر ها به مساله مربوط است که:

$$CLL(B \mid D) = \sum_{d=1}^{n} \log P_B(y_d \mid x_{d,1}, \ldots, x_{d,v-1})$$

قابل توجه است که رابطه امکان به نحوی قابل بازنویسی است که شامل رابطه امکان شرطی و یک جمله جمع دیگر می باشد:

$$LL(B \mid D) = CLL(B \mid D) + \sum_{d=1}^{n} \log P_B(x_{d,1}, \ldots, x_{d,v-1})$$

. بیشینه سازی $LL(B \mid D)$ می تواند منجر به کارایی ضعیف در دسته بندی شود. به طور خاص به این علت که ویژگی $CLL(B \mid D)$ می تواند تحت تاثیر عبارت معمولاً بزرگتر $\log P_B(x_{d,1}, \ldots x_{d,v-1})$ قرار بگیرد.

یک راه حل بهتر این است که از خود $CLL(B \mid D)$ به عنوان تابع هدف استفاده شود. این یک فرم از یادگیری تفکیکی[12] است. چرا که تمرکز آن بر روی تفکیک صحیح کلاس ها است. مساله مهم در ارتباط با این روش این است که بر عکس $LL(B \mid D)$ (رابطه (۲-۱))، مقدار امکان شرطی که به صورت زیر است

$$CLL(B \mid D) = \sum_{d=1}^{n} P(x_{d,1}, \ldots, x_{d,v-1}, y) / P(x_{d,1}, \ldots, x_{d,v-1})$$

به صورت عبارات مجزا برای هر متغیر تجزیه نمی شود، و در نتیجه فرم بسته ای برای برآورد بهینه پارامتر ها وجود ندارد. وقتی که ساختار معلوم است، برآورد های بهینه محلی را (برای پارامتر ها) می توان با روش های عددی مثل کاهش در راستای گرادیان بدست آورد. این همان الگوریتم ELR است که در [3] پیشنهاد شده است. وقتی ساختار معلوم نیست یک روش کاهش در راستای گرادیان برای هر شبکه کاندید تولید شده در یک مرحله جستجو لازم است. هزینه محاسباتی این مساله بسیار بالا خواهد بود. در [2] یک روش جستجوی ساختار بر مبنای CLL به همراه برآورد LL برای پارامتر ها آورده شده است. الگوریتم BNC پیشنهاد شده، از جستجوی تپه نوردی به همراه تابع

امتیاز امکان شرطی استفاده می کند. همچنین یک امتیاز تفکیکی دیگر نیز که در [15] آورده شده نرخ دسته بندی نام دارد:

$$CR = \frac{1}{|S|}\sum_{m\in S}^{|S|}\delta(B_S(x_{1:N}^m), c^m) \quad (۳)$$

که $|S|$ تعداد مثال های آموزشی S است. عبارت $\delta(B_S(x_{1:N}^m), c^m)=1$ در صورتی که شبکه بیزی دسته بندی کننده $B_S(x_{1:N}^m)$ که با مثال های S آموزش داده شده برچسب کلاس صحیح c^m را به مقادیر ویژگی های $x_{1:N}^m$ نسبت دهد. توجه کنید که تابع ارزیابی نرخ دسته بندی، از یک دسته بندی کننده آموزش داده شده آزمایش شده بر روی همان مثال های S بدست آمده است.

در این تحقیق یک روش جستجوی مبتنی بر اتوماتای یادگیر برای جستجوی ساختار پیشنهاد می شود که بر مبنای بازخورد کار می کند. این روش جستجو را می توان با توابع امتیاز دهی تفکیکی و تولیدی مختلف استفاده کرد. به علت جستجوی موازی و امکان بررسی کاندیدا های مختلف در این روش، می توان گفت که این روش با امید بیشتری به یافتن یک ساختار بهینه منجر می شود. نتایج آزمایشات مشاهده شده، این مساله را تایید می کنند.

