**بسم الله الرحمن الرحیم**

**پیش بینی وضعیت وام بانکی**

**سارا سلطانی گردفرامرزی**

**چکیده**

صنعت بانکداری همیشه برای بسیاری از مسائل به سیستم مدلسازی پیش بینی نیاز دارد. ما از وضعیت وام، مبلغ وام، میزان بهره، ... برای ایجاد یک مدل پیش‌بینی استفاده می‌کنیم. مدل برای تجزیه و تحلیل دقیق داده های اعتباری برای یافتن ناتوانان و مشتریان معتبر استفاده می شود. هدف این مقاله ایجاد یک مدل پیش‌بینی برای تعیین وضعیت وام است. تکنیک های مختلف یادگیری ماشین برای توسعه مدل استفاده می شود. ما از ترکیب نرمال سازی و طبقه بندی کننده( K-NN) و الگوریتم‌های یادگیری نیز استفاده می کنیم. هدف با استفاده از بسته نرم افزاری Jupyter notebook پیاده سازی شده است.

واژگان کلیدی: وضعیت وام، الگوریتم یادگیری، داده‌ها

**۱) مقدمه**

بازگشت وام یکی از مهمترین مشکلاتی است که در صنعت بانکداری باید مورد توجه قرار می‌گیرد. ریسک اعتباری به عنوان خطری تعریف می‌شود که وام گیرندگان در انجام تعهدات وام خود شکست می‌خورند. مدل ما برای پیش‌بینی توانایی پرداخت وام و کاهش فعالیت‌های غیرقانونی استفاده می‌شود. این مدل برای تصمیم گیری از اطلاعات وام گیرندگان استفاده می‌کند. برای اتخاذ تصمیمات وام، وام دهندگان می‌خواهند ریسک هر تصمیم وام‌دهی را به حداقل برسانند. به طور کلی، موفقیت و شکست صنعت بانکداری مبتنی بر ریسک اعتباری آن‌ها است و اگر مبلغ اعتبار به درستی جمع‌ آوری نشود، بانک ضرر خواهد کرد. بنابراین، سود بانکی با ریسک اعتباری آن‌ها مرتبط است و داده‌ها از سایت kaggle برای موضوع Bank Loan Defaulter Prediction (Hackathon) بدست آمده‌است.

بنابراین، هدف این مقاله فرآیندی شامل طبقه‌بندی، خوشه‌بندی و استخراج قوانین و پیش‌بینی است نتایج این مطالعه برای وام دهندگان در تصمیم‌گیری وام بسیار مفید خواهد بود. بقیه مقاله به شرح زیر سازماندهی شده است. در بخش ۲۳۴۵۵

**۲) آماده‌سازی داده**

در ایتدا برای اینکه بتوانیم مدل‌های مختلف را آموزش دهیم و سپس بین آن‌ها ارزیابی انجام دهیم مجوعه داده را به سه بخش آزمایش، ارزیابی و تست تقسیم کردیم که نسبت این بخش‌ها به کل داده‌ها به صورت ۸۰٪ داده برای آموزش، ۲۰٪ تست استو همچنین برای ارزیابی با دقت بالاتر توزیع داده‌های هدف در مجموعه validation و train یکسان است.

سپس مشاهده کردیم در بین ویژگی‌های مربوط به هر درخواست‌کننده‌ی وام ویژگی‌های ID دارای مقدار یکتا برای هر درخواست‌کننده و Payment Plan دارای مقدار یکسان برای همه‌ی درخواست‌کننده‌ها هست درنتیجه این دو ويژگی را حذف کردیم.

