یادگیری ساختاری شبکهٔ بیزی با به‌کارگیری پوشش مارکوفی در الگوریتم $K2$

وحید رضایی‌تبار[۱]، سلوا سلیمی[۲]

تاریخ دریافت: ۱۳۹۴/۱۲/۱۰
تاریخ پذیرش: ۱۳۹۵/۷/۱۷

چکیده:
شبکه‌های بیزی، مدل‌های گرافیکی احتمالی هستند که رابطهٔ علّت و معلولی بین متغیرها را تعیین می‌کنند و شامل یادگیری ساختاری و یادگیری پارامتری می‌باشند. الگوریتم $K2$ یکی از بهترین روش‌های یادگیری ساختار در شبکه‌های بیزی برای متغیرهای گسسته است. کارایی الگوریتم $K2$، به‌شدت تحت تأثیر ترتیب متغیرهای ورودی است. بنا بر این برای رسیدن به گراف دقیقی که توصیف‌کنندهٔ داده‌ها باشد، یافتن الگوریتمی که ترتیب دقیق‌تری از عناصر به‌عنوان ورودی $K2$ ارائه کند، مورد نیاز است. در این مقاله، نخست با استفاده از روش افزایشی-کاهشی، پوشش مارکوفی هر متغیر را یافته، سپس بر اساس فراوانی‌های شرطی و استفاده از تابع چگالی احتمال دیریکله، از بین پوشش مارکوفی هر متغیر، والدین احتمالی آن متغیر انتخاب می‌شوند. مجموعهٔ والدین انتخابی هر رأس به‌عنوان ورودی الگوریتم $K2$ مورد استفاده قرار می‌گیرد و شبکهٔ بیزی به دست می‌آید. نتایج حاصل از اعمال الگوریتم پیشنهادی بر روی چند مجموعه دادهٔ معیار و مقایسهٔ آن با روش‌های دیگر، نشان می‌دهد که الگوریتم پیشنهادی بسیار کاراتر از سایر روش‌ها است.

واژه‌های کلیدی: شبکهٔ بیزی، الگوریتم $K2$، پوشش مارکوفی، الگوریتم افزایشی-کاهشی

۱ مقدمه

چارچوب بیزی، روشی برای مدل‌بندی دنیای غنی و پیچیدهٔ اطراف ماست. با استفاده از این چارچوب می‌توان ساختار ذاتی یک فرایند را تفسیر نمود و اعمال مختلفی نظیر استنباط احتمالی و یادگیری را به‌سادگی امکان‌پذیر کرد. یک شبکهٔ بیزی، گرافی متشکل از رأس‌ها (گره‌ها) و یال‌های جهت‌دار میان آن‌هاست که رأس‌ها نشان‌دهندهٔ متغیرهای تصادفی‌اند (پیرل [۹]). دو نوع یادگیری در شبکه‌های بیزی مطرح است: یادگیری ساختاری و یادگیری پارامتری. در یادگیری ساختاری، گراف حاصل از روابط علّت و معلولی بین متغیرها ارائه می‌شود و در یادگیری پارامتری، توزیع‌های شرطی بین متغیرها برآورد می‌شود (هکرمن [۳]). یادگیری ساختاری شبکهٔ بیزی یک مسئلهٔ چندجمله‌ای سخت[۳] است؛ زیرا تعداد ساختارهای ممکن شبکه، با افزایش تعداد گره‌های آن به‌صورت نمایی افزایش می‌یابد و تولید همه ساختارهای ممکن و انتخاب بهترین ساختار از میان آن‌ها کاری بسیار پیچیده و زمان‌بر است (لارانیاگا و همکاران [۶]). الگوریتم $K2$ یکی از بهترین روش‌های یادگیری ساختاری شبکهٔ بیزی برای متغیرهای گسسته است. این الگوریتم بر پایهٔ ترتیب[۴] متغیرهای ورودی استوار است. دقت الگوریتم $K2$ وابستگی زیادی به‌ترتیب ارائه‌شده برای متغیرها دارد و هر اندازه این ترتیب به‌ترتیب واقعی متغیرها نزدیک باشد، شبکهٔ بیزی دقیق‌تری به دست می‌آید (کوپر و هرسکوویتس [۲]). به‌عنوان مثال، داشتن ترتیب A, B, C از متغیرها نشان‌دهندهٔ این است که A والد احتمالی[۵] B و A, B والد احتمالی C هستند. بر این اساس، تمرکز اصلی الگوریتم‌های ارائه‌شده برای بهبود تابع امتیازدهی $K2$، یافتن روشی است که منجر به رسیدن به‌ترتیب دقیقی از

[۱] استادیار گروه آمار، دانشگاه علامه طباطبایی، تهران، ایران
[۲] دانشجوی کارشناسی ارشد رشتهٔ علوم تصمیم و مهندسی دانش، دانشگاه خوارزمی، تهران، ایران

[۳] NP-hard
[۴] order
[۵] candidate parent
[۶] feature ranking

عناصر باشد. هروشکا و ایبکن [۴] به معرفی الگوریتمی بر اساس رتبه‌بندی بر پایهٔ انتخاب ویژگی[۶] پرداخته و بر اساس آزمون آماری χ^2 و بهرهٔ اطلاع[۷] رده‌بندی کرده و پس از ارزیابی میزان وابستگی متغیرها به هم، ترتیبی برای متغیرها در نظر گرفته‌اند. از معایب این روش آن است که روابط علّی میان متغیرها در نظر گرفته نمی‌شود؛ یعنی به‌وضوح مشخص نیست میان دو متغیری که وابستگی زیادی دارند، کدام‌یک والد احتمالی دیگری است. چن و همکاران [۱] با استفاده از ترکیب مفاهیم نظریهٔ اطلاع و توابع جستجو، الگوریتمی برای یافتن ترتیبی از عناصر ارائه کردند. الگوریتم آن‌ها از سه مرحلهٔ اصلی تشکیل شده است که در دو مرحلهٔ اول با استفاده از آزمون‌های استقلال و اطلاع متقابل و مجموعهٔ جداکننده،[۸] ساختار اولیهٔ گراف مشخص می‌شود و در مرحلهٔ آخر، یال‌ها با استفاده از آزمون استقلال و الگوریتم‌های جستجو، جهت‌دهی می‌شوند. الگوریتم کو و کیم [۵] که نسبت به سایر الگوریتم‌ها عملکرد بهتری دارد، بر این اساس شکل گرفته که هر فرزند تحت والد خود، فراوانی شرطی بهتری نسبت به سایر متغیرهایی دارد که والدش نیستند. بنا بر این امتیازدهی میان رئوس بر اساس فراوانی شرطی آن‌ها انجام شده و در نهایت بر اساس این امتیازات، والدین احتمالی هر رأس مشخص می‌شوند. این والدین احتمالی در نهایت به‌عنوان ورودی الگوریتم $K2$ به کار گرفته می‌شوند و شبکهٔ بیزی به دست می‌آید. یکی از مشکلات روش کو و کیم [۵]، تولید یال‌های اضافی در شبکهٔ بیزی است.

در این مقاله برای حل مشکل روش کو و کیم [۵]، نخست پوشش مارکوفی هر رأس را می‌یابیم. سپس والدین احتمالی هر رأس، فقط بر اساس امتیاز میان رأس مذکور و اعضای پوشش مارکوفی آن محاسبه می‌شوند و به‌عنوان ورودی الگوریتم $K2$ مورد استفاده قرار می‌گیرند. به عبارت دیگر، والدین احتمالی هر متغیر از بین اعضای پوشش مارکوفی آن متغیر انتخاب می‌شوند. نتایج حاصل از اعمال الگوریتم پیشنهادی بر روی چند مجموعه شبکهٔ معیار[۹] نشان می‌دهد، الگوریتم پیشنهادی با کاهش تعداد یال‌های اضافی منجر به دقیق‌تر شدن گراف حاصل می‌شود و در نتیجه گراف مذکور، انطباق بیشتری با داده‌ها دارد.

ساختار این مقاله به شرح زیر است: در بخش ۲ به تشریح چارچوب نظری تحقیق پرداخته شده، در بخش ۳ الگوریتم کو و کیم به اختصار توضیح داده شده و پس از یافتن اشکالات این الگوریتم، در بخش ۴ به تشریح الگوریتم پیشنهادی پرداخته شده است. در نهایت، نتایج حاصل از اعمال روش‌های معرفی‌شده توسط چن و همکاران [۱]، هروشکا و ایبکن [۴]، کو و کیم [۵] و همچنین الگوریتم پیشنهادی بر روی چند مجموعه دادهٔ معیار در بخش ۵ ارائه‌شده و گراف حاصل از این روش‌ها با یکدیگر مقایسه شده است.

