داده‌کاوی در R با استفاده از بسته Rattle

روح اله حسینی[۱] ، مجید سرمد[۲] ، مهدی جباری نوقابی[۳]

چکیده

در این مقاله ضمن معرفی مختصری از مفاهیم، روش‌ها و الگوریتم‌های داده‌کاوی، داده‌کاوی در نرم‌افزار آماری R با استفاده از بسته Rattle ارائه می شود. بسته Rattle فضای گرافیکی مناسب را برای انجام برخی از روش‌ها و الگوریتم‌ها، بدون نیاز به برنامه‌نویسی فراهم می‌کند. برخی از بخش‌های این بسته ضمن مثال، شرح داده خواهد شد.

واژه‌های کلیدی: داده‌کاوی، خوشه‌بندی، درخت‌تصمیم، قواعد پیوند، ماشین بردار پشتیبان، R، ،Rattle.

۱ مقدمه

اطلاعات در جهان امروز به عنوان یکی از عوامل تولیدی مهم محسوب می شود. داده‌ها نمایشی از واقعیت‌ها، معلومات، مفاهیم، رویدادها یا پدیده‌ها برای برقراری ارتباط، تفسیر یا پردازش، توسط انسان یا ماشین هستند. از سال ۱۹۵۰ با به کارگیری رایانه، در تحلیل و ذخیره‌سازی داده‌ها، حجم اطلاعات ذخیره شده پس از حدود ۲۰ سال دو برابر شده است. همزمان با پیشرفت فناوری اطلاعات(IT)، هر دو سال یکبار حجم داده‌ها در پایگاه داده‌ها، دو برابر می شود. امروزه نیز با وجود شبکه جهانی وب، سیستم‌های یکپارچه بانکی، سیستم‌های یکپارچه اطلاعاتی و ... هر لحظه حجم داده‌ها در پایگاه داده‌ها افزایش یافته و باعث به وجود آمدن انبارهای عظیمی از داده‌ها شده است. با توجه به شدت رقابت‌ها در عرصه‌های علمی، اجتماعی، اقتصادی، سیاسی و نظامی در بین کشورها، موسسات و شرکت‌ها، ضرورت کشف و استخراج سریع و دقیق دانش موجود در این داده‌ها بیش از پیش نمایان شده است. بنابراین نیاز به طراحی سیستم‌هایی که قادر به اکتشاف سریع اطلاعات مورد علاقه کاربران با تاکید بر حداقل مداخله انسانی باشند از یک طرف و روی آوردن به روش‌های تحلیل متناسب با حجم انبوه داده‌ها از سوی دیگر احساس می‌شود. رشد فزاینده تعداد روش های نوین ارائه شده برای استخراج اطلاعات از داده‌های خام موجب بوجود آمدن زمینه جدیدی از کاوش داده‌ها موسوم به داده‌کاوی[۴] شده است.

داده‌کاوی که در آغاز دهه ۹۰، با انجام تحقیقات در رشته‌های آمار، یادگیری ماشین، هوش مصنوعی، علوم رایانه و ...، پا به عرصه ظهور گذاشته، با نگرشی نو به مساله استخراج دانش از مجموعه عظیم داده‌ها می‌پردازد. اصطلاح داده‌کاوی نخسین بار توسط «فیاض»[۵] در اولین کنفرانس بین‌المللی « داده‌کاوی و کشف دانش »[۶] در سال ۱۹۹۵ مطرح گردید و همزمان با آن داده‌کاوی به‌طور جدی وارد مباحث آمار شد. با وجود این‌که داده‌کاوی یک رشته جدید علمی است، امروزه کاربردهای

[۱] کارشناس ارشد، دانشگاه فردوسی مشهد

[۲] عضو هیأت علمی گروه آمار دانشگاه فردوسی مشهد

[۳] عضو هیأت علمی گروه آمار دانشگاه فردوسی مشهد

[4] Data mining

[5] Fayyad

[6] Kowledge Discovery and Data mining

متنوع و گسترده‌ای در رشته‌های بازرگانی، پزشکی، مهندسی، علوم رایانه، صنعت، کنترل کیفیت، ارتباطات، کشاورزی و ... پیدا کرده است.

۲ داده‌کاوی

نگاهی به ترجمه لغوی داده‌کاوی، به ما در درک بهتر این واژه کمک می‌کند. واژه لاتین Mine به معنای استخراج از منابع نهفته و با ارزش زمین اطلاق می‌شود. ادغام این واژه با Data بر جستجویی عمیق از داده‌های قابل دسترس با حجم زیاد برای یافتن اطلاعات مفید که قبلاً نهفته بودند، تأکید دارند.

داده‌کاوی دارای تعاریف مختلفی است، این تعاریف به مقدار زیادی به پیش زمینه‌ها و نقطه نظرهای افراد بستگی دارد. هر نویسنده و محقق با توجه به دیدگاه و نوع نگرش خود تعریفی برای داده‌کاوی ارائه کرده است که برخی از آنها به صورت زیر است:

۱. داده‌کاوی فرآیند شناخت الگوهای معتبر، جدید، ذاتاً مفید و قابل فهم از داده‌ها است.[۲]

۲. فرآیند کشف الگوهای مفید از داده‌ها را داده‌کاوی گویند. [۳]

۳. داده‌کاوی، مجموعه‌ای از روش‌ها در فرآیند کشف دانش است که برای تشخیص الگوها و رابطه‌های نامعلوم در داده‌ها مورد استفاده قرار می‌گیرد.[۴]

۱.۲ آمار و داده‌کاوی

غالباً در آمار نیاز به فرضیه آماری داریم ولی در داده‌کاوی بدون داشتن فرضیه اولیه به تحلیل و کاوش در داده‌ها می‌پردازیم. روابط در داده‌کاوی معمولاً به صورت الگوها و مدل‌هایی از قبیل معادلات رگرسیونی، سری‌های زمانی، خوشه‌ها، رده‌بندی‌ها و گراف‌ها ارائه می‌شوند. در داده‌کاوی نیز همانند آمار غالباً داده‌هایی که تحلیل می‌شوند، نمونه‌ای از جامعه هستند که با توجه به حجم بزرگ جامعه با نمونه‌ای بزرگ مواجه هستیم. یک تفاوت در نوع داده‌ها است، آماردان‌ها با داده‌های دست اول که برای تحقیق درستی فرضیه‌های از پیش تعیین شده جمع‌آوری و تولید شده‌اند، مواجه هستند، اما داده‌کاوها با تحلیل داده‌های دست دوم[۷] و یا داده‌های مشاهده‌ای که اغلب از منابع مختلفی گردآوری شده‌اند، سروکار دارند و اغلب با اهداف ابتدایی که داده‌ها به‌خاطر آن جمع‌آوری شده‌اند، رابطه‌ای ندارند. در داده‌کاوی با انواع مختلفی از داده‌ها از جمله داده‌های عددی، متنی، رشته‌ای، دودویی، تاریخ و عکس سروکار داریم.

