عنوان: **فرآیند داده کاوی در Microsoft SQL Server (بخش یک)**
نویسنده: محمد رجبی
تاریخ: ۱۳۹۳/۰۸/۱۷ ۱۶:۱۸
آدرس: www.dotnettips.info
گروه‌ها: Analysis Services, data mining, Microsoft SQL Server

مقدمه

بطور کلی داده کاوی به دو قسمت زیر تقسیم می‌شود:

1- اهداف توصیفی (Descriptive Goal): بدنبال یافتن الگوها و روابط بین داده‌ها هستیم، بدین ترتیب مدلی برای توصیف بهتر داده‌ها بدست خواهد آمد.

2- اهداف پیش بینانه (Predictive Goal): بدنبال انجام پیش بینی با استفاده از الگوها و مدل‌های فوق هستیم.

همچنین مراحل اجرای یک پروژه داده کاوی شامل مراحل زیر است:

1- تحلیل: مهمترین فعالیت در این فاز، فهم عمیق مسئله و شناخت درست مسئله و شناسائی مفاهیم کلیدی (Key Concept) در مسئله است.

2- طراحی: مهمترین فعالیت این فاز، فرموله کردن مسئله با استفاده از مفاهیم کلیدی است.

3- پیاده سازی/ نگهداری و بهبود

مراحل کاری داده کاوی بر اساس استاندارد CRISP-DM

محصول مشترک شرکت‌های SPSS, Teradata, NCR و دایملر- کرایسلر است و یک فرآیند استاندارد Cross-Industry برای داده کاوی است که به طور گسترده ای استفاده می‌شود. مراحل کاری در این مدل به شش فاز اصلی به شرح زیر تقسیم می‌شوند:

1. درک پروژه و فهم حوزه کاربرد (Business Understanding):
به طور صریح و آشکار اهداف و نیازمندی‌ها مشخص می‌شود. ترجمه اهداف و محدودیت آن در قاعده سازی، تعریف مسئله داده کاوی و مهیا کردن استراتژی اولیه برای نائل شدن به اهداف در این مرحله تعریف می شود.

2. انتخاب داده‌ها (Data Understanding):
این مرحله شامل جمع آوری داده‌ها برای استفاده از تحلیل اکتشافی و مشخص کردن اطلاعات اولیه برای ارزیابی داده‌های با کیفیت و انتخاب داده‌های مفید و مورد نیاز می‌باشد.

3. آماده سازی داده‌ها (Data Preparation):
آماده کردن داده‌های اولیه خام به داده‌های نهایی، این دادها در کلیه مراحل بعدی استفاده می‌شود و از این نظر این مرحله تحلیل و تلاش بیشتری را می‌طلبد. انتخاب عناصر و شناسه‌های تحلیل شده را برای کاوش داده‌ها اختصاص می‌دهیم و با تمیز کردن داده‌های خام آن را برای ابزارهای مدل سازی آماده می کنیم.

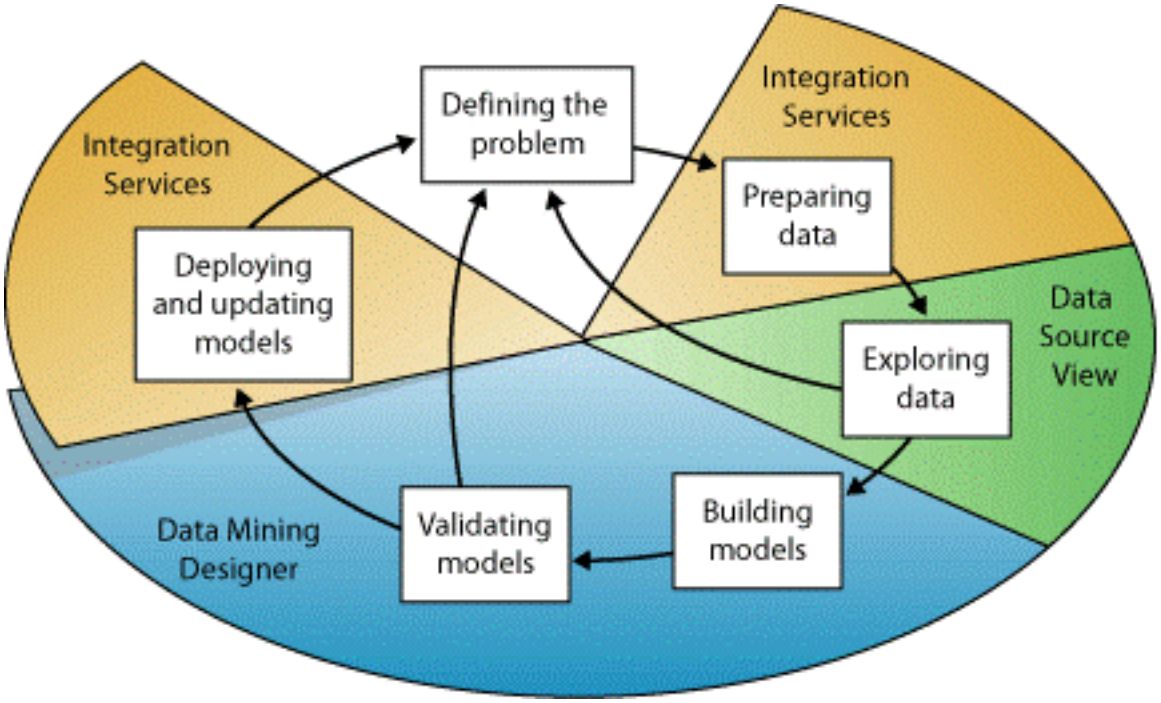
4. مدل سازی (Modeling):
با انتخاب و به کار بستن تکنیک‌های مدل سازی مناسب و روش داده کاوی معین نتایج مدل سازی را بهینه می کنیم، که در صورت نیاز می‌توانیم با برگشت به عقب تحلیل مدل سازی را بهینه‌تر نماییم.

5. ارزیابی (Evaluation):
مشخص کردن اینکه آیا مدل انتخابی، ما را به اهدافمان که در اولین مرحله تعیین کردیم، می رساند. اتخاذ تصمیم راجع به استفاده از نتایج داده کاوی برای اعتبارسنجی نیز در این مرحله انجام می شود.

6. استقرار (Deployment):
استفاده کردن از مدل ایجاد شده، برای مثال می‌تواند تولید یک گزارش ساده از خروجی‌ها را نام برد، و برای یک مثال پیچیده تکمیل کردن پردازش داده کاوی موازی در سایر حوزه‌ها می‌باشد، که این الگوها به یک دانش مفید و قابل استفاده تبدیل می‌شوند و پس از بهبود آنها، الگوهایی که کارا محسوب می شوند در یک سیستم اجرایی به کار گرفته خواهند شد.

مراحل کاری داده کاوی در بستر تکنولوژی Microsoft

داده- کاوی غالباً به عنوان فرآیند استخراج اطلاعات، الگوها و روندهای موجود در مجموعه- ی عظیمی از داده-ها یاد می- شود. این الگوها و روندها را می- توان به عنوان یک مدل کاوشی تعریف نمود. به بیانی دیگر ایجاد یک مدل کاوشی بخشی از فرآیند بزرگتری است که در برگیرنده- ی همه مراحل؛ از تعریف مسئله که مدل حل خواهد نمود تا اجرای مدل در محیط-های کاری است. می- توان این فرآیند را با استفاده از 6 مرحله اساسی زیر تعریف نمود:



باید در نظر داشت که تهیه یک مدل داده کاوی، فرآیندی چرخشی، پویا و تکرار پذیر می- باشد و ممکن است هر یک از این مراحل آن قدر تکرار شود، تا مدل مناسبی تهیه گردد.

تعریف مسئله (Defining the Problem):

تعریف روشنی از مشکل و مسئله کسب و کار است. این مرحله شامل تجزیه و تحلیل نیازمندی-های کسب و-کار، تعریف دامنه مشکل، تعریف معیارهایی که با آن مدل-ها ارزیابی خواهد شد و تعریف هدف نهایی پروژه- ی داده- کاوی است.

آماده- سازی داده-ها (Preparing Data):

یکپارچه -سازی و پالایش داده- هایی است که در مرحله- ی تعریف مسئله فرآیند معین شده است. SSIS حاوی تمامی ابزارهای ملزوم برای تکمیل این مرحله می-باشد.

بررسی داده-ها (Exploring Data):

به منظور تصمیم- گیری-های مناسب در هنگام تهیه مدل، می- بایست داده-ها را درک نمود و پس از آن می- توان تصمیم گیری در مورد وجود داده-های مخدوش در مجموعه داده و در نهایت استراتژی مناسب برای رفع این مشکلات اتخاذ نمود. Data Source view Designer موجود در BIDS حاوی ابزارهای جامعی برای بررسی و شناخت داده‌ها شامل محاسبه ارقام حداقل و حداکثر، محاسبه میانگین و انحراف معیار و بررسی توزیع داده-ها می- باشد.

تهیه مدل -ها (Building Models):

پیش از تهیه مدل باید، داده-ها را به دو دسته- ی داده-های آموزشی و اعتبارسنجی (آزمایشی) تقسیم نمود. از داده-های آموزشی برای تهیه مدل و از داده-های اعتبار-سنجی برای آزمایش صحت مدل با ایجاد سوالاتی در مورد صحت پیش- بینی-ها استفاده نمود. پس از تعریف ساختار کاوشی، می- بایست به پردازش مدل پرداخته شود و ساختارهای خالی با الگوهایی که مدل را توصیف می- نمایند، پُر شوند. این مرحله با عنوان آموزش مدل شناخته می- شود.

بررسی و ارزیابی مدل-ها (Exploring and Validating Models):

این مرحله شامل بررسی مدل-های ایجاد شده به منظور آزمودن کارایی آنهاست. می- توان مدل-ها را با ابزار-های موجود در Designer از جمله نمودار صعود و یا ماتریس دسته- بندی بررسی نمود.

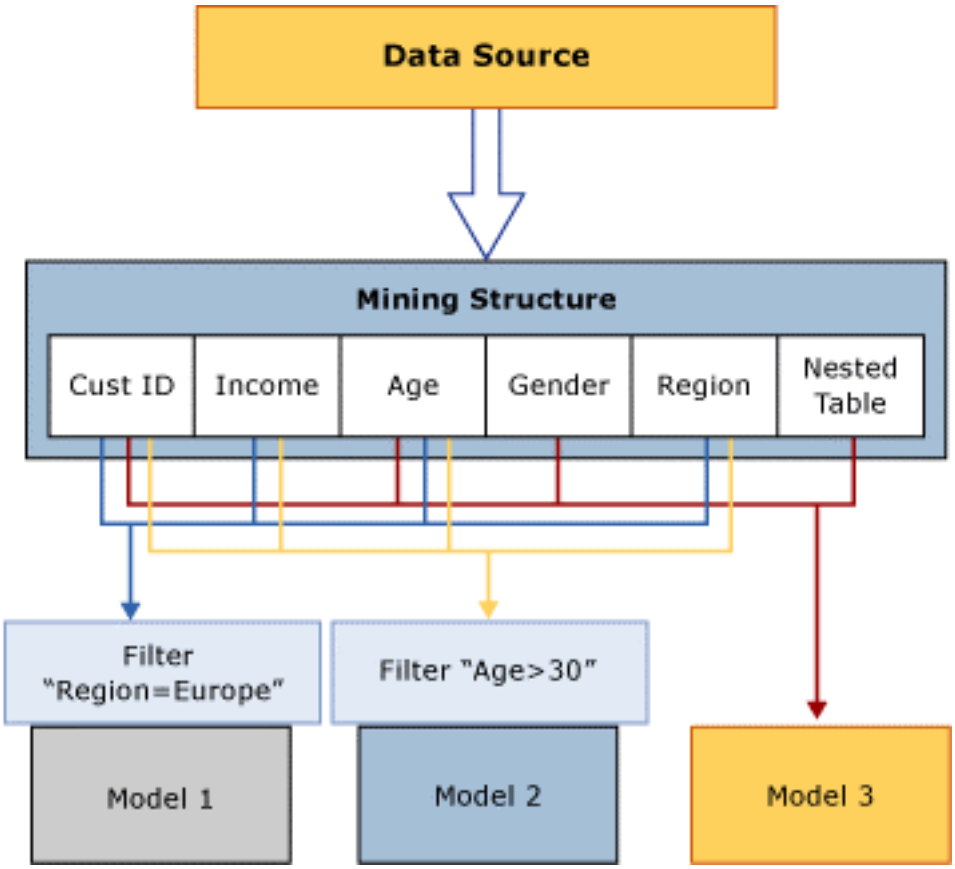
اجرا و بروزرسانی مدل-ها (Deploying and Updating Models):

این مرحله شامل اجرای مدل- هایی است که بهترین کارائی را در یک محیط عملیاتی داشته- اند. پس از استقرار مدل-های کاوشی در یک محیط عملیاتی می- توان از این مدل-ها برای پیش- بینی- هایی بهره گرفت.

مراحل سه گانه موجود در ساخت یک مدل کاوش

ایجاد ساختار کاوشی (Mining Structures): تعریف یک ساختار کاوشی شامل، تعیین تعداد ستون-های ورودی، تعداد ستون-های قابل پیش -بینی و الگوریتم وابسته به آن می-باشد. ساختار کاوشی یک ساختار داده- ای است که محدوده- ی داده- هایی را که از روی آنها مدل-های کاوش ساخته می- شود را تعریف می- نماید.

آموزش مدل (Model Training): یک مدل کاوشی، الگوریتم-های کاوش را به داده- هایی که ساختار کاوش ارائه می- نماید، اعمال می- کند. به بیان دیگر استفاده و کاربرد هر ستون و الگوریتمی که برای ساخت مدل استفاده می- شود را تعریف می- کند، پس شامل داده منبع اصلی نیست، بلکه شامل اطلاعاتی است که توسط الگوریتم کشف می- شود. به آموزش مدل، پردازش مدل نیز گفته می‌شود و زمانی که یک مدل پردازش می- شود داده- هایی که توسط ساختار کاوش تعریف شده- اند، از طریق الگوریتم-های داده- کاوی انتخابی منتقل می- شوند، الگوریتم؛ الگوها و روندها را جستجو می- کند و در ادامه این اطلاعات در مدل ذخیره می- شوند. از این رو پس از یادگیری و آموزش مدل، الگوهای بدست آمده در مدل کاوش ذخیره می- شوند.



پیش بینی مدل (Prediction): غالباً مهمترین مرحله و هدف نهایی در پروژه-های داده- کاوی است. پیش- بینی به کشف اطلاعات ناشناخته با استفاده از الگوهای یافته شده از سوابق داده-ها اشاره دارد. در پیش- بینی به یک مدل کاوشی آموزش دیده و یک مجموعه داده- ی جدید نیاز است. و در طول پیش- بینی موتور داده- کاوی، قواعد بدست آمده در مرحله یادگیری را در مورد مجموعه داده- ی جدید بکار می- برد و نتایج پیش- بینی را به هر Case ورودی تخصیص می- دهد.

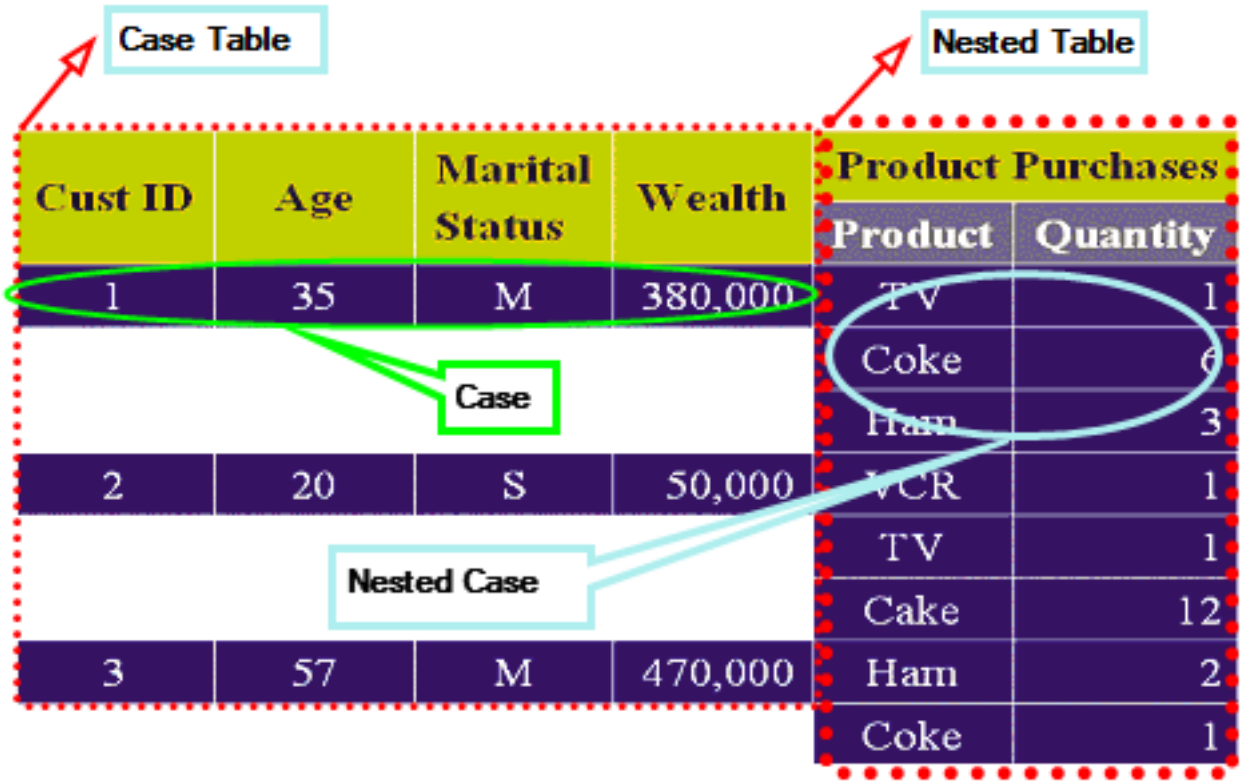
عنوان: **مفاهیم کلیدی مورد استفاده در مدل سازی داده کاوی (بخش دوم)**
نویسنده: محمد رجبی
تاریخ: ۱۳۹۳/۰۸/۱۸ ۱۲:۳۳
آدرس: www.dotnettips.info
گروه‌ها: Analysis Services, data mining, Microsoft SQL Server

مفاهیم کلیدی

Case مهمترین مفهومی است که در تحلیل یک مسئله داده کاوی می‌بایست شناسائی شود و تشخیص اشتباه در شناسائی آن منجر به عدم موفقیت پروژه داده کاوی خواهد شد. Case به معنای یک موجودیت پایه از اطلاعات می‌باشد که عملیات داده کاوی بر روی آن انجام می‌شود و هدف از معرفی آن، معرفی ساختار مسئله به موتور داده کاوی است. هر Case شامل مجموعه ای از ویژگی‌ها (Attributes) می‌باشد؛ مانند سن، جنسیت. ویژگی‌ها می‌توانند دارای یک مجموعه از مقادیر ممکن باشند که به آنها وضعیت یا مقدار (State/Value) می‌گویند؛ مانند جنسیت که دارای دو وضعیت زن یا مرد می‌باشد.
Case می‌تواند ساده باشد؛ برای نمونه زمانیکه قصد دارید «از اطلاعات آماری مشتریان به منظور تحلیل ریسک وام گرفتن» استفاده کنید، بدین ترتیب هر Case شامل اطلاعات یک مشتری و یا ردیفی از داده مشتریان است.

