



Studium Magisterskie

Kierunek: Big Data – Analiza danych

Mateusz Kuchta

Nr albumu: 116111

Wpływ wrogiego uczenia maszynowego na modele estymacji wiarygodności kredytowej

Praca magisterska

pod kierunkiem naukowym

dr Mariusza Rafała

Kolegium Analiz Ekonomicznych

Warszawa 2023

Spis treści

Wstęp	3
1 Scoring kredytowy	5
1.1 Scoring kredytowy - definicja i cechy	5
1.2 Wyznaczanie oceny punktowej	9
1.3 Score kredytowy w erze big data	15
2 Wrogie uczenie maszynowe	20
2.1 Wprowadzenie do technik uczenia maszynowego	20
2.2 Charakterystyka wrogiego uczenia maszynowego	24
2.3 Typy ataków na algorytmy uczenia maszynowego	28
2.4 Przykłady ataków na algorytmy uczenia maszynowego	32
3 Budowa modelu drzewa decyzyjnego dla scoringu kredytowego	37
3.1 Analiza oraz wstępne przetworzenie wybranego zbioru danych	37
3.2 Wybór narzędzi, budowa modelu i analiza wyników	45
4 Atak na opracowany model	53
4.1 Postawienie pytań badawczych oraz przyjęcie strategii ataku	53
4.2 Pytanie badawcze nr 1	55
4.3 Pytanie badawcze nr 2	57
4.4 Pytanie badawcze nr 3	60
4.5 Pytanie badawcze nr 4	62
4.6 Podsumowanie badań nad wrogim uczeniem maszynowym	64
Podsumowanie	67
Literatura	69
Spis rysunków	74
Spis tabel	76
Załączniki	77

Wstęp

Celem pracy magisterskiej było przeprowadzenie szczegółowej analizy bieżącego stanu wiedzy i praktycznych dokonań w zakresie wykorzystania modeli predykcyjnych służących do celu estymacji wiarygodności kredytowej wnioskującego o kredyt, jak również zbadanie wrażliwości tego typu narzędzi na potencjalne ataki. Po przeprowadzeniu rozważań teoretycznych w dziedzinach scoringu kredytowego, uczenia maszynowego, a także wrogiego uczenia maszynowego, na łamach dwóch pierwszych rozdziałów, w drugiej części pracy należało przejść do części praktycznej, w ramach której wymagane było zbudować model uczenia maszynowego, a następnie go zaatakować celem obniżenia skuteczności jego przewidywań.

Rozpoczęto od zrozumienia procesu oceny zdolności kredytowej. Zdefiniowanie i przedstawienie cech tego procesu ukazuje jego kluczową rolę w umożliwianiu instytucjom finansowym dokonania właściwych decyzji kredytowych. Następnie poruszono temat wyznaczania oceny punktowej, odsłaniając kryteria stosowane do oszacowania ryzyka kredytowego. W kolejnym podrozdziale uwypuklono dynamiczne zmiany, jakie niesie ze sobą rosnąca dostępność ogromnych ilości danych oraz ich analiza w procesie weryfikacji kredytowej. Rozdział podsumowano wprowadzeniem do metod uczenia maszynowego wykorzystywanych w dziedzinie scoringu kredytowego, co ukazało aktualny stan zaawansowania technologicznego w stosowaniu interpretowalnych modeli predykcyjnych.

Następnie pochyłono się nad teorią dotyczącą wrogiego uczenia maszynowego, gdzie poruszono temat ryzyka i bezpieczeństwa algorytmów. Zapoznanie się z tym pojęciem pozwoliło na poszerzenie wiedzy dotyczącej bieżącego stanu wiedzy w tej tematyce, a także pomogło uświadomić o skali niebezpieczeństwa związanego z działalnością adwersarzy, którzy próbują wykorzystywać podatność modeli na manipulacje i ataki. Charakterystyka oraz rodzaje ataków na algorytmy uczenia maszynowego rzucają światło na ten aspekt, ukazując konieczność analizy i implementacji mechanizmów obronnych.

W drugiej części pracy zajęto się czynnościami praktycznymi. Omówiono proces budowy modelu predykcyjnego dla wystąpienia zdarzenia wejścia klienta w opóźnienie trwające w sumie co najmniej 90 dni w ciągu pierwszych dwunastu miesięcy od zaciągnięcia zobowiązania kredytowego, do czego zastosowano algorytm XGBoost. Wykorzystany zestaw danych zawiera informacje o klientach aplikujących zarówno o kredyty gotówkowe, jak i ratalne. Przeanalizowano wybór danych, jakość zbiorów oraz wykorzystane narzędzia

programistyczne, a efektywność modelu oceniano przez współczynnik Gini.

Pracę sfinalizowano postawieniem czterech pytań badawczych dotyczących ataku na wcześniej zbudowany model. W ramach testów wykorzystano różne rodzaje manipulacji danymi, celem oszukania algorytmu i zmuszenia go do generowania błędnych klasyfikacji. Ostatnim elementem pracy było jej podsumowanie, jak również odniesienie się do jej celu, napotkanych ograniczeń oraz kierunków dalszych badań.

1 Scoring kredytowy

Ocena wiarygodności kredytowej jest stosowana na całym świecie do przetwarzania wielu rodzajów pożyczek. Jest ona wykorzystywana najszerzej i z największym powodzeniem w przypadku osobistych kart kredytowych oraz kredytów hipotecznych. Ryzyko spłaty tych zobowiązań jest ściśle powiązane z czynnikami weryfikowalnymi, takimi jak dochód, ocena Biura Informacji Kredytowej, czy też demografia, tj. wiek, wykształcenie, status cywilny itp. (Caire, Barton, de Zubiria, Alexiev, & Dyer, 2006). Nieraz trudno jest ocenić, czy dana osoba zasługuje na zaufanie, czy może tym razem bank powinien się wstrzymać, nie ryzykując problemami ze spłatą danego kredytobiorcy, jednocześnie rezygnując z potencjalnego zysku.

Dokładne określenie granicy między dobrym, a złym klientem jest trudne nawet dla najbardziej doświadczonych pracowników finansowych. Zwiększona konkurencja i rosnąca presja na generowanie przychodów skłoniły instytucje udzielające kredytów do poszukiwania skutecznych sposobów pozyskiwania nowych klientów, przy jednoczesnej kontroli zysków i strat. Agresywne działania marketingowe wymusiły konieczność dokładniejszej analizy danych potencjalnych klientów, a potrzeba szybkiego i efektywnego ich przetwarzania doprowadziła do rosnącej automatyzacji procesu składania wniosków kredytowych i ubezpieczeniowych, a co za tym idzie, skrócenia czasu ich rozpatrywania (Siddiqi, 2016).

1.1 Scoring kredytowy - definicja i cechy

Głównym celem instytucji bankowych w zakresie optymalizacji procesu zarządzania ryzykiem, w tym ryzykiem kredytowym, nie jest jego całkowita eliminacja, ale znalezienie tzw. punktu równowagi. Nazywa się tak sytuację, w którym szacowana potencjalna strata finansowa wynikająca z akceptowalnego poziomu ryzyka kredytowego jest porównywana z przewidywanym wynikiem finansowym wynikającym z działalności kredytowej (Prokopowicz, 2014). Innymi słowy, bank nie skupia się na minimalizacji ryzyka, a na szukaniu jego poziomu na jaki może sobie pozwolić. Najskuteczniejszym sposobem ograniczania strat spowodowanych nie spłacaniem zobowiązań finansowych przez klientów jest nie dawanie im żadnych pożyczek, jednakże w tej sytuacji, mimo uniknięcia strat, odbierana jest również możliwość osiągnięcia zysków, na co instytucje finansowe nie mogą sobie pozwolić. Dlatego też niezwykle istotne jest dokładne zmierzenie ryzyka kredytowego za-

równy w odniesieniu do portfela, jak i konkretnej transakcji. W tym drugim przypadku obecnym standardem, jakim banki komercyjne określają akceptowalny poziom ryzyka kredytowego, jest metoda scoringu kredytowego. Jej rozwój jest istotnym czynnikiem w procesie doskonalenia zarządzania ryzykiem (Prokopowicz, 2014).

Scoring kredytowy można zdefiniować jako zespół metod statystycznych, używany w celu wyznaczania prawdopodobieństwa nie wywiązania się wnioskodawcy ze spłaty zaciągniętych zobowiązań w ustalonym terminie, co pomaga ustalić, czy kredyt powinien być przyznany potencjalnemu kredytobiorcy. Jest jedną z metod systematycznej oceny, która została uznana za istotnie wpływającą na obniżenie poziomu ryzyka wygenerowania strat finansowych w bankach.

W literaturze można znaleźć wiele definicji scoringu, co wynika z faktu, że jest to pojęcie w znacznym stopniu subiektywne. Ważne jest, aby model scoringowy charakteryzował się wysoką skutecznością w oddzielaniu klientów przynoszących zyski od tych, którzy prawdopodobnie przyniosą straty (Wysiński, 2013). Jest to zatem kluczowe narzędzie w rękach instytucji finansowych, dające możliwość ograniczania potencjalnego ryzyka współpracy z niewiarygodnym klientem, przy jednoczesnym osiąganiu jak największych korzyści finansowych poprzez zawieranie umów z wartościowymi kredytobiorcami (direct.money.pl, 2022).

Scoring kredytowy charakteryzuje się kilkoma podstawowymi cechami (mfiles.pl, 2020):

- Dane historyczne jako baza – porównuje się ze sobą charakterystyki grup kredytobiorców rzetelnych z nierzetelnymi i na tej podstawie dokonuje się oceny, czy potencjalny klient będzie terminowo spłacał zaciągnięte zobowiązanie, czy też może istnieje wysokie prawdopodobieństwo, że zachowa się on podobnie jak usługobiorcy mający problemy z uiszczaniem kolejnych rat na czas;
- Okresowość – mechanizmy wyliczania oceny wymagają częstych aktualizacji o nowe dane, w celu zapewnienia jak najwyższej skuteczności otrzymanego wyniku;
- Rzetelne zbadanie zdolności kredytowej kredytobiorców jako środek do celu jakim jest ochrona interesów kredytodawcy;
- Realizowany za pomocą zaakceptowanych i zatwierdzonych metod statystycznych.

Zwykle mówi się o dwóch typach scoringu - aplikacyjnym i behawioralnym. Pierwszy z nich jest stosowany u klientów, o których informacje są dostępne jedynie na podstawie wypełnionych przez nich wniosków kredytowych oraz danych pozyskanych z zewnętrznych źródeł np. z BIK - Biura Informacji Kredytowej.

Scoring behawioralny bierze pod uwagę informacje zgromadzone podczas wcześniejszej współpracy z klientem. Stanowi również narzędzie wspomagające monitorowanie portfela kredytowego oraz ograniczanie wysokości rezerw tworzonych na tzw. kredyty zagrożone. System scoringu behawioralnego jest szczególnie przydatny przy określaniu nowego limitu kredytowego lub modyfikacji limitu przyznanego wcześniej. Stosuje się go również w celu przeciwdziałania przekraczaniu określonych przez bank limitów na rachunkach, czy przedłużaniu warunków umów na dodatkowe produkty (np. kartę kredytową). Zatem podstawową różnicę pomiędzy scoringiem aplikacyjnym, a behawioralnym stanowi grupa docelowa klientów (Matuszyk, 2009).

W analizie ryzyka przedsiębiorstw stosuje się inny typ oceny, tzw. profit scoring (ang. scoring zysku). Taki system punktowania pozwala na określenie maksymalnego zysku, zamiast minimalizacji ryzyka związanego z obsługą danego klienta. Profit scoring, jako rozszerzenie podstawowego modelu scoringowego, bierze pod uwagę szereg dodatkowych czynników ekonomicznych tj. strategie marketingowe, dobór polityki cenowej, czy też poziom obsługi (bankier.pl, 2007).

Punktowa ocena wiarygodności kredytowej budowana jest na zasadzie przyznawania punktów za określone cechy kredytobiorcy, gdzie im wyższy wynik wnioskodawca osiągnie, tym większa szansa, że spłaci kredyt w terminie. Tabela punktowa tworzona jest na podstawie analizy statystycznej bazy danych klientów z przeszłości, gdzie poszukuje się cech, które w jak najlepszy sposób oddzielają od siebie dobrych i złych biorców kredytowych (bankier.pl, 2012).

Dla pewnego modelu, wpływ posiadania własnego mieszkania na spłacenie zobowiązania bez opóźnień zwizualizowano na rysunku 1. Widać na nim, że wśród osób mających problemy ze spłatą pożyczki (200 klientów), aż 95 procent (190 klientów) stanowią osoby wynajmujące nieruchomość. Zatem posiadanie własnego mieszkania będzie stawiać w uprzywilejowanej pozycji potencjalnych klientów banku i otrzymają oni wyższą ocenę za tę cechę w ogólnym scoring'u. Taką logiką kierują się zarówno banki, jak i BIK. Sposób



Rysunek 1: Przykład cechy wykorzystanej w scoringu kredytowym(bankier.pl, 2012)

wyliczania oceny punktowej często jest tajemnicą w instytucjach finansowych i zwykle nie dowiemy się jaki score otrzymaliśmy oraz co jest powodem takiego wyniku.

W Biurze Informacji Kredytowej, mimo iż nie poznamy pełnej logiki oceniania, pośrednio wiadome są najważniejsze kryteria oszacowań. Wśród najważniejszych z nich należy wymienić(bankier.pl, 2012):

- Liczba nieterminowo spłacanych zobowiązań;
- Wysokość zobowiązań (np. na karcie kredytowej lub debetowej);
- Czas zwłoki ze spłatą zobowiązania;
- Czas jaki upłynął od ostatniego wykroczenia.

Nieodłącznym elementem predykcji zachowań potencjalnych kredytobiorców względem zaciągniętego zobowiązania jest szacowanie zdolności kredytowej. O ile w przypadku kredytów gotówkowych banki często stosują dosyć liberalne podejście, to gdy pod uwagę brane są duże kwoty pożyczki, tak jak ma to miejsce w przypadku kredytów hipotecznych, etap estymacji zdolności stanowi gęste sito dla wnioskujących, szczególnie w czasach kryzysów gospodarczych, gdzie najczęściej mamy do czynienia z wysokimi stopami procentowymi. Oczywiście analiza zdolności kredytowej opiera się na różnych kryteriach, które dodatkowo różnią się między poszczególnymi bankami, jednak relacja zarobków do wydatków, które są pośrednio zależne od poziomu stóp procentowych, stanowi bazę do

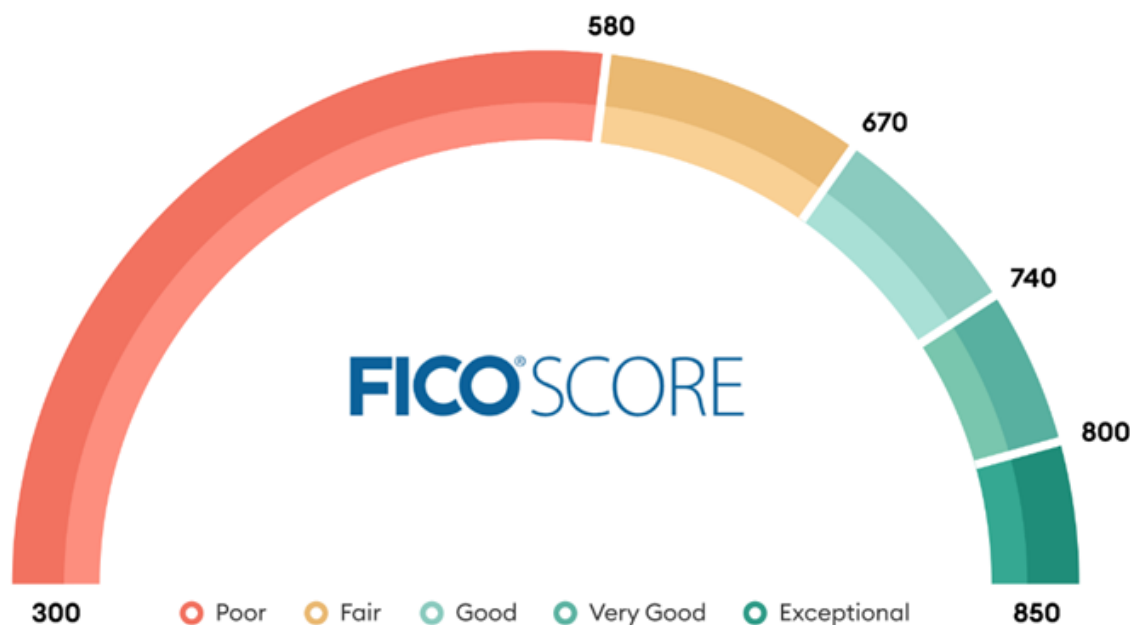
dalszej oceny(habza.com.pl, 2022).

Biuro Informacji Kredytowej nie ukrywa jakie cechy są preferowane przez banki oraz na co należy zwrócić uwagę, aby podnieść kwotę maksymalnego możliwego do uzyskania kredytu(bik.pl, 2022):

- Dochody i forma zatrudnienia - umowa o pracę to zdecydowanie wariant, który wpływa dobrze na zdolność kredytową. W przypadku umowy o dzieło czy samozatrudnienia bank ma mniejszą pewność co do stabilności zarobków. Mniej oczywista jest również kwestia dochodów mikroprzedsiębiorców;
- Obciążenia wydatkami - Koszty stałe, kredyty, pożyczki, karty kredytowe - im mniej z nich wnioskodawca posiada tym potencjalnie wyższa miesięczna rata jaką może obsłużyć;
- Warunki wnioskowanego kredytu:
 - Czas - wydłużenie okresu kredytowania może obniżyć wysokość miesięcznej raty, co ma bezpośredni wpływ na zdolność kredytową;
 - Wspólny kredyt - rodzice, partner, partnerka, rodzeństwo to potencjalnie dobrzy kandydaci do wspólnego kredytu. Takie rozwiązanie nie tylko poprawia zdolność kredytową, ale jednocześnie daje dodatkowe zabezpieczenie dla banku;
 - Równe raty - lepiej wybrać kredyt o równych ratach kapitałowo-odsetkowych, aby zwiększyć swoje szanse na pozytywną decyzję kredytową;
- Historia kredytowa w BIK- z Raportu BIK można dowiedzieć się jakie informacje na temat wnioskodawcy dotychczas przekazywały do BIK banki, SKOK-i i firmy pożyczkowe.

1.2 Wyznaczanie oceny punktowej

W Stanach Zjednoczonych score kredytowy zawiera się w granicach od 300 do 850 punktów, gdzie w zależności od instytucji za odpowiednio dobry wynik traktuje się wartości powyżej 661-670 punktów(experian.com, 2021). Przedziały ocen deklarowane przez FICO (instytucję opisano szerzej w podrozdziale 1.3) przedstawiono na rysunku 2.



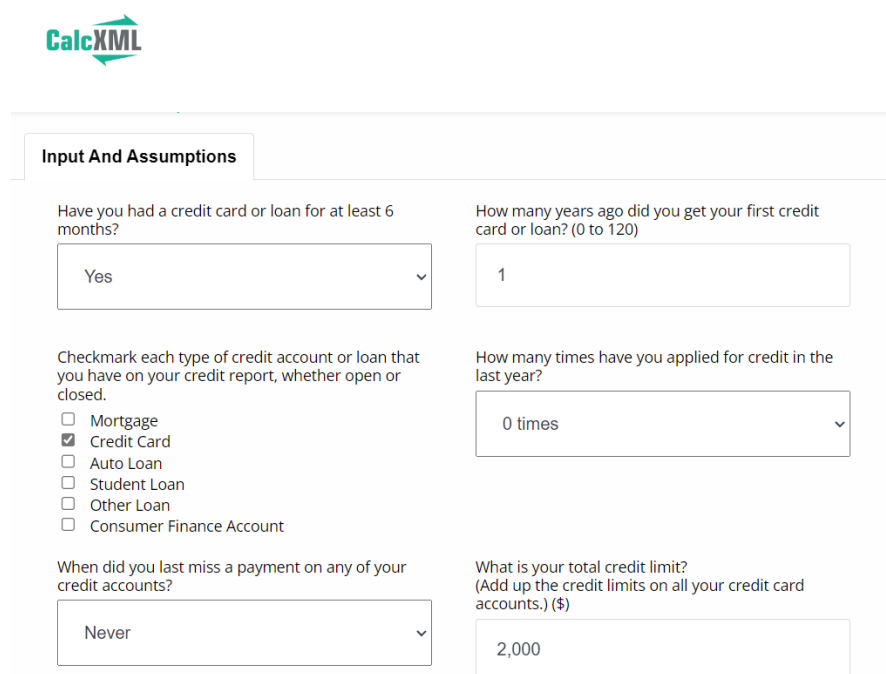
Rysunek 2: Diagram oceny wiarygodności kredytowej według firmy FICO(forbes.com, 2021)

W sieci można znaleźć darmowe kalkulatory score'u kredytowego i choć nie należy bezkrytycznie ufać ich kalkulacjom, jako że nie znamy dokładnych mechanizmów oraz zbiorów danych na podstawie których zbudowano dany model, to niektóre z nich potrafią zwrócić wyniki, które z pewną dozą niepewności możemy przyjąć jako prawdopodobne wartości wyliczane w profesjonalnych instytucjach. Jednym z takich narzędzi jest kalkulator na stronie CalcXML, który zwraca wynik w skali FICO. Osoba zainteresowana obliczeniem swojej indywidualnej oceny powinna przygotować kilka podstawowych informacji na swój temat(calcxml.com, 2023):

- Czy Wnioskujący posiadał pożyczkę lub kartę kredytową przez okres dłuższy niż 6 miesięcy;
- Ile lat temu Wnioskujący po raz pierwszy wziął pożyczkę lub posiadał kartę kredytową;
- Które z poniższych zobowiązań Wnioskujący posiada lub posiadał:
 - Kredyt hipoteczny;
 - Karta kredytowa;
 - Pożyczka na samochód/Pożyczka studencka/Inna pożyczka.
- Suma limitów ze wszystkich posiadanych przez Wnioskującego kart kredytowych;

- Czy Wnioskującego kiedykolwiek dotyczyło którekolwiek z poniższych "negatywnych zdarzeń":
 - Bankructwo;
 - Interwencja komornika/przejęcie własności;
 - Problemy podatkowe;
 - Inne "negatywne zdarzenie".
- Kiedy miało miejsce ostatnie "negatywne zdarzenie", które dotyczyło Wnioskującego.

Przykładowa kalkulacja, dla osoby posiadającej kartę kredytową z limitem 2000 dolarów od ponad roku, nie posiadającej dodatkowych zobowiązań, która w ostatnich dwunastu miesiącach nie wysyłała wniosków o kredyt, nie spóźniała się z spłatą zobowiązań oraz nie brała udziału w żadnym z "negatywnych zdarzeń", której fragment przedstawiono na rysunku 3, daje wynik od 740 do 790 punktów, co oznacza bardzo dobry score kredytowy.



