فصل پنجم

پایگاه‌داده

4-1: مقدمه

سیستم‌های شناسایی آماری الگو با استفاده از داده‌های آماری که از طریق نمونه‌برداری و مشاهدات بدست می‌آیند، از الگوریتم‌های معرفی شده در بخش قبل استفاده نموده و نمونه‌های جدید را شناسایی می‌کنند. به مجموعه‌ای از داده‌ها که در روش‌های شناسایی الگو مورد استفاده قرار می‌گیرد، پایگاه‌داده گفته می‌شود. پایگاه‌داده علاوه بر وظیفه ذاتی خود که تغذیه یک روش شناسایی آماری الگو است، باید دارای خصوصیات دیگری نیز باشد. یکی از مباحث مهم در کارهای آماری مبحث اعتبارسنجی[1] است. هنگام ارزیابی یک الگوریتم بهتر است از یک پایگاه‌داده مناسب استفاده شود تا محقق قادر باشد روش را به صورت کامل مورد بررسی و مقایسه قرار دهد. یکی از مواردی که در تایید اعتبار کار می‌تواند نقش داشته باشد، داشتن یک بانک اطلاعاتی[2] استاندارد است. بانک اطلاعاتی ما باید ویژگی‌های مناسب یک بانک اطلاعاتی مانند جامع بودن را داشته باشد تا وقتی ارزیابی‌کننده، روش را مورد بررسی قرار می‌دهد پیش‌بینی درستی از صحت روش مذکور داشته باشد. مثلا اگر بانک اطلاعاتی که در بررسی روش مذکور استفاده شده است کوچک باشد، اعتبار آن کم است. با توجه به اهمیت پایگاه‌داده در این فصل به بررسی آن خواهیم پرداخت.

4-2: بخش‌های مختلف بانک اطلاعاتی

در روش‌های دسته بندی، از یک الگوریتم شامل بخش‌های آموزش و تست کمک می‌گیرند. به این ترتیب داده‌های ورودی در پایگاه داده به مجموعه‌های مجزایی تقسیم می‌شوند. این دو مجموعه کاملا از هم مجزا هستند و نباید اشتراکی با هم داشته باشند. سپس بر اساس یک الگوریتم مشخص و بر اساس داده‌های آموزش، الگوریتم مشخص شده آموزش می‌بیند و در نهایت مطلوبیت کارکرد الگوریتم با داده‌های تست کنترل می‌شود. در ادامه به معرفی این بخش‌ها خواهیم پرداخت.

---

[1] Validation
[2] Database

4-2-1: مجموعه آموزش [3]

مجموعه آموزش، مجموعه‌ای از داده‌ها است که در زمینه‌های مختلف علم اطلاعات برای کشف ارتباطات قابل پیش بینی، مورد استفاده قرار می‌گیرد. مجموعه‌های آموزش در عرصه‌های مختلف مانند هوش مصنوعی، یادگیری ماشین، برنامه نویسی ژنتیک، سیستم‌های هوشمند و آمار استفاده می‌شود. در همه این زمینه‌ها، مجموعه آموزش نقش مشابهی دارد و در کنار مجموعه تست و اعتبارسنجی مورد استفاده قرار می‌گیرد.

در مسائل دسته‌بندی، اندازه‌گیری کارایی طبقه‌بندی‌کننده در رابطه با میزان خطا صورت می-گیرد. طبقه‌بندی‌کننده، کلاس هر نمونه را پیش بینی می‌کند. اگر طبقه‌بندی درست باشد عمل موفقیت‌آمیز است وگرنه خطا محسوب می‌شود که بررسی نتایج آموزش حاصل از مجموعه آموزشی، توسط مجموعه داده‌های دیگری که در ادامه معرفی می‌کنیم صورت می‌گیرد.

4-2-2: مجموعه تست [4]

یک مدل معمولا توسط ماکزیمم کردن کارایی روی داده‌های مجموعه آموزش، آموزش می-بیند اما تاثیر آن بر اساس کارایی روی داده‌های آموزشی تعیین نمی شود بلکه بر اساس توانایی انجام آن روی داده‌های جدید بررسی می‌شود. میزان خطا روی مجموعه آموزش شاخص خوبی برای کارایی آینده مدل نیست. زیرا طبقه‌بندی‌کننده از داده‌های مجموعه آموزشی، آموزش دیده است اما داده‌های جدید دقیقا همان داده‌های آموزش نخواهند بود. هر تخمینی درباره کارایی بر اساس آن داده‌ها خوش‌بینانه خواهد بود و ممکن است مایوس‌کننده باشد. به همین دلیل مجموعه‌های آموزش و تست، باید از دو مجموعه داده مستقل انتخاب شود. مجموعه تست مجموعه‌ای از داده‌هاست که از مجموعه آموزش مستقل است اما همان توزیع احتمال داده‌های آموزش را پیگیری می‌کند و برای ارزیابی کارایی یک طبقه‌بندی‌کننده که به صورت کامل آموزش دیده، استفاده می‌شود.

---

3 Train set
[4] Test set

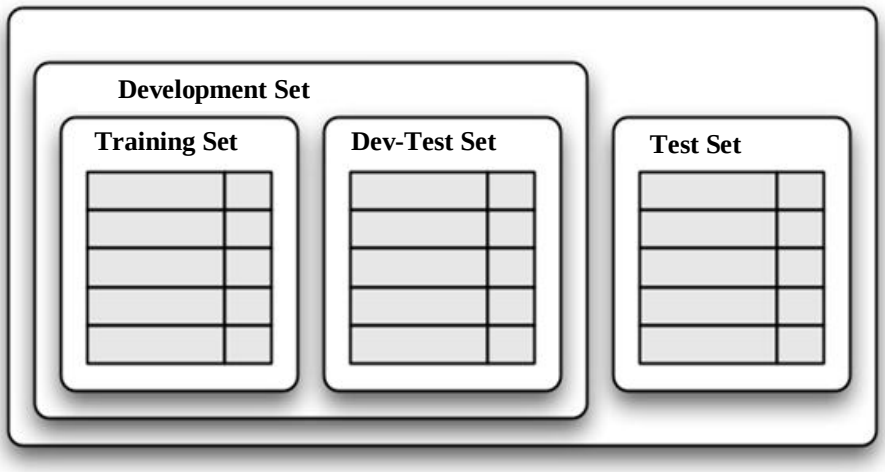
3-2-4: مجموعه اعتبار سنجی [5]

اگر مدل برای داده‌های آموزش خیلی مناسب‌تر از داده‌های تست ارزیابی شود، احتمالا علت آن این است که روی هم افتادگی داده‌ها[6] اتفاق افتاده است. به منظور اجتناب از این رویداد، وقتی پارامترهای طبقه‌بندی نیاز به تنظیم دارند، لازم است علاوه بر مجموعه آموزش و تست، مجموعه اعتبارسنجی نیز داشته باشیم. مثلا اگر به دنبال مناسب‌ترین طبقه‌بندی‌کننده برای مساله هستیم، مجموعه آموزش برای آموزش الگوریتم‌های کاندید استفاده می‌شود، مجموعه اعتبارسنجی برای مقایسه کارایی آن‌ها و تصمیم‌گیری برای انتخاب یکی از آن‌ها استفاده می‌شود و بالاخره مجموعه تست برای به دست آوردن مشخصات کارایی مانند صحت، حساسیت و مانند آن استفاده می‌شود.

از این مجموعه برای سازگار کردن پارامترهای طبقه‌بندی‌کننده در مساله استفاده می‌شود. مثلا در شبکه‌های MLP ما از این مجموعه برای پیدا کردن تعداد واحدهای بهینه لایه مخفی یا تعیین نقطه توقف برای الگوریتم انتشار به عقب[7] استفاده می‌کنیم.

---

[5] Validation set
[6] Overfitting
[7] Back propagation



شکل4-1: بخش‌های مختلف بانک اطلاعاتی

4-3: مشخصه‌های یک پایگاه‌داده استاندارد

در ابتدای بخش در مورد اهمیت استفاده از یک پایگاه‌داده استاندارد در ارزیابی الگوریتم مورد بررسی صحبت کردیم. در این بخش به بیان نکاتی که در مورد ویژگی‌های یک پایگاه‌داده مناسب وجود دارد می‌پردازیم.

4-3-1: در دسترس بودن [8]

منظور از در دسترس بودن پایگاه‌داده، این است که پایگاه‌داده برای هرکسی، صرف نظر از حرفه، هدف یا رابطه، در دسترس باشد.

[8] Publicly available

2-3-4: مجموعه تست جدید[9]

مجموعه تست باید جدید باشد، یعنی نمونه‌های آن در مرحله آموزش، آموزش ندیده باشند تا نرخ خطا روی مجموعه تست شاخص خوبی از کارایی آینده باشد.

3-3-4: جامعیت پایگاه‌داده

پایگاه‌داده باید جامع باشد و همه موارد لازم را در بر بگیرد و نماینده واقعی برای جامعه باشد.

4-3-4: پراکندگی[10]

در برخی از پایگاه‌داده‌ها، مانند پایگاه‌داده‌هایی با ویژگی‌های غیرمتقارن، بیشتر خصوصیات یک نمونه مقادیر صفر دارند. در بیشتر موارد، کمتر از 1% ورودی‌ها غیرصفر هستند. در شرایط عملی، پراکندگی، یک ویژگی خوب برای پایگاه‌داده است. زیرا عموما مقادیر غیرصفر پایگاه‌داده، نیاز به ذخیره‌سازی و دستکاری دارند که باعث صرفه‌جویی قابل توجهی در زمان محاسبات و ذخیره سازی می‌شود.

5-3-4: رزولوشن پایگاه‌داده

پایگاه‌داده‌ها دارای رزولوشن‌های متفاوتی هستند. داده‌ها در سطوح دقت متفاوتی می‌توانند بدست بیایند و ویژگی‌های داده‌ها، دارای دقت‌های متفاوتی هستند. مثلا، سطح زمین با دقت چند متر ناهموار به نظر می‌رسد، اما با دقت ده‌ها کیلومتر نسبتا صاف به نظر می‌رسد.

---

[9] Unsight
[10] Sparsity

4-3-6: اندازه پایگاه‌داده

اگر تعداد زیادی داده در دسترس باشد، مشکلی وجود ندارد. تعداد زیادی ار آنها را برای آموزش برمی‌داریم و مجموعه بزرگی از داده‌های مستقل متفاوت را برای مرحله تست استفاده می‌کنیم. به شرط آنکه نمونه‌های هر دو مجموعه قابل ارائه باشد، نرخ خطا روی مجموعه تست شاخص خوبی از کارایی آینده خواهد داد. عموما مجموعه آموزش بزرگ‌تر طبقه‌بندی بهتری را در برخواهد داشت، اگرچه وقتی داده‌های آموزش از حجم معینی تجاوز می‌کند بازده شروع به کاهش می‌کند. از سوی دیگر، مجموعه تست بزرگ‌تر، دقت بیشتری را روی تخمین خطا به وجود می‌آورد. دقت تخمین خطا می‌تواند از روی آمار ارزیابی شود.

مشکل واقعی وقتی اتفاق می‌افتد که منبع وسیعی از داده در دسترس نباشد. در بسیاری از شرایط داده‌های آموزش و داده‌های تست برای بدست آوردن تخمین‌های خطا باید به صورت دستی، دسته‌بندی شوند. این مساله تعداد داده‌ای که می‌تواند برای آموزش، تست و اعتبارسنجی استفاده شود را محدود می‌کند. از این پایگاه‌داده، تعداد خاصی برای تست نگه داشته می‌شود و باقیمانده آن برای مرحله آموزش استفاده می‌شود (اگر لازم باشد بخشی از آن برای اعتبارسنجی کنار گذاشته می‌شود). در اینجا یک مشکل وجود دارد: برای پیدا کردن یک طبقه‌بندی‌کننده خوب ما نیاز داریم هر اندازه از داده‌ها را که ممکن است برای آموزش استفاده کنیم و برای بدست آوردن تخمین خطای خوب، ما نیاز داریم هر اندازه از داده‌ها را که ممکن است برای تست استفاده کنیم. در بخش بعدی روش استفاده شده برای برخورد با این مساله را بررسی می‌کنیم.

4-3-6-1: اعتبارسنجی متقابل[11]

وقتی مجموعه پایگاه‌داده دارای تعداد محدودی داده باشد نمی توانیم دسته‌بندی مناسبی را برای مجموعه‌های آموزش و تست داشته باشیم. تعداد خاصی از داده‌ها را برای مرحله تست قرار می‌دهیم و باقیمانده داده‌ها را برای مرحله آموزش استفاده می‌کنیم (اگر نیاز باشد مقداری را برای مرحله اعتبارسنجی قرار می‌دهیم).