سپس سعی کردیم داده‌های آموزشی را به دلیل توزیع نامتوازن بین دو کلاس (تایید وام : ٪۶ و رد وام: ٪۹۴) متعادل کنیم بنابراین از روش up- sample برای افزایش تعداد داده‌ی کلاس کم‌ جمعیت‌‌تر استفاده کردیم و توزیع داد‌ه‌ها را ( تایید وام) ٪۱۷ به (رد وام) ۸۳٪ تبدیل کردیم البته برای متوازن‌سازی داده‌ها از روش near miss هم استفاده کردیم اما نتیجه بهتری دریافت نشد و کمتر از جواب پایه خروجی داد.

در بین داد‌ه‌ها هیچ داد‌های تکراری و گم‌شده‌ای وجود نداشت و برای شناسایی داده‌های پرت از سه روش IQR, Zscore و Percentile Method استفاده کردیم ولی به دلیل اینکه روش IQR مقدار داده‌ي زیادی را داده‌ي پرت تشخیص می‌داد و همچنین همبستگی بین ویژگی‌ها را هم کاهش می‌داد از روش Zscore درنهایت برای تشخیص داده‌های پرت استفاده کردیم، راه‌های مختلفی برای مقابله با داده‌ی پرت وجود دارد. برخی از آن عبارتند از: حذف موارد پرت، Capping & Flooring و ... ولی از آنجایی که مجموعه داده ما بزرگ نیست و حذف داده‌های پرت ممکن است کیفیت داده‌ها را کم کند ما با 5% و 95% به سراغ Capping & Flooring می رویم (یعنی داده‌هایی که کمتر از ٪5 داده‌های موجود و بیشتر از ٪95 داده‌های موجود باشد جایگزین ٪5 و ٪95 می‌شوند)

سپس ویژگی‌ Sub grade را طبقه‌بندی مجدد کردیم تا همه‌ی مقادیرG1,G2,…,G5 و ... و A1,A2,..,A5 در دسته‌ی Gو .... و A قرار بگیرند و در مرحله‌ی بعد ویژگی‌های طبقه‌بندی‌شده‌ را با روش encodingبه باتوجه به نوعشان (ترتیبی یا اسمی) به ویژگی عددی تبدیل کردیم

در مرحله‌ی استاندارسازی داده‌ها از دو روش Zscore و min-max استفاده کردیم که توزیع داده‌ها در هر دو روش بسیار مشابه است و تفاوتی نمی‌کند، ما از روش min-max استفاده کردیم

متغیرهای مقدار وام و با استفاده از دو روش زیر سبدبندی کردیم

۱ـ KBinsDiscretizer (دسته های سه یا چهار یا 5 تایی)

۲ـ Qcut (دسته های 3 یا 4 یا 5 تایی)

در هر دو روش مقدار هم‌بستگی(correlation و mutual information) تغییری نکرد و درنتیجه از سبدبندی برای داده‌های عددی استفاده نکردیم

**۳)تحلیل اکتشافی**

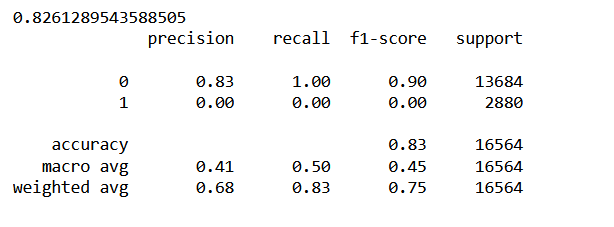
چون همبستگی بین ویژگی‌ها کم بود ما هیچ کدام از ویژگی‌ها را حذف نکردیم و برای قسمت استخراج متغیرهای جدید براساس ترکیب متغیرهای موجود

ابتدا همبستگی بین ویژگی‌ها استاندار شده با روش zscore را محاسبه کردیم و هفت ویژگی که همبستگی بیشتری نسبت به بقیه داشتند را انتخاب کردیم و سپس میانگین آن ویژگی‌ها را روی هر دو کلاس گرفتیم و سپس چهار ویژگی که هم‌نوایی یکسانی روی کلاس‌ها داشتند را ترکیب خطی کردیم ولی هم‌بستگی بین ویژگی جدید و هدف افزایش نیافت در نتیجه از ویژگی جدید استفاده نکردیم