۳- اتوماتا های یادگیر

یک اتوماتای یادگیر یک مدل تجریدی است که به طور اتفاقی یک عمل را از میان مجموعه ای محدود از عمل ها انتخاب کرده[۴] و آن را به محیط اعمال می کند. سپس محیط عمل انتخاب شده را ارزیابی کرده و با یک سیگنال تقویتی به اتوماتا پاسخ می دهد. بر مبنای عمل انتخاب شده و سیگنال تقویتی، اتوماتا[۵] حالت داخلی اش را به روز رسانی کرده و عمل بعدی اش را نتخاب می کند. شکل ۲ رابطه بین یک اتوماتا و محیط اش را بهتر نشان می دهد.

شکل ۲: اتوماتای یادگیر در تعامل با محیط

محیط را می توان با چند تایی $E=\{\alpha,\beta,c\}$ تعریف کرد که $\alpha=\{\alpha_1,\ldots\alpha_r\}$ یک مجموعه محدود از ورودی ها، $\beta=\{\beta_1,\ldots,\beta_r\}$ مجموعه خروجی ها، و $c=\{c_1,\ldots,c_r\}$ مجموع احتمالات جریمه است که هر عنصر c_i از c مربوط به یکی از عمل های ورودی α_i است. محیطی که β تنها می تواند مقادیر دودویی صفر و یک را اختیار کند محیط مدل P^{13} گفته می شود. تعمیم بیشتر محیط منجر به مجموعه خروجی محدودی با بیش از دوعنصر می شود که مقادیرش را از بازه [0 1] می گیرد. چنین محیطی به محیط مدل Q^{14} موسوم است. در نهایت وقتی که خروجی محیط یک متغیر تصادفی پیوسته از بازه [0 1] باشد، به آن مدل S^{15} گفته می شود. اتوماتا های یادگیر به دو دسته غیر قطعی با ساختار ثابت و غیر قطعی با ساختار متغیر تقسیم می شوند. در این تحقیق تنها اتوماتا های با ساختار متغیر مورد بحث هستند.

یک اتوماتای با ساختار متغیر با چهارتایی $\{\alpha,\beta,p,T\}$ تعریف می شود که $\alpha=\{\alpha_1,\ldots\alpha_r\}$ نشانگر مجموعه اعمال اتوماتا، $\beta=\{\beta_1,\ldots,\beta_r\}$ نشان دهنده مجموعه ورودی، $p=\{p_1,\ldots,p_r\}$ نشان دهنده مجموعه احتمالات عمل ها و در نهایت $p(n+1)=T[\alpha(n),\beta(n),p(n)]$ نشان دهنده الگوریتم یادگیری است. این اتوماتا به صورت زیر عمل می کند. بر مبنای مجموعه احتالات عمل ها p، اتوماتا یک عمل α_i را به طور اتفاقی انتخاب کرده و آن را به محیط اعمال می کند. پس از دریافت سیگنال تقویتی از محیط، اتوماتا مجموعه احتمالا ت عمل هایش را برای پاسخ های مطلوب با رابطه (۵) و برای پاسخ های نامطلوب با رابطه (۶) به روز رسانی می کند:

$$p_i(n+1)=p_i(n)+a\cdot(1-p_i(n))$$
$$p_j(n+1)=(1-a)\cdot p_j(n) \quad \forall j \;\; j\neq i$$

$$p_i(n+1)=(1-b)\cdot p_i(n)$$
$$p_j(n+1)=b/(r-1)+(1-b)\cdot p_j(n) \quad \forall j \;\; j\neq i$$

در این دو رابطه a و b به ترتیب پارامترهای پاداش و جریمه هستند. هرگاه $a=b$ باشد الگوریتم یادگیری پاداش-جریمه خطی[۱۶] یا L_{R-P} نامیده می شود، برای $a << b$ پاداش-جریمه خفیف خطی[۱۷] یا $L_{R\xi P}$، و برای $b=0$ پاداش-هیچ کار خطی[۱۸] یا L_{R-I} گفته می شود.