Case می‌تواند کمی پیچیده‌تر باشد؛ برای مثال زمانیکه می‌خواهید «رفتار خرید مشتری را بر اساس تاریخچه خرید مشتری» تحلیل کنید، که در این صورت هر Case شامل یک رکورد از اطلاعات مشتری به همراه لیستی از محصولاتی که خریداری کرده است، می‌باشد. (توجه کنید تعریف رفتار به طور ضمنی، بیانگر عملکرد در طول زمان می‌باشد)

Case مثال فوق نمونه ای از **Nested Case** است، که به اطلاعات Details در ساختار Master/Details اشاره دارد. چنانچه Case ای از نوع Nested باشد، الگوریتم‌ها به Case ای به عنوان ورودی فرمت مجموعه ردیف سلسله مراتبی (Hierarchical Row-set) نیاز دارند.



Case Key مشخصه ای است که یکتا بودن هر Case را مشخص می‌کند و اغلب Primary Key یک جدول رابطه ای است، همچنین ممکن است یک کلید ترکیبی باشد. ذکر این نکته ضروری است که بدانیم Case Key فقط یک شناسه است و شامل هیچ الگویی نمی‌باشد و بدین ترتیب غالباً بوسیله الگوریتم‌های داده کاوی نادیده گرفته می‌شود.

Nested Key مهمترین مشخصه ویژگی از بخش Nested هر Case است و در واقع کلید معنایی تحلیل می‌باشد که شامل اطلاعات مفیدی درباره‌ی الگوهاست. به بیان دیگر ویژگی است که عناصر مختلف موجود در Nested Case را به ازای هر Case تفکیک می‌کند. همچنین در نظر داشته باشید که Nested Key یک شناسه نیست و دارای مفهومی متفاوت با Foreign Key است، بدین ترتیب سایر مشخصه‌های دیگر در بخش Nested؛ جهت توصیف Nested Key بکار می‌روند. برای نمونه چنانچه مدلی برای یادگیری الگوهایی درباره رفتار خرید مشتری داشته باشیم، Nested Key برابر با محصول و میزان خرید است.

به همین ترتیب **Case Table** جدولی است شامل اطلاعات Case و بطور مشابه **Nested Table** جدولی است که شامل اطلاعات مرتبط با قسمت Nested از Case می‌باشد. از اپراتور **Shape** به منظور پیوند میان Case Table و Nested Table استفاده می‌شود.

در خصوص **Attribute** ها (ویژگی ها) از آنجا که هر ویژگی؛ توصیف کننده مسئله داده کاوی از یک منظر خاص می‌باشد، می‌توان اینگونه بیان نمود که هر چه تعداد ویژگی‌ها در یک پروژه بیشتر باشد، توان تحلیل در آن پروژه افزایش می‌یابد. انواع ویژگی‌ها به دو دسته **Discrete** (گسسته) و **Continuous** (پیوسته) تقسیم می‌شوند. برای نمونه ویژگی جنسیت، تحصیلات و ... گسسته و همچنین ویژگی سن، درآمد و ... پیوسته هستند. به مقادیر موجود در یک ویژگی پیوسته **Value** و بطور مشابه به وضعیت‌های موجود در یک ویژگی گسسته **State** گفته می‌شود. ویژگی‌ها در یک الگوریتم از حیث کاربرد (Attribute Usage) به دو دسته **Input** و **Output** تقسیم می‌شوند.

یک الگوریتم از ویژگی‌های ورودی (Input) استفاده می‌کند تا الگویی برای پیش بینی ویژگی‌های خروجی (Output) پیدا کند. همچنین لازم است در نظر داشته باشید که برخی الگوریتم‌ها نظیر Naïve Bayes صرفاً با داده‌های گسسته و بطور مشابه الگوریتم هایی نظیر Logistic Regression تنها با مقادیر پیوسته کار می‌کنند.

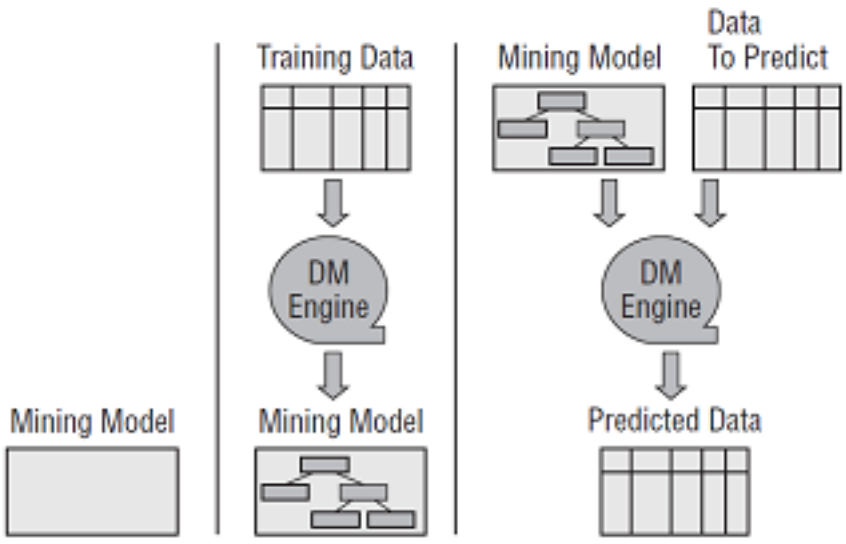
عنوان: **مروری مختصر بر زبان DMX**
نویسنده: محمد رجبی
تاریخ: ۱۳۹۳/۰۸/۲۶ ۱۸:۷
آدرس: www.dotnettips.info
گروه‌ها: Analysis Services, data mining, Microsoft SQL Server

این بخش مروری اجمالی است بر زبان DMX (Data Mining eXtensions) که به منظور انجام عملیات داده کاوی توسط شرکت ماکروسافت ایجاد شده است. (از آنجا که هدف این دوره معرفی الگوریتم‌های داده کاوی است از این رو به صورت کلی به بررسی این زبان می‌پردازیم)

برای بسیاری داده کاوی تنها مجموعه ای از تعدادی الگوریتم تعبیر می‌شود؛ به همان طریقی که در گذشته تصورشان از بانک اطلاعاتی تنها ساختاری سلسله مراتبی به منظور ذخیره داده‌ها بود. بدین ترتیب داده کاوی به ابزاری تبدیل شده که تنها در انحصار تعدادی متخصص (بویژه PhDهای علم آمار و یادگیری ماشین) قرار دارد که آشنائی با اصطلاحات یک زمینه خاص را دارند. هدف از ایجاد زبان DMX تعریف مفاهیمی استاندارد و گزارهایی متداول است که در دنیای داده کاوی استفاده می‌شود به شکلی که زبان SQL برای بانک اطلاعاتی این کار را انجام می‌دهد.

فرضیه اساسی در داده کاوی و همچنین یادگیری ماشین از این قرار است که تعدادی نمونه به الگوریتم نشان داده می‌شود و الگوریتم با استفاده از این نمونه‌ها قادر است به استخراج الگوها بپردازد. بدین ترتیب به منظور بازبینی و همچنین استنتاج از اطلاعات درباره نمونه‌های جدید می‌تواند مورد استفاده قرار گیرد.

ذکر این نکته ضروری است که الگوهای استخراج شده می‌توانند مفید، آموزنده و دقیق باشند. تصویر زیر به اختصار مراحل فرآیند داده کاوی را نمایان می‌سازد:



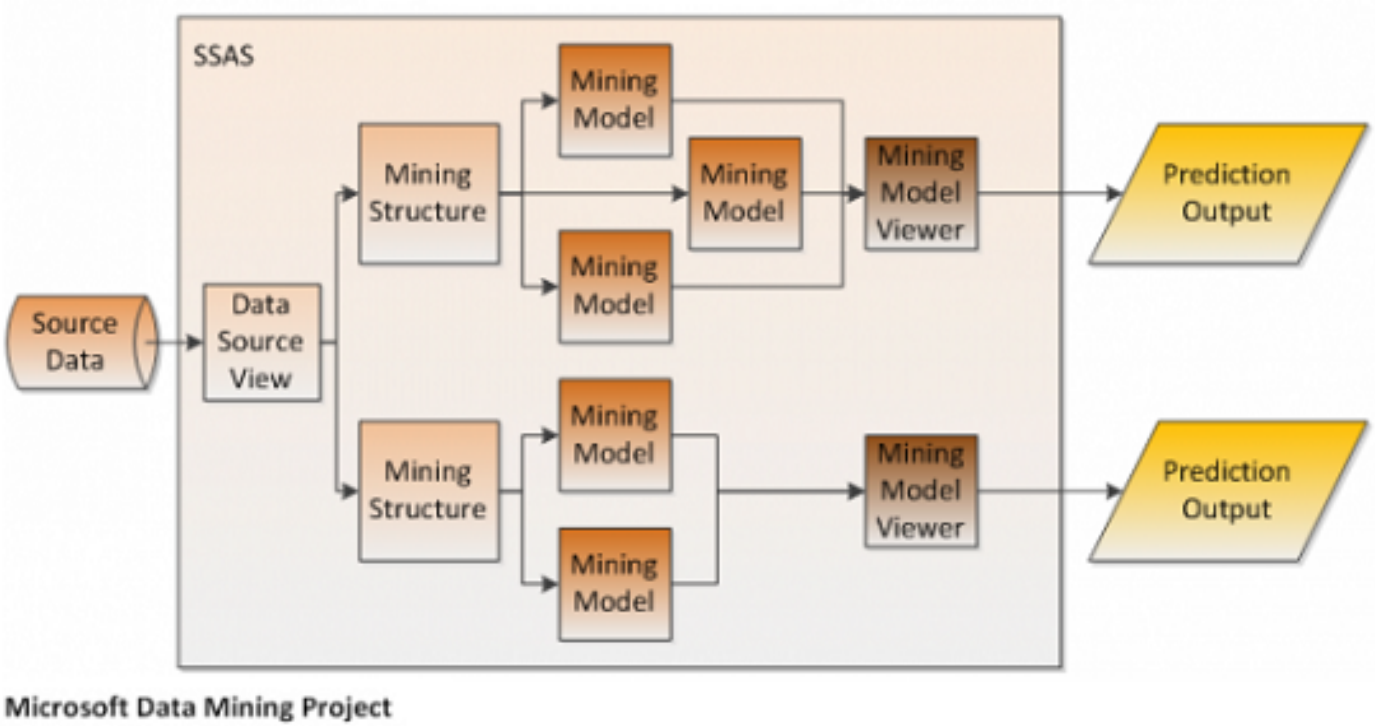
The data mining process

در گام نخست اقدام به تعریف مسئله و فرموله کردن آن می‌کنیم که اصطلاحاً Mining Model نامیده می‌شود. در واقع Mining Model توصیف کننده این است که داده نمونه به چه شکل به نظر می‌رسد و چگونه الگوریتم داده کاوی باید داده‌ها را تفسیر کند.

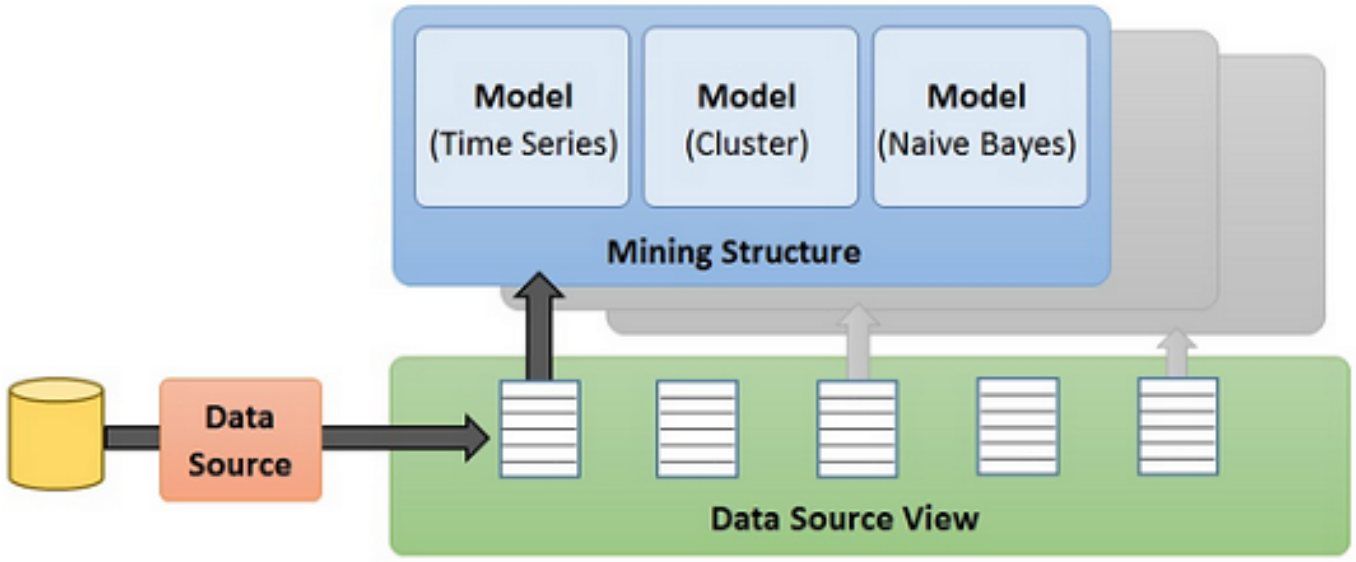
در گام بعدی به فراهم کردن نمونه‌های داده برای الگوریتم می‌پردازیم، الگوریتم با بهره گیری از Mining Model به طریقی که یک لنز داده‌ها را مرتب می‌کند، به بررسی داده‌ها و استخراج الگوها می‌پردازد؛ این عملیات را اصطلاحاً Training Model می‌نامیم.

هنگامی که این عملیات به پایان رسید، بسته به اینکه چگونه آنرا انجام داده اید، می‌توانید به تحلیل الگوهایی که توسط الگوریتم از روی نمونه هایتان بدست آمده بپردازید. و در نهایت می‌توانید اقدام به فراهم کردن داده‌های جدید و فرموله کردن آنها، به همان طریقی که نمونه‌ها آموزش دیده اند، به منظور انجام پیش بینی و استنتاج از اطلاعات با استفاده از الگوهای کشف شده توسط الگوریتم پرداخت.

زبان DMX وظیفه تبدیل داده‌های موجودتان (سطرها و ستون‌های Tables) به داده‌های مورد نیاز الگوریتم‌های داده کاوی (Cases و Attributes) را دارد. به منظور انجام این تبدیل به Mining Structure و Mining Model (که در قسمت اول به شرح آن پرداخته شد) نیاز است. بطور خلاصه Mining Structure صورت مسئله را توصیف می‌کند و Mining Model وظیفه تبدیل سطرهای داده ای به درون Caseها و انجام عملیات یادگیری ماشین با استفاده از الگوریتم داده کاوی مشخص شده را بر عهده دارد.



Microsoft Data Mining Project



Syntax زبان DMX

مشابه زبان SQL دستورات زبان DMX نیز به محیطی جهت اجرا نیاز دارند که می‌توان با استفاده از SQL Server Management Studio (SSMS) به اجرای دستورات DMX اقدام نمود. ایجاد ساختار کاوش (Mining Structure) و مدل کاوشی (Mining Model) مشابه دستورات ایجاد Table در زبان SQL می‌باشد. همانطور که اشاره شد، **گام اول** (از سه مرحله اصلی در داده کاوی) ایجاد یک مدل کاوش است؛ شامل تعیین تعداد ستون‌های ورودی، ستون‌های قابل پیش بینی و مشخص کردن نام الگوریتم مورد استفاده در مدل. **گام دوم** آموزش مدل که پردازش نیز نامیده می‌شود و **گام سوم** مرحله پیش بینی است که نیاز به یک مدل کاوش آموزش

دیده و مجموعه اطلاعات جدید دارد. در طول پیش بینی، موتور داده کاوی قوانین (Rules) پیدا شده در مرحله‌ی آموزش (یادگیری) را با مجموعه اطلاعات جدید تطبیق داده و نتیجه پیش بینی را برای هر Case ورودی انجام می‌دهد. دو نوع پرس و جوی پیش بینی وجود دارد Batch و Singleton که به ترتیب چند Case ورودی دارد و خروجی در یک جدول ذخیره می‌شود و دیگری تنها یک Case ورودی دارد و خروجی در زمان اجرا ساخته می‌شود.

در زبان DMX دو روش برای ساخت مدل‌های کاوش وجود دارد:
• ایجاد یک ساختار کاوش و مدل کاوش مربوط به هم و تحت یک نام، زمانی کاربرد دارد که یک ساختار کاوش فقط شامل یک مدل کاوش باشد.
• ایجاد یک ساختار کاوش و سپس اضافه نمودن یک مدل کاوش به ساختار تعریف شده، زمانی کاربرد دارد که یک ساختار کاوش شامل چندین مدل کاوشی باشد. دلایل مختلفی وجود دارد که ممکن است نیاز به این روش باشد، برای مثال ممکن است مدل‌های متعددی را با استفاده از الگوریتم‌های مختلف ساخت و سپس بررسی نمود که کدام مدل بهتر عمل خواهد کرد و یا مدل‌های متعددی را با استفاده از یک الگوریتم ولی با مجموعه پارامترهای متفاوت برای هر مدل ساخت و سپس بهترین را انتخاب نمود.