۲.۲ بیان دیدگاه‌ها

در متون مربوط به داده‌کاوی دو تعبیر مختلف از داده‌کاوی وجود دارد: برخی از جمله فیاض (۱۹۹۵) به داده‌کاوی به عنوان یک مرحله ضروری از فرآیند بزرگ‌تر «کشف دانش و شناخت از پایگاه داده‌ها»[۸] (KDD) می‌نگرند. برخی دیگر مانند چتفیلد(۱۹۹۵) داده‌کاوی را مترادف عبارت کشف دانش و شناخت از پایگاه‌داده‌ها می‌دانند. فرآیند KDD شامل مراحل زیر است:

۱. شناسایی و تعریف مساله

۲. دست‌یابی و پیش‌پردازش داده‌ها

- پاک‌سازی داده‌ها
- یک‌پارچه‌سازی داده‌ها
- تبدیل داده‌ها
- تلخیص داده‌ها

۳. داده‌کاوی

۴. ارزیابی الگوها

۵. ارائه دانش

۳.۲ روش‌های داده‌کاوی

روش‌های داده‌کاوی به دو دسته توصیفی و پیش‌بینانه تقسیم‌بندی می‌شوند. روش‌های توصیفی خواص عمومی داده‌ها را آشکار می‌کنند. هدف از توصیف، یافتن الگوهایی از داده‌هاست که برای انسان قابل تفسیر باشد. وظایف پیش‌بینانه به منظور پیش‌بینی رفتارهای آینده آن‌ها استفاده می‌شوند. منظور از پیش‌بینی به کارگیری چند متغیر برای پیش‌بینی مقادیر آینده یا ناشناخته سایر متغیرهای مورد علاقه است.

[7] Secondary Data Analysis

[8] Knowedge Discovery and Data Mining

۴.۲ الگوریتم‌ها و تکنیک‌های داده‌کاوی

الگوریتم‌ها و روش‌های داده‌کاوی راه‌های اجرا کردن عملیات‌های داده‌کاوی هستند. در بین الگوریتم‌های یاد شده بهترین به معنای مطلق وجود ندارد و لذا ابزار یا ابزارهایی انتخاب شده و مدل مناسب باید با توجه به داده ها و کارایی مورد نظر طراحی و اجرا شود. برخی از مهم‌ترین الگوریتم‌های داده‌کاوی عبارت از:

- خوشه‌بندی[9]
- قواعد پیوند[10]
- درخت‌های تصمیم
- روش‌های رگرسیونی
- شبکه عصبی مصنوعی
- ماشین بردار پشتیبان[11]
- الگوریتم ژنتیک

هستند.

باتوجه به نظرسنجی انجام شده از داده‌کاوان در سایت www.KDnuggets.com الگوریتم‌های درخت‌های تصمیم، رگرسیون و خوشه‌بندی در سال های ۲۰۱۰ و ۲۰۱۱ بیش از سایر الگوریتم‌ها مورد استفاده قرار گرفته‌اند.

۵.۲ نرم‌افزارهای داده‌کاوی

امروزه نرم‌افزارهای مختلفی با قابلیت انجام تحلیل های داده کاوی وجود دارد. بیشتر سیستم‌های نرم‌افزاری فقط جهت انجام روش‌های خاصی مناسب هستند و قابلیت انجام روش‌های دیگر را ندارند. برخی از نرم‌افزارهای آماری که قابلیت داده‌کاوی دارند عبارت از:

- R (Rattle package)
- SPSS Clementine
- SAS Enterprise Miner
- Rapid Miner

هستند. در سال های ۲۰۱۰ و ۲۰۱۱ نرم‌افزارهای Rapid Miner و R به‌دلیل منبع باز بودن، رایگان بودن، فضای گرافیکی مناسب و هم‌چنین قابلیت‌های دیگر بیش از سایر نرم‌افزارها مورد استفاده داده‌کاوها قرار گرفته‌اند.

۳ داده‌کاوی در R با استفاده از $Rattle$

همان‌گونه که اشاره شد، در بین نرم‌افزارهایی که قابلیت داده‌کاوی دارند نرم‌افزار آماری R مورد توجه داده‌کاوها قرار گرفته است. برخی از دلایل این رویکرد عبارت از:

۱. رایگان بودن و کد باز بودن (Open Source)

۲. قابلیت نصب روی اکثر سیستم‌های عامل از جمله نسخه‌های مختلف لینوکس، ویندوز، مکینتاش و یونیکس

۳. سرعت در دست‌یابی به تکنیک‌های جدید در قالب کتابخانه‌ها و توابع آماده

۴. قابلیت اضافه کردن و نوشتن برنامه‌های جدید به‌صورت بسته‌های جدید در R

۵. دارا بودن قابلیت‌های قابل ملاحظه گرافیکی

است. در نرم افزار R چندین بسته برای داده‌کاوی وجود دارد که می‌توان با توجه به الگوریتم مورد استفاده برای داده‌کاوی از آنها استفاده کرد. اما یکی از بهترین بسته‌ها که مجموعه‌ای از الگوریتم‌ها و توابع را در کنار هم جمع کرده تا بتوان از آنها برای داده‌کاوی استفاده کرد بسته Rattle است. این بسته علاوه براینکه شامل مجموعه‌ای از الگوریتم‌ها و روش‌های داده‌کاوی است کاربر را از برنامه‌نویسی و دستورنویسی به زبان R بی نیاز می کند و این امکان را فراهم می کند که و کاربر با چند کلیک ساده و انتخاب روش مورد نیاز برای داده‌کاوی به خواسته خود برسد.

[9] Clustering

[10] Association Rules

[11] Support Vector Machine

۱.۳ معرفی Rattle

همان‌گونه که بیان شد برای انجام داده‌کاوی در نرم‌افزار R می‌توان از ابزاری به‌نام Rattle که مخفف عبارت R Analytical Tool To Learn Easily است استفاده کرد. این ابزار گرافیکی داده‌کاوی نوشته شده در R، گذرگاهی به R محسوب می‌شود. این برنامه برای انجام محاسبات داده‌کاوی ساده تا تحلیل‌های پیچیده بر روی داده‌ها به کمک یک زبان بسیار قدرتمند آماری، طراحی شده است. Rattle به همراه خود انبوهی از بسته‌های R مورد نیاز برای یک داده‌کاو را فراهم می‌آورد. در این محیط هرگاه به بسته‌ای نیاز باشد پیغام نصب این بسته ظاهر می‌شود که در صورت اتصال به اینترنت با انتخاب گزینه Yes در پیغام ظاهر شده این بسته نصب می‌شود. برای استفاده از Rattle نیاز به شناخت و آشنایی با R وجود ندارد.

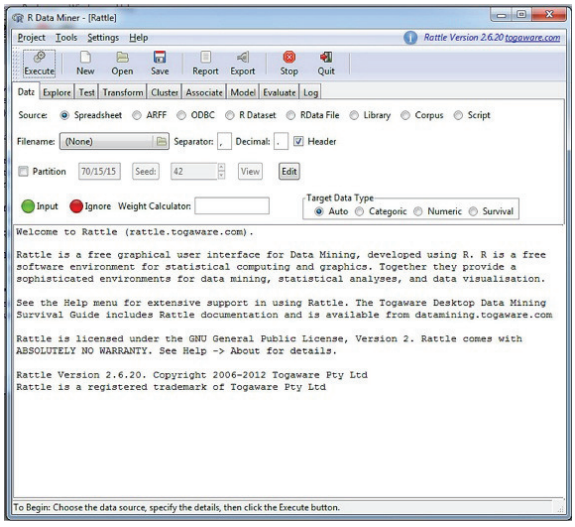
برای نصب Rattle کتابخانه‌های Gnom و Glade مورد نیاز است . با فرض اینکه R و کتابخانه‌های فوق در R نصب شده باشند برای استفاده از محیط Rattle باید بسته‌های RGtk2 و Rattle را در R نصب و فراخوانی کرد. با وارد کردن دستورات زیر در R می‌توان این دو بسته را نصب و بسته Rattle را فراخوانی کرد.

```
> install.packages(”RGtk2”)
> install.packages(”rattle”)
> library(rattle)
```

آخرین نسخه توسعه یافته Rattle را می‌توان با استفاده از دستور زیر در R نصب کرد:

```
> install.packages( ”rattle”, respos= ”http ://
cran.um.ac.ir”)
```

برای باز کردن محیط Rattle با وارد کردن دستور rattle() در R پنجره جدیدی مشابه شکل ۱ باز می‌شود. همان‌طور که ملاحظه می‌شود پنجره بازشده به فرم منویی بوده و دارای نوار عنوان، نوار ابزار و بخش خروجی نتایج است.