۲ چارچوب نظری

۱.۲ شبکهٔ بیزی

شبکه‌های بیزی از ابزارهای داده‌کاوی و گروهی از مدل‌های گرافیکی احتمالی هستند که عمدتاً نشان‌دهنده روابط علّت و معلولی بین متغیرها هستند. شبکهٔ بیزی یک گراف جهت‌دار بدون دور[۱۰] است. این گراف با مجموعه رئوس V و مجموعه یال‌های E که زیرمجموعه‌ای از $V \times V$ است، به‌صورت $G = (V, E)$ نشان داده می‌شود. هر گره گراف مربوط به یک متغیر تصادفی در دامنه است و این گراف، یک خانواده از توزیع‌های احتمال را بر روی متغیرهای V نشان می‌دهد. یال‌های جهت‌دار E نشان‌دهندهٔ وابستگی بین رأس‌های مربوط به آن یال هستند. یک یال $Y \to X$ در گراف، رابطهٔ بین والد Y و فرزند X را شرح می‌دهد. به‌علاوه، هر گره یک جدول احتمال شرطی دارد که احتمال هر ترکیب ممکن، یک گره و والدین خود را نشان می‌دهد. اگر یک گره والدی نداشته باشد، احتمال‌های شرطی همان احتمال‌های حاشیه‌ای آن گره خواهند بود (پیرل [۹]).

یکی از مشکلات استفاده از شبکه‌های بیزی، رسیدن به گرافی است که به‌درستی بیان‌کنندهٔ ارتباط میان متغیرهای مختلف در مجموعه‌داده باشد. بنا بر این تلاش‌های زیادی برای یادگیری شبکه‌های بیزی صورت گرفته است. در فرایند یادگیری باید بتوان ساختار شبکهٔ بیزی و جداول احتمالات شرطی آن را تعیین کرد. برای یادگیری خودکار ساختار شبکه، یکی از روش‌ها بر

[۶] information gain

[۸] d-sepration

[۹] benchmark

[۱۰] directed acyclic graph

پایهٔ تعیین وابستگی بین متغیرها بنا نهاده شده و در روش دیگر (یادگیری ساختاری) به دنبال یافتن ساختاری بهینه برای شبکهٔ بیزی هستیم که با داده‌های موجود، بیشترین تطابق را داشته باشد.

یادگیری ساختاری، خود دارای دو رویکرد اصلی است: رویکرد مبتنی بر قید که بر اساس تحلیل وابستگی موجود در داده‌ها، ساختار بهینه را می‌یابد و رویکرد مبتنی بر رتبه‌بندی-امتیازدهی که از یک تابع امتیاز برای رتبه‌بندی ساختارهای ممکن استفاده می‌کند و سپس با بهره‌گیری از یک الگوریتم جستجو، به دنبال کشف ساختاری با بیشترین امتیاز است. این روش نسبت به روش مبتنی بر قید، دقیق‌تر و خروجی آن ساختاری منحصربه‌فرد خواهد بود (نیلسن و ینسن [۸]).

۲.۲ الگوریتم $K2$

الگوریتم $K2$، الگوریتم جستجوی حریصانه‌ای[۱۱] است که ساختار یک شبکهٔ بیزی را بر اساس میزان تطابق آن با داده‌ها به دست می‌آورد. در واقع بر اساس این الگوریتم، ساختاری که تطابق بیشتری با داده‌ها دارد، امتیاز بیشتری کسب کرده، به‌عنوان بهترین ساختار توصیف‌کنندهٔ داده‌ها انتخاب می‌شود (کوپر و هرسکوویتس [۲]). الگوریتم $K2$ برای کاهش پیچیدگی زمانی، ترتیب اولیه‌ای از گره‌ها (متغیرها) را به‌عنوان ورودی در نظر می‌گیرد. بنا بر این یافتن ترتیبی از گره‌ها که والدین احتمالی هر گره را به‌درستی مشخص کند، در رسیدن به یک ساختار شبکهٔ بیزی با امتیاز زیاد، بسیار بااهمیت است. بر این اساس، تمرکز اصلی الگوریتم ارائه‌شده، یافتن ترتیب دقیقی از گره‌ها است. الگوریتم $K2$ به شرح زیر است (کوپر و هرسکوویتس [۲]):

الگوریتم $K2$

- ورودی و خروجی الگوریتم
  ورودی: مجموعه‌ای از n گره یا متغیر تصادفی گسسته، یک ترتیب روی گره‌ها، یک حد بالای u روی تعداد والدین هر گره و یک پایگاه داده شامل n نمونهٔ مستقل.
  خروجی: مجموعه والدین مربوط به هر گره.
- شروع الگوریتم $K2$
  برای i از یک تا n انجام شود.
  مجموعه والدین متغیر را صفر در نظر بگیر؛ یعنی
  $$Pa(x_i) = \emptyset$$
  $$P_{old} = g(x_i, Pa(x_i))$$
  (تابع g متر $K2$ شبکهٔ بیزی است.)
  تا زمانی که متغیری برای ادامه دادن وجود دارد و تعداد والدین گره x_i از u کمتر است، ادامه بده.
  شروع
  تا زمانی که گره z از مجموعه متغیرهایی که x_i می‌تواند به‌عنوان والدین خود انتخاب کند وجود دارد، آن را طوری انتخاب کنید که عبارت $g(x_i, Pa(x_i) \cup \{z\})$ را ماکسیمم نماید.
  $$P_{New} = g(x_i, Pa(x_i) \cup \{z\})$$
  اگر $P_{New} > P_{old}$
  شروع
  $$P_{old} := P_{New}$$
  $$Pa(x_i) = Pa(x_i) \cup \{z\}$$
  پایان
  اگر متغیری که به‌عنوان والدین متغیر انتخاب شود در ترتیب وجود نداشت.
  پایان
  والدین مربوط به متغیر x_i را چاپ کنید.

پایان الگوریتم $K2$

در این الگوریتم، تابع امتیاز به‌صورت زیر تعریف می‌شود (کوپر و هرسکوویتس [۲]):

$$g(x_i\ ,\ Pa(x_i)) = \prod_{j=1}^{q_i} \frac{(r_i-1)!}{(N_{ij}+r_i-1)!} \prod_{k=1}^{r_i} N_{ijk}$$

که در آن

$Pa(x_i)$: مجموعه والدین رأس x_i،

q_i : تعداد حالات والد رأس x_i،

r_i: تعداد حالات متغیر x_i،

N_{ijk}: تعداد نمونه‌های مجموعه‌دادهٔ D

که در آن x_i، kامین حالت و والد آن jامین حالت خود را دارد. همچنین $N_{ij} = \sum_{k=1}^{r_i} N_{ijk}$ تعداد واحدهای نمونه از مجموعه‌داده است که در آن والدین متغیر x_i، jامین حالت خود را دارند.

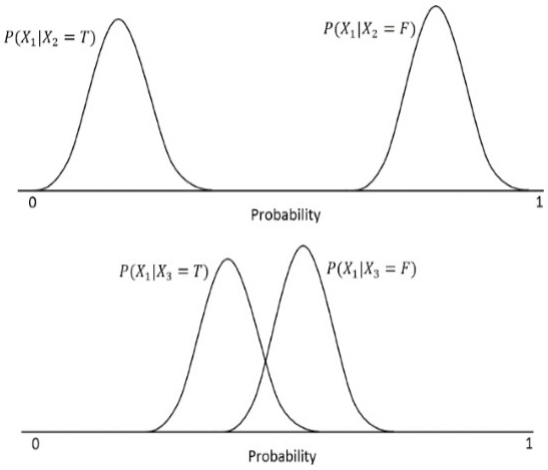
[۱۱] greedy search algorithm

۳ روش کو و کیم در یافتن والدین احتمالی

روش کو و کیم [۵] مبتنی بر توزیع دیریکله، به‌صورت

$$Dir\left(p\left(c_1\right),\ldots,p\left(c_{r_i}\right)\ ;f\left(c_1\right),\ldots,f\left(c_{r_i}\right)\right) \quad (1)$$
$$=\frac{\Gamma\left(\sum_{k=1}^{r_i} f\left(c_k\right)\right)}{\prod_{k=1}^{r_i}\Gamma\left(f\left(c_k\right)\right)}\prod_{k=1}^{r_i} p(c_k)^{f(c_k)-1}$$

است که در آن c_k، kامین سطح یک متغیر، $p(c_k)$ احتمال c_k و $f(c_k)$ فراوانی c_k است. هر اندازه فاصلهٔ توابع چگالی احتمال دیریکلهٔ متغیری بر روی تغییرات والدش بیشتر باشد بیان‌کنندهٔ آن است که متغیر موجود به‌درستی به‌عنوان والد انتخاب شده است.