CalcXML

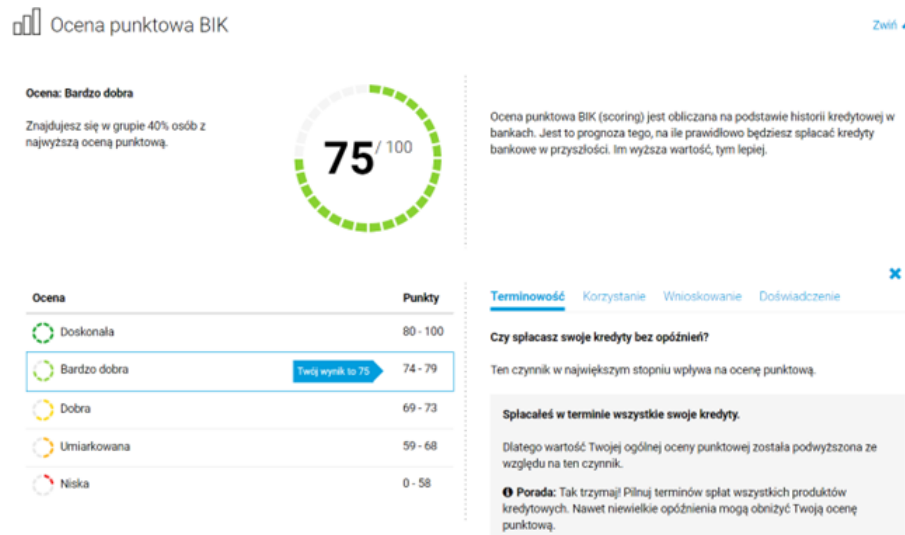
Input And Assumptions

Have you had a credit card or loan for at least 6 months? <input type="text" value="Yes"/>	How many years ago did you get your first credit card or loan? (0 to 120) <input type="text" value="1"/>
Checkmark each type of credit account or loan that you have on your credit report, whether open or closed. <input type="checkbox"/> Mortgage <input checked="" type="checkbox"/> Credit Card <input type="checkbox"/> Auto Loan <input type="checkbox"/> Student Loan <input type="checkbox"/> Other Loan <input type="checkbox"/> Consumer Finance Account	How many times have you applied for credit in the last year? <input type="text" value="0 times"/>
When did you last miss a payment on any of your credit accounts? <input type="text" value="Never"/>	What is your total credit limit? (Add up the credit limits on all your credit card accounts.) (\$) <input type="text" value="2,000"/>

Rysunek 3: Fragment kalkulacji ze strony calcxml.com (calcxml.com, 2023)

Od 21 grudnia 2017 roku BIK generuje ocenę punktową w nieco inny sposób. Dotychczas score zawierał się w zakresie od 192 do 631 punktów, co wizualizowane było

za pomocą gwiazdek, gdzie im więcej gwiazdek, tym wyższa szansa na otrzymanie kredytu. Aby uczynić reprezentację graficzną bardziej czytelną dla osób fizycznych, ocenę przekształca się do postaci od 1 do 100 (totalmoney.pl, 2020) Zostało to zilustrowane w postaci wykresu kołowego na rysunku 4.



Rysunek 4: Wizualizacja oceny punktowej w BIK (źródło opracowanie własne)

Przed zmianą z grudnia 2017, Biuro Informacji Kredytowej do wyliczania oceny punktowej wykorzystywało tzw. Model II generacji o nazwie BIKSco CreditRisk. Dla tego algorytmu zakres możliwych do uzyskania rezultatów zawierał się pomiędzy 192, a 631 punktów i nie był w żaden sposób przekształcany. W najnowszych raportach wykorzystywany jest model III generacji o nazwie BIKSco CreditRisk 3, a punktacja zawiera się od 98 do 711 punktów. Jednakże otrzymany wynik ulega przekształceniu i tak jak jest to widoczne na rysunku 4, przedstawiony jest w zakresie od 1 do 100, co może mylnie wskazywać, że jest to procent maksymalnej, możliwej do uzyskania punktacji (scoringexpert.pl, 2018).

Bazując na cytowanym źródle, wynik wyliczonej punktacji podlega procesowi normalizacji według równania 1:

Równanie 1. Wzór na przekształcenie oceny wyliczonej z modelu BIKSco CreditRisk 3 do przedziału znormalizowanego 1-100 (experian.com, 2021)

$$S_{new} = \frac{S - min}{max - min} * (new_{max} - new_{min}) + new_{min}$$

gdzie:

- S_{new} - ocena punktowa z raportu BIK;
- S - oryginalna ocena punktowa z modelu BIKSco CreditRisk 3;
- min - minimalna wartość punktów z modelu BIKSco CreditRisk 3 (wynosi 98);
- max - maksymalna wartość punktów z modelu BIKSco CreditRisk 3 (wynosi 711);
- new_{max} - maksymalna wartość punktów z nowego zakresu (wynosi 100);
- new_{min} - minimalna wartość punktów z nowego zakresu (wynosi 1).

Na podstawie wartości widocznej dla osoby fizycznej (punktacji z przedziału 1-100), można uzyskać wynik wyliczony z algorytmu na podstawie równania 2:

Równanie 2. Wzór na wyliczenie wartości otrzymanej z modelu BIKSco CreditRisk 3 na podstawie wartości uzyskanej z Biura Informacji Kredytowej(experian.com, 2021)

$$S = \frac{613 * S_{new} + 9089}{99}$$

Biuro Informacji Kredytowej proponuje swoją interpretację znormalizowanej oceny punktowej, przedstawioną na rysunku 4, według której potencjalny kredytobiorca może ocenić swoje aktualne szanse na otrzymanie pożyczki. Na podstawie tabeli 1, można zinterpretować również ocenę nieznormalizowaną.

Ocena punktowa BIK	Słowna ocena	Komentarz
550+ (74+)	Bardzo dobra	Twój „scoring BIK” przewyższa średni „scoring BIK” Polaków. W ocenie banków Twoja wiarygodność kredytowa powinna być bardzo wysoka
500-549 (66-73)	Dobra	Twój „scoring BIK” oscyluje blisko średniego „scoringu BIK” Polaków. Banki oceniają Cię jako rzetelnego kredytobiorcę
400-499 (52-65)	Przeciętna	Twój „scoring BIK” jest poniżej średniego „scoringu BIK” Polaków. Tylko część banków oceni Twoją wiarygodność kredytową jako wystarczającą do uzyskania kredytu
poniżej 400 (poniżej 52)	Słaba	Twój „scoring BIK” jest znacznie poniżej średniego „scoringu BIK” Polaków. Taki scoring wskazuje, że jesteś ryzykownym kredytobiorcą dla banków. Możesz więc mieć problem z uzyskaniem kredytu, jeśli bank oprze się na „scoringu BIK” przy ocenie Twojego ryzyka kredytowego

Tabela 1: Interpretacja oceny punktowej BIK(scoringexpert.pl, 2017)

Za jedną z najbardziej znanych metod scoringu kredytowego jest uważana przytaczana na łamach tej pracy amerykańska metoda FICO (Fair, Isaac and Company). Zdefiniowana w 1989 roku, opiera się na 5 czynnikach(pl.economy pedia.com, 2021):

- Historia płatności (35 procent punktów) – na bazie dotychczasowych zobowiązań ocenia się, czy dana osoba wywiązuje się z nich na czas;
- Wykorzystanie kredytu (30 procent) – jeśli potencjalny klient instytucji finansowej dotychczas wykorzystywał niewielki procent dostępnych limitów kredytowych (np. limit na karcie kredytowej), ma on większe szanse na wyższą ocenę;
- Długość historii kredytowej (15 procent) – jeśli wnioskodawca jest doświadczonym pożyczkobiorcą i przez długi czas poprawnie wywiązuje się z zobowiązań otrzymuje wyższą notę w tabeli punktowej;
- Nowe kredyty (10 procent) – złożenie wielu wniosków kredytowych przez osobę poszukującą kredytu, może wzbudzać wątpliwości instytucji przed udzieleniem pożyczki;
- Rodzaje wykorzystanego kredytu (10 procent) – dla banków mile widziane jest doświadczenie wnioskującego w zarządzaniu różnymi rodzajami kredytów (karta kredytowa, kredyt hipoteczny, pożyczka gotówkowa itp.).

Rozwój scoringu kredytowego jest jednym z powodów, dla których rynek kredytów konsumenckich w Stanach Zjednoczonych w latach 90. XX w. eksplodował. Kredytodawcy czuli się bardziej pewni w udzielaniu pożyczek szerszym grupom ludzi, ponieważ mieli dokładniejsze narzędzie do pomiaru ryzyka. Scoring kredytowy pozwolił im również na szybsze podejmowanie decyzji, umożliwiając rozpatrzenie większej liczby wniosków, czego rezultatem był bezprecedensowy wzrost ilości dostępnych kredytów konsumenckich(Weston, 2012). Banki bardzo skrupulatnie podchodzą do operowania swoimi pieniędzmi, dokładnie „prześwietlając” swoich klientów pod kątem wypłacalności. Polskie instytucje finansowe często posiłkują się opinią BIK, jednakże równie chętnie stosują także własne algorytmy oceny, których szczegółów zwykle szerzej nie udostępniają.

1.3 Score kredytowy w erze big data

Historia rozwoju modeli scoringowych sięga tak daleko, jak historia pożyczania i spłacania. Widać to w szczególności w potrzebie ustalenia odpowiedniej stopy procentowej, uwzględniającej ryzyko braku możliwości odzyskania pożyczonych pieniędzy. Wraz z nadejściem współczesnej ery statystyki w XX. wieku rozpoczęto opracowywanie technik oceny prawdopodobieństwa niewykonania przez kredytobiorcę zobowiązania do zwrotu środków. Pod uwagę brano podobieństwo przypisanych cech do tych, którymi charakteryzują się osoby niewywiązujące się ze zobowiązań w przeszłości. Tego typu metody statystyczne są obecnie stosowane powszechnie przez praktycznie wszystkie banki, a również znaczną część pozostałych instytucji finansowych(prawniczydotblog.wordpress.com, 2019).

W 1956 roku powstała wcześniej wspomniana firma FICO. Przedsiębiorstwo założyli inżynier Bill Fair oraz matematyk Earl Judson Isaac(Wikipedia, 2022). Firma zajmowała się budową systemów scoringowych, jednakże przez długi czas stosowano zapis papierowy(StatSoft, 2010). Początkowo zajmowano się głównie procesami weryfikacji wniosków kredytowych w bankach, stosując proste tabele punktowe i głównie opierając się na tzw. metodzie eksperckiej(Thonabauer & Nosslinger, 2004). Sposób musiał być na tyle łatwy i intuicyjny, aby dawać możliwość obiektywnej oceny zdolności potencjalnego kredytobiorcy do wywiązania się z zaciągniętego zobowiązania kredytowego również mniej doświadczonym i niewykwalifikowanym pracownikom banku(Thomas, Edelman, & Crook, 2002). Jednym z pierwszych i najważniejszych osiągnięć Fair Isaac & Company było utworzenie pierwszego, wykorzystywanego komercyjnie, systemu scoringowego(Poon, 2007).

Jednakże zanim FICO rozpoczęło swoją działalność, ludzkość zebrała już pierwsze doświadczenia ze złymi kredytobiorcami, a co za tym idzie zaczęto zastanawiać się nad ich właściwym zidentyfikowaniem zanim zostanie im udzielona pożyczka. Rok 1826 przyjmuje się za początek wymiany informacji między wierzycielami, co miało na celu podniesienie jakości "filtracji" właściwych klientów, natomiast w 1899 roku w Atlancie rozpoczęto zbieranie danych na większą skalę. Po powstaniu Fair Isaac & Company rozwój w dziedzinie udzielania kredytów znacząco przyspieszył, co skutkowało powstawaniem konkurencyjnych dla FICO firm (wśród nich warto wymienić chociażby Experian czy Equifax), ale też było bodźcem do podwyższenia jakości usług(ecomparemo.com, 2020). Zwieńczenie

i zarazem wykładnik wzrostu znaczenia score'u kredytowego stanowiło wejście Fair, Issac and Company na giełdę w Nowym Jorku w lipcu 1987 roku(fico.com, 2023).

Klasyczne karty punktowe cechują się prostotą w interpretacji wyników, a wykorzystanie takich jak tabela 2 nie wymaga wiedzy analitycznej wymaganej w stosowanych współcześnie metodach komputerowych. Trudno zatem oprzeć się wrażeniu, iż klasyczny scoring kredytowy usuwa się w cień na korzyść narzędzi z dziedziny big data, choć w rzeczywistości staje się jedną z poddziedzin szerokiej tematyki „dużych danych” (Przanowski, 2014). Lecz czym właściwie jest big data?

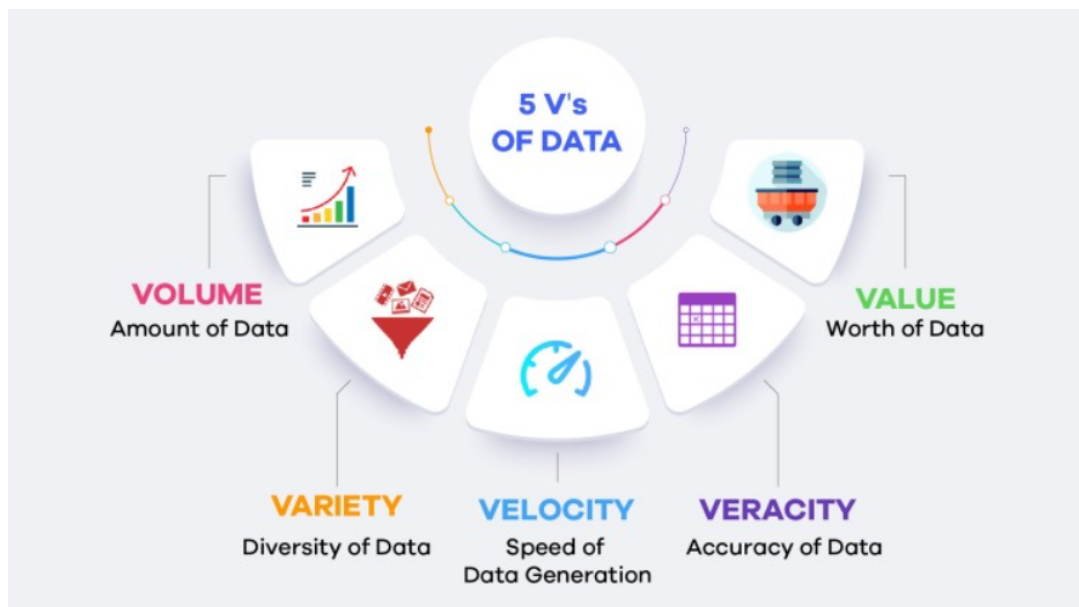
Attribute	Variable	Partial Score
<20	Age	10
20>= and <34		20
35>=		30
Bad	Payment history	10
Not good		25
Good		40

Tabela 2: Klasyczna karta oceny punktowej(Przanowski, 2023)

Pojęcie to powinno iść w parze z rozbudowanymi systemami informatycznymi, które dają możliwość przetwarzania dużych danych. Często spotykanym jest tzw. 5V big data (zwizualizowane na rysunku 5(researchgate.net, 2017)), dające ogólne spojrzenie na podstawowe cechy tejże dziedziny(techtarget.com, 2021):

- Volume (ang. wolumen, wielkość) – jeśli mamy do czynienia z subiektywnie dużą ilością danych (co dokładnie oznacza „duża ilość” nie zostało precyzyjnie zdefiniowane), możemy nazwać ich przetwarzanie jako big data;
- Velocity (ang. szybkość) – ze względu na ich ilość, dane muszą być odpowiednio szybko procesowane, najczęściej w czasie rzeczywistym;
- Variety (ang. różnorodność) – technologia musi radzić sobie z operowaniem na danych zróżnicowanego, nie koniecznie uporządkowanego typu;
- Veracity (ang. wiarygodność) – dane najczęściej nie są wysokiej jakości, a ich braki czy też błędne informacje w nich zawarte stawiają wymóg odpowiedniej odporności na tego rodzaju zaburzenia;

- Value (ang. wartość) – kluczowa przy przetwarzaniu danych jest możliwość uzyskania na ich bazie istotnych informacji, które mogą wnieść pewną wartość dla przedsiębiorstwa, dokonującego lub zlecającego wykonanie takich analiz.



Rysunek 5: Schemat 5V big data(researchgate.net, 2017)

Początkowo scoring kredytowy specjalizował się głównie we wspomaganiu procesów decyzyjnych w bankach, a narzędzia big data stosowane były w globalnych firmach świadczących usługi w świecie wirtualnym tj. Google, Amazon czy Facebook. Z kolei w Polsce zarządzaniem dużymi danymi na poważnie zainteresowały się jako pierwsze Onet czy portal Nasza Klasa(Przanowski, 2014). Mimo zainteresowania różnymi branżami, big data i scoring kredytowy poruszają podobne problemy ze strony merytorycznej, gdzie głównym i najpoważniejszym problemem zawsze był kluczowy element ich funkcjonowania – dane.

Modele scoringu kredytowego służą do prognozowania zjawisk na podstawie dotychczas zaobserwowanej i zebranej historii danych. Proces spłacania kredytów najczęściej trwa wiele lat, zatem potrzeba dużo czasu, aby zebrać dostatecznie reprezentatywną pulę informacji rzeczywistych, którą następnie można wykorzystać do sprawdzenia użyteczności i poprawności skonstruowanego modelu(Przanowski, 2014).

W przypadku danych bankowych, sytuacja jest jeszcze trudniejsza z uwagi na wrażliwość informacji. Skutkuje to koniecznością występowania do instytucji finansowych z oficjalnymi podaniami, a otrzymane dane często są zafałszowane i zanonimizowane, co zwy-

kle uniemożliwia ich zinterpretowanie. Znacząco utrudnia to tworzenie odpowiednio wiarygodnych modeli scoringowych, na co analitycy odpowiadają tworzeniem własnych, symulowanych danych (Przanowski, 2014).

Dzięki big data banki mogą także skuteczniej reklamować swoje usługi. Stosowanie scoringu behawioralnego ułatwia odgadnięcie intencji klientów – internautów. Na podstawie przeglądanych stron internetowych bank jest w stanie przewidywać, jaki produkt może zainteresować takiego obywatela. Następnie klient składa wniosek o pożyczkę, co znacznie ułatwia podjęcie decyzji o przyznaniu finansowania. Wystarczy odtworzyć historię klienta. Według raportu „Banks Betting Big on Big Data and Real-Time Customer Insight”, przygotowanego przez Bloomberg Businessweek oraz SAP, aż 86 proc. największych banków na świecie deklaruje, że w najbliższych latach priorytetem będzie dla nich zorientowanie działań na konsumenta, w tym przede wszystkim: dopasowanie oferty do konkretnych, indywidualnych potrzeb klienta (pkobp.pl, 2018).

Dziś banki mogą pozyskiwać informacje o klientach nawet z portali zakupowych czy profili w mediach społecznościowych. Dzięki analizie takich danych można spośród dużej liczby klientów wybrać tych, którzy mogą być zainteresowani kredytem gotówkowym lub mieszkaniowym. Coraz częściej pojawia się tzw. social scoring. Wystarczy że internauta zaloguje się do banku za pośrednictwem portalu społecznościowego (takiego jak Facebook czy LinkedIn) i wyrazi zgodę na automatyczne wykorzystywanie danych (takich jak adres e-mail, rok urodzenia itp.), by bank automatycznie uwzględnił te dane we wniosku kredytowym. Analitycy bankowi mogą dzięki temu stworzyć precyzyjne profile klientów, które zawierają informacje o upodobaniach i preferencjach zakupowych (pkobp.pl, 2018).

Najważniejsza w całym procesie jest umiejętność wyciągania wniosków z ogromnej ilości danych. Szacuje się, że każdego dnia do sieci przesyłanych jest ponad dwa i pół miliarda gigabajtów nowych danych. Nie oznacza to jednak, że banki mogą wykorzystywać wszystko co dostępne. Podstawą jest wskazanie, jakie dane instytucja finansowa wykorzystuje i jaką korzyść taka analiza jej przynosi (pkobp.pl, 2018).

Skuteczna analiza danych jest dziś polem do konkurencji bankowej. Instytucje finansowe walczą między sobą już nie na dobrą ofertę dla wszystkich, ale na umiejętne dotarcie z nią do potencjalnego klienta. Przetwarzanie danych w czasie rzeczywistym staje się przewagą konkurencyjną, której wartość jest nie do przecenienia. Jednakże coraz bardziej

świadomości klienci nie zawsze chętnie chcą spinać aplikacje bankowe z profilami powiązanymi z mediami społecznościowymi. Nie do końca zdają sobie sprawę z korzyści z tego płynących. Efekt takiego „otwarcia się klienta” pozwala na przedstawienie mu bardzo dobrze dobranej – z perspektywy użytkownika i zupełnie bezpiecznej z punktu widzenia banku – oferty kredytowej (pkobp.pl, 2018).

Dziedzina big data nie jest bez wad i mimo jej niekwestionowanej przydatności, bez trudu można wymienić niepowodzenia i wyzwania jakie napotyka, gdzie najistotniejsze z nich to (Przanowski, 2023):

- Dane często nie są zbierane, a jeśli ktoś już je magazynuje, nie zapewnia odpowiedniej ich przydatności i interpretowalności;
- Problem jakości danych;
- Liczne braki danych;
- Brak publicznych danych, dostępnych i przykładowych;
- Brak inwestycji w przygotowanie i wykształcenie inżyniera danych.

W erze big data, ocena kredytowa nabiera nowego wymiaru, opierając się na znacznie obszerniejszych i różnorodnych źródłach danych. Tradycyjne kryteria oceny kredytowej zostają wzbogacone o dane związane z zachowaniami cyfrowymi, transakcjami online i interakcjami społecznymi, umożliwiając tworzenie bardziej precyzyjnych i wszechstronnych profili kredytowych. Jednakże, wykorzystanie tych danych stawia wyzwania związane z prywatnością, dokładnością i interpretacją, co wymaga zrównoważonego podejścia do tworzenia skutecznych modeli oceny kredytowej.

2 Wrogie uczenie maszynowe

Modele uczenia maszynowego otworzyły zupełnie nowe możliwości w dziedzinie automatyzacji, a wizja wszechobecnej sztucznej inteligencji skutecznie rozpala wyobrażenia ludzi o świecie, w którym na porządku dziennym będziemy wykorzystywać roboty, posiadające własną świadomość. Jednakże należy pamiętać, że z wielką mocą wiąże się wielka odpowiedzialność, a wykorzystanie nowych możliwości w złym celu, może nieść ze sobą groźne skutki.

2.1 Wprowadzenie do technik uczenia maszynowego

Karty punktowe, mimo że łatwe w interpretacji zarówno przez wnioskujących, jak i sprzedawców, nie są najbardziej optymalnym narzędziem do oceny wiarygodności kredytowej. Przez ostatnie kilkadziesiąt lat pojawiło się wiele nowych możliwości analizy, co jest związane z nieustającym rozwojem informatyzacji, a niektóre z nich zostały dopasowane do dziedziny scoringu kredytowego, dając lepszą efektywność predykcji oraz podnosząc zyski instytucji finansowych.

Uczenie maszynowe, samouczenie się maszyn lub systemy uczące się, w języku angielskim tłumaczone jako Machine Learning jest dziedziną wchodzącą w skład nauk, zajmujących się sztuczną inteligencją. Głównym jej celem jest tworzenie automatycznego systemu, który potrafi doskonalić się na bazie doświadczenia i nabywać na tej podstawie nową wiedzę. W uproszczeniu proces polega na znalezieniu wzorca w dostarczonych danych. Modele uczenia maszynowego powszechnie wykorzystywane są w wielu dziedzinach, w których zachodzi potrzeba predykcji pewnego zjawiska(gov.pl, 2021).

Dotychczas w bankowości nie stosowano powszechnie niektórych technik uczenia maszynowego do zarządzania ryzykiem, a geneza takiego postępowania była zrozumiała – modele są trudne w interpretacji, a ponadto generują popyt na wysoce wyspecjalizowanych pracowników. Z drugiej zaś strony, rynek konkurencyjny zmienia się - transformacja cyfrowa, czy też nowy model bankowości otwartej wywierają wpływ na praktyki zarządzania ryzykiem. W tym kontekście stosowanie technik uczenia maszynowego zapewnia istotną przewagę, skracając czas podejmowania decyzji w procesach kredytowych oraz podnosząc ich skuteczność (crif.pl, 2018).

Zadania uczenia maszynowego ograniczone są do wąskiego, specyficznego zakresu,

w którym ma działać dany system. W przeciwieństwie do sztucznej inteligencji, proces nie jest w stanie stworzyć czegoś nowego, a jedynie uzyskiwać najbardziej optymalne rozwiązania w zadanym problemie. Najpopularniejszymi aplikacjami wykorzystującymi możliwości uczenia maszynowego są wyszukiwarki online, algorytmy podpowiadające najciekawsze dla użytkowników materiały w mediach społecznościowych, rozpoznawanie obrazów czy filtrowanie spamu ze skrzynek e-mail(elektronikab2b.pl, 2021).