۴- روش آموزش پیشنهادی

روش پیشنهادی از مجموعه ای از LA ها برای جستجو استفاده می کند. در این روش به ازای هر یک از اتصالات ممکن در گراف شبکه یک LA در نظر گرفته می شود. این اتصالات می توانند هر یک از کمان های یک گراف جهت دار کامل با n گره باشند. هر LA دارای ویژگی های زیر می باشد:

- مجموعه عمل ها (α): هر LA متناظر با یک کمان ممکن در شبکه در نظر گرفته شده است. بنا براین عمل هایی که می تواند داشته باشد باید متناظر با وضعیت هر کمان باشد. یک کمان بین دو نود x_i و x_j می تواند وجود نداشته یا با دوجهت مختلف وجود داشته باشد. بنا بر این LA متناظر با آن یکی از این ۳ عمل را می تواند انتخاب کند.
- خروجی(β): خروجی یا بازخورد دریافت شده در واقع باید با تابع امتیاز دهی شبکه بیزی متناظر باشد. امتیاز های تفکیکی استفاده شده در این تحقیق نرخ دسته بندی، امکان و امکان شرطی می باشند که باید بیشینه شوند. اگر امتیاز بدست آمده در مرحله n ام از امتیاز مرحله n-1 ام بیشتر باشد به کلیه LA مشمول پاداش شده و در غیر این صورت کلیه آنها مشمول جریمه خواهند شد.
- تابع یادگیری(T): در هر مرحله از یادگیری بر اساس سیگنال تقویتی دریافت شده از محیط، باید به LA ها پاداش داده یا آنها را جریمه کرد. هر سه الگوریتم یادگیری عنوان شده در بخش ۳ آزمایش شده و نتایج آنها مقایسه شده است.

کلیه LA ها به صورت موازی در هر مرحله جستجو، یک عمل را بر مبنای بردار احتمالات خود انتخاب می کنند. عمل انتخاب شده هر LA به شبکه اعمال شده و باعث تولید یک گراف می گردد. سپس گراف جهت دار تولید شده بررسی می شود تا یک گراف قابل قبول باشد. برای این منظور باید اولاً متغیر کلاس هیچ والدی نداشته باشد و ثانیاً هیچ دوری در گراف بدست آمده وجود نداشته باشد. بررسی قابل قبول بودن یک گراف، هزینه نسبتاً بالایی داشته و در بد ترین حالت از مرتبه $O(n^2)$ نسبت به تعداد متغیر های شبکه می باشد. اگر شبکه حاصله قابل قبول نباشد کلیه LA ها جریمه شده و اگر شبکه قابل قبول باشد بر اساس مقایسه امتیاز مرحله فعلی و قبلی مورد پاداش و یا جریمه قرار می گیرد. جستجو تا جایی ادامه می یابد که امتیاز مرحله فعلی از امتیاز مرحله قبلی بهتر نشود.

۵- نتایج آزمایشها

آزمایشات بر روی ۲۵ مجموعه داده مختلف انجام شده که ۲۳ تای آنها از دانشگاه UCI ([11])و ۲ تای بقیه از مجموعه داده های مربوط به انتخاب ویژگی معرفی شده در [12] می باشند. برای آزمایش طبق مفروضات انجام شده در [1] از اعتبار سنجی ضربدری ۵ تایی[19] برای تحصیل نتایج بهتر استفاده شده است. شرح مختصری از این مجموه داده ها در جدول ۱ آمده است.

