دانشگاه صنعتی شریف
دانشکده مهندسی کامپیوتر

پایان‌نامه کارشناسی
مهندسی کامپیوتر

تشخیص کلاهبرداری از توالی تراکنش‌های بانکی با استفاده از الگوریتم‌های مختلف یادگیری ماشین

نگارش

کهبد آئینی

استاد راهنما

علی محمد افشین همت‌یار

شهریور ۱۴۰۱

به نام خدا

دانشگاه صنعتی شریف

دانشکده مهندسی کامپیوتر

پایان‌نامه کارشناسی

این پایان‌نامه به عنوان تحقق بخشی از شرایط دریافت درجه کارشناسی است.

عنوان: تشخیص کلاهبرداری از توالی تراکنش‌های بانکی با استفاده از الگوریتم‌های مختلف یادگیری ماشین

نگارش: کهبد آئینی

کمیته ممتحنین

استاد راهنما: علی محمد افشین امضاء:

همت‌یار

استاد مشاور: استاد مشاور امضاء:

استاد مدعو: استاد ممتحن امضاء:

تاریخ:

سپاس

از استاد بزرگوارم، جناب آقای دکتر همت‌یار که با کمک‌ها و راهنمایی‌های بی‌دریغشان، مرا در به سرانجام رساندن این پایان‌نامه یاری داده‌اند، تشکر و قدردانی می‌کنم. همچنین از همکاران عزیزی که با راهنمایی‌های خود در بهبود نگارش این نوشتار سهیم بوده‌اند، صمیمانه سپاسگزارم.

چکیده

امروزه روش‌های مختلف یادگیری ماشین به خصوص برای مجموعه داده‌های جدولی و یادگیری نظارت‌شده ارائه شده است. این روش‌ها در مقایسه با یکدیگر مزایا و معایبی دارند. در این پروژه سعی بر این است که در درجه‌ی اول یک مسئله‌ی واقعی، که تشخیص کلاهبرداری بر اساس توالی تراکنش‌های بانکی می‌باشد، با استفاده از روش‌های انتخابی حل شود و سپس با استفاده از نتایج حل مسئله توسط هر کدام از این روش‌ها، سه روش مختلف را با یکدیگر مقایسه کرده و سعی کنیم نتیجه‌گیری‌ای منطقی داشته باشیم. سه روش انتخاب شده برای بررسی در این پروژه، الگوریتم‌های یادگیری ماشین رگرسیون منطقی، درخت تصمیم و نیوی بیز می‌باشد. این الگوریتم‌ها به دلیل پرکاربرد بودن در حوزه‌ی یادگیری ماشین انتخاب شده‌اند. در نهایت با مقایسه‌ی سه روش، به برتری نیوی بیز و رگرسیون منطقی بر الگوریتم درخت تصمیم رسیدیم و دلایل آن را در فصل آخر بررسی می‌کنیم.

کلیدواژه‌ها: یادگیری نظارت‌شده، مجموعه داده، رگرسیون منطقی، درخت تصمیم، نیوی بیز، تشخیص کلاهبرداری، اعتبارسنجی مدل یادگیری ماشین

فهرست مطالب

فهرست جداول

فهرست تصاویر

فصل ۱

مقدمه

یادگیری ماشین به عنوان یک حوزه مهم در علم داده‌ها، با استفاده از الگوریتم‌ها و روش‌های متنوعی، توانسته است در حل مسائل پیچیده و پیش‌بینی دقیق اطلاعاتی مؤثر باشد. از روش‌های پرکاربرد در یادگیری ماشین می‌توان روش‌های درخت تصمیم، رگرسیون منطقی و نیوی بیز را نام برد که در حوزه یادگیری ماشین جایگاه مهمی دارند. ما در این پروژه ابتدا یک مسئله‌ی واقعی در زمینه‌ی تشخیص کلاهبرداری از روس توالی تراکنش‌های کارت بانکی را به روش هر سه الگوریتم به صورت جداگانه حل کرده و سپس به مقایسه‌ی کیفیت هر الگوریتم در حل این مسئله پرداخته و در نهایت به دنبال چرایی این تفاوت خواهیم گشت.

۱-۱ تعریف مسئله

مسائل یادگیری نظارت‌شده از پرکاربردترین و به‌روزترین مسائل حوزه‌ی علوم کامپیوتر است. این مسائل به خصوص بر روی مجموعه داده‌های جدولی می‌توانند در مسائل بسیار گسترده‌ای راه‌گشا باشند، زیرا مجموعه داده‌های جدولی می‌توانند اطلاعاتی از هر حوزه‌ای را نمایش دهند و در نتیجه مدل یادگیری ماشین می‌تواند مسائل پیش‌بینی و کلاسه‌بندی را در علوم مختلف حل کند. در این پروژه، سعی بر این است که با استفاده از مجموعه داده‌ی مناسب بتوانیم با ورودی گرفتن مجموعه‌ای از تراکنش‌ها، تشخیص دهیم که این تراکنش‌ها توسط یک کلاهبردار انجام شده یا کارت اعتباری در دست صاحب واقعی حساب بوده است. در این پروژه ما مسئله یادگیری نظارت‌شده را بر روی مجموعه داده‌ی تراکنش‌ها با سه الگوریتم متفاوت رگرسیون منطقی، درخت تصمیم و نیوی بیز حل کرده و سه مدل مختلف را آموزش می‌دهیم. سپس این سه مدل را از لحاظ کارکرد و دقت ارزیابی، اعتبارسنجی و مقایسه می‌کنیم. حال که سه مدل مناسب برای حل این مسئله را آموزش دادیم و توانستیم مشکل تشخیص کلاهبرداری را مرتفع کنیم، به بررسی دلایل تفاوت

نتایج حاصل از این سه مدل، با توجه به ماهیت مدل و آموزش آن پرداخته تا بتوانیم به تئوری یادگیری ماشین نیز نگاهی داشته باشیم.

۱-۲ اهمیت موضوع

این مسئله می‌تواند به تشخیص پول‌شویی و دزدی در دنیای امروز کمک به سزایی کند و بانک‌های پیشرفته می‌توانند برای رساندن خدمات بهتر به مشتریان خود، از این مدل‌های یادگیری و تشخیص کلاهبرداری بهره ببرند. همچنین امروزه از آموزش مدل‌های یادگیری ماشین در اقتصاد و مسائل مالی استقبال زیادی شده است و حل چنین مسئله‌ای می‌تواند کاربرد خوبی در دنیای اقتصاد، بانکداری و امنیت مالی داشته باشد. همچنین در انتهای پروژه نیز با بررسی و مقایسه‌ی نتایج مدل‌های مختلف می‌توانیم به دیدگاه بهتری راجع به هر کدام از این مدل‌ها و الگوریتم‌های یادگیری ماشین داشته باشیم.

۱-۳ ادبیات موضوع

در این پروژه از روش‌های رگرسیون منطقی، درخت تصمیم و نیوی بیز برای یادگیری نظارت‌شده روی داده‌های جدولی مورد بررسی و مقایسه قرار گرفتند. نتایج به دست آمده نشان می‌دهد کدام روش بهترین عملکرد را در پیش‌بینی داده‌های جدولی دارد و کدام روش برای مسائل مشابه می‌تواند بهترین گزینه باشد. این پروژه به افزایش دانش و درک ما در حوزه یادگیری ماشین کمک خواهد کرد و در انتخاب روش مناسب برای مسائل واقعی به ما کمک می‌کند.