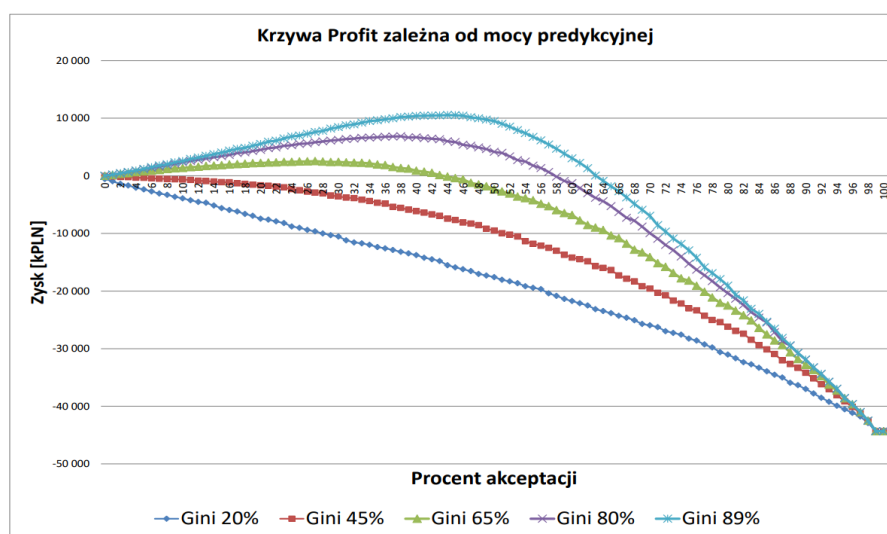
Eksperti SAS wskazują 4 podstawowe techniki uczenia maszynowego(sas.com, 2018):

- Supervised Learning (ang. Uczenie Nadzorowane) - maszyny uczą się na podstawie dostarczonych przykładów, a dane wejściowe są wykorzystywane do wyszukiwania zależności, służących do rozwiązania określonego problemu. Gdy uda się ustalić pewien wzorzec, jest on wykorzystywany w podobnych przypadkach. Do przykładowych zastosowań tej metody należą zarządzanie ryzykiem, rozpoznawanie mowy, tekstu i obrazu, a także segmentacja klientów.
- Semi - Supervised Learning (ang. Uczenie Częściowo Nadzorowane) - maszyna otrzymuje zarówno dane wejściowe oznaczone (zawierające odpowiadające im dane wyjściowe, konkretne przykłady), jak i nieoznaczone (wymagające przyporządkowania do danych wyjściowych, znalezienia odpowiedzi). Ten rodzaj uczenia wykorzystuje się w sytuacjach, gdy dana instytucja dysponuje zbyt dużą ilością danych lub gdy informacje cechują się wysokim zróżnicowaniem, które uniemożliwia przyporządkowanie odpowiedzi do każdej z nich. W takiej sytuacji system sam proponuje odpowiedzi i jest w stanie stworzyć ogólne wzorce. Metoda znajduje zastosowanie w rozpoznawaniu mowy, obrazów, jak również w klasyfikacji stron internetowych.
- Unsupervised Learning (ang. Uczenie Nienadzorowane) - maszyna nie posiada „klucza odpowiedzi” i musi sama analizować dane, szukać wzorców i odnajdować relacje. Ten typ uczenia maszynowego najbardziej przypomina sposób działania ludzkiego mózgu, który wyciąga wnioski na podstawie spontanicznej obserwacji i intuicji. Wraz ze zwiększaniem się rozmiaru zbiorów danych prezentowane wnioski są coraz bardziej precyzyjne. Przykładami wykorzystania są analiza koszyka zakupowego, wykrywanie anomalii, czy też rozpoznawanie podobnych obiektów.
- Reinforcement Learning (ang. Uczenie Wzmocnione) - maszyna otrzymuje gotowy zestaw dozwolonych działań, reguł i stwierdzeń oraz wykorzystuje je w taki spo-

sób, aby osiągnąć pożądany efekt. Można to porównać do nauki gry np. w darta. Zasady określające, ile punktów musi zdobyć zawodnik oraz fakt zakończenia wartością podwójną pozostają niezmiennie. Natomiast najbardziej optymalna kombinacja punktów otrzymanych z maksymalnie trzech rzuconych lotek zależy od indywidualnej decyzji gracza. Przykłady zastosowań to nawigacja GPS (wybór trasy bazując na danych o natężeniu ruchu i pogodzie), przemysł gamingowy (dopasowanie scenariuszy rozgrywki do działań gracza), jak również robotyka (dostosowanie natężenia pracy robotów do popytu).

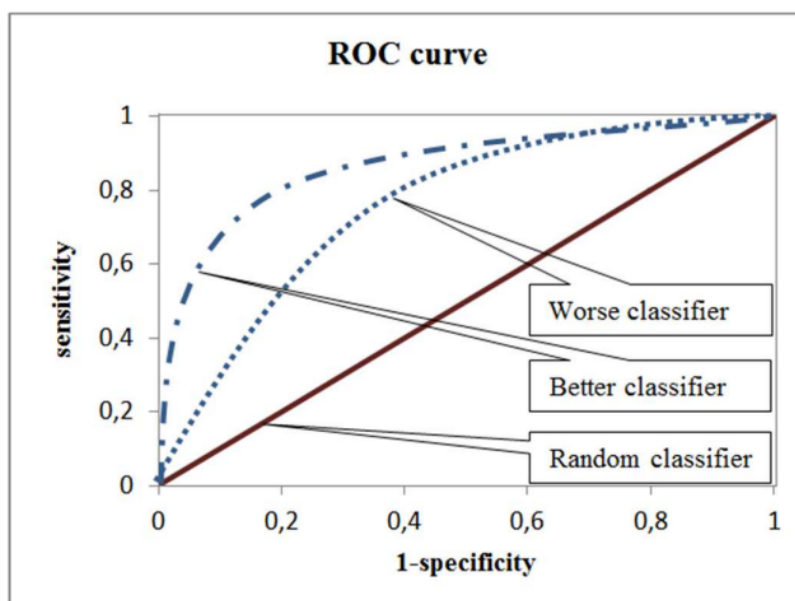
Wraz z rozwojem technik informatycznych rozpoczęto wdrażanie bardziej zautomatyzowanych procesów scoringowych. Najpowszechniej do tego celu wykorzystywano regresję logistyczną, jednakże dziś stosuje się szeroką gamę różnych metod predykcyjnych tj. sieci neuronowe czy drzewa decyzyjne(Przanowski, 2014).

Jednym z podstawowych podejść w modelowaniu scoringu kredytowego jest wyliczenie prawdopodobieństwa wystąpienia zdarzenia "default", które cechuje się wejściem w opóźnienie w spłacie zobowiązania wynoszące sumarycznie co najmniej 90 dni. Finalnie, model powinien obliczyć szansę na to, że wnioskodawca "wpadnie w default", gdzie np. zmienna default12 wskazuje, iż dłużnik w ciągu 12 miesięcy od zaciągnięcia długu spóźnił się ze spłatą raty o co najmniej 90 dni(Przanowski, 2015). Pozornie logika nakazywałaby odrzucać wnioski, dla których prawdopodobieństwo problemów ze spłatą wynosi więcej niż 50 %, jednakże nie jest to tak oczywiste, jak może się na wydawać. Celem zobrazowania tego zagadnienia warto omówić działanie Krzywej Profit(rysunek 6).



Rysunek 6: Krzywa Profit zależna od mocy predykcyjnej(Przanowski, 2023)

Na Krzywej Profit wizualizuje się wartość zysku w zależności od procentu zaakceptowanych wniosków dla różnych modeli predykcyjnych, gdzie każdy z nich opisuje się współczynnikiem Giniego. Model buduje się "trenując go" na zbiorze treningowym, złożonym z danych historycznych, gdzie algorytm wychwytuje zależności cechujące klientów spłacających zobowiązania bez opóźnień, a także uczy się odróżniać niewiarygodnych wnioskodawców. Następnie na zbiorze testowym porównuje się wnioski prognozowane przez model z sytuacją jaka w rzeczywistości miała miejsce i na tej podstawie buduje się tzw. krzywą ROC - przykładowa wizualizacja na rysunku 7.



Rysunek 7: Krzywa ROC i jej możliwe warianty (Gajowniczek et al., 2014)

Do skonstruowania krzywej ROC niezbędne jest wyliczenie czterech parametrów, składających się na tzw. macierz pomyłek (Fawcett, 2005):

- False Positive (FP) - model wskazał, że dana osoba wpadnie w opóźnienie w spłacie, mimo iż w rzeczywistości sumiennie spłacała kredyt;
- False Negative (FN) - model wskazał, że dana osoba nie wpadnie w opóźnienie w spłacie, mimo iż w rzeczywistości miała kłopoty ze spłatą;
- True Positive (TP) - model wskazał, że dana osoba wpadnie w opóźnienie w spłacie, co okazało się zgodne z rzeczywistością;
- True Negative (TN) - model wskazał, że dana osoba nie wpadnie w opóźnienie w spłacie, co okazało się zgodne z rzeczywistością.

Na podstawie powyższych wartości oblicza się wielkości takie jak czułość, swoistość itp., które pozwalają na zwizualizowanie jakości modelu na krzywej ROC. Jednakże do tego celu należy wyliczyć wiele punktów, a dokonuje się tego poprzez badanie powyżej opisanych właściwości w zależności od przyjętego punktu odcięcia (Fawcett, 2005).

W procesie predykcyjnym wyznaczane są prawdopodobieństwa wejścia w opóźnienie w spłacie kredytu, a decyzja o tym czy dana osoba zostanie zaklasyfikowana jako wystarczająco wiarygodna zależy od przyjętego poziomu akceptacji. Jeśli dla pewnego wnioskodawcy prawdopodobieństwo wejścia w opóźnienie w spłacie zostało wyznaczone na 35%, a próg odcięcia założono na poziomie 40%, kredyt zostanie udzielony. Jednakże w przypadku zdefiniowania punktu podziału równego 30%, wniosek zostanie odrzucony. W celu stworzenia wykresu jakości modelu należy wyliczyć wartości macierzy pomyłek dla wielu progów odcięcia (Fawcett, 2005). Po wykonaniu tych operacji możemy wyznaczyć wartość współczynnika Giniego dla rozpatrywanego algorytmu, który wynosi dwukrotność pola pod krzywą ROC, a nad krzywą klasyfikatora losowego (na rysunku 7 oznaczona jako "Random classifier").

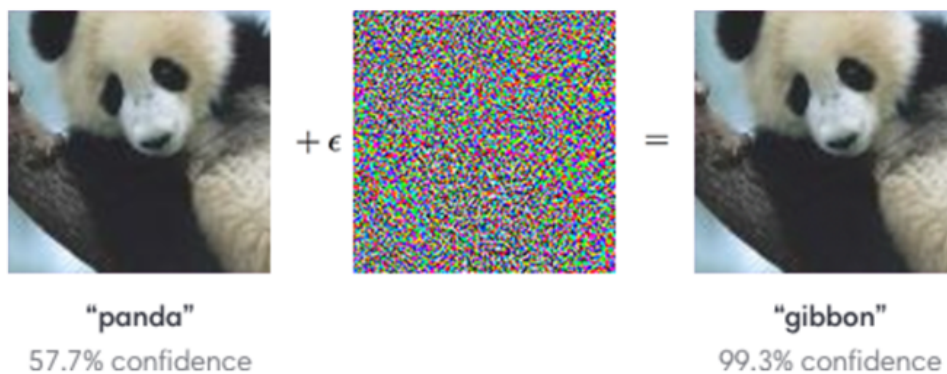
Wracając do Krzywej Profit (rysunek 6), łatwo zauważyć, że im skuteczniejszy model (im wyższy współczynnik Giniego), tym wykres jest bardziej wybrzuszony, co oznacza możliwość osiągnięcia wyższych zysków. Jako że straty powodowane przez jednego dłużnika potrafią przewyższyć profity uzyskiwane przez bank we współpracy z wieloma wiarygodnymi klientami, bardzo istotny jest dobór odpowiedniego progu akceptacji. Przytoczony wykres, dla najlepszego z modeli, sugeruje wybór progu odcięcia pomiędzy 40%, a 46% (Przanowski, 2023).

2.2 Charakterystyka wrogiego uczenia maszynowego

Adversarial Machine Learning, tłumaczone na język polski jako kontrydiktoryjne uczenie maszynowe lub wrogie uczenie maszynowe to dziedzina, która koncentruje się na opracowywaniu modeli uczenia maszynowego odpornych na ataki kontrydiktoryjne, stosowane do oszukiwania lub manipulowania tymi modelami poprzez imputację złośliwych lub wprowadzających w błąd danych podczas faz uczenia lub wnioskowania. Ataki te mogą mieć poważne konsekwencje, od kradzieży poufnych informacji po nieprawidłowe działanie krytycznych systemów, takich jak samojezdne samochody lub urządzenia medyczne.

Przeciwnicy (w języku angielskim nazywani są jako „attackers”, tłumaczone na język

polski również jako „napastnicy”, czy też ”adwersarze”) mogą wprowadzać dane, które mają za zadanie zmanipulować rezultaty wyjściowe, wykorzystując luki w modelu. Nie jesteśmy w stanie ich zidentyfikować ludzkim okiem, choć mogą powodować nieprawidłowe działanie modelu. W systemach sztucznej inteligencji występują różne formy danych, takie jak tekst, pliki audio, obrazy. Dużo łatwiej jest przeprowadzać ataki cyfrowe, takie jak manipulowanie tylko jednym pikselem w obrazu wejściowych, co może prowadzić do błędnej klasyfikacji(zephyrnet.com, 2022).



Rysunek 8: Przykład ataku na system rozpoznawania obrazów (openai.com, 2017)

Przykład manipulacji obrazu przedstawiono na rysunku 8. W tym przypadku zaatakowano system rozpoznawania zwierząt, nauczony na bazie pewnej puli zdjęć. Przed atakiem, model rozpoznał, że na fotografii znajduje się panda, określając to z pewnością bliską 58%. Po dodaniu szumu, zmanipulowano system do tego stopnia, iż z niemal 100% przekonaniem sklasyfikował zdjęcie pandy jako gibbona(openai.com, 2017). Łatwo zauważyć, że jednym zaburzeniem napastnik zmienił status modelu z przydatnego na bezużyteczny.

O ile zmanipulowanie systemu rozpoznawania obrazów zwierząt może wydawać się niegroźne i niedostatecznie ukazywać niebezpieczeństwo płynące z tego typu ataków, o tyle wpływ napastników np. na działanie samochodów autonomicznych wskazuje na tragiczne skutki jakie mogą zostać spowodowane. Jednym z mniej skomplikowanych algorytmów uczenia maszynowego zastosowanych w tych środkach transportu jest system rozpoznawania znaków drogowych, jako że ich liczba jest skończona i względnie nieduża, a ich kształt, kolor i rozmiar jest ściśle znormalizowany. Dla przykładu rozważmy typową sytuację drogową.

Na rysunku 9 zamieszczono widok z przedniej kamery samochodu autonomicznego, tuż przed skrzyżowaniem. Auto bez problemu rozpoznaje znak STOP, a następnie wykonuje odpowiednie czynności, aby zatrzymać się przed skrzyżowaniem. Jednakże bardzo



Rysunek 9: Widok z kamery przedniej samochodu autonomicznego. Właściwie rozpoznany znak STOP (Surma, 2020)

łatwo jest wprowadzić system w błąd, co może wydarzyć się na skutek zabrudzenia znaku, czy pomalowania go farbą, czego przykład przedstawiono na rysunku 10 (Surma, 2020).

Wskutek tego typu zaburzenia danych wejściowych, system oparty na uczeniu maszynowym może nie tylko nie rozpoznać tego znaku jako nakaz zatrzymania się, a wręcz może przypisać do otrzymanego obrazu zupełnie inny znak, mający w danym momencie krytyczne znaczenie dla bezpieczeństwa kierowcy. Na rysunku 11 przedstawiono przykładową interpretację przez model, gdzie zabrudzony znak STOP został przyjęty jako znak pierwszeństwa (Surma, 2020).

Zakładając, że aby dotrzeć do celu kierowca na danym skrzyżowaniu musi skrócić w lewo, samochód bez zatrzymywania się przejedzie przez to skrzyżowanie. Skutki takiej decyzji z wysokim prawdopodobieństwem będą tragiczne, co ukazuje jak duże znaczenie ma jakość danych dostarczanych do modelu i jak niewielkie zaburzenie może wpłynąć na jego niezawodność, a co za tym idzie, uzasadnienie dalszego wykorzystania zaprojektowanego narzędzia w biznesie (Surma, 2020).



Rysunek 10: Widok z kamery przedniej samochodu autonomicznego. Niewłaściwie rozpoznany znak STOP (Surma, 2020)



Rysunek 11: Widok z kamery przedniej samochodu autonomicznego. Znak STOP błędnie rozpoznany jako znak bezwzględnego pierwszeństwa przy skrócie w lewo (Surma, 2020)

Jednym z głównych czynników blokujących wykorzystanie sztucznej inteligencji w coraz to istotniejszych aspektach życia jest jej wrażliwość na wrogą ingerencję. Opisany powyżej przypadek obrazuje problemy z jakimi spotykają się współcześni modelarze systemów uczących się. Wyzwaniem najbliższych lat jest uczynienie tego typu narzędzi bezpiecznymi dla codziennego użytku. Wciąż trwa wypracowywanie najlepszych technik wzmacniania niezawodności uczenia maszynowego, jak również definiowane są tzw. dobre praktyki, będące tymczasową odpowiedzią na niemoc w niektórych obszarach.

Badacze z instytutu naukowo-badawczego w Albuquerque jako cel obrali sobie stworzenie w pełni wiarygodnego i skutecznego sposobu na obronę przed wrogą ingerencją w modele uczenia maszynowego. Aby tego dokonać, przeprowadzili szereg badań nad wieloma zbudowanymi przez nich sieciami neuronowymi ukierunkowanymi na typowe dla tej tematyki rozpoznawanie obrazów. W pierwszym kroku wytrenowali modele próbując osiągnąć możliwie najwyższą skuteczność. Kolejnym etapem było obniżanie ich jakości poprzez "zatrutowanie" danych stosowanych do uczenia tychże modeli. Finalnie porównywano stopień degradacji skuteczności klasyfikatorów (Short, Pay, & Gandhi, 2019).

W celu podniesienia odporności modeli, stosuje się trening kontradyktoryjny, polegający na wprowadzaniu do modelu mylących danych, mających za zadanie wprowadzenie algorytmu w błąd, jednakże tego typu informacje, wprowadzone na etapie uczenia systemu, cechują się potencjałem do zmniejszania jego wrażliwości na ataki. Zatem można przyjąć, że poza technikami defensywnymi, równie szerokim zagadnieniem jest tworzenie strategii ataku. W literaturze znajdują się odniesienia do kilku głównych metod kreacji wrogich danych. Niektóre z nich to (Short et al., 2019):

- Fast Gradient Sign Method - celem trenowania modelu jest minimalizacja funkcji

błędu, natomiast poprzez wprowadzenie FGSM zwiększana jest wartość tej funkcji;

- Basic Iterative Method - metoda będąca wprost rozwinięciem techniki FGSM, polegająca na wielokrotnym, aczkolwiek jasno wcześniej zdefiniowanym, jej powtórzeniu;
- Metoda Carliniego Wagnera (C&W) - technika generowania wrogich danych, skupiająca się na maksymalizacji podobieństwa między oryginalnymi informacjami wejściowymi, a zmanipulowanymi, zachowując efekt błędnej klasyfikacji przez model.

Naukowcy z Albuquerque wskazują wysoki potencjał zauważony w metodzie C&W. Mimo wielu prób, badacze nie byli w stanie zastosować skutecznej obrony przed wrogimi danymi wygenerowanymi w ten sposób (Short et al., 2019).

2.3 Typy ataków na algorytmy uczenia maszynowego

Na daną chwilę większość algorytmów proponowanych przez badaczy, naukowców i specjalistów z branży R&D skupia się głównie na wysokiej wydajności i niskiej liczbie błędnych klasyfikacji. Jednakże nawet gdy wskazane cele zostają osiągnięte, modele te często nie powinny być implementowane w środowiskach produkcyjnych, zwłaszcza w domenach krytycznych, które mogą mieć wpływ na życie znacznej części społeczeństwa, nie uwzględniając innych kryteriów i wymagań dotyczących sztucznej inteligencji. Są nimi: bezpieczeństwo algorytmów, ich interpretowalność i uczciwość. Co więcej, rezultaty osiągane są na danych, które odpowiednio przygotowano w warunkach laboratoryjnych i możliwe jedynie w sytuacji gdy implementacja również zachodzi w takich warunkach (Pawlicki, 2020).

Zastosowanie sztucznej inteligencji na wielką skalę stało się rzeczywistością, za czym idzie świadomość, że bezpieczeństwo algorytmów uczenia maszynowego wymaga natychmiastowej uwagi. Adwersarze potrafią starannie dobrać próbki danych wejściowych, aby zmieniały wyniki klasyfikacji w oczekiwany przez nich sposób. Świadomość zagrożeń związanych z ich użyciem, a także ich podatność na ingerencje jest wciąż dość niewielka (Pawlicki, 2020), jednakże już powstały definicje określające poszczególne typy ataków.

Jednym z najbardziej znanych podziałów jest klasyfikacja ze względu na dostęp do modelu (zephyrnet.com, 2022):

- Atak białoskrzynkowy - odnosi się do sytuacji, w której atakujący ma pełny dostęp do modelu docelowego. Obejmuje to architekturę i parametry, które pozwalają im tworzyć próbki danych na modelu docelowym. Osoby atakujące będą miały ten

dostęp tylko wtedy, gdy testują model jako programista. Mają oni detaliczną wiedzę na temat architektury sieci oraz znają tajniki modelu i tworzą strategię ataku;

- Atak czarnoskrzynkowy - odnosi się do sytuacji, w której atakujący nie ma dostępu do modelu docelowego i może jedynie zbadać dane wyjściowe.

Podział ataków na czarnoskrzynkowe i białoskrzynkowe oparty jest o umiejscowienie atakującego. Inna klasyfikacja bazuje na strategii ingerencji w model. Atak zatruwający (typu poisoning) skupia się na danych ze zbioru uczącego. Atakujący zmienia istniejące lub wprowadza nieprawidłowo oznakowane dane. Wskutek takiego działania, model przeszkolony na „zatrutym” zbiorze będzie tworzył błędne predykcje na prawidłowo oznakowanych danych(towardsdatascience.com, 2021). W literaturze znaleźć można kilka artykułów na temat ataków tego typu. W jednym z nich, autorzy opisują użycie tzw. wrogiego szumu etykiet (ang. adversarial label noise). W tym artykule przedstawiona jest metoda wykorzystująca sposób działania algorytmu Support Vector Machines (ang. Metoda Wektorów Nośnych), którego działanie polega na mapowaniu danych na wielowymiarową przestrzeń właściwości w sposób umożliwiający kategoryzację punktów danych(ibm.com, 2021).

Ogólnym założeniem ataku jest wprowadzenie do zbioru treningowego próbki, która znacząco zmieni wynik klasyfikacji, obniżając skuteczność modelu. Taką próbkę można stworzyć poprzez rozwiązanie problemu optymalizacyjnego, polegającego na wyszukiwaniu lokalnych maksimów powierzchni funkcji błędu, do czego wykorzystano algorytm gradient ascent. Atak wykorzystuje odwracanie etykiet konkretnych próbek w klasyfikacji binarnej zbioru uczącego, przy założeniu, że dane w zbiorze walidacyjnym nie są w żaden sposób zmieniane(Biggio, Nelson, & Laskov, 2012).

Ataki unikowe (typu evasion), w odróżnieniu od ataków zatruwających, nie skupiają się na danych używanych do uczenia modelu, lecz na odpowiedniej manipulacji danymi wejściowymi, dla których model wydaje prognozowany rezultat. Polegają one na modyfikowaniu danych, aby wydawały się uzasadnione, lecz by prowadziły do błędnej prognozy. Przykładem wykorzystania tego typu ataków są modele oceny wiarygodności kredytowej. Ubiegając się o kredyt, osoba atakująca może zamaskować swój prawdziwy kraj pochodzenia za pomocą usługi VPN, ukrywając w ten sposób np. iż jest obywatelem kraju uznawanego przez model jako bardziej ryzykowny, co mogłoby zmniejszyć jego szanse na pozytywną ocenę jego wniosku(towardsdatascience.com, 2021).

Innym kierunkiem wykorzystania ataków unikowych są modele służące do odfiltrowy-

wania wiadomości e-mail będących spamem. Ich podejście może polegać na eksperymentowaniu z mailami, które model już wytrenował w zakresie sprawdzania i rozpoznawania jako spam. Jeśli model został wyszkolony do filtrowania wiadomości e-mail zawierających konkretne słowa, atakujący może tworzyć nowe e-maile zawierające powiązane z tym słowa, które przejdą przez algorytm, co spowoduje, że wiadomość e-mail, która w typowym procesie zostałaby sklasyfikowana jako spam, spamem nie jest, co w oczywisty sposób pogarsza skuteczność modelu(zephyrnet.com, 2022).