جدول ۱: ویژگی های مجموعه داده های استفاده شده

مجموعه داده	تعداد کلاس ها	تعداد خصیصه ها	تعداد مثال ها
australian	2	14	690
breast	2	10	683
chess	2	36	2130
cleve	2	13	296
corral	2	6	128
crx	2	15	653
diabetes	2	8	768
flare	2	10	1066
german	2	20	1000
glass	7	9	214
glass2	2	9	163
heart	2	13	270
hepatitis	2	19	80
iris	3	4	150
letter	26	16	15000
lymphography	4	18	148
mofn-3-7-10	2	10	300
pima	2	8	768
shuttle-small	7	9	3866
vote	2	16	435
satimage	6	36	4435
segment	7	19	1540
soybean-large	19	35	562
vehicle	4	18	846
waveform-21	3	21	300

الگوریتم مبتنی بر LA پیشنهاد شده در ۳ نسخه پیاده سازی و تست شده است. تفاوت این ۳ نسخه در تابع امتیاز دهی است که برای ارزیابی ساختار شبکه استفاده شده است. این ۳ الگوریتم با ۷ الگوریتم مختلف دیگر مقایسه شده است. ویژگی های کلیه روش های مقایسه شده در ذیل آمده است:

- LA-based (ML): در این روش از الگوریتم مبتنی بر LA پیشنهادی به همراه تابع امتیاز دهی امکان بیشینه استفاده شده است. برای برآورد پارامتر ها در۳ روش پیشنهادی مبتنی بر LA از روش امکان بیشینه استفاده شده است.
- LA-based (CL): تابع امتیاز دهی استفاده شده در آن امکان شرطی می باشد. در واقع تکنیک استفاده شده در این روش همان تکنیک BNC پیشنهاد شده در [2] می باشد با این تفاوت که روش جستجوی ساختار روش مبتنی بر LA است.
- LA-based (CR): از تابع نرخ دسته بندی[20] برای ارزیابی شبکه استفاده شده است. این تابع در [4] به عنوان تابع ارزیابی استفاده شده و نتایج نسبتاً خوبی نیز ارائه کرده است.

- NB(ML): از ساده ترین ساختار دسته بندی بیزی است. برای برآورد پارامتر ها نیز از امکان بیشینه استفاده شده است که یک مدل تولیدی محسوب می شود.
- TAN(ML): یک روش توسعه یافته از NB می باشد. همانند روش قبل از امکان بیشینه برای برآورد پارامتر ها استفاده شده است.
- ML-2P: در این روش برای ارزیابی ساختار از امکان بیشینه استفاده می شود. روش جستجو نیز تپه نوردی است. مساله مهم در این روش استفاده از حداکثر دو والد برای هر نود به منظور جلوگیری از فرایابی است. در [4] نشان داده شد که استفاده از امتیاز MDL و جریمه مربوط به آن به عنوان رقیب این روش، نتایج چندان خوبی ندارد و به این منظور این روش در این تحقیق آزمایش نشده است.
- BNC-2P(CL): روش پیشنهاد شده در [2] می باشد که از تپه نوردی به همراه امتیاز امکان شرطی برای جستجوی ساختار استفاده کرده است. تعداد والد های هر نود به ۲ عدد محدود شده و برای برآورد پارامتر ها از امکان بیشینه استفاده شده است.
- BNC(CR): همان روش قبل می باشد با ابن تفاوت که امتیاز استفاده شده برای ارزیابی ساختار، نرخ دسته بندی است.
- NB: برای برآورد پارامترها در NB، از روش ELR پیشنهاد شده در [3] استفاده شده است. این روش از امتیاز امکان شرطی به همراه جستجوی کاهش در راستای گرادیان استفاده کرده است. نتایج نیز نشان داده اند که روش پیشنهادی موثر بوده است.
- TAN(ML): همانند روش قبل از ساختار TAN به همراه روش ELR استفاده شده است.

جدول ۲ نتایج آزمایشات انجام شده با روش های فوق، بر روی مجموعه داده های مورد بحث را نشان می دهد. اعداد ثبت شده، نرخ خطای دسته بندی بر روی مثال های هر مجموعه داده هستند. آخرین سطر جدول نتایج میانگین بدست آمده را نشان می دهد. بهترین نتیجه در میانگین به صورت ضخیم نمایش داده شده است.