شکل۱. تابع چگالی احتمال متغیر X_1 به‌شرط دو متغیر X_2 و X_3

بنا بر این با توجه به شکل ۱، متغیر X_2 انتخاب درست‌تری برای X_1 به‌عنوان والد است. برای تعیین والدین هر متغیر، تابع چگالی احتمال دیریکلهٔ آن، در صورتی‌که مطابق شکل ۱ و نمودار مربوط به متغیر X_2 باشد، باید امتیاز بیشتری دریافت کند. برای این منظور، در ابتدا جدول توافقی بین هر جفت متغیر تشکیل شده، فرض می‌شود که متغیر فرزند با سطوح مختلف آن در سطر و متغیر والد با سطوح مختلف آن در ستون جدول توافقی باشد. در این صورت برای تشخیص این‌که آیا متغیر ستون می‌تواند والد احتمالی متغیر سطر باشد به‌صورت زیر عمل می‌شود:

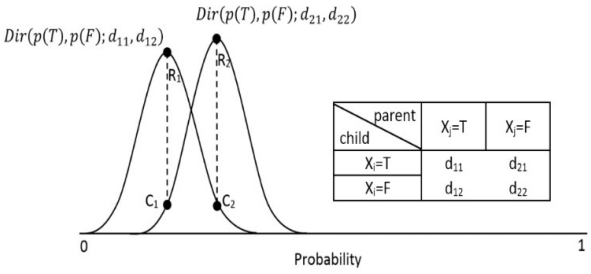
- مقدار احتمال توزیع دیریکله برای ستون اول با در نظر گرفتن فراوانی و احتمال‌های مربوط به آن محاسبه می‌شود.
- سپس مقدار احتمال توزیع دیریکله برای سایر ستون‌ها، با ثابت در نظر گرفتن احتمال‌های مربوط به فراوانی ستون اول محاسبه می‌شود. به‌عبارت دیگر در رابطهٔ (۱)، احتمال $p(c_i)$ها همان احتمالات ستون اول است و فقط فراوانی‌ها از ستون‌های دیگر برای محاسبهٔ احتمال دیریکله انتخاب می‌شوند. هدف از این‌کار، محاسبهٔ احتمال دیریکلهٔ سایر ستون‌ها با در نظر گرفتن احتمال‌های ستون اول است.
- مقدار مینیمم و ماکسیمم قدم‌های قبل به‌ترتیب به‌عنوان c_1 و r_1 در نظر گرفته می‌شوند (شکل ۲). به‌عنوان مثال برای نمونه‌ای مانند شکل ۲ داریم:

$$r_1=\frac{\Gamma(d_{11}+d_{12})}{\Gamma(d_{11})\Gamma(d_{12})}\left(\frac{d_{11}}{d_{11}+d_{12}}\right)^{d_{11}}\left(\frac{d_{12}}{d_{11}+d_{12}}\right)^{d_{12}}$$
$$c_1=\frac{\Gamma(d_{11}+d_{12})}{\Gamma(d_{11})\Gamma(d_{12})}\left(\frac{d_{21}}{d_{21}+d_{22}}\right)^{d_{11}}\left(\frac{d_{22}}{d_{21}+d_{22}}\right)^{d_{12}}$$

- این عملیات برای همهٔ سطوح متغیر والد تکرار می‌شود. بنا بر این به تعداد سطوح متغیر والد، مقادیر r_1 و c_1 خواهیم داشت.
- در نهایت با استفاده از رابطهٔ ۲، میزان وابستگی متغیر فرزند (X_i) به متغیر والد X_j محاسبه می‌شود. در واقع رابطهٔ ۲، فاصلهٔ بین سطوح متغیر والد احتمالی و فرزند را محاسبه می‌کند، که بیشتر بودن این فاصله نشان‌دهندهٔ انتخاب صحیح متغیر والد است. هرچه میزان این فاصله بیشتر باشد نشان از تحت تأثیر بودن متغیر فرزند تحت متغیر والد است و بیان‌کنندهٔ انتخاب درست متغیر والد احتمالی است (به مثال ۱ رجوع شود).

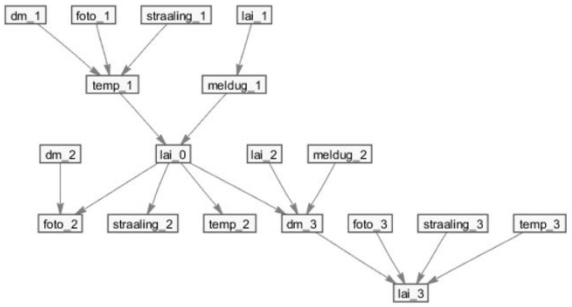
$$score\left(X_j,X_i\right)=Dep\left(X_i|X_j\right) \quad (2)$$
$$=\sum_{k=1}^{r_j}(\log r_k-\log c_k)$$

شایان ذکر است که امتیاز میان هر دو متغیر (رأس یا گره)، با توجه به رابطهٔ (۲)، محاسبه می‌شود و در صورتی که $score(A,B)>score(B,A)$ باشد، رأسA به‌عنوان والد احتمالی رأس B انتخاب شده، الگوریتم $K2$ برای هر رأس بر اساس والدین احتمالی‌اش فراخوانی و محاسبه می‌شود (منظور از $score$، امتیاز بین دو متغیر A و B بر اساس تابع امتیاز $K2$ است).



شکل۲. تابع چگالی احتمال دیریکله، کو و کیم [۵]

مثال ۱.۳. روند کلّی الگوریتم کو و کیم برای مجموعه‌دادهٔ *car* که شامل ۱۸ متغیر (رأس) و ۱۷ یال می‌باشد، در ادامه بررسی شده است. شبکهٔ اصلی مربوط به داده‌های *car*، در شکل ۳ نشان داده شده است. شایان ذکر است که مجموعه‌دادهٔ *car* همراه با شبکهٔ اصلی آن در نرم‌افزار R و در بستهٔ نرم‌افزاری *bnlearn* موجود است. منظور از شبکهٔ اصلی، شبکه‌ای است که الگوریتم‌های یادگیری به دنبال یافتن بیشترین تطابق با این شبکه، برای بررسی میزان کارایی خود می‌باشند.



شکل۳. شبکهٔ اصلی مربوط به مجموعه‌دادهٔ *car*، بستهٔ نرم‌افزاری *bnlearn* در نرم‌افزار R

در الگوریتم کو و کیم [۵]، در ابتدا فراوانی سطوح مختلف هر متغیر به‌ازای سطوح مختلف متغیر دیگر محاسبه می‌شود و سپس احتمال رخداد آن‌ها با استفاده از توزیع دیریکله به دست می‌آید.

جدول ۱. جدول توافقی و احتمالات نظیر آن‌ها برای چهار متغیر مجموعه‌دادهٔ *car*

meldug_1\dm_1	۱	۲
۱	۹۳۳۳	۴
۲	۶۶۳	۰

⇨

meldug_1\dm_1	۱	۲
۱	۰.۹۳۳۷	۰.۹۹
۲	۰.۰۶۶۳	۰.۰۱

meldug_1\foto_1	۱	۲	۳
۱	۹۲۹۶	۲۰	۲۱
۲	۶۵۸	۳	۲

⇨

meldug_1\foto_1	۱	۲	۳
۱	۰.۹۳۴	۰.۸۷	۰.۹۱۳
۲	۰.۰۶۶	۰.۱۳	۰.۰۸۷

meldug_1\straaling_1	۱	۲
۱	۱	۹۳۳۶
۲	۰	۶۶۳

⇨

meldug_1\straaling_1	۱	۲
۱	۰.۹۹	۰.۹۳۳۷
۲	۰.۰۱	۰.۰۶۶۳

meldug_1\temp_1	۱	۲
۱	۹۲۹۲	۴۵
۲	۶۵۸	۵

⇨

meldug_1\temp_1	۱	۲
۱	۰.۹۳۳۹	۰.۹
۲	۰.۰۶۶۱	۰.۱

همان‌گونه که در جدول ۱ نشان داده شده است، برای متغیر *meldug_۱*، فراوانی‌ها و احتمالات آن برای چهار متغیر دیگر محاسبه شده است. همین روند برای همهٔ جفت متغیرها محاسبه می‌شود. شایان ذکر است که به‌دلیل جلوگیری از خطای صفر به توان صفر برای حالتی که فراوانی و در نتیجه احتمال نظیر آن صفر است، در جداول توافقی، هرگاه احتمالی برابر با صفر بود، ۰.۰۱ در نظر گرفته می‌شود.

برای دو متغیر *meldug_۱* (به‌عنوان فرزند) و *foto_۱* (به‌عنوان والد) که فراوانی‌ها و احتمالات آن‌ها در جدول ۱ نیز آمده است، داریم:

$$Dir(۰/۹۳۳۹, ۰/۶۶۱; ۹۲۹۶, ۶۵۸) = -۴/۱۲۹۵$$

$$Dir(۰/۹۳۳۹, ۰/۶۶۱; ۲۰, ۳) = -۲/۰۳۸۱$$

$$\underbrace{Dir(۰/۹۳۳۹, ۰/۶۶۱; ۲۱, ۲) = -۱/۳۳۵۹}_{\max - \min = R_۱ - C_۱ = ۲/۷۹۳۶}$$

$$Dir(۰/۸۶۰۶, ۰/۱۳۰۴; ۹۲۹۶, ۶۵۸) = -۲۲۰/۳۹۵۴$$

$$Dir(۰/۸۶۹۶, ۰/۱۳۰۴; ۲۰, ۳) = -۱/۴۲۶۶$$

$$\underbrace{Dir(۰/۸۶۹۶, ۰/۱۳۰۴; ۲۱, ۲) = -۱/۴۷۵۴}_{\max - \min = R_۲ - C_۲ = ۲۱۸/۹۶۸۸}$$

$$Dir(۰/۹۱۳۰, ۰/۰۸۷۰; ۹۲۹۶, ۶۵۸) = -۳۳/۶۳۸۱$$

$$Dir(۰/۹۱۳۰, ۰/۰۸۷۰; ۲۰, ۳) = -۱/۶۶۷۲$$

$$\underbrace{Dir(۰/۹۱۳۰, ۰/۰۸۷۰; ۲۱, ۲) = -۱/۲۶۱۷}_{\max - \min = R_۳ - C_۳ = ۳۲/۳۷۶۴}$$

$$score(foto\_۱, meldug\_۱)$$

$$= ۲/۷۹۳۶ + ۲۱۸/۹۶۸۸ + ۳۲/۳۷۶۴ = ۲۵۴/۱۳۱۸$$

امتیاز میان هر دو رأس، همانند روش فوق‌الذکر محاسبه و در جدولی مانند جدول ۲ ذخیره می‌شود. در جدول مربوط، علامت ★ نشان‌دهندهٔ والدین احتمالی است.