Atak poszukiwawczy (typu exploratory) może wystąpić po wytrenowaniu algorytmu, a jego zadaniem jest odkrywanie informacji o wewnętrznym działaniu modelu, w celu identyfikacji słabych punktów. W tym podejściu ingerencja jest ukierunkowana na poszukiwanie informacji o(Shi, Sagduyu, & Grushin, 2017):

- granicy decyzyjnej używanej przez algorytm (np. hiperpłaszczyzny maszyny wektorów nośnych (SVM) algorytm);
- ogólnym zestawie reguł, którymi kieruje się algorytm;
- zestawie logicznych lub probabilistycznych właściwości algorytmu;
- danych, które zostały wykorzystane (lub nie wykorzystane) do uczenia algorytmu.

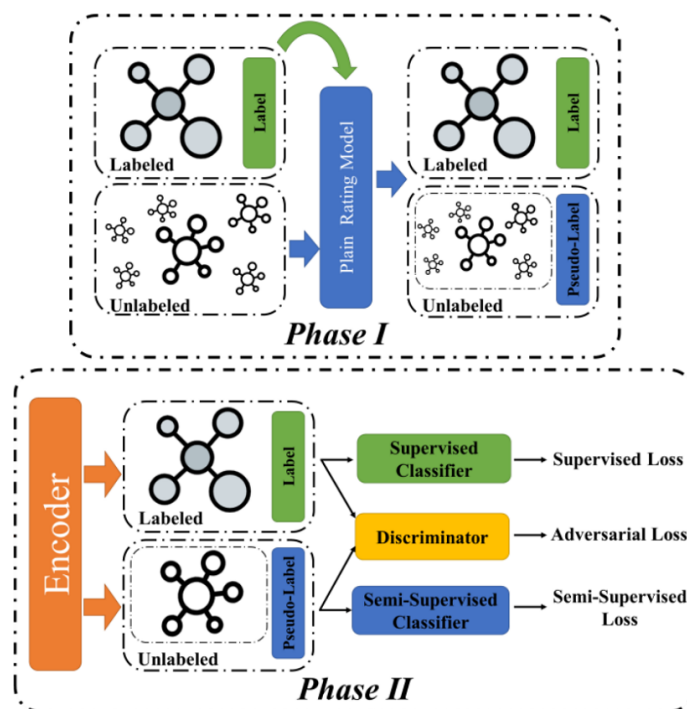
W ostatnich latach powstało kilka prac, badających ataki na Głębką Sieć Neuronową (DNN). W artykule z 2017 roku(Shi et al., 2017) naukowcy zbudowali tzw. funkcjonalny ekwiwalent klasyfikatorów modelu DNN, opierając się na algorytmach SVM oraz naiwnego klasyfikatora Bayes’a. Z kolei w badaniu z 2019 roku(Shi, Sagduyu, Davaslioglu, & Li, 2019) opisano jak przy pomocy Głębokiej Sieci Neuronowej można wydobyć klasyfikator wdrożony w warunkach produkcyjnych (Pawlicki, 2020).

Jak zatem zabezpieczyć się przed wrogimi ingerencjami w model? Jedną z możliwości stanowi tzw. trening kontradyktoryjny, będący jednym z podejść do poprawy wydajności i bezpieczeństwa systemów uczenia maszynowego. Polega on na kontrolowanym atakowaniu modelu na wiele sposobów, znajdując w ten sposób jego „czułe punkty”. Na skutek takich badań, modelarze mogą zweryfikować jego odporność na wrogie ataki, a następnie podjąć odpowiednie kroki celem poprawy jego bezpieczeństwa(zephyrnet.com, 2022).

Chińscy badacze postanowili wykorzystać zjawisko wrogiego uczenia maszynowego do wyznaczania korporacyjnych ratingów kredytowych. Dotychczas wykorzystanie standardowych metod predykcyjnych nie przynosiło satysfakcjonujących rezultatów, a ma na to

wpływ specyfika rynku. Zatrudnienie i opłacenie zespołu specjalistów, których zadaniem będzie oszacowanie stopnia wiarygodności firmy nie jest małym wydatkiem i mogą sobie na nie pozwolić jedynie duże korporacje. Implikuje to względnie niedużą próbę obserwacji i możliwość ich uzasadnionego wykorzystania jedynie dla podobnych instytucji, jak te posiadające swój rating. Większość średnich i praktycznie wszystkie małe przedsiębiorstwa nie są w stanie otrzymać własnego rate'u, a co za tym idzie, potencjalny model nie ma na czym nauczyć się, a potem przewidywać wiarygodność tych firm(Feng & Xue, 2021).

Naukowcy zaproponowali proces uczenia modelu, zobrazowany na rysunku 12, rozpoczynający się od podzielenia danych na oznaczone(ang. labeled data) - firmy ze znanym rating'iem, a także nieoznaczone(ang. unlabeled data) - firmy bez rating'u. Na danych oznaczonych nauczono model gradientowo wzmocnionego drzewa decyzyjnego, który następnie wykorzystano do wyznaczenia najbardziej prawdopodobnych rating'ów dla obserwacji nieoznaczonych. W ten sposób otrzymano pełen zestaw niewybrakowanych danych, jednakże kluczowym elementem było zakwalifikowanie ocen wyliczonych przez model jako 'pseudooceny' - rate'y z ograniczonym zaufaniem(Feng & Xue, 2021).



Rysunek 12: Schemat procesu kontradyktoryjnego uczenia częściowo nadzorowanego dla korporacyjnego ratingu kredytowego (Feng B., 2021)

Proces ten można porównać do nauki ucznia w szkole. W trakcie lekcji, otrzymuje on zestaw zadań i poprawnych odpowiedzi, które dostarcza nauczyciel. Jednakże podczas

sprawdzianu, gdy pojawiają się zadania nieco inne niż podczas zajęć, podejrzenie odpowiedzi u kolegi daje informację o tym jak do danego pytania ustosunkowała się inna osoba, ale nie musi to oznaczać, że rozwiązanie ”ściągnięte” od sąsiada jest zgodne z rzeczywistością, więc trzeba je traktować z ograniczonym zaufaniem(Feng & Xue, 2021).

W drugim etapie na przygotowanych danych zastosowano podejście, polegające na ograniczeniu zaufania do danych wypredykowanych z obserwacji nieoznaczonych, dzięki czemu mogą one brać udział w analizie. Na podstawie danych oznaczonych oraz nieoznaczonych obliczono klasyfikator nadzorowany i klasyfikator częściowo-nadzorowany, a dodatkowo wyznaczono kontradyktoryjną stratę między zbiorami co pozwoliło na zastosowanie obydwu zbiorów jako całości i wyznaczenie modelu opisanego jako ASSL4CCR - kontradyktoryjne uczenie częściowo nadzorowane dla korporacyjnego ratingu kredytowego (Feng & Xue, 2021).

Model	Recall	Accuracy	F1-score
LR	0.76250	0.80970	0.81946
SVM	0.83750	0.89247	0.88961
MLP	0.91406	0.93568	0.93245
Xgboost	0.92343	0.94225	0.94133
CCR-CNN	0.92812	0.95253	0.94518
CCR-GNN	0.93437	0.95012	0.95177
ASSL4CCR	0.95321	0.96115	0.96252

Tabela 3: Skuteczność klasyfikacji dla poszczególnych modeli sprawdzanych w badaniu korporacyjnych ratingów kredytowych(Feng B., 2021)

Chińscy naukowcy, jako zwieńczenie projektu, zrealizowali porównanie podstawowych parametrów pozwalających na obiektywną ocenę jakości poszczególnych modeli tj. recall, dokładność oraz F1-score. Z analizy wyników przedstawionych w tabeli 3 wynika, że model ASSL4CCR osiąga najlepsze rezultaty, co dowodzi przydatności świadomego wykorzystania wrogiego uczenia maszynowego w praktyce(Feng & Xue, 2021).

2.4 Przykłady ataków na algorytmy uczenia maszynowego

Wrogie uczenie maszynowe jest względnie nową poddziedziną analizy danych i na tę chwilę nie łatwo jest znaleźć wzięte z życia przykłady ingerencji w rzeczywiste, używane produkcyjnie modele uczenia maszynowego. Dzięki wysokiej świadomości analityków nie trzeba uczyć się na nieświadomie popełnionych błędach następnie wykorzystanych przez adversarzy, a problem został zidentyfikowany zanim pojawiły się jego potencjalnie fatalne skutki. W związku z powyższym, w ostatnich latach, naukowcy prowadzą intensywne ba-

dania, samodzielnie generując różne scenariusze ingerencji w model, jednakże wciąż pozostaje w tej dziedzinie wiele przestrzeni dla nowych odkryć.

Obecnie lwia część społeczeństwa posiada skrzynkę mailową, a coraz częściej jedna osoba posiada kilka takich komunikatorów. Łatwość tworzenia nowych kont i docierania do innych użytkowników na masową skalę, wciąż stanowi bardzo popularny sposób nie tylko do porozumiewania się w ważnych sprawach tj. komunikacja służbowa czy też potwierdzenia transakcji internetowych, lecz także daje możliwość przesyłania ofert i reklam. Wśród takich wiadomości łatwo jest umieszczać niebezpieczne linki, kierujące nieświadomych odbiorców do domen stanowiących zagrożenie dla ich prywatności. Według raportu Message Labs Intelligence pośród całej globalnej komunikacji mailowej, wiadomości typu spam stanowią aż 88%(Kuchipudi, Nannapaneni, & Liao, 2020). W celu uniknięcia podejrzanych wiadomości oraz zalewaniu skrzynki przez nachalną propagandę reklamową stosuje się tzw. filtry antyspamowe oparte na algorytmach uczenia maszynowego.

Algorytmy uczone są wykrywania "wrogich" słów w zawartości e-maili. W zależności od modelu różne słowa w zróżnicowanym stopniu obniżają lub podnoszą poziom zaufania modelu do danej wiadomości, a po przeprowadzeniu oceny podejmowana jest decyzja o umieszczeniu jej w skrzynce odbiorczej lub folderze spam. Na polu filtracji niechcianych wiadomości toczona jest ciągła walka między deweloperami, a adversarzami mającymi na celu oszukać model, aby dokonał błędnej klasyfikacji(Cheng, Xu, Li, & Ding, 2022).

Jednym ze sposobów wykorzystywanych przez napastników jest zaszycie w wiadomości wielu silnie zaufanych słów, które przeważają ocenę klasyfikatora z negatywnej na pozytywną. Jednakże umieszczenie wielu słów z bardzo wąskiej i zróżnicowanej grupy może wypaczyć przekaz wiadomości do tego stopnia, że będzie ona łatwo wykrywalna nawet dla ludzkiego oka, a potencjalna ofiara nie da się nabrać na atak. Pomysłów adversarzy na uniknięcie takiej sytuacji jest wiele i odwołują się one głównie do ich kreatywności, jak np. wpisanie kilkudziesięciu zaufanych słów w zawartość maila, stosując dla nich kolor czcionki identyczny z barwą tła, co skutecznie ukrywa podejrzaną zawartość przed użytkownikiem, czy też umieszczenie literówek w słowach znacznie obniżających prognozowaną przez algorytm ocenę takiej zawartości, dzięki czemu uzyskują one wagę neutralną i nie są wychwytywane przez filtry jako niebezpieczne(Cheng et al., 2022).

Zespół naukowców z Uniwersytetu Michigan postanowił przeprowadzić badania mające na celu sprawdzić skuteczność ataków na algorytmy anty-spamowe. Wykorzystali do

tego celu trzy względnie proste techniki(Kuchipudi et al., 2020):

- synonym replacement (ang. podmiana słowa przy pomocy synonimu);
- ham word injection (ang. wprowadzenie zaufanych słów);
- spam word spacing (ang. wprowadzenie przerw/spacji pomiędzy literami w słowach związanych ze spamem).

Modelem, którego skuteczność do obrony badano, był algorytm naiwnego klasyfikatora Bayesa. Badacze argumentują wybór tej techniki implementacji jej popularnością w tego typu zastosowaniach, co empirycznie wykazało jej wysoką skuteczność w wychwytywaniu wrogich, bądź też niechcianych wiadomości (Kuchipudi et al., 2020).

Naukowcy wskazują również trudności, jakie mogą wystąpić przy próbie tego typu ataków. Główną przeszkodą może być "czarnoskrzynkowość", czyli ograniczony lub całkowicie uniemożliwiony dostęp do informacji na temat wybranego i nauczonego modelu filtracji antyspamowej, jak również nieznaną ilość danych wejściowych użytych do jego budowy. Jednakże warto zauważyć, że badania wykazują, iż do przeprowadzenia udanego ataku na algorytmy uczenia maszynowego filtrów spam, często wystarczy znajomość zaledwie 1% danych treningowych, co ułatwia zadanie adversarzy(Kuchipudi et al., 2020).

W pierwszym podejściu, zastosowano synonimy słów, wychwytywanych przez filtr jako niebezpieczne, nie zmieniając przy tym sensu samej wiadomości. Wykorzystano do tego celu technikę NLP - Natural Language Processing(ang. przetwarzanie języka naturalnego) - będącą poddziedziną sztucznej inteligencji i odpowiedzialną za rozumienie języka ludzkiego przez maszyny i roboty(Castagno, 2020). Na łamach pracy, jako przykład przetworzenia wiadomości niebezpiecznej zmanipulowanej w sposób pozwalający przejść przez filtrację podano zdanie: "Ringtone Club: Get the UK singles chart on your mobile each week and choose any top quality ringtone! This message is free of charge." Różnie zmodyfikowane wiadomości wraz z rezultatem ich klasyfikacji przez model antyspamowy przedstawiono w tabeli 4(Kuchipudi et al., 2020). Łatwo zauważyć, że wiadomość, która została przepuszczona przez klasyfikator jako zaufana, zapewne nie zostałaby potraktowana poważnie przez użytkownika, co wskazuje na wdrożenie niezbędnych poprawek w zestawach synonimów, wykluczając te mające wpływ na kontekst informacji.

Drugie podejście opiera się na manipulacji częstotliwością pojawiania się słów zaufanych. Wśród publicznie dostępnych zbiorów można znaleźć zestawy słów znanych jako

Modified Message	Cosine Similarity	Prediction
Ringtone Club: acquire the UK single graph on your Mobile_River each hebdomad and take any top_side caliber ringtone! This content is free_people of charge.	0.583	spam
Ringtone Club: become the UK bingle graph on your nomadic each workweek and select any upper_side caliber ringtone! This subject_matter is liberate of charge.	0.583	spam
Ringtone Club: go the UK one graph on your peregrine each calendar_week and pick_out any upside character ringtone! This substance is release of charge.	0.583	ham

Tabela 4: Wrogie próbki zastosowane na atakowanym modelu filtra antyspamowego (Kuchipudi et al., 2020)

słowa związane ze zjawiskiem spamu i z wysokim prawdopodobieństwem generujące uruchomienie filtra. W związku z powyższym, jako zaufane należy traktować wyrazy, które nie znajdują się w tym zbiorze, dzięki czemu wprowadzenie ich do zawartości wiadomości podnosi prawdopodobieństwa przejścia przez filtrację (Kuchipudi et al., 2020).

Przykładowa wiadomość klasyfikowana początkowo jako spam: "Congratulations ur awarded 500 of CD vouchers or 125gift guaranteed and Free entry 2 100 wkly draw txt MUSIC to 87066 TnCs www.Ldew.com1win150ppmx3age16", po wprowadzeniu kilkakrotnie słów zaufanych tj. good, love, deal, jest w stanie przejść przez filtr jako zaufana: "Congratulations good ur awarded good 500 of CD vouchers or 125 good gift guaranteed love and Free entry 2 good 100 wkly draw txt MUSIC to 87066 TnCs www.Ldew.com1win150ppmx3age16 good good good good good deal" (Kuchipudi et al., 2020).

Podczas badań tego podejścia zauważono pewną właściwość filtrów antyspamowych. Z testów można wnioskować, że model jest wyczulony na używanie skrótów w mailach tj. użycie "U" zamiast "you" (ang. Ty) czy też "R" w miejsce "are" (ang. jesteś/jest/jesteśmy/jesteście/są). Wynika z tego, że stosowanie bardziej zadbanego i oficjalnego języka tekstu daje większą szansę na ominięcie filtracji i skuteczne umieszczenie potencjalnie wrogiej wiadomości w atakowanej skrzynce odbiorczej (Kuchipudi et al., 2020).

Trzecia technika polegała na wprowadzeniu odstępów między znakami w słowach prawdopodobnie klasyfikowanych jako niebezpieczne. Wysoki współczynnik podobieństwa, przy pomocy którego badano podobieństwo znaczeniowe wiadomości oryginalnej

i zmanipulowanej, wskazuje iż w tym przypadku treść jest najmniej zaburzona względem odniesienia i najlepiej oddaje sens zamierzony przez autora. W przytoczonym przykładzie edytowanie słów "sexy" i "flirt" do form "s e x y" oraz "f l i r t" skutkuje oszukaniem modelu i uznaniem wiadomości za zaufaną (Kuchipudi et al., 2020).

W podsumowaniu pracy można znaleźć konkluzję, iż przy wykorzystaniu trzech opisanych powyżej sposobów potrafiąco w około 60 % przypadków oszukać filtr antyspamowy. W trakcie badań wykorzystano zbiór danych z witryny Kaggle, zawierający 5572 wiadomości, z czego 747 zostały oznaczone jako spam. Wielkość zestawu jest wystarczająca do budowy takiego mechanizmu, jednakże zdecydowanie pewniejszy model można byłoby uzyskać przy zebraniu nieco większej próbki do jego nauki, co przyczyniłoby się do zebrania bardziej wiarygodnych wniosków (Kuchipudi et al., 2020).

Na łamach pierwszych dwóch rozdziałów opisano tematykę scoringu kredytowego, jak również wrogiego uczenia maszynowego. Rozdziały 3 i 4 stanowią będą praktyczne połączenie tych dwóch zagadnień.

3 Budowa modelu drzewa decyzyjnego dla scoringu kredytowego

3.1 Analiza oraz wstępne przetworzenie wybranego zbioru danych

Wybór zbioru danych jest jednym z najtrudniejszych, a jednocześnie najważniejszych etapów budowy modelu predykcyjnego. To na tym zestawie informacji algorytm decyzyjny uczy się kluczowych właściwości, które dają mu możliwość odpowiedniej klasyfikacji. W przypadku scoringu kredytowego, uzyskanie dostępu do obszernych zbiorów rzeczywistych danych jest często niemożliwe. Właścicielami tego typu zestawów informacji najczęściej są banki, które niechętnie dzielą się takimi starannie utworzonymi zbiorami, bądź też argumentują swoją odmowę wrażliwością tychże danych oraz restrykcyjnymi zasadami narzuconymi przez Nadzorcę (w przypadku Polski mowa o Komisji Nadzoru Finansowego).

W przypadku zestawów dostępnych na publicznych witrynach internetowych jak np. Kaggle.com, można na nich zbudować model predykcyjny, jednakże należy zadać sobie pytanie odnośnie jego rzetelności i skuteczności. Często zebrane tam zbiory nie są dostatecznie duże, cechują się niską zawartością klientów, którzy weszli w opóźnienie w spłacie kredytu, bądź też są to dane symulacyjne i nierzeczywiste. Dzięki życzliwości Pana dr Karola Przanowskiego, pełniącego rolę nauczyciela akademickiego w Zakładzie Metod Statystycznych i Analiz Biznesowych na Szkole Głównej Handlowej w Warszawie, który udzielił zgody na wykorzystanie dużego i rzetelnego zbioru na potrzeby tej pracy. Rozważany zestaw danych wykorzystywany jest do celów dydaktycznych np. do budowy karty scoringowej w ramach przedmiotu "Credit Scoring - automatyzacja procesu biznesowego".

Zbiór zawiera 219 zmiennych, zarówno ciągłych, jak i kategoriowych. Umieszczono w nim 68499 obserwacji, gdzie każda dotyczy innej aplikacji o produkt gotówkowy lub ratalny. Tak duży zbiór uniemożliwia pełne opisanie każdej ze znajdujących się w nim zmiennych, zatem szerzej przeanalizowano jedynie potencjalnie najważniejsze z nich lub opisano grupy tychże zmiennych. Pełną dokumentację wszystkich zmiennych zamieszczono w pliku Excel dostępnym pod linkiem https://github.com/MateuszKuchta88/Master_Diploma/blob/main/PracaTresc/VariablesDescription.xlsx, natomiast wspomniany opis, wraz z przykładowymi wartościami przedstawiono w tabeli 5.

Nazwa zmiennej	Opis	Przykładowe wartości
cid	Identyfikator klienta	0000024576
aid	Identyfikator rachunku/wniosku kredytowego	css1970022600002
product	Typ produktu, o który aplikuje klient	'css', 'ins'
period	Sześciocyfrowa liczba wskazująca na moment aplikacji o dany produkt, gdzie pierwsze cztery cyfry oznaczają rok, a pozostałe dwie informują o miesiącu	197304
act_...	Zmienne bieżące, zebrane z niezależnego źródła (tj. BIK) opisujące sytuację klienta w momencie aplikowania o produkt	wiek, suma zaciągniętych zobowiązań, liczba spłaconych kredytów
act_age	Wiek klienta w latach	54
act_cus_active	Zmienna binarna informująca o tym, że klient miesiąc temu miał aktywną pożyczkę (był w trakcie jej spłacania)	'1' lub brak danych
act_ccss_cc	Zdolność kredytowa klienta określana stosunkiem sumy raty i wydatków do przychodów	0.45
app_...	Zmienne aplikacyjne, deklarowane przez klienta w trakcie składania aplikacji	płeć, miejsce zamieszkania, stan cywilny
app_income	Zarobki klienta	1512
app_spendings	Wydatki klienta	500 \$
app_char_cars	Informacja czy klient posiada auto	'Owner', 'No'
agr_...	Zmienne behawioralne, wyliczane jako agregaty z 3, 6, 9, lub 12 ostatnich miesięcy, bez uwzględniania braków danych	minimalna, maksymalna oraz średnia liczba dni spóźnienia w spłacie kredytu ratalnego w ciągu ostatnich 9 miesięcy
ags_...	Zmienne behawioralne, wyliczane jako agregaty z 3, 6, 9, lub 12 ostatnich miesięcy, z uwzględnieniem braków danych	minimalna, maksymalna oraz średnia liczba dni spóźnienia w spłacie kredytu gotówkowego w ciągu ostatnich 6 miesięcy
default3, default6, ...	Binarna zmienne informująca o wystąpieniu zdarzenia wejścia klienta w opóźnienie w spłacie równe co najmniej 90 dni w ciągu x miesięcy od zaciągnięcia zobowiązania	'1' lub '0'

Tabela 5: Opis zmiennych oraz grup zmiennych w zbiorze danych (źródło opracowanie własne)

Realizowanie kodu rozpoczęto od wczytania zbioru danych, którym jest plik SAS-owy, o rozszerzeniu sas7bdat, dostępny pod linkiem https://github.com/MateuszKuchta88/Master_Diploma/blob/main/PracaTresc/abt_app.sas7bdat. Zastosowano funkcję `read_sas` z biblioteki `pandas`. W zbiorze danych znajdują się informacje o wielu klientach, którzy wielokrotnie aplikują o dwa różne produkty - kredyt gotówkowy oraz kredyt ratalny. Różnią się one nie tylko pożyczanymi kwotami, ale przede wszystkim, mają innych klientów docelowych. Kredyty ratalne związane są np. z kupnem łódówki, czy laptopa, a banki najczęściej nie zarabiają na tych produktach, oferując zerowe rzeczywiste stopy procentowe. Taka operacja ma na celu pozyskanie klienta celem zebrania informacji takich jak zmienne behawioralne. W przypadku, gdy klient sumiennie spłacał zobowiązanie w odpowiednim czasie, taka informacja jest odnotowywana, a kredytobiorca staje się celem dla ofert kredytów gotówkowych. W przeciwnym przypadku, klient staje się niewiarygodny, a taka informacja również zawiera się w bazie danych pożyczkobiorców instytucji finansowych. Banki zarabiają głównie na "gotówce", natomiast kredyty ratalne w najlepszym przypadku nie przynoszą strat.

W związku z powyższym, do dalszej analizy należało wybrać jeden z tych produktów. Opierając się na potrzebie realizowania wyższych zysków, wybrano kredyt gotówkowy, czego efektem było zmniejszenie liczby obserwacji do 34188.

W opracowywanym zbiorze zbadano rozkład zmiennej binarnej 'default12', reprezentującej wystąpienie zdarzenia spóźnienie się ze spłacaniem zobowiązania sumarycznie przez co najmniej 90 dni w trakcie pierwszych 12 miesięcy od wypłacenia kredytu, która została wybrana jako modelowana zmienna celu.