از میان الگوریتم های پیشنهادی، الگوریتم هایی که از امتیاز امکان شرطی و نرخ دسته بندی استفاده کرده اند نتایج قابل قبولی داشته اند. استفاده از امتیاز امکان با روش جستجوی پیشنهادی، بر اساس انتظار نتایج خوبی را ارئه نکرده است. استفاده از امکان شرطی به عنوان تابع امتیاز دهی گرچه نتایج خوبی داشته است، اما بهترین نتایج را نشان نداده است. علت این است که در کاربرد های دسته بندی، انتخاب بهترین کلاس هدفی است که امکان شرطی الزاماً به بهترین نحو آن را انجام نمی دهد بلکه هدف امکان شرطی ارائه دقیق مقدار احتمال شرطی کلاس ها با دانستن مقدار سایر خصیصه ها است.

از میان کلیه روش های مقایسه شده، روش جستجوی تپه نوردی به همراه امتیاز نرخ دسته بندی ((BNC(CR) توانسته نتایج نزدیکی به نتایج روش مبتنی بر LA ارائه دهد. این مساله به دلیل استفاده از تابع نرخ دسته بندی است که به علت نزدیک بودن به هدف مساله یعنی همان دسته بندی، توانسته نتایج نسبتاً خوبی ارائه دهد. روش جستجوی پیشنهادی بر روی این امتیاز جستجوی بهتری انجام داده و نتایج آن نیز بهتر است.

شکل ۳ نیز نمودار پراکندگی آزمایشات انجام شده مختلف را در مقایسه با بهترین نسخه روش پیشنهادی نشان می دهد. نقاط قرارگرفته بالای خط y=x نشانگر نقاطی هستند که الگوریتم پیشنهادی بهتر عمل کرده است. واضح است که در اکثر نمودار ها پراکندگی به سمت بالای این خط بوده و الگوریتم پیشنهادی کارایی بهتری داشته است.

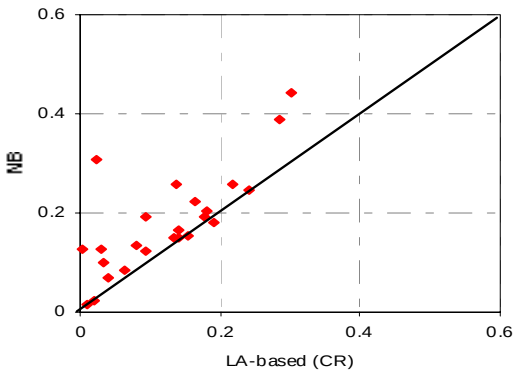
۶- نتیجه گیری

در این مقاله یک روش مبتنی بر اتوماتاهای یادگیر برای تعیین ساختار بهینه در شبکه های بیزی پیشنهاد گردید. با توجه به نتایج آزمایشها این روش میتواند به عنوان رقیبی برای روش های جستجوی اکتشافی مانند تپه نوردی مطرح گردد. نشان داده شد که استفاده از روش جستجوی هنگامی که تابع امتیاز دهی، نرخ دسته بندی میباشد بهترین نتایج را در مقایسه با روشهای اکتشافی مانند تپه نوردی تولید میکند.

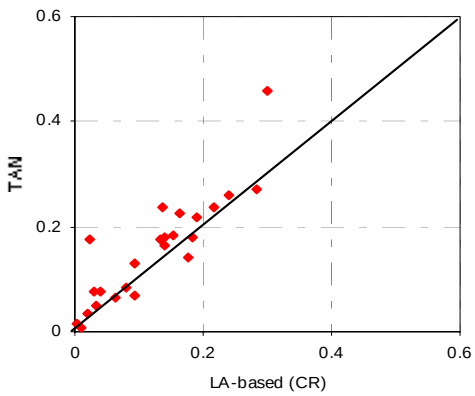
جدول ۲: مقایسه نرخ خطای دسته بندی برای مجموعه داده های مختلف

