بررسی روش‌های یادگیری ماشین در تشخیص تقلب مالی

نام نویسنده: امیرحسین رنجبر

چکیده

تشخیص تراکنش‌های مالی تقلبی یکی از مسائل مهم در حوزه یادگیری ماشین است. نامتوازن بودن داده‌ها، تعداد بسیار کم نمونه‌های تقلبی و هزینه بالای تشخیص نادرست از جمله چالش‌های اصلی این مسئله هستند.

در این گزارش، ابتدا چند مدل پایه شامل درخت تصمیم، KNN و رگرسیون لجستیک بررسی می‌شوند. سپس مدل‌های پیشرفته‌تر شامل XGBoost و جنگل تصادفی مورد بررسی قرار می‌گیرند. همچنین KNN به همراه XGBoost و جنگل تصادفی در یک روش Ensemble ترکیب می‌شود.

عملکرد مدل‌ها با استفاده از معیارهایی مانند Precision، Recall، F۱-score و PR-AUC ارزیابی می‌شود. همچنین برای انتخاب تنظیمات مناسب مدل‌ها از روش‌های اعتبارسنجی متقابل و بهینه‌سازی هایپرپارامترها استفاده می‌شود.

واژگان کلیدی: یادگیری ماشین، تشخیص تقلب، تراکنش مالی، درخت تصمیم، KNN، رگرسیون لجستیک، جنگل تصادفی، XG-Boost، Ensemble، داده‌های نامتوازن

۱. مقدمه

افزایش استفاده از سامانه‌های پرداخت الکترونیکی باعث شده است که حجم بسیار زیادی از تراکنش‌های مالی در هر روز ثبت شود. در کنار این افزایش، احتمال وقوع تراکنش‌های تقلبی نیز وجود دارد.

تشخیص خودکار تراکنش‌های تقلبی می‌تواند نقش مهمی در کاهش خسارت‌های مالی داشته باشد. با این حال، مسئله تشخیص تقلب معمولاً با یک مشکل مهم یعنی نامتوازن بودن کلاس‌ها مواجه است.

در بسیاری از مجموعه‌داده‌های واقعی، تعداد تراکنش‌های سالم بسیار بیشتر از تراکنش‌های تقلبی است. بنابراین یک مدل ممکن است دقت بسیار بالایی داشته باشد، اما در تشخیص نمونه‌های تقلبی عملکرد مناسبی نداشته باشد.

به همین دلیل، در این مسئله معیارهایی مانند Precision، Recall، F۱-score و PR-AUC اهمیت زیادی پیدا می‌کنند.

در این پژوهش، ابتدا مدل‌های پایه بررسی شده و سپس مدل‌های پیشرفته‌تر و در نهایت روش Ensemble مورد استفاده قرار گرفته‌اند.

۲. معرفی داده‌ها

در این پژوهش از یک مجموعه‌داده شامل تراکنش‌های مالی استفاده شده است. هر نمونه شامل تعدادی ویژگی مربوط به تراکنش و یک برچسب دودویی است.

برچسب صفر نشان‌دهنده تراکنش سالم و برچسب یک نشان‌دهنده تراکنش تقلبی است.

یکی از ویژگی‌های مهم داده، عدم توازن شدید بین دو کلاس است. به همین دلیل، استفاده از Accuracy به‌تنهایی نمی‌تواند معیار مناسبی برای ارزیابی مدل باشد.

داده‌ها پس از انجام مراحل پیش‌پردازش به دو مجموعه آموزش و آزمون تقسیم شدند.

۳. پیش‌پردازش داده‌ها

پیش از آموزش مدل‌ها، داده‌ها بررسی و آماده‌سازی شدند. مراحل پیش‌پردازش شامل بررسی مقادیر گمشده، داده‌های تکراری و آماده‌سازی ویژگی‌ها برای ورود به مدل است.

همان‌طور که در بررسی اولیه داده‌ها مشاهده شد، مجموعه‌داده دارای تعدادی مقدار تکراری بود. از آنجا که وجود داده‌های تکراری می‌تواند بر فرآیند آموزش مدل تأثیر بگذارد، این نمونه‌ها شناسایی و حذف شدند.