پس از تکمیل جدول، اگر $score(A,B) < score(B,A)$ برقرار باشد، B به‌عنوان والد احتمالی A انتخاب می‌شود. بنا بر این با توجه به جدول ۲، والدین احتمالی متغیر *meldug_۱* عبارت‌اند از:

$$\{dm\_۱, foto\_۱, straaling\_۱, temp\_۱, lai\_۲, dm\_۳,$$

$$meldug\_۲, foto\_۳, straaling\_۳, temp\_۳\}$$

جدول ۲. انتخاب والد احتمالی رأس ۱_*meldug* از جدول امتیازات

	dm_۱	foto_۱	straaling_۱	temp_۱	lai_۱	meldug_۱	lai_۰	dm_۲	foto_۲	straaling_۲	temp_۲	lai_۲	meldug_۲	dm_۳	foto_۳	straaling_۳	temp_۳	lai_۳
dm_۱	۰	۱۳۵.۰۱۶	۹۶۸.۰۷۶۹	۱۳۵۲.۰۸	۷۷۶۴.۱۴۷	۷۱۵.۶۴۰۷ ★	۱۰۵۳۹.۶۴	۱۴۶۵.۰۸	۱۵۰۲۲.۴۳	۷۶۱۷.۰۲۸	۷۸۶۲.۹۱۶	۶.۷۴۵۳۰۳	۹۶۸.۰۷۶۹	۱۰۷۴۴۵	۷۸.۵۴۷۱۱	۷۱.۴۵۷۷۲	۹۳.۱۹۲۵۱	۱۰۶۸۸.۱
foto_۱	۱۳۵.۰۱۶	۰	۱۹۱.۳۹۶۲	۹۱۷۵۳.۴۳	۵۸۲.۰۹۲۸	۲۵۴.۱۳۸۸ ★	۸۰۷۹۴.۱۸	۱۰۱.۱۷۵۸	۱۶۶۲۷.۸۹	۶۸۲۶۹.۳	۳۸۶۲.۲۸	۹.۷۵۹۴۷۹	۱۹۱.۳۹۶۲	۲۰۹۱۲.۹۷	۱۵۴.۰۱۳	۱۳۹.۶۸۰۴	۱۸۳.۸۶۶	۲۰۸۰.۱۵
straaling_۱	۹۶۸.۰۷۶۹	۱۹۱.۳۹۶۲	۰	۴۵۵۱۷.۷۶	۲۶۱۹.۸۳	۷۱۵.۷۰۱۱ ★	۴۰۲۴۹.۶۳	۱۶۲۳۲.۴۵	۱۵۰۲۲.۴۹	۳۴۰۸۷.۲۱	۳۴۶۴.۲۹	۶۸۰۵۶۱۶	۹۶۸۶۷۹۹	۴۰۶۴۶.۲۴	۷۸.۶۰۷۴۲	۷۱.۵۱۸.۰۳	۹۳.۲۵۲۸۲	۴۰۵۳۷.۶
straaling_۱	۸۶.۸۳۳۷۱	۱۳.۴۶۸۹۷	۰	۲۹۸۸۴۲۲	۲۲۳.۸۰۹۲	۲۵۶۳۳۱۳ ★	۴۰۵۱۰.۲	۵.۳۴۱۱۴۳	۴۶۰۶۷۰۹	۳۴۲۶۰.۱۵	۱۹۲۷۵.۲۶	۵۸۱۹۱۵۶	۹۵۸۸۳۰۴	۱۰۵۲۳.۸۹	۷۷.۶۲۲۲۹	۷۰.۵۲۲۹۶	۹۲.۲۶۸۸۶	۱۰۴۶۶.
temp_۱	۶۲۴.۴۷۸۹	۲۵۱۹۷.۰۵	۲۹۸۸۴۲۲	۰	۲۲۳.۸۰۹۲	۷۵۶۳۳۱۳ ★	۴۰۵۱۰.۲	۰.۸۵۰۰۷۵	۹۹.۷۴۸۱۳	۸۸۱.۳۰۸۳	۸۱۵.۵۶۸۸	۶.۳۲۰.۸۴۲	۴۵۰.۷۱۷۶	۱۲۱۱.۲۴۲	۰.۸۷۳۰.۸۳	۳.۸۴۱۳۸۵	۰.۷۶۷۵۴۴	۱۲۷۸.۱۶
lai_۱	۲۶.۵۴۰.۵۴	۶.۲۲۴۸۷۲	۴۵۰.۷۱۷۶	۵۲۰.۲۷۰۶	۲۴۸۳.۳۳۱	۲۲۸۱.۶۴۱	۲۲۸۱.۶۴۱	۰.۸۵۰۰۷۵	۹۹.۷۴۸۱۳	۸۸۱.۳۰۸۳	۸۱۵.۵۶۸۸	۶.۳۲۰.۸۴۲	۴۵۰.۷۱۷۶	۱۲۱۱.۲۴۲	۰.۸۷۳۰.۸۳	۳.۸۴۱۳۸۵	۰.۷۶۷۵۴۴	۱۲۷۸.۱۶
meldug_۱	۷۳.۵۱۷۴	۱۱.۷۲۳۶۷	۸۳.۵۵۸۹۳	۷۰.۲۵۱۱۹	۱۰۹۲۱.۵۵	۰	۴۶۰۳۷.۲۲	۴.۶۳۳۹۳۳	۴۳۳۳.۲۷۹	۳۶۸۹۹.۳	۱۷۵۳۵	۹.۲۱۹۵۳۸	۸۳.۵۵۸۹۳	۱۵۰۰۸.۱۴	۱۶.۳۶۵۷۷	۱۱.۲۰.۵۳۳	۷۹.۹۴۱۳۴	۱۴۸۷۳.۶
lai_۰	۴۲.۸۸۰۷۴	۶۹۱.۳۳۷۴	۱۰۰.۳۳۷	۷۱۲۰.۸۶۸	۹۰۱۴.۱۴۲	۲۸۳۹۹.۴۷	۰	۴.۶۳۳۷۵۶	۴۳۳۳.۱۵۶	۳۷۱۳۲.۹۷	۱۷۹۰۷۰۹	۱۱.۵۲۲۲۴	۸۲.۶۵۴۰۴	۱۵۲۶۹.۵۳	۱۴.۶۸۱۲۴	۹.۷۴۸۴۴۶	۷۹۰.۲۶۱	۱۵۱۱۰.
dm_۲	۲.۲۲۵۳۲۹	۰.۸۴۸۶۵	۴.۲۶۷۴۰۴	۰.۰۰۰۱۶۲	۰.۰۰۶۷۳۱	۰.۵۶۷۶۰۲	۰.۶۳۱۰۶۷	۰	۳۵۴۲۷.۱۹	۰.۳۵۳۹۹۱	۱.۵۴۱۱۴۳	۰.۷۵۹۶۴۸	۴.۲۶۷۴۰۴	۰.۵۷۲۷۲۹	۰.۳۰۱۶۵۴	۰.۰۰۰۲۳۱	۷.۴۹۶۰۶۶	۰.۲۱۹۴۹
foto_۲	۹.۶۸۷۸۱۲	۴۸۵۰۵۱۲	۱.۱۲۹۰۸۸	۸۰۰۴۴۰۵۵	۸۳۸۵۶۲۵	۹۷۱.۹۳۱۳	۱۰۵۶۵۶۵	۳۶۰۱۸.۹۴	۰	۳۲۴.۰۲۴	۲۲۱.۶۹۷	۰.۴۰۳۳۳۱	۱.۱۲۹۰۸۸	۵۸۲.۰۹۱۸	۰.۲۸۳۷۲۹	۰.۰۴۹۷۷۴	۱۲.۹۵۷۱۵	۵۴۵.۰۳۳
straaling_۲	۱۷.۰۰۸۳۱	۳۲۶.۳۶۸۴	۷۴.۵۳۷۸۹	۳۱۵.۳۷۹	۱۳۷۲.۷۶۶	۶۲۲۰.۶۵۳	۶۸۰۸.۷۷۷	۲۰۶۶۵۷۸	۵۹۵.۸۴۰۸	۰	۳۱۸۲.۲۸	۱.۶۲۰۴۸۷	۶۳.۸۶۴۲	۴۵۷۷.۰۹۳	۱۵.۹۷۷۸۴	۱.۵۳۱۱۹۶	۶۰.۳۵۴۲۷	۴۴۷۶.۹۱
temp_۲	۱۸.۱۸۵۶۳	۳۷۰.۸۵۳۸	۷۶.۶۲۲۷۹	۴۰۷.۷۴۶۱	۱۳۳۱.۳۴۳	۵۶۳۹.۲۱۴	۶۱۵۸.۸۳۱	۳.۷۳۳۴۶۸	۴۲۵.۶۴۳۶	۳۳۴۹.۵۱	۰	۱.۶۸۵۸۶۸	۷۶.۶۲۲۷۹	۳۸۳۹.۲۷۶	۲.۵۰۰۰۳	۱۰۳۸۶۷۲	۸۳.۸۴۰۸۴	۳۸۱۱.۰۹
lai_۲	۸۴.۶۸۲۶۹	۱۴.۶۲۴۶۱	۹۴.۷۱۷۱۶	۱۹.۰۴۳۰۱	۱۲۶.۷۱۳۵	۴۴.۹۹۸۸۵ ★	۶۳.۷۷۵۵۲	۲۱.۹۱۸۳۵	۱۳.۷۳۱۳۱	۴۸۳۳۰.۲۳	۴.۷۷۰.۶۷۴	۰	۹۴.۷۱۷۱۶	۴۱۲۵۲.۴۸	۷۶.۴۵۶۲۲	۶۹.۳۶۶۶۵	۹۱.۱۰.۱۶۲	۴۱۱۳۸.۲
meldug_۲	۸۶.۸۳۳۷۱	۱۳.۴۶۸۶۷	۹۶.۸۶۷۹۹	۲۱.۱۹۶۸۱	۲۶۱۹۹.۸۳	۷۱۵.۷۰۱۱ ★	۸۰۵.۱۲۱۲	۱۶۲۳۲.۴۵	۱۵۰۲۲.۴۹	۳۱۷۱.۲۴۳	۳۴۶۴.۲۹	۶.۸۰۵۶۱۶	۰	۴۰۶۴۶.۲۴	۷۸.۶۰۷۴۲	۷۱.۵۱۸.۰۳	۹۳.۲۵۲۸۲	۴۰۵۳۷.۶
dm_۳	۴۶.۶۲۲۳۴	۵۱۸.۵۶۵۵	۱۰۲.۵۸۹۸	۵۸۵.۲۶۱۶	۵۱۱۲.۲۵۶	۱۵۴۵۴.۴۲ ★	۱۸۰۵۴.۳۱	۴۷۰۰۴۰۲	۲۵۲۸.۱۲۸	۱۴۰۹۴.۷۷	۱۰۲۸۸.۹۷	۱۸۰۵.۸۸	۱۰۲.۵۸۹۸	۰	۱۶.۷۷۴۲۷	۱۱.۵۶۱۵۴	۸۰.۱۲۲۵	۴۷۰۵۹.۴
foto_۳	۸۶.۷۱۲۱	۱۳.۵۳۴۲۱	۹۶.۷۴۷۳۸	۲۱۰.۷۶۱۹	۱۱.۷۴۵۲۳	۱۵۲۸.۵۸۵ ★	۱۴۶۲.۴۵۸	۱۱۵.۲۵۶	۵۱.۰۷۷۶۹	۳۵۵۳.۰۱۹	۴۸۱.۱۹۱۷	۶.۶۸۴۱۷۹	۹۶.۷۴۷۳۸	۱۵۵۵.۹۳۹	۰	۷۱.۳۹۷۴۲	۹۳.۱۳۲۲۱	۴۰۵۶۷.۱
straaling_۳	۸۶.۶۵۲۸	۱۳.۵۶۶۸۱	۹۶.۶۸۷۰۸	۲۱۰.۱۵۸۸	۸۳۴.۶۰۱۲	۷۱۶.۸۰۳ ★	۶۶۱.۸۲۶۴	۶۹.۱۴۶۷۶	۱۹.۲۵۹۷۲	۵۰.۱۶۰.۷۲	۱۴۳۲.۹۹۶	۶.۶۲۴۶۴۶	۹۶.۶۸۷۰۸	۷۲۹.۴۵۲۷	۷۸.۴۲۶۵۱	۰	۹۳۰.۷۱۹۱	۴۰۵۸۳.۴
temp_۳	۸۶.۸۱۳۶۱	۱۳.۴۷۹۸۵	۹۶.۸۴۷۸۹	۲۱.۱۷۶۷۱	۸۳۶.۰۲۱	۷۱۵.۶۸۱ ★	۸۰۵.۱۰۱۱	۱۶۰۷۶.۱۹	۱۷۳۱۰.۸۴	۳۱۷۱.۲۲۳	۳۴۶۴.۲۶	۶.۷۸۵۵۱۳	۹۶.۸۴۷۸۹	۶۹۶.۲۴۶۵	۷۸.۵۸۷۳۲	۷۱.۴۹۷۹۳	۴۰۵۴۰.۱	
lai_۳	۴۵۵۴۹.۱۹	۵۰۶.۱۷۲۹	۱۰.۱۹۶۶	۵۷۱.۲۴۰۲	۴۸۴۳.۷۹	۱۴۶۶۸.۴۷	۱۶۹۷۵.۷۳	۳.۳۶۸۹۵۴	۲۳۰۴.۹۲۵	۹۸۷۲.۵۱۱	۱۷۵۹.۲۵۷	۱۰۱.۹۶۶	۳۸۲۶۸.۲۶	۱۸۸.۲۹۱۵	۲۲۹.۶۸۱۵	۱۱۶.۸۷۶۵	۰	