۱-۴ اهداف پژوهش

در این پژوهش سعی شده است تا مروری کامل بر سه مدل یادگیری ماشین رگرسیون منطقی، درخت تصمیم و نیوی بیز در مسائل یادگیری نظارت‌شده شود و سپس مدلی از هر الگوریتم را روی مجموعه داده‌هایمان آموزش دهیم و مسئله‌ی تشخیص کلاهبرداری را با هر یک از این مدل‌ها حل کنیم. هدف بعدی نیز مقایسه‌ی سه مدل آموزش‌داده‌شده توسط این سه الگوریتم است که بتوانیم بهترین مدل را با توجه به نیاز مسئله برای تشخیص واقعی کلاهبرداری در دنیای واقعی ارائه دهیم. هدف آخر این پروژه نیز تلاش برای درک دلیل تفاوت کارکرد مدل‌های آموزش‌دیده و سعی بر نتیجه‌گیری بر اساس ماهیت این الگوریتم‌ها می‌باشد.

۱-۵ ساختار پایان‌نامه

در فصل اول به تعریف مسئله و آشنایی با ادبیات موضوع و هدف پروژه آشنا می‌شویم. همچنین در این فصل ساختار پروژه را نیز بررسی می‌کنیم. در فصل دوم به مفاهیم اصلی پروژه، یعنی آشنایی و شناسایی کامل مجموعه داده‌ها و بررسی کارکرد کلی هر سه روش می‌پردازیم. سپس در فصل سوم به کارهای پیشین در یادگیری نظارت‌شده، یعنی نگاهی کلی به الگوریتم‌های دیگر قابل استفاده در این مسئله می‌پردازیم. در فصل چهارم روش کار این الگوریتم‌ها را روی مجموعه داده مشاهده می‌کنیم و به عبارتی دیگر، مسئله را به سه روش حل می‌کنیم. در فصل پنجم و بررسی نتایج جدید، به مقایسه‌ی آماری نتایج به دست آمده از هر الگوریتم می‌پردازیم تا در فصل آخر و نتیجه‌گیری نهایی بتوانیم نتایج منطقی از مقایسه‌ی خروجی الگوریتم‌های مختلف بگیریم.

فصل ۲

مفاهیم اولیه

در این فصل به بررسی و مرور مفاهیم اولیه‌ای می‌پردازیم که در این پایان‌نامه مورد استفاده قرار خواهند گرفت. بررسی این مفاهیم به درک عمیق‌تر روش کار در فصول بعدی و در نهایت علل تفاوت الگوریتم‌های مورد استفاده، کمک شایانی خواهد کرد. همچنین در ابتدای این فصل به آشنایی کامل و کاوش عمیق مجموعه داده‌ی خود خواهیم پرداخت تا در پیاده‌سازی الگوریتم‌ها بتوانیم به بهترین و دقیق‌ترین شکل عمل کرده و نتایج واقعی‌تری را کسب کنیم.

۲-۱ مجموعه داده

در این قسمت می‌خواهیم مجموعه‌داده‌ی استفاده شده در این پایان‌نامه را به دقت و تفصیل بشناسیم و آن را آماده‌ی استفاده برای پیاده‌سازی الگوریتم‌ها کنیم.

۲-۱-۱ معرفی مجموعه دادها

هدف ما از انجام این پروژه، آموزش مدلی برای تشخیص کلاهبرداری از کارت بانکی با استفاده از دنباله‌ای از تراکنش‌های بانکی می‌باشد. در نتیجه مجموعه داده‌ی انتخاب شده مجموعه‌ای از دنباله‌ای از تراکنش‌ها و سلامت تراکنش‌ها می‌باشد. مجموعه داده‌ی ما دارای ۳۱ ستون می‌باشد. ستون اول بیانگر زمان به دست آوردن مجموعه تراکنش‌ها می‌باشد که در مسئله‌ی یادگیری ما کمکی نمی‌کند و از مجموعه داده‌ی تحت آموزش ما حذف خواهد شد. ستون دو تا بیست و نه مقدار تراکنش را نشان می‌دهد. یعنی در کل ما به ازای هر نمونه، ۲۸ تراکنش را نظارت کردیم. همچنین ستون سی‌ام بیانگر مقدار موجودی کارت پس از انجام

این تراکنش‌ها می‌باشد. در نهایت در ستون سی و یکم ما کلاسه‌بندی هر داده را داریم. کلاسه‌بندی به این صورت انجام شده است که اگر یک نمونه به عنوان داده‌ی کلاهبرداری مشخص باشد، سطر متناظر داده در ستون کلاس برابر با یک خواهد بود و در غیر این صورت که نمونه سالم بوده، سطر متناظر در ستون کلاس برابر صفر خواهد بود.

۲-۱-۲ نرمال‌سازی مجموعه داده

از آنجایی که مقادیر موجودی کارت‌ها می‌تواند تفاوت زیادی داشته باشد یا اعداد بسیار بزرگی باشند، برای سادگی کار این ستون از مجموعه داده را نرمال‌سازی می‌کنیم. این کار را می‌توان به سادگی با تابع scale در زبان R انجام داد. این کار باعث می‌شود که اعداد بیش از اندازه بزرگ یا بیش از اندازه کوچک در محاسبات ما نباشند و تمام اعداد را در یک بازه‌ی خاص قرار می‌دهیم تا احتمال کارکرد نادرست توابع یادگیری را کاهش دهیم.

۲-۱-۳ بررسی دقیق مجموعه داده

حال با استفاده از دستور head در زبان R پنج داده‌ی ابتدایی مجموعه داده را می‌بینیم.

```
> head(dataframe)
        V1          V2        V3         V4          V5          V6
1 -1.3598071 -0.07278117 2.5363467  1.3781552 -0.33832077  0.46238778
2  1.1918571  0.26615071 0.1664801  0.4481541  0.06001765 -0.08236081
3 -1.3583541 -1.34016307 1.7732093  0.3797796 -0.50319813  1.80049938
4 -0.9662717 -0.18522601 1.7929933 -0.8632913 -0.01030888  1.24720317
5 -1.1582331  0.87773675 1.5487178  0.4030339 -0.40719338  0.09592146
6 -0.4259659  0.96052304 1.1411093 -0.1682521  0.42098688 -0.02972755
         V7         V8         V9         V10         V11         V12
1  0.23959855  0.09869790  0.3637870  0.09079417 -0.5515995 -0.61780086
2 -0.07880298  0.08510165 -0.2554251 -0.16697441  1.6127267  1.06523531
3  0.79146096  0.24767579 -1.5146543  0.20764287  0.6245015  0.06608369
4  0.23760894  0.37743587 -1.3870241 -0.05495192 -0.2264873  0.17822823
5  0.59294075 -0.27053268  0.8177393  0.75307443 -0.8228429  0.53819555
6  0.47620095  0.26031433 -0.5686714 -0.37140720  1.3412620  0.35989384
        V13        V14        V15        V16        V17         V18
1 -0.9913898 -0.3111694  1.4681770 -0.4704005  0.20797124  0.02579058
2  0.4890950 -0.1437723  0.6355581  0.4639170 -0.11480466 -0.18336127
3  0.7172927 -0.1659459  2.3458649 -2.8900832  1.10996938 -0.12135931
4  0.5077569 -0.2879237 -0.6314181 -1.0596472 -0.68409279  1.96577500
5  1.3458516 -1.1196698  0.1751211 -0.4514492 -0.23703324 -0.03819479
6 -0.3580907 -0.1371337  0.5176168  0.4017259 -0.05813282  0.06865315
        V19         V20         V21         V22         V23         V24
1  0.40399296  0.25141210 -0.018306778  0.277837576 -0.11047391  0.06692807
2 -0.14578304 -0.06908314 -0.225775248 -0.638671953  0.10128802 -0.33984648
3 -2.26185710  0.52497973  0.247998153  0.771679402  0.90941226 -0.68928096
4 -1.23262197 -0.20803778 -0.108300452  0.005273597 -0.19032052 -1.17557533
5  0.80348692  0.40854236 -0.009430697  0.798278495 -0.13745808  0.14126698
6 -0.03319379  0.08496767 -0.208253515 -0.559824796 -0.02639767 -0.37142658
        V25         V26         V27         V28      Amount Class
1  0.1285394 -0.1891148  0.133558377 -0.02105305  0.24496383     0
2  0.1671704  0.1258945 -0.008983099  0.01472417 -0.34247394     0
3 -0.3276418 -0.1390966 -0.055352794 -0.05975184  1.16068389     0
4  0.6473760 -0.2219288  0.062722849  0.06145763  0.14053401     0
5 -0.2060096  0.5022922  0.219422230  0.21515315 -0.07340321     0
6 -0.2327938  0.1059148  0.253844225  0.08108026 -0.33855582     0
```