عناصر سازنده‌ی ساختار کاوش، ستون‌های ساختار کاوشی هستند که داده هایی را که منبع اصلی داده فراهم می‌کند، توصیف می‌کند. این ستون‌ها شامل اطلاعاتی از قبیل نوع داده (Data Type)، نوع محتوا (Content Type)، ماهیت داده و اینکه داده چگونه توزیع شده است می‌باشند. نوع محتوا پیوسته و یا گسسته بودن آن را مشخص می‌کند و بدین ترتیب به الگوریتم راه درست مدل کردن ستون را نشان می‌دهیم. کلمه کلیدی Discrete برای ماهیت گسسته داده و از کلمه Continuous برای ماهیت پیوسته داده استفاده می‌شود. مقادیر نوع داده و نوع محتوا به قرار زیر می‌باشند:

Data Type	کاربرد
LONG	اعداد صحیح
DOUBLE	اعداد اعشاری
TEXT	داده‌های رشته ای
DATE	داده‌های تاریخی
BOOLEAN	داده‌های منطقی (True و False)
TABLE	برای تعریف Nested Case

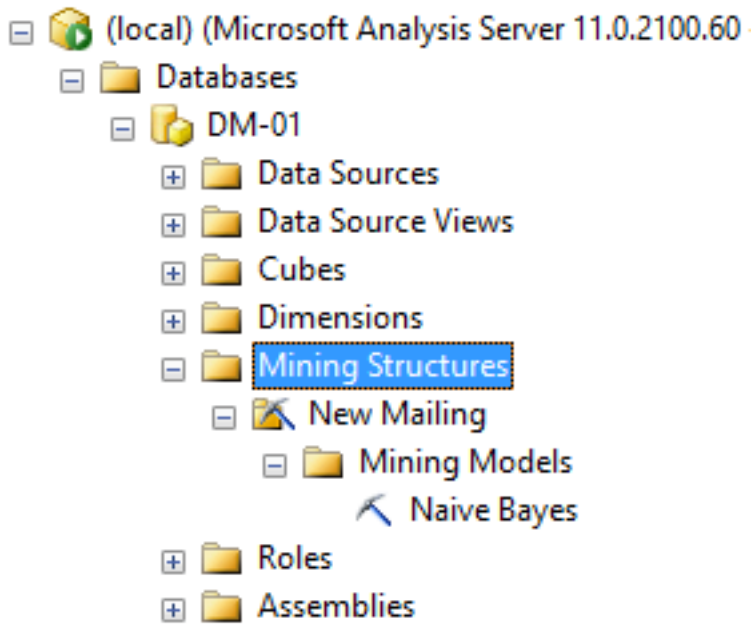
Content Type	کاربرد
KEY	مشخص کننده کلید
DISCRETE	داده‌های گسسته
CONTINUOUS	داده‌های پیوسته
DISCRETIZED	داده‌های گسسته شده
KEY TIME	کلید زمان، تنها در مدل‌های Time Series استفاده می‌شود
KEY SEQUENCE	کلید توالی، تنها در بخش Nested Table مدل‌های Sequence Clustering استفاده می‌شود

همچنین یک مدل کاوش استفاده و کاربرد هر ستون و الگوریتمی که برای ساخت مدل استفاده می‌شود را تعریف می‌کند، می‌توانید با استفاده از کلمه کلیدی Predict و یا Predict_Only خاصیت پیش بینی را به ستون‌ها اضافه نمود، برای نمونه به دستورات زیر توجه نمائید:

```
CREATE MINING STRUCTURE [New Mailing]
(
CustomerKey LONG KEY,
Gender TEXT DISCRETE,
```

```
[Number Cars Owned] LONG DISCRETE,
[Bike Buyer] LONG DISCRETE
)
GO
ALTER MINING STRUCTURE [New Mailing]
ADD MINING MODEL [Naive Bayes]
(
CustomerKey,
Gender,
[Number Cars Owned],
[Bike Buyer] PREDICT
)
USING Microsoft_Naive_Bayes
```

شکل زیر نشان دهنده ارتباط بین ساختار کاوش و مدل کاوشی پس از ایجاد در محیط SSMS می‌باشد.



به منظور آموزش یک مدل کاوش از دستور Insert به شکل زیر استفاده می‌شود:

```
INSERT INTO <mining model name>
[<mapped model columns>]
<source data query>
```

که source data query می‌تواند یک پرس و جوی Select از بانک اطلاعاتی باشد که معمولاً با استفاده از سه طریق OPENQUERY، OPENROWSET و SHAPE بدست می‌آید.

در ادامه به شکل عملی می‌توانید با طی مراحل و اجرای کوئری‌های زیر به بررسی بیشتر موضوع بپردازید.

ابتدا به سرویس SSAS متصل شوید و اقدام به ایجاد یک Database با تنظیمات پیش فرض (مثلاً با نام DM-02) نمائید و در ادامه کوئری XMLA زیر را جهت ایجاد Data Source ای به بانک AdventureWorksDW2012 موجود روی دستگاه تان، اجرا نمائید.

```
<Create xmlns="http://schemas.microsoft.com/analysisservices/2003/engine">
<ParentObject>
  <DatabaseID>DM-02</DatabaseID>
</ParentObject>
<ObjectDefinition>
  <DataSource xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:ddl2="http://schemas.microsoft.com/analysisservices/2003/engine/2"
xmlns:ddl2_2="http://schemas.microsoft.com/analysisservices/2003/engine/2/2"
xmlns:ddl100_100="http://schemas.microsoft.com/analysisservices/2008/engine/100/100"
```

```
xmlns:ddl200="http://schemas.microsoft.com/analysisservices/2010/engine/200"
xmlns:ddl200_200="http://schemas.microsoft.com/analysisservices/2010/engine/200/200"
xmlns:ddl300="http://schemas.microsoft.com/analysisservices/2011/engine/300"
xmlns:ddl300_300="http://schemas.microsoft.com/analysisservices/2011/engine/300/300"
xmlns:ddl400="http://schemas.microsoft.com/analysisservices/2012/engine/400"
xmlns:ddl400_400="http://schemas.microsoft.com/analysisservices/2012/engine/400/400"
xsi:type="RelationalDataSource">
<ID>Adventure Works DW2012</ID>
<Name>Adventure Works DW2012</Name>
<ConnectionString>Provider=SQLNCLI11.1;Data Source=(local);Integrated Security=SSPI;
Initial Catalog=AdventureWorksDW2012</ConnectionString>
<ImpersonationInfo>
<ImpersonationMode>ImpersonateCurrentUser</ImpersonationMode>
</ImpersonationInfo>
<Timeout>PT0S</Timeout>
  </DataSource>
</ObjectDefinition>
</Create>
```

و در ادامه کوئری‌های DMX زیر را اجرا نمائید و خروجی هر یک را تحلیل نمائید.

```
/* Step 1 */
CREATE MINING MODEL [NBSample]
(
CustomerKey LONG KEY,
Gender TEXT DISCRETE,
[Number Cars Owned] LONG DISCRETE,
[Bike Buyer] LONG DISCRETE PREDICT
)
USING Microsoft_Naive_Bayes
Go

/* Step 2 */
INSERT INTO NBSample (CustomerKey, Gender, [Number Cars Owned],
[Bike Buyer])
OPENQUERY([Adventure Works DW2012],'Select CustomerKey, Gender, [NumberCarsOwned], [BikeBuyer]
FROM [vTargetMail]')

/*  */
SELECT * FROM [NBSample].CONTENT

/*  */
SELECT * FROM [NBSample_Structure].CASES

/* Step 3*/

SELECT FLATTENED MODEL_NAME,
(SELECT ATTRIBUTE_NAME, ATTRIBUTE_VALUE, [SUPPORT], [PROBABILITY], VALUETYPE FROM NODE_DISTRIBUTION) AS
t
FROM [NBSample].CONTENT
WHERE NODE_TYPE = 26
```

در قسمت‌های بعد تا حدی که از هدف اصلی دوره بررسی الگوریتم‌های داده کاوی موجود در SSAS دور نیافتیم، به بررسی بیشتر دستورات DMX می‌پردازیم. جهت اطلاعات بیشتر در مورد زبان DMX می‌توانید به Books Online for SQL Server مراجعه نمائید.

عنوان: **پیش نیاز ورود به دنیای داده کاوی**
نویسنده: محمد رجبی
تاریخ: ۱۳۹۳/۰۸/۳۰ ۲۲:۴۴
آدرس: www.dotnettips.info
گروه‌ها: Analysis Services, data mining, Microsoft SQL Server

علم داده کاوی از علوم مختلفی از جمله علم آمار، هوش مصنوعی، یادگیری ماشین، شناسائی الگو و پایگاه داده نشات گرفته است و این علوم ریشه‌های علم داده کاوی هستند. برای مثال الگوریتم هایی که یک مدل را یاد می‌گیرند یا الگویی را شناسائی می‌کنند؛ معمولا وجه مشترک یادگیری ماشین و شناسائی الگو با داده کاوی هستند.

در این قسمت پیش از درگیر شدن با جزئیات هر الگوریتم تمایل دارم خوانندگان محترم را با مطالبی که شاید کمتر در دنیای IT با آن درگیر بوده اند؛ آشنا کنم. این کار به این دلیل انجام می‌شود که برای مثال در کشف قوانین انجمنی یا دسته بند مبتنی بر قانون (مثال متداول آن تحلیل سبد خرید مشتری در هایپر مارکت است) خروجی به شکل مجموعه ای قانون «اگر الف؛ آنگاه ب» و ... بدست می‌آید. بنابراین برای تفسیر صحیح این مدل‌ها علاوه بر آشنائی با کسب وکار مربوطه؛ نیازی نسبی به آشنائی با سایر علوم نیز می‌باشد و بدین ترتیب از اتلاف انرژی و زمان و همچنین از بروز خطا در استدلالمان جلوگیری می‌کنیم. جمله معروفی با این مضمون در سایر فرهنگ‌ها وجود دارد که اعداد دروغ نمی‌گویند؛ ولی فردی دروغگو می‌تواند از اعداد سوء استفاده کند. بنابراین زمان مناسبی است که با بعضی مغالطات آشنا شویم.

اساس کار علمی به بیان ساده عبارت است از: به پرسش گرفتن همه چیز و دنبال کردن مدارک و شواهد به هر کجا که ما را رهنمون سازد؛ اینکار بوسیله آزمودن هر نظر و ایده ای، با انجام آزمایش روی آن‌ها و مشاهده نتایج بدست آمده و سپس توسعه دادن مواردی که از آزمایشات موفق بیرون آمده اند و رد کردن آنهایی که در آزمون شکست خورده اند، انجام می‌گیرد. روش علمی آنچنان قدرتمند است که در طی چهار قرن گذشته (قرن 16 میلادی) ما را از نخستین نگاهی که گالیله از درون تلسکوپ به دنیای دیگر انداخت، به گام گذاشتن بر روی ماه رسانده است و به ما اجازه داده تا به پهنه فضا و زمان بنگریم تا کشف کنیم که در کجا و در چه زمانی از عالم قرار داریم.

اجداد ما ستاره شناسان خانه به دوشی بودند که در گروه‌های کوچک زندگی می‌کردند، آسمان تقویم و راهنمای زندگی آنها بود، بقای شان به این وابسته بود که بدانند چگونه ستاره‌ها را بخوانند و بدین ترتیب بتوانند فرا رسیدن زمستان را پیش بینی کنند و زمان کوچ کردن را بدست آورند. در واقع نعمت **تشخیص الگو** باعث شانس بیشتر زنده ماندن و تولید مثل آنها بود و بدین ترتیب ژنهای تشخیص الگو را به نسل‌های آینده منتقل می‌کردند. آنها وقتی که ارتباط مستقیمی بین حرکت ستارگان و گردش فصلی حیات روی زمین پیدا کردند، نتیجه گرفتند که اتفاقاتی که آن بالا می‌افتد به ما در پائین مربوط می‌شود و آنرا به خود می‌گرفتند!؟ آنها توضیح منطقی دیگری برای اتفاق پیش آمده نداشتند. کلمه یونانی Dis-aster به معنی "ستاره شوم" حتی برای اقوام مختلف به معنای جنگ، قحطی، مریضی و ... تعبیر می‌شد. (در فرهنگ ما نیز جملاتی با این مضمون کم وجود ندارد، برای مثال:" قمر در عقرب است"، پس اتفاق بدی خواهد افتاد!. البته منظور قرار گرفتن ماه در برج عقرب است و ...).

می توان گفت استعداد انسان در تشخیص الگو شمشیری دو لبه است، ما انسان‌ها قادریم در تشخیص الگوهائی که اصلاً وجود ندارند نیز خیلی خوب عمل کنیم!، چیزی که به معنای "تشخیص الگوی اشتباه" است. ما عاشق خاص بودن هستیم و با داشتن این هدف همواره در تلاش برای فریب خود و دیگران هستیم. علم در مرز میان دانایی و جهالت گام بر می‌دارد، از نظر یک محقق هیچ شرمساری در ندانستن وجود ندارد، تنها شرمساری در آن است که تظاهر کنیم همه جواب‌ها را می‌دانیم. علم راهی است که انسان را از فریب خود و دیگران باز می‌دارد و امروزه به نیکی می‌دانیم هر چه علم بیشتر در اختیار ابنای بشر قرار گیرد، امکان سوء استفاده از آن کمتر خواهد شد. بدین ترتیب با دانستن ارزش‌های علمی تقاضا برای جهالت و تعصب کم خواهد شد.
ارزش‌های علمی مختصراً به شرح زیر هستند: قدرت سوال کردن، وقتی موضوعی را بررسی می‌کنید تنها چیزی که باید از خودتان بپرسید این است که واقعیت‌ها در این موضوع (فلسفه) چه هست و چه حقایقی در آن نهفته است. هیچگاه به خودتان اجازه ندهید که آنچه را دوست دارید، حقیقت داشته باشد (اگر یک ایده دلخواه در یک آزمایش خوب مردود شد، پس اشتباه است و از آن عبور کنید)، همچنین آنچه را که فکر می‌کنید حقیقت بودنش برای بشر سودمند است شما را منحرف نکند (برای خودتان فکر کنید و از خودتان بپرسید)، فقط و تنها به این که واقعیت چه هست بنگرید، در ضمن اگر مدرکی ندارید؛ قضاوت نکنید و مهمترین قانون؛ به یاد داشته باشید که شما انسان هستید و می‌توانید اشتباه کنید، همانطور که مهمترین دانشمندان در مواردی اشتباهاتی داشته اند.

منطق ابزاری علمی است که بکارگیری آن ذهن انسان را از خطای در تفکر باز می‌دارد، مبارزه با مغالطات و لغزش‌های اندیشه هدف علم منطق است. مغالطه منحصر به استدلال نیست، به بیان دقیق‌تر شکل هایی از استدلال است که نتیجه تابع مقدمه یا مقدمه هایش نیست. مغالطه ای که عمدی یعنی با آگاهی از عدم اعتبار انجام می‌شود اما به ظاهر معتبر و مجاب کننده و در واقع فریب دهنده مخاطب است سفسطه نامیده می‌شود. عدم اعتبار یک استدلال ممکن است به دلایل زیر باشد: ناشی از نادرستی یکی از مقدمات استدلال باشد و یا علی رغم درستی مقدمات؛ نظم و صورت استدلال نادرست باشد. برای آشنایی ذهن خواننده به معرفی نمونه ای از این مغالطات اشاره می‌شود؛ برای مثال این مغالطه بر این پیش فرض استوار است که هر زمان دو حادثه با یکدیگر اتفاق افتاد؛ می‌توان یکی را علت و دیگری را معلول آن به حساب آورد. برای مثال در تحقیقی به ارتباط مستقیم میان وجود داشتن چتر در ماشین به هنگام تصادفات رانندگی پرداخته شده و به این نتیجه رسیده اند زمانی که تصادفی رخ می‌دهد با احتمال بسیار بالاتری چتر در ماشین وجود دارد به نسبت حالتی که چتر در ماشین وجود ندارد؛ به همین دلیل چتر عامل تصادف است! برای اجتناب از این مغالطات باید قادر به تفکیک اصل علیت (Causality) و همبستگی (Correlation) باشیم. (در توضیح مثال فوق لغزندگی جاده عامل تصادف در روزی بارانی است نه چتر!).

همچنین استفاده از آمار و اطلاعات آماری علی رغم فوائد زیاد در اطلاع رسانی، می‌تواند لغزشگاهی باشد که زمینه ارتکاب برخی مغالطات را نیز فراهم کند در ادامه به معرفی تعدادی از این مغالطات آماری (**Statistical Fallacies**) می‌پردازیم:
مغالطه متوسط که می‌تواند با سوء استفاده از برخی اصطلاحات آماری مطابق با اهداف و اغراضی که موسسات ارائه دهنده اطلاعات آماری دنبال می‌کنند، متوسط یک مجموعه را کم یا زیاد اعلام کنند! به بیان دیگر کلمه متوسط در نوبت‌های مختلف به معانی متداولی استعمال می‌شود که عبارتند از:
میانگین (Average) یا معدل که برای چند عدد برابر است با مجموع آنها تقسیم بر تعدادشان.
میانه (Median) که یک مجموعه عددی را به دو نیم تقسیم می‌کند؛ نیمی که هر یک از اعداد آن بیشتر از میانه و نیمی که کمتر از میانه است.
نما (Mode) که در یک مجموعه؛ عددی است که بیش از دیگر اعداد تکرار شده است.
پس می‌توان نتیجه گرفت وقتی اعلام می‌شود که در یک جامعه آماری فلان عدد یک متوسط است هنوز اطلاع دقیقی داده نشده و باید صراحتا مشخص کنند کدامیک از معانی متوسط مورد نظر است.
باید در نظر داشته باشید این مغالطه زمانی استفاده می‌شود که دامنه تغییرات در میان جامعه آماری بسیار زیاد است، چنانچه دامنه تغییرات حداقل و حداکثر نسبت به تعداد افراد جامعه زیاد نباشد، مقادیر میانگین؛ میانه و نما تقریبا منطبق بر هم خواهند شد (برای مثال در محاسبه متوسط طول قد افراد یک کشور). اما در مواردی که تغییرات مذکور زیاد باشد باید با هوشیاری از وقوع این مغالطه جلوگیری نمود (از مصادیق و زمینه‌های بارز و مهم ارتکاب این مغالطه محاسبه متوسط حقوق و درآمد افراد است).

مغالطه نمودارهای گمراه کننده (Misleading Graph) استفاده از نمودار می‌تواند وسیله ای موثر در بیان مغالطه آمیز بودن اطلاعات آماری باشد. برای مثال نمودار رشد سود خالص شرکتی را در نظر بگیرید که در محور افقی آن بعد زمان و در محور عمودی مقادیر مالی درج شده است. با رسم نمودار مذکور سود خالص هر ماه به صورت واضح و آشکار مثلا رشدی ده درصدی را نمایش می‌دهد چنانچه شرکت مذکور اصول اخلاقی را رعایت نکند و برای جذابیت بیشتر و جذب سرمایه‌های بیشتر؛ قسمت هایی از نمودار را به گونه ای حذف کند که حاصل کار این شود که خواننده احساس کند سود خالص شرکت در عرض دوازده ماه به بالای کاغذ رسیده (یعنی به طور ضمنی افزایشی معادل صد در صد) و یا نسبت بین خطوط افقی و عمودی را بگونه ای تغییر دهد تا رشد ده درصدی را بسیار بزرگتر نشان داده شود (می تواند با تقلیل مقیاس واحد مالی به یک دهم به این هدف برسد) بدین ترتیب نمودار حاصل چنان جذاب می‌شود که هر کس با تماشای آن رگه‌های موفقیت و پیشرفت را در شرکت متقلب بوضوح مشاهده می‌کند.