تعداد نمونه‌ها و تراکنش‌های تقلبی قبل و بعد از حذف داده‌های تکراری در جدول زیر ارائه شده است.

Data	Before	After
Total transactions	284807	284315
Fraudulent transactions	492	473

جدول ۱: مقایسه داده‌ها قبل و بعد از حذف نمونه‌های تکراری

همان‌طور که مشاهده می‌شود، تعداد کل تراکنش‌ها از ۲۸۴۸۰۷ به ۲۸۴۳۱۵ کاهش یافته است. همچنین تعداد تراکنش‌های تقلبی از ۴۹۲ نمونه به ۴۷۳ نمونه رسیده است. بنابراین حذف نمونه‌های تکراری باعث حذف بخشی از تراکنش‌های تقلبی نیز شده است.

داده‌ها به دو مجموعه آموزش و آزمون تقسیم شدند:

$$D = D_{\text{train}} \cup D_{\text{test}}$$

به‌طوری‌که مجموعه آزمون در فرآیند آموزش مدل استفاده نمی‌شود.

در صورت وجود ویژگی‌هایی با مقیاس‌های متفاوت، می‌توان از روش‌های استانداردسازی برای آماده‌سازی داده‌ها استفاده کرد. استانداردسازی به‌خصوص برای مدل‌هایی مانند KNN و رگرسیون لجستیک اهمیت دارد.

۴. مدل‌های پایه

در مرحله اول پژوهش، سه مدل پایه برای ایجاد یک معیار اولیه از عملکرد سیستم مورد بررسی قرار گرفتند:

- درخت تصمیم
- KNN
- رگرسیون لجستیک

هدف از استفاده از مدل‌های پایه، بررسی عملکرد الگوریتم‌های ساده‌تر و ایجاد یک مبنا برای مقایسه با مدل‌های پیشرفته‌تر است.

۱.۴. درخت تصمیم

درخت تصمیم داده‌ها را با استفاده از مجموعه‌ای از قوانین شرطی به بخش‌های مختلف تقسیم می‌کند. هدف الگوریتم ایجاد تقسیم‌هایی است که نمونه‌های متعلق به کلاس‌های مختلف را تا حد امکان از یکدیگر جدا کنند.

یکی از پارامترهای مهم درخت تصمیم، عمق درخت است. افزایش بیش از حد عمق می‌تواند باعث بیش‌برازش شود. از طرف دیگر، عمق بسیار کم نیز ممکن است باعث کم‌برازش مدل شود.

برای انتخاب مناسب‌ترین تنظیمات مدل، در این پژوهش از روش Search Grid استفاده شد. در این روش، مجموعه‌ای از مقادیر مختلف برای هایپرپارامترهای مدل تعریف شده و ترکیب‌های مختلف آن‌ها به صورت سیستماتیک بررسی شدند.

پارامترهایی مانند عمق درخت (max_depth)، حداقل تعداد نمونه مورد نیاز برای تقسیم یک گره (min_samples_split) و حداقل تعداد نمونه در هر برگ (min_samples_leaf) مورد بررسی قرار گرفتند.

برای ارزیابی ترکیب‌های مختلف از اعتبارسنجی متقابل Strat-ified K-Fold استفاده شد تا نسبت کلاس‌ها در بخش‌های مختلف داده تا حد امکان حفظ شود.

معیار F۱-score به عنوان معیار اصلی انتخاب مدل در نظر گرفته شد؛ زیرا در مسئله تشخیص تقلب، توجه هم‌زمان به Pre-cision و Recall اهمیت زیادی دارد.

۲.۴. K نزدیک‌ترین همسایه

الگوریتم KNN یکی از روش‌های مبتنی بر فاصله است که برای طبقه‌بندی یک نمونه جدید، نزدیک‌ترین نمونه‌های موجود در داده‌های آموزشی را بررسی می‌کند.