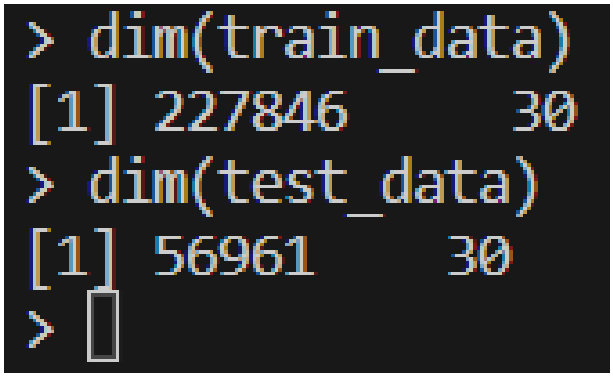
شکل ۲-۱: ۵ داده‌ی اول مجموعه داده

```
> dim(dataframe)
[1] 284807     30
>
```

شکل ۲-۲: ابعاد مجموعه داده

همانطور که مشخص است V۱ تا V۲۸ نشان‌دهنده‌ی ۲۸ تراکنش مورد نظر بوده و مقدار پول داخل حساب نیز در ستون بعدی یعنی Amount آمده است. همچنین ستون آخر نیز کلاس این مجموعه تراکنش‌ها را نشان می‌دهد و عدد صفر نشان‌دهنده‌ی سلامت تراکنش‌ها و عدد یک بیانگر کلاهبرداری می‌باشد. حال که با ستون‌های این مجموعه داده آشنا شدیم به بررسی سطرها (نمونه‌ها) می‌پردازیم.

با استفاده از تابع dim ابعاد مجموعه داده‌های خود را دیدیم که دارای ۳۰ ستون (ویژگی) و ۲۸۴۸۰۷ سطر (نمونه) می‌باشد. یعنی به تعداد ۲۸۴۸۰۷ مجموعه تراکنش داریم که هر کدام به عنوان مجموعه‌ی سالم و یا کلاهبرداری علامت‌گذاری شده اند.



شکل ۲-۳: ابعاد زیر مجموعه داده آموزش و آزمون

۲-۱-۴ تقسیم‌بندی مجموعه داده

حال که با مجموعه داده به طور کامل آشنا شدیم، باید مجموعه داده را برای پیاده‌سازی الگوریتم‌های مختلف یادگیری ماشین به دو بخش داده‌های یادگیری [۱] و داده‌های آزمون [۲] تقسیم‌بندی کنیم. در این تقسیم‌بندی، هشتاد درصد داده‌های به یادگیری مدل و بیست درصد باقی‌مانده برای انجام آزمون مدل و اعتبارسنجی آن است.

همان‌طور که دیدیم، ۲۲۷۸۴۶ داده به یادگیری و ۵۶۹۶۱ داده به اعتبارسنجی مدل تخصیص داده شده‌اند. شایان ذکر است که این داده‌ها کاملا به صورت تصادفی به این بخش تقسیم شده‌اند. از آنجایی که هدف این پایان‌نامه اعتبارسنجی و مقایسه مدل‌های مختلف می‌باشد، دقیقا همین زیرمجموعه داده‌های به دست آمده را برای یادگیری و آزمودن مدل‌های مختلف استفاده می‌کنیم تا از تفاوت احتمالی ناشی از متفاوت بودن مجموعه داده‌ی در اختیار مدل‌ها، جلوگیری شود.

۲-۲ رگرسیون منطقی

۲-۲-۱ معرفی الگوریتم

الگوریتم رگرسیون منطقی [۳] [۴] از روش‌های پرکاربرد در مسائل یادگیری نظارت‌شده می‌باشد. این الگوریتم غالبا برای مسائل کلاسه‌بندی استفاده می‌شود. این روش همانند دیگر الگوریتم‌های یادگیری ماشین، با

[۱] Train
[۲] Test
[۳] Logistic Regression
[۴]

یادگیری از یک مجموعه داده، هدف پیش‌بینی یک ویژگی با توجه به ویژگی‌های دیگر یک نمونه را دارد. پیاده‌سازی و مثال این روش در بخش نتایج به تفصیل آمده است. این الگوریتم، مدلی را آموزش می‌دهد که می‌تواند احتمال این که ورودی به یک کلاس تعلق داشته باشد را محاسبه کند. به عنوان مثال در مسئله‌ی این پایان‌نامه، هدف ما آموزش مدلی است که با دریافت دنباله‌ای از تراکنش‌های بانکی تشخیص دهد که آیا این تراکنش‌ها از طرف یک کلاهبردار بوده یا کارت بانکی در دست صاحب حساب بوده است. حال اگر یک مدل رگرسیون منطقی را روی مجموعه داده‌ی مناسب آموزش دهیم، این مدل پس از آموزش دیدن، به ازای ورودی گرفتن یک دنباله از تراکنش‌های بانکی، احتمال این که این تراکنش‌ها توسط یک کلاهبردار انجام شده باشد را به ما خروجی می‌دهد. این احتمال، یک عدد بین صفر و یک خواهد بود، اما خروجی کد ما باید به صورت کلاهبرداری یا غیر کلاهبرداری باشد. بنابراین ما باید این عدد احتمال را به فضای دو تایی نگاشت کنیم. برای این کار یک آستانه‌ای را تعریف می‌کنیم، که در چنین مسائلی غالبا آستانه را برابر ۵.۰ در نظر می‌گیرند. در این مدل نیز چنین کاری انجام شد و خروجی مدل برای هر نمونه، اگر بزرگ‌تر و یا مساوی ۵.۰ بود، آن نمونه به عنوان کلاهبرداری و اگر کوچک‌تر از ۵.۰ بود، آن نمونه به عنوان مجموعه تراکنش‌های سالم در نظر گرفته می‌شود.

۲-۲-۲ روش کارکرد رگرسیون منطقی

حال به بررسی دقیق روش کارکرد رگرسیون منطقی می‌پردازیم.

رگرسیون منطقی بسیار مشابه با رگرسیون خطی می‌باشد با این تفاوت که رگرسیون منطقی، خروجی به دست آمده از رگرسیون خطی که به صورت پیوسته است را با استفاده از تابع سیگموید به یک عدد بین صفر و یک نگاشت می‌کند و همان‌طور که در قبل اشاره شد، این عدد به تعبیری احتمال تعلق نمونه به دو گروه صفر و یک است.

فرضیه در رگرسیون منطقی این است که وجود یک تابع خطی در فضای ویژگی‌ها وجود دارد که توانایی تمیز بین دو دسته‌بندی را دارد. این تابع خطی معمولاً به صورت زیر تعریف می‌شود:

$$z = b_0 + b_1 x_1 + b_2 x_2 + ... + b_n x_n \qquad (۱-۲)$$

که در آن z نشان‌دهنده خروجی نهایی مدل است و x_1 تا x_n ویژگی‌های ورودی می‌باشند. عوامل b_0 تا b_n نیز وزن‌ها (weights) مدل هستند که با یادگیری و بهینه‌سازی مقادیر آنها تعیین می‌شود.

با استفاده از تابع لجستیک، خروجی z به دست آمده از تابع خطی به مقداری بین ۰ و ۱ نگاشت می‌شود.