مغالطه تصاویر یک بعدی (One Dimensional Pictures) از روش‌های تقلب دیگر می‌تواند باشد که باید توجه کرد آیا نسبت القا شده بوسیله تصاویر با نسبت اعداد مطابقت دارد یا خیر.

می دانیم آنچه پایه و اساس آمار استنباطی را تشکیل می‌دهد روش‌های نمونه گیری است که اتفاقا این روش‌ها منشاء برخی مغالطات و ترفندهای آماری نیز هست در این قسمت به معرفی تعدادی از این موارد می‌پردازیم:

نمونه ناکافی (Deficient Examples) چنانچه در روش نمونه گیری مقدار و نسبت «نمونه» به «جامعه آماری» به اندازه کافی

بزرگ باشد و به طرز صحیحی انتخاب شده باشد؛ غالبا می‌تواند معرف خوبی برای جامعه آماری باشد. اما چنانچه نمونه به اندازه کافی بزرگ نباشد؛ گرچه اطلاعاتی را در خصوص جامعه آماری در اختیارمان قرار می‌دهد ولیکن احتمال وقوع خطا در چنین حالتی بسیار زیاد است که این مغالطه دارای این شرایط است؛ البته باید توجه داشت که کافی یا ناکافی بودن تعداد نمونه‌ها نسبت به جامعه آماری امری نسبی است. بنابراین جهت اجتناب از بروز این مغالطه باید همواره در نظر داشت آیا تعداد نمونه‌ها در مقایسه با کل جامعه آماری راضی کننده و کافی است یا خیر.

نمونه غیر تصادفی (Deliberate Examples) برای بدست آوردن اطلاعات آماری در روش نمونه برداری؛ کافی بودن نمونه‌ها شرط لازم است و کافی نیست؛ یکی از مواردی که باید مورد توجه قرار داد تصادفی بودن نمونه‌ها می‌باشد. به بیان دیگر تنها کافی بودن نمونه‌ها یا فراوانی آنها برای تعمیم دادن حکمی به کل آن جامعه آماری کفایت نمی‌کند. تصادفی بودن نمونه‌ها بدین معناست که نمونه‌ها نباید نماینده و بیانگر دسته و گروه خاصی از جامعه آماری باشند. همچنین در روش نمونه برداری افراد جامعه آماری باید از شانس یکسانی برای انتخاب شدن در نمونه برداری برخوردار باشند از راه‌های تحقق این هدف تقسیم افراد جامعه آماری به دسته‌ها و طبقات مختلف و تعیین کردن درصد و نسبت هر یک از آنها به کل مجموعه می‌باشد بدین ترتیب در نمونه برداری نیز سعی می‌شود این نسبت لحاظ گردد؛ این روش اصطلاحا روش نمونه گیری تصادفی طبقه ای نامیده می‌شود روش‌های دیگری نیز به منظور اینکه کلیه افراد جامعه آماری از شانسی یکسان برای انتخاب شدن در نمونه برخوردار باشند وجود دارد مانند روش‌های نمونه گیری تصادفی ساده؛ نمونه گیری تصادفی خوشه ای و نمونه گیری تصادفی سیستماتیک.

عدم واقع نمائی نمونه‌ها (Unrealistic Examples) در نمونه برداری به صورت پرسش‌های شفاهی از جامعه آماری انسانی مسئله عدم واقع نمائی نمونه‌ها رخ می‌دهد بدین ترتیب همواره موجب بروز خطاهای جدی در بدست آوردن اطلاعات آماری دقیق است. این مشکل عملا به روش جمع آوری داده‌ها از طریق مصاحبه بر می‌گردد خواه به صورت نمونه ای یا سرشماری باشد.

نظرات خوانندگان

نویسنده: مصطفی وکیلی
تاریخ: ۱۳۹۳/۱۰/۲۰ ۱۷:۴۷

یه سوالی که برای من چند وقتیه به وجود اومده اینه که وضعیت بازار داده کاوی تو ایران در چه وضعیتی هستش؟ متخصص این زمینه چقده حقوق دریافتی داره
اگه کسی اطلاعات داره خوشحال میشم بهم بگه

عنوان: **یادگیری مدل در داده کاوی**
نویسنده: محمد رجبی
تاریخ: ۱۳۹۳/۰۹/۰۱ ۱۳:۴۶
آدرس: www.dotnettips.info
گروه‌ها: Analysis Services, data mining, Microsoft SQL Server

مقدمه هدف اصلی داده کاوی کشف دانش است، که این دانش نظمی که در داده‌ها وجود دارد را نمایان می‌سازد. پس از کشف دانش ممکن است با دو وضعیت مواجه شویم:
حالت اول هنگامی است که افراد خبره در دامنه داده مورد کاوش، آگاه به دانش استخراج شده باشند که در این صورت آن دانش به عنوان یک قانون صحیح تلقی خواهد شد.
در حالت دوم ممکن است دانش کشف شده، یک دانش جدید بوده و در بین افراد خبره در آن حوزه شناخته شده نباشد، در این صورت این دانش بررسی شده و در صورت منطقی بودن تبدیل به فرضیه شده و در نهایت درست یا غلط بودن این فرضیه با آزمایشات و بررسی‌های متعدد اثبات می‌شود و در صورت درست بودن فرضیه تبدیل به قانون خواهد شد.

روش‌های یادگیری مدل در داده کاوی پیشتر به معرفی مراحل کاری در داده کاوی که مشتمل بر سه مرحله اساسی: **آماده سازی داده** ، **یادگیری مدل** و در نهایت **ارزیابی و تفسیر مدل** می‌باشد، پرداختیم.
در مرحله یادگیری مدل با استفاده از الگوریتم‌های متنوع و با در نظر گرفتن ماهیت داده، نظم‌های مختلف موجود در داده‌ها شناسائی می‌شود. بطور کلی روش‌های مختلف کاوش داده را به دو گروه روش‌های پیش بینی و روش‌های توصیفی طبقه بندی می‌کنند.

در **روش‌های پیش بینی** از مقادیر بعضی ویژگی‌ها برای پیش بینی کردن مقدار یک ویژگی مشخص استفاده می‌کنند. این روش‌ها در متون علمی با نام روش‌های با ناظر (Supervised Methods) نیز شناخته می‌شوند. الگوریتم‌های با ناظر از دو مرحله با عنوان مرحله آموزش (یادگیری) و مرحله ارزیابی تشکیل شده اند.
در مرحله آموزش؛ با استفاده از مجموعه داده‌های آموزشی مدل ساخته می‌شود. شکل مدل ساخته شده به نوع الگوریتم یادگیرنده بستگی دارد.
در مرحله ارزیابی؛ از مجموعه داده‌های آزمایشی برای اعتبارسنجی و محاسبه دقت مدل ساخته شده استفاده می‌شود، در واقع از داده هایی که در مرحله آموزش و ساخت مدل؛ الگوریتم این مجموعه داده‌ها را ندیده است (Previously Unseen Data) استفاده می‌شود.
برای نمونه روش‌های **دسته بندی** (Classification)، **رگرسیون** (Regression) و **تشخیص انحراف** (Anomaly Detection) سه روش یادگیری مدل در داده کاوی با ماهیت پیش بینی هستند.
در **روش‌های توصیفی** همانطور که انتظار داریم الگوهای قابل توصیف از روابط حاکم بر داده‌ها بدون در نظر گرفتن هر گونه برچسب و یا متغیر خروجی بدست می‌آید. این روش‌ها در متون علمی با نام روش‌های بدون ناظر (Unsupervised Methods) نیز شناخته می‌شوند. برای نمونه روش‌های **خوشه بندی** (Clustering)، **کاوش قوانین انجمنی** (Association Rules Mining) و **کشف الگوهای ترتیبی** (Sequential Pattern Discovery) سه روش یادگیری مدل در داده کاوی با ماهیت توصیفی هستند.

در ادامه به معرفی هر کدام از این روش‌ها می‌پردازیم:

دسته بندی: در الگوریتم‌های دسته بندی مجموعه داده اولیه به دو مجموعه داده با عنوان مجموعه داده‌های آموزشی (Train Dataset) و مجموعه داده‌های آزمایشی (Test Dataset) تقسیم می‌شود. می‌دانیم هر Case شامل مجموعه ای از Attribute هاست، که یکی از این ویژگی‌ها **ویژگی دسته** نامیده می‌شود.
در مرحله آموزش؛ مجموعه داده‌های آموزشی به یکی از الگوریتم‌های دسته بندی داده می‌شود تا بر اساس سایر ویژگی‌ها برای مقادیر ویژگی دسته، مدل ساخته شود.
پس از ساخت مدل، در مرحله ارزیابی؛ دقت مدل ساخته شده به کمک مجموعه داده‌های آزمایشی ارزیابی خواهد شد. در الگوریتم‌های دسته بندی از آنجا که ویژگی دسته مربوط به هر Case مشخص است به صورت الگوریتم‌های با ناظر محسوب می‌شوند. بدیهی است که تشخیص بر اساس دسته هایی است که مدل در مرحله آموزش با آنها روبرو شده است؛ بنابراین امکان تشخیص دسته جدید در کاربرد دسته بندی وجود نخواهد داشت.

رگرسیون: رگرسیون در علوم آمار و شبکه‌های عصبی بطور وسیعی مورد بررسی و مطالعه قرار می‌گیرد. پیش بینی مقدار یک متغیر پیوسته بر اساس مقادیر سایر متغیرها بر مبنای یک مدل وابستگی خطی یا غیر خطی رگرسیون نامیده می‌شود. یک نوع خاصی از رگرسیون، **پیش بینی سری‌های زمانی** (Time Series Prediction) است؛ برای مثال تغییرات قیمت سهام شرکتی را به صورت نمودار داریم؛ می‌خواهیم ادامه روند این نمودار را برای مدتی مشخص پیش بینی کنیم. در مسائل سری‌های زمانی یکی از متغیرهای اصلی زمان می‌باشد. بدیهی است که رگرسیون لزوماً سری زمانی نیست و همانند دسته بندی کاربرد رگرسیون نیز از نوع پیش بینی با ناظر است و بطور مشابه در رگرسیون هم دو مرحله آموزش و ارزیابی نیز وجود دارد. مثال هایی از رگرسیون می‌تواند شامل موارد زیر باشد: پیش بینی میزان فروش یک محصول جدید، براساس میزان فروش محصولات گذشته و یا براساس میزان تبلیغات انجام شده و ... همچنین مسائل مربوط به پیش بینی سری‌های زمانی از قبیل بورس و ... .

تشخیص انحراف: از کاربردهای متداول تشخیص انحراف، می‌توان به **کشف کلاهبرداری** کارت‌های اعتباری (Credit Card Fraud Detection) اشاره کرد. در مواقعی از این کاربرد استفاده می‌شود که تنها نمونه هایی با یک برچسب یکسان که معمولاً وضعیت نرمال را نشان می‌دهند در دسترس می‌باشند و امکان مالکیت بر داده‌ها با تمامی برچسب‌های موجود به دلایل مختلف وجود ندارد. بنابراین چون فقط نمونه‌های دسته نرمال در اختیار است، الگوریتم برای وضعیت نرمال و با توجه به یک آستانه (Threshold) مشخص مدل را می‌سازد و هر گونه تخطی از آن آستانه را؛ بعنوان وضعیت غیرنرمال در نظر می‌گیرد. توجه شود روش‌های دسته بندی تنها قادر به شناسائی دسته هایی هستند که در مرحله آموزش، نمونه ای از آنها به الگوریتم ارائه شده است، بنابراین امکان تشخیص هیچ گونه کلاهبرداری توسط روش‌های دسته بندی وجود ندارد.

خوشه بندی: در این مسائل از آنجا که بر خلاف دسته بندی هیچ گونه دسته خاصی وجود ندارد، بنابراین براساس معیار شباهت داده‌ها گروه بندی و خوشه بندی صورت می‌گیرد. بدین ترتیب Case هایی که بیشترین شباهت را به یکدیگر دارند در یک خوشه قرار می‌گیرند، به بیان دیگر Caseهای موجود در خوشه‌های متفاوت کمترین شباهت را به یکدیگر خواهند داشت. بدیهی است که خوشه بندی براساس ویژگی ورودی نمونه‌ها انجام می‌گیرد و از آنجائی که برای این الگوریتم‌ها ویژگی دسته تعریف نمی‌شود و Caseها برچسب خاصی ندارند، جزء الگوریتم‌های بدون ناظر محسوب می‌شوند. در واقع هدف در تمامی الگوریتم‌های خوشه بندی **کمینه کردن فاصله درون خوشه ای** (Intra-Cluster Density) و **بیشینه نمودن فاصله بین خوشه ای** (Inter-Cluster Density) است و عملکرد خوب یک الگوریتم خوشه بندی زمانی محرز می‌شود که تا حد امکان خوشه‌ها را از یکدیگر دورتر کند و در ضمن Caseهای موجود در یک خوشه بیشترین شباهت را به یکدیگر داشته باشند.

کشف قوانین انجمنی: قوانین وابستگی (انجمنی) اتفاق و وقوع یک شیء را براساس وقوع سایر اشیاء توصیف می‌کنند، برای مثال در یک سوپر مارکت هدف در کاوش قوانین انجمنی؛ یافتن نظم حاکم بر سبد خرید می‌باشد، در این کاربرد به ازای هر سبد؛ یک قانون پیدا می‌شود و بررسی خواهد شد که این قانون در چه تعداد از سبدها صدق می‌کند و در نهایت یک مجموعه قوانین که در بیشترین تعداد از سبدها صدق می‌کند به عنوان مجموعه قوانین انجمنی خروجی ارائه می‌شود. به بیان دیگر در این کاربرد به دنبال پیدا کردن یک مجموعه از قوانین وابستگی هستیم تا براساس آن قوانین بتوانیم نتیجه گیری کنیم وجود کدامیک از مجموعه اشیاء (Item Set) بر وجود چه مجموعه اشیاء دیگری تاثیر گذار است.

کشف الگوهای ترتیبی: در این کاربرد به دنبال کشف الگوهایی هستیم که وابستگی‌های ترتیبی محکمی را در میان وقایع مختلف نشان می‌دهند. این کاربرد مشابه کاوش قوانین انجمنی می‌باشد با این تفاوت که در کاوش قوانین انجمنی زمان و ترتیب زمانی مطرح نیست، اما در کشف الگوهای ترتیبی زمان و ترتیب اهمیت ویژه ای دارند برای مثال می‌توان به دنباله‌های تراکنش‌های فروش اشاره نمود.

منبع: با اندکی تغییر و تلخیص "داده کاوی کاربردی در RapidMiner، انتشارات نیاز دانش"

عنوان: **مروری بر روش ها و رویکردهای مختلف در یادگیری مدل**
نویسنده: محمد رجبی
تاریخ: ۱۳۹۳/۰۹/۰۹ ۱۷:۳۵
آدرس: www.dotnettips.info
گروه‌ها: Analysis Services, data mining, Microsoft SQL Server

مقدمه

همان گونه که اشاره شد در روش‌های با ناظر (برای مثال الگوریتم‌های دسته بندی) کل مجموعه داده‌ها به دو بخش مجموعه داده‌های آموزشی و مجموعه داده‌های آزمایشی تقسیم می‌شود. در مرحله یادگیری (آموزش) مدل، الگوریتم براساس مجموعه داده‌های آموزشی یک مدل می‌سازد که شکل مدل ساخته شده به الگوریتم یادگیرنده مورد استفاده بستگی دارد. در مرحله ارزیابی براساس مجموعه داده‌های آزمایشی دقت و کارائی مدل ساخته شده بررسی می‌شود. توجه داشته باشید که مجموعه داده‌های آزمایشی برای مدل ساخته شده پیش از این ناشناخته هستند.

در مرحله یادگیری مدل؛ برای مقابله با مشکل به خاطرسپاری (Memorization) مجموعه داده‌های آموزشی، در برخی موارد بخشی از مجموعه داده‌های آموزشی را از آن مجموعه جدا می‌کنند که با عنوان مجموعه داده ارزیابی (Valid Dataset) شناسائی می‌شود. استفاده از مجموعه داده ارزیابی باعث می‌شود که مدل ساخته شده، مجموعه داده‌های آموزشی را حقیقتاً یاد بگیرد و در پی به خاطرسپاری و حفظ آن نباشد. به بیان دیگر در مرحله یادگیری مدل؛ تا قبل از رسیدن به لحظه ای، مدل در حال یادگیری و کلی سازی (Generalization) است و از آن لحظه به بعد در حال به خاطرسپاری (Over Fitting) مجموعه داده‌های آموزشی است. بدیهی است به خاطرسپاری باعث افزایش دقت مدل برای مجموعه داده‌های آموزشی و بطور مشابه باعث کاهش دقت مدل برای مجموعه داده‌های آزمایشی می‌شود. بدین منظور جهت جلوگیری از مشکل به خاطرسپاری از مجموعه داده ارزیابی استفاده می‌شود که به شکل غیر مستقیم در فرآیند یادگیری مدل، وارد عمل می‌شوند. بدین ترتیب مدلی که مفهومی را از داده‌های آموزشی فرا گرفته، نسبت به مدلی که صرفاً داده‌های آموزشی را به خوبی حفظ کرده است، برای مجموعه داده آزمایشی دقت به مراتب بالاتری دارد. این حقیقت در بیشتر فرآیندهای آموزشی که از مجموعه داده ارزیابی بهره می‌گیرند قابل مشاهده است.

در روش‌های بدون ناظر یا روش‌های توصیفی (برای مثال خوشه بندی) الگوریتم‌ها فاقد مراحل آموزشی و آزمایشی هستند و در پایان عملیات یادگیری مدل، مدل ساخته شده به همراه کارائی آن به عنوان خروجی ارائه می‌شود، برای مثال در الگوریتم‌های خوشه بندی خروجی همان خوشه‌های ایجاد شده هستند و یا خروجی در روش کشف قوانین انجمنی عبارت است از مجموعه ای از قوانین «اگر- آنگاه» که بیانگر ارتباط میان رخداد توامان مجموعه ای از اشیاء با یکدیگر می‌باشد.