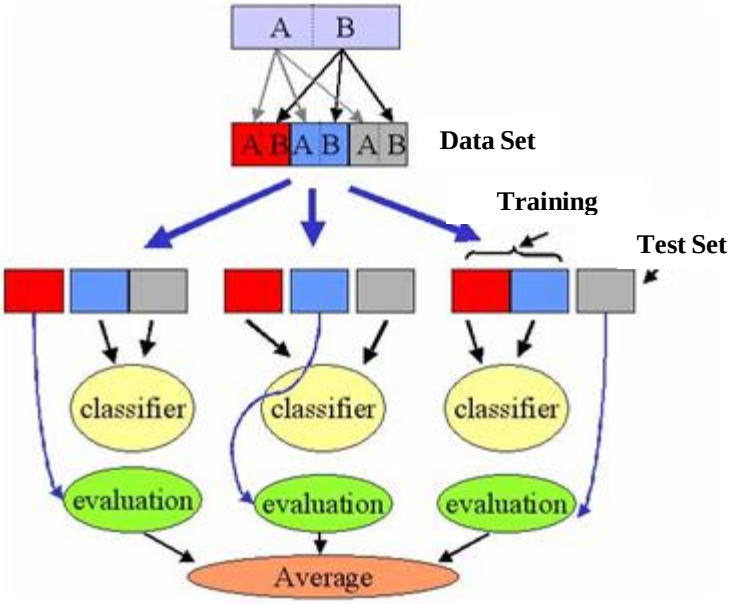
---

[11] Cross-validation

در موارد عملی معمولا یک سوم داده‌ها برای تست در نظر گرفته می‌شود و دوسوم باقیمانده برای آموزش استفاده می‌شود. ممکن است بدشانس باشیم و نمونه مورد استفاده برای آموزش یا تست ممکن است نماینده خوبی نباشد. به طورکلی، شما نمی‌توانید بگویید که یک نمونه نماینده خوبی هست یا نه. اما یک بررسی ساده وجود دارد که ممکن است ارزشمند باشد: هرکلاس در پایگاه‌داده، باید در حدود نسبتا مناسبی در مجموعه‌های آموزش و تست نشان داده شده باشد.

اگر با بدشانسی، مجموعه آموزش فاقد نمونه‌ای از یک کلاس خاص باشد، شما به سختی می‌توانید انتظار داشته باشید که طبقه‌بندی‌کننده از آن داده‌ها روی نمونه‌های آن کلاس به خوبی آموزش دیده باشد. در چنین مواقعی از روش اعتبارسنجی متقابل استفاده می‌کنیم که از طریق آن می‌توانیم همه داده‌ها را برای آموزش استفاده کنیم و به صورت غیرمستقیم عمل تست را با همه داده‌ها انجام دهیم.

یک تکرار از اعتبارسنجی متقابل شامل تقسیم‌بندی نمونه به زیر مجموعه‌هایی از داده‌های مکمل است، سپس آنالیز روی مجموعه آموزش انجام می‌شود و عمل اعتبارسنجی روی زیر مجموعه دیگر (مجموعه اعتبارسنجی) انجام می‌شود. معمولا چندین تکرار از اعتبارسنجی متقابل با استفاده از تقسیم‌بندی‌های متفاوت انجام می‌شود و میانگین نتایج اعتبارسنجی در بیشتر از دو تکرار را به دست می‌آوریم. انواع مختلفی برای اعتبارسنجی متقابل وجود دارد که در ادامه به معرفی آن‌ها خواهیم پرداخت.

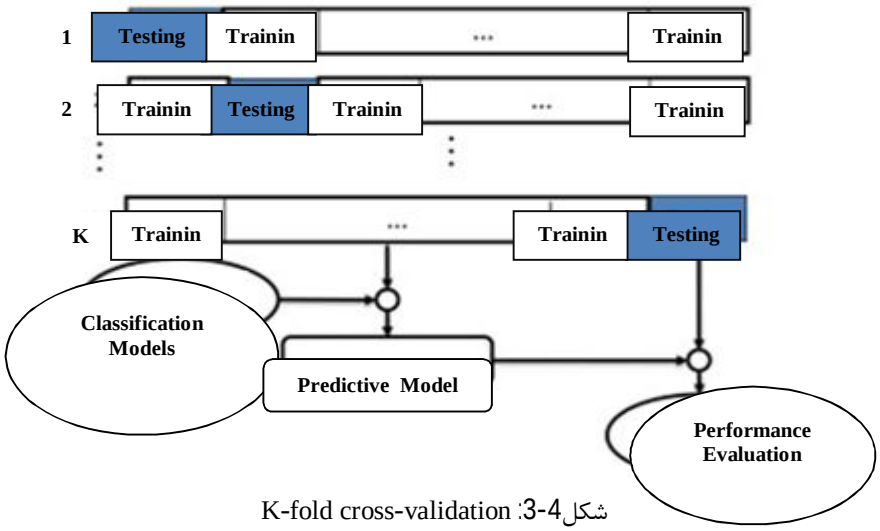


شکل4-2: cross–validation

4-3-6-2: اعتباردهی دسته‌ای با k زیرنمونه[12]

در این نوع از اعتبارسنجی متقابل ، نمونه اصلی به صورت تصادفی به K زیرنمونه با اندازه مساوی تقسیم می‌شود. از بین این k زیرنمونه، یکی از آنها به عنوان داده اعتبارسنجی برای مرحله تست مدل نگه داشته می‌شود و K-1 زیرنمونه باقیمانده به عنوان داده‌های آموزش استفاده می‌شود. سپس فرآیند اعتبارسنجی متقابل به تعداد K بار تکرار می‌شود و هر یک از K زیرنمونه دقیقا یک بار به عنوان داده اعتبارسنجی مورد استفاده قرار می‌گیرد. سپس از K نتیجه حاصل میانگین گرفته می‌شود تا یک برآورد تولید کند. فایده این روش نسبت به نمونه‌گیری تصادفی تکرار شده این است که در این روش، همه نمونه‌ها هم برای آموزش و هم برای تست استفاده می‌شوند و ازهر نمونه دقیقا یک بار برای اعتبارسنجی استفاده می‌شود. معمولا در این روش از K=10 استفاده می‌شود اما در کل K یک پارامتر غیرثابت است.

[12] K-fold cross-validation



شکل4-3: K-fold cross-validation

4-3-6-3: اعتباردهی دسته‌ای با دو زیرنمونه[13]

اعتباردهی دسته‌ای با دو زیرنمونه ساده‌ترین نوع اعتباردهی دسته‌ایاست که همچنین به آن روش بیرون‌نگهدار[14] نیز گفته می‌شود. برای هر زیرنمونه، ما به طور تصادفی داده‌هایی را به دو مجموعه d_0 و d_1 نسبت می‌دهیم به طوری‌که هر دو مجموعه اندازه یکسانی دارند (برای این کار معمولا آرایه داده‌ها را که نامرتب هستند به دو قسمت تقسیم می‌کنیم). سپس عمل آموزش را بر روی d_0 و تست را بر روی d_1 انجام می‌دهیم. در ادامه آموزش را بر روی مجموعه d_1 و تست را بر روی مجموعه d_0 انجام می‌دهیم. مزیت استفاده از این روش این است که مجموعه‌های آموزش و تست، هر دو بزرگ هستند و در هر زیرنمونه، هر داده هم برای آموزش و هم برای تست استفاده می‌شود.

---

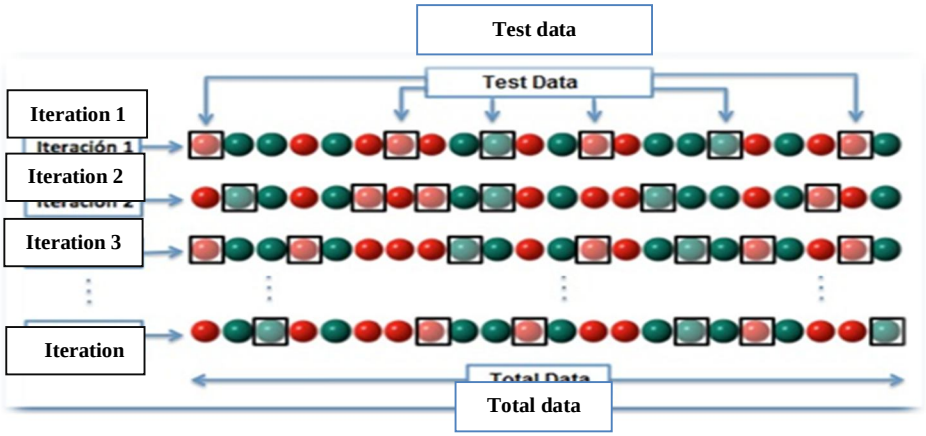
[13] 2-fold cross-validation
[14] Holdout

4-6-3-4: زیرنمونه‌برداری تصادفی تکراری[15]

روش بیرون‌نگهدار برای بهبود تخمین کارایی طبقه‌بند، می‌تواند چندین بار تکرار شود. این روش به عنوان زیرنمونه‌گیری تصادفی شناخته می‌شود. در نظر بگیرید که acc_i دقت مدل برای تکرار iام باشد. دقت کلی به این صورت محاسبه می‌شود:

$$acc = \frac{acci}{k}$$

این روش هنوز با برخی مشکلاتی که در روش بیرون‌نگهدار وجود دارد مواجه است زیرا با وجود امکان آن، داده زیادی را برای آموزش به کار نمی برد. همچنین بر تعداد دفعاتی که یک رکورد برای آموزش و تست استفاده می‌شود کنترلی ندارد. در نتیجه بعضی از رکوردها ممکن است بیشتر از دیگر رکوردها برای آموزش استفاده شوند.



شکل4-4: روش زیرنمونه‌برداری تصادفی

[15] Repeated random sub-sampling

4-3-6-5: اعتباردهی دسته‌ای با یک نمونه بیرون[16]

همان‌طور که از نام این روش پیداست، این روش شامل استفاده از یکی از رکوردهای پایگاه‌داده به عنوان مجموعه تست و باقیمانده رکوردها، به عنوان داده برای مرحله آموزش می‌باشد. این روند تکرار می‌شود، به طوری‌که هر رکورد در نمونه یک بار به عنوان داده تست استفاده شود. این روش، مشابه اعتباردهی دسته‌ای با k زیرنمونه است که در آن k برابر تعداد رکوردها در نمونه اصلی است. مزیت این روش استفاده از تعداد داده زیاد، تا جایی که ممکن است، برای مرحله آموزش است. عیب این روش این است که از لحاظ محاسباتی گران است، زیرا به تعداد تکرارهای زیادی برای آموزش نیاز دارد.

شکل۴-۵: روش اعتباردهی دسته‌ای با یک نمونه بیرون

[16] Leave-one-out cross-validation

4-4: روش‌های نمونه‌گیری

1-4-4: حجم نمونه

حجم نمونه مستقل از حجم جمعیت است و بستگی به منابع قابل دسترس و میزان دقتی که احتیاج است دارد. در اینجا اشاره به این نکته لازم است که جمعیت‌های متغیر(جمعیت‌هایی که دارای تغییرات زیاد هستند) احتیاج به حجم نمونه بزرگتری دارند. به عنوان مثال اگر بدانیم جمعیت مورد مطالعه شامل افرادی با عقاید متفاوت یا اشیایی از انواع مختلف هستند به یک نمونه با حجم بزرگ احتیاج داریم تا بیانگر جمعیت باشد.

حجم نمونه از نظر تامین دقت نتایج نمونه‌گیری و صرفه‌جویی در میزان وقت و هزینه آن، از اهمیت خاصی برخوردار است. بدیهی است بزرگ بودن حجم نمونه موجب صرف هزینه و وقت زیاد و کوچک بودن آن موجب عدم دقت کافی برآوردگرهاست.

روش‌های انتخاب نمونه را می‌توان به دو گروه احتمالی و غیراحتمالی (تصادفی و غیرتصادفی) تقسیم کرد. در روش احتمالی این امکان وجود دارد که بر اساس نتایج حاصل از نمونه، با اطمینان قابل اندازه‌گیری، درباره پارامترهای جامعه قضاوت کرد. در صورتی که نمونه‌گیری غیراحتمالی فاقد این خاصیت است. انتخاب نمونه در نمونه‌گیری غیراحتمالی بر اساس تشخیص و صلاح محقق انجام می‌گیرد نه بر اساس تصادف و احتمال تعیین شده قبلی. می‌توان گفت، تفاوت نمونه‌گیری احتمالی و غیراحتمالی در این است که در نمونه‌گیری غیراحتمالی شانس (تصادف) دخیل نیست. به عبارت دیگر نمونه‌های غیراحتمالی نمی‌توانند به تئوری احتمال وابسته باشند (برای تعیین میزان خطای برآورد، نمی‌توان از روش‌های آماری مبتنی بر اصول احتمال استفاده کرد.). البته این بدان معنا نیست که نمونه‌های غیراحتمالی "نماینده" جامعه نیستند.