در این روش، پارامتر K تعداد همسایه‌هایی را مشخص می‌کند که در تصمیم‌گیری مورد استفاده قرار می‌گیرند.

برای مثال، اگر $K = ۵$ باشد، پنج نمونه نزدیک‌تر به نمونه مورد نظر بررسی شده و بر اساس کلاس آن‌ها تصمیم‌گیری می‌شود.

از آنجا که KNN به فاصله بین نمونه‌ها وابسته است، استانداردسازی ویژگی‌ها اهمیت زیادی دارد.

در این پژوهش مقادیر مختلف K بررسی شدند و عملکرد مدل با استفاده از Recall Precision، و F۱score- مقایسه شد.

۳.۴. رگرسیون لجستیک

رگرسیون لجستیک یکی از روش‌های کلاسیک برای مسائل طبقه‌بندی دودویی است. این مدل ابتدا ترکیب خطی ویژگی‌ها را محاسبه کرده و سپس با استفاده از تابع سیگموید احتمال تعلق نمونه به کلاس مثبت را به دست می‌آورد.

تابع سیگموید به صورت زیر تعریف می‌شود:

$$\sigma(z) = \frac{۱}{۱ + e^{-z}}$$

که در آن:

$$z = w^T x + b$$

است.

خروجی مدل در بازه صفر تا یک قرار دارد و می‌توان آن را به عنوان احتمال تعلق نمونه به کلاس تقلب در نظر گرفت.

در این پژوهش برای مقابله با عدم توازن کلاس‌ها از class_weight="balanced" استفاده شد و برای جلوگیری از اثر تفاوت مقیاس ویژگی‌ها، استانداردسازی داده‌ها نیز انجام گرفت.

۵. مدل‌های پیشرفته

پس از بررسی مدل‌های پایه، مدل‌های قدرتمندتر مبتنی بر درخت تصمیم بررسی شدند.

در این مرحله مدل‌های زیر مورد استفاده قرار گرفتند:

- XGBoost
- Forest Random
- KNN

KNN در این مرحله نیز مورد استفاده قرار گرفت تا مشخص شود آیا ترکیب آن با مدل‌های درختی می‌تواند عملکرد Ensemble را بهبود دهد.

۱.۵. جنگل تصادفی

جنگل تصادفی از تعداد زیادی درخت تصمیم تشکیل شده است. هر درخت روی نمونه‌ها و ویژگی‌های متفاوتی آموزش داده می‌شود و در نهایت خروجی درخت‌ها با یکدیگر ترکیب می‌شوند.

احتمال تعلق یک نمونه به کلاس تقلب را می‌توان به صورت زیر در نظر گرفت:

$$P(y = ۱|x) = \frac{۱}{T} \sum_{t=۱}^{T} P_t(y = ۱|x)$$

که در آن T تعداد درخت‌ها است.

مزیت مهم جنگل تصادفی کاهش واریانس و کاهش احتمال بیش‌برازش نسبت به یک درخت تصمیم منفرد است.

در این پژوهش پارامترهایی مانند تعداد درخت‌ها، max_depth، min_samples_split، min_samples_leaf و max_features مورد بررسی قرار گرفتند.

برای انتخاب تنظیمات مناسب، از Cross- K-Fold Stratified Validation و معیار F۱score- استفاده شد.

۲.۵. XGBoost

XGBoost یکی از الگوریتم‌های قدرتمند مبتنی بر روش Gradi-ent Boosting است که از مجموعه‌ای از درخت‌های تصمیم برای ایجاد یک مدل نهایی استفاده می‌کند.

برخلاف روش‌هایی مانند جنگل تصادفی که درخت‌ها را به صورت مستقل آموزش می‌دهند، در XGBoost درخت‌ها به صورت ترتیبی ساخته می‌شوند. هر درخت جدید تلاش می‌کند خطاهای مدل‌های قبلی را کاهش دهد.