شکل ۱: محیط گرافیکی Rattle

به طور کلی روند اجرای یک پروژه داده‌کاوی در Rattle را می‌توان به شرح زیر خلاصه کرد:

۱. بارگذاری یک مجموعه داده و انتخاب متغیرها

۲. کاوش کردن در داده‌ها برای شناخت توزیع‌های آن‌ها

۳. آزمون کردن توزیع‌ها و پراکندگی‌ها

۴. تبدیل داده‌ها برای انطباق با مدل انتخاب شده برای داده‌کاوی

۵. ساخت مدل یا مدل‌ها

۶. ارزیابی مدل‌ها

۷. بازنگری مراحل و روش‌های داده‌کاوی

۲.۳ داده‌ها در Rattle

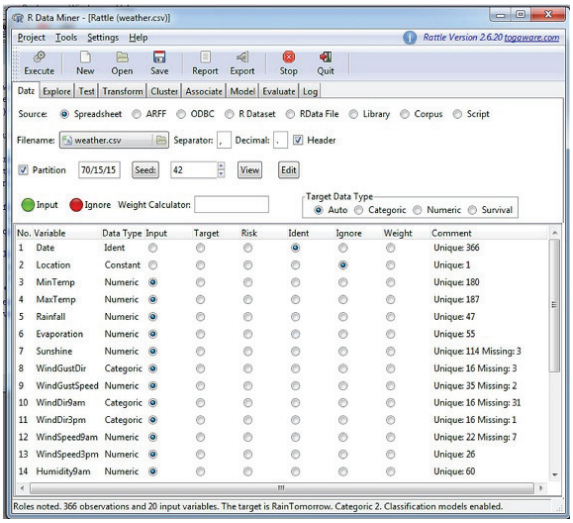
Rattle قابلیت فراخوانی داده‌ها را از منابع گوناگونی دارد که قالب پیش‌فرض آن CSV (Comma Separated Values) است. از جمله قالب‌های دیگری که Rattle قابلیت فراخوانی آن‌ها را دارد می‌توان به موارد زیر اشاره کرد:

TXT (tab separated data), ARFF (a common data mining dataset format which adds type information to a CSV file)

با نصب بسته ODBC در R و استفاده از زیربخش ODBC بخش Data در Rattle اجازه دستیابی به بسیاری از منابع داده از جمله:

MySQL, SQLite, Postgress, MS/Excel, MS/Access, SQL Server, Oracle, IBM DB2

را داریم. Rattle همچنین اجازه دستیابی و استفاده از مجموعه داده‌های بسته‌های نصب شده در R و مجموعه داده‌های استفاده شده و فراخوانی شده در محیط R را نیز به کاربر می‌دهد. بسته Rattle دارای یکی از این مجموعه داده‌ها بنام weather.csv شامل ۳۶۶ مشاهده با ۲۴ متغیر از داده‌های هواشناسی است. یک راه ساده برای فراخوانی این مجموعه داده در Rattle کلیک بر روی Execute و انتخاب Yes در پیغام ظاهر شده است. با انجام این کار نام متغیرهای این مجموعه داده همراه با نقش هریک از متغیرها، که Rattle بطور پیش‌فرض با توجه به نوع متغیر در نظر گرفته است، در پنجره خروجی Rattle نمایش داده می‌شود (شکل ۲).

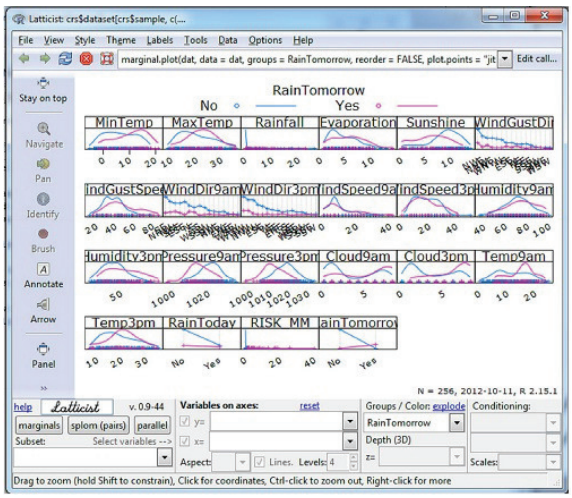


شکل ۲: فراخوانی مجموعه داده هواشناسی در Rattle

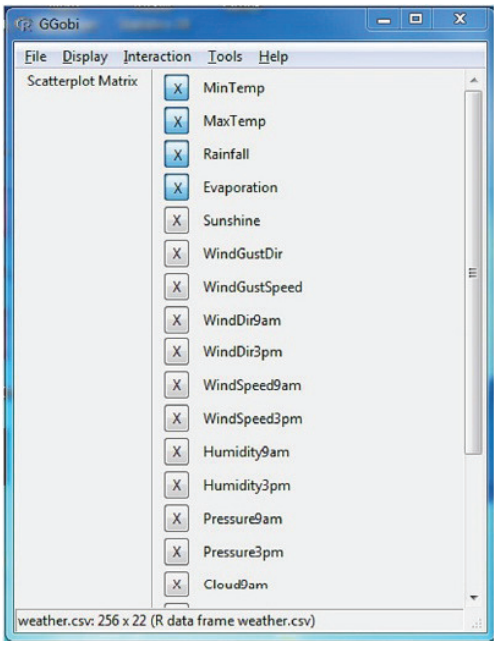
۳.۳ کاوش در داده‌ها (*Explore*)

گاهی با بررسی داده‌ها و تحلیل کاوشگرانه آن‌ها می‌توان به اطلاعات و دانش جدید در مورد داده‌ها دست یافت. بخش Explore در منوی Rattle بسیاری از ابزارهای عددی و گرافیکی را برای بدست آوردن آماره‌های توصیفی، انواع نمودارها و توزیع‌ها ، همبستگی بین متغیرها و مشاهدات، مولفه‌های اصلی و ... را فراهم می‌کند. با انتخاب زیربخش interactive می‌توان با استفاده از گزینه‌های Latticist ، GGobi و Plot Builder نمودارها و توزیع‌های آماری را در هر کدام از فضاهای گرافیکی موجود به نمایش درآورد. برای استفاده از فضای گرافیکی Latticist باید بسته‌های latticist و playwith را در R نصب کرد.

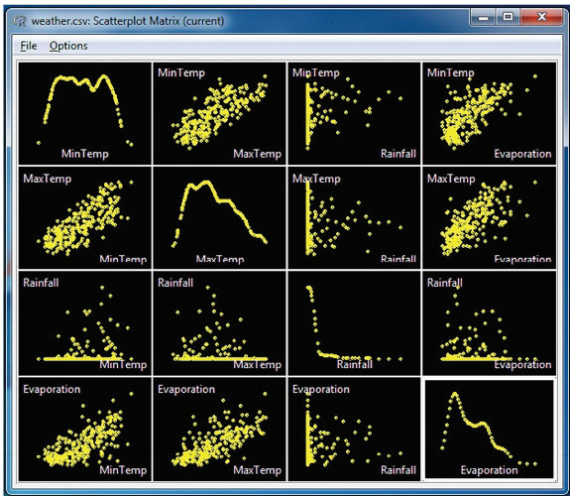
همچنین برای استفاده از فضای گرافیکی GGobi باید بسته rggobi را در R نصب کرد. بعلاوه باید نرم‌افزار GGobi را در سیستم نصب کرد این نرم‌افزار را می‌توان از http://www.ggobi.org/ دانلود کرد. این فضاهای گرافیکی برای کاوش کردن در داده‌ها با ابعاد بالا همچنین نمایش انواع نمودارها و توزیع‌ها بصورت محاوره‌ای مناسب هستند. نمودار پراکندگی چند متغیر از مجموعه داده هواشناسی را در این دو محیط گرافیکی رسم کرده‌ایم که به صورت شکل‌های ۳، ۴ و ۵ هستند.