در نمونه‌گیری احتمالی، چون هر عضو (واحد جامعه) احتمال انتخاب معلومی دارد، برآوردهای به‌دست آمده از جمعیت قابل اعتماد هستند. حداقل با یک نمونه احتمالی می‌دانیم که بخت یا احتمال اینکه نمونه "نماینده" جامعه باشد چقدر است. همچنین می‌توان فواصل اطمینان را

برای پارامتر جامعه برآورد کرد. نمونه‌های غیراحتمالی ممکن است نماینده جامعه باشند و یا نباشند و تشخیص این موضوع اغلب مشکل و گاهی غیرممکن است.

4-4-2: انواع روش‌های نمونه‌گیری

4-4-2-1: نمونه‌گیری تصادفی ساده[17]

نمونه‌گیری تصادفی ساده روشی است که در آن هر واحد نمونه‌گیری احتمال انتخاب برابر دارد و تنها شانس معین می‌کند کدام یک از واحدهای خاص جامعه انتخاب شود. چهارچوب نمونه‌گیری، تقسیم‌بندی نمی شود. علاوه بر این، هر زوج مشخصی از عناصر به همان نسبت زوج‌های دیگر، شانس یکسانی برای انتخاب دارند که باعث می‌شود خطا حداقل شود و آنالیز نتایج را ساده می‌کند. واریانس نتایج مجزا بین نمونه‌ها، شاخص خوبی از واریانس کل جمعیت است که تخمین دقت نتایج را نسبتا آسان می‌سازد.

همچنین وقتی نمونه‌گیری از یک جمعیت هدفی که به طور غیرمعمول بزرگ است انجام می‌شود، نمونه‌گیری تصادفی ممکن است خسته‌کننده و کند باشد. نمونه‌گیری تصادفی ساده همیشه یک طراحی ESP (احتمال مساوی برای انتخاب) است، اما همه طراحی‌های ESP[18] نمونه‌گیری تصادفی ساده نیستند.

تهیه چهارچوب نمونه‌گیری، تصمیم درباره اندازه نمونه و انتخاب تصادفی تعداد داده‌های مورد نیاز اصول این روش را تشکیل می‌دهند. نمونه‌گیری تصادفی تضمین نمی کند که ویژگی‌های نمونه و جمعیت دقیقا یکی خواهد بود. تفاوت‌های شانسی وجود خواهد داشت، ولی با روش‌های آماری مناسب می‌توان احتمالی که این تفاوت‌ها در محدوده معینی قرار دارند، را محاسبه کرد.

---

[1] Simple random sampling
[18] Equal probability of selection

4-4-2-2: نمونه‌گیری سیستماتیک [19]

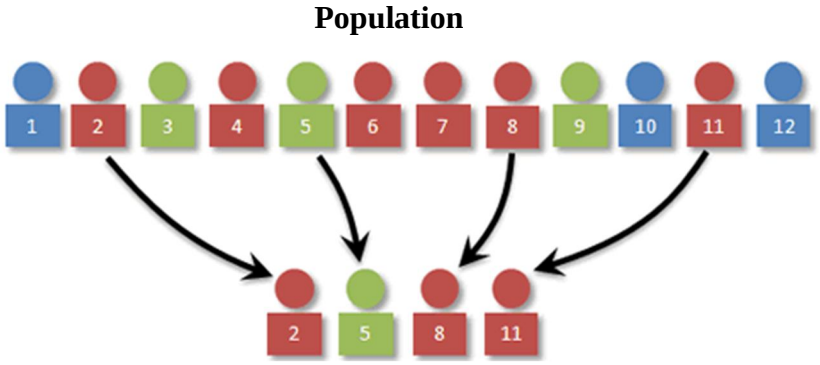
نمونه‌گیری سیستماتیک روی مرتب کردن جمعیت مبدا براساس برخی طرح‌های مرتب‌سازی و سپس انتخاب عناصر در فواصل منظم از لیست مرتب‌شده، تکیه می‌کند. نمونه‌گیری سیستماتیک شامل یک شروع تصادفی و سپس عملکرد با انتخاب هر عنصر Kام از آن نقطه به سمت جلو می‌باشد. در مورد k (اندازه جمعیت/ اندازه نمونه) مهم است که نقطه شروع به صورت خودکار اولین نقطه لیست نباشد، اما در عوض به صورت تصادفی از بین اولین تا kامین نقطه لیست انتخاب شده باشد. یک نمونه ساده، انتخاب هر دهمین نام از راهنمای تلفن (هر دهمین تلفن، همچنین نمونه‌گیری با جهش 10تایی) می‌باشد. تا زمانیکه نقطه شروع تصادفی است، نمونه‌گیری سیستماتیک نوعی نمونه‌گیری احتمالی است.

نمونه‌گیری سیستماتیک یک روش EPS است، زیرا همه عناصر شانس یکسانی برای انتخاب دارند. این روش نمونه‌گیری تصادفی ساده نیست زیرا زیرمجموعه‌های متفاوتی از همان اندازه احتمال انتخاب متفاوتی دارند. مثلا احتمال انتخاب در مجموعه {4,14,24,...,994} یک دهم است اما احتمال انتخاب در مجموعه {4,13,24,34,...} صفر است.

هنگامی که واحدهای جامعه ترتیب خاصی در چهارچوب نمونه‌گیری داشته باشند، کارآیی این روش افزایش می‌یابد. روش نمونه‌گیری سیستماتیک به خصوص وقتی استخراج نمونه درباره جامعه‌ها و پدیده‌های طبیعی است با صرفه‌جویی اساسی در وقت و هزینه نمونه‌گیری همراه است. در کتابداری، نمونه‌گیری از خاک به منظور تعیین مقاومت خاک، هواشناسی، جنگلداری و نظایر این‌ها نمونه‌گیری سیستماتیک استفاده می‌شود.

---

[1] Systematic sampling



شکل4-6: نمونه‌گیری سیستماتیک با ازای هر سه نمونه

4-4-2-3: نمونه‌گیری طبقه‌بندی[20]

وقتی جمعیت تعدادی دسته متفاوت را در بر می‌گیرد، چهارچوب نمونه‌گیری می‌تواند بوسیله این دسته‌ها به طبقات مجزا سازماندهی شود. سپس هر طبقه به عنوان یک زیرجمعیت مستقل نمونه‌گیری می‌شود که از هر طبقه عناصر مجزا می‌توانند به صورت تصادفی نمونه‌گیری شوند. چندین مزیت برای نمونه‌گیری طبقه‌بندی وجود دارد. اول اینکه، تقسیم جمعیت به طبقات مستقل و متمایز می‌تواند محققان را قادر سازد استنتاج‌هایی درباره زیرگروه‌های خاص که ممکن است در یک نمونه تصادفی عمومی‌تر، از دست رفته باشند، انجام دهند.

دوم اینکه، بکار بردن یک روش نمونه‌گیری طبقه‌بندی تخمین‌های آماری کاراتری منجر شود. حتی اگر یک روش نمونه‌گیری طبقه‌بندی‌شده منجر به افزایش کارایی آماری نشود، چنین روشی کارایی کمتر از نمونه‌گیری تصادفی ساده نخواهد داشت به شرط اینکه هر طبقه متناسب با اندازه گروه در جمعیت باشد.

سومین مزیت این است که گاهی اوقات داده‌ها برای طبقات از قبل موجود مجزا، در یک جمعیت نسبت به کل جمعیت قابل دسترس‌تر هستند. در چنین مواردی، استفاده از یک روش

[20] Stratified sampling

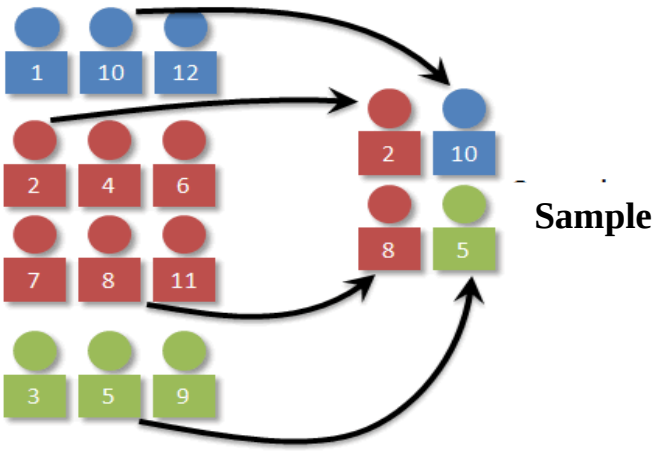
نمونه‌گیری طبقه‌ای راحت‌تر از تجمع داده‌ها داخل گروه‌ها است. در نهایت، چون هر طبقه به عنوان یک جمعیت مستقل تلقی می‌شود، روش‌های نمونه‌گیری متفاوت می‌تواند برای طبقات متفاوت بکار برده شود که محققان را قادر می‌سازد مناسبترین روش (مقرون به صرفه‌ترین روش) را برای هر زیرگروه شناسایی شده داخل جمعیت استفاده کنند.

برای استفاده از این روش، نخست جمعیت (چهارچوب نمونه‌گیری) بر مبنای یک یا چند ویژگی، مانند جنس و گروه‌های سنی، به زیر گروه‌ها یا دسته‌ها تقسیم می‌شود و سپس در هر دسته به طور مستقل نمونه‌گیری انجام می‌شود. بنابراین در این روش، جامعه‌ای با حجم N ابتدا به L زیر جامعه به حجم‌های $N_1, N_2, ..., N_L$ تقسیم می‌شود. این زیر جامعه‌ها متداخل نیستند و اجتماع آنها برابر با کل جامعه است، یعنی:

$$N_1 + N_2 + ... + N_L = N$$

هر زیر جامعه را یک طبقه می‌نامند. برای بهره‌گیری عمده از روش طبقه‌بندی باید مقدارهای $N_h \quad h = 1, ..., L$، را بدانیم. وقتی طبقات مشخص شدند، از هر طبقه نمونه‌ای انتخاب می‌شود. انتخاب‌ها در هر طبقه مستقل از طبقه دیگر صورت می‌گیرند.

اگر از هر طبقه، نمونه‌ای به روش تصادفی ساده گرفته شود، شیوه کلی نمونه‌گیری را نمونه‌گیری تصادفی طبقه‌بندی می‌نامند.



شکل4-7: نمونه‌گیری طبقه بندی تصادفی

تعداد نمونه‌ای که از هر طبقه انتخاب می‌شود، باید متناسب با تعداد نمونه باشد. مثلا در شکل بالا، یک چهارم جمعیت آبی هستند، بنابراین یک جهارم نمونه‌ها باید آبی باشد. در شکل بالا یک نمونه‌گیری تصادفی طبقه‌بندی شده انجام شده است که 4 نمونه دارد که شامل یک نمونه آبی، یک نمونه سبز و دو نمونه قرمز (متناسب با تعداد) می‌باشد.

4-4-2-4: نمونه‌گیری احتمالی متناسب با اندازه [21]

در برخی موارد طراح نمونه به یک متغیر کمکی یا مقیاس اندازه دسترسی دارد که گمان می‌کند با متغیر مورد علاقه برای هر عنصر در جمعیت، ارتباط دارد. این داده‌ها می‌توانند برای بهبود دقت در طراحی نمونه استفاده شوند. یک گزینه این است که از متغیر کمکی به عنوان پایه‌ای برای طبقه‌بندی استفاده شود (به همان صورت که در بخش قبلی بیان شد).

گزینه دیگر، نمونه‌گیری احتمالی متناسب با اندازه (PPS) است که در آن احتمال انتخاب برای هر عنصر متناسب با مقیاس اندازه آن است و حداکثر برابر 1 است. در طراحی یک PPS ساده، این احتمال‌های انتخابی میتوانند به عنوان پایه‌ای برای نمونه‌گیری پوآسون انتخاب شوند. به هر

[21] Probability proportional to size

حال این روش نقطه ضعف اندازه نمونه متغیر را دارد و بخش‌های متفاوت جمعیت، به خاطر شانس متفاوت در انتخاب ممکن است هنوز بیش از حد نمایش داده شده باشند یا کمتر نمایش داده شده باشند. برای حل این مساله، PPS ممکن است با یک روش سیستماتیک ترکیب شود.

مثال) تصور کنید ما 6 مدرسه با جمعیت 150، 180، 200، 220، 260 و 490 دانش آموز (کلا 1500 دانش آموز) داریم و میخواهیم جمعیت دانش آموزی را به عنوان پایه‌ای برای یک نمونه PPS با اندازه 3 استفاده کنیم. برای انجام این کار، به اولین مدرسه اعداد 1 تا 150، به دومین مدرسه اعداد 151 تا 330 (=150+180)، سومین مدرسه اعداد 331 تا 530 و همینطور به آخرین مدرسه 1011 تا 1500 را تخصیص می‌دهیم. سپس یک شروع تصادفی بین 1 و 500 را تولید می‌کنیم (برابر 1500/3) و در جمعیت مدرسه با ضرایبی از 500، روند انتخاب را ادامه می‌دهیم. اگر شروع تصادفی از 137 باشد، ما مدارسی را که اعداد 137، 637 و 1137 به آن تخصیص داده شده است، یعنی مدارس اول، چهارم و ششم را انتخاب می‌کنیم.