در ادامه، الگوریتم $K2$ امتیاز میان هر رأس و اعضای مجموعهٔ والدین احتمالی آن محاسبه می‌شود و در صورتی که امتیاز زیادی به دست آید، آن متغیر به‌عنوان والد متغیر مذکور در نظر گرفته شده، یالی بین آن‌ها ایجاد می‌شود.

۴ روش پیشنهادی

یکی از مشکلات روش کو و کیم، ایجاد یال‌های اضافی است. بنا بر این برای رفع این مشکل، در ابتدا پوشش مارکوفی هر رأس را یافته، سپس والدین احتمالی از بین آن‌ها انتخاب می‌شوند.

۱.۴ استقلال شرطی

اگر X و Y و Z سه متغیر تصادفی با مقادیر گسسته باشند، می‌گوییم X و Y به‌شرط دانستن Z مستقل‌اند اگر و تنها اگر:

$$(\forall\, x_i,\, y_j,\, z_k) \quad P\left(X = x_i |\, Y = y_j,\, Z = z_k\right) = P\left(X = x_i | Z = z_k\right)$$

مهم‌ترین آزمون آماری که از آن برای ارزیابی مستقل بودن رأس‌ها استفاده می‌شود، آزمون آماری χ^2 است که به‌صورت زیر فرمول‌بندی می‌شود:

$$\chi^2 = \sum \frac{(O-E)^2}{E}$$

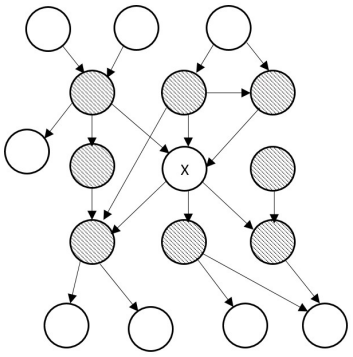
که در آن E تعداد دفعاتی است که انتظار داشتیم یک اتفاق رخ دهد و O تعداد دفعاتی است که این اتفاق رخ داده است.

۲.۴ پوشش مارکوفی

پوشش مارکوفی هر رأس در شبکه‌های بیزی، دربردارندهٔ همسایه‌های آن رأس می‌باشد. به عبارت دیگر می‌توان گفت، پوشش مارکوفی هر رأس، مجموعه‌ای از رئوس است که رأس مذکور را از سایر رئوس شبکه جدا می‌سازد و همهٔ اطلاع لازم برای پیش‌بینی رفتار هر رأس را می‌توان با کمک مجموعهٔ اعضای پوشش مارکوفی آن به دست آورد، که این اطلاع قابل استخراج از سایر متغیرها نمی‌باشد (پیرل [۹]). پوشش مارکوفی رأس X به‌صورت $MB(X)$ نشان داده می‌شود. اگر V مجموعهٔ رئوس موجود در مجموعه‌داده‌ها باشد، داریم:

$$\forall\, X, Y \in V \;:\; (X \in MB\left(Y\right) \Leftrightarrow X \not\perp Y \mid V \setminus \{X, Y\})$$

یعنی به‌ازای هر رأس X و Y که عضو مجموعهٔ رئوس باشند، X عضو مجموعهٔ پوشش مارکوفی رأس Y است اگر و تنها اگر دو متغیر X و Y به‌شرط حداقل یکی از اعضای مجموعهٔ V (به‌جز خود X و Y) به هم وابسته باشند.