تابع لجستیک یا sigmoid به صورت زیر تعریف می‌شود:

$$p = \frac{۱}{(۱ + e^{(} - z))} \tag{۲-۲}$$

که در آن p احتمال تعلق داده به یکی از دو دسته‌بندی است. این تابع، مقدار z را به بازه‌ی [۰، ۱] محدود می‌کند و احتمال متناظر با هر دسته‌بندی را محاسبه می‌کند.

برای آموزش مدل رگرسیون منطقی، از روش بهینه‌سازی به نام "گرادیان کاهشی" [۵] استفاده می‌شود. هدف در این روش، کمینه کردن تابع هزینه [۶] است که به صورت زیر تعریف می‌شود:

$$cost = \frac{-۱}{m * (y * log(p) + (۱ - y) * log(۱ - p))} \tag{۳-۲}$$

در این تابع m برابر تعداد نمونه‌ها، y برابر برچسب‌های واقعی در کلاس و p نیز احتمالات محاسبه شده توسط مدل می‌باشد.

هدف در یادگیری مدل، بهینه کردن مقادیر وزن‌ها b تا b_n به گونه‌ای می‌باشد که تابع هزینه کمینه شود.

۲-۳ درخت تصمیم

۲-۳-۱ معرفی الگوریتم

الگوریتم درخت تصمیم [۷] یک روش یادگیری ماشین نظارت‌شده است که برای حل مسائل پیش‌بینی و طبقه‌بندی در داده‌های جدولی استفاده می‌شود. این الگوریتم بر مبنای ساختار درختی تصمیم‌گیری کار می‌کند که با تقسیم بندی داده‌ها به گروه‌های جزئی‌تر، معیاری را برای تصمیم‌گیری در مورد هر نمونه ارائه می‌دهد.

فرایند ساخت یک درخت تصمیم با الگوریتم درخت تصمیم شامل چند مرحله است. ابتدا، از روی مجموعه داده اولیه، یک ریشه برای درخت تصمیم ساخته می‌شود. سپس، با توجه به ویژگی‌های موجود در داده‌ها، بهینه‌سازی معیاری برای تقسیم بندی داده‌ها انتخاب می‌شود. این معیار معمولاً بر اساس اطلاعات و راهنمایی موجود در داده‌ها است که سعی در جداکردن دسته‌ها با بیشترین اطلاعات ممکن دارد.

[۵] Gradient Descent
[۶] Cost Function
[۷]

بعد از تقسیم بندی اولیه، فرایند ساخت درخت بازگشتی ادامه می‌یابد. هر بخش از درخت به عنوان یک گره در نظر گرفته می‌شود که به دو یا چند زیرگره تقسیم می‌شود. این تقسیم براساس معیاری است که بهینه‌سازی شده است و هدف آن افزایش خلوص و یکنواختی هر زیرگره است.

فرایند ساخت درخت تصمیم به صورت بازگشتی ادامه می‌یابد تا به یک شرط پایان برسد، مانند دستیابی به عمق مشخص یا تقسیم‌بندی دسته‌ها به حداقل تعداد نمونه‌ها. در نهایت، درخت تصمیم ساخته شده برای پیش‌بینی و طبقه‌بندی داده‌های جدید استفاده می‌شود.

الگوریتم درخت تصمیم به دلیل سادگی، قابلیت تفسیر، قدرت در پردازش داده‌های جدولی و عملکرد خوب در مسائل مختلف، محبوبیت بالایی در زمینه یادگیری ماشین برای داده‌های جدولی به دست آورده است.

۲-۳-۲ روش کارکرد درخت تصمیم

این الگوریتم برای ما یک درخت تشکیل می‌دهد و در هر نود میانی این درخت بر اساس مقدار یک ویژگی (صفت) داده، مجموعه داده را به دو بخش تقسیم می‌کند تا جایی که در برگ‌ها ما بر اساس مقادیر صفات بررسی‌شده، می‌توانیم به یک طبقه یا کلاسه‌بندی برسیم. انتخاب بهترین صفت و مقدار آن برای تقسیم زیرمجموعه‌ی حاضر در نود میانی، بر اساس میزان پراکندگی داده‌ها می‌باشد که با معیارهایی مانند ضریب جینی یا ضریب تحمل خطا اندازه‌گیری می‌شوند.

ابتدا برخی از مفاهیم و تعاریف پرکاربرد در این الگوریتم را بررسی می‌کنیم.

ناخالصی [۸] به اندازه‌گیری یکنواختی متغیر هدف در یک زیرمجموعه از داده‌ها که به درجه تصادف یا عدم قطعیت در مجموعه‌ای از نمونه‌ها اشاره دارد ناخالصی می‌گوییم.

اندازه‌ی افزایش اطلاعات [۹] بهره‌ی اطلاعات میزان کاهش انتروپی یا واریانس است که ناشی از تقسیم یک مجموعه داده براساس یک ویژگی خاص است.

انتروپی [۱۰] انتروپی اندازه‌گیری درجه تصادف یا عدم قطعیت در مجموعه داده است. در صورت طبقه‌بندی، آنتروپی بر اساس توزیع برچسب‌های کلاس در مجموعه داده اندازه‌گیری می‌شود

ناخالصی جینی [۱۱] ناخالصی جینی امتیازی است که میزان دقت یک تقسیم بین گروه‌های طبقه‌بندی شده را ارزیابی می‌کند. بی‌نقصی ژینی امتیازی را در محدوده‌ی ۰ تا ۱ ارزیابی می‌کند، که ۰ وقتی است که

[۸] Impurity
[۹] Information Gain
[۱۰] Entropy
[۱۱] Gene Impurity

تمام مشاهدات به یک کلاس تعلق داشته باشند و ۱ یک توزیع تصادفی از عناصر در داخل کلاس‌ها است.

در ادامه، به توضیح مراحل کارکرد این الگوریتم به صورت ریاضی پرداخته می‌شود

مرحله ۱: **انتخاب ویژگی تقسیم‌کننده** در این مرحله، الگوریتم برای تصمیم‌گیری درباره‌ی کدام ویژگی باید از آن برای تقسیم داده‌ها استفاده کند، از یک معیار استفاده می‌کند. معیارهای متداول می‌توانند شامل اندازه‌ی افزایش اطلاعات، ،ضریب جینی انتروپی یا ضریب تحمل خطا باشند. الگوریتم برای هر ویژگی مقدار معیار را محاسبه کرده و ویژگی را انتخاب می‌کند که مقدار معیار بیشترین افزایش یا کمترین خطا را به همراه دارد.

مرحله ۲: **تقسیم داده‌ها بر اساس ویژگی تقسیم‌کننده** در این مرحله، با استفاده از ویژگی تقسیم‌کننده انتخاب شده در مرحله قبل، داده‌ها به دو یا چند زیرگروه تقسیم می‌شوند. داده‌هایی که مقدار ویژگی تقسیم‌کننده در آنها برابر با یک مقدار خاص است، در یک زیرگروه قرار می‌گیرند و داده‌هایی که مقدار ویژگی تقسیم‌کننده در آنها برابر با مقدار دیگری است، در زیرگروه دیگری قرار می‌گیرند.

مرحله ۳: **تولید زیردرخت‌ها** در این مرحله، الگوریتم به صورت بازگشتی روی هر زیرگروه داده‌ها عمل می‌کند. اگر زیرگروه داده‌ها تماماً به یک دسته تعلق داشته باشند، یعنی همه‌ی داده‌های زیرگروه در یک دسته قرار بگیرند، آنگاه به عنوان یک برگ مشخص می‌شود. در غیر این صورت، مرحله‌های ۱ و ۲ را برای زیرگروه جدید تکرار می‌کنیم و یک تقسیم بندی جدید انجام می‌دهیم.

مرحله ۴: **توقف الگوریتم** مرحله‌ی توقف الگوریتم ممکن است در شرایطی که تعیین شده است، مانند عمق مشخص یا تعداد داده‌های مورد نیاز برای تقسیم، رخ دهد. در این صورت، فرآیند ساخت درخت تصمیم پایان می‌یابد و درخت نهایی برای پیش‌بینی و طبقه‌بندی داده‌های جدید استفاده می‌شود.