Podczas analizy rozkładu wartości zmiennej `default12` wychwycono 4411 obserwacji z brakami danych w zmiennej celu. Ze względu na dostatecznie dużą liczbę obserwacji bez braków danych, podjęto decyzję o nie stosowaniu technik imputacji danych i usunięcie wybrakowanych tychże danych z dalszych rozważań. W zestawie danych pozostało 29777 obserwacji.

Jako zmienne objaśniające rozpatrywano te posiadające przedrostki 'app', 'act', 'agr' lub 'ags'. W ramach wstępnej analizy należało zapoznać się również z typem zmiennych, ponieważ analiza danych jakościowych znacząco różni się od podejścia w przypadku danych ilościowych. Dokonano analizy liczby tychże zmiennych, z podziałem ze względu na ich typ.

W podrozdziale 3.2, jako jeden z etapów budowy modelu, opisano realizację modelu wstępnego, celem wybrania potencjalnie najciekawszych zmiennych, na których można zaprezentować przykładową analizę eksploracyjną. Wyboru zmiennych dokonano na podstawie ich ważności w modelu, co zaprezentowano na rysunku 20. Sugerując się wynikami tego wskaźnika, wybrano trzy najistotniejsze zmienne we wstępnym modelu i przeprowadzono ich analizę.

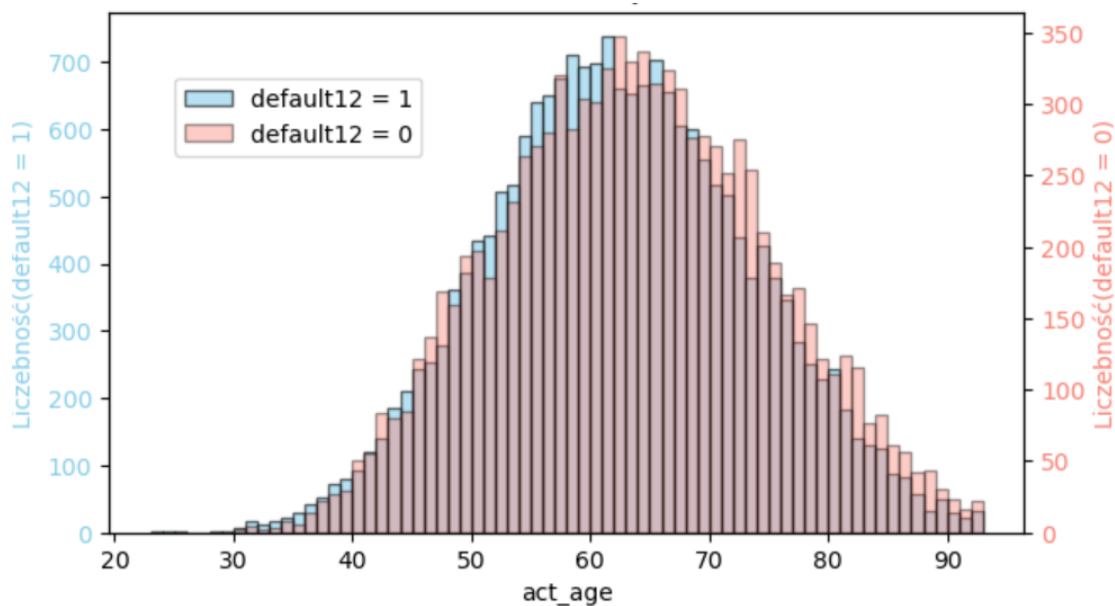
Analizę eksploracyjną rozpoczęto od podziału przetworzonego zbioru danych na dwa podzbiory - 'data0', zawierający obserwacje z zmienną celu równą '0' oraz 'data1', gdzie 'default12' jest równe '1', a następnie uzyskano podstawowe statystyki dla poszczególnych cech. Jako pierwszą przeanalizowano zmienną 'act_age', wskazującą wiek klienta w momencie składania wniosku, która została zakwalifikowana przez model wstępny jako najważniejsza. Wyniki analizy przedstawiono w tabeli 6, na podstawie której można zauważyć, że podzbiór klientów stwarzających wyższe ryzyko cechuje się niższym wiekiem w chwili aplikacji, co widać porównując wartości poszczególnych centyli oraz średnich.

Statystyka	default12 = 0	default12 = 1
Liczba obserwacji	9791	19986
Liczba braków danych	0	0
Średnia	63.028904	61.882468
Odchylenie standardowe	11.340882	11.004611
Wartość minimalna	30	23
Centyl 25%	55	54
Centyl 50%	63	62
Centyl 75%	71	69
Wartość maksymalna	93	93

Tabela 6: Podstawowe statystyki zmiennej 'act_age' (źródło opracowanie własne)

Celem wizualizacji rozkładu liczebności zmiennej 'act_age' w zależności od wartości zmiennej celu, wygenerowano histogram przedstawiony na rysunku 13. Mimo, iż model wstępny wskazał istotność wieku w predykcji zdarzenia wejścia klienta w opóźnienie w spłacie kredytu wynoszące w sumie co najmniej 90 dni w ciągu pierwszych dwunastu miesięcy od zaciągnięcia zobowiązania, wykres nie wskazuje wyraźnych różnic pomiędzy podgrupami. Przy dokładnej analizie można zauważyć delikatne przesunięcie się w lewo rozkładu dla 'default12' = 1, co znajduje swoje potwierdzenie w średniej z tabeli 6. W kodzie źródłowym można znaleźć również wizualizacje przy pomocy wykresów wiolinowych oraz pudełkowych, jednakże różnice są na nich równie niewielkie, stąd decyzja o nie umiesz-

czaniu ich w pracy.



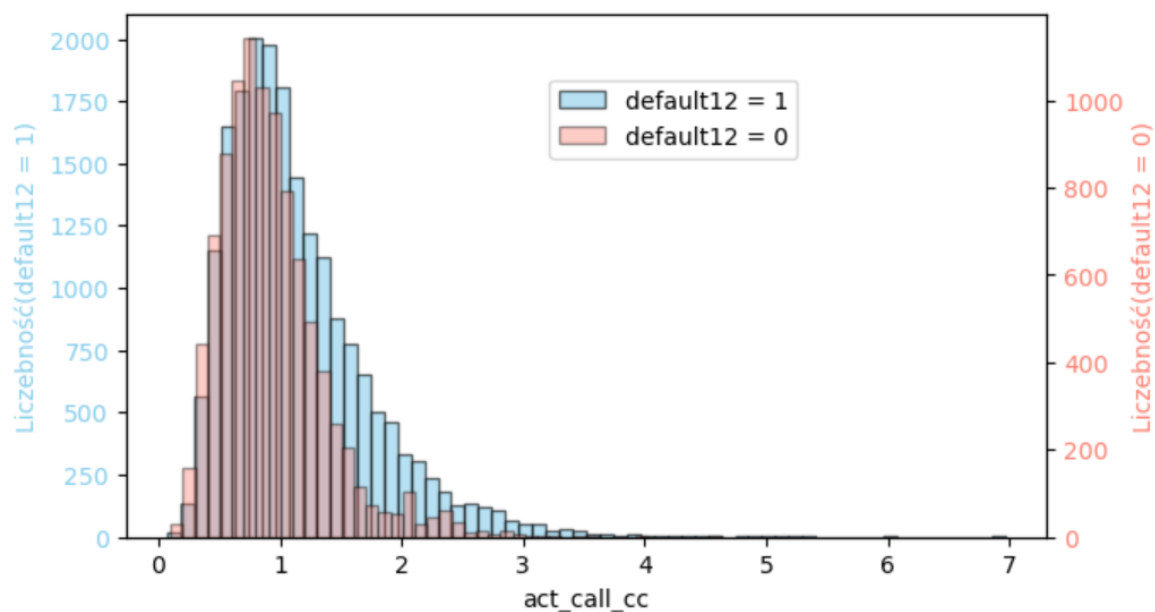
Rysunek 13: Rozkład zmiennej 'act_age' w zależności od wartości zmiennej 'default12' (źródło opracowanie własne)

Jako drugą przeanalizowano zmienną `act_call_cc`, oznaczającą sumę potencjalnych rat kredytu i wydatków w stosunku do dochodów klienta. Z tabeli 7 wynika, iż istnieje różnica pomiędzy dobrymi, a złymi klientami, możliwa do zauważenia na podstawie względnego obciążenia finansowego klientów wynikającego z otrzymania nowego produktu kredytowego. Prawie każda ze statystyk (oprócz wartości minimalnej) cechuje się wyższą wartością dla zmiennej 'default12' równej 1.

Statystyka	default12 = 0	default12 = 1
Liczba obserwacji	9791	19986
Liczba braków danych	0	0
Średnia	0.925345	1.141037
Odchylenie standardowe	0.436998	0.590710
Wartość minimalna	0.099325	0.075747
Centyl 25%	0.628913	0.721041
Centyl 50%	0.851278	1.010427
Centyl 75%	1.126885	1.423545
Wartość maksymalna	4.622150	6.964356

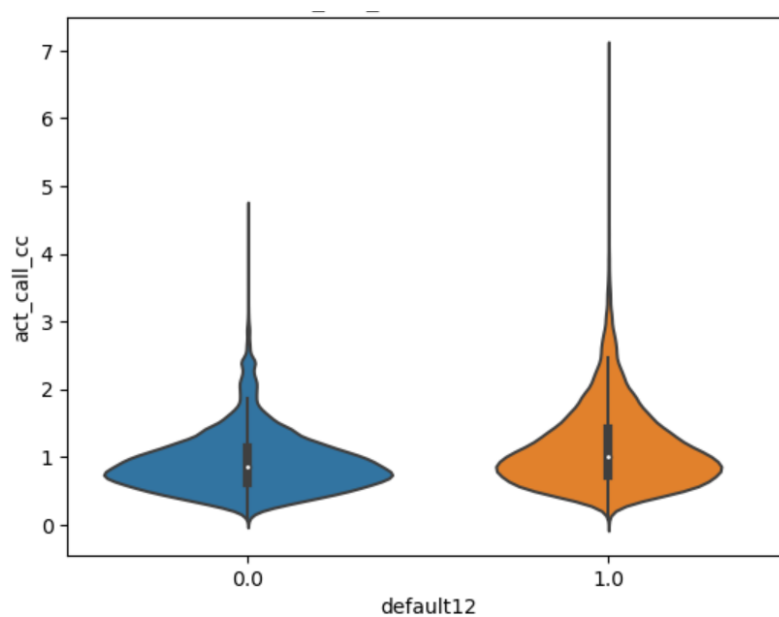
Tabela 7: Podstawowe statystyki zmiennej 'act_call_cc' (źródło opracowanie własne)

Histogram widoczny na rysunku 14 wykazuje liczniejsze wystąpienia klientów z wyższą wartością zmiennej 'act_call_cc' wnioskodawców z podgrupy 'default12' = 1.

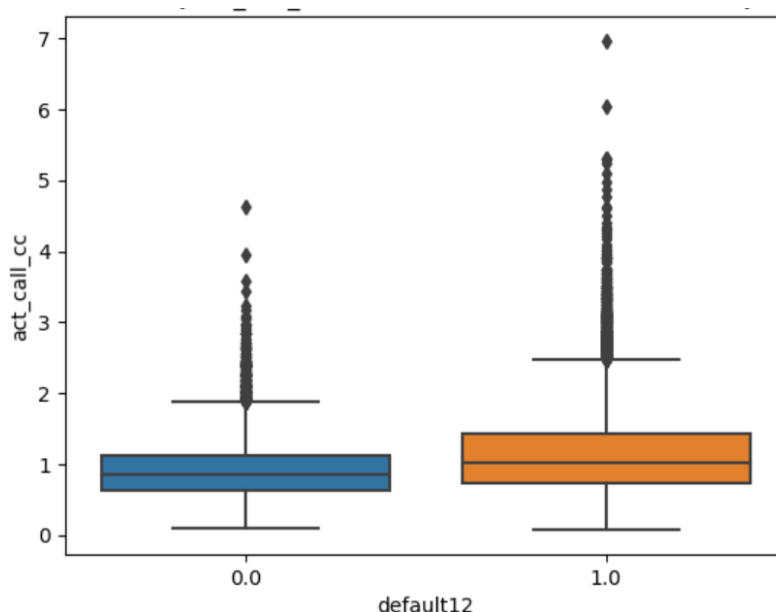


Rysunek 14: Rozkład zmiennej 'act_call_cc' w zależności od wartości zmiennej 'default12' (źródło opracowanie własne)

Ciekawe wizualizacje występującego zjawiska udało się osiągnąć na wykresach - wiolinowym, na rysunku 15 oraz pudełkowym, na rysunku 16.



Rysunek 15: Rozkład zmiennej 'act_call_cc' w zależności od wartości zmiennej 'default12' na wykresie wiolinowym (źródło opracowanie własne)



Rysunek 16: Rozkład zmiennej 'act_call_cc' w zależności od wartości zmiennej 'default12' na wykresie pudełkowym (źródło opracowanie własne)

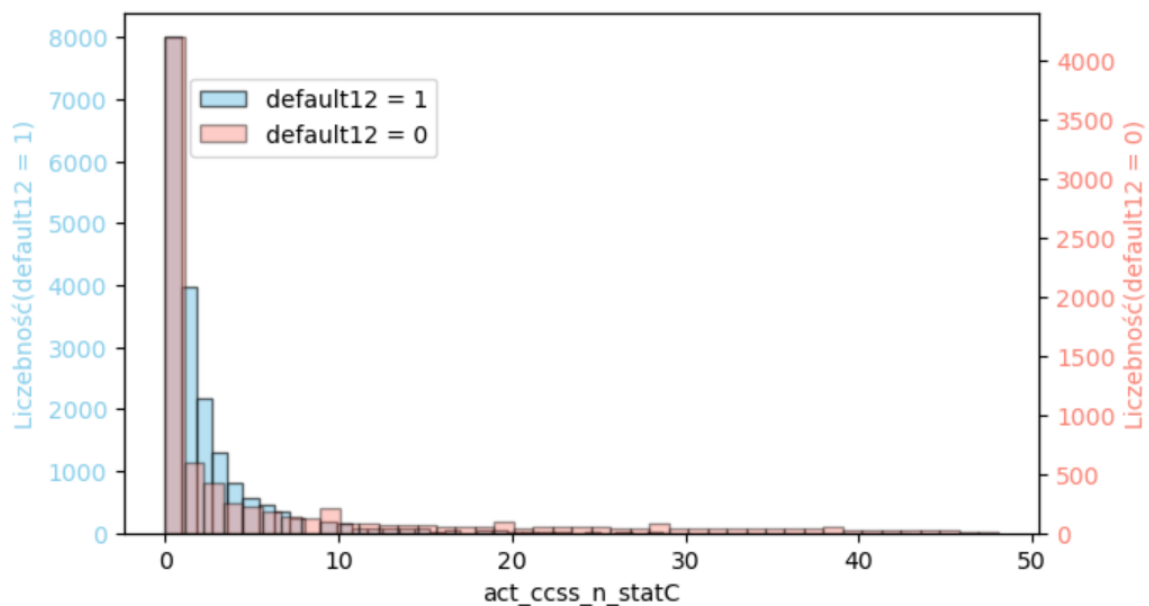
Analizę eksploracyjną zakończono na trzeciej najważniejszej zmiennej, według modelu wstępnego, którą była 'act_ccss_n_statC', oznaczająca liczbę poprawnie w pełni spłaconych zobowiązań klienta w momencie składania wniosku kredytowego. W tabeli 8 ponownie można zauważyć, iż istnieje różnica pomiędzy dobrymi, a złymi klientami, możliwa do wywnioskowania na podstawie pozytywnej przeszłości kredytowej klienta.

Statystyka	default12 = 0	default12 = 1
Liczba obserwacji	7889	18894
Liczba braków danych	1902	1092
Średnia	6.196096	2.110194
Odchylenie standardowe	10.407138	3.654164
Wartość minimalna	0	0
Centyl 25%	0	0
Centyl 50%	1	1
Centyl 75%	48	3
Wartość maksymalna	48	43

Tabela 8: Podstawowe statystyki zmiennej 'act_ccss_n_statC' (źródło opracowanie własne)

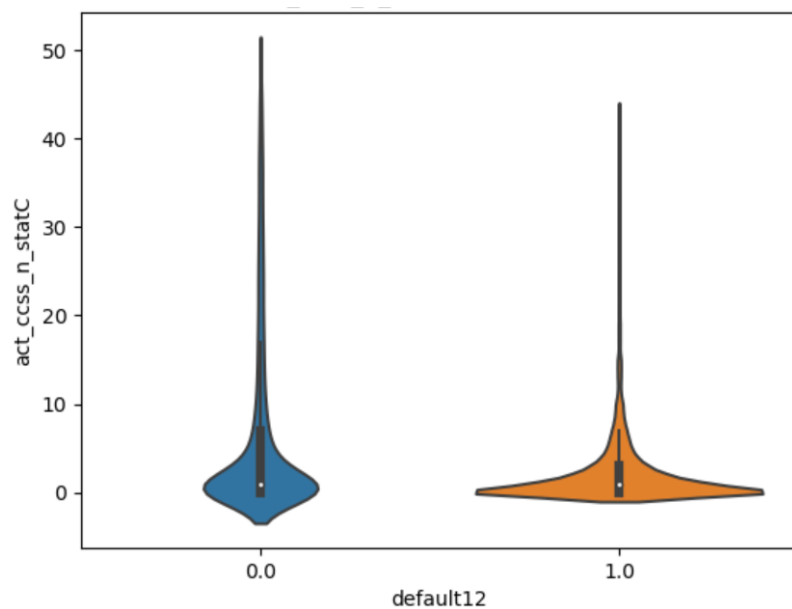
Histogram widoczny na rysunku 17 wykazuje koncentrację większości klientów blisko wartości 0 oraz 1. Warto zwrócić uwagę na sposób skalowania osi y, jako że dla obydwu podgrup zastosowane są różne skale, co może dawać złudzenie większego podobieństwa rozkładów niż ma to miejsce w rzeczywistości.

Warto zwrócić uwagę na wykres wiolinowy, na rysunku 18. Kształty dla obydwu pod-



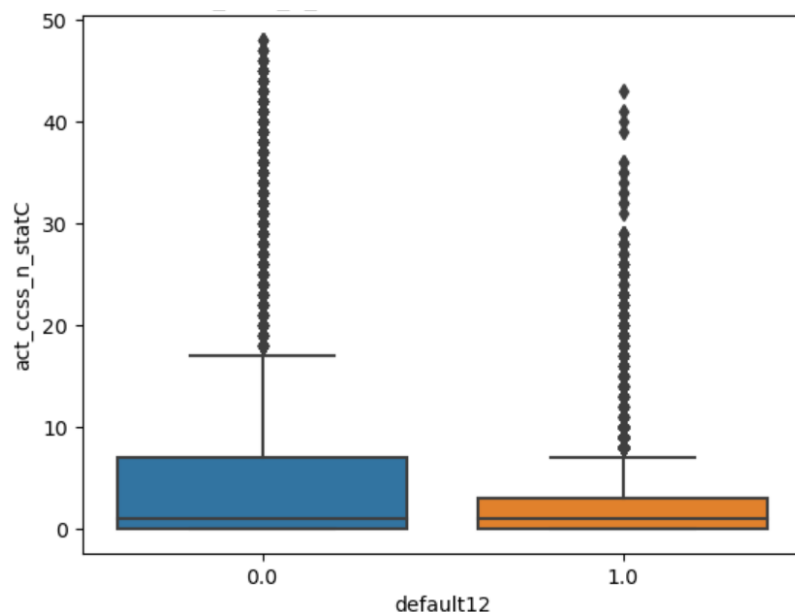
Rysunek 17: Rozkład zmiennej 'act_ccss_n_statC' w zależności od wartości zmiennej 'default12' (źródło opracowanie własne)

grup zupełnie się różnią. Wydłużenie w okolicach wartości 0 utwierdza w przekonaniu o korelacji małego pozytywnego doświadczenia kredytowego w przypadku złych klientów.



Rysunek 18: Rozkład zmiennej 'act_ccss_n_statC' w zależności od wartości zmiennej 'default12' na wykresie wiolinowym (źródło opracowanie własne)

Wizualizacja pudełkowa z rysunku 19, dodatkowo zwraca uwagę na szerokie spektrum wartości przyjmowanych przez zmienną 'act_ccss_n_statC'.



Rysunek 19: Rozkład zmiennej 'act_ccss_n_statC' w zależności od wartości zmiennej 'default12' na wykresie pudełkowym (źródło opracowanie własne)

Po zapoznaniu się z charakterystyką wybranego zbioru danych, przetworzeniu go do celów budowy modelu oraz analizie eksploracyjnej najważniejszych zmiennych, przystąpiono do budowy modelu uczenia maszynowego wykrywającego potencjalnie nierzetelnych klientów wnioskujących o gotówkowy produkt kredytowy.

3.2 Wybór narzędzi, budowa modelu i analiza wyników

Etap budowy modelu rozpoczęto od określenia odpowiednich prerekwizytów. W części programistycznej skorzystano z języka python w wersji 3.11.1. Rolę edytora kodu oraz silnika obliczeniowego pełniło popularne oprogramowanie Jupyter. Jako model predykcyjny wybrano drzewo decyzyjne, jednakże w odpowiedzi na rozwój rozwiązań drzewiastych, a w szczególności poprawę skuteczności decyzyjnej będącej skutem ich zastosowań, postanowiono skorzystać z modelu XBoost (ang. Extreme Gradient Boosting), nazywanego także wzmocnionym drzewem decyzyjnym.

XGBoost to algorytm uczenia maszynowego, który zyskał dużą popularność zarówno w środowisku akademickim, jak i przemysłowym, ze względu na swoją wysoką wydajność w różnych zadaniach związanych z analizą danych. Jest to metoda tzw. uczenia zespołowego, która łączy zalety wzmocniania gradientowego i technik regularyzacji, aby tworzyć bardzo dokładne i efektywne modele predykcyjne. Jest dobrze przystosowany do danych tablicowych i skutecznie radzi sobie z brakami danych. XGBoost wyróżnia się w zadaniach

takich jak klasyfikacja, czy regresja(datascience.eu, 2019).

Główną siłą tego algorytmu jest jego zdolność do obsługi złożonych interakcji między cechami. Wykorzystuje on wzmacnianie gradientowe, iteracyjnie poprawiając kolejne drzewa decyzyjne poprzez podniesienie ich skuteczności, jednocześnie minimalizując określoną funkcję straty. Aby dalej poprawić uogólnienie modelu i uniknąć przeuczenia, XGBoost wykorzystuje techniki regularyzacji, takie jak regularyzacja L1 i L2. Wysoka skalowalność i możliwość równoległego przetwarzania sprawiają, że XGBoost dobrze sprawdza się w pracy z dużymi zbiorami danych(datascience.eu, 2019).

Celem skorzystania z algorytmu XGBoost, należało na wstępie zainstalować w środowisku programistycznym pakiet o tej samej nazwie. Dodatkowo zainstalowano także pakiet 'category_encoders'. Niezbędne było również zaimportowanie kilku bibliotek, a pierwsza z nich to 'pandas' - otwarta biblioteka Pythona zaprojektowana do manipulacji i analizy danych. Wykorzystano również dwie funkcjonalności importowane z biblioteki 'sklearn'. Funkcja 'train_test_split' służy do podziału danych na zbiory treningowe i testowe w celu oceny modelu, natomiast 'roc_auc_score', przeznaczona jest do oceny wydajności modeli klasyfikacji binarnej. Inną niezbędną funkcją z biblioteki sklearn jest 'roc_curve', która służy do generowania krzywych ROC dla modeli klasyfikacji binarnej. Ostatnią funkcją z tejże biblioteki jest 'accuracy_score', która wykorzystywana jest do oceny dokładności modeli klasyfikacyjnych.

W celu wizualizacji wyników zaimportowano moduł 'pyplot' z biblioteki 'matplotlib', który daje możliwość tworzenia różnych typów wykresów, diagramów i wizualizacji. Wykorzystano również bibliotekę 'seaborn', która służy do tworzenia wizualizacji skomplikowanych zbiorów danych. Ostatnią zastosowaną biblioteką była 'math', udostępniająca funkcje matematyczne oraz stałe do różnych obliczeń, w tym operacji arytmetycznych, trygonometrii, czy logarytmów(python.org, 2023).