در این قسمت عملیات ساخت مدل در فرآیند داده کاوی برای سه روش دسته بندی، خوشه بندی و کشف قوانین انجمنی ارائه می‌شود. بدیهی است برای هر کدام از این روش‌ها علاوه بر الگوریتم‌های معرفی شده، الگوریتم‌های متنوعی دیگری نیز وجود دارد. در ادامه سعی می‌شود به صورت کلان به فلسفه یادگیری مدل پرداخته شود. فهرست مطالب به شرح زیر است:

-1 دسته بندی:

-1-1 دسته بندی مبتنی بر درخت تصمیم (Decision Tree based methods) :

-1-2 دسته بندهای مبتنی بر قانون (Rule based methods) :

-1-3 دسته بندهای مبتنی بر نظریه بیز (Naïve Bayes and Bayesian belief networks) :

-2 خوشه بندی:

-2-1 خوشه بندی افرازی (Centroid Based Clustering) :

-2-1-1 الگوریتم خوشه بندی K-Means :

-2-1-2 الگوریتم خوشه بندی K-Medoids :

-2-1-3 الگوریتم خوشه بندی Bisecting K-Means :

-2-1-4 الگوریتم خوشه بندی Fuzzy C-Means :

-2-2 خوشه بندی سلسله مراتبی (Hierarchical Clustering) Connectivity Based Clustering :

-2-2-1 روش‌های خوشه بندی تجمیعی (Agglomerative Clustering) :

-2-2-2 روش‌های خوشه بندی تقسیمی (Divisive Clustering) :

-2-3 خوشه بندی مبتنی بر چگالی (Density Based Clustering) :

-3 کشف قوانین انجمنی :

-3-1 الگوریتم های Brute-Force ، Apriori و FP-Growth:

-1 دسته بندی:

در الگوریتم‌های دسته بندی، برای هر یک از رکوردهای مجموعه داده مورد کاوش، یک برچسب که بیانگر حقیقتی از مساله است تعریف می‌شود و هدف الگوریتم یادگیری؛ یافتن نظم حاکم بر این برچسب هاست. به بیان دیگر در مرحله آموزش؛ مجموعه داده‌های آموزشی به یکی از الگوریتم‌های دسته بندی داده می‌شود تا بر اساس سایر ویژگی‌ها برای مقادیر ویژگی دسته، مدل ساخته شود. سپس در مرحله ارزیابی؛ دقت مدل ساخته شده به کمک مجموعه داده‌های آزمایشی ارزیابی خواهد شد. انواع گوناگون الگوریتم‌های دسته بندی را می‌توان بصورت ذیل برشمرد:

-1-1 دسته بندی مبتنی بر درخت تصمیم (Decision Tree based methods):

از مشهورترین روش‌های ساخت مدل دسته بندی می‌باشد که دانش خروجی را به صورت یک درخت از حالات مختلف مقادیر ویژگی‌ها ارائه می‌کند. بدین ترتیب دسته بندی‌های مبتنی بر درخت تصمیم کاملاً قابل تفسیر می‌باشند. در حالت کلی درخت تصمیم بدست آمده برای یک مجموعه داده آموزشی؛ واحد و یکتا نیست. به بیان دیگر براساس یک مجموعه داده، درخت‌های تصمیم مختلفی می‌توان بدست آورد. عموماً به منظور فراهم نمودن اطلاعات بیشتری از داده ها، از میان ویژگی‌های موجود یک Case ابتدا آنهایی که دارای خاصیت جداکنندگی بیشتری هستند انتخاب می‌شوند. در واقع براساس مجموعه داده‌های آموزشی از میان ویژگی ها، یک ویژگی انتخاب می‌شود و در ادامه مجموعه رکوردها براساس مقدار این ویژگی شکسته می‌شود و این فرآیند ادامه می‌یابد تا درخت کلی ساخته شود. پس از ساخته شدن مدل، می‌توان آن را بر روی مجموعه داده‌های آزمایشی اعمال (Apply) نمود. منظور از اعمال کردن مدل، پیش بینی مقدار ویژگی یک دسته برای یک رکورد آزمایشی براساس مدل ساخته شده است. توجه شود هدف پیش بینی ویژگی دسته این رکورد، براساس درخت تصمیم موجود است.

بطور کلی الگوریتم‌های تولید درخت تصمیم مختلفی از جمله CART، ID3، C4.5، SLIQ، SPRINT و HUNT وجود دارد. این الگوریتم‌ها به لحاظ استفاده از روش‌های مختلف جهت انتخاب ویژگی و شرط توقف در ساخت درخت با یکدیگر تفاوت دارند. عموماً الگوریتم‌های درخت تصمیم برای شناسائی بهترین شکست، از یک مکانیزم حریصانه (Greedy) استفاده می‌کنند که براساس آن شکستی که توزیع دسته‌ها در گره‌های حاصل از آن همگن باشد، نسبت به سایر شکست‌ها بهتر خواهد بود. منظور از همگن بودن گره این است که همه رکوردهای موجود در آن متعلق به یک دسته خاص باشند، بدین ترتیب آن گره به برگ تبدیل خواهد شد. بنابراین گره همگن گره ای است که کمترین میزان ناخالصی (Impurity) را دارد. به بیان دیگر هر چه توزیع دسته‌ها در یک گره همگن‌تر باشد، آن گره ناخالصی کمتری خواهد داشت. سه روش مهم برای محاسبه ناخالصی گره وجود دارد که عبارتند از: ضریب GINI، روش Entropy و Classification Error.

از مزایای درخت تصمیم می‌توان به توانایی کار با داده‌های گسسته و پیوسته، سهولت در توصیف شرایط (با استفاده از منطق بولی) در درخت تصمیم، عدم نیاز به تابع تخمین توزیع، کشف روابط غیرمنتظره یا نامعلوم و ... اشاره نمود.

همچنین از معایب درخت تصمیم نسبت به دیگر روش‌های داده کاوی می‌توان این موارد را برشمرد: تولید درخت تصمیم گیری هزینه بالائی دارد، در صورت همپوشانی گره‌ها تعداد گره‌های پایانی زیاد می‌شود، طراحی درخت تصمیم گیری بهینه دشوار است، احتمال تولید روابط نادرست وجود دارد و ... .

می‌توان موارد استفاده از دسته بند درخت تصمیم نسبت به سایر دسته بندی کننده‌های تک مرحله ای رایج را؛ حذف محاسبات غیر ضروری و انعطاف پذیری در انتخاب زیر مجموعه‌های مختلفی از صفات برشمرد. در نهایت از جمله مسائل مناسب برای یادگیری درخت تصمیم، می‌توان به مسائلی که در آنها نمونه‌ها به شکل جفت‌های «صفت-مقدار» بازنمائی می‌شود و همچنین مسائلی که تابع هدف، مقادیر خروجی گسسته دارد اشاره نمود.

-1-2 دسته بندهای مبتنی بر قانون (Rule based methods):

این دسته بندها دانش خروجی خود را به صورت یک مجموعه از قوانین «اگر-آنگاه» نشان می‌دهند. هر قانون یک بخش شرایط (LHS: Left Hand Side) و یک بخش نتیجه (RHS: Right Hand Side) دارد. بدیهی است اگر تمام شرایط مربوط به بخش مقدم یک قانون درباره یک رکورد خاص درست تعبیر شود، آن قانون آن رکورد را پوشش می‌دهد. دو معیار *Accuracy* و *Coverage* برای هر قانون قابل محاسبه است که هر چه میزان این دو معیار برای یک قانون بیشتر باشد، آن قانون؛ قانونی با ارزش‌تر محسوب می‌شود.

Coverage یک قانون، برابر با درصد رکوردهایی است که بخش شرایط قانون مورد نظر در مورد آنها صدق می‌کند و درست تعبیر می‌شود. بنابراین هر چه این مقدار بیشتر باشد آن قانون، قانونی کلی‌تر و عمومی‌تر می‌باشد.

Accuracy یک قانون بیان می‌کند که در میان رکوردهایی که بخش شرایط قانون در مورد آنها صدق می‌کند، چند درصد هر دو قسمت قانون مورد نظر در مورد آنها صحیح است.

چنانچه مجموعه همه رکوردها را در نظر بگیریم؛ مطلوب‌ترین حالت این است که همواره یک رکورد توسط یک و تنها یک قانون

پوشش داده شود، به بیان دیگر مجموعه قوانین نهایی به صورت جامع (Exhaustive Rules) و دو به دو ناسازگار (Mutually Exclusive Rules) باشند. جامع بودن به معنای این است که هر رکورد *حداقل* توسط یک قانون پوشش داده شود و معنای قوانین مستقل یا دو به دو ناسازگار بودن بدین معناست که هر رکورد *حداکثر* توسط یک قانون پوشش داده شود.
مجموعه قوانین و درخت تصمیم عیناً یک مجموعه دانش را نشان می‌دهند و تنها در شکل نمایش متفاوت از هم هستند. البته روش‌های مبتنی بر قانون انعطاف پذیری و تفسیرپذیری بالاتری نسبت به روش‌های مبتنی بر درخت دارند. همچنین اجباری در تعیین وضعیت هایی که در یک درخت تصمیم برای ترکیب مقادیر مختلف ویژگی‌ها رخ می‌دهد ندارند و از این رو دانش خلاصه‌تری ارائه می‌دهند.

-1-3 دسته بندهای مبتنی بر نظریه بیز (Naïve Bayes and Bayesian belief networks):

دسته بند مبتنی بر رابطه نظریه بیز (Naïve Bayes) از یک چهارچوب احتمالی برای حل مسائل دسته بندی استفاده می‌کند. براساس نظریه بیز رابطه **I** برقرار است:

$$P(C|A) = \frac{P(A|C)*P(C)}{P(A)} : \mathrm{I}$$

$$\mathrm{P}\ (\mathrm{C}|\ \mathrm{A}_1, \mathrm{A}_2, \mathrm{A}_3, \dots, \mathrm{A}_n) : \mathrm{II}$$

هدف محاسبه دسته یک رکورد مفروض با مجموعه ویژگی‌های (A1,A2,A3,...,An) می‌باشد. در واقع از بین دسته‌های موجود به دنبال پیدا کردن دسته ای هستیم که مقدار **II** را بیشینه کند. برای این منظور این احتمال را برای تمامی دسته‌های مذکور محاسبه نموده و دسته ای که مقدار این احتمال به ازای آن بیشینه شود را به عنوان دسته رکورد جدید در نظر می‌گیریم. ذکر این نکته ضروری است که بدانیم نحوه محاسبه برای ویژگی‌های گسسته و پیوسته متفاوت می‌باشد.

-2 خوشه بندی:

خوشه را مجموعه ای از داده‌ها که به هم شباهت دارند تعریف می‌کنند و هدف از انجام عملیات خوشه بندی فهم (Understanding) گروه رکوردهای مشابه در مجموعه داده‌ها و همچنین خلاصه سازی (Summarization) یا کاهش اندازه‌ی مجموعه داده‌های بزرگ می‌باشد. خوشه بندی از جمله روش هایی است که در آن هیچ گونه برچسبی برای رکوردها در نظر گرفته نمی‌شود و رکوردها تنها براساس معیار شباهتی که معرفی شده است، به مجموعه ای از خوشه‌ها گروه بندی می‌شوند. عدم استفاده از برچسب موجب می‌شود الگوریتم‌های خوشه بندی جزء روش‌های بدون ناظر محسوب شوند و همانگونه که پیشتر ذکر آن رفت در خوشه بندی تلاش می‌شود تا داده‌ها به خوشه هایی تقسیم شوند که شباهت بین داده ای درون هر خوشه بیشینه و بطور مشابه شباهت بین داده‌ها در خوشه‌های متفاوت کمینه شود.
چنانچه بخواهیم خوشه بندی و دسته بندی را مقایسه کنیم، می‌توان بیان نمود که در دسته بندی هر داده به یک دسته (طبقه) از پیش مشخص شده تخصیص می‌یابد ولی در خوشه بندی هیچ اطلاعی از خوشه‌ها وجود ندارد و به عبارتی خود خوشه‌ها نیز از داده‌ها استخراج می‌شوند. به بیان دیگر در دسته بندی مفهوم دسته در یک حقیقت خارجی نهفته است حال آنکه مفهوم خوشه در نهان فواصل میان رکورد هاست. مشهورترین تقسیم بندی الگوریتم‌های خوشه بندی به شرح زیر است:

-1-2 خوشه بندی افرازی (Centroid Based Clustering) :

تقسیم مجموعه داده‌ها به زیرمجموعه‌های بدون همپوشانی، به طریقی که هر داده دقیقاً در یک زیر مجموعه قرار داشته باشد. این الگوریتم‌ها بهترین عملکرد را برای مسائل با خوشه‌های به خوبی جدا شده از خود نشان می‌دهند. از الگوریتم‌های افرازی می‌توان به موارد زیر اشاره نمود:

2-1-1- الگوریتم خوشه بندی K-Means :

در این الگوریتم عملاً مجموعه داده‌ها به تعداد خوشه‌های از پیش تعیین شده تقسیم می‌شوند. در واقع فرض می‌شود که تعداد خوشه‌ها از ابتدا مشخص می‌باشند. ایده اصلی در این الگوریتم تعریف K مرکز برای هر یک از خوشه‌ها است. بهترین انتخاب برای مراکز خوشه‌ها قرار دادن آنها (مراکز) در فاصله هر چه بیشتر از یکدیگر می‌باشد. پس از آن هر رکورد در مجموعه داده به نزدیکترین مرکز خوشه تخصیص می‌یابد. معیار محاسبه فاصله در این مرحله هر معیاری می‌تواند باشد. این معیار با ماهیت مجموعه داده ارتباط تنگاتنگی دارد. مشهورترین معیارهای محاسبه فاصله رکوردها در روش خوشه بندی معیار فاصله اقلیدسی و فاصله همینگ می‌باشد. لازم به ذکر است در وضعیتی که انتخاب مراکز اولیه خوشه‌ها به درستی انجام نشود، خوشه‌های حاصل در پایان اجرای الگوریتم کیفیت مناسبی نخواهند داشت. بدین ترتیب در این الگوریتم جواب نهائی به انتخاب مراکز اولیه خوشه‌ها وابستگی زیادی دارد که این الگوریتم فاقد روالی مشخص برای محاسبه این مراکز می‌باشد. امکان تولید خوشه‌های خالی توسط این الگوریتم از دیگر معایب آن می‌باشد.

2-1-2- الگوریتم خوشه بندی K-Medoids :

این الگوریتم برای حل برخی مشکلات الگوریتم K-Means پیشنهاد شده است، که در آن بجای کمینه نمودن مجموع مجذور اقلیدسی فاصله بین نقاط (که معمولاً به عنوان تابع هدف در الگوریتم K-Means مورد استفاده قرار می‌گیرد)، مجموع تفاوت‌های فواصل جفت نقاط را کمینه می‌کنند. همچنین بجای میانگین گیری برای یافتن مراکز جدید در هر تکرار حلقه یادگیری مدل، از میانه مجموعه اعضای هر خوشه استفاده می‌کنند.

2-1-3- الگوریتم خوشه بندی Bisecting K-Means :

ایده اصلی در این الگوریتم بدین شرح است که برای بدست آوردن K خوشه، ابتدا کل نقاط را به شکل یک خوشه در نظر می‌گیریم و در ادامه مجموعه نقاط تنها خوشه موجود را به دو خوشه تقسیم می‌کنیم. پس از آن یکی از خوشه‌های بدست آمده را برای شکسته شدن انتخاب می‌کنیم و تا زمانی که K خوشه را بدست آوریم این روال را ادامه می‌دهیم. بدین ترتیب مشکل انتخاب نقاط ابتدایی را که در الگوریتم K-Means با آن مواجه بودیم نداشته و بسیار کاراتر از آن می‌باشد.

2-1-4- الگوریتم خوشه بندی Fuzzy C-Means:

کارائی این الگوریتم نسبت به الگوریتم K-Means کاملاً بالاتر می‌باشد و دلیل آن به نوع نگاهی است که این الگوریتم به مفهوم خوشه و اعضای آن دارد. در واقع نقطه قوت الگوریتم Fuzzy C-Means این است که الگوریتمی همواره همگراست. در این الگوریتم تعداد خوشه‌ها برابر با C بوده (مشابه الگوریتم K-Means) ولی برخلاف الگوریتم K-Means که در آن هر رکورد تنها به یکی از خوشه‌های موجود تعلق دارد، در این الگوریتم هر کدام از رکوردهای مجموعه داده به تمامی خوشه‌ها متعلق است. البته این میزان تعلق با توجه به عددی که درجه عضویت تعلق هر رکورد را نشان می‌دهد، مشخص می‌شود. بدین ترتیب عملاً تعلق فازی هر رکورد به تمامی خوشه‌ها سبب خواهد شد که امکان حرکت ملایم عضویت هر رکورد به خوشه‌های مختلف امکان پذیر شود. بنابراین در این الگوریتم امکان تصحیح خطای تخصیص ناصحیح رکوردها به خوشه‌ها ساده‌تر می‌باشد و مهم‌ترین نقطه ضعف این الگوریتم در قیاس با K-Means زمان محاسبات بیشتر آن می‌باشد. می‌توان پذیرفت که از سرعت در عملیات خوشه بندی در برابر رسیدن به دقت بالاتر می‌توان صرفه نظر نمود.

2-2- خوشه بندی سلسله مراتبی (Hierarchical Clustering) Connectivity Based Clustering:

در پایان این عملیات یک مجموعه از خوشه‌های تودرتو به شکل سلسله مراتبی و در قالب ساختار درختی خوشه بندی بدست می‌آید که با استفاده از نمودار Dendrogram چگونگی شکل گیری خوشه‌های تودرتو را می‌توان نمایش داد. این نمودار درخت مانند، ترتیبی از ادغام و تجزیه را برای خوشه‌های تشکیل شده ثبت می‌کند، یکی از نقاط قوت این روش عدم اجبار برای تعیین تعداد خوشه‌ها می‌باشد (بر خلاف خوشه بندی افرازی). الگوریتم‌های مبتنی بر خوشه بندی سلسله مراتبی به دو دسته مهم تقسیم بندی می‌شوند:

2-2-1- روش‌های خوشه بندی تجمیعی (Agglomerative Clustering) :

با نقاطی به عنوان خوشه‌های منحصر به فرد کار را آغاز نموده و در هر مرحله، به ادغام خوشه‌های نزدیک به یکدیگر می‌پردازیم، تا زمانی که تنها یک خوشه باقی بماند.

عملیات کلیدی در این روش، چگونگی محاسبه میزان مجاورت دو خوشه است و روش‌های متفاوت تعریف فاصله بین خوشه‌ها باعث تمایز الگوریتم‌های مختلف مبتنی بر ایده خوشه بندی تجمیعی است. برخی از این الگوریتم‌ها عبارتند از: خوشه بندی تجمیعی

- کمینه ای، خوشه بندی تجمیعی - بیشینه ای، خوشه بندی تجمیعی - میانگینی، خوشه بندی تجمیعی - مرکزی.