با تکرار مراحل فوق، درخت تصمیم کامل می‌شود و قادر است به صورت سلسله‌مراتبی داده‌ها را طبقه‌بندی کند و پیش‌بینی کند.

۲-۴ نیوی بیز

۲-۴-۱ معرفی الگوریتم

الگوریتم نیوی بیز [۱۲] [۱۳] از روش‌های یادگیری ماشین مبتنی بر اصول بیز است که برای مسائل دسته‌بندی استفاده می‌شود. این الگوریتم بر اساس قاعده بیز استنتاج می‌کند و فرضیه ناخودآگاه "بیزی" را معتبر در

[۱۲] Naive Bayes
[۱۳]

نظر می‌گیرد.

فرضیه ناخودآگاه "نیوی بیز" این است که ویژگی‌های ورودی مستقل از یکدیگر هستند و تاثیری تعاملی بین آنها ندارند (از اینجا نیوی می‌گویند). این فرضیه ساده‌سازی قوانین بیز را فراهم می‌کند و از محاسبات پیچیده جلوگیری می‌کند.

الگوریتم نیوی بیز برای مسائل دسته‌بندی از احتمالات شرطی استفاده می‌کند. در فرآیند آموزش، احتمالات شرطی برای هر ویژگی به شرط داشتن دسته‌ها محاسبه می‌شوند و در فرآیند پیش‌بینی، با استفاده از قاعده بیز و با توجه به ویژگی‌های جدید، احتمال تعلق به هر دسته محاسبه می‌شود. سپس دسته با بیشترین احتمال به عنوان پاسخ انتخاب می‌شود.

از مزایای الگوریتم نیوی بیز می‌توان به سادگی پیاده‌سازی، سرعت بالا در آموزش و پیش‌بینی، کارایی خوب در مجموعه داده‌های بزرگ و قابلیت استفاده در مسائل با ویژگی‌های متنی اشاره کرد. با این حال، این الگوریتم فرضیات ساده‌سازی خود را دارد و ممکن است در برخی موارد کمیت داده‌ها یا وجود وابستگی‌های بین ویژگی‌ها تاثیر مخربی داشته باشد.

۲-۴-۲ روش کارکرد نیوی بیز

فرض کنید مجموعه داده‌های ما شامل n نمونه است که هر کدام دارای m ویژگی هستند. همچنین داده‌های ما به k دسته مختلف تقسیم شده‌اند. هدف ما در نیوی بیز، پیش‌بینی دسته‌بندی یا دسته‌ی مناسب برای داده‌های جدید است.

برای این منظور، از قاعده بیز استفاده می‌کنیم که بر اساس احتمالات شرطی عمل می‌کند. بر اساس قاعده بیز، ما به دنبال یافتن احتمال شرطی $P(C|X)$ هستیم، به این معنی که ما می‌خواهیم دسته‌بندی (C) را با توجه به ویژگی‌ها (X) پیش‌بینی کنیم.

با استفاده از قاعده بیز، می‌توانیم این احتمال را به صورت زیر محاسبه کنیم:

$$P(C|X) = \frac{P(X|C)P(C)}{P(X)} \qquad (۴-۲)$$

در این رابطه:

- $P(C|X)$ نشان‌دهنده احتمال شرطی دسته‌بندی C به شرط داشتن ویژگی‌ها X است.
- $P(X|C)$ نشان‌دهنده احتمال شرطی داشتن ویژگی‌ها X به شرط دسته‌بندی C است.

- $P(C)$ نشان‌دهنده احتمال عادی دسته‌بندی است.
- $P(X)$ نشان‌دهنده احتمال عادی وقوع ویژگی‌ها X است.

در الگوریتم نیوی بیز، فرض بر این است که ویژگی‌ها به صورت مستقل از یکدیگر عمل می‌کنند، به این معنی که وجود یک ویژگی خاص در دسته‌بندی هیچ تأثیری بر روی وجود یا عدم وجود ویژگی‌های دیگر ندارد. این فرض ساده‌سازی شده به ما کمک می‌کند تا محاسبات را سریع‌تر انجام دهیم و الگوریتم را قابل اجرا در مواقعی با حجم بالای داده‌ها کنیم.

الگوریتم نیوی بیز در مسائل دسته‌بندی، بر اساس قاعده بیز کار می‌کند و از احتمالات شرطی استفاده می‌کند. این الگوریتم بر اساس فرضیه‌ای ساده به نام "فرضیه نیوی بیز" عمل می‌کند.

فرضیه نیوی بیز این است که ویژگی‌ها در دسته‌بندی مستقل از یکدیگر عمل می‌کنند، به این معنی که وجود یک ویژگی خاص در دسته‌بندی هیچ تأثیری بر روی وجود یا عدم وجود ویژگی‌های دیگر ندارد. این فرضیه ساده‌ترین شکل از استقلال شرطی را مطرح می‌کند.

برای استفاده از الگوریتم نیوی بیز در دسته‌بندی، ابتدا باید مجموعه داده‌های آموزشی را آماده کنیم. این مجموعه داده‌ها باید شامل ویژگی‌ها و برچسب‌های دسته‌بندی مربوط به هر نمونه باشد.

سپس، با استفاده از مجموعه داده‌های آموزشی، باید مدل نیوی بیز را آموزش دهیم. در آموزش این مدل، برای هر دسته‌بندی محتمل، میزان وقوع هر ویژگی به شرط دسته‌بندی محاسبه می‌شود. برای محاسبه این احتمالات شرطی، از مجموعه داده‌های آموزشی استفاده می‌شود.

بعد از آموزش مدل، می‌توان از آن برای پیش‌بینی دسته‌بندی داده‌های تست استفاده کرد. با استفاده از مدل نیوی بیز، احتمال وقوع هر دسته‌بندی برای داده‌های تست محاسبه می‌شود و دسته‌بندیی که احتمال بالاتری دارد، به عنوان پیش‌بینی نهایی برای داده‌های تست انتخاب می‌شود.

فصل ۳

کارهای پیشین

۳-۱ مسائل یادگیری نظارت‌شده

مقالات و پژوهش‌های زیادی در زمینه یادگیری نظارت‌شده صورت گرفته است که تکنیک‌ها و روش‌های مختلفی را برای حل مسائل پیش‌بینی و دسته‌بندی ارائه می‌دهند. در ادامه، چندین پاراگراف درباره کارهای مرتبط با یادگیری نظارت‌شده به زبان فارسی ارائه می‌شود:

یکی از رویکردهای پژوهشی در یادگیری نظارت‌شده استفاده از شبکه‌های عصبی عمیق است. این شبکه‌ها با تعداد زیادی لایه و واحد محاسباتی، توانایی استخراج ویژگی‌های پیچیده از داده‌ها را دارند و در حل مسائل پیچیده و با ابعاد بالا موفق عمل می‌کنند. مقالات بسیاری در این حوزه منتشر شده‌اند که به آموزش، ساختار و بهینه‌سازی شبکه‌های عصبی عمیق، و نحوه استفاده از آنها در حوزه‌های مختلف تمرکز دارند.

یکی از موضوعات مهم در یادگیری نظارت‌شده، استفاده از روش‌های تقویتی است. در این روش‌ها، مدل‌های یادگیری نظارت‌شده با استفاده از بازخورد و تعامل مستقیم با محیط، تلاش می‌کنند تا عملکرد خود را بهبود دهند. این روش‌ها در مسائلی که محیط تغییر پذیر و پیچیده است، عملکرد بهتری نسبت به روش‌های سنتی ارائه می‌دهند و در حوزه‌هایی مانند بازی‌های رایانه‌ای و رباتیک کاربرد گسترده‌ای دارند.