روش PPS می‌تواند دقت را برای اندازه نمونه مشخصی توسط تمرکز نمونه روی عناصر بزرگی که بیشترین تاثیر را روی تخمین‌های جمعیت دارند، بهبود ببخشد. نمونه‌گیری PPS عموما برای نظرسنجی از کسب و کار استفاده می‌شود، که در آن اندازه عنصر تا حد زیادی متفاوت است و اطلاعات کمکی اغلب در دسترس است. مثلا یک نظرسنجی که برای اندازه‌گیری تعداد مهمان شب در هتل‌ها جستجو می‌کند، ممکن است از شماره اتاق‌های هر هتل به عنوان یک متغیر کمکی استفاده کند. در برخی موارد، موقعی که برای تولید تخمین‌های رایج تر تلاش می‌کند، یک مقیاس قدیمی تر از متغیر مورد علاقه می‌تواند به عنوان یک متغیر کمکی استفاده شود.

4-4-2-5: نمونه‌گیری خوشه‌ای[22]

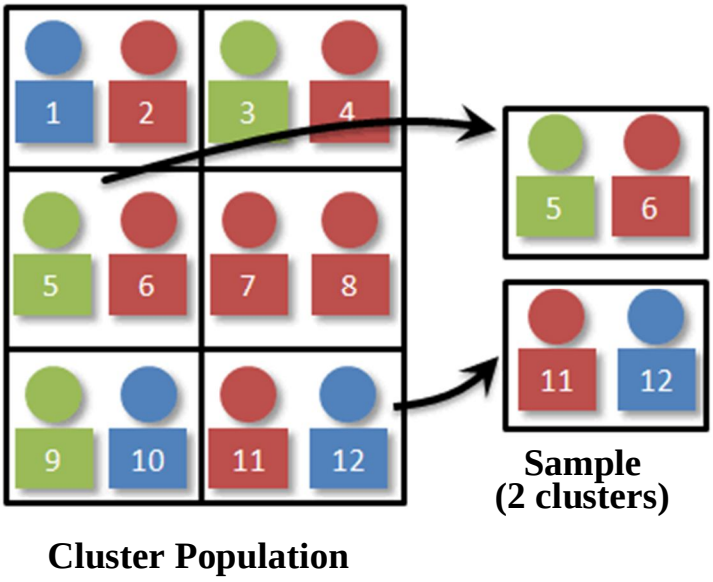
نمونه‌گیری خوشه‌ای شامل تشکیل گروه‌ها یا خوشه‌هائی مناسب از واحدهای نمونه‌گیری و سپس آمارگیری از تمام یا بخشی از واحدهای خوشه انتخاب شده می‌باشد.

[22] Cluster sampling

هنگامی از این نوع نمونه‌گیری استفاده می‌شود که جامعه مورد پژوهش از دسته‌های جداگانه‌ای تشکیل شود و عناصر آن جامعه در این دسته‌ها توزیع شده باشند. علاوه بر این اگر هزینه بدست آوردن چهارچوبی که نام همه عناصر جامعه را در برداشته باشد سنگین یا هزینه گردآوری مشاهدات و داده‌های پژوهش زیاد باشد، می‌توان از نمونه‌برداری خوشه‌ای استفاده کرد که از نمونه‌برداری ساده یا طبقه‌ای به مراتب آسان‌تر و ارزان‌تر خواهد بود. بنابراین منطق اساسی نمونه‌گیری خوشه‌ای در حقیقت، رعایت اصل اقتصاد و راحتی اجرای آن می‌باشد.

مزایای نمونه‌گیری خوشه‌ای از نظر هزینه آماری تا حد زیادی وابسته به این حقیقت است که گردآوری اطلاعات از واحدهای نزدیک به هم آسان‌تر، سریع‌تر، ارزان‌تر و بالاخره راحت‌تر از جمع‌آوری اطلاعات از واحدهائی می‌باشد که در تمام حوزه آمارگیری یک نمونه بررسی قرارگرفته اند. برای مثال بسیار ساده تر است که تمام کشاورزان واقع در یک روستا را آمارگیری کنیم تا همین تعداد کشاورز نمونه را بصورت تصادفی از بین تمام کشاورزان یک دهستان انتخاب کنیم.

به علت آسانی عملیات میدانی و کم شدن هزینه آمارگیری، نمونه‌گیری خوشه‌ای در بسیاری از آمارگیری‌ها به‌کار برده می‌شود و بطورکلی برای یک نمونه با حجم معین سودبخشی نمونه‌گیری خوشه‌ای در مقایسه با نمونه‌گیری پراکنده واحدهایی که ازجامعه بصورت واحد به واحد انتخاب می‌شود کمتر است. علت آن واریانس نمونه‌گیری است. زیرا در انتخاب اخیر امکان برگزیدن واحدها به طور جزئی از تمام قسمت جامعه وجود دارد. بهترین نمونه خوشه‌ای، نمونه‌ای است که واحدهای خوشه در بین خود تا سرحد امکان با یکدیگر متفاوت باشند. (یعنی واریانس داخل خوشه حداکثر باشد). درعمل منظور از نمونه‌گیری خوشه‌ای آن است که از واحدهای نزدیک به هم جامعه و یا واحدهائی که بتوان آنها را به‌راحتی با یکدیگر نمونه‌گیری نمود، خوشه‌هائی تشکیل داد و از بین خوشه‌های تشکیل شده نمونه‌ای انتخاب کرد.



شکل4-8: نمونه‌گیری خوشه ای

4-4-2-6: نمونه‌گیری سهمیه‌ای[23]

در نمونه‌گیری سهمیه‌ای بر طبق یک سهمیه ثابت، واحدها به‌صورت غیرتصادفی انتخاب می‌شوند. دو نوع نمونه‌گیری سهمیه‌ای وجود دارد: نسبتی و غیرنسبتی.

در نمونه‌گیری سهمیه‌ای نسبتی، برای نشان دادن مشخصه‌های اصلی جمعیت، نسبتی از هر یک را نمونه‌گیری می‌کنیم. برای مثال، اگر شما بدانید که 40 % جمعیت از زنان و 60% آن از مردان تشکیل شده و بخواهید اندازه نمونه 100 داشته باشید، به نمونه‌گیری ادامه می‌دهید تا وقتی که این نسبت‌ها را در نمونه خود به‌دست آورید. بنابراین اگر 40 زن در نمونه وجود داشته باشند، اما نمونه هنوز شامل 60 مرد نباشد، شما به نمونه‌گیری ادامه خواهید داد تا 60 مرد هم در نمونه قرار گیرند. اما حتی اگر بعد از چهل زن، زن یا زنان دیگری شرط شامل شدن در نمونه را دارا بودند، نباید آنها را وارد نمونه کنید زیرا قبلا سهمیه‌شان پر شده است. بدین ترتیب

[23] Quota sampling

نمونه‌ای از زنان و مردان با توجه به سهمی که در جمعیت دارند، به‌دست می‌آید. مشکلی که اینجا وجود دارد (همچنین در بسیاری از نمونه‌گیری‌های هدفمند) این است که بایستی در مورد مشخصه‌های خاص که بر اساس آنها سهمیه دارید، تصمیم بگیرید. آیا این مشخصه‌ها سن، جنس، تحصیلات، نژاد و موقعیت و... هستند؟

نمونه‌گیری سهمیه‌ای غیرنسبتی محدودیت‌های کمتری دارد. در این روش کمترین مقدار واحدهای نمونه که می‌خواهید در هر گروه باشد را مشخص می‌کنید. در این حالت شما نگران نیستید که مقدار نمونه مطابق با نسبت در جامعه باشد. در عوض شما می‌خواهید حداقل، از هر زیرگروه کوچک، تعدادی را در نمونه داشته باشید که این تعداد دقیقا مشخص نمی‌شود. نمونه‌گیری سهمیه‌ای مشابه نمونه‌گیری طبقه‌بندی در روش احتمالی است.

4-4-2-7: نمونه‌گیری اتفاقی [24]

نمونه‌گیری اتفاقی نوعی نمونه‌گیری غیراحتمالی است که در آن انتخاب نمونه‌ها از مجموعه‌ای از واحدهای جامعه که به آسانی دست یافتنی (قابل دسترس) باشند صورت می‌گیرد. یعنی یک جمعیت به دلیل اینکه در دسترس و راحت است انتخاب می‌شود.

محقق چنین نمونه‌ای را نمی تواند به طور علمی به کل جمعیت عمومیت دهد زیرا به قدر کافی قابل ارائه نیست. مثلا اگر مصاحبه‌کننده صبح زود در یک روز خاص برای انجام چنین بررسی در یک مرکز خرید باشد، افرادی که او با آنها مصاحبه می‌کند به آنهایی که در آن زمان خاص هستند محدود می‌شود که دیدگاه اعضای دیگر جامعه را در آن ناحیه ارائه نمی دهد. این نوع نمونه‌گیری اغلب برای آزمایش مقدماتی مفید است.

انتخاب یک نمونه از واحدهایی که قابل دسترس هستند اغلب ساده به نظر می‌رسند. اما این سادگی می‌تواند ما را به اشتباه بیندازد. بطور مثال، فرض کنید یک کارخانه آب میوه‌گیری، محصول سیب باغی را که در صندوق‌هایی چیده شده، می‌خرد. اگر بازرس کارخانه، علاقه‌مند به بررسی کیفیت این سیب‌ها باشد، برای راحتی، به نظر می‌رسد که نمونه‌ای از سیب‌هایی که

[24] Accidental sampling

در سطح جعبه‌ها قرار گرفته‌اند، را انتخاب کند، اما این سیب‌ها نمی‌توانند "نماینده" تمام سیب‌های داخل جعبه باشند. مثلا سیبهای ته جعبه بیشتر از سیب‌های دیگر محموله، آسیب می‌بینند. این روش ممکن است بازرس را گمراه کند که همه سیب‌های محموله سالم هستند و یا حتی تمام سیب‌های فاسد در ته جعبه‌ها چیده شده‌اند. مثال دیگری از این نوع نمونه‌گیری، بررسی در مورد افرادی که به نوعی بیماری خاص مبتلا هستند، می‌باشد. برای داشتن نمونه‌ای از آنها به بیمارستان یا کلینیکی که مخصوص آنهاست مراجعه و از آنها نمونه‌گیری می‌شود. بدیهی است این نمونه نمی‌تواند "نماینده" همه افراد مبتلا به این بیماری خاص در جامعه باشد.

4-4-2-8: نمونه‌گیری خط مقطعی [25]

یک روش نمونه‌گیری عناصر در یک ناحیه است که در آن اگر یک پاره خط انتخاب شده به نام "transect " عنصر را قطع کند، آن عنصر نمونه برداری شده است.

4-4-2-9: نمونه‌گیری پانلی [26]

در این روش ابتدا، یک گروه از شرکت‌کنندگان از طریق یک روش نمونه‌گیری تصادفی انتخاب می‌شوند و سپس از همان گروه برای همان اطلاعات، چندین بار در طی یک دوره زمانی سوال می‌شود. بنابراین، به هر شرکت‌کننده، همان بررسی طی دو یا چند نقطه زمانی داده می‌شود. هر دوره جمع‌آوری داده یک "موج" نامیده می‌شود.

---

[25] Line-intercept sampling
[26] Panel sampling

4-5: کیفیت داده

واقع بینانه نیست که انتظار داشته باشیم داده جمع‌آوری شده برای پایگاه‌داده کامل باشد. ممکن است مسائلی به علت خطای انسانی، محدودیت‌های دستگاه‌های اندازه‌گیری یا کمبودهایی در فرآیند جمع‌آوری داده وجود داشته باشد. مقادیری از داده یا کل آن ممکن است اشتباه باشد. در موارد دیگری ممکن است داده‌های اضافی یا ناخواسته وجود داشته باشد، یعنی چندین داده که همگی به یک شئ خاص اشاره دارند. مثلا ممکن است دو رکورد متفاوت برای یک شخص وجود داشته باشد که اخیرا در دو آدرس متفاوت زندگی کرده است. حتی اگر همه داده‌ها در دیتابیس حاضر هستند و خوب به نظر می‌رسند، ممکن است ناسازگاری‌هایی وجود داشته باشد، مثلا شخصی که قدش دو متر است، اما وزنش دو کیلوگرم است. در این بخش، مباحث مرتبط با کیفیت داده‌های دیتابیس را مورد بررسی قرار می‌دهیم.