-2-2-2 روش های خوشه بندی تقسیمی (Divisive Clustering) :
با یک خوشه‌ی دربرگیرنده‌ی همه نقاط کار را آغاز نموده و در هر مرحله، خوشه را می‌شکنیم تا زمانی که K خوشه بدست آید و یا در هر خوشه یک نقطه باقی بماند.

-2-3 خوشه بندی مبتنی بر چگالی (Density Based Clustering):
تقسیم مجموعه داده به زیرمجموعه هایی که چگالی و چگونگی توزیع رکوردها در آنها لحاظ می‌شود. در این الگوریتم مهمترین فاکتور که جهت تشکیل خوشه‌ها در نظر گرفته می‌شود، تراکم و یا چگالی نقاط می‌باشد. بنابراین برخلاف دیگر روش‌های خوشه بندی که در آنها تراکم نقاط اهمیت نداشت، در این الگوریتم سعی می‌شود تنوع فاصله هایی که نقاط با یکدیگر دارند، در عملیات خوشه بندی مورد توجه قرار گیرد. الگوریتم **DBSCAN** مشهورترین الگوریتم خوشه بندی مبتنی بر چگالی است.

به طور کلی عملکرد یک الگوریتم خوشه بندی نسبت به الگوریتم‌های دیگر، بستگی کاملی به ماهیت مجموعه داده و معنای آن دارد.

-3 کشف قوانین انجمنی :
الگوریتم‌های کاشف قوانین انجمنی نیز همانند الگوریتم‌های خوشه بندی به صورت روش‌های توصیفی یا بدون ناظر طبقه بندی می‌شوند. در این الگوریتم‌ها بدنبال پیدا کردن یک مجموعه از قوانین وابستگی یا انجمنی در میان تراکنش‌ها (برای مثال تراکنشهای خرید در فروشگاه، تراکنشهای خرید و فروش سهام در بورس و ...) هستیم تا براساس قوانین کشف شده بتوان میزان اثرگذاری اشیایی را بر وجود مجموعه اشیاء دیگری بدست آورد. خروجی در این روش کاوش، به صورت مجموعه ای از قوانین «اگر-آنگاه» است، که بیانگر ارتباطات میان رخداد توامان مجموعه ای از اشیاء با یکدیگر می‌باشد. به بیان دیگر این قوانین می‌تواند به پیش بینی وقوع یک مجموعه اشیاء مشخص در یک تراکنش، براساس وقوع اشیاء دیگر موجود در آن تراکنش بپردازد. ذکر این نکته ضروری است که بدانیم قوانین استخراج شده تنها استلزام یک ارتباط میان وقوع توامان مجموعه ای از اشیاء را نشان می‌دهد و در مورد چرایی یا همان علیت این ارتباط سخنی به میان نمی‌آورد. در ادامه به معرفی مجموعه ای از تعاریف اولیه در این مبحث می‌پردازیم (در تمامی تعاریف تراکنش‌های سبد خرید مشتریان در یک فروشگاه را به عنوان مجموعه داده مورد کاوش در نظر بگیرید):
• **مجموعه اشیاء:** مجموعه ای از یک یا چند شیء. منظور از مجموعه اشیاء K عضوی، مجموعه ای است که شامل K شیء باشد.
برای مثال:{مسواک، نان، شیر}
• **تعداد پشتیبانی (Support Count) :** فراوانی وقوع مجموعه‌ی اشیاء در تراکنش‌های موجود که آنرا با حرف **σ** نشان می‌دهیم.
برای مثال: 2=({مسواک، نان، شیر})σ
• **مجموعه اشیاء مکرر (Frequent Item Set) :** مجموعه ای از اشیاء که تعداد پشتیبانی آنها بزرگتر یا مساوی یک مقدار آستانه (Min Support Threshold) باشد، مجموعه اشیاء مکرر نامیده می‌شود.
• **قوانین انجمنی:** بیان کننده ارتباط میان اشیاء در یک مجموعه از اشیاء مکرر. این قوانین معمولاً به شکل X=>Y هستند.
برای مثال:{نوشابه}=>{مسواک، شیر}

مهمترین معیارهای ارزیابی قوانین انجمنی عبارتند از:
• **Support:** کسری از تراکنش‌ها که حاوی همه اشیاء یک مجموعه اشیاء خاص هستند و آنرا با حرف S نشان می‌دهند.
برای مثال: 2.2=({نان، شیر})S
• **Confidence:** کسری از تراکنش‌های حاوی همه اشیاء بخش شرطی قانون انجمنی که صحت آن قانون را نشان می‌دهد که با آنرا حرف C نشان می‌دهند. برخلاف Support نمی‌توانیم مثالی برای اندازه گیری Confidence یک مجموعه اشیاء بیاوریم زیرا این معیار تنها برای قوانین انجمنی قابل محاسبه است.

با در نظر گرفتن قانون X=>Y می‌توان Support را کسری از تراکنش هایی دانست که شامل هر دو مورد X و Y هستند و Confidence برابر با اینکه چه کسری از تراکنش هایی که Y را شامل می‌شوند در تراکنش هایی که شامل X نیز هستند، ظاهر می‌شوند. هدف از کاوش قوانین انجمنی پیدا کردن تمام قوانین Rx است که از این دستورات تبعیت می‌کند:

$$\text{I} \qquad \text{Support}\,(R_x) \geq \text{Supp}_{MIN}$$

$$\text{II} \qquad \text{Confidence}\,(R_x) \geq \text{Conf}_{MIN}$$

$$\text{III} \qquad 3^d\text{-}2^{d+1}\text{+}1$$

در این دستورات منظور از SuppMIN و ConfMIN به ترتیب عبارت است از کمترین مقدار برای Support و Confidence که بایست جهت قبول هر پاسخ نهائی به عنوان یک قانون با ارزش مورد توجه قرار گیرد. کلیه قوانینی که از مجموعه اشیاء مکرر یکسان ایجاد می‌شوند دارای مقدار Support مشابه هستند که دقیقاً برابر با تعداد پشتیبانی یا همان σ شیء مکرری است که قوانین انجمنی با توجه به آن تولید شده اند. به همین دلیل فرآیند کشف قوانین انجمنی را می‌توان به دو مرحله مستقل «تولید مجموعه اشیاء مکرر» و «تولید قوانین انجمنی مطمئن» تقسیم نمائیم.

در مرحله نخست، تمام مجموعه اشیاء که دارای مقدار Support ≥ SuppMIN می‌باشند را تولید می‌کنیم. رابطه **I**

در مرحله دوم با توجه به مجموعه اشیاء مکرر تولید شده، قوانین انجمنی با اطمینان بالا بدست می‌آیند که همگی دارای شرط Confidence ≥ ConfMIN هستند. رابطه **II**

3-1- الگوریتم های Brute-Force ، Apriori و FP-Growth:

یک روش تولید اشیاء مکرر روش **Brute-Force** است که در آن ابتدا تمام قوانین انجمنی ممکن لیست شده، سپس مقادیر Support و Confidence برای هر قانون محاسبه می‌شود. در نهایت قوانینی که از مقادیر آستانه‌ی SuppMIN و ConfMIN تبعیت نکنند، حذف می‌شوند. تولید مجموعه اشیاء مکرر بدین طریق کاری بسیار پرهزینه و پیچیده ای می‌باشد، در واقع روش‌های هوشمندانه دیگری وجود دارد که پیچیدگی بالای روش Brute-Force را ندارند زیرا کل شبکه مجموعه اشیاء را به عنوان کاندید در نظر نمی‌گیرند. همانند تولید مجموعه اشیاء مکرر، تولید مجموعه قوانین انجمنی نیز بسیار پرهزینه و گران است.

چنانچه یک مجموعه اشیاء مکرر مشخص با d شیء را در نظر بگیریم، تعداد کل قوانین انجمنی قابل استخراج از رابطه **III** محاسبه می‌شود. (برای مثال تعداد قوانین انجمنی قابل استخراج از یک مجموعه شیء 6 عضوی برابر با 602 قانون می‌باشد، که با توجه به رشد d؛ سرعت رشد تعداد قوانین انجمنی بسیار بالا می‌باشد.)

الگوریتم‌های متعددی برای تولید مجموعه اشیاء مکرر وجود دارد برای نمونه الگوریتم‌های **Apriori** و **FP-Growth** که در هر دوی این الگوریتم ها، ورودی الگوریتم لیست تراکنش‌ها و پارامتر SuppMIN می‌باشد. الگوریتم Apriori روشی هوشمندانه برای یافتن مجموعه اشیاء تکرار شونده با استفاده از روش تولید کاندید است که از یک روش بازگشتی برای یافتن مجموعه اشیاء مکرر استفاده می‌کند. مهمترین هدف این الگوریتم تعیین مجموعه اشیاء مکرری است که تعداد تکرار آنها حداقل برابر با SuppMIN باشد. ایده اصلی در الگوریتم Apriori این است که اگر مجموعه اشیایی مکرر باشد، آنگاه تمام زیر مجموعه‌های آن مجموعه اشیاء نیز باید مکرر باشند. در واقع این اصل همواره برقرار است زیرا Support یک مجموعه شیء هرگز بیشتر از Support زیرمجموعه‌های آن مجموعه شیء نخواهد بود. مطابق با این ایده تمام ابرمجموعه‌های مربوط به مجموعه شیء نامکرر از شبکه مجموعه اشیاء حذف خواهند شد (هرس می‌شوند). هرس کردن مبتنی بر این ایده را هرس کردن بر پایه Support نیز عنوان می‌کنند که باعث کاهش قابل ملاحظه ای از تعداد مجموعه‌های کاندید جهت بررسی (تعیین مکرر بودن یا نبودن مجموعه اشیاء) می‌شود.

الگوریتم FP-Growth در مقایسه با Apriori روش کارآمدتری برای تولید مجموعه اشیاء مکرر ارائه می‌دهد. این الگوریتم با ساخت یک درخت با نام FP-Tree سرعت فرآیند تولید اشیاء مکرر را به طور چشمگیری افزایش می‌دهد، در واقع با یکبار مراجعه به مجموعه تراکنش‌های مساله این درخت ساخته می‌شود. پس از ساخته شدن درخت با توجه به ترتیب نزولی Support مجموعه اشیاء تک عضوی (یعنی مجموعه اشیاء) مساله تولید مجموعه اشیاء مکرر به چندین زیر مسئله تجزیه می‌شود، که هدف در هر کدام از این زیر مساله ها، یافتن مجموعه اشیاء مکرری است که به یکی از آن اشیاء ختم خواهند شد.

الگوریتم Aprior علاوه بر تولید مجموعه اشیاء مکرر، اقدام به تولید مجموعه قوانین انجمنی نیز می‌نماید. در واقع این الگوریتم با استفاده از مجموعه اشیاء مکرر بدست آمده از مرحله قبل و نیز پارامتر ConfMIN قوانین انجمنی مرتبط را که دارای درجه اطمینان

بالائی هستند نیز تولید می‌کند. به طور کلی Confidence دارای خصوصیت هماهنگی (Monotone) نیست ولیکن Confidence قوانینی که از مجموعه اشیاء یکسانی بوجود می‌آیند دارای خصوصیت ناهماهنگی هستند. بنابراین با هرس نمودن کلیه ابرقوانین انجمنی یک قانون انجمنی یا Confidence (Rx) ≥ ConfMIN در شبکه قوانین انجمنی (مشابه با شبکه مجموعه اشیاء) اقدام به تولید قوانین انجمنی می‌نمائیم. پس از آنکه الگوریتم با استفاده از روش ذکر شده، کلیه قوانین انجمنی با اطمینان بالا را در شبکه قوانین انجمنی یافت، اقدام به الحاق نمودن آن دسته از قوانین انجمنی می‌نماید که پیشوند یکسانی را در توالی قانون به اشتراک می‌گذارند و بدین ترتیب قوانین کاندید تولید می‌شوند.

جهت آشنائی بیشتر به List of machine learning concepts مراجعه نمائید.

عنوان: **ارزیابی و تفسیر مدل در داده کاوی**
نویسنده: محمد رجبی
تاریخ: ۱۳۹۳/۰۹/۱۱ ۱۸:۰
آدرس: www.dotnettips.info
گروه‌ها: Analysis Services, data mining, Microsoft SQL Server

مقدمه

دانشی که در مرحله یادگیری مدل تولید می‌شود، می‌بایست در مرحله ارزیابی مورد تحلیل قرار گیرد تا بتوان ارزش آن را تعیین نمود و در پی آن کارائی الگوریتم یادگیرنده مدل را نیز مشخص کرد. این معیارها را می‌توان هم برای مجموعه داده‌های آموزشی در مرحله یادگیری و هم برای مجموعه رکوردهای آزمایشی در مرحله ارزیابی محاسبه نمود. همچنین لازمه موفقیت در بهره مندی از علم داده کاوی تفسیر دانش تولید و ارزیابی شده است.

ارزیابی در الگوریتم‌های دسته بندی

برای سادگی معیارهای ارزیابی الگوریتم‌های دسته بندی، آنها را برای یک مسئله با دو دسته ارائه خواهیم نمود. در ابتدا با مفهوم ماتریس درهم ریختگی (Classification Matrix) آشنا می‌شویم. این ماتریس چگونگی عملکرد الگوریتم دسته بندی را با توجه به مجموعه داده ورودی به تفکیک انواع دسته‌های مساله دسته بندی، نمایش می‌دهد.

		Predicted	
		Positive	Negative
Actual	Positive	True Positives (TP)	False Negatives (FN)
	Negative	False Positives (FP)	True Negatives (TN)

هر یک از عناصر ماتریس به شرح ذیل می‌باشد:

TN: بیانگر تعداد رکوردهایی است که دسته واقعی آنها منفی بوده و الگوریتم دسته بندی نیز دسته آنها را بدرستی منفی تشخیص داده است.

TP: بیانگر تعداد رکوردهایی است که دسته واقعی آنها مثبت بوده و الگوریتم دسته بندی نیز دسته آنها را بدرستی مثبت تشخیص داده است.

FP: بیانگر تعداد رکوردهایی است که دسته واقعی آنها منفی بوده و الگوریتم دسته بندی دسته آنها را به اشتباه مثبت تشخیص داده است.

FN: بیانگر تعداد رکوردهایی است که دسته واقعی آنها مثبت بوده و الگوریتم دسته بندی دسته آنها را به اشتباه منفی تشخیص داده است.

مهمترین معیار برای تعین کارایی یک الگوریتم دسته بندی دقت یا نرخ دسته بندی (Classification Accuracy - Rate) است که این معیار دقت کل یک دسته بند را محاسبه می‌کند. در واقع این معیار مشهورترین و عمومی‌ترین معیار محاسبه کارایی الگوریتم‌های دسته بندی است که نشان می‌دهد، دسته بند طراحی شده چند درصد از کل مجموعه رکوردهای آزمایشی را بدرستی دسته بندی کرده است.

دقت دسته بندی با استفاده از **رابطه I** بدست می‌آید که بیان می‌کند دو مقدار TP و TN مهمترین مقادیری هستند که در یک مسئله دودسته ای باید بیشینه شوند. (در مسائل چند دسته ای مقادیر قرار گرفته روی قطر اصلی این ماتریس - که در صورت کسر محاسبه CA قرار می‌گیرند - باید بیشینه باشند.)

معیار خطای دسته بندی (Error Rate) دقیقاً برعکس معیار دقت دسته بندی است که با استفاده از **رابطه II** بدست می‌آید. کمترین مقدار آن برابر صفر است زمانی که بهترین کارایی را داریم و بطور مشابه بیشترین مقدار آن برابر یک است زمانی که کمترین

کارائی را داریم.
ذکر این نکته ضروری است که در مسائل واقعی، معیار دقت دسته بندی به هیچ عنوان معیار مناسبی برای ارزیابی کارایی الگوریتم‌های دسته بندی نمی‌باشد، به این دلیل که در رابطه دقت دسته بندی، ارزش رکوردهای دسته‌های مختلف یکسان در نظر گرفته می‌شوند. بنابراین در مسائلی که با دسته‌های نامتعادل سروکار داریم، به بیان دیگر در مسائلی که ارزش دسته ای در مقایسه با دسته دیگر متفاوت است، از معیارهای دیگری استفاده می‌شود.
همچنین در مسائل واقعی معیارهای دیگری نظیر DR و FAR که به ترتیب از **روابط III و IV** بدست می‌آیند، اهمیت ویژه ای دارند. این معیارها که توجه بیشتری به دسته بند مثبت نشان می‌دهند، توانایی دسته بند را در تشخیص دسته مثبت و بطور مشابه تاوان این توانایی تشخیص را تبیین می‌کنند. معیار DR نشان می‌دهد که دقت تشخیص دسته مثبت چه مقدار است و معیار FAR نرخ هشدار غلط را با توجه به دسته منفی بیان می‌کند.

$$\text{I:} \qquad CA = \frac{TN+TP}{TN+FN+TP+FP}$$

$$\text{II:} \qquad ER = \frac{FN+FP}{TN+FN+TP+FP} = 1 - CA$$

$$\text{III:} \qquad DR = \frac{TP}{FN+TP}$$

$$\text{IV:} \qquad FAR = \frac{FP}{TN+FP}$$

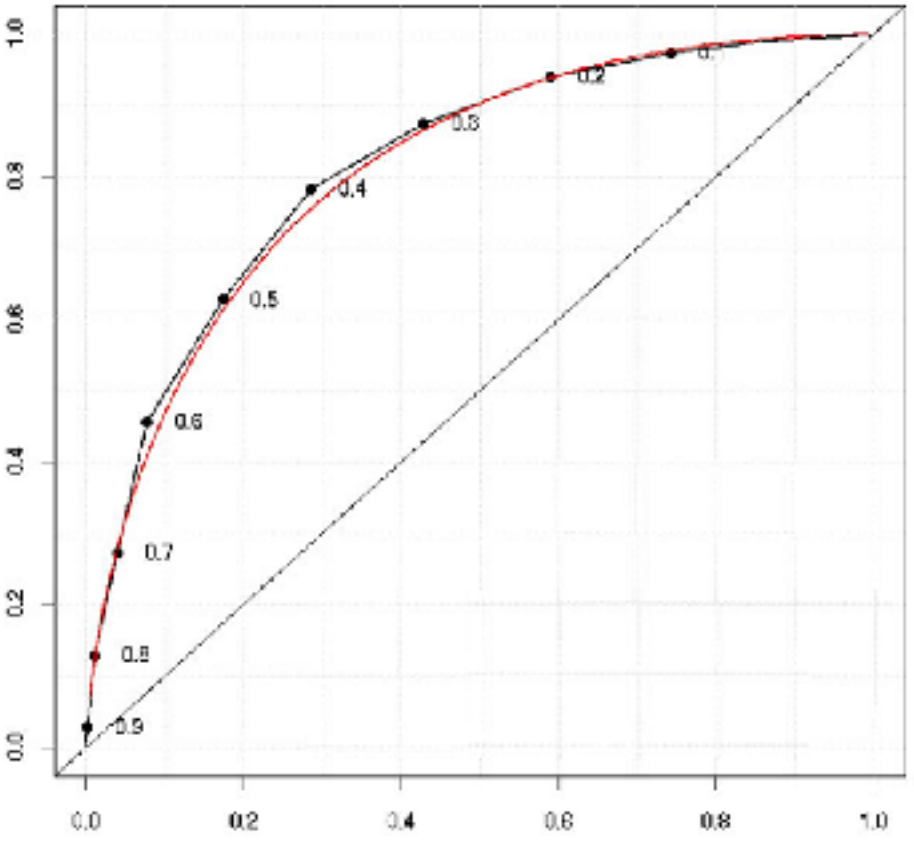
معیار مهم دیگری که برای تعیین میزان کارایی یک دسته بند استفاده می‌شود معیار (Area Under Curve) **AUC** است.