شکل۴. پوشش مارکوفی رأس X

در واقع پوشش مارکوفی هر رأس شامل والد، فرزند و دیگر والد فرزند (همسر) است. شکل ۴، پوشش مارکوفی رأس X را نشان می‌دهد. الگوریتم‌های مختلفی برای یافتن پوشش مارکوفی رئوس پیشنهاد شده که در این مقاله از روش افزایشی-کاهشی برای یافتن پوشش مارکوفی استفاده شده است. این الگوریتم، یکی از روش‌هایی است که با استفاده از آزمون‌های استقلال شرطی، وابستگی یا عدم وابستگی رئوس را تشخیص می‌دهد و بر این اساس، ساختار محلی در اطراف هر رأس به دست می‌آید. این الگوریتم دارای دو مرحلهٔ افزایش و کاهش است. برای یافتن همسایه‌های هر رأس مانند X، نخست یک مجموعهٔ بدون عضو مانند S در نظر گرفته و در مرحلهٔ افزایش، متغیری را که به‌شرط مجموعهٔ S وابسته به X است، به S افزوده و همین روند برای تمامی متغیرهای دیگر به‌شرط وجود مجموعهٔ S بررسی می‌شود. در این مرحله ممکن است رئوسی به مجموعهٔ S اضافه شده باشند که در همسایگی رأس X قرار ندارند. برای مشخص شدن این رئوس، مرحلهٔ کاهش اجرا شده، اگر رأسی مانند Y در مجموعهٔ S وجود داشته باشد که $X \perp Y | S - \{Y\}$ باشد، Y از مجموعهٔ S حذف می‌شود ($\perp$ نشان‌دهندهٔ استقلال است). این مرحله نیز به‌ازای همهٔ اعضای مجموعهٔ S اجرا شده و در نهایت، رئوس موجود در S نشان‌دهندهٔ همسایه‌های (پوشش مارکوفی) رأس X می‌باشند (مارگاریتیس و ترون [۷]). بررسی استقلال شرطی رئوس با استفاده از آزمون آماری χ^2 صورت می‌گیرد که پیش‌تر توضیح داده شد.

[۱۲] grow shrink

الگوریتم GS[۱۲] برای یافتن اعضای پوشش مارکوفی رئوس

- ورودی و خروجی الگوریتم

ورودی: مجموعه‌داده.

خروجی: اعضای پوشش مارکوفی هر گره.

- شروع الگوریتم

مجموعهٔ S را تهی در نظر بگیر؛ یعنی $S = \emptyset$.

مرحلهٔ افزایش:

به‌ازای همهٔ متغیرهای موجود در مجموعه‌داده، اگر $X \not\perp Y|S$ برقرار باشد:

$S \leftarrow S \cup \{Y\}$

مرحلهٔ کاهش:

تا زمانی که رأسی مانند Y در مجموعهٔ S وجود داشته باشد $X|S - \{Y\}$ برقرار باشد. در این صورت:

$S \leftarrow S - \{Y\}$

مجموعهٔ باقی‌مانده، اعضای پوشش مارکوفی رأس X می‌باشند.

پایان الگوریتمGS

در الگوریتم پیشنهادی نیز امتیازات بین هر دو متغیر به‌شیوهٔ روش کو و کیم محاسبه می‌شود؛ با این تفاوت که امتیاز، فقط بین هر متغیر و متغیرهای موجود در پوشش مارکوفی آن متغیر محاسبه می‌گردد (جدول ۳).

به‌عنوان مثال، پوشش مارکوفی رأس $meldug\_۱$، مجموعهٔ $temp\_۱, lai\_۱, lai\_۰$ است. بنا بر این داریم:

$$score(meldug\_۱, temp\_۱) < score(temp\_۱, meldug\_۱)$$

$$score(meldug\_۱, lai\_۱) > score(lai\_۱, meldug\_۱)$$

$$score(meldug\_۱, lai\_۰) > score(lai\_۰, meldug\_۱)$$

بنا بر این فقط متغیر $temp\_۱$ به‌عنوان والد احتمالی $meldug\_۱$ انتخاب می‌شود. محاسبات در الگوریتم $K۲$ برای این متغیر به‌صورت زیر انجام می‌شود:

امتیاز	اضافه شدن به $meldug\_۱$
-۲۴۴۴.۴۴۳۹	— — —
-۲۴۴۶.۶۴۱۸	$temp\_۱$

نتیجه: هیچ رأس دیگری به‌عنوان والد به $meldug\_۱$ افزوده نمی‌شود.

به‌دلیل این‌که امتیاز متغیر $meldug\_۱$ بدون داشتن والد، بیشتر از امتیاز حاصل از افزودن $temp\_۱$ به‌عنوان والد است، متغیر $meldug\_۱$ هیچ والدی ندارد. در مقایسه با روش کو و کیم، چهار یال اضافی که برای متغیر $meldug\_۱$ تولید شده بودند، در الگوریتم پیشنهادی به شبکه افزوده نمی‌شوند. بنا بر این خطای الگوریتم پیشنهادی کمتر از الگوریتم کو و کیم است.

۵ نتیجه‌گیری

در این بخش، نتایج حاصل از اجرای چهار الگوریتم با عناوین الگوریتم پیشنهادی، کو و کیم، چن و همکاران، و هروشکا و ایبکن، بر روی مجموعه‌داده‌های $Asia$، $Alarm$، Car، $Hail$، $Midway$ که به‌ترتیب شامل ۸، ۳۷، ۱۸، ۵۶ و ۲۶ متغیر و ۸، ۴۶، ۱۷، ۶۶ و ۳۸ یال می‌باشند، ارائه‌شده است. برای ارزیابی، نخست شبکه‌های اصلی این مجموعه‌داده‌ها در نظر گرفته شده، سپس تعداد یال‌های انطباقی با جهت یکسان، تعداد یال‌های انطباقی با جهت معکوس، تعداد یال‌های اضافی و تعداد یال‌های گم‌شده برای هر چهار الگوریتم محاسبه شده است. میانگین نتایج حاصل از اعمال هر چهار الگوریتم بر نمونه‌های ۱۰۰۰۰تایی، در شکل ۴ نشان داده شده است.

شایان ذکر است که مجموعه‌داده‌های ذکر شده همراه با شبکه‌های اصلی آن‌ها در بستهٔ نرم‌افزاری $bnlearn$ در نرم‌افزار R موجود است.

با توجه به شکل ۴، الگوریتم پیشنهادی توانسته یکی از مشکلات روش کو و کیم را حل کند، که ایجاد یال‌های اضافی بود. با این حال در دو مجموعه‌دادهٔ $midway$ و $hail$ تعداد یال‌های صحیح نسبت به الگوریتم کو و کیم کاهش و تعداد یال‌های گم‌شده افزایش یافته است. اما همان‌طور که پیش‌تر نیز بیان شد چون هدف شبکه‌های بیزی رسیدن به گرافی است که در شناسایی ارتباطات میان متغیرها عملکرد دقیق‌تری داشته باشد، به این معنا که اگر دو متغیر با هم در ارتباط نباشند هیچ یال اضافی یا حتی معکوس میان آن دو وجود نداشته باشد، به‌منظور ارزیابی دقت الگوریتم‌ها، خطای شبکهٔ برابر با مجموع یال‌های معکوس، اضافی و گم‌شده تعریف می‌شود و روشی دقیق‌تر است که بتواند در مجموع، تعداد این یال‌ها را کاهش دهد. همان‌طور که در شکل ۴ نیز مشخص است، خطای الگوریتم پیشنهادی در مقایسه با سایر روش‌ها به‌میزان قابل توجهی کاهش یافته است.

همچنین به‌منظور ارزیابی میزان دقت هر یک از الگوریتم‌های فوق، از رابطهٔ (۳) به‌صورت زیر استفاده شده است. رابطهٔ (۳) در واقع بیان‌کنندهٔ نسبت تعداد یال‌های انطباقی با جهت یکسان به کل یال‌هاست.

$$accuracy = \frac{\#correct\ edge}{\#correct\ edge + \#graph\ error} \qquad (۳)$$

نتایج حاصل از محاسبهٔ میزان دقت هر یک از روش‌ها با استفاده از رابطهٔ (۳)، در شکل ۵ نشان داده شده است. همان‌طور که مشخص است، نتایج، نشان‌دهندهٔ دقت زیاد الگوریتم پیشنهادی است.