الگوریتم / مجموعه داده	LA-based (ML)	LA-based (CL)	LA-based (CR)	NB(ML)	TAN(ML)	ML-2P	BNC-2P	BNC(CR)	NB-ELR	TAN-ELR
australian	0.1367	0.1149	0.1341	0.1489	0.1751	0.1391	0.1296	0.1405	0.1488	0.1723
breast	0.0449	0.0411	0.0210	0.0245	0.0351	0.0668	0.0425	0.0231	0.0339	0.0351

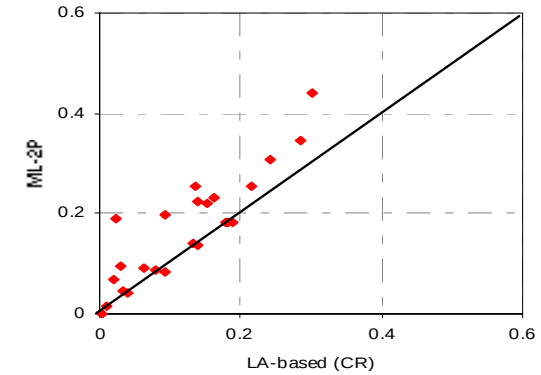
chess	0.1123	0.0469	0.0312	0.1266	0.076	0.0966	0.0422	0.0314	0.06	0.0375
cleve	0.1725	0.1863	0.1894	0.1791	0.2164	0.182	0.1997	0.1914	0.166	0.2164
corral	0.0289	0.0197	0.0041	0.1277	0.0143	0	0.0119	0.0320	0.1273	0.0771
crx	0.1431	0.1401	0.1392	0.1505	0.1631	0.137	0.158	0.1455	0.1505	0.1603
diabetes	0.2563	0.268	0.2168	0.2571	0.2384	0.255	0.2666	0.2421	0.2419	0.2384
flare	0.1904	0.174	0.1823	0.2024	0.178	0.181	0.1804	0.1736	0.1813	0.178
german	0.2156	0.2551	0.2412	0.2458	0.2609	0.3085	0.2643	0.2770	0.2456	0.2609
glass	0.4410	0.3992	0.3012	0.4412	0.4578	0.4412	0.4173	0.2856	0.422	0.5018
glass2	0.2257	0.2413	0.1634	0.2236	0.2249	0.2329	0.2691	0.1775	0.1938	0.2249
heart	0.1890	0.1751	0.1526	0.155	0.1847	0.2221	0.1874	0.1593	0.155	0.1847
hepatitis	0.1945	0.1728	0.9382	0.193	0.1302	0.1967	0.1618	0.0833	0.1294	0.1302
iris	0.0533	0.0411	0.0401	0.0699	0.0763	0.0414	0.042	0.0667	0.0485	0.0763
letter	0.1643	0.1061	0.023	0.3068	0.1752	0.1896	0.183	0.2241	0.3068	0.1752
lymphography	0.1886	0.1554	0.1396	0.1662	0.1784	0.2247	0.1635	0.1461	0.147	0.1784
mofn-3-7-10	0.1021	0.0866	0.0800	0.1328	0.085	0.0859	0.0859	0.0898	0.1367	0
pima	0.2472	0.1823	0.1375	0.2571	0.2384	0.255	0.2666	0.1435	0.2505	0.2384
satimage	0.1848	0.1867	0.1781	0.1915	0.1395	0.184	0.1745	0.1459	0.173	0.142
segment	0.997	0.0662	0.0944	0.1221	0.0675	0.0831	0.0571	0.1403	0.0701	0.0571
shuttle-small	0.0148	0.0041	0.0090	0.014	0.0093	0.0145	0.0047	0.0088	0.0083	0.0052
soybean-large	0.0883	0.0786	0.0629	0.0852	0.0644	0.0922	0.0754	0.0722	0.092	0.0663
vehicle	0.3659	0.2987	0.2836	0.3892	0.2718	0.3451	0.2922	0.3222	0.3453	0.2727
vote	0.0651	0.0374	0.0319	0.0991	0.0509	0.0467	0.0417	0.0618	0.037	0.0487
waveform-21	0.2306	0.1846	0.1892	0.2142	0.2534	0.2543	0.267	0.2181	0.1772	0.2534
میانگین	0.202116	0.146492	**0.125584**	0.18094	0.1586	0.171016	0.159376	0.144072	0.161916	0.157252