AUC نشان دهنده سطح زیر نمودار (Receiver Operating Characteristic) **ROC** می‌باشد که هر چه مقدار این عدد مربوط به یک دسته بند بزرگتر باشد کارایی نهایی دسته بند مطلوب‌تر ارزیابی می‌شود. نمودار ROC روشی برای بررسی کارایی دسته بندها می‌باشد. در واقع منحنی‌های ROC منحنی‌های دو بعدی هستند که در آنها DR یا همان نرخ تشخیص صحیح دسته مثبت (True Positive Rate - TPR) روی محور Y و بطور مشابه FAR یا همان نرخ تشخیص غلط دسته منفی (False Positive Rate - FPR) روی محور X رسم می‌شوند. به بیان دیگر یک منحنی ROC مصالحه نسبی میان سودها و هزینه‌ها را نشان می‌دهد.

بسیاری از دسته بندها همانند روش‌های مبتنی بر درخت تصمیم و یا روش‌های مبتنی بر قانون، به گونه ای طراحی شده اند که تنها یک خروجی دودویی (مبنی بر تعلق ورودی به یکی از دو دسته ممکن) تولید می‌کنند. به این نوع دسته بندها که تنها یک خروجی مشخص برای هر ورودی تولید می‌کنند، دسته بندهای گسسته گفته می‌شود که این دسته بندها تنها یک نقطه در فضای ROC تولید می‌کنند.
بطور مشابه دسته بندهای دیگری نظیر دسته بندهای مبتنی بر روش بیز و یا شبکه‌های عصبی نیز وجود دارند که یک احتمال و یا امتیاز برای هر ورودی تولید می‌کنند، که این عدد بیانگر درجه تعلق ورودی به یکی از دو دسته موجود می‌باشد. این دسته بندها پیوسته نامیده می‌شوند و بدلیل خروجی خاص این دسته بندها یک آستانه جهت تعیین خروجی نهایی در نظر گرفته می‌شود.

یک منحنی ROC اجازه مقایسه تصویری مجموعه ای از دسته بندی کننده‌ها را می‌دهد، همچنین نقاط متعددی در فضای ROC قابل توجه است. نقطه پایین سمت چپ (0,0) استراتژی را نشان می‌دهد که در یک دسته بند مثبت تولید نمی‌شود. استراتژی مخالف،

که بدون شرط دسته بندهای مثبت تولید می‌کند، با نقطه بالا سمت راست (1,1) مشخص می‌شود. نقطه (0,1) دسته بندی کامل و بی عیب را نمایش می‌دهد. بطور کلی یک نقطه در فضای ROC بهتر از دیگری است اگر در شمال غربی‌تر این فضا قرار گرفته باشد. همچنین در نظر داشته باشید منحنی‌های ROC رفتار یک دسته بندی کننده را بدون توجه به توزیع دسته‌ها یا هزینه خطا نشان می‌دهند، بنابراین کارایی دسته بندی را از این عوامل جدا می‌کنند. فقط زمانی که یک دسته بند در کل فضای کارایی به وضوح بر دسته دیگری تسلط یابد، می‌توان گفت که بهتر از دیگری است. به همین دلیل معیار AUC که سطح زیر نمودار ROC را نشان می‌دهد می‌تواند نقش تعیین کننده ای در معرفی دسته بند برتر ایفا کند. برای درک بهتر نمودار ROC زیر را مشاهده کنید.



مقدار AUC برای یک دسته بند که بطور تصادفی، دسته نمونه مورد بررسی را تعیین می‌کند برابر 0.5 است. همچنین بیشترین مقدار این معیار برابر یک بوده و برای وضعیتی رخ می‌دهد که دسته بند ایده آل بوده و بتواند کلیه نمونه‌های مثبت را بدون هرگونه هشدار غلطی تشخیص دهد. معیار AUC برخلاف دیگر معیارهای تعیین کارایی دسته بندها مستقل از آستانه تصمیم گیری دسته بند می‌باشد. بنابراین این معیار نشان دهنده میزان قابل اعتماد بودن خروجی یک دسته بند مشخص به ازای مجموعه داده‌های متفاوت است که این مفهوم توسط سایر معیارهای ارزیابی کارایی دسته بندها قابل محاسبه نمی‌باشد. در برخی از مواقع سطح زیر منحنی‌های ROC مربوط به دو دسته بند با یکدیگر برابر است ولی ارزش آنها برای کاربردهای مختلف یکسان نیست که باید در نظر داشت در این گونه مسائل که ارزش دسته‌ها با یکدیگر برابر نیست، استفاده از معیار AUC مطلوب نمی‌باشد. به همین دلیل در این گونه مسائل استفاده از معیار دیگری به جزء هزینه (Cost Matrix) منطقی به نظر نمی‌رسد. در انتها باید توجه نمود در کنار معیارهای بررسی شده که همگی به نوعی دقت دسته بند را محاسبه می‌کردند، در دسته بندهای قابل تفسیر نظیر دسته بندهای مبتنی بر قانون و یا درخت تصمیم، پیچیدگی نهایی و قابل تفسیر بودن مدل یاد گرفته شده نیز از اهمیت بالایی برخوردار است.

از روش‌های ارزیابی الگوریتم‌های دسته بندی (که در این الگوریتم روال کاری بدین صورت است که مدل دسته بندی توسط مجموعه داده آموزشی ساخته شده و بوسیله مجموعه داده آزمایشی مورد ارزیابی قرار می‌گیرد.) می‌توان به روش **Holdout** اشاره کرد که در این روش چگونگی نسبت تقسیم مجموعه داده‌ها (به دو مجموعه داده آموزشی و مجموعه داده آزمایشی) بستگی به تشخیص تحلیلگر دارد که معمولاً دو سوم برای آموزش و یک سوم برای ارزیابی در نظر گرفته می‌شود. مهمترین مزیت این روش

سادگی و سرعت بالای عملیات ارزیابی است ولیکن روش Holdout معایب زیادی دارد از جمله اینکه مجموعه داده‌های آموزشی و آزمایشی به یکدیگر وابسته خواهند شد، در واقع بخشی از مجموعه داده اولیه که برای آزمایش جدا می‌شود، شانسی برای حضور یافتن در مرحله آموزش ندارد و بطور مشابه در صورت انتخاب یک رکورد برای آموزش دیگر شانسی برای استفاده از این رکورد برای ارزیابی مدل ساخته شده وجود نخواهد داشت. همچنین مدل ساخته شده بستگی فراوانی به چگونگی تقسیم مجموعه داده اولیه به مجموعه داده‌های آموزشی و آزمایشی دارد. چنانچه روش Holdout را چندین بار اجرا کنیم و از نتایج حاصل میانگین گیری کنیم از روشی موسوم به **Random Sub-sampling** استفاده نموده ایم. که مهمترین عیب این روش نیز عدم کنترل بر روی تعداد دفعاتی که یک رکورد به عنوان نمونه آموزشی و یا نمونه آزمایشی مورد استفاده قرار می‌گیرد، است. به بیان دیگر در این روش ممکن است برخی رکوردها بیش از سایرین برای یادگیری و یا ارزیابی مورد استفاده قرار گیرند.

چنانچه در روش Random Sub-sampling به شکل هوشمندانه‌تری عمل کنیم به صورتی که هر کدام از رکوردها به تعداد مساوی برای یادگیری و تنها یکبار برای ارزیابی استفاده شوند، روش مزبور در متون علمی با نام **Cross Validation** شناخته می‌شود.

همچنین در روش جامع **k-Fold Cross Validation** کل مجموعه داده‌ها به k قسمت مساوی تقسیم می‌شوند. از k-1 قسمت به عنوان مجموعه داده‌های آموزشی استفاده می‌شود و براساس آن مدل ساخته می‌شود و با یک قسمت باقی مانده عملیات ارزیابی انجام می‌شود. فرآیند مزبور به تعداد k مرتبه تکرار خواهد شد، به گونه ای که از هر کدام از k قسمت تنها یکبار برای ارزیابی استفاده شده و در هر مرتبه یک دقت برای مدل ساخته شده، محاسبه می‌شود. در این روش ارزیابی دقت نهایی دسته بند برابر با میانگین k دقت محاسبه شده خواهد بود. معمول‌ترین مقداری که در متون علمی برای k در نظر گرفته می‌شود برابر با 10 می‌باشد. بدیهی است هر چه مقدار k بزرگتر شود، دقت محاسبه شده برای دسته بند قابل اعتمادتر بوده و دانش حاصل شده جامع‌تر خواهد بود و البته افزایش زمان ارزیابی دسته بند نیز مهمترین مشکل آن می‌باشد. حداکثر مقدار k برابر با تعداد رکوردهای مجموعه داده اولیه است که این روش ارزیابی با نام **Leaving One Out** شناخته می‌شود.

در روش هایی که تاکنون به آن اشاره شده، فرض بر آن است که عملیات انتخاب نمونه‌های آموزشی بدون جایگذاری صورت می‌گیرد. به بیان دیگر یک رکورد تنها یکبار در یک فرآیند آموزشی مورد توجه واقع می‌شود. چنانچه هر رکورد در صورت انتخاب شدن برای شرکت در عملیات یادگیری مدل بتواند مجدداً برای یادگیری مورد استفاده قرار گیرد روش مزبور با نام **Bootstrap** و یا Bootstrap 0.632 شناخته می‌شود. (از آنجا که هر Bootstrap معادل 0.632 مجموعه داده اولیه است)

ارزیابی در الگوریتم‌های خوشه بندی

به منظور ارزیابی الگوریتم‌های خوشه بندی می‌توان آنها به دو دسته تقسیم نمود:

شاخص‌های **ارزیابی بدون ناظر،** که گاهی در متون علمی با نام معیارهای داخلی شناخته می‌شوند، به آن دسته از معیارهایی گفته می‌شود که تعیین کیفیت عملیات خوشه بندی را با توجه به اطلاعات موجود در مجموعه داده بر عهده دارند. در مقابل، معیارهای ارزیابی با ناظر با نام معیارهای خارجی نیز شناخته می‌شوند، که با استفاده از اطلاعاتی خارج از حیطه مجموعه داده‌های مورد بررسی، عملکرد الگوریتم‌های خوشه بندی را مورد ارزیابی قرار می‌دهند.

از آنجا که مهمترین وظیفه یک الگوریتم خوشه بندی آن است که بتواند به بهترین شکل ممکن فاصله درون خوشه ای را کمینه و فاصله بین خوشه ای را بیشینه نماید، کلیه معیارهای ارزیابی بدون ناظر سعی در سنجش کیفیت عملیات خوشه بندی با توجه به دو فاکتور تراکم خوشه ای و جدائی خوشه ای دارند. برآورده شدن هدف کمینه سازی درون خوشه ای و بیشینه سازی میان خوشه ای به ترتیب در گرو بیشینه نمودن تراکم هر خوشه و نیز بیشینه سازی جدایی میان خوشه‌ها می‌باشد. طیف وسیعی از معیارهای ارزیابی بدون ناظر وجود دارد که همگی در ابتدا تعریفی برای فاکتورهای تراکم و جدائی ارائه می‌دهند سپس توسط تابع F(Cohesion, Separation) مرتبط با خود، به ترکیب این دو فاکتور می‌پردازند. ذکر این نکته ضروری است که نمی‌توان هیچ کدام از معیارهای ارزیابی خوشه بندی را برای تمامی کاربردها مناسب دانست.

ارزیابی با ناظر الگوریتم‌های خوشه بندی، با هدف آزمایش و مقایسه عملکرد روش‌های خوشه بندی با توجه به حقایق مربوط به رکوردها صورت می‌پذیرد. به بیان دیگر هنگامی که اطلاعاتی از برچسب رکوردهای مجموعه داده مورد بررسی در اختیار داشته باشیم، می‌توانیم از آنها در عملیات ارزیابی عملکرد الگوریتم‌های خوشه بندی بهره بریم. لازم است در نظر داشته باشید در این بخش از برچسب رکوردها تنها در مرحله ارزیابی استفاده می‌شود و هر گونه بهره برداری از این برچسب‌ها در مرحله یادگیری مدل، منجر به تبدیل شدن روش کاوش داده از خوشه بندی به دسته بندی خواهد شد. مشابه با روش‌های بدون ناظر طیف وسیعی از معیارهای ارزیابی با ناظر نیز وجود دارد که در این قسمت با استفاده از روابط زیر به محاسبه معیارهای **Rand Index** و **Jaccard** می پردازیم به ترتیب در **رابطه I و II** نحوه محاسبه آنها نمایش داده شده است:

$$I: \quad RI = \frac{TP + TN}{TP + FP + TN + FN}$$

$$II: \quad Jaccard = \frac{TP}{TP + FP + TN}$$

Rand Index را می‌توان به عنوان تعداد تصمیمات درست در خوشه بندی در نظر گرفت.
TP : به تعداد زوج داده هایی گفته می‌شود که باید در یک خوشه قرار می‌گرفتند، و قرار گرفته اند.
TN : به تعداد زوج داده هایی گفته می‌شود که باید در خوشه‌های جداگانه قرار داده می‌شدند و به درستی در خوشه‌های جداگانه جای داده شده اند.
FN : به تعداد زوج داده هایی گفته می‌شود که باید در یک خوشه قرار می‌گرفتند ولی در خوشه‌های جداگانه قرار داده شده اند.
FP : به تعداد زوج داده هایی اشاره دارد که باید در خوشه‌های متفاوت قرار می‌گرفتند ولی در یک خوشه قرار گرفته اند.

ارزیابی در الگوریتم‌های کشف قوانین انجمنی

به منظور ارزیابی الگوریتم‌های کشف قوانین انجمنی از آنجایی که این الگوریتم‌ها پتانسیل این را دارند که الگوها و قوانین زیادی تولید نمایند، جهت ارزیابی این قوانین به عواملی همچون شخص استفاده کننده از قوانین و نیز حوزه ای که مجموعه داده مورد بررسی به آن تعلق دارد، وابستگی زیادی پیدا می‌کنیم و بدین ترتیب کار پیدا کردن قوانین جذاب، به آسانی میسر نیست. فرض کنید قانونی با نام R داریم که به شکل **A=>B** می‌باشد، که در آن A و B زیر مجموعه ای از اشیاء می‌باشند.

پیشتر به معرفی دو معیار Support و Confidence پرداختیم. می‌دانیم از نسبت تعداد تراکنش هایی که در آن اشیاء A و B هر دو حضور دارند، به کل تعداد رکوردها **Support** بدست می‌آید که دارای مقداری عددی بین صفر و یک می‌باشد و هر چه این میزان بیشتر باشد، نشان می‌دهد که این دو شیء بیشتر با هم در ارتباط هستند. کاربر می‌تواند با مشخص کردن یک آستانه برای این معیار، تنها قوانینی را بدست آورد که Support آنها بیشتر از مقدار آستانه باشد، بدین ترتیب می‌توان با کاهش فضای جستجو، زمان لازم جهت پیدا کردن قوانین انجمنی را کمینه کرد. البته باید به ضعف این روش نیز توجه داشت که ممکن است قوانین با ارزشی را بدین ترتیب از دست دهیم. در واقع استفاده از این معیار به تنهایی کافی نیست. معیار **Confidence** نیز مقداری عددی بین صفر و یک می‌باشد، که هر چه این عدد بزرگتر باشد بر کیفیت قانون افزوده خواهد شد. استفاده از این معیار به همراه Support مکمل مناسبی برای ارزیابی قوانین انجمنی خواهد بود. ولی مشکلی که همچنان وجود دارد این است که امکان دارد قانونی با Confidence بالا وجود داشته باشد ولی از نظر ما ارزشمند نباشد.

از معیارهای دیگر قوانین انجمنی می‌توان به معیار **Lift** که با نام‌های Intersect Factor یا Interestingness نیز شناخته می‌شود اشاره کرد، که این معیار میزان استقلال میان اشیاء A و B را نشان می‌دهد که می‌تواند مقدار عددی بین صفر تا بی نهایت باشد. در واقع Lift میزان هم اتفاقی بین ویژگی‌ها را در نظر می‌گیرد و میزان رخداد تکی بخش تالی قانون (یعنی شیء B) را در محاسبات خود وارد می‌کند. (بر خلاف معیار Confidence)

مقادیر نزدیک به عدد یک معرف این هستند که A و B مستقل از یکدیگر می‌باشند، بدین ترتیب نشان دهنده قانون جذابی نمی‌باشند. چنانچه این معیار از عدد یک کمتر باشد، نشان دهنده این است که A و B با یکدیگر رابطه منفی دارند. هر چه مقدار این معیار بیشتر از عدد یک باشد، نشان دهنده این است که A اطلاعات بیشتری درباره B فراهم می‌کند که در این حالت جذابیت قانون A=>B بالاتر ارزیابی می‌شود. در ضمن این معیار نسبت به سمت چپ و راست قانون متقارن است در واقع اگر سمت چپ و راست قانون را با یکدیگر جابجا کنیم، مقدار این معیار تغییری نمی‌کند. از آنجائی که این معیار نمی‌تواند به تنهایی برای ارزیابی مورد استفاده قرار گیرد، و حتماً باید در کنار معیارهای دیگر باشد، باید مقادیر آن بین بازه صفر و یک نرمال شود. ترکیب این معیار به همراه Support و Confidence جزو بهترین روش‌های کاوش قوانین انجمنی است. مشکل این معیار حساس بودن به تعداد نمونه‌های مجموعه داده، به ویژه برای مجموعه تراکنش‌های کوچک می‌باشد. از این رو معیارهای دیگری برای جبران این نقص معرفی شده اند.