تابع کلی مدل XGBoost را می‌توان به صورت زیر نمایش داد:

$$\hat{y}_i = \sum_{k=۱}^{K} f_k(x_i)$$

که در آن K تعداد درخت‌های موجود در مدل و f_k تابع مربوط به درخت تصمیم k-ام است.

تابع هدف XGBoost شامل تابع خطا و جمله منظم‌سازی است:

$$\mathcal{L} = \sum_{i=۱}^{n} l(y_i, \hat{y}_i) + \sum_{k=۱}^{K} \Omega(f_k)$$

وجود جمله منظم‌سازی به کنترل پیچیدگی مدل و کاهش احتمال بیش‌برازش کمک می‌کند.

از پارامترهای مهم XGBoost می‌توان به تعداد درخت‌ها (n_estimators)، عمق درخت (max_depth) و نرخ یادگیری (learning_rate) اشاره کرد.

از آنجا که مجموعه‌داده دارای عدم توازن شدید بین کلاس سالم و تقلبی است، از پارامتر scale_pos_weight نیز استفاده شد.

مقدار این پارامتر به صورت زیر تعیین شد:

$$\text{scale_pos_weight} = \frac{N_۰}{N_۱}$$

که در آن $N_۰$ تعداد نمونه‌های سالم و $N_۱$ تعداد نمونه‌های تقلبی است.

افزایش وزن کلاس اقلیت باعث می‌شود مدل اهمیت بیشتری برای شناسایی تراکنش‌های تقلبی قائل شود، هرچند ممکن است باعث افزایش مثبت‌های کاذب شود.

۶. عدم توازن کلاس‌ها

در مسئله تشخیص تقلب، تعداد نمونه‌های کلاس تقلب معمولاً بسیار کمتر از نمونه‌های سالم است.

فرض کنیم:

$$N_۰ \gg N_۱$$

که در آن $N_۰$ تعداد نمونه‌های سالم و $N_۱$ تعداد نمونه‌های تقلبی است.

در چنین شرایطی، یک مدل ممکن است تقریباً تمام نمونه‌ها را سالم پیش‌بینی کند و همچنان Accuracy بسیار بالایی به دست آورد.

بنابراین معیارهای دیگری مانند Recall Precision، و F۱-score اهمیت بیشتری پیدا می‌کنند.

Precision به صورت زیر تعریف می‌شود:

$$\text{Precision} = \frac{TP}{TP + FP}$$

Recall نیز برابر است با:

$$\text{Recall} = \frac{TP}{TP + FN}$$

و F۱-score به صورت زیر محاسبه می‌شود:

$$F۱ = ۲\frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

۷. اعتبارسنجی و بهینه‌سازی

برای جلوگیری از وابستگی بیش از حد انتخاب مدل به یک مجموعه Validation، از Cross-Validation K-Fold Stratified استفاده شد.

در این روش داده‌های آموزشی به پنج بخش تقسیم می‌شوند و مدل در پنج مرحله آموزش داده می‌شود. در هر مرحله، چهار بخش برای آموزش و یک بخش برای اعتبارسنجی استفاده می‌شود.

همچنین برای انتخاب هایپرپارامترهای مناسب برخی مدل‌ها از Search Grid استفاده شد.

معیار اصلی انتخاب مدل F۱-score در نظر گرفته شد؛ زیرا این معیار تعادل میان Precision و Recall را در نظر می‌گیرد.

به صورت کلی:

$$\theta^* = \arg\max_{\theta \in \Theta} F۱(\theta)$$

که در آن Θ مجموعه ترکیب‌های مختلف هایپرپارامترها است.

۸. روش Ensemble

پس از بررسی مدل‌های مختلف، برای بهبود عملکرد از ترکیب XGBoost، Random Forest و KNN استفاده شد.