شکل ۳: محیط گرافیکی latticist



شکل ۴: محیط گرافیکیGGobi



شکل ۵: محیط گرافیکیGGobi

۴.۳ آزمون‌های آماری

بخش Test در Rattle امکان دستیابی و اجرای بسیاری از آزمون‌های آماری پارامتری و ناپارامتری از توزیع‌ها را برای داده‌ها فراهم می‌کند. این ویژگی به تازگی به Rattle افزوده شده است و در حال تکمیل شدن است. با استفاده از این بخش آزمون‌های Kolmogorov-Smirnov، Wilcoxon Rank-Sum، T-test، F-test، Wilcoxon Signed Rank و Pearson's correlation را می‌توان برای داده‌ها و توزیع داده‌ها انجام داد.

۵.۳ تبدیل داده‌ها

آماده‌سازی داده‌ها برای داده‌کاوی هنر فشردن داده‌های موجود و استخراج داده‌های با ارزش است. در حالی‌که داده‌کاوی هنر کشف الگوهای معنی‌دار در داده‌ها است. معناداری الگو بستگی به مساله دارد. آماده‌سازی نیز به عنوان جزئی از داده‌کاوی بستگی به نوع مساله و نیز روش‌ها و ابزارهایی دارد که می‌خواهیم بر روی داده‌ها به کار ببندیم. به عنوان مثال شبکه عصبی نیازمند ارائه داده‌هایی است که حداقل عددی یا ترتیبی باشند و به داده‌های گمشده بسیار حساس است. درخت‌های تصمیم‌گیری اغلب بر روی داده‌های طبقه‌ای کار می‌کنند.

هنگامی که داده‌های مورد نیاز انتخاب شدند و داده‌های مورد کاوش مشخص گردیدند، معمولاً به تبدیلات خاصی روی داده‌ها نیاز است. نوع تبدیل به عملیات و تکنیک داده‌کاوی مورد استفاده بستگی دارد. تبدیلات ساده‌ای همچون تبدیل نوع داده‌ای به نوع دیگر تا تبدیلات پیچیده‌تر همچون تعریف صفات جدید با انجام عملیات‌های ریاضی و منطقی روی صفات موجود. بسیاری از این تبدیلات در بخش Transform بسته Rattle گنجانده شده است. تبدیلاتی همچون تغییر مقیاس، تبدیل داده‌های کمی به کیفی، جایگذاری مقادیر گمشده (با صفر، میانگین مشاهدات، میانه، مد و یا یک مقدار ثابت)، حذف مقادیر (گمشده، پرت، موارد انتخاب شده توسط کاربر) و برخی دیگر را می‌توان انجام داد.

۶.۳ تحلیل خوشه‌ای

همان‌گونه که بیان شد روش‌های داده‌کاوی به دو دسته توصیفی و پیش‌بینانه تقسیم می‌شوند. تحلیل خوشه‌ای[12] یکی از روش‌های داده‌کاوی توصیفی است. در تحلیل خوشه‌ای تلاش می‌شود تا مشاهدات به k خوشه مختلف گروه‌بندی شوند، به طوریکه مشاهداتی که در یک خوشه قرار می‌گیرند به یکدیگر شبیه باشند و مشاهدات خوشه‌های مختلف با یکدیگر بیشترین تفاوت را داشته باشند. می‌توان گفت در تحلیل خوشه‌ای سه موضوع مورد توجه است؛ اولین موضوع تعداد خوشه‌ها است، چه تعداد خوشه می‌تواند دانش نهفته در داده‌ها را کشف کند. دومین نکته آن چیزی است که با نام شباهت[13] و یا فاصله[14] از آن یاد می‌شود. اینکه چه هنگام دو مشاهده به یکدیگر شبیه هستند و چه هنگام با یکدیگر تفاوت دارند و در نهایت پس از آن‌که تعداد خوشه‌ها و معیار شباهت داده‌ها مشخص شد باید با استفاده از روش‌های مختلف، که با نام روش‌های خوشه‌بندی[15] از آن‌ها یاد می‌شود، مشاهدات را در خوشه‌های معین قرار داد.

کیفیت در خوشه‌بندی با این معیار تعریف می‌شود که مشاهدات هر خوشه بیشترین شباهت را به یکدیگر داشته و از کمترین شباهت با خوشه‌های دیگر برخوردار باشند.

[12] Cluster analysis
[13] Similarity
[14] Distance
[15] Clustering
[16] Crisp or hard
[17] Fuzzy

به طور کلی روش‌های خوشه‌بندی به دو دسته کلی قطعی[۱۶] و فازی[۱۷] تقسیم‌بندی می‌شوند. روش‌های قطعی نیز خود به دو نوع افرازی[۱۸] و سلسله مراتبی[۱۹] تقسیم می‌شوند. روش‌های سلسله مراتبی نیز به دو نوع تجزیه‌ای یا تقسیمی[۲۰] و تجمیعی[۲۱] تقسیم می‌شود.

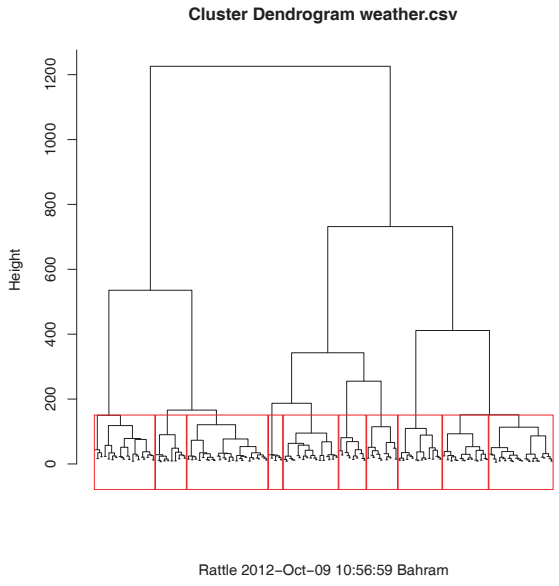
در خوشه‌بندی قطعی هر مشاهده تنها در یک خوشه می‌تواند قرار بگیرد ولی در خوشه‌بندی فازی هر مشاهده با یک درجه عضویتی، بین صفر و یک برای هر خوشه، می‌تواند در چندین خوشه قرار بگیرد. در روش‌های خوشه‌بندی افرازی تعداد خوشه‌ها از قبل مشخص است. هدف تعیین کردن این است که هر مشاهده در کدام خوشه جای می‌گیرد. این روش‌ها برای مسائلی که تعداد متغیرها و یا تعداد مشاهدات و یا هر دو زیاد باشند مناسب هستند. در این روش‌ها به دنبال تعریف تابع خطا و حداقل کردن آن هستیم. در روش‌های ادغامی نخست هر مشاهده به صورت یک خوشه مستقل در نظر گرفته می‌شود، سپس در فرآیند خوشه‌بندی خوشه‌ها با هم ادغام می‌شوند تا به خوشه یکتایی برسیم. در روش‌های تجزیه‌ای کار برعکس است، ابتدا تمامی مشاهدات یک خوشه در نظر گرفته می‌شود و در فرآیند خوشه‌بندی، خوشه‌ها را به چند خوشه تجزیه می‌کنیم.