Mając dostęp do wyżej wymienionych narzędzi, rozpoczęto proces budowy modelu. Kod programistyczny wykorzystany w realizacji praktycznej części pracy został umieszczony na Githubie w postaci pliku Jupyter Notebook, który jest dostępny pod linkiem "https://github.com/MateuszKuchta88/Master_Diploma/blob/main/PracaTresc/XGBoostModel_MK116111_1.ipynb", a także w wersji pliku HTML, dostępnego w lokalizacji https://github.com/MateuszKuchta88/Master_Diploma/blob/main/PracaTresc/XGBoostModel_MK116111_1.html. Przy budowie modelu skorzystano ze wstępnie przetworzonego

zbioru 'data_css' oraz wcześniej przygotowanych list zmiennych jakościowych i ilościowych. Model XGBoost nie jest w stanie operować na zmiennych jakościowych, co wymusza konieczność zastosowania kategoryzacji tychże zmiennych. W tym celu skorzystano z funkcji 'BinaryEncoder' pakietu 'category_encoders' realizującej kodowanie binarne. 7 zmiennych jakościowych zostało poddane temu procesowi, na skutek czego powstało 16 nowych zmiennych binarnych. Wynikiem tej operacji był zbiór zbinowanych zmiennych 'data_encoded'.

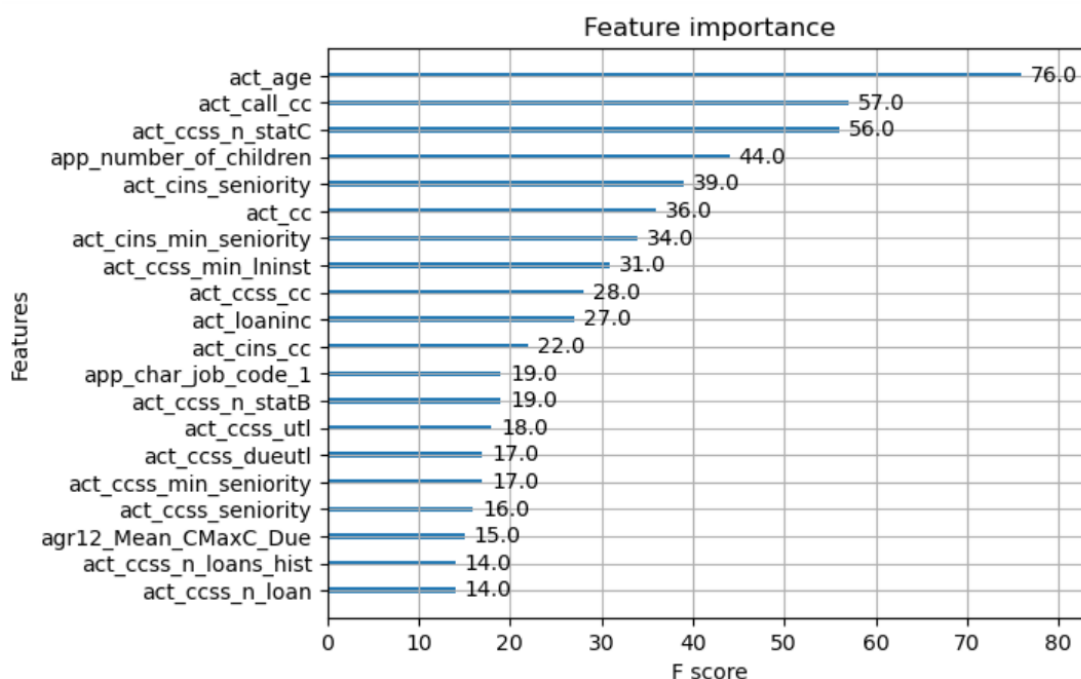
W kolejnym kroku dokonano konkatenaacji zbiorów 'data_css' oraz 'data_encoded' tworząc ramkę danych 'data_full'. Następnie zrealizowano podział finalnego zbioru na podzbiory treningowy i testowy przy pomocy funkcji 'train_test_split'. Wymiary podzielonych zbiorów to 20843 obserwacje dla zbioru treningowego oraz 8934 dla zbioru testowego. Ostatnią operacją niezbędną do rozpoczęcia tworzenia algorytmu XGBoost i testów jego wyników, było utworzenie ściśle określonych struktur danych, które otrzymano jako wynik funkcji 'DMatrix()' z biblioteki 'xgboost'.

Strategia modelowa obejmowała wytrenowanie modelu na zbiorze treningowym, a następnie jego walidację na zbiorze testowym. Przeprowadzono szereg treningów i testów celem znalezienia najlepszego modelu, w tym celu manipulując parametrami uczenia. W następnym etapie użyto nauczonego modelu do utworzenia nowych zmiennych, w których przechowywano klasyfikację wykonaną przez XGBoost. Finalnie podsumowano te kroki obliczeniem mocy predykcyjnej modelu dla zbioru testowego i treningowego wyliczając współczynnik GINI przy pomocy funkcji 'roc_auc_score' z biblioteki 'sklearn'.

Pierwszy zbudowany model, stanowił element poglądowy do dalszej analizy. Jego głównym zadaniem, było zbudowanie prostego modelu XGBoost na przygotowanych danych, celem zgrubnego oszacowania ważności poszczególnych zmiennych, a wyniki tego badania zwizualizowano na rysunku 20.

Iteracyjne testy jakości zróżnicowanych modeli XGBoost, celem wyboru najlepszego z nich, rozpoczęto od zdefiniowania benchmarków, które muszą być przestrzegane przez wybrany algorytm. Zdefiniowano je w dwóch parametrach:

- 'gini_benchmark' = 0.8 - wartość oznaczająca minimalną wartość współczynnika Gini, która musi zostać spełniona zarówno przez model opracowany na danych treningowych, jak i na danych testowych;
- 'gini_diff_benchmark' = 0.04 - ze względu na złożoność i specyfikę rynku kredytowego



Rysunek 20: Analiza ważności zmiennych dla modelu wstępnego (źródło opracowanie własne)

wego, modele predykcyjne powinny skupiać się nie tylko na odpowiednim rozróżnianiu dobrych i złych klientów, ale również muszą być stabilne w zależności od danych. Stąd potrzeba zdefiniowania odpowiednio niskiej wartości benchmarku, wychwytyującego jedynie modele o niewielkiej różnicy w jakości predykcji pomiędzy zbiorem treningowym, a testowym.

Do wytrenowania modelu zastosowano funkcję `'train()'` z biblioteki `'xgboost'`. Oferuje ona możliwość manipulowania wieloma wewnętrznymi ustawieniami modelu. W ramach budowy zdecydowano się na manipulację trzema z nich:

- `'max_depth'` - określa maksymalną głębokość pojedynczego drzewa w modelu, przez co kontroluje on złożoność modelu i potencjalne przetrenowanie. Wyższa wartość parametru pozwala drzewu na zastosowanie bardziej skomplikowanych reguł, co może zwiększyć dopasowanie do danych treningowych. Dla niższej wartości, model będzie bardziej regularny, co może pomóc w ogólnej generalizacji i stabilności względem nowych danych. Standardowa głębokość drzewa to 6, ale dozwolone są wartości z zakresu liczb nieujemnych;
- `'learning_rate'` - kontroluje krok, o jaki model się dostosowuje podczas każdej iteracji. Niska wartość zmniejsza wpływ pojedynczych drzew na model, zapobiegając

przetrenowaniu, natomiast wysoka wartość przyspiesza uczenie, lecz może prowadzić do przeuczenia. Optymalna wartość zależy od danych i zadania. Ważne jest strojenie tego parametru wraz z innymi, takimi jak liczba drzew i głębokość, by uzyskać najlepszą jakość predykcji. Z defaultu przyjmuje się 'learning_rate' równe 0.3, a zakres dozwolonych wartości to [0,1];

- 'min_split_loss' - wartość minimalnej utraty dzielenia węzła podczas budowy drzewa. Jeśli wzrost w funkcji zysku nie przekracza tej wartości, węzeł nie będzie dalej podzielony. Pomaga to kontrolować strukturę drzewa, zapobiegając nadmiernemu dopasowaniu. Wyższa wartość parametru prowadzi do bardziej konserwatywnego modelu, ograniczając ryzyko przeuczenia. Standardowo założone jest 'min_split_loss' równe 0, jednakże może przyjmować wartości z zakresu liczb nieujemnych.

Poza parametrami, które należało dostroić w ramach testów iteracyjnych, zdefiniowano również inne wymagane ustawienia funkcji 'train', które pozostawały niezmienione w całej fazie badań:

- 'objective' - określa rodzaj problemu, który model ma rozwiązywać, np. czy to jest problem klasyfikacji, regresji, czy inny. Dostępne opcje to m.in. 'reg:squarederror' dla regresji, 'binary:logistic' lub 'binary:logitraw' dla klasyfikacji binarnej, czy też 'multi:softmax' dla wieloklasowej klasyfikacji;
- 'eval_metric' - definiuje miarę, na podstawie której algorytm ocenia efektywność modelu na zbiorze walidacyjnym. Przykładowe metryki to 'rmse' (średni błąd kwadratowy), 'mae' (średni błąd bezwzględny), 'logloss' (logarytmiczna funkcja straty), czy 'auc' (obszar pod krzywą ROC);
- 'num_boost_round' - określa liczbę iteracji w procesie treningu, gdzie każda runda dodaje nowy drzewowy model do kompozycji, poprawiając predykcje. Ważne jest, aby wybrać optymalną wartość, unikając niedouczenia lub przeuczenia. Zbyt mała wartość może prowadzić do niedokładnych predykcji, podczas gdy zbyt duża może spowodować nadmierne dopasowanie.

Wybrane parametry lub testowane ich zakresy zostały przedstawione w tabeli 9.

Parametr	Testowane wartości
max_depth	3 - 5
seed	1998
objective	binary:logitraw
learning_rate	0.5 - 0.9
min_split_loss	3 - 7
eval_metric	auc
num_boost_round	1000

Tabela 9: Parametry funkcji 'train' z biblioteki 'xgboost' zastosowane do wyboru odpowiedniego modelu (źródło opracowanie własne)

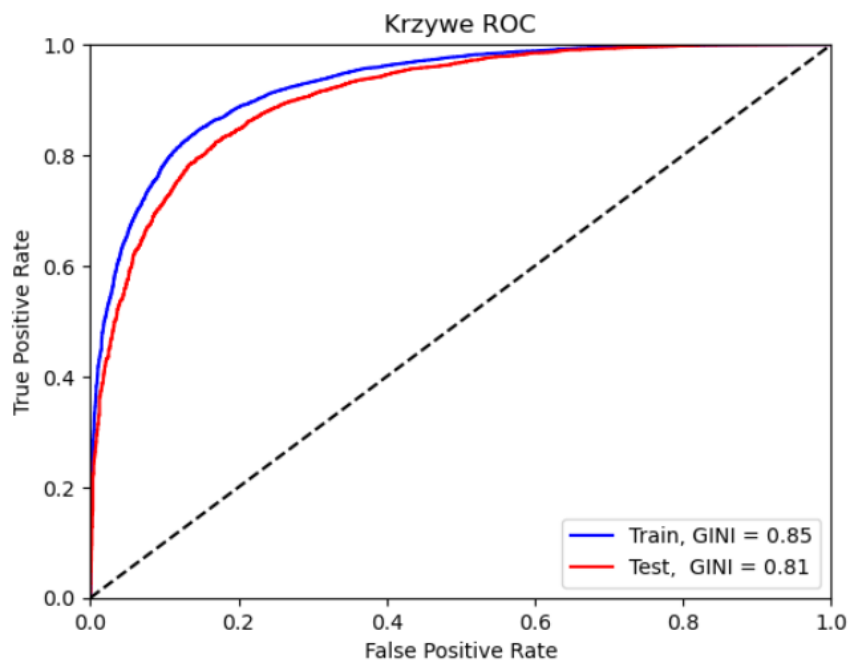
Na podstawie wybranych parametrów zbudowano kod, którego wynikiem była ramka danych opisująca statystyki współczynników Giniego. Wykorzystując wcześniej zdefiniowane benchmarki wyfiltrowano modele spełniające założone kryteria wysokiej skuteczności predykcji oraz stabilności. Następnie posortowano je rosnąco względem różnicy między współczynnikami Giniego na zbiorze treningowym, a zbiorze testowym. Najlepszy model zapisano jako 'M_chosen_model' celem późniejszego wykorzystania, a jego parametry przedstawiono w tabeli 10.

Parametr	Wartości
max_depth	3
learning_rate	0.5
min_split_loss	7
GINI_train	85.28%
GINI_test	81.29%
GINI_diff	4.00%

Tabela 10: Parametry finalnego modelu XGBoost (źródło opracowanie własne)

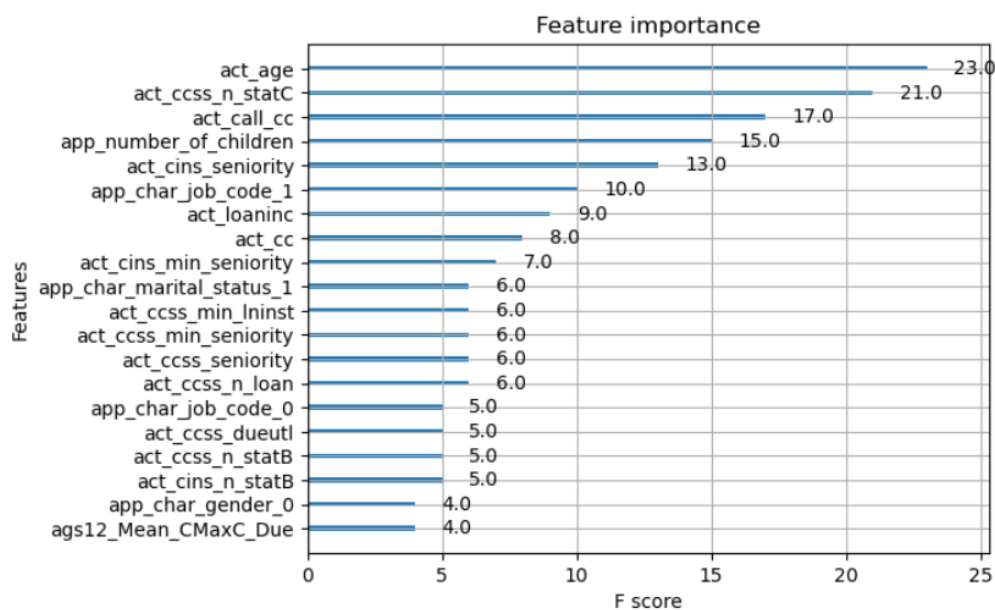
Podczas testów iteracyjnych zauważono, że model XGBoost jest w stanie generować bardzo wysoką skuteczność predykcji dla zbioru treningowego, osiągając wartości przekraczające 90%, jednakże wraz ze wzrostem współczynnika Giniego rosła także niestabilność modelu. Chcąc uzyskać znacznie bardziej satysfakcjonującą różnicę w jakości klasyfikacji pomiędzy zbiorami, następowały istotne spadki Giniego, co zabierało istotny argument za wyborem algorytmów uczenia maszynowego zamiast standardowych kart scoringowych. W związku z powyższym zaakceptowano 4% różnicy, zachowując wysoką skuteczność predykcji przekraczającą 80%. Zauważono również, że zwiększanie liczby iteracji treningowych modelu powyżej użytej wartości 1000 nie przynosi korzyści, jednakże odnotowano istotną różnicę jakości predykcji w zależności od parametru 'num_boost_round'. Jakość

klasyfikacji wybranego modelu zwizualizowano na rysunku 21 za pomocą krzywych ROC.



Rysunek 21: Wizualizacja wybranego modelu krzywymi ROC (źródło opracowanie własne)

Analizując wykresy ROC dla zbiorów treningowego oraz testowego odnotowano podobny, stabilny przebieg krzywych, co upewniło o poprawnie zrealizowanej budowie. Jako ostatni element tego etapu pracy, zbadano ważność zmiennych modelu finalnego, celem porównania jego wskazań z analizą wstępną, co zrealizowano na rysunku 22.



Rysunek 22: Analiza ważności zmiennych dla finalnego modelu (źródło opracowanie własne)

Pierwsze pięć najbardziej istotnych zmiennych, według modelu finalnego, nie różni się od wskazań wstępnych. Co więcej, z dwudziestu najważniejszych zmiennych, piętnaście powtarza się w obydwu analizach, choć w zmienionej kolejności. Zauważono, iż model finalny widzi większy potencjał niż model wstępny w zmiennych zbinowanych, wskazując cztery z nich wśród zwizualizowanych dwudziestu, zamiast wstępnej sugestii o jednej takiej zmiennej.

Opracowany model klasyfikacji binarnej XGBoost do celów predykcji wystąpienia zdarzenia wejścia w opóźnienie w spłacie w ciągu pierwszych 12 miesięcy od zaciągnięcia zobowiązania kredytowego przez klienta, zostanie przetestowany pod kątem jego odporności na ingerencje z dziedziny wrogiego uczenia maszynowego.

4 Atak na opracowany model

4.1 Postawienie pytań badawczych oraz przyjęcie strategii ataku

Analiza zrealizowana na łamach rozdziału drugiego umożliwiła zapoznanie się z aktualnym stanem wiedzy i badań dotyczących wrogiego uczenia maszynowego, uwzględniając szerszy opis typów ataków, jakie można stosować w próbach zaburzenia działania modeli. W rozdziale trzecim opisano proces budowy i walidacji algorytmu XGBoost, wykorzystanego w klasyfikacji wniosków kredytowych względem wystąpienia zdarzenia wejścia w opóźnienie w spłacie w ciągu pierwszych dwunastu miesięcy od zaciągnięcia zobowiązania. Natomiast rozdział czwarty stanowi opis weryfikacji pytań badawczych, które dotyczą zachowania modelu klasyfikacji binarnej pod wpływem zastosowania czynności z dziedziny wrogiego uczenia maszynowego.

W celu rzetelnego zbadania wrażliwości przygotowanego w rozdziale trzecim modelu XGBoost dla scoringu kredytowego, postawiono cztery pytania badawcze:

- Pytanie 1 - Czy odwrócenie wartości binarnej zmiennej celu w 1 % danych zbioru treningowym uniemożliwia dalsze wykorzystanie modelu XGBoost w bankowym środowisku produkcyjnym;
- Pytanie 2 - Czy zdublowanie z odwróconymi wartościami binarnej zmiennej celu 1 % danych w zbiorze treningowym uniemożliwia dalsze wykorzystanie modelu XGBoost w bankowym środowisku produkcyjnym;
- Pytanie 3 - Czy zmanipulowanie jednocześnie trzema z pięciu najważniejszych zmiennych modelowych w zakresie +/- 1-5 % wartości tych zmiennych pozwala na odwrócenie klasyfikacji binarnej otrzymanej jako odpowiedź z modelu XGBoost w ponad 10 % przypadków;
- Pytanie 4 - Czy zmanipulowanie jedną z pięciu najważniejszych zmiennych modelowych w zakresie +/- 1-10 % wartości tej zmiennej pozwala na odwrócenie klasyfikacji binarnej otrzymanej jako odpowiedź z modelu XGBoost w ponad 10 % przypadków;

Pytanie 1 zostało zweryfikowane przy pomocy zbioru danych zastosowanego do budowy modelu XGBoost w rozdziale trzecim. Podział na podzbiory treningowy i testowy zrealizowano również identycznie, aby mieć pewność, iż zarówno dane użyte do modelu

finalnego, jak również ataku białoskrzynkowego były ze sobą tożsame, dzięki czemu badanie było rzetelne. W kolejnym kroku dokonano szeregu testów polegających na losowym wyborze procentowo zdefiniowanej części zbioru testowego, odwróceniu wartości binarnej zmiennej celu i wytrenowaniu nowego modelu, zachowując parametry modelu finalnego. Wyniki przedstawiono za pomocą tabeli wartości opartych na współczynniku Giniego w zależności od procentowo zdefiniowanej wielkości atakowanej części zbioru treningowego. Dodatkowo rezultaty zwizualizowano za pomocą zbiorowego wykresu liniowego dla współczynnika Giniego dla zbioru testowego oraz treningowego w zależności od procenta atakowanych danych.

Pytanie 2 również zostało sprawdzone przy użyciu zbioru danych z rozdziału trzeciego. Ponownie podzielono podzbiory w ten sam sposób. Podział na podzbiory treningowy i testowy zrealizowano również identycznie, aby mieć pewność, iż zarówno dane użyte do modelu finalnego, jednakże etap testowy nieco się różni. Wylosowana próba została skopiowana na zewnątrz zbioru modelowego, a podmiana wartości binarnej zmiennej celu nastąpiła bez modyfikacji zbioru treningowego. Następnie spreparowany podzbiór skonkatenowano z treningowym zestawem modelowym. Kolejne kroki zrealizowano analogicznie jak dla pytania 1.

Do weryfikacji pytania 3 wciąż korzystano z tych samych podzbiorów jak przy badaniu innych pytań. W zakresie części praktycznej wybrano losowo pewną liczbę obserwacji oznaczonych jako wnioski odrzucone w danych wejściowych, jednocześnie będące zakwalifikowane jako potencjalni dłużnicy przez model finalny. Następnie dla każdej z tych obserwacji podjęto próbę wygenerowania wrogich próbek, poprzez odchylenia wartości trzech z pięciu najistotniejszych zmiennych na podstawie ich ważności w modelu XGBoost wytrenowanym w rozdziale drugim. Wartości wybranych zmiennych modyfikowano poprzez dodanie lub odjęcie pewnych procentowych wartości tych zmiennych, gdzie najmniejsza manipulacja wynosiła 1 % wartości, a największa 50 %. Tę część sfinalizowano poprzez zweryfikowanie odpowiedzi modelu na wprowadzone, zmanipulowane próbki oraz sprawdzenie skuteczności ataku porównując liczbę kwalifikacji zapytań do modelu jako wniosek zaakceptowano w stosunku do liczby wszystkich zapytań/próbek.

Badania nad pytaniem 4 przebiegały analogicznie jak dla pytania 3. Różnica polegała na generacji wrogich próbek, w których tym razem manipulacji poddano nie wszystkie

trzy zmienne jednocześnie, a każdą z osobna. Zakres odchyień wartości ponownie wyniósł od 1 % do 50 %, a porównanie odpowiedzi modelu finalnego zrealizowano dla poszczególnych zmiennych.

W części programistycznej weryfikacji postawionych pytań skorzystano z kilku narzędzi nie stosowanych w poprzednich etapach pracy. W ramach kodu kilkakrotnie wykorzystano funkcję `reset_index()`, której zadaniem było ponowne nadanie numeracji wierszy w zbiorze wylosowanych obserwacji, co ułatwiło iterowanie po zestawie danych oraz realizowanie zarówno modyfikacji, jak i kopiowanie. Skorzystano również z funkcji `query()`, która umożliwia prostą filtrację zbioru po konkretnych wartościach zmiennej. Funkcja `insert()`, dała możliwość wprowadzenia nowej zmiennej z wartościami na określoną przez użytkownika pozycję w docelowej ramce danych. Skorzystano także z funkcjonalności `drop()`, która usuwa zawartość zbioru, jednocześnie pozostawiając jego pustą strukturę. Z kolei stosując funkcję `accuracy_score` z biblioteki `'sklearn'`, oceniono skuteczność wrogich próbek, porównując klasyfikację nadaną im przez model z rzeczywistymi oznaczeniami ze zbioru treningowego. Używano także funkcji `concat()` z biblioteki `'pandas'`, która służyła do dopisywania kolejnych wierszy do ramki danych. Kluczowym aspektem w weryfikacji pytań badawczych było wykorzystanie funkcji `sample()`, pozwalającej na losowy wybór zdefiniowanej liczby wierszy ze wskazanego zbioru danych. Mając dostęp do tych narzędzi, przeprowadzono szereg badań, celem przetestowania rozpatrywanych scenariuszy ataku na model XGBoost, które opisano w kolejnych podrozdziałach.