4-5-1: خطاهای اندازه‌گیری و جمع آوری داده

اصطلاح خطای اندازه‌گیری، به هر مشکلی که در نتیجه فرآیند اندازه‌گیری حاصل شود، گفته می‌شود. یک مساله معمول این است که مقدار ثبت شده تا حدودی با مقدار واقعی فرق داشته باشد. برای صفات پیوسته، تفاوت عددی اندازه‌گیری شده و مقدار واقعی، خطا نامیده می‌شود.

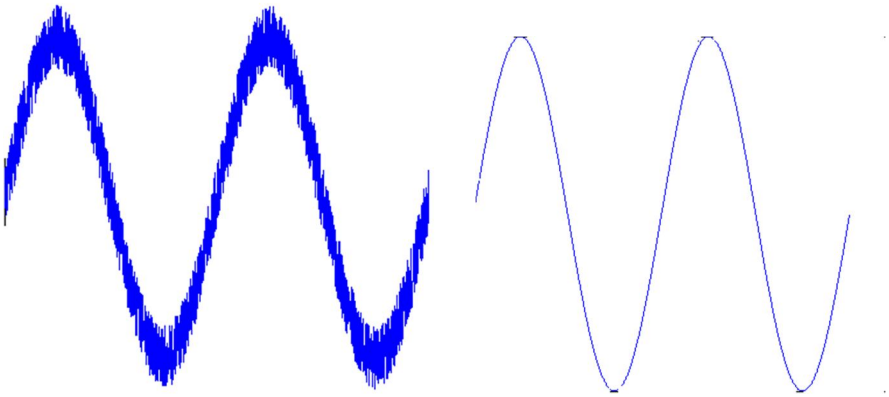
اصطلاح خطای جمع‌آوری داده، به خطاهایی مانند حذف اشیاء داده یا مقادیر صفات آن و یا شامل شدن یک شئ داده به طور غیرمقتضی، اشاره دارد. مثلا در مطالعه حیوانات یک گونه خاص، ممکن است حیوانات گونه‌های خویشاوندی که ظاهری شبیه به گونه مورد نظر دارند نیز شامل شود.

خطای اندازه‌گیری و جمع‌آوری داده می‌توانند به طور سیستماتیک یا تصادفی باشند. انواع خاصی از خطاهای داده در حوزه‌های خاص وجود دارد که معمول هستند و اغلب، تکنیک‌های خوبی برای کشف و تصحیح این خطاها وجود دارد. مثلا خطاهای مربوط به صفحه کلید وقتی

داده را به صورت دستی وارد می‌کنیم، خطاهایی معمول هستند و در بسیاری از برنامه‌های ثبت داده، تکنیک هایی برای کشف این نوع خطاها و تصحیح آنها وجود دارد.

4-5-2: نویز داده

نویز، جزء تصادفی خطای اندازه‌گیری است که ممکن است شامل اعوجاج یک مقدار یا ضمیمه داده‌های نادرست باشد. شکل4-9 یک سری زمانی، قبل و بعد از اینکه با نویزهای تصادفی آمیخته شود را نشان می‌دهد.



شکل4-9: نویز در سری‌های زمانی الف) سری زمانی ب) سری زمانی نویزدار

اصطلاح نویز، اغلب در رابطه با داده‌ای که یک جزء زمانی یا فضایی دارد استفاده می‌شود. در چنین مواردی، تکنیک‌های پردازش تصویر یا سیگنال می‌توانند برای کاهش نویز استفاده شوند. حذف نویز کار سختی است و در روش‌های داده‌کاوی، توجه زیادی به ابداع الگوریتم‌های قوی که نتایج قابل قبولی را حتی با وجود نویز تولید کند، شده است.

4-5-3: دقت، بایاس و درستی

در آمار و علوم تجربی، کیفیت فرآیند اندازه‌گیری و داده‌های بدست آمده، به وسیله دقت و بایاس اندازه‌گیری می‌شود. در این بخش، تعاریف دقت و بایاس را بیان می‌کنیم. برای تعریف آنها، فرض می‌کنیم که اندازه‌گیری یک کمیت را به صورت مکرر انجام دادیم و این مجموعه

مقادیر را برای محاسبه میانگین استفاده کردیم که به عنوان تخمینی از مقدار واقعی به کار می‌رود.

دقت: نزدیکی اندازه‌گیری‌های مکرر کمیت به یکدیگر

بایاس: تغییرات سیستماتیک اندازه‌گیری کمیت مورد نظر

دقت، اغلب توسط انحراف استاندارد مجموعه‌ای از مقادیر اندازه‌گیری می‌شود. درحالی‌که بایاس، با در نظر گرفتن اختلاف بین میانگین مجموعه‌ای از مقادیر و مقدار شناخته شده‌ای از کمیت، محاسبه می‌شود. به عنوان مثال تصور کنید میخواهیم دقت و بایاس را، برای یک وزنه یک کیلویی با یک مقیاس آزمایشگاهی جدید ارزیابی کنیم. وزنه را پنج بار وزن می‌کنیم و پنج مقدار به صورت زیر بدست می‌آید:

{1.015, 0.990, 1.013, 1.001, 0.986}

میانگین این مقادیر برابر 1,001 است و بنابراین بایاس برابر 0,001 است و دقت، همانطور که با انحراف استاندارد اندازه‌گیری شد، برابر 0,013 است.

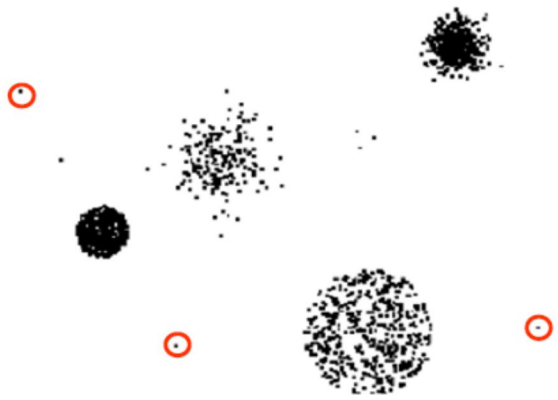
درستی: به درجه خطای اندازه‌گیری داده اشاره می‌کند. به عبارت دیگر، درستی به نزدیکی اندازه‌گیری‌ها به مقدار واقعی کمیت مورد نظر اشاره دارد. درستی به دقت و بایاس نیز بستگی دارد اما یک مفهوم عمومی است و هیچ فرمول خاصی برای درستی بر حسب این دو کمیت وجود ندارد.

4-5-4: مقادیر دور افتاده [27]

مقادیر دور افتاده اشیاء داده‌ای هستند که دارای ویژگی‌هایی می‌باشند که با اغلب اشیاء داده‌ای دیگر در پایگاه‌داده فرق دارند یا مقادیری از یک صفت که با توجه به مقادیر معمول آن صفت، غیرمعمول هستند و در واقع بحث در مورد اشیاء یا مقادیر غیرعادی است. مهم است که بین مفاهیم نویز و مقادیر دور افتاده تمایز قائل شویم. مقادیر دور افتاده، ممکن است اشیاء یا

[27] Outliers

مقادیر داده مشروع باشند و برخلاف نویز، در برخی موارد مورد توجه هستند. به عنوان مثال، در تشخیص مزاحمت‌های شبکه‌ای و کلاه برداری‌ها، هدف پیدا کردن اشیاء یا رویدادهای غیرعادی در بین تعداد زیادی از موارد نرمال است.



شکل4-10: مقادیر دور افتاده

4-5-5: مقادیر از دست رفته

برای یک شئ، معمول نیست که یک یا تعدادی از مقادیر صفات خود را از دست داده باشند. در برخی موارد، اطلاعات مورد نظر جمع‌آوری نشده‌اند، مثلا برخی افراد از دادن اطلاعات در مورد سن یا وزنشان امتناع می‌کنند. در موارد دیگری، برخی صفات به همه اشیاء قابل اطلاق نیستند. مثلا اغلب فرم‌ها دارای بخش‌های مشروطی هستند که تنها وقتی شخص سوال قبلی را در شرایط خاص جواب می‌دهد، پر می‌شوند، اما برای سادگی همه فیلدها ذخیره می‌شوند. با این وجود، مقادیر از دست رفته، باید در طی آنالیز داده در نظر گرفته شوند. برای مقابله با این داده‌های از دست رفته، استراتژی‌هایی وجود دارد که هر کدام ممکن است در شرایط خاص مناسب باشد. مثلا یک روش، حذف مقادیر از دست‌رفته می‌باشد، یا در مواردی این داده‌ها می‌توانند تخمین زده شوند.

6-5-4: مقادیر متناقض

داده ممکن است در برخی موارد، دارای مقادیر متناقضی باشد. مثلا یک فیلد آدرس را در نظر بگیرید که در آن کد پستی و شهر لیست شده‌اند، اما کد پستی مشخص شده در آن شهر وجود ندارد. ممکن است افراد موقع وارد کردن این اطلاعات، دو رقم را جابجا کنند، یا شاید یک رقم، موقعی که اطلاعات از یک فرم دست نویس اسکن شده است، اشتباه خوانده شده باشد.

صرف نظر از علت مقادیر متناقض، تشخیص آنها و اینکه اگر ممکن است چنین مسائلی را تصحیح کنیم، دارای اهمیت است. تشخیص بعضی از انواع تناقض‌ها ساده است، مثلا قد شخص نباید منفی شود. در سایر موارد، ممکن است نیاز به مراجعه به یک منبع خارجی اطلاعات باشد. مثلا وقتی یک شرکت بیمه درخواست‌های بازپرداخت را بررسی می‌کند، نام و آدرس روی فرم‌های بازپرداخت را با دیتابیس مشتری‌هایش چک می‌کند.

وقتی یک تناقض آشکار می‌شود در بعضی از موارد، تصحیح داده ممکن است. مثلا امکان چک کردن مجدد کد محصول در برابر لیستی از کدهای محصول شناخته شده، وجود دارد و بنابراین اگر کد اشتباه بوده ولی به کد شناخته‌شده نزدیک باشد، می‌توان آن را اصلاح کرد. تصحیح یک تناقض، نیاز به اطلاعات افزونه یا اضافی دارد.

7-5-4: داده‌های تکراری

یک پایگاه‌داده ممکن است شامل اشیاء داده‌ای باشد که تکراری یا تقریبا تکراری هستند. بسیاری از افراد نامه‌های پستی تکراری دریافت می‌کنند، زیرا آنها چندین بار در دیتابیس با نام‌های کمی متفاوت ظاهر می‌شوند. برای شناسایی و حذف چنین تکرارهایی، باید دو بحث مهم را مورد توجه قرار دهیم. اول اینکه اگر دوشئ وجود دارند که واقعا نشان‌دهنده یک شئ هستند، ممکن است مقادیر صفات مربوطه متفاوت باشد و باید امکان رفع این مقادیر متناقض وجود داشته باشد. دوم، باید مراقب باشیم تا از ترکیب داده‌هایی که شبیه هستند اما تکراری نیستند، اجتناب کنیم، مانند دو فرد متفاوت با نام‌های یکسان.

اصطلاح حذف تکرار[28]، اغلب برای اشاره به فرآیندهایی که با این مسائل سر و کار دارند استفاده می‌شود. در برخی موارد، دو یا چند شئ یکسان با توجه به صفات اندازه‌گیری شده در دیتابیس، یکسان هستند اما آنها هنوز اشیاء متفاوتی را نشان می‌دهند. در اینجا تکرارها قانونی هستند اما اگر امکان اختصاص اشیاء یکسان در طراحی آنها وجود نداشته باشد، هنوز ممکن است باعث بروز مسائلی برای برخی از الگوریتم‌ها شوند.

4-6: ساخت پایگاه‌داده

4-6-1: نمونه چیست؟

ورودی برای یادگیری ماشین مجموعه‌ای از نمونه‌ها است. این نمونه‌ها اشیائی هستند که باید طبقه‌بندی، به هم پیوسته و یا خوشه‌بندی شوند. تا کنون به این موارد Example می‌گفتیم، اما از این به بعد اصطلاح اختصاصی‌تر instance را برای ورودی‌ها به کار می‌بریم. هر instance یک Example مستقل منحصر به فرد از مفهومی که باید یادگیری شود می‌باشد. به علاوه، هریک به وسیله مقادیر مجموعه‌ای از ویژگی‌های از پیش تعریف شده مشخص می‌شود. هر پایگاه‌داده، به صورت ماتریسی از نمونه‌هایی که دارای ویژگی‌هایی (ستونهایی) هستند، نمایش داده می‌شود.