# هم‌چنین برای اینکه رابطه‌ی بین ویژگی‌ها و وضعیت وام را تحلیل کنیم برای هر جفت ویژگی و وضعیت وام نمودار overlay را کشیدیم اما ارتباطی بین آن‌ها وجود نداشت

**۴) آزمایشات فاز مدل سازی**

جواب پایه با DummyClassifier و استراتژی most\_frequent(بیشترین فراوانی بین کلاس‌ها را برمی‌گرداند) انجام شده که جواب تقریبا 5 .82 درصد داده است.



11

برای آموزش مدل از الگوریتم‌های, random forest classifier knn ,mlp, naïve bayes, xgboost استفاده کردیم که الگوریتم‌های, random forest classifier knn ,mlp, naïve bayes دقت بهتری نسبت به جواب پایه در مجموعه داده‌های ارزیابی بدست نیاوردند اما الگوریتم xgboost در مجموعه داده‌ی ارزیابی دقت بهتری نسبت به جواب پایه بدست آورد ولی این الگوریتم هم روی مجموعه‌ی تست نتیجه‌ای بهتر از جواب پایه بدست نمی‌آورد

در قسمت تنظیم بهینه مدل و هایپر پارامترها‌، همه‌ی مدل‌های ما بعد از تنظیم کردن هایپر پارامترهایش با تابع RandomizedSearchCV دقت بالاتری روی داده آموزش بدست می‌آورند ولی با این وجود روی داده‌ی تست دقت بالاتر از جواب پایه ندارند

در اینجا مجموعه معیارهای ارزیابی برای این مدل‌هایمان را معرفی می‌کنیم:

* Recall: برای یک طبقه‌بندی خوب، recall باید 1 (بالا) باشد. recall تنها زمانی 1 می‌شود که صورت و مخرج باشند یعنی TP = TP + FN، این همچنین به این معناست که FN صفر شود. با افزایش FN مقدار مخرج بزرگتر از صورت می‌شود و مقدار recall کاهش می‌یابد (که ما نمی‌خواهیم).
* Precision
* Accuracy: تعداد نمونه‌های داده‌ای را که به درستی طبقه‌بندی شده‌اند در کل تعداد نمونه‌های داده را نشان می‌دهد.
* F-score: امتیاز F1 تنها زمانی 1 می‌شود که accuracy و recall هر دو 1 باشد. امتیاز F1 تنها زمانی بالا می‌شود که accuracy و precision هر دو بالا باشد.

Precision = TP / (TP+FP)

Recall = TP / (TP+FN)

F1 Score = 2 \* (Precision \* Recall) / (Precision + Recall)

و در نهایت بین مدل‌ها با توجه به معیارهای ارزیابی‌مان به مقایسه می‌پردازیم

* Knn

بهبود دقت نسبت به جواب پایه

بهبود precision ,f1\_score

بهبود recall روی کلاس یک

* Mlp

بهبود دقت نسبت به بیس لاین

بهبود precision ,f1\_score

بهبود recall روی کلاس یک

* :Xgboost

بهبود دقت نسبت به بیس لاین

بهبود precision , recall , f1\_score

نیجه : بهترین مدل با توجه به معیارها xgboost است

این دیتاست دارای داده‌های فیک و رندوم بوده و هیچکدام از ویژگی‌ها کورولیشنی با مقادیر هدف نداشتند از طرف دیگر توزیع دادها در کلاس‌های مختلف متناسب نبوده که این عوامل منجر به این شد که نتوان هوشمندسازی در داده‌ها را به نحو احسنت انجام داد در داده‌های واقعی قطعا الگوهایی برای ساخت مدل وجود دارد اما وجود الگو در داده‌های رندوم احتمال خیلی کمی دارد.

12