یکی از معروفترین و پر کاربردترین روش‌های افرازی روش k- میانگین[۲۲] است که نخستین بار توسط مک کوئین (۱۹۶۷) ارائه شد. این روش برای خوشه‌بندی داده‌های با مقادیر کمی طراحی شده است. هر خوشه دارای مرکزی بنام میانگین است. در ابتدا داده‌ها به صورت تصادفی به k خوشه تقسیم می‌شوند. در مرحله بعد، فاصله هر یک از داده‌ها از مرکز خوشه خود، که همان میانگین مشاهدات هر خوشه است، محاسبه می‌شود. در صورتی که فاصله مشاهده مورد نظر از میانگین خوشه خود زیاد و به خوشه دیگری نزدیکتر باشد، این مشاهده به خوشه‌ای که نزدیکتر است اختصاص می‌یابد. این کار تا حداقل شدن تابع خطا، که معمولا مجموع فواصل مشاهدات از مرکز خوشه خودش است، و یا تغییر نیافتن اعضای خوشه‌ها ادامه می‌یابد.

روش دیگر روش k- مدوید[۲۳] است. در این روش به جای استفاده از میانگین خوشه، می‌توان از مدوید (مشاهده‌ای که در مرکزی‌ترین مکان خوشه قرار دارد) خوشه استفاده کرد. تفاوت میانگین با مدوید در این است که میانگین داده‌ها ممکن است وجود خارجی نداشته باشد، در صورتی که مدوید وجود خارجی دارد یعنی یکی از مشاهدات مجموعه داده است.

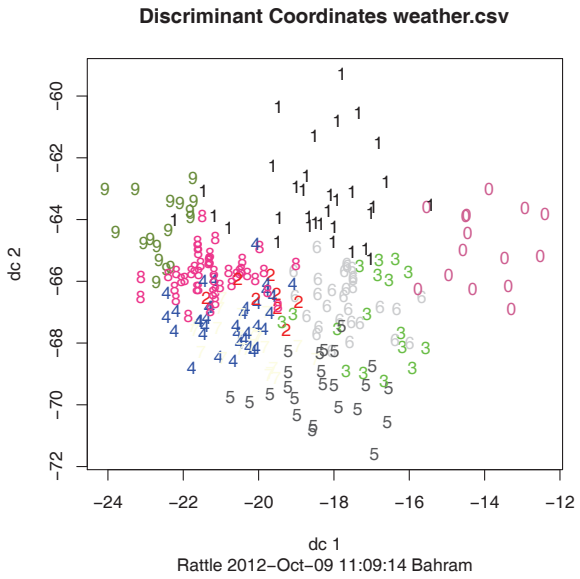
در Rattle با انتخاب بخش Cluster می‌توان به برخی از این روش‌ها از جمله روش k- میانگین، روش سلسله مراتبی ادغامی، روش BiCluster و Ewkm دسترسی پیدا کرد.

درختی که خوشه‌بندی سلسله مراتبی را نشان می‌دهد دندروگرام[۲۴] نامیده می‌شود که از رده‌بندی جانداران بر گرفته شده است. در Rattle این امکان وجود دارد که دندروگرام و همچنین نمودارهای تشخیصی[۲۵] و نمودار پراکندگی داده‌ها و نیز نمودار پراکندگی مشاهدات در خوشه‌ها را ترسیم نماییم. به عنوان مثال برای داده‌های هواشناسی بیان شده، خوشه‌بندی را به روش سلسله مراتبی با معیار فاصله اقلیدسی انجام داده و مشاهدات را در ۱۰ خوشه، خوشه‌بندی نموده‌ایم که دندروگرام و نمودار تشخیصی آن به صورت شکل‌های ۶ و ۷ هستند.



شکل ۶: دندروگرام برای مجموعه داده هواشناسی

[18] Partitional
[19] Hierarchical
[20] divisive
[21] agglomerative
[22] k-means
[23] k-medoids
[24] Dendogram
[25] Discriminant plot



شکل ۷: نمودار تشخیصی برای مجموعه داده هواشناسی

۷.۳ قواعد پیوند

یکی از وظایف مهم داده‌کاوی، کاوش در پایگاه‌های داده زمانی به منظور استخراج اطلاعات و الگو از آن‌ها است. برای این منظور رویکردهای مختلفی پیشنهاد شده است که از جمله آن‌ها می‌توان به تحلیل سری‌های زمانی، قواعد پیوند و دنباله‌کاوی[۲۶] یا تحلیل توالی اشاره نمود. از میان این روش‌ها، قواعد پیوند به بیان الگوهای موجود در یک سری از وقایع می‌پردازد، بدین‌ترتیب که بیان می‌کند، اگر بعضی از وقایع رخ دهند، آن‌گاه وقایعی دیگر نیز رخ خواهد داد. به عبارتی دیگر، این قوانین وابستگی‌ها را در حجم انبوهی از داده‌ها بیان می‌کنند.

عبارت $X \Rightarrow Y$ را در نظر بگیرید. به عنوان مثال فرض می‌کنیم عبارت X آنگاه Y بدین معنی باشد که هرگاه خریداری محصول X را بخرد، محصول Y را نیز خواهد خرید. اگر ۹۰٪ خریداران هنگامی که X را می‌خرند Y را نیز بخرند، ۹۰٪ به این قاعده اطمینان خواهیم داشت. اما این کافی نیست چون باید برای اطمینان بیشتر یک میزان تکرار مشخصی را از این قانون مشاهده کنیم. آنچه این قاعده را پشتیبانی می‌کند درصد خریدهایی است که هر دوی این محصول را با یکدیگر شامل شود. حال اگر $X \Rightarrow Y$ را یک قاعده‌ی پیوند بنامیم، مساله قواعد پیوند در اصل پیدا کردن قواعدی است که از یک حداقل پشتیبانی برخوردار بوده و به اندازه کافی به آنها اعتماد داشته باشیم. فرض کنید$J = (i_1, i_2, \cdots, i_n)$ یک مجموعه از عناصر مجزا باشد (مانند نام کالاهای مختلف یک فروشگاه)، D را مجموعه‌ای از تراکنش‌های پایگاه داده‌ای فرض کنید و T تراکنشی که زیر مجموعه‌ای از J است. اگر $A \subseteq T$ باشد آنگاهT شامل A خواهد بود. اگر $A \subset J$ ، $B \subset J$ و $A \cap B = \oslash$ آنگاه $A \Rightarrow B$ یک قاعده پیوند است. پشتیبان این قاعده، که با نماد s نشان می‌دهیم، در D بصورت درصد تراکنش‌هایی تعریف می‌شود که $A \cup B$ را شامل شوند که آن را می‌توان معادل احتمال اجتماع A و B دانست، یعنی $P(A \cup B)$. C میزان اطمینان به این قاعده را به صورت درصد تراکنش‌هایی در D که علاوه بر شمول A ، B را نیز شامل شوند تعریف می‌کنیم و معادل احتمال وقوع B است به شرط وقوع A یعنی $P(B \mid A)$.

اگر یک قاعده پیوند حداقل پشتیبانی لازم را داشته باشد مکرر[۲۷] خوانده می‌شود. به قواعدی که هم حداقل پشتیبانی را داشته و هم به میزان خاصی به آن‌ها اعتماد داشته باشیم قواعد قوی[۲۸] گفته می‌شود. به طور معمول مقادیر کمینه‌ای برای میزان پشتیبانی و اطمینان، مشخص می‌شود و فقط قواعدی مورد تأیید قرار می‌گیرند که میزان پشتیبانی و اطمینان آن‌ها از این مقدار آستانه بیشتر باشد. تحلیل‌گران قواعدی را ترجیح می‌دهند که یا میزان اطمینان و یا میزان پشتیبانی و یا هر دو آن‌ها بالا باشند.