4-6-2: ویژگی چیست؟

هر نمونه مستقل منحصر به فرد، داده ورودی را به وسیله مقادیر ویژگی‌ها یا خصوصیات از پیش تعریف شده ثابتی برای یادگیری ماشین فراهم می‌کند. نمونه‌ها همان سطرهای جدول پایگاه‌داده هستند و خصوصیات آنها ستون‌ها هستند. استفاده از یک مجموعه ثابت از ویژگی‌ها، محدودیت دیگری را بر روی انواع مسائلی که به طورکلی در حوزه پردازش داده‌ها در کاربردهای عملی در نظر گرفته می‌شود، تحمیل می‌کند.

---

[28] Deduplication

چه اتفاقی می‌افتد اگر نمونه‌های متفاوت خصوصیات متفاوت داشته باشند؟ مثلا اگر نمونه‌ها وسیله نقلیه باشند، تعداد چرخ‌ها ویژگی هست که برای اکثر وسایل نقلیه به‌کار برده می‌شود اما برای کشتی نمی توان آن را بیان کرد. درحالی‌که تعداد دکل‌ها ویژگی است که برای کشتی‌ها استفاده می‌شود اما برای وسایل نقلیه زمینی به‌کار نمی رود. راه حل استاندارد این هست که هر خصوصیت ممکن را ایجاد کنیم و نشانه خاصی به صورت "مقدار نامربوط" را برای نشان دادن اینکه یک ویژگی خاص برای یک مورد خاص در دسترس نیست، استفاده کنیم. وضعیت مشابه، وقتی مطرح می‌شود که وجود یک ویژگی (مثلا نام همسر) بستگی به ارزش ویژگی دیگر (متاهل یا مجرد) دارد. ارزش یک ویژگی برای یک نمونه خاص، اندازه‌گیری مقداری است که به آن صفت اشاره دارد.

3-6-4: پیش پردازش

بعد از اینکه نمونه‌گیری برای پایگاه‌داده انجام شد، ملاحظات و پیش پردازش‌هایی باید بر روی داده‌های مورد نظر انجام شود. الگوریتم‌های مورد نظر برای تشخیص الگو تنها الگوهایی را که واقعا در داده حاضر هستند تشخیص می‌دهند. داده هدف، باید به قدر کافی بزرگ باشد تا این الگوها را در برداشته باشد و به قدر کافی مختصر باشد تا در محدوده زمان قابل قبولی بتوانیم آنها را پردازش کنیم. یک منبع معمول برای داده‌ها، انبار داده می‌باشد. پیش پردازش، برای آنالیز پایگاه‌داده‌های چند متغیره، قبل از به‌کارگیری الگوریتم‌های پردازش ورودی، ضروری است. سپس مجموعه هدف پاکسازی می‌شود.

در یک کاربرد تجاری، لازم است تا منابع را از بخش‌های مختلف گردآوری کنیم. مثلا، برای داده‌های مطالعه بازاریابی نیاز است تا داده‌ها را از بخش فروش، بخش صورت حساب مشتری و بخش خدمات مشتری جمع‌آوری کنیم. جمع‌آوری داده از منابع مختلف معمولا چالش‌هایی را به وجود می‌آورد. بخش‌های مختلف، روش‌های متفاوتی را برای نگه‌داری رکوردها استفاده می‌کنند. روش‌های متفاوت، دوره‌های زمانی متفاوت، درجات متفاوتی برای جمع‌آوری داده‌ها، کلیدهای اصلی متفاوت و انواع متفاوتی از خطاها دارند. داده باید جمع‌آوری، ترکیب و

پاک‌سازی شود. پاک‌سازی داده، همچنین مشاهداتی را که دارای نویز یا مقادیر از دست‌رفته هستند حذف می‌کند. در ادامه این بخش، به بررسی موارد ذکر شده می‌پردازیم.

4-6-4: پاک‌سازی داده

پاک‌سازی داده فرایند تشخیص، اصلاح و حذف خطاهای موجود در داده‌هاست. خطاهای داده شامل داده‌های غلط، ناقص، تکراری، متناقض و یا با ساختار نامناسب هستند. برای بیان این تعریف، از عبارات تمیز کردن داده یا پالایش داده هم استفاده می‌شود. ابزارهای پیچیده‌ای با استفاده از الگوریتم‌ها، قوانین و جداول جستجو، پاک‌سازی داده را انجام می‌دهند. این ابزارها قابلیت اصلاح خودکار برخی خطاها مانند پیدا کردن و حذف داده تکراری را دارند.

هدف اصلی پاک‌سازی داده‌ها، از بین بردن ناسازگاری، مقادیر نادرست و سایر کمبودهای یکپارچگی داده‌ها از بانک‌های اطلاعاتی موجود است. علاوه برآن، باعث ایجاد سازگاری بین مجموعه‌های مختلف داده که با یکدیگر ادغام شده‌اند، می‌شود که این مسأله می‌تواند هدف دیگری برای پاک‌سازی داده باشد. عمل پاک‌سازی، تبدیل داده‌های موجود در سیستم‌های فعلی را به فرمت و ساختار مورد نیاز برای سیستم‌های جدید تسهیل می‌کند. بدیهی است، افرادی که درگیر پروژه پاک‌سازی داده می‌شوند، لازم است تا با ساختار داده‌های موجود و همچنین ساختار داده‌ها در سیستم هدف آشنایی کافی داشته باشند. در نتیجه این امکان برای متخصصین سیستم‌های موجود فراهم می‌شود تا خود را با پایگاه‌داده هدف و ابزارهای توسعه و عملکرد این سیستم آشنا سازند. از سوی دیگر توسعه‌دهندگان سیستم هدف، با ساختار و محتوی سیستم‌های موجود در سازمان آشنا می‌شوند.

4-6-4-1: جایگاه پاکسازی داده در فرآیند کشف دانش

کشف دانش در پایگاه‌داده فرایند شناسایی درست، ساده، مفید، و نهایتا الگوها و مدل‌های قابل فهم در داده‌ها می‌باشد. کشف دانش دارای مراحل تکراری زیر است که پاک‌سازی داده، مرحله اول از فرایند کشف دانش است:

- پاک‌سازی داده‌ها (از بین بردن ناسازگاری داده‌ها)

- یکپارچه‌سازی داده‌ها (ترکیب چندین منبع داده)
- انتخاب داده‌ها (بازیابی داده‌های مرتبط با تحلیل از پایگاه‌داده)
- تبدیل داده‌ها (تبدیل داده‌ها به فرم مناسب برای پردازش مثل خلاصه‌سازی و همسان‌سازی)
- داده کاوی (استفاده از روال‌های هوشمند برای استخراج الگوها از داده ها)
- ارزیابی الگو (مشخص‌کردن الگوهای صحیح و مورد نظر براساس معیارهای اندازه‌گیری)
- ارائه دانش (استفاده از تکنیک‌های بازنمایی دانش برای ارائه دانش کشف‌شده به کاربر)

پاک‌سازی داده، زمانی مورد نیاز است که مشکلات داده‌ای در سیستم رخ دهد. استراتژی‌های جلوگیری از خطا می‌تواند بسیاری از مشکلات داده‌ای را کاهش دهد، ولی نمی تواند آنها را از بین ببرد. روش‌های مختلفی در فرآیند پاک‌سازی داده مورد استفاده قرار می‌گیرد. در اینجا به مهم‌ترین آنها اشاره می‌شود:

الف) تجزیه

در فرآیند پاک‌سازی داده از تجزیه برای شناسایی خطاهای نحوی استفاده می‌شود. یک تجزیه‌کننده گرامر G، برنامه‌ای است که تشخیص می‌دهد آیا رشته ورودی، متعلق به زبان تعریف شده توسط گرامر هست یا خیر. در زبان‌های برنامه‌سازی، رشته یک برنامه است و در پاک‌سازی داده، رشته‌ها می‌توانند رکورد و یا مقادیر داده‌ای باشند. رشته‌هایی که خطای نحوی دارند، باید اصلاح شوند. تعداد خطاهای نحوی موجود در داده بستگی به محدودیت‌های محیطی دارد که داده در آن ذخیره شده است. اگر داده در فایل بدون ساختار ذخیره شده باشد دارای خطاهای نحوی و دامنه بیشتری خواهد بود.

ب) تبدیل داده

تبدیل داده، به عمل نگاشت داده از یک فرمت به فرمت مورد نظر با استفاده از یک نرم افزار گفته می‌شود. این تبدیل هم الگوی رکورد و هم دامنه مقادیر را تغییر می‌دهد. به این ترتیب که ابتدا داده‌های چند منبع به یک الگوی مشترک که نیازها را به نحو مطلوبی برآورده می‌سازد تبدیل می‌شوند. سپس اصلاح مقادیر در صورتی انجام می‌شود که داده‌های ورودی با الگوی

مشترک مطابقت نداشته باشند و این عدم تطابق باعث شکست تبدیل داده شود. استانداردسازی و نرمال‌سازی، تبدیل‌هایی هستند که در سطح نمونه برای از بین‌بردن ناهنجاری‌ها استفاده می‌شوند.

ج) اعمال محدودیت‌های جامعیت

این روش، محدویت‌های جامعیت را پس از انجام تراکنش‌های تغییر داده (شامل حذف، اضافه و به‌روز رسانی) تأمین و تضمین می‌کند. دو رویکرد مختلف برای این روش، کنترل محدودیت جامعیت و حفظ محدودیت جامعیت است. در روش اول، تراکنش‌هایی که نقض کننده جامعیت هستند، برگشت داده می‌شوند و در روش دوم، تراکنش‌های به‌روز رسانی و اعمال تغییرات در داده‌های اصلی شناسایی می‌شوند تا داده‌ها پس از تغییر، هیچ یک از محدودیت‌های جامعیت را نقض نکنند.

د) از بین بردن داده‌های تکراری

روش‌های متفاوتی برای حذف داده‌های تکراری وجود دارد که در همه این روش‌ها، باید الگوریتمی وجود داشته باشد که تشخیص دهد دو یا چند رکورد، نمایش‌های تکراری از یک موجودیت می‌باشند.

برای یک تشخیص کارا، هر رکورد باید با همه رکوردهای دیگر مقایسه شود. به عنوان مثال با استفاده از روش همسایگی مرتب شده، می‌توان تعداد مقایسه‌ها را به حداقل رساند. به این ترتیب که رکوردها بر اساس کلیدی مرتب می‌شوند که رکوردهای تکراری نزدیک هم قرار گیرند. سپس رکوردهایی که در یک پنجره کوچک شناور قرار دارند، با یکدیگر مقایسه می‌شوند. تشخیص رکوردهای تکراری بر اساس قوانین موجود در دانش حوزه مورد مطالعه است.

ه) روشهای آماری

در کاربرد، روش‌های آماری در پاک‌سازی داده ارائه شده است. این روش‌ها هم برای بررسی داده و هم برای اصلاح ناهنجاری داده مورد استفاده قرار می‌گیرند. تشخیص و از بین بردن خطاهای پیچیده با استفاده از کنترل و اعمال محدودیت‌های جامعیت امکان پذیر نیست. با

تحلیل داده‌ها بر اساس مقادیر میانگین، انحراف معیار و الگوریتم‌های خوشه‌بندی، اشخاص خبره ممکن است مقادیر داده‌های پیش بینی نشده‌ای را پیدا کنند که نشان‌دهنده رکوردهای نامعتبر است. معمولا اصلاح این خطاها غیرممکن است، زیرا صفات جدول دارای مقادیر داده‌ای صحیح هستند. یک راه حل ممکن با استفاده از روش‌های آماری، قراردادن مقادیر آماری از قبیل میانگین در این صفات است. ناهنجاری دیگری که به وسیله روش‌های آماری اصلاح می‌شود، مقادیر خالی است که با داده‌های قابل قبول جایگزین می‌شود. تولید این داده‌ها، نیازمند الگوریتم‌های تولید داده وسیع است.

چالش‌ها و مشکلات موجود در حوزه پاک‌سازی داده‌ها را می‌توان به چهار دسته طبقه‌بندی نمود که در زیر به آن اشاره می‌شود:

- **تشخیص خطا و رفع مشکل**

در بیشتر موارد، اطلاعات و دانش کافی برای تعریف اصلاحات صحیح رکوردها وجود ندارد. در نتیجه حذف رکورد به عنوان تنها راه حل عملی انجام می‌شود. اگر آن رکورد نامعتبر نباشد این حذف منجر به فقدان اطلاعات در سیستم خواهد شد. برای جلوگیری از حذف این گونه اطلاعات، باید روی رکوردها برچسب مقدار نامعتبر گذاشته شده ولی تا وقتی که اطلاعات مناسب برای تصحیح آنها به‌دست آید، در مجموعه داده حفظ شوند. در این صورت وظیفه مدیر سیستم، کنترل این مساله است که رکوردهای برچسب دار در تحلیل و پردازش مورد استفاده قرار نگیرند.