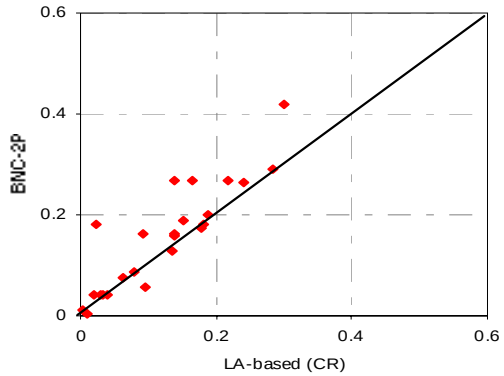
(الف)



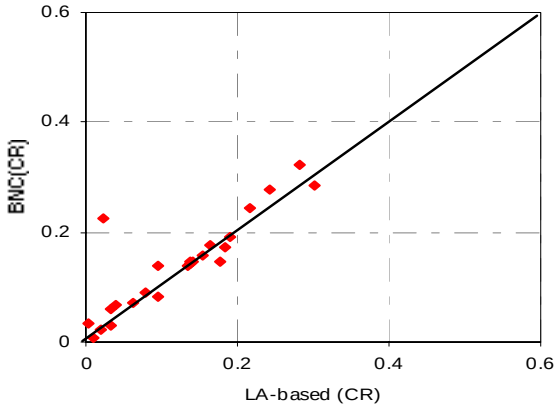
(ب)



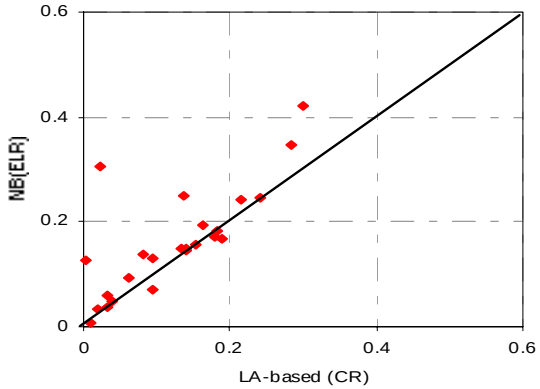
(ج)



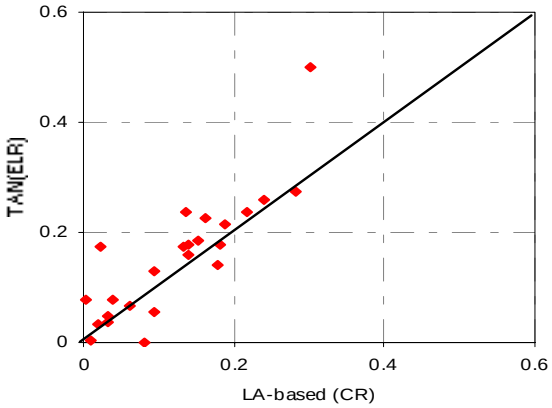
(د)



(ح)



(و)



(ز)

شکل ۳ : نمودار پراکندگی روش های مختلف با روش مبتنی بر LA با امتیاز نرخ دسته بندی. نقاط قرارگرفته بالای خط y=x نشانگر نقاطی هستند که الگوریتم پیشنهادی بهتر عمل کرده است.