برای هر نمونه، هر مدل احتمال تعلق نمونه به کلاس تقلب را تولید می‌کند. سپس این احتمال‌ها با وزن‌های مختلف ترکیب می‌شوند:

$$P_{\text{ensemble}} = w_{\text{XGB}} P_{\text{XGB}} + w_{\text{RF}} P_{\text{RF}} + w_{\text{KNN}} P_{\text{KNN}}$$

به طوری که:

$$w_{\text{XGB}} + w_{\text{RF}} + w_{\text{KNN}} = ۱$$

در این پژوهش وزن مدل‌ها به صورت دستی ثابت انتخاب نشدند، بلکه ترکیب‌های مختلف وزن‌ها بررسی شدند.

۱.۸. Predictions Out-of-Fold

برای انتخاب وزن‌های Ensemble از Out-of-Fold Predictions استفاده شد.

در روش پنج‌گانه Cross-Validation، در هر مرحله مدل روی چهار Fold آموزش داده شده و روی Fold باقی‌مانده پیش‌بینی می‌کند.

در نتیجه برای هر نمونه یک پیش‌بینی وجود خواهد داشت که مدل تولیدکننده آن نمونه را در فرآیند آموزش خود مشاهده نکرده است.

این پیش‌بینی‌ها برای هر سه مدل ذخیره شدند:

$$P_{\text{XGB}}^{OOF}, \qquad P_{\text{RF}}^{OOF}, \qquad P_{\text{KNN}}^{OOF}$$

سپس از این پیش‌بینی‌ها برای انتخاب وزن مناسب مدل‌ها استفاده شد.

۲.۸. انتخاب آستانه

پس از ترکیب احتمال مدل‌ها، برای تبدیل احتمال نهایی به برچسب کلاس از یک آستانه استفاده می‌شود:

$$\hat{y} = \begin{cases} ۱ & P_{\text{ensemble}} \geq \tau \\ ۰ & P_{\text{ensemble}} < \tau \end{cases}$$

تغییر مقدار τ باعث ایجاد تغییر در Precision و Recall می‌شود.

در این پژوهش هدف انتخاب آستانه‌ای است که:

$$\text{Precision} \geq ۰/۹۰$$

و

$$\text{Recall} \geq ۰/۸۰$$

باشد.

در میان تمام ترکیب‌هایی که این دو شرط را برآورده می‌کنند، ترکیبی انتخاب می‌شود که بیشترین F۱-score را داشته باشد.

۹. ارزیابی مدل‌ها

برای مقایسه مدل‌ها از ماتریس درهم‌ریختگی و معیارهای Precision، Recall، F۱-score و PR-AUC استفاده می‌شود.

ماتریس درهم‌ریختگی به شکل زیر است:

$$\begin{bmatrix} TN & FP \\ FN & TP \end{bmatrix}$$

در مسئله تشخیص تقلب، مقدار FN اهمیت زیادی دارد؛ زیرا هر FN نشان‌دهنده یک تراکنش تقلبی است که توسط مدل به عنوان تراکنش سالم شناسایی شده است.

از طرف دیگر، مقدار FP نیز اهمیت دارد؛ زیرا نشان‌دهنده تراکنش سالمی است که به اشتباه به عنوان تقلب شناسایی شده است.

بنابراین در این پژوهش تلاش می‌شود میان Precision و Recall تعادل مناسبی ایجاد شود.

۱۰. نتایج

در این بخش نتایج حاصل از اجرای مدل‌های مختلف بررسی می‌شوند.

۱.۱۰. نتایج مدل‌های پایه

در مرحله اول، عملکرد مدل‌های پایه شامل درخت تصمیم، KNN و رگرسیون لجستیک بررسی شد.

معیارهای Precision، Recall و F۱-score برای کلاس تقلب به عنوان معیارهای اصلی مقایسه در نظر گرفته شدند.

نتایج واقعی مدل‌ها پس از اجرای آزمایش‌ها در جدول زیر قرار می‌گیرند.

Model	Precision	Recall	F1	PR-AUC
Decision Tree	–	–	–	–
KNN	–	–	–	–
Logistic Regression	–	–	–	–

جدول ۲: نتایج مدل‌های پایه

۲.۱۰. نتایج مدل‌های پیشرفته

در مرحله بعد، مدل‌های XGBoost و Random Forest بررسی شدند. همچنین KNN به عنوان یکی از مدل‌های مورد استفاده در Ensemble در این مرحله حفظ شد.