مسأله دیگر در اصلاح داده‌های نادرست، این است که در برخی مواقع یک مقدار صحیح قطعی و دقیق برای اصلاح داده غلط وجود ندارد، بلکه مجموعه‌ای از مقادیر صحیح ممکن وجود دارد. این مسأله به‌خصوص در ادغام رکوردهای تکراری که از چند منبع به دست آمده اند، ظاهر می‌شود. در نتیجه اصلاح خطا تا وقتی‌که یکی از گزینه‌ها به عنوان جواب صحیح انتخاب شود به تاخیر می‌افتد. نگهداری این گزینه‌های ممکن، بر مدیریت و پردازش داده تاثیر منفی می‌گذارد. در حقیقت، هر یک از گزینه‌های ممکن یک نسخه از مجموعه داده را ایجاد می‌کند، زیرا گزینه‌های ممکن دوبه دو ناسازگار هستند. مدیریت نسخه‌های مختلف مجموعه داده‌ها یک

چالش بزرگ است. به همین دلیل نگهداری سوابق پاک‌سازی داده تا حدی می‌تواند مشکل را حل کند.

- **نگهداری داده‌های پاکسازی شده**

پاک‌سازی داده یک فرآیند زمان‌بر و هزینه‌بر است. پس از اتمام فرآیند پاک‌سازی و به‌دست آوردن یک مجموعه داده بدون خطا، چنانچه مقادیر داده تغییر کند، پاک‌سازی مجدد کل مجموعه داده‌ها منطقی و مقرون به‌صرفه نیست، بلکه باید آن قسمتی از فرآیند پاک‌سازی دوباره انجام شود که داده‌های آن تغییر کرده است. تشخیص این مسأله که چه قسمت‌هایی تغییر کرده است با استفاده از تحلیل سوابق پاک‌سازی انجام می‌شود. در نتیجه، پس از تغییر برخی مقادیر در مجموعه داده بر اساس سوابق پاک‌سازی، رکوردهایی که تغییرات داشته‌اند، شناسایی شده و فرآیند پاک‌سازی برای آنها دوباره انجام می‌شود. برای مدیریت ردیابی سوابق پاک‌سازی باید از روش‌های کارایی استفاده شود. همچنین ممکن است جمع‌آوری اطلاعات اضافی در هنگام پاک‌سازی اولیه، این فرآیند را تسریع کند.

- **پاک‌سازی داده‌ها در محیط‌های یکپارچه مجازی**

مشکلاتی که در بخش‌های قبل مطرح شد، در محیط‌هایی که منابع داده به صورت مجازی با هم یکپارچه هستند، مضاعف می‌شود. در این محیط‌ها، اغلب به دلیل ماهیت مستقل منابع، امکان منتقل کردن اصلاحات انجام شده به منابع اصلی داده غیرممکن است. در نتیجه پاک‌سازی داده باید در هر بار دسترسی به داده انجام شود که این مساله به‌طور قابل ملاحظه‌ای، زمان پاسخ را کاهش می‌دهد. با داشتن متاداده مناسب و جمع‌آوری اطلاعات اضافی مانند سابقه پاک‌سازی و عملیات انجام‌شده در نرم‌افزارهای میانی برای پاک‌سازی داده، می‌توان کارایی را افزایش داد. همچنین اگر داده‌ها در فاصله بین دسترسی به منابع تغییر نکرده باشند، این روش می‌تواند از اجرای غیرضروری پاک‌سازی داده جلوگیری کند..[1]

4-6-5: تشخیص ناهنجاری

روش تشخیص ناهنجاری که به عنوان تشخیص موارد دورافتاده *نیز* شناخته شده است، به تشخیص الگوهای موجود در یک مجموعه اطلاعات داده شده، که با رفتار هنجار (نرمال) از پیش مقررشده، مطابقت ندارد، اشاره می کند. بنابراین الگوهای تشخیص داده شده، ناهنجاری نامیده می‌شوند و اغلب به اطلاعات حیاتی و کارآمد، در چندین حوزه کاربرد، ترجمه می‌شوند. همچنین ناهنجاری‌ها به عنوان دورافتادگی، تغییر، انحراف، تعجب، نابجایی، صفات عجیب و غیره ارجاع می‌شوند.

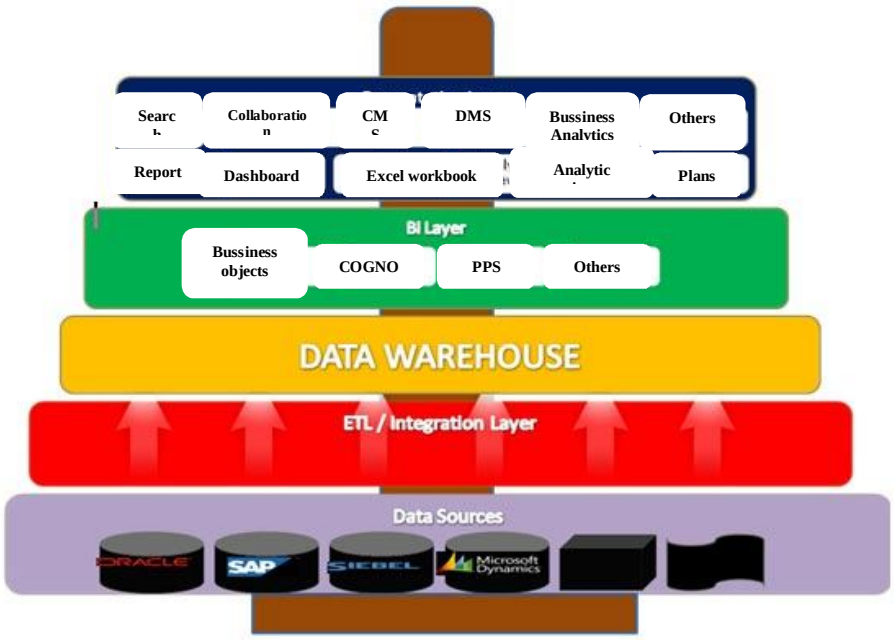
سه دسته گسترده از فنون تشخیص ناهنجاری وجود دارد. فنون تشخیص ناهنجاری بدون ناظر، (ناهنجاری‌ها را در یک مجموعه داده تست بدون برچسب، تحت این فرض که اکثریت موارد در مجموعه داده‌ها هنجار هستند و با جستجو به دنبال مواردی که حداقل تناسب را با بقیه مجموعه داده‌ها دارند، تشخیص می‌دهند). فنون تشخیص ناهنجاری باناظر،(نیاز به یک مجموعه داده‌ها دارد که با عنوان بهنجار و نابهنجار نشانه‌دار شده و شامل آموزش طبقه‌بندی‌شده (تفاوت کلیدی بسیاری از مسایل طبقه‌بندی آماری، ماهیت نامتعادل ذاتی، ناشی از تشخیص دورافتادگی است) باشد). فنون تشخیص ناهنجاری نیمه‌نظارتی، (که یک مدل که نشان‌دهنده رفتار طبیعی با توجه به یک مجموعه داده است، می‌سازند و سپس احتمال یک مورد تست تولیدی به وسیله مدل آموخته شده را می‌سنجند).

4-6-6: انبار داده [29]

انبار داده یک بانک اطلاعاتی بزرگ می‌باشد که شامل مجـموعـه‌ای از داده‌ها است که از منابع مختلف اطلاعاتی سازمان جمع‌آوری، دسته‌بندی و ذخیره می‌شود. در واقع یک انبار داده مخزن اصلی کلیه داده‌های حال و گذشته یک سازمان می‌باشد که برای همیشه جهت انجام عملیات گزارش‌گیری و آنالیز در دسترس مدیران می‌باشد.

---

[29] Data warehouse



شکل 4-11: ساختار انبار داده

انبار داده سه لایه دارد که به ترتیب بخش ورود[30] بخش یکپارچه‌سازی[31] و بخش نمایش[32] نام دارد. داده‌های خام از منابع اطلاعاتی مختلفی جمع‌آوری شده ودر لایه بخش ورود وارد می‌شوند. منبع داده خام می‌تواند یک سیستم ERP، پایگاه‌داده یک برنامه کاربردی و یا یک فایل اکسل باشد. ایجاد یکنواختی بین داده‌های وارد شده به انبار، در دومین لایه یعنی بخش یکپارچه‌سازی انجام می‌شود. به عنوان مثال حذف رکوردهای تکراری و یا نرمال‌سازی داده‌ها. در لایه بخش نمایش داده‌ها در دسترس کاربران قرار می‌گیرد. نرم‌افزارهای تهیه گزارش با دسترسی به این لایه می‌توانند اطلاعات مورد نیاز مدیران و تحلیلگران را استخراج و در قالب گزارش عرضه نمایند. در لایه بخش نمایش می‌تواند چندین داده‌گاه وجود داشته باشد.

[30] Staging
[31] Integration
[32] Presentation

داده‌گاه زیرمجموعه‌ای از داده‌های انبار است که در آخرین لایه یعنی بخش نمایش قرار دارد. یک داده‌گاه، مجموعه خاصی از اطلاعات را در خود نگه می‌دارد که برای گروهی از کاربران انبار داده مورد نیاز است. برای مثال اطلاعات فروش می‌تواند یک داده‌گاه را تشکیل دهد. انبار داده می‌تواند چندین داده‌گاه را در خود جای دهد. چندین داده‌گاه می‌توانند به صورت مستقل در لایه دسترسی، بخش نمایش، قرار داشته باشند. بدین ترتیب تغییر در یک داده‌گاه اثری بر روی داده‌گاه‌های دیگر نخواهد داشت.

دلیل اصلی ساخت انبار داده‌ها، بهبود کیفیت اطلاعات در سازمان است، در واقع داده‌های قابل دسترس از هر محل درون سازمان داده‌ها، از منابع داخلی و خارجی فراهم می‌شوند و به اشکال گوناگون از داده‌های ساخت‌یافته گرفته تا داده‌های ساخت نیافته مانند فایل‌های متنی یا چند رسانه‌ای، در مخزنی مجتمع می‌شوند. انبار داده‌ها یا DWH مخزنی از این داده‌هاست که به صورتی قابل درک در دسترس کاربران نهایی کسب و کار قرار می‌گیرد.

انبار داده متشکل از یک پایگاه‌داده و تعدادی جزء متصل با ویژگی‌های زیر می‌باشد:

- موضوع‌گرا :[33] پایگاه‌داده به گونه‌ای سازماندهی شده است که تمامی اطلاعاتی که به یک موضوع یا موجودیت خاص مربوط هستند با یکدیگر مرتبط می باشند.
- متغیر با زمان: تغییرات ایجاد شده در پایگاه‌داده اولیه در آن اعمال می‌شوند.
- ماندگار[34]: داده‌های اطلاعاتی هرگز حذف نشده، با داده‌های جدید جایگزین نمی‌شوند.
- یکپارچه: اطلاعات موجود در پایگاه‌داده از سراسر سازمان جمع‌آوری شده‌اند و با هم سازگاری دارند.

برخی از سازمان‌ها تمایل دارند انبارداده به صورت سراسری طراحی شود. به طوریکه تمامی اطلاعات موجود در سازمان در آن قرار گیرند. طراحی و استفاده از انبار داده به این صورت کاری پیچیده و زمان‌بر است. به همین علت در بسیاری از سازمان‌ها داده‌گاه استفاده می‌شود.

---

[33] Subject oriented

[34] Non-volatile

7-4: ذخیره سازی پایگاه‌داده

وقتی داده روی دیسک ذخیره می‌شود، دسترسی به داده دلخواه نسبت به زمانی که همان داده در حافظه اصلی قرار دارد، زمان بیشتری را نیاز دارد. وقتی پایگاه‌داده بزرگ باشد، داشتن الگوریتم‌هایی برای نگه‌داری داده در حافظه اصلی اهمیت پیدا می‌کند. وقتی که با داده‌هایی با اندازه بزرگ سر و کار داریم زمان انجام محاسبات، وقتی داده روی دیسک قرار دارد در مقایسه با زمانی که داده در حافظه اصلی قرار دارد قابل توجه است. ویژگی‌های فیزیکی دیسک نیز موضوع دیگری است که می‌تواند در این مورد بررسی شود. دیسک‌ها به صورت بلاک‌هایی سازماندهی شده‌اند که مینیمم واحدهایی می‌باشد که سیستم‌عامل برای جابه‌جایی داده بین حافظه اصلی و دیسک استفاده می‌کند. مثلا سیستم عامل ویندوز بلاک‌های 64 کیلو بایتی را استفاده می‌کند. دسترسی و خواندن یک بلاک دیسک در این حالت، حدودا ده میلیون ثانیه طول می‌کشد. این تاخیر حداقل پنج برابر کندتر از زمان صرف‌شده برای خواندن یک کلمه از حافظه اصلی است در حالی‌که همه آنچه ما می‌خواهیم دسترسی به چند بایت است. در حقیقت اگر ما بخواهیم کار ساده‌ای را با هر بلاک دیسک انجام دهیم، مثلا بلاک‌ها را به عنوان باکت هایی از یک جدول هش در نظر بگیریم و به دنبال مقدار خاصی از کلید در میان همه رکوردهای آن باکت جست‌وجو کنیم، بنابراین زمان برای جابه‌جایی بلاک از دیسک به حافظه اصلی بزرگتر از زمان انجام محاسبات است. با سازماندهی داده به طوری‌که داده‌های مرتبط روی یک سیلندر خاص (مجموعه‌ای از بلاک‌های در دسترس در شعاع یکسان از مرکز دیسک و در دسترس بدون حرکت هد دیسک) ذخیره شوند، ما می‌توانیم همه بلاک‌های روی سیلندر را، در زمانی که به طور قابل توجهی کمتر از ده میلیون ثانیه به ازای هر بلاک است، به حافظه اصلی ببریم. فارغ از این‌که داده چه‌طور سازماندهی شده‌است، شما می‌توانید تصور کنید که دیسک نمی تواند داده را بیشتر از صد میلیون بایت در ثانیه به حافظه اصلی انتقال دهد. وقتی اندازه پایگاه‌داده در حد مگابایت است این موضوع چندان مهم نیست، اما برای یک پایگاه‌داده با صدها گیگابایت یا ترابایت حجم، انجام هر کاری در این زمینه مفید خواهد بود.