یکی از معروفترین الگوریتم‌های مورد استفاده در قواعد پیوند، الگوریتم Apriori است. به طور خلاصه می‌توان گفت الگوریتم Apriori برای پیدا کردن مجموعه وقایع مکرر، چندین گذر را بر روی مشاهده ها انجام می‌دهد. در k-امین گذر، این الگوریتم همه مجموعه وقایع دارای k عنصر را پیدا می‌کند. در Rattle این الگوریتم و الگوریتم سبد خرید[۲۹] در بخش Associate گنجانده شده است. برای استفاده از این الگوریتم‌ها در Rattle باید میزان پشتیبانی و اطمینان را مشخص کرد که بصورت پیش فرض مقادیر$0/1$ در نظر گرفته شده است. در Rattle بصورت پیش‌فرض الگوریتم Apriori مورد استفاده قرار می‌گیرد و برای استفاده از روش سبدخرید باید گزینه Baskets را در بخش Associate انتخاب نمود.

[26] sequence mining

[27] Frequent

[28] Strong

[29] Baskets

۸.۳ مدل‌یابی برای پیشگویی

رده‌بندی در دسته وظایف پیش‌بینانه داده‌کاوی قرار می‌گیرد و عبارت است از به کارگیری تعدادی متغیر برای پیش‌بینی مقادیر ناشناخته سایر متغیرها. در این فرآیند داده‌ها به دو دسته آموزش و آزمایش تقسیم می‌شوند، داده‌های آموزش برای یادگیری مدل متغیر پیش‌بینی شونده به‌کار می‌رود و داده‌های آزمایش برای تعیین صحت مدل به‌کار می‌رود. رده‌بندی از یک تابع یا مدل نتیجه می‌شود که برچسب رده یک مشاهده را بر اساس ویژگی‌هایش مشخص می‌کند.

بخش Model در Rattle شامل بسیاری از روش‌ها و الگوریتم‌های داده‌کاوی برای مدل‌سازی و رده‌بندی مجموعه داده‌ها است. روش‌هایی چون درخت تصمیم، انواع رگرسیون (خطی، لجستیک، پروبیت، چندگانه)، ماشین بردار پشتیبان، شبکه عصبی، Forest و boost را می‌توان با استفاده از این بخش انجام داد.

۱.۸.۳ درخت تصمیم

درخت تصمیم یک روش داده‌کاوی است که غالباً برای رده‌بندی و پیش‌بینی به‌کار می‌رود. یکی از مزایای درخت تصمیم در رده‌بندی داده‌ها نسبت به سایر روش‌های رده‌بندی مانند شبکه عصبی، سادگی تفسیر و فهم برای تصمیم‌گیرندگان جهت مقایسه نتایج با دانش حوزه خودشان و تعیین اعتبار و تعدیل تصمیماتشان است. علاوه بر این، درخت‌های تصمیم می‌توانند داده‌های مختلفی را تحلیل کنند بدون این‌که به فرضیاتی در مورد توزیع داده‌ها نیاز باشد.

درخت تصمیم در ساختار درختی با شاخه و برگ ارائه می‌شود. ساختار سلسله مراتبی درخت می‌تواند سطوح مختلف فاکتورها را تحلیل کند. هر برگ نشان‌دهنده نتیجه رده‌بندی، و هر شاخه نشانگر شرایط متغیرها است. با داشتن یک مجموعه آموزشی، شامل متغیرهای ورودی و یک متغیر خروجی، می‌توان یک درخت تصمیم ایجاد کرد، این مسأله به نوع استراتژی به کار گرفته شده برای یادگیری بستگی دارد. از جمله الگوریتم‌هایی که برای ایجاد درخت تصمیم توسعه یافته‌اند می‌توان به CART[۳۰]، CHAID[۳۱]، ID3، C4.5 اشاره کرد.

CHAID یک درخت تصمیم غیر دو-دویی است که به طور خاص برای متغیرهای رده‌ای طراحی شده و می‌تواند بهترین رده‌بندی داده‌ها را بر اساس آزمون‌های معنی‌داری تعیین کند. معیار انشعاب در این الگوریتم مقدار P-value برای آزمون Chi-square می‌باشد.

CART یک درخت تصمیم دو-دویی است که گوناگونی را به عنوان معیار انشعاب قرار می‌دهد و برای این کار از معیار شاخص جینی و یا شاخص آنتروپی استفاده می‌کند. این روش درخت را با کمینه‌سازی خطای تخمینی رده‌بندی نادرست، هرس می‌نماید. این روش برای متغیرهای رده‌ای و پیوسته کاربرد دارد.

C4.5 گونه توسعه یافته یک الگوریتم درخت تصمیم معروف به نام ID3 است. معیار انشعاب در الگوریتم C4.5، نسبت gain است که نسبت اطلاعات تولید شده با هر انشعاب را نشان می‌دهد. هرس نمودن این درخت بر اساس معیار خطا است.

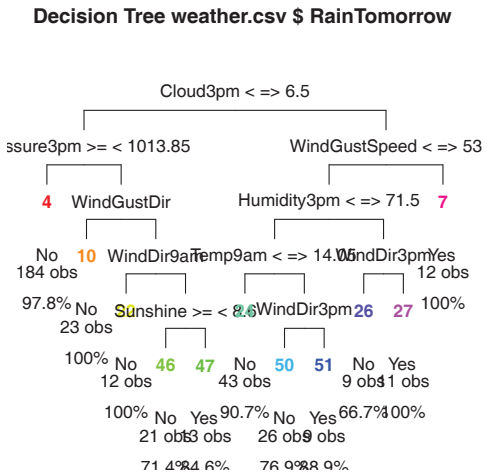
به طور کلی هدف الگوریتم‌های درخت تصمیم، بیشینه‌سازی فاصله بین رده‌ها است. اختلاف این الگوریتم‌ها در معیارهای مختلف فاصله‌ای، روش‌های هرس کردن، وضعیت داده‌های گمشده، تعداد شاخه‌ها در هر گره و نوع داده‌هایی است که با آن سر و کار دارند. می‌توان به طور خلاصه برخی از ویژگیهای درخت تصمیم را به صورت زیر بیان کرد:

- برای تقریب توابع گسسته بکار می‌رود
- برای داده‌های با حجم بالا کاراست، از این رو در داده‌کاوی استفاده می‌شود.
- می‌توان درخت را به صورت قوانین «اگر ـ آنگاه» نمایش داد که قابل فهم برای کاربر است.
- امکان ترکیب عطفی و فصلی فرضیه‌ها را می‌دهد.
- نسبت به داده‌های پرت مقاوم است.

در بخش Model با انتخاب گزینه Tree می‌توان برخی از الگوریتم‌های درخت تصمیم را انجام داد. با انتخاب گزینه Traditional می‌توان الگوریتم‌های CART ، ID3 و $C_{4.5}$ را انجام داد. برای این منظور باید مقادیر تعداد انشعاب، عمق درخت، و میزان پیچیدگی را مشخص کرد که نرم‌افزار به صورت پیش‌فرض، مقادیری را برای هر یک در نظر گرفته است. در اینجا، برای داده‌های هواشناسی این کار انجام شده است که درخت تصمیم آن به صورت شکل ۸ است.