4.2 Pytanie badawcze nr 1

Pytanie pierwsze poruszało kwestię, czy odwrócenie wartości binarnej zmiennej celu w 1 % danych zbioru treningowego uniemożliwia dalsze wykorzystanie modelu XGBoost w bankowym środowisku produkcyjnym. Oznacza to, że dostęp do niewielkiej części danych, na których następuje uczenie, jest wystarczający do skutecznej degradacji wiarygodności zawartych w nim informacji i zmusza analityków odpowiedzialnych za skuteczność modelu do przededefiniowania aktualnie stosowanych parametrów. Pytanie zakłada, iż atak na dane znacząco obniży współczynnik Giniego zarówno dla weryfikacji na próbce treningowej, jak i testowej, jednocześnie doprowadzając do nieakceptowalnego powiększenia się różnicy w Ginim pomiędzy podzbiorami. Założenie, że dostęp do zaledwie 1 % zbioru jest dostateczny zostało zasugerowane w cytowanych w rozdziale drugim badaniach nad

atakami na systemy filtracji antyspamowej w wiadomościach mailowych (Kuchipudi et al., 2020), co postanowiono sprawdzić w przypadku modelu XGBoost. Dla bardziej rzetelnej oceny, zrealizowano szereg testów, zakładając dostęp różnych zakresów danych, sięgając do 15 % zawartości.

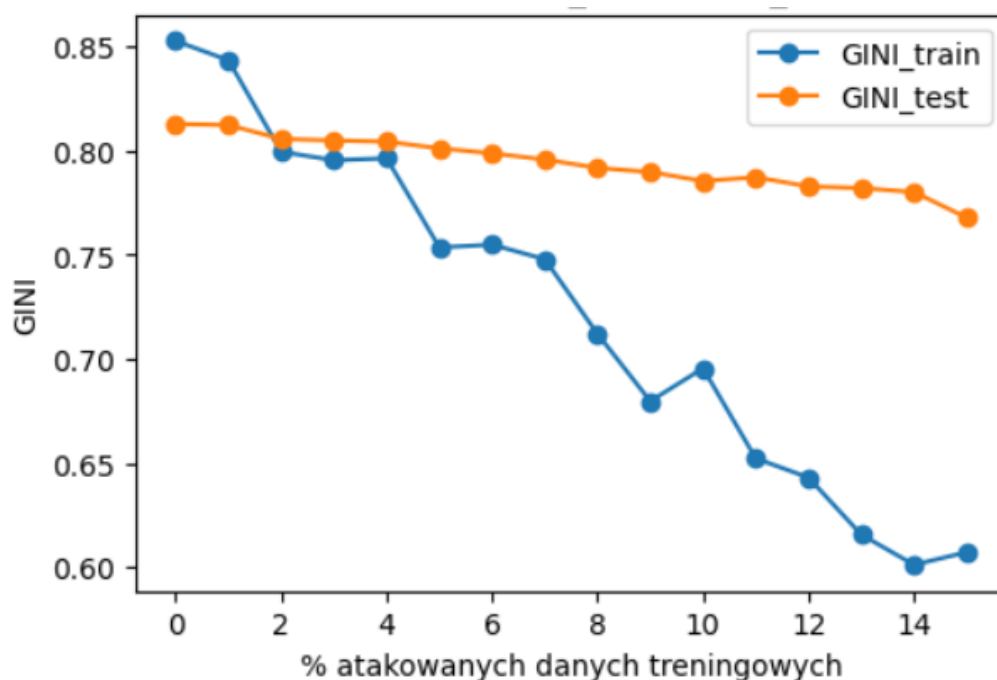
W ramach weryfikacji pytania 1 wybrano w zbiorze treningowym pewną część znajdujących się w nim obserwacji i zmieniono wartość zmiennej celu 'default12' na odwrotną. Rozpoczęto od utworzenia ramek danych właściwych dla tego badania. Następnie wybrano zakres wartości procentowej modyfikowanych danych jako parametr 'attacked_proc1', co dla "attacked_proc1" = 15" oznacza, że wykonano 15 testów, gdzie kolejne wartości procentowe zmienionych danych znajdowały się w zakresie liczb całkowitych od 1 do 15. W kolejnym kroku wykorzystano pętlę celem losowania liczby manipulowanych obserwacji, obliczając ją na podstawie zdefiniowanej wcześniej wartości 'attacked_proc1'. Dla wylosowanej próby dokonywano podmianę wartości zmiennej 'default12' bezpośrednio w zbiorze danych. W następnym etapie dokonano standardowych operacji przygotowania zmanipulowanego zbioru do wykonania na nim procesu przetrenowania modelu, stosując parametry identyczne jak w modelu finalnym.

Po zweryfikowaniu przetrenowanych modeli na zbiorze walidacyjnym, obliczono statystyki współczynnika Giniego i przedstawiono je w tabeli 11. Łatwo zauważyć, że manipulacja 1 % danych zaburza działanie modelu nie pozwala na odpowiedź twierdzącą przy pytaniu badawczym nr 1. XGBoost wciąż przestrzega założone benchmarki, co więcej, algorytm może wydawać się bardziej stabilny, co widać poprzez niższą wartość 'GINI_diff', przy praktycznie nie zmienionej wartości 'GINI_test'. Dużo większy wpływ widać przy zastosowaniu manipulacji 2 % danych treningowych, gdyż to zaburzenie wybija statystyki modelu poza założone benchmarki. Ogólna analiza wyników jednoznacznie wskazuje istotny wpływ ataku na zbiór uczący na wskaźniki 'GINI_train', przy względnej stabilności Giniego dla zbioru testowego. Współczynnik 'GINI_test', mimo manipulacji w 15 % danych obniżył się tylko o 4.5 %, jednakże należy pamiętać o nieakceptowalnie dużej wartości różnicy Giniego pomiędzy podzbiorami.

Analiza przebiegu wartości współczynnika Giniego dla obydwu podzbiorów w zależności od wielkości zmanipulowanej części podzbioru, zwizualizowana na rysunku 23, jednoznacznie upewnia o konieczności zanegowania pytania 1.

	GINI_train	GINI_test	GINI_diff	% atakowanych danych treningowych
0	85.28%	81.29%	4.00%	0%
1	84.35%	81.24%	3.11%	1%
2	79.95%	80.57%	-0.62%	2%
3	79.55%	80.50%	-0.96%	3%
4	79.64%	80.44%	-0.80%	4%
5	75.36%	80.12%	-4.76%	5%
6	75.49%	79.88%	-4.39%	6%
7	74.79%	79.58%	-4.80%	7%
8	71.20%	79.18%	-7.98%	8%
9	67.97%	78.98%	-11.00%	9%
10	69.55%	78.55%	-9.00%	10%
11	65.22%	78.73%	-13.50%	11%
12	64.30%	78.29%	-13.99%	12%
13	61.56%	78.21%	-16.65%	13%
14	60.11%	78.02%	-17.91%	14%
15	60.71%	76.79%	-16.08%	15%

Tabela 11: Wartość współczynnika Giniego w zależności od procenta atakowanych danych dla pytania 1 (źródło opracowanie własne)



Rysunek 23: Wykres wartości współczynnika Giniego dla zbioru treningowego i testowego dla pytania 1 (źródło opracowanie własne)

4.3 Pytanie badawcze nr 2

Pytanie drugie poruszało kwestię czy skopiowanie 1 % danych zbioru treningowego i odwrócenie wartości binarnej zmiennej celu w kopii, a następnie dołączenie tego zmo-

dyfikowanej części informacji do oryginalnych danych uniemożliwia dalsze wykorzystanie modelu XGBoost w bankowym środowisku produkcyjnym. Ponownie zbadanie tego pytania pozwoliło na weryfikację osądu, iż dostęp do niewielkiej części danych uczących, jest wystarczający do zdeprecjonowania przydatności modelu predykcyjnego XGBoost. Pytanie zakłada, iż atak istotnie wpłynie na wartość współczynnika Giniego tak dla weryfikacji na podzbiorze treningowym, jak i testowym, przy jednoczesnym zwiększeniu się różnicy w Ginim. Badania urzeczono poprzez realizację, szeregu prób testowych, zakładając dostęp do różnych wielkości zestawu danych, maksymalnie osiągając 15 % zbioru uczącego.

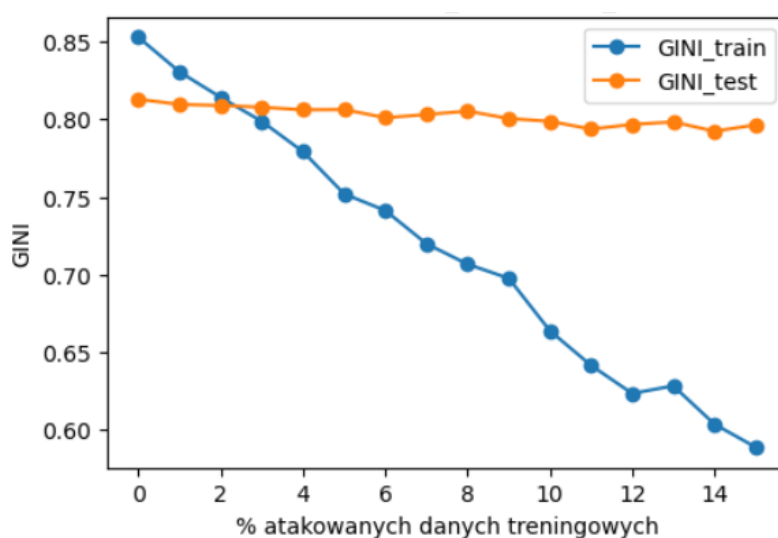
Rozpoczęto od ponownego podzielenia zbioru wejściowego na części identyczne jak w przypadku weryfikacji pierwszego z pytań. Następnie wybrano zakres w jakim manipulowano wartością procentową atakowanych danych jako parametr 'attacked_proc2', co dla "attacked_proc2" = 15" oznacza, że wykonano 15 testów, gdzie kolejne wartości procentowe zmienionych danych znajdowały się w zakresie liczb całkowitych od 1 do 15. W kolejnym kroku zapętłono losowanie liczby manipulowanych obserwacji, wylosowany podzbiór kopiowano, podmieniano wartości binarnej zmiennej celu 'default12' i finalizowano dokonaniem spreparowanej próby do trenowanego zbioru. Ostatnim etapem było wykonanie czynności przygotowujących utworzony zestaw testowy do zrealizowania na nim procesu treningu modelu, stosując parametry tożsame z algorytmem wybranym w rozdziale 3.

Po zbadaniu właściwości modeli na zbiorze walidacyjnym, wyliczono rozkład wartości współczynnika Giniego, a wyniki przedstawiono w tabeli 12. Ponownie manipulacja jedynie 1 % danych treningowych pogarsza wyniki modelu w zbyt małym stopniu, nawiązując do postawionych w rozdziale trzecim benchmarków. W przypadku testu dla pytania 2, zauważono obniżenie współczynnika Giniego o wartość nieco większą dla obydwu podzbiorów niż dla weryfikacji pytania 1. Ponadto, można wywnioskować, iż atak zakładający dołożenie dodatkowych obserwacji do zbioru treningowego, jest w stanie zmniejszyć wartość 'GINI_train' do poziomu nawet do dwóch procent niższego niż dla podmiany binarnej wartości zmiennej celu bezpośrednio w zbiorze treningowym. Warto również zauważyć, że rozpatrywany rodzaj ataku jest znacznie mniej inwazyjny dla zbioru testowego, ponieważ jesteśmy w stanie obniżyć skuteczność klasyfikacji poniżej 80 % dopiero przy manipulacji dziesięcioma procentami oryginalnych danych, gdzie dla ingerencji bez kopiowania obserwacji na zewnątrz osiągnięto to zjawisko przy dostępie do 5 % podzbioru, a dla 15 % stopnia manipulacji 'GINI_test' jest niższy jedynie o 1.65 %.

	GINI_train	GINI_test	GINI_diff	% atakowanych danych treningowych
0	85.28%	81.29%	4.00%	0%
1	83.05%	80.96%	2.09%	1%
2	81.41%	80.89%	0.52%	2%
3	79.86%	80.78%	-0.92%	3%
4	77.91%	80.61%	-2.70%	4%
5	75.19%	80.63%	-5.44%	5%
6	74.12%	80.10%	-5.98%	6%
7	71.98%	80.31%	-8.32%	7%
8	70.66%	80.53%	-9.87%	8%
9	69.75%	80.05%	-10.30%	9%
10	66.37%	79.87%	-13.50%	10%
11	64.14%	79.37%	-15.23%	11%
12	62.33%	79.67%	-17.34%	12%
13	62.82%	79.83%	-17.01%	13%
14	60.35%	79.23%	-18.88%	14%
15	58.87%	79.64%	-20.77%	15%

Tabela 12: Wartość współczynnika Giniego w zależności od procenta atakowanych danych dla pytania 2 (źródło opracowanie własne)

Analiza przebiegu wartości współczynnika Giniego dla obydwu podzbiorów w zależności od wielkości zmanipulowanej części podzbioru, zwizualizowana na rysunku 24, jednoznacznie upewnia o konieczności odpowiedzenia negatywnie na pytanie badawcze 2. Co więcej, zauważalna jest różnica w przebiegu 'GINI_train', gdzie obniżanie się tej wartości ma charakter niemal liniowy, w odróżnieniu od sytuacji dla pytania 1, gdzie obserwowano trudne do przewidzenia skoki.



Rysunek 24: Wykres wartości współczynnika Giniego dla zbioru treningowego i testowego dla pytania 2 (źródło opracowanie własne)

4.4 Pytanie badawcze nr 3

Trzecie pytanie badawcze dotyczyła manipulacji wartościami zmiennych objaśniających celem oszukania XGBoosta i zmuszenia go do zmiany klasyfikacji. W tym przypadku mowa o odchyleniu jednocześnie trzech z pięciu najważniejszych zmiennych modelowych w zakresie $\pm 1-5\%$ ich wartości w ponad 10% przypadków. Tym razem zbadanie pytania badawczego zakładało atak z typu czarnoskrzynkowych, gdzie nie adversarz nie ma dostępu do danych modelowych i nie może ich w żaden sposób modyfikować. Poprzez wielokrotne odpytywanie modelu danymi jedynie minimalnie różniącymi się od siebie można ocenić, czy oszukiwanie na polu którejkolwiek z nich ma sens, a jeśli tak, to w którym momencie następuje zmiana klasyfikacji.

Rozpoczęto od wyboru atakowanych zmiennych. Na podstawie ważności poszczególnych cech, wyliczanych dla modelu finalnego i zwizualizowanych na rysunku 22 w rozdziale trzecim, zdefiniowano pięć najistotniejszych zmiennych jako:

- 'act_age';
- 'act_ccss_n_statC';
- 'act_call_cc';
- 'app_number_of_children';
- 'act_cins_seniority'.

Spośród wyżej wymienionych zmiennych, wybrano trzy, a dokonano tego kierując się nie tylko ich ważnością, lecz także typem i rozkładem, co miało znaczenie dla kolejnych kroków. W kolejnym etapie, założono wybranie losowych obserwacji z atakowanego zbioru, co ułatwiło wybór wartości wszystkich opisujących go zmiennych i zapobiegło konieczności tworzenia sztucznych wniosków przez adversarza nie znającego zakresu możliwych danych. Spowodowało to, iż konieczne było odrzucenie zmiennej 'act_ccss_n_statC', która mimo iż jest ważna, to posiada znaczne braki danych, co skomplikowałoby proces generowania wrogich próbek. Jako drugą odrzucono zmienną 'app_number_of_children', ze względu na fakt, iż może ona przyjmować jedynie liczby całkowite oraz uniemożliwione jest wprowadzenie szerokiego zakresu wartości. Do dalszych badań wybrano zatem zmienne:

- 'act_age';

- 'act_call_cc';
- 'act_cins_seniority'.

Następny krok procesu, wyglądał analogicznie jak dla poprzednich pytań badawczych. Rozpoczęto od utworzenia ramek danych właściwych dla danego testu, a następnie wylosowano ze zbioru treningowego obserwacje, które zostały poddane manipulacji celem zmiany ich klasyfikacji. Rozpatrywano jedynie wnioski odrzucone ("default12 == 1"), ponieważ celem ataku było zakwalifikowanie klienta potencjalnie niewiarygodnego, jako wiarygodnego. Liczbę losowanych próbek dobrano w sposób taki, aby po odrzuceniu wylosowanych obserwacji, dla których wartość rzeczywista zmiennej 'default12' jest różna od klasyfikacji wynikającej z modelu finalnego, liczba modyfikowanych dalej obserwacji była równa 100. Na końcu zresetowano numery wierszy w ramce danych, aby ułatwić iterowanie po niej.

W ramach badania niezbędne było utworzenie dużego zbioru wrogich próbek, korzystając z wylosowanych wniosków odrzuconych. Przeprowadzono to poprzez dodanie lub odjęcie od wartości każdej ze zmiennych pewnej niewielkiej części jego wartości. W przypadku zmiennych ciągłych, do których należą 'act_call_cc' oraz 'act_cins_seniority', obliczano 1 % wartości dla każdej obserwacji, a następnie poprzez dodanie bądź odjęcie tej wartości, generowano nowy wiersz danych. Dla każdej ze zmiennych definiowano 'count', którego wartość oznaczała wartość graniczną dla manipulacji, tak że np. dla "count = 10" generowało się 20 nowych obserwacji, dla każdego ze 100 wylosowanych wniosków, co dawało 2000 wierszy ramki danych, które były zbiorem wejściowym dla wykonania tej samej operacji dla kolejnej zmiennej. W przypadku zdefiniowania wartości "count = 5", dla każdej ze zmiennych, finalnie otrzymano zbiór zawierający aż 100 tysięcy obserwacji. W związku z istotnym wzrostem rozmiarów ramki danych zawierających wrogie próbki wraz ze inkrementacją wartości 'count', a co za tym idzie wydłużeniem się czasu obliczeń, wybrano 5 jako wielkość graniczną i analizowano jedynie pięć testów. Warto również zauważyć, że dla zmiennej 'act_age', nie było możliwe modyfikowanie jest wielkości o procent jej wartości, ponieważ wiek przez nią opisywany może przyjmować jedynie liczby całkowite.

Finalnym krokiem było przebadanie odpowiedzi modelu z rozdziału trzeciego na wprowadzone, zaburzone dane. Wykonano 5 testów, gdzie w ramach pierwszego z nich wychy-

lano wielkości zmiennych jedynie o 1 procent lub wartość 1, a dla testu piątego sprawdzano kombinacje od 1 do 5 procent, jak również od 1 do 5. Wyniki takie jak liczba obserwacji, dla których model zmienił klasyfikację, jak również względna skuteczność XGBoosta zamieszczono w tabeli 13.

Procent/Wartość odchylenia	Względna skuteczność modelu	Liczba ataków	Liczba zmienionych klasyfikacji
1	100 %	800	0
2	99.75 %	6400	16
3	99.50 %	21600	108
4	99.38 %	51200	320
5	99.20 %	100000	800

Tabela 13: Wyniki testów weryfikacyjnych pytania 3 (źródło opracowanie własne)

Na podstawie analizy wyników wywnioskowano, iż model XGBoost do celu scoringu kredytowego wykazuje się wysoką odpornością na względnie nieduże manipulacje wartości kilku kluczowych zmiennych objaśniających i wymusza odpowiedź przeczącą dla trzeciego pytania badawczego. Jednakże warto zauważyć, że zwiększanie stopnia zmiany wartości podnosi liczbę udanych ataków. Przy większych manipulacjach należy wziąć pod uwagę rzetelność zmiennych, gdyż niewielkie oszustwo w trudniej weryfikowanych cechach klienta takich jak zarobki czy wydatki może zostać niezauważone przez oddział analityczny, co jest niełatwe do osiągnięcia np. w kwestii wieku wnioskodawcy.

4.5 Pytanie badawcze nr 4

Weryfikacja odpowiedzi na czwarte pytanie badawcze, podobnie jak w przypadku pytania trzeciego, polegało na manipulacji wartościami zmiennych objaśniających celem oszukania modelu XGBoost i zmuszenia go do zmiany klasyfikacji, jednakże tym razem badanie przeprowadzono dla każdej zmiennej z osobna i w znacznie większym zakresie. Wielkości każdej z cech zmieniono o +/- 1-50 %, co wykraczało poza obszar rozważań w ramach pytania badawczego 4, w której założono modyfikację do 10 %, co miało wygenerować odwrócenie się binarnej etykiety na zmiennej celu w co najmniej jednym na dziesięć ataków. Ponownie założono brak możliwości wprowadzania zaburzeń bezpośrednio w danych uczących. wrogie próbki generowano w oparciu o zmienne wybrane dla weryfikacji trzeciego pytania badawczego:

- 'act_age';

- 'act_call_cc';
- 'act_cins_seniority'.

Budowę zbioru danych atakujących ponownie uzyskano w ten sam sposób jak dla poprzednich pytań badawczych. W drugim kroku wylosowano ze zbioru treningowego 100 obserwacji opisujących dłużników, przypisanych przez model finalny do "default12' == 1", jednakże w odróżnieniu do weryfikacji pytania 3, gdzie dokonano jednego losowania, w tym przypadku zdecydowano się przeprowadzić je niezależnie dla każdej ze zmiennych.

W celu zbadania odpowiedzi na czwarte pytanie badawcze, należało utworzyć zbiór wrogich próbek dla każdej z analizowanych cech, opracowując go na podstawie wylosowanych wniosków odrzuconych. Przeprowadzono to analogicznie do procedury zastosowanej w przypadku trzeciego z pytań, jak również ponownie dla każdego z testów zdefiniowano zmienną 'count', jednakże w przypadku manipulacji tylko jedną cechą, wielkość zbioru atakującego nie była problemem. W przypadku zdefiniowania wartości "count = 10", dla każdej ze zmiennych otrzymano zbiory o zawartości dwóch tysięcy obserwacji.

W ostatnim etapie zbadano odpowiedzi modelu finalnego na zbiór atakujący. Wykonano po 6 testów dla każdej z wybranych zmiennych, gdzie w ramach pierwszego z nich edytowano wielkości cech o 5 procent lub wartość 5, a dla testu szóstego założono 50-stopniową manipulację. Rezultaty badań, reprezentowane przez wskaźniki takie jak liczba obserwacji, dla których XGBoost został oszukany, czy też względna skuteczność modelu, umieszczono w tabeli 14.

Analizując skuteczność ataków dla trzech zmiennych z osobna, łatwo zauważyć, iż najwięcej błędnych klasyfikacji wygenerował model atakowany różnymi wartościami wieku wnioskodawcy. Ingerencja w wartości zmiennych ciągłych nie dała satysfakcjonujących rezultatów nawet przy możliwości odchylenia o 50 %. Model finalny wydaje się być najodporniejszy na manipulację cechą 'act_call_cc', gdzie udało się wygenerować zaledwie 14 obserwacji z odwróconą klasyfikację. W kontekście pytania 4, musi zostać ona odrzucona, co wskazuje na wyższy stopień skomplikowania algorytmu XGBoost dla scoringu kredytowego, niż dla systemów antyspamowych. Co więcej, opierając się na wynikach tego testu można przyjąć, iż model jest bardzo stabilny i niewrażliwy na odchylenia jednej ze zmiennych objaśniających. Mimo osiągnięcia kilkuprocentowej skuteczności ataku dla odchylenia wieku wnioskodawcy o 50, jednakże w praktyce jest to cecha, która może być

Zmienna	Procent/ wartość odchylenia	Względna skuteczność modelu	Liczba ataków	Liczba zmienionych klasyfikacji
act_age	5	100 %	1000	0
	10	99.4 %	2000	12
	20	97.8 %	4000	89
	30	96.7 %	6000	200
	40	95.9 %	8000	331
	50	95.3 %	10000	471
act_call_cc	5	100 %	1000	0
	10	100 %	2000	0
	20	100 %	4000	0
	30	100 %	6000	0
	40	100 %	8000	0
	50	99.9 %	10000	14
act_cins_seniority	5	100 %	1000	0
	10	99.8 %	2000	4
	20	99.7 %	4000	14
	30	99.5 %	6000	30
	40	99.3 %	8000	58
	50	99.1 %	10000	91

Tabela 14: Wyniki testów weryfikacyjnych pytania 4 (źródło opracowanie własne)

nierzeczywista dla tak dużych manipulacji.

4.6 Podsumowanie badań nad wrogim uczeniem maszynowym

W ramach badań nad wrogim uczeniem maszynowym zrealizowano zestaw kompleksowych testów, mających na celu negatywne wpłynięcie na osiągi modelu zbudowanego w rozdziale trzecim, weryfikując prawdziwość czterech postawionych na wstępie pytań badawczych. Realizacja manipulacji modelem XGBoost przeznaczonym do oceny wiarygodności kredytowej została przeprowadzona z zaawansowanym wykorzystaniem procesów przetwarzania danych zaimplementowanych w języku python. Każde pytanie badawcze poruszało tematykę podatności modelu na ataki adversarzy, mając na celu wykrycie potencjalnych punktów wrażliwych i ocenę niezawodności algorytmu.