Model	Precision	Recall	F1	PR-AUC
XGBoost	–	–	–	–
Random Forest	–	–	–	–
KNN	–	–	–	–

جدول ۳: نتایج مدل‌های پیشرفته

۳.۱۰. نتایج Ensemble

در مرحله نهایی، احتمال‌های تولیدشده توسط XGBoost، Random Forest و KNN با وزن‌های بهینه ترکیب شدند.

وزن‌ها و آستانه نهایی با استفاده از پیش‌بینی‌های Out-of-Fold انتخاب شدند.

هدف انتخاب ترکیبی بود که Precision حداقل ۹۰ درصد و Recall حداقل ۸۰ درصد داشته باشد و در میان این ترکیب‌ها بیشترین F۱score- را ایجاد کند.

Parameter	Value
XGBoost Weight	–
Random Forest Weight	–
KNN Weight	–
Threshold	–
Precision	–
Recall	–
F1-score	–
PR-AUC	–

جدول ۴: نتایج Ensemble نهایی

۱۱. بحث

نتایج نشان می‌دهد که ارزیابی مدل در مسائل نامتوازن باید فراتر از Accuracy انجام شود.

در این مسئله، Precision و Recall هر دو اهمیت دارند. Precision بالا باعث کاهش تعداد تراکنش‌های سالمی می‌شود که به اشتباه به عنوان تقلب شناسایی شده‌اند.

از طرف دیگر، Recall بالا باعث می‌شود تعداد بیشتری از تراکنش‌های تقلبی شناسایی شوند.

به همین دلیل، در مرحله Ensemble هدف تنها افزایش Recall نبوده است، بلکه تلاش شده است Precision نیز در سطح مناسبی حفظ شود.

استفاده از Out-of-Fold Predictions باعث می‌شود انتخاب وزن‌های Ensemble و آستانه تصمیم بر اساس پیش‌بینی‌هایی انجام شود که مدل مربوطه نمونه مورد نظر را در زمان آموزش مشاهده نکرده است.

۱۲. نتیجه‌گیری

در این گزارش، مسئله تشخیص تراکنش‌های تقلبی با استفاده از روش‌های مختلف یادگیری ماشین بررسی شد.

در مرحله اول، مدل‌های پایه شامل درخت تصمیم، KNN و رگرسیون لجستیک مورد بررسی قرار گرفتند.

در مرحله بعد، مدل‌های پیشرفته‌تر شامل XGBoost و Random Forest مورد استفاده قرار گرفتند.

در نهایت، XGBoost، Random Forest و KNN در قالب یک روش Ensemble ترکیب شدند. برای انتخاب وزن مدل‌ها از Out-of-Fold Predictions حاصل از Stratified K-Fold Cross-Validation استفاده شد.

همچنین Threshold تصمیم به عنوان یکی از پارامترهای مهم سیستم بررسی شد؛ زیرا تغییر آن می‌تواند تعادل میان Precision و Recall را تغییر دهد.

هدف نهایی انتخاب ترکیبی بود که Precision حداقل ۹۰ درصد و Recall حداقل ۸۰ درصد داشته باشد و در میان ترکیب‌های مجاز، بیشترین F۱score- را ایجاد کند.

در نتیجه، استفاده هم‌زمان از مدل‌های مختلف، اعتبارسنجی متقابل، Out-of-Fold Predictions و انتخاب مناسب Threshold می‌تواند چارچوب مناسبی برای توسعه یک سیستم تشخیص تقلب بر پایه یادگیری ماشین فراهم کند.

مراجع

[۱] Breiman, L. "Random Forests," Machine Learning, ۲۰۰۱.

[۲] T. Chen and C. Guestrin, "XGBoost: A Scalable Tree Boosting System," Proceedings of the ۲۲nd ACM SIGKDD, ۲۰۱۶.

[۳] J. Han, M. Kamber and J. Pei, Data Mining: Concepts and Techniques.