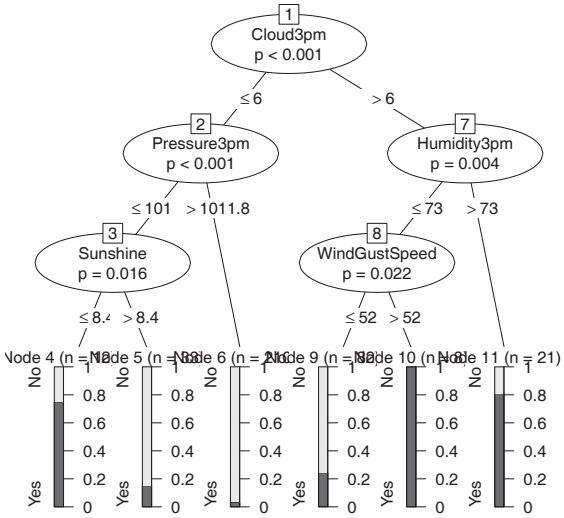
[30] Classification and regression tree

[31] Chi-squared automatic interaction detection



شکل ۸: درخت تصمیم ساخته شده به روش CART

با انتخاب گزینه Conditional می‌توان درخت تصمیم را با الگوریتم CHAID انجام داد. برای داده‌های هواشناسی درخت تصمیم به صورت شکل ۹ بدست آمده است.



شکل ۹: درخت تصمیم ساخته شده به روش CHAID

برای ساخت درخت‌های تصمیم فوق از ۳۶۶ مشاهده، ۲۰ متغیر ورودی و متغیر هدف Rain Tomorrow، که دو- دویی است، از مجموعه داده هواشناسی استفاده شده است. از میان متغیرهای ورودی ۹ متغیر برای ساخت درخت تصمیم به روش CART استفاده شده است که نام این متغیرها را می‌توان در شکل ۸ مشاهده نمود. این درخت شامل ۵۱ گره است که شماره آن‌ها به صورت اعداد رنگی در شکل ۸ نشان داده شده است. در هر گره تعداد مشاهداتی که متغیر هدف برای آن‌ها یکی از دو مقدار (Yes , No) را دارند مشخص شده است. همچنین به صورت درصد فراوانی نیز نشان داده شده است. معیار انشعاب در این درخت شاخص جینی است.

درخت ایجاد شده با روش CHAID معیار انشعاب در این درخت بر اساس مقدار P-Value برای آزمون Chi-square است. این درخت دارای ۱۱ گره می باشد. در انتهای هر شاخه میزان فراوانی هر یک از مقادیر متغیر هدف به‌صورت نمودار ستونی نشان داده شده است.

۲.۸.۳ ماشین بردار پشتیبان

ماشین بردار پشتیبان (SVM) یکی دیگر از روش‌های رده‌بندی داده‌هاست که بر پایه مفهوم صفحات تصمیم که مرز تصمیم را تعریف می‌کنند، بنا شده است. یک صفحه تصمیم داده‌های با برچسب رده مختلف را از هم تفکیک می‌کند. در برخی مواقع، به ساختارهای غیرخطی برای جداسازی بهینه مشاهدات نیاز است. فلسفه‌ای که در SVM وجود دارد این است که زمانی که برای تفکیک داده‌ها به ساختارهای پیچیده و غیرخطی صفحه تصمیم نیاز است، داده‌های اصلی با به کارگیری مجموعه‌ای از توابع ریاضی که کرنل[۳۲] نام دارند، در فضای جدیدی نگاشت داده می‌شوند. در فضای جدید، داده‌های نگاشت‌شده به صورت خطی قابل تفکیک هستند. بنابراین به جای ساختن یک منحنی پیچیده جداساز، باید به دنبال یک خط بهینه جداساز باشیم. مسأله یادگیری SVM می‌تواند به صورت یک مسأله بهینه‌سازی محدب فرمول‌بندی شود.

الگوریتم‌های مبتنی بر ماشین‌های بردار پشتیبان الگوریتم‌هایی هستند که سعی می‌کنند یک حاشیه[۳۳] را بیشینه کنند. این الگوریتم‌ها، در فضای دو بعدی، برای پیدا کردن خط جدا کننده رده‌ها، از دو خط موازی شروع کرده، این خطوط را در خلاف جهت یکدیگر حرکت می‌دهند تا هر کدام از خطوط به یک مشاهده از یک رده خاص برسد. پس از انجام این مرحله، میان دو خط موازی یک نوار یا حاشیه شکل می‌گیرد. هر چه پهنای این نوار بیشتر باشد، به این معنا است که الگوریتم توانسته است هدف بیشینه‌سازی حاشیه را محقق سازد. در مرکز حاشیه، خط

[32] kernel

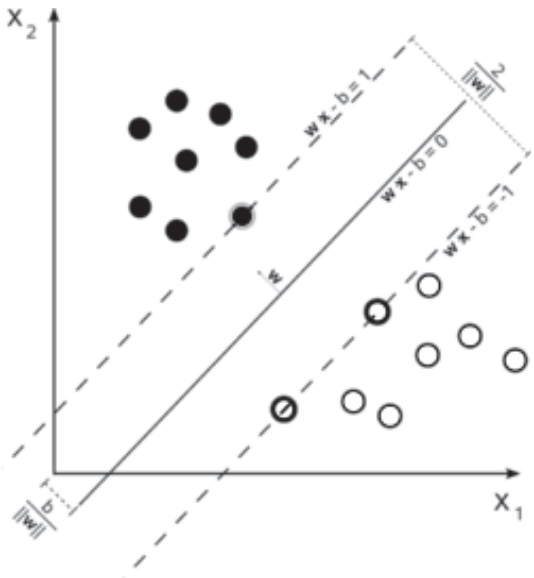
[33] Margin

جداکننده رده‌ها یا همان خط مرکزی قرار می‌گیرد. حال از بین خطوطی که رسم می‌شوند، خطی که حاشیه کناری آن بیشترین باشد، به عنوان خط جدا کننده رده‌ها انتخاب می‌شود شکل (۱۰). محاسبه حاشیه به صورت رابطه (۱) است

$$M \arg in = \frac{2}{||\overrightarrow{w}||^2} \qquad (۱)$$

بیشینه نمودن حاشیه معادل با کمینه کردن رابطه (۲) است

$$L(w) = \frac{||\overrightarrow{w}||^2}{2} \qquad (۲)$$



شکل ۱۰: حاشیه خطوط رده‌بند در ماشین بردار پشتیبان

فرض کنید برای n مشاهده، مقادیر d متغیر بیان‌کنندهٔ ویژگی‌ها و یک متغیر هدف به‌صورت دو-دویی داریم. هدف اصلی، حل یک مسئله رده‌بندی دو کلاسه به صورت بهینه است. فرض کنید بخواهیم این دو کلاس را از یکدیگر تفکیک کنیم در این صورت تابع تمایز $f(x)$ و اَبَرصفحه H به‌صورت رابطه (۳) تعریف می‌شوند

$$\begin{aligned} H &: w.x + b = 0 \\ f(x) &= sign(w.x + b) \end{aligned} \qquad (۳)$$

در این رابطه بردار وزن w، بردار عمود بر ابرصفحه جداکننده بوده و b مقدار اریبی است و منظور از $w.x$ حاصل‌ضرب داخلی است. واپنیک[۳۴] (۱۹۹۵) ثابت کرد که بُعد VC[۳۵] برای رده‌بندی کننده‌هایی از نوع ابرصفحات کانونی، دارای یک کران بالاست که این کران بالا با توان دوم نرم بردار وزن یعنی $||w||^2$ نسبت مستقیم دارد. در واقع اگر ما $||w||^2$ را محدود کرده و کمینه کنیم، بعد VC رده‌بند را کمینه کرده و تخمین مقدار ریسک به صورت احتمالی دقیق‌تر بوده و خاصیت تعمیم رده‌بند بیشتر خواهد بود. فرض کنید داده‌های دو کلاس جدایی‌پذیر باشند و بردارهای ویژگی مرزی کلاس اول روی ابرصفحه H^+ و بردارهای ویژگی کلاس دوم روی ابرصفحه H^- قرار گیرند. ابرصفحات H^+ و H^- به صورت رابطه (۴) تعریف می‌شوند.