در نهایت، با پیشرفت تکنولوژی و روش‌های یادگیری نظارت‌شده، پژوهش‌های بیشتری در این حوزه صورت می‌گیرد. این پژوهش‌ها به بهبود روش‌های یادگیری نظارت‌شده، مقایسه و ارزیابی روش‌های مختلف، و افزایش کارایی و دقت مدل‌ها تمرکز دارند. همچنین، بررسی تاثیر پارامترهای مختلف در عملکرد الگوریتم‌ها و توسعه روش‌های جدید نیز از جمله موضوعات پژوهشی مهم در یادگیری نظارت‌شده

است.

۳-۲ مجموعه‌داده‌های جدولی

به منظور تسهیم دانش و افزایش دسترسی به اطلاعات در حوزه یادگیری ماشین بر روی داده‌های جدولی، تعداد زیادی پژوهش و کار پیشین در این زمینه انجام شده است. در ادامه، چندین پاراگراف درباره کارهای مرتبط با یادگیری ماشین بر روی داده‌های جدولی به زبان فارسی ارائه می‌شود:

یکی از موضوعات پرطرفدار در یادگیری ماشین بر روی داده‌های جدولی، استفاده از روش‌های ترکیبی است. در این رویکرد، مدل‌های مختلفی مانند درخت تصمیم، شبکه‌های عصبی و روش‌های مبتنی بر مجموعه‌های مدل یادگیری ماشین، با هم ترکیب می‌شوند تا یک مدل قوی‌تر و دقیق‌تر ایجاد شود. این روش‌های ترکیبی به کمک یکدیگر، توانایی پردازش داده‌های جدولی پیچیده را دارند و نتایج بهتری در پیش‌بینی و دسته‌بندی ارائه می‌دهند.

یکی از مسائل مهم در یادگیری ماشین بر روی داده‌های جدولی، مدیریت و پردازش داده‌های ناهمگن است. داده‌های جدولی ممکن است شامل انواع مختلفی از مشخصه‌ها مانند متن، عدد، تاریخ و شناسه باشند. در این صورت، روش‌های مختلفی مانند تبدیل داده، کدگذاری متغیرها و تحلیل اجزای مشترک مورد استفاده قرار می‌گیرند تا با تنوع داده‌ها به درستی برخورد شود و از اطلاعات مفیدی که در هر مشخصه جای دارد، استفاده شود.

یکی از رویکردهای پژوهشی مورد توجه در حوزه یادگیری ماشین بر روی داده‌های جدولی، استفاده از روش‌های خودکارسازی و خوشه‌بندی است. این روش‌ها به کمک الگوریتم‌های مختلفی مانند کی‌ـ میانگین، اندازه‌گیری فاصله و اشتباه‌یابی، داده‌ها را به گروه‌های مشابه تقسیم بندی می‌کنند و با تجمیع اطلاعات درون هر گروه، تحلیل و پیش‌بینی دقیق‌تری را انجام می‌دهند.

در نهایت، با توجه به پیشرفت روزافزون تکنولوژی و روش‌های یادگیری ماشین، پژوهش‌های بیشتری در حوزه یادگیری ماشین بر روی داده‌های جدولی انجام می‌شود. این پژوهش‌ها بر طراحی و توسعه روش‌های بهینه و قدرتمند، بهبود عملکرد پیش‌بینی و دسته‌بندی، و استفاده از اطلاعات جانبی موجود در داده‌های جدولی تمرکز دارند.

۳-۳ پروژه‌های تشخیص کلاهبرداری با استفاده از یادگیری ماشین

پروژه‌های یادگیری ماشین در حوزه تشخیص تقلب بسیار رایج هستند و در صنایع مختلف به کار می‌روند. تشخیص تقلب شامل شناسایی و جلوگیری از فعالیت‌ها یا تراکنش‌های تقلبی با استفاده از تکنیک‌های پیشرفته تحلیل داده می‌شود. الگوریتم‌های یادگیری ماشین به دلیل قدرت خود در تشخیص الگوها، ناهنجاری‌ها و رفتار مشکوک در مجموعه داده‌های بزرگ، بسیار موثر در وظایف تشخیص تقلب هستند. این الگوریتم‌ها می‌توانند بر روی داده‌های تاریخی که شامل مثال‌های معتبر و تقلبی است، آموزش داده شوند تا الگوهای مرتبط با فعالیت‌های تقلبی یاد بگیرند و شناسایی کنند.

پروژه‌های یادگیری ماشین در حوزه تشخیص تقلب معمولاً شامل چند مرحله است. در ابتدا، مجموعه داده باید آماده شود که شامل پیش‌پردازش داده، مهندسی ویژگی و توازن داده با استفاده از تکنیک‌های مناسب باشد تا مجموعه داده برای آموزش مدل مناسب باشد. سپس الگوریتم مناسبی از میان الگوریتم‌های یادگیری ماشین مانند سه الگوریتم استفاده شده در این پایان‌نامه یا دیگر الگوریتم‌ها مانند جنگل تصادفی [۱] [۲] یا شبکه‌های عصبی [۳] [۴] انتخاب و با استفاده از مجموعه داده آموزش داده می‌شود. سپس مدل با استفاده از معیارهای عملکرد مناسب مانند دقت، صحت، بازخوانی و امتیاز F۱ ارزیابی می‌شود تا کارایی آن در تشخیص تقلب ارزیابی شود.

برای بهبود عملکرد مدل‌های تشخیص تقلب، روش‌های ترکیبی مانند رشد تدریجی و جنگل تصادفی قابل استفاده هستند. این روش‌ها با ترکیب چندین مدل، یک سیستم تشخیص تقلب قوی و دقیق‌تر ایجاد می‌کنند. همچنین، روش‌های انتخاب ویژگی و کاهش بعد می‌توانند برای بهبود کارایی و کارآیی مدل‌ها استفاده شوند.

سیستم‌های تشخیص تقلب بر مبنای یادگیری ماشین به صورت زمان‌بندی واقعی [۵] نیز توسعه داده می‌شوند که مدل‌ها در محیط‌های عملیاتی برای پیش‌بینی و تشخیص فعالیت‌های تقلبی به صورت پیوسته استفاده می‌شوند. این سیستم‌ها معمولاً شامل تکنیک‌های تشخیص ناهنجاری، تحلیل رفتار و تحلیل شبکه هستند که به منظور شناسایی الگوها و تراکنش‌های مشکوک به صورت زمان‌بندی واقعی استفاده می‌شوند.

در کل، پروژه‌های یادگیری ماشین در زمینه تشخیص تقلب به منظور ایجاد سیستم‌های هوشمندی است که بتوانند به طور موثر فعالیت‌های تقلبی را شناسایی و جلوگیری کنند، از زیان‌های مالی جلوگیری کنند و کسب‌وکارها و افراد را در برابر رفتارهای تقلبی محافظت کنند.

---

[۱] Random Forest
[۲]
[۳]
[۴] Neural Networks
[۵] Real-Time

فصل ۴

نتایج جدید

در این فصل به حل مسئله با استفاده از هر سه الگوریتم می‌پردازیم. در سه قسمت، مسئله را به ترتیب با استفاده از آموزش مدل یادگیری رگرسیون منطقی، درخت تصمیم و سپس نیوی بیز حل می‌کنیم. در این فصل فقط به توضیح چگونگی حل مسئله و پیاده‌سازی آن در زبان R می‌پردازیم.

تمامی پیاده‌سازی‌ها و کدها در گیت‌هاب و همچنین فایل‌های ذخیره‌ی پارامترهای ارزش‌گذاری مدل‌ها [۱] بارگذاری شده‌اند و قابل مشاهده می‌باشند.

۴-۱ رگرسیون منطقی

برای پیاده‌سازی مدل رگرسیون منطقی از کتابخانه‌ی caret و تابع glm استفاده می‌شود و می‌توان از تابع predict برای پیش‌بینی مدل بر روی داده‌های آزمون استفاده کرد. همچنین تمام پارامترهای ارزش‌گذاری مدل با استفاده از تابع confusionMatrix استفاده کرد. این پارامترها نیز در فایل LR-ConfusionMatrix.csv ذخیره شده‌اند.