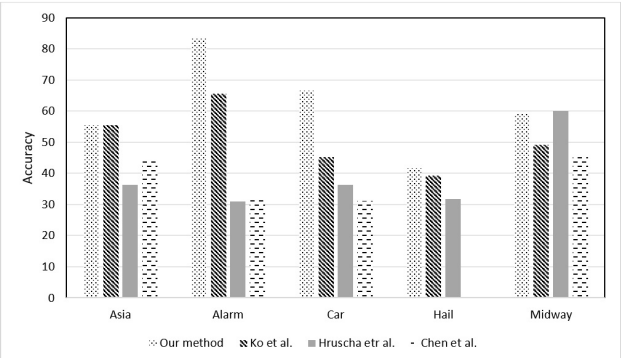
در این مقاله، روش پیشنهادی کو و کیم برای تعیین ترتیب بین متغیرها، به‌عنوان ورودی الگوریتم $K2$ تعمیم داده شد. به این منظور برای هر رأس، در ابتدا اعضای پوشش مارکوفی آن رأس با استفاده از الگوریتم افزایشی-کاهشی تعیین شده، سپس والدین احتمالی رأس مذکور، از میان اعضای پوشش مارکوفی و با توجه به امتیاز میان هر جفت متغیر (رأس)، به دست می‌آید. در نهایت الگوریتم $K2$ به کار رفته، ساختار شبکهٔ بیزی حاصل می‌شود. نتایج حاصل از ارزیابی میزان دقت الگوریتم‌ها نشان می‌دهد که الگوریتم پیشنهادی با دقت بیشتری قادر به تشخیص ارتباطات صحیح بین متغیرها در مقایسه با شبکه‌های اصلی است.

جدول ۳. انتخاب والد احتمالی رأس ۱_$meldug$ از جدول امتیازات پس از اعمال پوشش مارکوفی

	dm_۱	foto_۱	straaling_۱	temp_۱	lai_۱	meldug_۱	lai_۰	dm_۲	foto_۲	straaling_۲	temp_۲	lai_۲	meldug_۲	dm_۳	foto_۳	straaling_۳	temp_۳	lai_۳
dm_۱	۰	۱۳۵۰۱۶	۹۶۸۰۷۶۹	۱۳۵۲۰۰۸	۷۷۶۴۰۱۴۷	۷۱۵۰۶۴۰۷	۱۰۵۳۶۰۶۴	۱۴۶۵۰۰۸	۱۵۰۲۲۰۴۳	۷۶۱۷۰۲۸	۷۸۶۲۰۹۱۶	۶۰۷۴۵۳۰۳	۹۶۸۰۷۶۹	۱۰۷۴۴۵	۷۸۰۵۴۷۱۱	۷۱۰۴۵۷۷۲	۹۳۰۱۹۲۵۱	۱۰۶۴۸۰۱
foto_۱	۱۳۰۵۰۱۶	۰	۱۹۱۰۳۹۶۲	۹۱۷۵۳۰۴۳	۵۸۲۰۰۹۲۸	۲۵۴۰۱۳۸۸	۸۰۷۹۴۰۱۸	۱۰۱۰۱۷۵۸	۱۶۶۶۷۰۸۹	۶۸۳۶۰۳	۳۸۹۲۴۰۲۸	۹۰۷۵۹۴۰۹	۱۹۰۱۰۳۹۶۲	۲۰۹۱۲۰۹۷	۱۵۴۰۰۱۳	۱۳۹۰۶۸۰۴	۱۸۳۰۸۶۶	۲۰۸۰۱۵
straaling_۱	۱۳۰۴۶۸۹۷	۲۵۱۹۷۰۵	۲۶۸۰۸۴۴۲	۴۵۵۱۷۰۷۶	۲۶۱۹۰۸۳	۷۱۵۰۷۰۱۱	۴۰۲۴۹۰۶۳	۱۶۲۳۲۰۴۵	۱۵۰۲۲۰۴۹	۳۴۰۸۷۰۲۱	۳۴۶۴۴۰۲۹	۶۸۰۵۶۱۶	۹۶۸۶۷۹۹	۴۰۶۴۶۰۲۴	۷۸۰۶۰۷۴۳	۷۱۰۵۱۸۰۳	۹۳۰۲۵۲۸۲	۴۰۵۳۷۰۶
temp_۱	۶۲۴۰۴۷۸۹	۲۵۱۹۷۰۵	۲۶۸۰۸۴۴۲	۰	۲۳۳۸۰۹۲	۷۵۰۶۳۳۱۳ ★	۴۰۵۱۰۲	۵۰۳۴۱۱۴۳	۴۶۰۶۰۷۰۹	۳۴۲۶۰۰۱۵	۱۹۷۳۵۰۲۶	۵۸۱۹۲۵۶	۹۵۸۸۳۰۴	۱۰۵۲۳۰۸۹	۷۷۰۶۲۳۳۹	۷۰۰۵۳۲۹۶	۹۲۰۲۶۷۸۶	۱۰۴۶۶۰
lai_۱	۲۶۰۵۴۰۰۵۴	۶۰۲۲۴۸۷۲	۸۳۰۵۵۸۱۳	۵۲۰۲۷۰۶	۰	۲۴۸۳۰۳۲۱	۲۲۸۱۰۶۴۱	۰۸۵۰۰۷۵	۹۹۰۷۴۸۱۳	۸۸۱۰۳۰۸۳	۸۱۵۰۶۸۸	۶۰۲۳۰۸۴۲	۴۵۰۰۷۱۷۶	۱۳۱۱۰۲۴۲	۰۸۷۳۰۸۳	۳۰۸۳۱۳۸۵	۰۷۶۷۵۴۴	۱۲۷۸۰۱۶
meldug_۱	۷۳۰۵۱۷۴	۱۱۰۷۲۲۶۷	۸۳۰۵۵۸۱۳	۷۰۰۳۵۱۱۹	۱۰۹۴۱۰۵۵	۰	۴۶۰۳۷۰۲۲	۴۶۰۳۲۹۳۳	۴۳۳۲۰۲۷۹	۳۶۸۹۹۰۳	۱۷۵۳۵	۹۰۲۱۹۵۲۸	۸۳۰۵۵۸۹۳	۱۵۰۰۸۰۱۴	۱۶۰۳۶۵۷۷	۱۱۰۲۰۵۳۳	۷۹۰۹۴۱۳۴	۱۴۸۷۳۰۶
lai_۰	۴۲۰۸۲۰۷۴	۶۹۰۲۳۷۴	۱۰۰۳۷۳۷	۷۱۲۰۸۶۸	۹۰۱۴۰۱۴۲	۲۸۳۹۹۰۴۷	۰	۴۰۶۳۲۷۵۶	۴۳۳۰۱۵۶	۳۷۱۳۲۰۹۷	۱۷۹۰۷۰۰۹	۱۱۰۵۲۲۲۴	۸۲۰۶۵۴۰۴	۱۵۲۶۹۰۵۳	۱۴۰۶۸۱۲۴	۹۰۲۴۸۴۴۶	۷۹۰۰۳۶۱	۱۵۱۱۰۰
dm_۲	۲۰۲۲۵۳۲۹	۰۰۸۴۸۶۵	۴۰۲۶۷۴۰۴	۰۰۰۱۱۶۲	۰۰۰۰۶۷۳۱	۰۰۵۶۷۶۰۲	۰۰۶۳۱۰۶۷	۰	۳۵۴۲۷۰۱۹	۰۳۵۳۶۹۱	۱۰۵۴۱۱۴۳	۰۰۷۵۹۶۴۸	۴۰۲۶۷۴۰۴	۰۰۵۷۲۷۲۹	۰۰۳۰۱۶۵۴	۰۰۰۰۲۳۱	۷۰۴۹۶۰۶۶	۰۰۲۱۹۴۹
foto_۲	۹۰۶۸۷۸۱۲	۴۸۰۵۰۵۱۲	۱۰۱۲۹۰۸۸	۸۰۰۴۴۰۵۵	۸۳۰۸۵۶۲۵	۹۷۱۰۹۳۱۳	۱۰۵۶۵۶۵	۳۶۰۱۸۰۹۴	۰	۳۲۴۰۰۲۴	۲۲۱۰۶۹۷	۰۴۰۳۲۳۱	۱۰۱۲۹۰۷۸	۵۸۲۰۰۹۱۸	۰۰۲۸۳۳۹	۰۰۰۴۹۷۷۴	۱۲۰۹۵۷۱۵	۵۴۵۰۲۳
straaling_۲	۱۷۰۰۰۸۳۱	۳۲۶۰۳۶۸۴	۷۴۰۵۳۷۸۹	۳۱۵۰۳۷۹	۱۳۷۳۰۷۶۶	۶۲۲۰۰۶۵۳	۶۸۰۸۰۷۷۷	۲۰۰۶۵۸۷۸	۵۹۵۰۴۴۰۸	۰	۳۱۸۲۰۲۸	۱۰۶۳۰۴۸۷	۶۳۰۹۸۶۴۲	۴۵۷۷۰۰۹۳	۱۵۰۹۷۷۸۴	۱۰۵۳۱۱۹۶	۶۰۰۳۵۴۲۷	۴۴۷۰۶۹۱۱
temp_۲	۱۸۰۱۸۵۶۲	۳۷۰۰۸۵۳۸	۷۶۰۶۲۲۷۹	۴۰۷۰۷۴۶۱	۱۳۳۱۰۳۴۳	۵۶۳۹۰۲۱۴	۶۱۵۸۰۸۳۱	۳۰۷۳۴۶۸	۴۲۵۰۶۴۳۶	۳۳۴۹۰۵۱	۰	۱۰۶۸۵۸۶۸	۷۶۰۶۲۲۷۹	۳۸۴۹۰۲۷۶	۲۰۵۰۰۰۳	۱۰۰۳۸۶۷۳	۸۳۰۸۴۰۸۴	۳۸۱۱۰۰۹
lai_۲	۸۴۰۶۸۲۶۹	۱۴۰۶۲۳۹۱	۹۴۰۷۱۷۱۶	۱۹۰۰۴۳۰۱	۱۲۶۰۷۱۳۵	۴۴۰۹۹۸۸۵	۶۳۰۷۷۵۵۲	۲۱۰۹۱۸۳۵	۱۳۰۷۳۱۳۱	۴۰۸۳۳۰۲۳	۴۰۷۷۰۶۷۴	۰	۹۴۰۷۱۷۱۶	۴۱۵۲۰۴۸	۷۶۰۴۵۶۲۲	۶۹۰۳۶۶۶۵	۹۱۰۱۰۱۹۲	۴۱۱۳۸۰۲
meldug_۲	۸۶۰۸۳۳۷۱	۱۳۰۴۶۸۹۷	۹۶۰۸۶۷۹۹	۲۱۰۱۹۶۸۱	۲۶۱۹۹۰۸۳	۷۱۵۰۷۰۱۱	۸۰۵۰۱۲۱۲	۱۶۲۲۲۰۴۵	۱۵۰۲۲۰۴۹	۳۱۷۱۰۲۴۳	۳۴۶۴۴۰۲۹	۶۰۸۰۵۶۱۶	۰	۴۰۶۴۶۰۲۴	۷۸۰۶۰۷۴۳	۷۱۰۵۱۸۰۳	۹۳۰۲۵۲۸۲	۴۰۵۳۷۰۶
dm_۳	۴۶۰۶۳۳۳۴	۵۱۸۰۵۴۵۵	۱۰۲۰۵۸۸۸	۵۸۵۰۳۶۱۶	۵۱۱۲۰۲۵۶	۱۵۵۴۰۴۲	۱۸۰۵۴۰۳۱	۴۷۰۰۴۰۲	۲۵۲۸۰۱۳۸	۱۴۰۹۴۰۷۷	۱۰۲۸۸۰۹۷	۱۸۰۵۰۸۸	۱۰۲۰۵۸۹۸	۰	۱۶۰۷۷۴۳۷	۱۱۰۵۶۱۵۴	۸۰۰۱۴۲۵	۴۷۰۵۹۰۴
foto_۳	۸۴۰۷۱۳۱	۱۳۰۵۳۴۲۱	۹۴۰۷۴۷۲۸	۲۱۰۷۰۷۶۱۹	۱۱۰۷۴۵۲۳	۱۵۲۸۰۵۸۵	۱۴۴۲۰۴۵۸	۱۱۵۰۲۵۴	۵۱۰۰۷۷۶۹	۳۵۵۳۰۱۹	۴۸۱۰۱۹۱۷	۶۰۶۸۴۹۷۹	۹۶۰۷۴۷۳۸	۱۵۵۵۰۹۳۹	۰	۷۱۰۳۹۷۴۲	۹۳۰۱۳۲۲۱	۴۰۵۴۷۰۱
straaling_۳	۸۴۰۶۵۲۸	۱۳۰۵۶۶۷۱	۹۶۰۸۷۰۷۸	۲۱۰۱۵۸۸	۸۳۴۰۶۰۱۲	۷۱۶۸۰۰۳	۶۶۱۸۲۶۴	۶۰۹۱۴۶۷۶	۱۹۰۲۵۹۷۲	۵۰۰۱۶۰۷۲	۱۴۳۰۹۹۶	۶۰۶۴۶۴۶	۹۶۰۸۷۰۷۸	۷۲۹۰۴۵۶۷	۷۸۰۸۴۶۵۱	۷۱۰۴۹۷۹۳	۹۳۰۷۱۹۱	۴۰۵۸۳۰۴
temp_۳	۸۶۰۸۱۳۶۱	۱۳۰۴۷۸۸۵	۹۶۰۸۴۷۸۹	۲۱۰۱۷۶۷۱	۷۱۵۰۶۸۱	۱۴۶۶۵۰۴۷	۱۶۹۷۵۰۷۳	۱۶۰۷۶۰۱۹	۱۷۳۱۰۰۸۴	۳۱۷۱۰۲۲۳	۳۴۶۴۰۲۶	۶۰۷۸۵۵۱۳	۹۶۰۸۴۷۸۹	۶۹۶۰۲۴۶۵	۷۸۰۵۸۷۳۳	۷۱۰۴۹۱۹۳	۰	۴۰۵۴۰۰۱
lai_۳	۴۵۰۵۴۹۱۹	۵۰۶۰۱۳۲۹	۱۰۱۰۹۶۶۶	۵۷۱۰۳۴۰۲	۴۸۴۰۳۷۹	۱۴۶۶۵۰۴۷	۱۶۹۷۵۰۷۳	۲۳۰۲۹۸۵۴	۲۳۰۴۰۹۲۵	۱۳۱۶۰۰۷۹	۹۸۷۲۰۵۱۱	۱۷۵۹۰۳۵۸	۱۰۱۰۹۶۶۶	۳۸۲۶۸۰۲۶	۱۸۸۰۲۹۱۵	۲۲۹۰۶۸۱۵	۱۱۶۰۸۷۶۵	۰