1-7-4: ساختارهای داده برای یادگیری سریع

یادگیری در شبکه‌های بیزین شامل محاسبات زیادی می‌باشد. برای هر ساختار شبکه در نظر گرفته شده در جستجو، داده باید برای به دست آوردن تعداد نمونه مورد نیاز برای پرکردن جداول احتمال شرطی، مجددا اسکن شود. آیا این داده می‌تواند به صورت ساختار داده‌ای ذخیره شود که نیاز برای اینکه داده بارها و بارها اسکن شود را حذف کند؟ یک روش این است که تعداد را از قبل محاسبه کنیم و مقادیر غیرصفر را در جدول ذخیره کنیم. پایگاه‌داده نمایش داده شده در جدول زیر را در نظر بگیرید. پنج ویژگی در این جدول وجود دارد، دو ویژگی با سه مقدار و سه ویژگی با دو مقدار که تعداد حالات ممکن برابر 4*4*3*3*3=432 حالت خواهد بود. هر جزء حاصلضرب مربوط به یک صفت می‌باشد و سهم آن جز در حاصلضرب بیشتر از تعداد مقادیرش است. در پایگاه‌داده‌های بزرگ، حتی بدون ذخیره‌کردن مواردی که صفر هستند، این طرح ساده باز هم با مشکلات اجرایی در حافظه مواجه خواهد بود.

جدول4-1: داده‌های آب و هوا

Outlook	Temperature	Humidity	Windy	Play
Sunny	Hot	High	False	**No**
Sunny	Hot	High	True	**No**
Overcast	Hot	High	False	**Yes**
Rainy	Mild	High	False	**Yes**
Rainy	Cool	Normal	False	**Yes**
Rainy	Cool	Normal	True	**No**
Overcast	Cool	Normal	True	**Yes**
Sunny	Mild	High	False	**No**
Sunny	Cool	Normal	False	**Yes**

Rainy	Mild	Normal	False	**Yes**
Sunny	Mild	Normal	True	**Yes**
Overcast	Mild	High	True	**Yes**
Overcast	Hot	Normal	False	**Yes**
Rainy	Mild	High	True	**No**

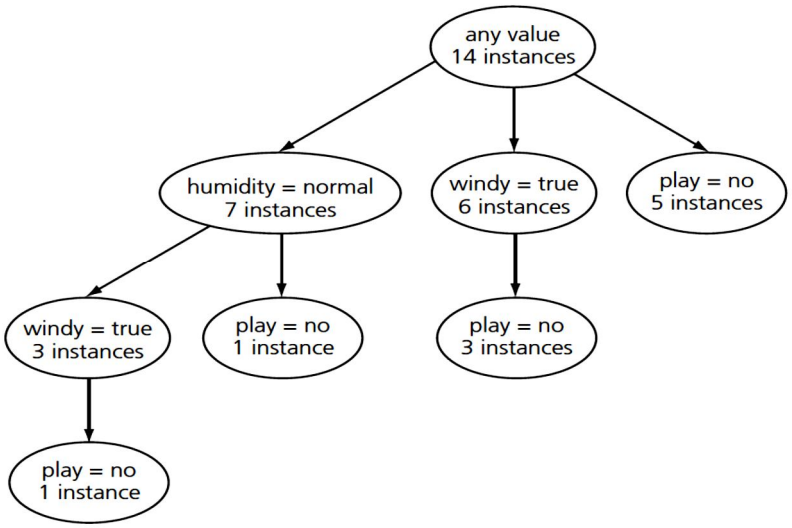
به نظر می‌رسد که تعداد نمونه‌ها را می‌توان به طور موثر در یک ساختار به نام درخت AD[35] ذخیره کرد. ما این حالت را با استفاده از نسخه کاهش یافته‌ای از داده‌های آب و هوا در جدول 4-2 که شامل ویژگی‌های humidity، windy و play می‌شود، شرح می‌دهیم. شکل 12 داده را خلاصه کرده است. تعداد نمونه ممکن برابر 27= 3*3*3 است اگر چه تنها 8 مورد از آنها نشان داده شده اند. مثلا تعداد play = no برابر 5 است.

جدول4-2 داده‌های مربوط به پایگاه‌داده آب و هوا- بخشی از نسخه کاهش یافته پایگاه‌داده

Humidity	Windy	Play	Count
High	True	Yes	1
High	True	No	2
High	False	Yes	2
High	False	No	2
Normal	True	Yes	2
Normal	True	No	1
Normal	False	Yes	4
Normal	False	No	0

[35]All--dimensions

شکل 12 یک درخت AD را برای این داده نشان می‌دهد. هر گره نشان‌دهنده این است که چطور تعداد زیادی نمونه مقادیر ویژگی را که در طی مسیر از ریشه تا آن گره آزمایش شده‌اند نشان می‌دهد. مثلا سمت چپ‌ترین برگ بیان می‌کند که نمونه‌ای با مقادیر humidity = normal، windy = true و play = no وجود دارد و سمت راست‌ترین برگ بیان می‌کند که پنج نمونه با مقدار play = no وجود دارد.



شکل4- 12: بخشی از درخت AD مربوط به پایگاه‌داده آب و هوا

4-7-2: کاهش ابعاد داده

همانطور که در بخش‌های قبل بیان کردیم نیاز به روش‌هایی برای کاهش حجم داده‌ها وجود دارد. کاهش ابعاد داده یکی از راه‌های موثر برای کاهش اندازه پایگاه‌داده است. بسیاری از تکنیک‌های یادگیری ماشین و داده‌کاوی ممکن است برای داده‌هایی با ابعاد زیاد موثر نباشد. در مواردی، انجام برخی پیش پردازش‌ها بر روی داده‌ها برای کاهش حجم داده‌ها مفید خواهد بود. در ادامه به برخی از این موارد اشاره می‌کنیم.

3-7-4: متراکم کردن داده[36]

در این حالت، دو یا چند ویژگی را به صورت یک ویژگی ذخیره می‌کنیم، مثلا مقادیر تاریخ برای 365 روز را 12 ماه کاهش می‌دهیم. انجام این عملیات معمولا برای کاهش حجم داده است که از طریق کاهش تعداد ویژگی‌ها انجام می‌شود، همچنین این عمل روی داده‌ها، دید سطح بالایی از داده را فراهم می‌کند و کشف الگوی داده‌ها ساده‌تر انجام می‌شود.

4-7-4: ایجاد ویژگی[37]

ایجاد ویژگی‌های جدیدی که بتواند اطلاعات مهمی را درباره پایگاه‌داده بدهد کاراتر از ذخیره ویژگی‌های اصلی داده برای حل مساله مورد نظر است. برای این امر روش‌هایی وجود دارد. یکی از روش‌هایی که در این مورد استفاده می‌شود روش ساخت ویژگی[38] است. مثلا خصوصیت *چگالی* به صورت زیر با ترکیب دو خصوصیت *جرم* و *حجم* ساخته می‌شود.

چگالی = جرم/حجم

5-7-4:کاهش تعداد ویژگی

بعضی از الگوریتم‌هایی که برای پردازش داده‌های پایگاه‌داده مورد استفاده قرار می‌گیرند با تعداد ویژگی‌های کمتر عملکرد بهتری دارند و وقتی داده‌های اصلی را که پیش پردازشی روی آنها انجام نشده است، به عنوان ورودی به آنها اعمال می‌کنیم نتیجه مناسبی نخواهد داشت و از طریق کاهش تعداد ویژگی‌ها، زمان و حافظه مورد نیاز برای انجام این الگوریتم‌ها کاهش می‌یابد. همچنین انجام این عمل می‌تواند در کاهش نویز نیز موثر باشد. از روش‌های کاهش ابعاد ویژگی‌ها، به انتخاب ویژگی[39] و استخراج ویژگی[40] می‌توان اشاره نمود. در روش انتخاب ویژگی، خصوصیات مهم با استفاده از برخی تکنیک‌های جستجو انتخاب می‌شوند و این خصوصیات برای

---

[36] Aggregation
37 Feature
38 Feature construction
39 Feature selection
40 Feature extraction

مساله جاری استفاده می‌شوند. هدف از انجام این روش، کاهش ابعاد و حذف نویز می‌باشد، همچنین در سرعت یادگیری نیز موثر است. در روش استخراج ویژگی، خصوصیات اصلی به خصوصیات با ابعاد کمتر تبدیل می‌شوند و از آنها برای مساله استفاده می‌شود. تفاوت استفاده از این دو روش در این است که در روش استخراج ویژگی، همه خصوصیات اصلی مورد استفاده قرار می‌گیرند و در واقع خصوصیات جدید در واقع ترکیب خطی از خصوصیات اصلی هستند. اما در روش انتخاب ویژگی تنها زیرمجموعه‌ای از خصوصیات اصلی انتخاب می‌شوند. دو نمونه از الگوریتم‌هایی که به صورت خطی برای کاهش ابعاد داده استفاده می‌شوند، تحلیل مولفه‌های اصلی[41] و آنالیز تفکیک خطی[42] میباشند. در فصل‌های بعدی به صورت کامل‌تر به بررسی انواع روش‌های مورد استفاده در کاهش ابعاد ویژگی‌ها خواهیم پرداخت.

[41] PCA (Principle Component Analysis)
[42] LDA (Linear Discriminative Analysis)

مراجع

[1]-Ian H.Witten and Eibe Frank, Data Mining: Practical Machine Learning, Second Edition, 2005

[2]-Rahm, H. Hai Do, Data Cleaning: Problems and Current Approaches, 2005

[3]-Daphne Koller, Mehran Sahami, Toward Optimal Feature Selection, 1996

[4]-Lei Yu, Jieping Ye, Huan Liu, Dimensionality Reduction for Data Mining,

[5]-Z. Zhao, H. Liu, Semi-supervised Feature Selection via Spectral Analysis, SDM 2007

[6]- H. Muller, J. Freytag,Problems, Methods, and Challenges in Comprehensive Data Cleansing, 2003

[7]-A. Hernandez, S.J. Stolfo,The merge/purge problem for large databases. Proceedings of the ACM SIGMOD Conference, 1995

[8]-J.I. Maletic, A. Marcus, Data Cleansing: Beyond Integrity Analysis, Proceedings of the Conference on Information Quality, October 2000

[9]-Mong Li Lee, Hongjun Lu, Tok Wang Ling, Yee Teng Ko, Cleansing data for mining and warehousing. Proceedings of the 10th International Conference on Database and Expert Systems Applications, Florence, Italy, August 1999

[10]-S. M. Embury, S. M. Brand, J. S. Robinson, I. Sutherland, F. A. Bisby, W. Gray, A. C. Jones, R. J. White Adapting integrity enforcement techniques for data reconciliation, Information Systems, Vol. 26, 2005, 657-689

[11]-K.-U. Sattler, E. Schallehn A Data Preparation Framework based on a Multi database Language. International Database Engineering Applications Symposium (IDEAS), Grenoble, France, 2001

[12]-V. Raman, J.M. Hellerstein Potter's Wheel: An Interactive Framework for Data Transformation and Cleaning, Proceedings of the 27th VLDB Conference, Roma, Italy, 2001

[13]-M. Tierstein, A Methodology for Data Cleansing and Conversion, 2004

[14]- J. Jebamalar Tamilselvi and V. Saravanan, A Unified Framework and Sequential Data Cleaning Approach for a Data Warehouse, International Journal of Computer Science and Network Security, VOL.8 No.5, May 2008