Pytanie badawcze nr 1 weryfikowało wpływ odwrócenia wartości binarnej zmiennej 'default12' w niewielkiej części danych treningowych. Celem takiego działania było ustalenie, czy tego typu manipulacja może znacząco wpłynąć na osiągi modelu i spowodować istotny dla przyszłego wykorzystania spadek współczynnika Giniego. Przeprowadzenie licznych

testów ujawniło, że przy manipulacji tylko 1% danych treningowych, skuteczność modelu pozostała stosunkowo stabilna, zgodnie z ustalonymi benchmarkami. Analiza współczynników Giniego wskazała, że odporność XGBoost została utrzymana, a odpowiedź na postawione pytanie badawcze była negatywna. Jednakże rewidując osiągi modelu na zbiorze walidacyjnym zauważono, iż realna jest możliwość wywarcia wpływu na mechanizm predykcyjny, choć wymaga ona dużej ingerencji w zestaw danych uczących, bądź bezpośrednio w informacje testowe, zatem prawdziwe jest stwierdzenie, że algorytm XGBoost jest odporny na ataki wewnątrz kilkuprocentowej próbki danych treningowych.

Pytanie badawcze nr 2 poruszało temat duplikowania i odwracania zmiennej celu w skopiowanym podzbiorze danych treningowych. Celem była ocena, czy ta strategia ataku może prowadzić do obniżenia dokładności modelu. Stwierdzono, iż ten typ ataku jest mniej skuteczny w kontekście wpływu na osiągi predykcji na zbiorze testowym. Znaczące osłabienie się działania modelu na zbiorze treningowym, osiągnięte dla wyższych wartości objętości zainfekowanych danych, ma marginalny wpływ dla skuteczności w danych testowych. Drugie pytanie badawcze zostało definitywnie odrzucone, a dodatkowym wnioskiem jest przewaga ataku bez kopiowania danych, który silniej pogarsza wyniki walidacyjne, a dodatkowo generuje bardzo niestabilny przebieg wartości współczynnika Giniego w zależności od procenta atakowanych danych dla etapu uczenia.

Pytanie badawcze nr 3 wymagało zagłębienia się w manipulowanie wartościami kluczowych zmiennych objaśniających w celu oszukania klasyfikacji otrzymanej z modelu. Weryfikacja polegała na wielokrotnym odpytywaniu o wynik z nieznacznie zmienionymi danymi. Analiza wykazała, że model cechuje się wysokim poziomem odporności na tego rodzaju ataki przy modyfikacji kluczowych cech o maksymalnie 5%. Wyniki sugerują, że drobne manipulacje nie mają znaczącego wpływu na wydajność modelu, ale większe zmiany mogą wpłynąć na jego przewidywania. Jednakże w trzecim pytaniu badawczym zauważono największy potencjał, jednakże wymagane są tutaj szersze badania, w ramach których zawierałoby się zaatakowanie większej liczby zmiennych jednocześnie, na wyższych poziomach manipulacji np. 10 %. Przy dokładniejszej analizie wychwycono możliwą tendencję do coraz szybszego wzrostu liczby skutecznie spreparowanych danych, jednakże w tym celu należałoby bliżej przyjrzeć się wydajności kodu generującego wrogie próbki, który jest zbyt wolny by tworzyć większe atakujące zbiory.

Weryfikacja odpowiedzi na pytanie badawcze nr 4 zakładała większe manipulacje na pojedynczych zmiennych. Analiza wykazała zdolność modelu do radzenia sobie ze zmianami od -50% do +50% wartości każdej zmiennej. Celem było ustalenie, czy duże zmiany w pojedynczych zmiennych mogą prowadzić do znacznych wypaczeń w przewidywaniach modelu. Wyniki ukazały, że model pozostał stosunkowo stabilny nawet wtedy, gdy zmienne były manipulowane w szerokim zakresie. Nie odnotowano jakichkolwiek sygnałów, aby ta metoda mogła być przydatna w realnych atakach, jako że XGBoost dobrze radzi sobie z tego typu pojedynczymi zaburzeniami, nawet gdy osiągają duże wartości, jednakże by móc to jednoznacznie stwierdzić zaleca się przetestowanie pod tym kątem wszystkich zmiennych modelowych.

Podsumowując, analiza prezentuje dogłębne badanie odporności modelu XGBoost na różne ataki w kontekście oceny wiarygodności kredytowej. Wyniki ukazują, że model wykazuje wysoką odporność na większość scenariuszy ataków. Choć niektóre manipulacje prowadziły do niewielkich zmian współczynników Giniego lub sugerowanego prawdopodobieństwa zjawiska wejścia w opóźnienie w spłacie w ciągu pierwszych dwunastu miesięcy od zaciągnięcia zobowiązania kredytowego, ogólna wydajność pozostała w akceptowalnych granicach. Niemniej jednak badanie sugeruje, że w niektórych przypadkach większe manipulacje mogą prowadzić do zauważalnych zmian. W związku z tym analiza podkreśla znaczenie ciągłego monitorowania i stałej walidacji skuteczności predykcji, aby zapewnić, że algorytmy oceny kredytowej pozostaną skuteczne i niezawodne w rzeczywistym środowisku bankowym.

Podsumowanie

Praca została rozpoczęta od zrozumienia sposobu, w jaki ocenia się zdolność kredytową. Wyjaśnienie i przedstawienie poszczególnych etapów tej procedury ujawniło jej istotną rolę w umożliwianiu instytucjom finansowym podejmowania trafnych decyzji związanych z udzielaniem kredytów. Następnie omówiono temat tworzenia punktacji kredytowej, ujawniając kryteria wykorzystywane do oceny ryzyka w branży bankowej. Następnie skupiono się na ewolucji wynikającej z coraz większej dostępności ogromnych ilości danych oraz trudnościach w ich analizie w procesie weryfikacji kredytowej. Zreazliowano również wprowadzenie do technik uczenia maszynowego stosowanych w scoringu kredytowym, co ukazało, aktualne osiągnięcia technologiczne w dziedzinie wyjaśnialnych modeli predykcyjnych.

Kolejnym krokiem było skupienie się na teorii dotyczącej zabezpieczania procesu uczenia maszynowego przed atakami. Omówiono zagadnienia związane z ryzykiem i bezpieczeństwem algorytmów. Poznanie tych konceptów poszerzyło świadomość o aktualnym stanie wiedzy w tej dziedzinie oraz ukazało, jak duże jest zagrożenie wynikające z działań przeciwników, którzy starają się wykorzystać słabości modeli do manipulacji. Przedstawienie różnych rodzajów ataków na algorytmy uczenia maszynowego rzuciło światło na tę kwestię i podkreśliło potrzebę analizy oraz wdrożenia środków obronnych. W części praktycznej opisano zrealizowany proces tworzenia modelu przewidywania ryzyka niewypłacalności w ciągu pierwszych dwunastu miesięcy po udzieleniu kredytu, wykorzystując do tego celu algorytm XGBoost. Wykorzystany zestaw danych zawierał informacje o klientach ubiegających się zarówno o kredyty gotówkowe, jak i ratalne. Przeanalizowano dobór danych, jakość zbiorów oraz użyte narzędzia programistyczne, a skuteczność modelu oceniano przy użyciu wskaźnika Gini. Pracę sfinalizowano poprzez weryfikację postawionych pytań badawczych, które bezpośrednio nawiązywały do kolejnych scenariuszy ataku.

Celem pracy magisterskiej było przeprowadzenie dokładnej analizy bieżącego stanu wiedzy i praktycznych dokonań w zakresie wykorzystania modeli predykcyjnych stosowanych w celu estymacji wiarygodności kredytowej wnioskującego o kredyt, jak również zbadanie podatności tego typu narzędzi na potencjalne ataki. Cel dotyczący analizy teoretycznej został skutecznie osiągnięty na łamach dwóch pierwszych rodzajów pracy. W części trzeciej zbudowano model XGBoost odróżniających klientów ze względu na ich wiarygod-

ność kredytową, a następnie w rozdziale czwartym zweryfikowano postawione pytania badawcze. Wnioski z części praktycznej jednoznacznie wskazują na wysoką odporność modelu wzmocnionych drzew decyzyjnych na proste ataki z dziedziny wrogiego uczenia maszynowego. Model XGBoost radził sobie dobrze zarówno przy nieznaczących, jak i względnie dużych manipulacjach pojedynczymi zmiennymi. W przypadku ataków bezpośrednio na zbiór uczący, wykazał się wysoką odpornością na ingerencje w kilkuprocentowej części zbioru. W momencie zakłócenia większych części zbioru uczącego zauważono tendencje modelu do stania się niedostatecznie skutecznym dla praktycznego wykorzystania, jednakże duże ingerencje w dane nie są pożądane we wrogich atakach na uczenie maszynowe.

Potencjalnym kierunkiem rozwinięcia przeprowadzonych badań byłoby zrealizowanie ataków polegających na niewielkich manipulacjach wartościami więcej niż trzech zmiennych dla potencjalnego wnioskującego o kredyt. Przy podniesieniu wydajności kodu, można również spróbować większych manipulacji na mniejszej liczbie zmiennych, co ze względu na długi czas obliczeń zostało ograniczone w przeprowadzonej pracy. Warto zastanowić się na nieco bardziej skomplikowanym poszukiwaniu optymalnych kierunków odchylenia wartości zmiennych, co może oszczędzić generowania dużych zbiorów wrogich danych i podnieść skuteczność oszukiwania modelu. Zawsze w przypadku uczenia maszynowego, jak również innych poddziedzin big data, ograniczeniem są zastosowane dane modelowe, które mimo iż w przypadku tej pracy wydają się wystarczające, zwiększenie liczby zmiennych, jak i obserwacji, z pewnością stanowiłoby cenne ulepszenie procesu uczenia finalnego modelu.

Literatura

1. bankier.pl. (2007). Ryzykowny sektor. <https://www.bankier.pl/wiadomosc/Ryzykowny-sektor-1662142.html>, 03.12.2007.
2. bankier.pl. (2012). Tajemnicza liczba, czyli credit scoring. <https://www.bankier.pl/wiadomosc/Tajemnicza-liczba-czyli-credit-scoring-2512458.html>, 02.04.2012.
3. Biggio, B., Nelson, B., & Laskov, P. (2012). Poisoning attacks against support vector machines.
4. bik.pl. (2022). Jak poprawić swoją zdolność kredytową? <https://www.bik.pl/poradnik-bik/jak-poprawic-swoja-zdolnosc-kredytowa>, 5.10.2022.
5. Caire, D., Barton, S., de Zubiria, A., Alexiev, Z., & Dyer, J. (2006). A handbook for developing credit scoring systems in a microfinance context. Washington, Development Alternatives, Inc.
6. calcxml.com. (2023). Financial calculators. <https://www.calcxml.com/do/credit-score-calculator-new?skn=results>, 2023.
7. Castagno, P. (2020). How to identify spam using natural language processing (nlp)? <https://towardsdatascience.com/how-to-identify-spam-using-natural-language-processing-nlp-af91f4170113>, 2020.
8. Cheng, Q., Xu, A., Li, X., & Ding, L. (2022). Adversarial email generation against spam detection models through feature perturbation. Information Security Institute, Johns Hopkins University, Baltimore, MD; Department of Computer Science, American University, Washington, D.C.
9. crif.pl. (2018). Rola „machine learning” w procesach kredytowych. <https://www.crif.pl/wiadomo%C5%9Bci/dla-prasy/2020/sierpie%C5%84/rola-machine-learning-w-procesach-kredytowych/>, 2018.
10. datascience.eu. (2019). Why the xgboost machine learning algorithm is taking over? <https://datascience.eu/computer-programming/xgboost/>, 2019.
11. direct.money.pl. (2022). Co to jest scoring kredytowy? jak banki ustalają credit scoring i jakich używają systemów? <https://direct.money.pl/artykuly/porady/czym-jest-credit-scoring>,

18.01.2022.

12. ecomparemo.com. (2020). A brief history of credit scoring in the world. <https://www.ecomparemo.com/info/a-brief-history-of-credit-scoring-in-the-world>, 23.11.2020.
13. elektronikab2b.pl. (2021). Czym jest uczenie maszynowe i jak można je wykorzystać? <https://elektronikab2b.pl/biznes/53039-czym-jest-uczenie-maszynowe-i-jak-mozna-je-wykorzystac>, 11.12.2020.
14. experian.com. (2021). What is a good credit score? <https://www.experian.com/blogs/ask-experian/credit-education/score-basics/what-is-a-good-credit-score/>, 11.02.2021.
15. Fawcett, T. (2005). An introduction to roc analysis. Institute for the Study of Learning and Expertise, 2164 Staunton Court, Palo Alto, CA 94306, USA.
16. Feng, B., & Xue, W. (2021). Adversarial semi-supervised learning for corporate credit ratings. Center for Research on Intelligent Perception and Computing, National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Science, Beijing China, School of Artificial Intelligence, University of Chinese Academic of Science.
17. fico.com. (2023). Corporate information. <https://fico.gcs-web.com/corporate-information/>, 2023.
18. Gajowniczek, K., Ząbkowski, T., & Szupiluk, R. (2014). Estimating the roc curve and its significance for classification models' assessment. Warszawa, Department of Informatics, Faculty of Applied Informatics and Mathematics, Warsaw University of Life Sciences - SGGW; Szkoła Główna Handlowa.
19. gov.pl. (2021). Co to jest uczenie maszynowe – inteligentna analiza danych? <https://www.gov.pl/web/popcwsparcie/co-to-jest-uczenie-maszynowe-inteligentna-analiza-danych>, 15.06.2021.
20. habza.com.pl. (2022). Zdolność kredytowa a stopy procentowe. <https://habza.com.pl/zdolnosc-kredytowa-a-stop-y-procentowe/>, 15.06.2022.
21. ibm.com. (2021). Sposób działania algorytmu svm. <https://www.ibm.com/docs/pl/spss-modeler/saas?topic=models-how-svm-works>, 17.08.2021.
22. Kuchipudi, B., Nannapaneni, R. T., & Liao, Q. (2020). Adversarial machine

learning for spam filters. Department of Computer Science Central Michigan University Mt. Pleasant, Michigan, USA.

23. Matuszyk, A. (2009). Dotychczasowe oraz nowe trendy w metodzie "credit scoring".
24. mfiles.pl. (2020). Credit scoring. https://mfiles.pl/pl/index.php/Credit_scoring, 19.05.2020.
25. openai.com. (2017). Attacking machine learning with adversarial examples. <https://openai.com/research/attacking-machine-learning-with-adversarial-examples>, 24.02.2017.
26. Pawlicki, M. (2020). Zastosowanie metod uczenia maszynowego do wykrywania ataków sieciowych. Bydgoszcz, Uniwersytet Technologiczno-Przyrodniczy im. Jana i Jędrzeja Śniadeckich.
27. pkobp.pl. (2018). Kredyt na klik – big data w scoringu kredytowym. <https://bankomania.pkobp.pl/bankofinanse/nowe-technologie/kredyt-na-klik-big-data-w-scoringu-kredytowym/>, 2018.
28. pl.economy_pedia.com. (2021). Punktacja kredytowa. https://pl.economy_pedia.com/11030209-credit-scoring, 2021.
29. Poon, M. (2007). Scorecards and devices for consumer credits: The case of fair, isaac and company incorporated. *The Sociological Review*, Issue Supplement S2, 55, s. 284–306.
30. prawnicydotblog.wordpress.com. (2019). Credit scoring. <https://prawnicydotblog.wordpress.com/2019/04/08/credit-scoring/>, 08.04.2019.
31. Prokopowicz, D. (2014). Credit scoring w kontekście doskonalenia procesu zarządzania ryzykiem kredytowy. Wyższa Szkoła Przedsiębiorczości i Nauk Społecznych w Otwocku.
32. Przanowski, K. (2014). Credit scoring w erze big-data. Warszawa, Szkoła Główna Handlowa.
33. Przanowski, K. (2015). Credit scoring : Studia przypadków procesów biznesowych. Warszawa, Szkoła Główna Handlowa.
34. Przanowski, K. (2023). Credit scoring - automatyzacja procesu biznesowego - prezentacja do przedmiotu. Warszawa, Szkoła Główna Handlowa.

35. python.org. (2023). math — mathematical functions.
<https://docs.python.org/3/library/math.html>, 2023.
36. researchgate.net. (2017). 5 v's of big data.
https://www.researchgate.net/figure/The-5V-of-Big-Data-Characteristics_fig1_321050765, październik 2017.
37. sas.com. (2018). Cztery typy uczenia maszynowego.
https://www.sas.com/pl_pl/news/informacje-prasowe-pl/2018/cztery-typy-uczenia-maszynowego.html, 22.08.2018.
38. scoringexpert.pl. (2018). 8 ważnych informacji potrzebnych do zrozumienia nowej oceny punktowej, którą bik sprzedaje konsumentom.
<http://scoringexpert.pl/2018/02/21/ocena-punktowa-bik-skala-do-100/>, 21.02.2018.
39. Shi, Y., Sagduyu, Y., Davaslioglu, K., & Li, J. (2019). Generative adversarial networks for black-box api attacks with limited training data.
40. Shi, Y., Sagduyu, Y., & Grushin, A. (2017). How to steal a machine learning classifier with deep learning. Rockville, MD 20855, USA, Intelligent Automation, Inc.,.
41. Short, A., Pay, T. L., & Gandhi, A. (2019). Defending against adversarial examples. y Sandia National Laboratories, operated for the United States Department of Energy by National Technology Engineering Solutions of Sandia, LLC.
42. Siddiqi, N. (2016). Credit risk scorecards developing and implementing intelligent credit scoring. New Jersey, John Wiley Sons, Inc.
43. StatSoft. (2010). Zastosowanie metod scoringowych w działalności bankowej. https://media.statsoft.pl/_old_dnn/downloads/zast_met_skoringowych_w_dz_bankowej.pdf, 2010.
44. Surma, J. (2020). Prezentacja pt. hakowanie sztucznej inteligencji. Warszawa, Szkoła Główna Handlowa.
45. techtarget.com. (2021). 5 v's of big data.
<https://www.techtarget.com/searchdatamanagement/definition/5-Vs-of-big-data>, marzec 2021.
46. Thomas, L., Edelman, D., & Crook, J. (2002). Credit scoring and its applications. Philadelphia, Society for Industrial and Applied Mathematics.

47. Thonabauer, G., & Nosslinger, B. (2004). Guidelines on credit risk management. credit approval process and credit risk management. Oesterreichische Nationalbank and Austrian Financial Market Authority.
48. totalmoney.pl. (2020). Ocena punktowa bik-u – jaka wartość scoringu bik-u jest dobra i daje szansę na kredyt? <https://www.totalmoney.pl/artykuly/179147,kredyty-gotowkowe,ocena-punktowa-bik-u—jaka-wartosc-scoringu-bik-u-jest-dobra-i-daje-szanse-na-kredyt,1,1>, 03.10.2020.
49. towardsdatascience.com. (2021). What is adversarial machine learning? <https://towardsdatascience.com/what-is-adversarial-machine-learning-dbe7110433d6>, 12.07.2021.
50. Weston, L. (2012). Your credit score. New Jersey, Pearson Education, Inc.
51. Wikipedia. (2022). Fico. <https://en.wikipedia.org/wiki/FICO>, 24.12.2022.
52. Wyśiński, P. (2013). Zastosowanie scoringu kredytowego w bankowości. Gdańsk, Uniwersytet Gdański.
53. zephyrnet.com. (2022). Co to jest kontradyktoryjne uczenie maszynowe? <https://zephyrnet.com/pl/co-to-jest-kontradyktoryjne-uczenie-maszynowe/>, 3.03.2022.

Spis rysunków

1	Przykład cechy wykorzystanej w scoringu kredytowym(bankier.pl, 2012) . . .	8
2	Diagram oceny wiarygodności kredytowej według firmy FICO(forbes.com, 2021)	10
3	Fragment kalkulacji ze strony calcxml.com (calcxml.com, 2023)	11
4	Wizualizacja oceny punktowej w BIK (źródło opracowanie własne)	12
5	Schemat 5V big data(researchgate.net, 2017)	17
6	Krzywa Profit zależna od mocy predykcyjnej(Przanowski, 2023)	22
7	Krzywa ROC i jej możliwe warianty(Gajowniczek et al., 2014)	23
8	Przykład ataku na system rozpoznawania obrazów (openai.com, 2017) . . .	25
9	Widok z kamery przedniej samochodu autonomicznego. Właściwie rozpoznany znak STOP (Surma, 2020)	26
10	Widok z kamery przedniej samochodu autonomicznego. Niewłaściwie rozpoznany znak STOP (Surma, 2020)	26
11	Widok z kamery przedniej samochodu autonomicznego. Znak STOP błędnie rozpoznany jako znak bezwzględnej pierwszeństwa przy skręcie w lewo (Surma, 2020)	27
12	Schemat procesu kontradyktoryjnego uczenia częściowo nadzorowanego dla korporacyjnego ratingu kredytowego (Feng B., 2021)	31
13	Rozkład zmiennej 'act_age' w zależności od wartości zmiennej 'default12' (źródło opracowanie własne)	41
14	Rozkład zmiennej 'act_call_cc' w zależności od wartości zmiennej 'default12' (źródło opracowanie własne)	42
15	Rozkład zmiennej 'act_call_cc' w zależności od wartości zmiennej 'default12' na wykresie wiolinowym (źródło opracowanie własne)	42
16	Rozkład zmiennej 'act_call_cc' w zależności od wartości zmiennej 'default12' na wykresie pudełkowym (źródło opracowanie własne)	43
17	Rozkład zmiennej 'act_ccss_n_statC' w zależności od wartości zmiennej 'default12' (źródło opracowanie własne)	44
18	Rozkład zmiennej 'act_ccss_n_statC' w zależności od wartości zmiennej 'default12' na wykresie wiolinowym (źródło opracowanie własne)	44

19	Rozkład zmiennej 'act_ccss_n_statC' w zależności od wartości zmiennej 'default12' na wykresie pudełkowym (źródło opracowanie własne)	45
20	Analiza ważności zmiennych dla modelu wstępnego (źródło opracowanie własne)	48
21	Wizualizacja wybranego modelu krzywymi ROC (źródło opracowanie własne)	51
22	Analiza ważności zmiennych dla finalnego modelu (źródło opracowanie własne)	51
23	Wykres wartości współczynnika Giniego dla zbioru treningowego i testowego dla pytania 1 (źródło opracowanie własne)	57
24	Wykres wartości współczynnika Giniego dla zbioru treningowego i testowego dla pytania 2 (źródło opracowanie własne)	59

Spis tabel

1	Interpretacja oceny punktowej BIK(scoringexpert.pl, 2017)	13
2	Klasyczna karta oceny punktowej(Przanowski, 2023)	16
3	Skuteczność klasyfikacji dla poszczególnych modeli sprawdzanych w badaniu korporacyjnych ratingów kredytowych(Feng B., 2021)	32
4	Wrogie próbki zastosowane na atakowanym modelu filtra antyspamowego (Kuchipudi et al., 2020)	35
5	Opis zmiennych oraz grup zmiennych w zbiorze danych (źródło opracowanie własne)	38
6	Podstawowe statystyki zmiennej 'act_age' (źródło opracowanie własne)	40
7	Podstawowe statystyki zmiennej 'act_call_cc' (źródło opracowanie własne) . .	41
8	Podstawowe statystyki zmiennej 'act_ccss_n_statC' (źródło opracowanie własne)	43
9	Parametry funkcji 'train' z biblioteki 'xgboost' zastosowane do wyboru odpowiedniego modelu (źródło opracowanie własne)	50
10	Parametry finalnego modelu XGBoost (źródło opracowanie własne)	50
11	Wartość współczynnika Giniego w zależności od procenta atakowanych danych dla pytania 1 (źródło opracowanie własne)	57
12	Wartość współczynnika Giniego w zależności od procenta atakowanych danych dla pytania 2 (źródło opracowanie własne)	59
13	Wyniki testów weryfikacyjnych pytania 3 (źródło opracowanie własne) . . .	62
14	Wyniki testów weryfikacyjnych pytania 4 (źródło opracowanie własne) . . .	64

Załączniki

1. Plik VariablesDescription.xlsx z opisem zmiennych w zbiorze danych
2. Plik XGBoostModel_MK116111_1.html z kodem
3. Plik XGBoostModel_MK116111_1.ipynb z kodem
4. Plik abt_app.sas7bdat z danymi do modelu