مراجع

[1] Friedman, J. H., "On Bias, Variance, 0/1 - Loss, and the Curse-of-Dimensionality", Springer, Data Mining and Knowledge Discovery Journal, 1, 55-77, 1997.

[2] Grossman, D. and Domingos, P., "Learning Bayesian Network Classifiers by Maximizing Conditional Likelihood", ACM, 21st International Conference of Machine Learning (ICML), 69, 361–368, 2004.
[3] Greiner, R. and Zhou, W., "Structural Extension to Logistic Regression: Discriminative Parameter Learning of Belief Net Classifiers", AAAI, 18th National Conference on Artificial Intelligence, 167-173, 2002.
[4] Pernkopf, F. and Bilmes, J., "Discriminative versus Generative Parameter and Structure Learning of Bayesian Network Classifiers", ACM, 22nd International Conference on Machine learning, 119, 657-664, 2005.
[5] Chow, C. K. and Liu, C. N., "Approximating Discrete Probability Distributions with Dependence Trees", IEEE Transactions on, Information Theory, 14, 462-467, 1968.
[6] Spirtes, P., Glymour, C. and Scheines, R., "Causation, Prediction, and Search", Springer, Lecture Notes in Statistics, 1993.
[7] Heckerman, D., Geiger, D. and Chickering, D. M., "Learning Bayesian Networks: The Combination of Knowledge and Statistical Data", Springer Verlag, Machine Learning, 20, 197-243, 1995.
[8] Lam, W. and Bacchus, F., "Learning Bayesian Belief Networks: An Approach Based on the MDL Principle", Elsevier, Computational Intelligence, 10, 269-293, 1994.
[9] Pedro, M., Yosu, R.H. and Cindy, M.H., "Learning of Bayesian Networks by Genetic Algorithms: A Performance Analysis of Control Parameters", IEEE Transactions on, Pattern Analysis and Machine Intelligence, 18, 912-926, 1996.
[10] Thathachar, M. A. L. and Sastry P. S., "Varieties of Learning Automata: An Overview", IEEE Transaction on, Systems, Man, and Cybernetics, 32, 711- 722, 2002.
[11] Blake, C. and Merz, C. J., "UCI Repository of Machine Learning Databases", Department of Information and Computer Science, University of California, Irvine, tp://www.ics.uci.edu/mlearn/MLRepository.html, 2000.
[12] Kohavi, R. and John, G., "Wrappers for Feature Subset Selection", Elsevier Science, Artificial Intelligence, 97, 273-324, 1997.
[13] Larranaga, P., and Kuijpers, C. M. H., Murga, R. H. and Yurramendi Y., "Learning Bayesian Network Structures by Searching for the Best Ordering with Genetic Algorithms", IEEE Transactions on, Systems, Man and Cybernetics, 26, 487-493, 1996.
[14] Murphy, K. P., "An Introduction to Graphical Models", Technical Report, Intel Research Technical Report, 2001.
[15] Pernkopf, F., "Bayesian Network Classifiers versus Selective k-NN Classifier", Elsevier, Pattern Recognition, 38, 1–10, 2005.
[16] Wong, M. L. and Leung, K. S., "An Efficient Data Mining Method for Learning Bayesian Networks Using an Evolutionary Algorithm Based Hybrid Approach", IEEE Transactions on, Evolutionary Computation, 8, 378-404, 2004.

---

[1] Joint Probability Distribution

[2] Class variable

[3] Naïve Bayes classifier

[4] Conditional independent

[5] Tree augmented Naïve Bayes

[6] Joint probability distribution

[7] Independence Tests

[8] Hill climbing

[9] Over fitting

[10] Minimum Description Length

[11] generative

[12] Discriminative Learning

[13] P-model

[14] Q-model

[15] S-model

[16] Linear Reward-Penalty

[17] Linear Reward epsilon Penalty

[18] Linear Reward Inaction

[19] 5-fold cross validation

[20] Classification Rate