۴-۲ درخت تصمیم

برای پیاده‌سازی مدل درخت تصمیم از کتابخانه‌ی caret و rpart و تابع rpart استفاده می‌شود و می‌توان از تابع predict برای پیش‌بینی مدل بر روی داده‌های آزمون استفاده کرد. همچنین تمام پارامترهای

[۱] https://github.com/kahbodaeini/Fraud-Detection

ارزش‌گذاری مدل با استفاده از تابع confusionMatrix استفاده کرد. این پارامترها نیز در فایل DT-ConfusionMatrix.csv ذخیره شده‌اند.

۳-۴ نیوی بیز

برای پیاده‌سازی مدل نیوی بیز نیز از کتابخانه‌ی caret و e۱۰۷۱ و تابع naiveBayes استفاده می‌شود و می‌توان از تابع predict برای پیش‌بینی مدل بر روی داده‌های آزمون استفاده کرد. همچنین تمام پارامترهای ارزش‌گذاری مدل با استفاده از تابع confusionMatrix استفاده کرد. این پارامترها نیز در فایل NB-ConfusionMatrix.csv ذخیره شده‌اند.

فصل ۵

بررسی نتایج و نتیجه‌گیری

در این فصل با استفاده از نتایج به دست‌آمده در فصل قبل توسط هر الگوریتم را بررسی کرده و با یک دیگر مقایسه می‌کنیم و در بخش آخر به حدس و ایده‌پردازی برای دلیل این تفاوت می‌گردیم.

۵-۱ مقایسه مدل‌ها

در این قسمت به مقایسه‌ی نتایج یه دست آمده توسط سه مدل می‌پردازیم. این مقایسه با پارامترهای مختلفی انجام می‌شود که نشان‌دهنده‌ی ضعف و یا قدرت هر مدل می‌باشد. حال به مرور تعاریف هر پارامتر می‌پردازیم.

- حساسیت [۱]: نسبت تعداد مثبت‌های واقعی راستی‌یافته (True Positive) به تعداد کل مثبت‌های واقعی (True Positive + False Negative) است.

$$\frac{TP}{TP+FN} \tag{۵-۱}$$

- اخصاصیت [۲]: نسبت تعداد منفی‌های واقعی راستی‌یافته (True Negative) به تعداد کل منفی‌های واقعی (True Negative + False Positive) است. همچنین با نام قابلیت تمییزدهی هم شناخته می‌شود.

$$\frac{TN}{TN+FP} \tag{۵-۲}$$

- ارزش پیش‌بینی مثبت [۳]: نسبت تعداد مثبت‌های واقعی راستی‌یافته (True Positive) به تعداد کل

[۱] Sensitivity
[۲] Specificity
[۳] Positive Predictive Value

مثبت‌های پیش‌بینی شده (True Positive + False Positive) است.

$$\frac{TP}{TP+FP} \tag{۳-۵}$$

- ارزش پیش‌بینی منفی [۴]: نسبت تعداد منفی‌های واقعی راستی‌یافته (True Negative) به تعداد کل منفی‌های پیش‌بینی شده (True Negative + False Negative) است.

$$\frac{TN}{TN+FN} \tag{۴-۵}$$

- دقت [۵]: نسبت تعداد مثبت‌های واقعی راستی‌یافته (True Positive) به تعداد کل مثبت‌های پیش‌بینی شده (True Positive + False Positive) است. همچنین با نام ارزش پیش‌بینی مثبت هم شناخته می‌شود.

$$\frac{TP}{TP+FP} \tag{۵-۵}$$

- بازخوانی [۶]: نسبت تعداد مثبت‌های واقعی راستی‌یافته (True Positive) به تعداد کل مثبت‌های واقعی (True Positive + False Negative) است.

$$\frac{TP}{TP+FN} \tag{۶-۵}$$

- امتیاز F1 [۷]: معیاری است که ترکیبی از دقت و بازخوانی مدل را بررسی می‌کند و تعادلی بین دو معیار ایجاد می‌کند.

$$\frac{۲ * Precision * Recall}{Precision + Recall} \tag{۷-۵}$$

- شیوع [۸]: نسبت تعداد مثبت‌های واقعی (True Positive + False Negative) به تعداد کل نمونه‌ها است.

$$\frac{TP+FN}{n} \tag{۸-۵}$$

- نرخ تشخیص [۹]: نسبت تعداد مثبت‌های واقعی راستی‌یافته (True Positive) به تعداد کل مثبت‌های واقعی (True Positive + False Negative) است.

$$\frac{TP}{TP+FN} \tag{۹-۵}$$

[۴] Negative Predictive Value
[۵] Precision
[۶] Recall
[۷] F1 Score
[۸] Prevalence
[۹] Detection Rate

- شیوع تشخیص [۱۰]: نسبت تعداد مثبت‌های پیش‌بینی شده (True Positive + False Positive) به تعداد کل نمونه‌ها است.

$$\frac{TP + FP}{n} \qquad (۱۰-۵)$$

- دقت متوازن [۱۱]: میانگین دقت حساسیت و دقت اخصاصیت است. محاسبه دقت متوازن بر اساس فرمول زیر انجام می‌شود

$$\frac{Sensitivity + Specificity}{۲} \qquad (۱۱-۵)$$

استفاده از این معیارها در ارزیابی عملکرد مدل‌های یادگیری ماشین برای مسائل دسته‌بندی باینری مفید است و به ما اطلاعاتی درباره دقت، قابلیت تشخیص، و قابلیت تمییزدهی مدل را می‌دهد.

لازم به ذکر است، برخی از پارامترهای ذکر شده تعریف یکسانی دارند اما به دلیل شناخته شدن تحت اسامی مختلف، تمام نام‌های معروف برای این پارامترها آورده شد.

حال در جدول زیر، پارامترهای ارزش‌گذاری ذکرشده را برای هر مدل نوشتیم.

---

[۱۰] Detection Prevalence
[۱۱] Balanced Accuracy

جدول ۵-۱: پارامترهای ارزش‌گذاری هر سه الگوریتم یادگیری

Naive Bayes	Decision Tree	Logistic Regression	Parameter
98.285352513937	99.9683449694881	99.9876897103565	Sensitivity
85.7142857142857	78.5714285714286	58.1632653061224	Specificity
99.9749561733033	99.9630710793796	99.9279399616852	Pos Pred Value
7.9320113314449	81.0526315789465	89.0625000000039	Neg Pred Value
99.9749561733033	99.9630710793796	99.9279399616852	Precision
98.285352513937	99.9683449694881	99.9876897103565	Recall
99.1229548175409	99.9657079548752	99.957805907173	F1
99.8279524586998	99.8279524586998	99.8279524586998	Prevalence
98.1162549814786	99.7963518898896	99.8156633486069	Detection Rate
98.1408332016643	99.8332192201682	99.887642422008	Detection Prevalence
91.9998191141114	89.2698867704583	79.0754775082395	Balanced Accuracy

در جدول ۵-۲ نیز به ازای هر پارامتر، مدل یادگیری با بالاترین مقدار را انتخاب می‌کنیم. یعنی مدل‌های ذکرشده به ازای هر پارامتر بهترین عملکرد را داشته‌اند.

همان طور که در این جدول دیدیم رگرسیون منطقی و بیزی نیو به طور کلی عملکرد بهتری نسبت به درخت تصمیم داشتند.