$$\begin{aligned} H^+ &: w.x + b = +1 \\ H^- &: w.x + b = -1 \end{aligned} \qquad (۴)$$

در واقع معادله یک رده‌بند ابرصفحه‌ای با ناحیه حاشیه بهینه به صورت رابطه (۵) خواهد بود.

$$\begin{aligned} &Min \quad \phi(w) = \tfrac{1}{2}||w||^2 \\ &Subject\ \ to \quad y_i(w.x_i + b) \geq 1 \\ &i = 1, 2, ..., n \end{aligned} \qquad (۵)$$

مساله فوق، یک مساله بهینه‌سازی از نوع محدّب و درجه دوم است. برای حل این مساله باید تابع لاگرانژ را تشکیل داد (معادله ۶) و ضرایب لاگرانژ را بدست آورد. در واقع هر یک از ضرایب لاگرانژ بدست آمده متناظر با یکی از الگوهای X است. الگوهای x_i را که متناظر با ضرایب مثبت α_i هستند بردار پشتیبان svi می نامند.

$$L(w, b, \alpha) = \frac{1}{2} w.w - \sum_{i=1}^{n} \alpha_i (y_i(w.x_i + b) - 1) \qquad (۶)$$

همچنین می‌توان با بدست آوردن مقدار w و قرار دادن مقدار آن در رابطه (۶) مسئله را به مسئله دوگان تبدیل کرد و جواب‌های بهینه را برای مسئله دوگان بدست آورد.

برای اجرای روش ماشین بردار پشتیبان در Rattle باید در بخشModel گزینه SVM را انتخاب کرد. در ابتدا باید تابع کرنل مورد نظر را انتخاب کرد. Rattle از چندین تابع کرنل پشتیبانی می‌کند، توابعی همچون خطی ساده، چندجمله‌ای، لاپلاس، بسل، تانژانت هیپربولیک و ... .

۳.۸.۳ روش‌های تجمیعی($Boosting$)

روش‌های تجمیعی، الگوریتم‌هایی هستند که یک مجموعه از رده‌بندهای ضعیف را گرفته و خروجی آنها را با یکدیگر ترکیب می‌نماید تا رده‌بند

[34] Vapink

[35] Vapnik-chervonenkis

نهایی به گونه‌ای تشکیل شود که کارایی آن از کارایی تک‌تک رده‌بندهای استفاده شده در الگوریتم بیشتر باشد. در نهایت رده مشاهدات استفاده نشده در مرحله ارزیابی را با ترکیب خروجی تک‌تک رده‌بندهای استفاده شده تعیین می‌کنند.

استفاده از الگوریتم‌های تجمیعی باعث کاهش خطا می‌شود. در واقع هنگامی که برای رسیدن به خروجی نهایی، خروجی حاصل از چند رده‌بند با یکدیگر ترکیب می شوند، رده‌بندها خطای یکدیگر را می پوشانند. به طور کلی، هر چه تعداد رده‌بندهای ضعیف در روش‌های تجمیعی بیشتر باشد، خطای رده‌بند نهایی به میزان بیشتری کاهش می‌یابد. بدیهی است که نکته منفی که در اینجا وجود دارد کندتر شدن فرآیند یادگیری الگوریتم تجمیعی خواهد بود. به هر حال توجه داریم که الگوریتم‌های تجمیعی کندتر از الگوریتم‌های معمولی هستند. در الگوریتم‌های تجمیعی دو ایده برای ترکیب رده‌بندها وجود دارند که

- Bagging
- Boosting

Boosting: هستند. روش *Boosting* از یک الگوریتم تکرار شونده استفاده می‌کند تا به طور تطبیقی، توزیع نمونه‌های آموزشی را تغییر دهد. همچنین این روش در فرآیند یادگیری بیشتر روی مشاهداتی که در مراحل قبلی به اشتباه رده‌بندی شده‌اند تمرکز می‌کند. در ابتدا به تمام N مشاهده موجود وزن برابر نسبت داده می‌شود. در انتهای هر مرحله ممکن است وزن مشاهدات تغییر کند، به این صورت که وزن مشاهداتی که به اشتباه رده‌بندی شده‌اند افزایش یافته و وزن مشاهداتی که به درستی رده‌بندی شده‌اند کاهش می‌یابد. همچنین وزن مشاهداتی که مدل ساخته شده نتوانسته آنها را رده‌بندی کند ثابت باقی می‌ماند. در این صورت پس از گذشت چند مرحله رده‌بندها برای یادگیری به سمت مشاهداتی می‌روند که وزن بیشتری داشته باشند، یعنی مشاهداتی که به اشتباه رده‌بندی شده‌اند. زیرا اگر براساس این مشاهدات آموزش ببینند و مدل بسازند، کارایی مدل ساخته شده افزایش می‌یابد. این فرآیند تا زمانی که مجموع وزن مشاهدات از آستانه‌ای مشخص کمتر شود ادامه می‌یابد. در Rattle نیز امکان استفاده از روش Boosting در زیربخش Boost برای رده‌بند درخت تصمیم فراهم شده است. این روش را می توان با انتخاب گزینه Boost و مشخص کردن تعداد درخت‌ها، عمق درخت‌ها، میزان پیچیدگی و ... انجام داد. شایان ذکر است در Rattle برای هر یک از این موارد مقداری به صورت پیش فرض در نظر گرفته شده است که کاربر می‌تواند آن را به میزان دلخواه خود تغییر دهد.

۹.۳ ارزیابی

دانشی که در مرحله یادگیری مدل تولید می‌شود، باید در مرحله ارزیابی مورد تحلیل قرار بگیرد تا بتوان ارزش آن را تعیین نمود و در پی آن کارایی الگوریتم یادگیرنده مدل را نیز مشخص کرد. بخش Evaluate در Rattle شامل بسیاری از معیارهای ارزیابی مدل ایجاد شده توسط کاربر است. در این بخش الگوریتم‌هایی را که کاربر مورد استفاده قرار داده است، برای ارزیابی فعال نموده است. از جمله معیارهای ارزیابی که در این بخش قابل اجرا هستند می‌توان به ماتریس خطا یا ماتریس درهم‌ریختگی[۳۶]، منحنیROC [۳۷]، Cost Curve و ... اشاره کرد. می‌توان ارزیابی را بر روی هر یک از نمونه آموزشی، نمونه برای ارزیابی، نمونه برای آزمون و یا تمام مجموعه داده‌ها انجام داد. در Rattle این کار با انتخاب هر یک از گزینه‌های زیربخش Data در بخش Evaluate امکان‌پذیر است. همچنین می‌توان داده‌های جدیدی را فراخوانی کرد و ارزیابی را بر روی آنها انجام داد که این امر با انتخاب هر یک از گزینه‌های CSV File و R Dataset زیر بخش Data در بخش Evaluate امکان‌پذیر است.

[36] Confusion Matrix

[37] Receiver Operating Characteristic

مراجع

[۱] مشکانی،ع و ناظمی،ع . (۱۳۸۸)، *مقدمه ای بر داده کاوی* ، ناشر علی مشکانی ، مشهد

[2] Fayyad. U.M, Piatetsky Shapiro, Smyth P and Uthurusamy R. (1996), *Advances in Knowledge Discovry and Data Mining* , Menlo park California:AAAI Press.

[3] George, H. J. (1996) *Enhancements to the Data Mining Process*, Ph. D. thesis, Department of Computer Science, Stanford University.

[4] George, H. j. (1997) *Enhancements to the Data Mining Process*, Ph. D. thesis, Department of Computer Science, Stanford University.

[5] Williams. G, (2011) *Data Mining with Rattle and R: The Art of Excavating Data for Knowledege Discovery*, Use R, Springer Science+Business Media, LLC 2011.