معیار **Conviction** برخی ضعف‌های معیارهای Confidence و Lift را جبران می‌نماید. محدوده قابل تعریف برای این معیار در حوزه 0.5 تا بی نهایت قرار می‌گیرد که هر چه این مقدار بیشتر باشد، نشان دهنده این است که آن قانون جذاب‌تر می‌باشد. بر خلاف Lift این معیار متقارن نمی‌باشد و مقدار این معیار برای دلالت‌های منطقی یعنی در جایی که Confidence قانون یک می‌باشد برابر با بی نهایت است و چنانچه A و B مستقل از هم باشند، مقدار این معیار برابر با عدد یک خواهد بود.

$$Conf\ (A \rightarrow B) = \frac{SUP(A \cup B)}{SUP(A)}$$

$$Lift\ (A \rightarrow B) = \frac{Conf(A \rightarrow B)}{SUP(B)}$$

$$Conv\ (A \rightarrow B) = \frac{1 - SUP(B)}{1 - Conf(A \rightarrow B)}$$

معیار **Leverage** که در برخی متون با نام Novelty (جدید بودن) نیز شناخته می‌شود، دارای مقداری بین 0.25- و 0.25+ می‌باشد. ایده مستتر در این معیار آن است که اختلاف بین میزان هم اتفاقی سمت چپ و راست قانون با آن مقداری که مورد انتظار است به چه اندازه می‌باشد.

معیار **Jaccard** که دارای مقداری عددی بین صفر و یک است، علاوه بر اینکه نشان دهنده وجود نداشتن استقلال آماری میان A و B می‌باشد، درجه همپوشانی میان نمونه‌های پوشش داده شده توسط هر کدام از آنها را نیز اندازه گیری می‌کند. به بیان دیگر این معیار فاصله بین سمت چپ و راست قانون را بوسیله تقسیم تعداد نمونه هایی که توسط هر دو قسمت پوشش داده شده اند بر نمونه هایی که توسط یکی از آنها پوشش داده شده است، محاسبه می‌کند. مقادیر بالای این معیار نشان دهنده این است که A و B تمایل دارند، نمونه‌های مشابهی را پوشش دهند. لازم است به این نکته اشاره شود از این معیار برای فهمیدن میزان همبستگی میان متغیرها استفاده می‌شود که از آن می‌توان برای یافتن قوانینی که دارای همبستگی بالا ولی Support کم هستند، استفاده نمود. برای نمونه در مجموعه داده سبد خرید، قوانین نادری که Support کمی دارند ولی همبستگی بالایی دارند، توسط این معیار می‌توانند کشف شوند.

$$Leve\ (A \rightarrow B) = SUP(A \cup B) - SUP(A) \times SUP(B)$$

$$Jaac\ (A \rightarrow B) = \frac{SUP(A \cup B)}{SUP(A) + SUP(B) - SUP(A \cup B)}$$

معیار **Coefficient (ϕ)** نیز به منظور اندازه گیری رابطه میان A و B مورد استفاده قرار می‌گیرد که محدوده این معیار بین 1- و 1+ می‌باشد.
از دیگر معیارهای ارزیابی کیفیت قوانین انجمنی، طول قوانین بدست آمده می‌باشد. به بیان دیگر با ثابت در نظر گرفتن معیارهای دیگر نظیر Support، Confidence و Lift قانونی برتر است که طول آن کوتاه‌تر باشد، بدلیل فهم آسانتر آن.

$$\Phi\ (A \rightarrow B) = \frac{Leve\ (A \rightarrow B)}{\sqrt{SUP(A) \times SUP(B) \times (1 - SUP(A)) \times (1 - SUP(B))}}$$

در نهایت با استفاده از **ماتریس وابستگی** (Dependency Matrix)، می‌توان اقدام به تعریف معیارهای متنوع ارزیابی روش‌های تولید قوانین انجمنی پرداخت. در عمل معیارهای متعددی برای ارزیابی مجموعه قوانین بدست آمده وجود دارد و لازم است با توجه به تجارب گذشته در مورد میزان مطلوب بودن آنها تصمیم گیری شود. بدین ترتیب که ابتدا معیارهای برتر در مسئله مورد کاوش پس از مشورت با خبرگان حوزه شناسائی شوند، پس از آن قوانین انجمنی بدست آمده از حوزه کاوش، مورد ارزیابی قرار گیرند.

نظرات خوانندگان

نویسنده: محمد باقر سیف اللهی
تاریخ: ۱۳۹۳/۰۹/۱۲ ۲۲:۳۳

خسته نباشید میگم. مطالب مفیدی است ولی نکته ای رو لازم می‌دونم بیان کنم و آن، اینکه مفاهیم بیان شده در حد مطالعه خوب است ولی بدون نمونه و مثال، مفاهیم بسیار سختی دارند. برای مثال روش تهیه ROC از روی جدول اطلاعاتی مرتبط با TP - TN - FN وFP و TP Rate و FP rate مشکل است خصوصا با داده‌های زیاد. و یا روش‌های Rule Pruning و Rule Growing در رده بندی و امثالهم. (که البته مفاهیم بسیار سنگین‌تری نیز وجود دارند)
به همین دلیل پیشنهاد می‌کنم در صورت امکان، در انتهای آموزش خود، تا جاییکه امکان دارد با مثال و شکل این موارد بیان شوند.

همچنین ابزارهای دیگری به غیر از SQL Server نیز مرور شوند (و یا صرفا معرفی شوند) تا با رویکرد عملیاتی ساختن و استفاده کاربردی از داده کاوی، بتوان از آنها بهره برد
(موردی بود که بنده مجبور بودم در یک پروسه، از 3 ابزار برای کارهای مختلف استفاده کنم) اسلاید هایی هم در این زمینه‌ها وجود دارند که برای شروع مناسب هستند + و +
موفق باشید

نویسنده: محمد رجبی
تاریخ: ۱۳۹۳/۰۹/۱۴ ۱۰:۴۲

با سلام و احترام، ضمن تشکر از Feedback ای که ارسال نمودید، حقیقتاً در ابتدای امر چنین قصدی داشتم ولی با مشورت دوستانم بر آن شدم، که مشخصاً به بیان مباحث تئوری موضوع بپردازم. از آنجا که به نظر می‌رسد بر خلاف رویه ماکروسافت که معمولاً مفاهیم را در مجموعه‌های آموزشی از مباحث پایه و مقدماتی شروع می‌کند و تا سطح پیشرفته؛ با جزئیات کامل به بیان موضوع می‌پردازد. متاسفانه در بحث داده کاوی چنین رویه ای را در پیش نگرفته و فرض را بر آن گذاشته است که خواننده با مفاهیم کلی علم داده کاوی آشناست و با این پیش فرض به بررسی الگوریتم‌ها و نحوه استفاده از آنها می‌پردازد. از این رو تصمیم گرفتم بیشتر خلاصه مباحث تئوری را بیان کنم و از آنجایی که به منظور انجام عملیات داده کاوی در گام نخستین شخص داده کاو می‌بایست از داده‌های مورد کاوش، شناخت و آگاهی کافی داشته باشد و همانطور که می‌دانیم، جهت اهداف آموزشی بانک اطلاعاتی Adventure Works (که حاوی اطلاعات حوزه‌های متفاوت در یک کمپانی می‌باشد) و بانک Adventure Works DW (که در واقع انبار داده حوزه فروش بانک Adventure Works است)، موجود می‌باشد. کلیه مثال‌های موجود در Books Online که برای هر الگوریتم و زمینه کاری متناظر با آن ارائه شده است روی این بانک‌های اطلاعاتی انجام می‌گیرد.
لینک زیر دانلود مجموعه آموزش SQL Server 2012 Tutorials - Analysis Services Data Mining می‌باشد. که شامل موارد زیر است:

Basic Data Mining Tutorial

```
Lesson 1: Preparing the Analysis Services Database
```

```
Lesson 2: Building a Targeted Mailing Structure
```

```
Lesson 3: Adding and Processing Models
```

```
Lesson 4: Exploring the Targeted Mailing Models
```

```
Lesson 5: Testing Models
```

```
Lesson 6: Creating and Working with Predictions
```

Intermediate Data Mining Tutorial

```
Lesson 1: Creating the Intermediate Data Mining Solution
```

```
Lesson 2: Building a Forecasting Scenario
```

```
Lesson 3: Building a Market Basket Scenario
```

```
Lesson 4: Building a Sequence Clustering Scenario
```

```
Lesson 5: Building Neural Network and Logistic Regression Models
```

Creating and Querying Data Mining Models with DMX: Tutorials

```
Lesson 1: Bike Buyer
```

```
Lesson 2: Market Basket
```

```
Lesson 3: Time Series Prediction
```

بدین ترتیب برای مخاطبان این دوره که ممکن است آشنائی با مفاهیم تئوری علم داده کاوی نداشته باشند، سعی شده است مطالب به گونه ای بیان شود که با مطالعه این مجموعه، سر نخ هایی از موضوع بدست آورند و طبیعتاً در صورت علاقه مندی به موضوع به مطالعه عمیق هر الگوریتم بپردازند.

از آنجا که با سونامی «تحصیلات تکمیلی» در کشور مواجه هستیم و بسیاری از پایان نامه‌ها پیرامون موضوع Data Mining می‌باشد و همچنین مشابه بسیاری از موضوعات دیگر؛ بدون در نظر گرفتن زیر ساخت‌ها و فلسفه پیدایش موضوع و دستاوردهای آن و ... پروژه‌های داده کاوی نیز به صورت وارداتی به کشور و به طبع سازمان‌ها تحمیل می‌شود و ... امیدوارم توانسته باشم، هم زبانان نا آشنا را تا حدی که در توان داشتم با موضوع آشنا کرده باشم. برای مطالعه منابع غیر از SQL Server کتاب‌های « داده کاوی کاربردی - RapidMiner » انتشارات نیاز دانش و همچنین کتاب « داده کاوی با کلمنتاین » انتشارات جهاد دانشگاهی واحد صنعتی امیر کبیر نیز به بیان موضوع می‌پردازد. موفق و سلامت باشید.

عنوان: **مروری اجمالی بر الگوریتم های داده کاوی و پارامترهای مرتبط با آنها موجود در SSAS**
نویسنده: محمد رجبی
تاریخ: ۱۳۹۳/۰۹/۱۲ ۱۳:۱۹
آدرس: www.dotnettips.info
گروه‌ها: Analysis Services, data mining, Microsoft SQL Server

این بخش مروری اجمالی بر الگوریتم‌های موجود در Analysis Services و پارامترهای قابل تنظیم و مقدار پیش فرض هر پارامتر می‌باشد، به منظور بررسی بیشتر هر یک به لینک‌های زیر مراجعه کنید:

Data Mining Algorithms (Analysis Services - Data Mining (Algorithm Parameters (SQL Server Data Mining Add-ins)

Microsoft Association Rules - 1

به منظور ایجاد قوانینی که توصیف کننده این موضوع باشد که چه مواردی احتمالاً با یکدیگر در تراکنش‌ها ظاهر می‌شوند، استفاده می‌شود.

Parameter	Default	Range
MAXIMUM_ITEMSET_COUNT	200000	(...,1]
MAXIMUM_ITEMSET_SIZE	3	[0,500]
MAXIMUM SUPPORT	1.0	(...,0.0)
MINIMUM IMPORTANCE	999999999 -	(...,...)
MINIMUM_ITEMSET_SIZE	1	[1,500]
MINIMUM PROBABILITY	0.4	[0.0,1.0]
MINIMUM SUPPORT	0.0	(...,0.0]

Microsoft Clustering - 2

به منظور شناسائی روابطی که در یک مجموعه داده ممکن است از طریق مشاهده منطقی به نظر نرسد، استفاده می‌شود. در واقع این الگوریتم با استفاده از تکنیک‌های تکرار شونده رکوردها را در خوشه هایی که حاوی ویژگی‌های مشابه هستند گروه بندی می‌کند.

Parameter	Default	Range
CLUSTER COUNT	10	(...,0]
CLUSTER SEED	0	(...,0]
CLUSTERING METHOD	1	1,2,3,4
MAXIMUM_INPUT_ATTRIBUTES	255	[0,65535]
MAXIMUM STATES	100	[2,65535],0
MINIMUM SUPPORT	1	(...,0)
MODELLING_CARDINALITY	10	[1,50]
SAMPLE SIZE	50000	(...,100], 0
STOPPING TOLERANCE	10	(...,0)

Microsoft Decision Trees - 3

مبتنی بر روابط بین ستونهای یک مجموعه داده ای باعث پیش بینی روابط مدل‌ها می‌شود، که به صورت یک سری درختوار ویژگی‌ها در آن شکسته می‌شوند.

به منظور انجام پیش بینی از هر دو ویژگی پیوسته و گسسته پشتیبانی می‌شود.

Parameter	Default	Range
COMPLEXITY_PENALTY		(0.0,1.0)
FORCE REGRESSOR		
MAXIMUM_INPUT_ATTRIBUTES	255	[0,65535]
MAXIMUM_OUTPUT_ATTRIBUTES	255	[0,65535]
MINIMUM SUPPORT	10.0	(...,0.0)
SCORE METHOD	4	1,3,4
SPLIT METHOD	3	[1,3]

Microsoft Linear Regression - 4

چنانچه یک وابستگی خطی میان متغیر هدف و متغیرهای مورد بررسی وجود داشته باشد، کارآمدترین رابطه میان متغیر هدف و ورودی‌ها را پیدا می‌کند.

به منظور انجام پیش بینی از ویژگی پیوسته پشتیبانی می‌کند.

Parameter	Default	Range
FORCE REGRESSOR		
MAXIMUM_INPUT_ATTRIBUTES	255	[0,65535]
MAXIMUM_OUTPUT_ATTRIBUTES	255	[0,65535]

Microsoft Logistic Regression - 5

به منظور تجزیه و تحلیل عواملی که در یک تصمیم گیری مشارکت دارند که پی آمد آن به وقوع یا عدم وقوع یک رویداد می‌انجامد از این الگوریتم استفاده می‌شود.

جهت انجام پیش بینی از هر دو ویژگی پیوسته و گسسته پشتیبانی می‌کند.

Parameter	Default	Range
HOLDOUT_PERCENTAGE	30	(0,100)
HOLDOUT SEED	0	(...,...)
MAXIMUM_INPUT_ATTRIBUTES	255	[0,65535]
MAXIMUM_OUTPUT_ATTRIBUTES	255	[0,65535]
MAXIMUM STATES	100	[2,65535], 0
SAMPLE SIZE	10000	(...,0]

Microsoft Naïve Bayes - 6

احتمال ارتباط میان تمامی ستون‌های ورودی و ستون‌های قابل پیش بینی را پیدا می‌کند. همچنین این الگوریتم برای تولید سریع مدل کاوش به منظور کشف ارتباطات بسیار سودمند می‌باشد. تنها از ویژگی‌های گسسته یا گسسته شده پشتیبانی می‌کند و با تمامی ویژگی‌های ورودی به شکل مستقل رفتار می‌کند.

Parameter	Default	Range
MAXIMUM_INPUT_ATTRIBUTES	255	[0,65535]
MAXIMUM_OUTPUT_ATTRIBUTES	255	[0,65535]
MAXIMUM STATES	100	[2,65535], 0
MINIMUM_DEPENDENCY_PROBABILITY	0.5	(0,1)

Microsoft Neural Network - 7

به منظور تجزیه و تحلیل داده‌های ورودی پیچیده یا مسائل بیزنسی که برای آنها مقدار قابل توجهی داده آموزشی در دسترس می‌باشد اما به آسانی نمی‌توان با استفاده از الگوریتم‌های دیگر این قوانین را بدست آورد، استفاده می‌شود. با استفاده از این الگوریتم می‌توان چندین ویژگی را پیش بینی نمود. همچنین این الگوریتم می‌تواند به منظور طبقه بندی برای ویژگی‌های گسسته و ویژگی‌های پیوسته رگرسیون مورد استفاده قرار گیرد.

Parameter	Default	Range
HIDDEN_NODE_RATIO	4.0	(...,0]
HOLDOUT PERCENTAGE	30	(0,100)
HOLDOUT SEED	0	(...,...)
MAXIMUM_INPUT_ATTRIBUTES	255	[0,65535]
MAXIMUM_OUTPUT_ATTRIBUTES	255	[0,65535]
MAXIMUM STATES	100	[2,65535], 0
SAMPLE SIZE	10000	(...,0]

Microsoft Sequence Clustering - 8

به منظور شناسائی ترتیب رخدادهای مشابه در یک دنباله استفاده می‌شود. در واقع این الگوریتم ترکیبی از تجزیه تحلیل توالی و خوشه را فراهم می‌کند.

Parameter	Default	Range
CLUSTER COUNT	10	(...,0]
MAXIMUM_SEQUENCE_STATES	64	[2,65535], 0
MAXIMUM STATES	100	[2,65535], 0
MINIMUM SUPPORT	10	(...,0]

Microsoft Time Series - 9

به منظور تجزیه و تحلیل داده‌های زمانی (داده‌های مرتبط با زمان) در یک درخت تصمیم گیری خطی استفاده می‌شود. الگوهای کشف شده می‌توانند به منظور پیش بینی مقادیر آینده در سری‌های زمانی استفاده شوند.

Parameter	Default	Range
AUTO_DETECT_PERIODICITY	0.6	[0.0,1.0]
COMPLEXITY_PENALTY	0.1	(1.0,...)
FORECAST METHOD	MIXED	ARIMA,ARTXP,MIXED

Parameter	Default	Range
HISTORIC_MODEL_COUNT	1	[0,100]
HISTORIC_MODEL_GAP	10	(...,1]
INSTABILITY_SENSITIVITY	1.0	[0.0,1.0]
MAXIMUM_SERIES_VALUE	1E308 +	[...,column maximum]
MINIMUM_SERIES_VALUE	1E308 -	[column minimum,...]
MINIMUM SUPPORT	10	(...,1]
MISSING_VALUE_SUBSTITUTION	None	None,Previous,Mean
PERIODICITY_HINT	{1}	{...list of integers...}
PREDICTION SMOOTHING	0.5	[0.0,1.0]

عنوان: **اضافه نمودن Add-Ins برای Excel جهت استفاده در داده کاوی**
نویسنده: محمد رجبی
تاریخ: ۱۳۹۳/۰۹/۱۷ ۸:۳۷
آدرس: www.dotnettips.info
گروه‌ها: Analysis Services, data mining, Microsoft SQL Server

نرم افزار Excel حاوی مجموعه ای از ابزارهای تحلیلی با ماهیت پیش بینی می‌باشد. در این صورت قادر هستید با افزودن این مجموعه Add-Insها یکسری کارهای معمول در داده کاوی را انجام دهید. برای بررسی بیشتر به لینک‌های زیر مراجعه کنید.
Data Mining Client for Excel (SQL Server Data Mining Add-ins Microsoft® SQL Server® 2012 Data Mining Add-ins) for Microsoft® Office® 2010 Data Mining Part 19: Excel and Data Mining, Samples, Queries How to Use the SQL Server Data Mining Add-ins with PowerPivot for Excel