مجموعه داده	ملاک (تعداد یالها)	روش پیشنهادی	روش کو	روش هروشکا	روش چن
Asia	انطباقی با جهت درست	۵	۵	۴	۵
	گمشده	۰	۰	۱	۱/۱
	انطباقی با جهت معکوس	۳	۳	۳	۱/۹
	اضافی	۱	۱	۳	۳/۴
	خطا (جمع گمشده، معکوس و اضافی)	۴	۴	۷	۶,۳
Alarm	انطباقی با جهت درست	۴۰	۴۰	۲۲	۲۳,۹
	گمشده	۲	۲	۱	۲
	انطباقی با جهت معکوس	۴	۴	۲۳	۲۰,۱
	اضافی	۲	۱۵	۲۵	۳۰,۲
	خطا	۸	۲۱	۴۹	۵۲,۳
Car	انطباقی با جهت درست	۱۴	۱۴	۱۲	۱۱,۷
	گمشده	۱	۱	۱	۱,۷
	انطباقی با جهت معکوس	۲	۲	۴	۳,۹
	اضافی	۴	۱۴	۱۶	۲۰,۵
	خطا	۷	۱۷	۲۱	۲۵,۸
Hail	انطباقی با جهت درست	۳۰	۳۷	۳۲	-
	گمشده	۲۷	۱۹	۱۵	-
	انطباقی با جهت معکوس	۹	۱۰	۱۹	-
	اضافی	۶	۲۸	۳۵	-
	خطا	۴۲	۵۷	۶۹	-
Midway	انطباقی با جهت درست	۲۳	۲۷	۲۷	۲۳,۴
	گمشده	۱۰	۳	۳	۴,۴
	انطباقی با جهت معکوس	۵	۸	۸	۱۰,۲
	اضافی	۱	۱۷	۷	۱۳,۳
	خطا	۱۶	۲۸	۱۸	۲۷,۹

شکل ۵. نتایج حاصل از اعمال الگوریتم‌ها بر روی مجموعه‌داده‌های مختلف



شکل ۶. میزان دقت الگوریتم‌ها

مراجع

[1] Chen, X. W., Anantha, G., and Lin, X. (2008). Improving Bayesian network structure learning with mutual information-based node ordering in the K2 algorithm. *IEEE Transactions on Knowledge and Data Engineering*, **20(5)**, 628-640.

[2] Cooper, G. F. and Herskovits, E. (1992). A Bayesian method for the induction of probabilistic networks from data. Machine learning, **9(4)**, 309-347.

[3] Heckerman, D. (1998). *A tutorial on learning with Bayesian networks*. Springer Netherlands.

[4] Hruschka, E. R. and Ebecken, N. F. (2007). Towards efficient variables ordering for Bayesian networks classifier. *Data and Knowledge Engineering*, **63(2)**, 258-269.

[5] Ko, S. and Kim, D. W. (2014). An efficient node ordering method using the conditional frequency for the K2 algorithm. *Pattern Recognition Letters*, **40**, 80-87.

[6] Larrañaga, P., Poza, M., Yurramendi, Y., Murga, R. H., and Kuijpers, C. M. (1996). Structure learning of Bayesian networks by genetic algorithms: A performance analysis of control parameters. *IEEE transactions on pattern analysis and machine intelligence*, **18(9)**, 912-926.

[7] Margaritis, D. and Thrun, S. (1999). *Bayesian network induction via local neighborhoods*. Carnegie-Mellon University Pittsburgh Pa Department of Computer Science.

[8] Nielsen, T. D. and Jensen, F. V. (2009). *Bayesian networks and decision graphs*. Springer Science Business Media.

[9] Pearl, J. (1988). *Probabilistic reasoning in intelligent systems: Networks of plausible reasoning*. San Mateo, CA: Kaufmann.