نکته‌ای که باید به آن توجه داشته باشیم این است که ماهیت مسئله‌ی ما چیست و چه پارامترهای ارزش‌گذاری‌ای برای این مسئله اهمیت بالاتری دارند. به طور مثال در مسائل تشخیص بیماری، بیمار تشخیص دادن به اشتباه (FP) هزینه و ضرر کمتری از سالم تشخیص دادن به اشتباه (FN) دارد. این مسئله نیز شباهتی با تشخیص بیماری یا هر حالت غیر دلخواه دیگری دارد. یعنی در این مسائل ما به دنبال یافتن یک رخداد تلخ مانند بیماری یا کلاهبرداری هستیم و پیدا شدن این نمونه‌ها اهمیت بالایی مانند شروع درمان یا جلوگیری از کلاهبرداری بیشتر دارد. پس در این مسئله نیز ما می‌خواهیم تشخیص اشتباه سلامت را به حداقل برسانیم، یعنی پارامترهایی که نشان‌دهنده‌ی کمینه بودن تشخیص سالم بودن محموعه تراکنش‌ها به اشتباه هستند، برای ما هدف است.

جدول ۵-۲: بهترین مدل و نتیجه به ازای هر پارامتر

Best Result	Best Model	Parameter
99.9876897103565	Logistic Regression	Sensitivity
85.7142857142857	Naive Bayes	Specificity
99.9749561733033	Naive Bayes	Pos Pred Value
89.0625000000039	Logistic Regression	Neg Pred Value
99.9749561733033	Naive Bayes	Precision
99.9876897103565	Logistic Regression	Recall
99.9657079548752	Decision Tree	F1
99.8279524586998	Logistic Regression	Prevalence
99.8156633486069	Logistic Regression	Detection Rate
99.887642422008	Logistic Regression	Detection Prevalence
91.9998191141114	Naive Bayes	Balanced Accuracy

۵-۲ حدس دلیل تفاوت و انتخاب مدل برتر

همان طور که دیدیم درخت تصمیم عملکرد مناسبی نسبت به الگوریتم‌های رگرسیون منطقی و بیزی نیو نداشت. درخت تصمیم بر اساس هر ستون محموعه داده یا همان ویژگی نمونه، داده‌ها را به دو قسمت تقسیم می‌کند و هر بار با استفاده از یک ویژگی دیگر این کار را تکرار می‌کند تا به عمق دلخواه درخت یا شرایط بازداری از ادامه الگوریتم برسد. اما نکته‌ای که باید به آن توجه کرد این است که در این الگوریتم هر ستون مجموعه داده به صورت یک ویژگی مستقل از نمونه شناخته می‌شود. یعنی به طور مثال می‌توان از این الگوریتم برای کلاسه‌بندی قیمت خانه یا ماشین استفاده کرد و هر ستون مجموعه داده یک ویژگی خاص نمونه‌ها مانند تعداد اتاق خواب یا متراژ ساختمان خانه باشد. در این صورت ماهیت هر ویژگی متفاوت است و می‌توان بر اساس آن داده‌ها را جدا کرد، اما در این مسئله، ماهیت هر ستون مجموعه داده یکسان است و هر نمونه صرفا توالی تراکنش‌های بانکی می‌باشد که نمی‌توان ماهیت آن را از دیگر ستون‌ها متفاوت دانست و نتیجه‌گیری و پیش‌بینی مدل باید بر اساس مجموعه و توالی این تراکنش‌ها باشد و نمی‌توان بر اساس یک تراکنش (ستون) تقسیم‌بندی مناسبی بر مجموعه داده انجام داد.

دو الگوریتم رگرسیون منطقی و بیزی نیو نیز عملکرد بسیار نزدیک و مناسبی داشتند و نمی‌توان برتری ثابتی برای یک الگوریتم نسبت به دیگری در نظر بگیریم و بهتر است که مناسب بودن الگوریتم‌ها را برای

این مسئله‌ی خاص در نظر بگیریم. اما با توجه به پارامترهایی که هر کدام از این دو الگوریتم در آن برتر بودند، مانند Precision برای نیوی بیز و Prevalence Detection برای رگرسیون منطقی می‌توانیم به این نتیجه برسیم که رگرسیون منطقی روشی است که به طور کلی دقیق‌تر نمونه‌ها را تشخیص می‌دهد اما نیوی بیز محتاط‌تر است و ممکن است به اشتباه توالی‌های سالم را کلاهبرداری تشخیص دهد اما احتمال این که یک کلاهبرداری را سالم تشخیص دهد کمتر می‌باشد و به نظر می‌رسد که مدل مناسب‌تری برای این مسئله باشد. شایان ذکر است که این دو الگوریتم عملکرد بسیار مشابه و نزدیکی داشته‌اند و تفاوت‌های به دست‌آمده را نمی‌توان چشم‌گیر و موثر دانست و در نهایت می‌توان برای حل این مسئله به طور سلیقه‌ای و یا با تشخیص مهندس یادگیری ماشین از هر دوی این الگوریتم‌ها به صورت تکی یا ترکیبی استفاده کرد.

واژه‌نامه

الف

اخصاصیت ..... Specificity

آزمون ..... Test

آموزش ..... Train

اندازه‌ی افزایش اطلاعات ..... Information Gain

انتروپی ..... Entropy

ب

بازخوانی ..... Recall

ت

تابع هزینه ..... Cost Function

ج

جنگل تصادفی ..... Random Forest

ح

حساسیت ..... Sensitivity

د

داده ..... Data

دقت ..... Precision

دقت متوازن ..... Balanced Accuracy

ر

رگرسیون منطقی ..... Logistic Regression

ز

زمان‌بندی واقعی ..... Real-Time

ش

شبکه‌ی عصبی ..... Neural Network

شیوع ..... Prevalence

ک

کلاسه‌بندی ..... Classification

گ

گرادیان کاهشی ..... Gradient Descent

م

مجموعه داده ..... Dataset

ن

نرخ تشخیص ..... Detection Rate
نرخ شیوع ..... Prevalence Rate
نرمال‌سازی ..... Normalization
ناخالصی ..... Impurity
ناخالصی جینی ..... Gene Impurity
نیوی بیز ..... Naive Bayes

پیوست آ

مطالب تکمیلی

Cox, D. R. (1958). The regression analysis of binary sequences. Journal of the Royal Statistical Society: Series B (Methodological), 20(2), 215–232.

Wu, X., Kumar, V., Quinlan, J. R., Ghosh, J., Yang, Q., Motoda, H., … others. (2008). Top 10 algorithms in data mining. Knowledge and Information Systems, 14(1), 1–37.

TY - JOUR AU - Rish, Irina PY - 2001/01/01 SP - T1 - An Empirical Study of the Naïve Bayes Classifier VL - 3 JO - IJCAI 2001 Work Empir Methods Artif Intell ER -

McCulloch, W. S., Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. The Bulletin of Mathematical Biophysics, 5(4), 115–133.

Ho, T. K. (1995). Random decision forests. In Proceedings of 3rd international conference on document analysis and recognition (Vol. 1, pp. 278–282).

Abstract

Today, various machine learning methods have been introduced, especially for tabular datasets with labeled data. These methods have their own advantages and disadvantages when compared to each other. The aim of this project is primarily to solve a real-world problem, which is detecting fraud from credit card sequence of transactions, using selected methods and then compare three different methods based on the results obtained from solving the problem using each of these methods, in order to draw logical conclusions. The three selected methods for investigation in this project are Logistic Regression, Decision Tree, and naive Bayes machine learning algorithms. These algor thms have high usage in machine learning field. In the end, we conclude the priority of Naive Bayes and Logistic Regression to Decision Tree and explore the reasons behind that in the last chapter.

Keywords: Supervised Learning, Dataset, Logistic Regression, Decision Tree, Naive Bayes, Fraud Detection, Validation of a Machine Learning Model

Sharif University of Technology

Department of Computer Engineering

B.Sc. Thesis

Fraud Detection Based on Credit Card Sequence of Transactions using Various Machine Learning Algorithms

By:

Kahbod Aeini

Supervisor:

Pr. Ali Mohammad Afshin Hemmatyar

June 2023