1. Wprowadzenie

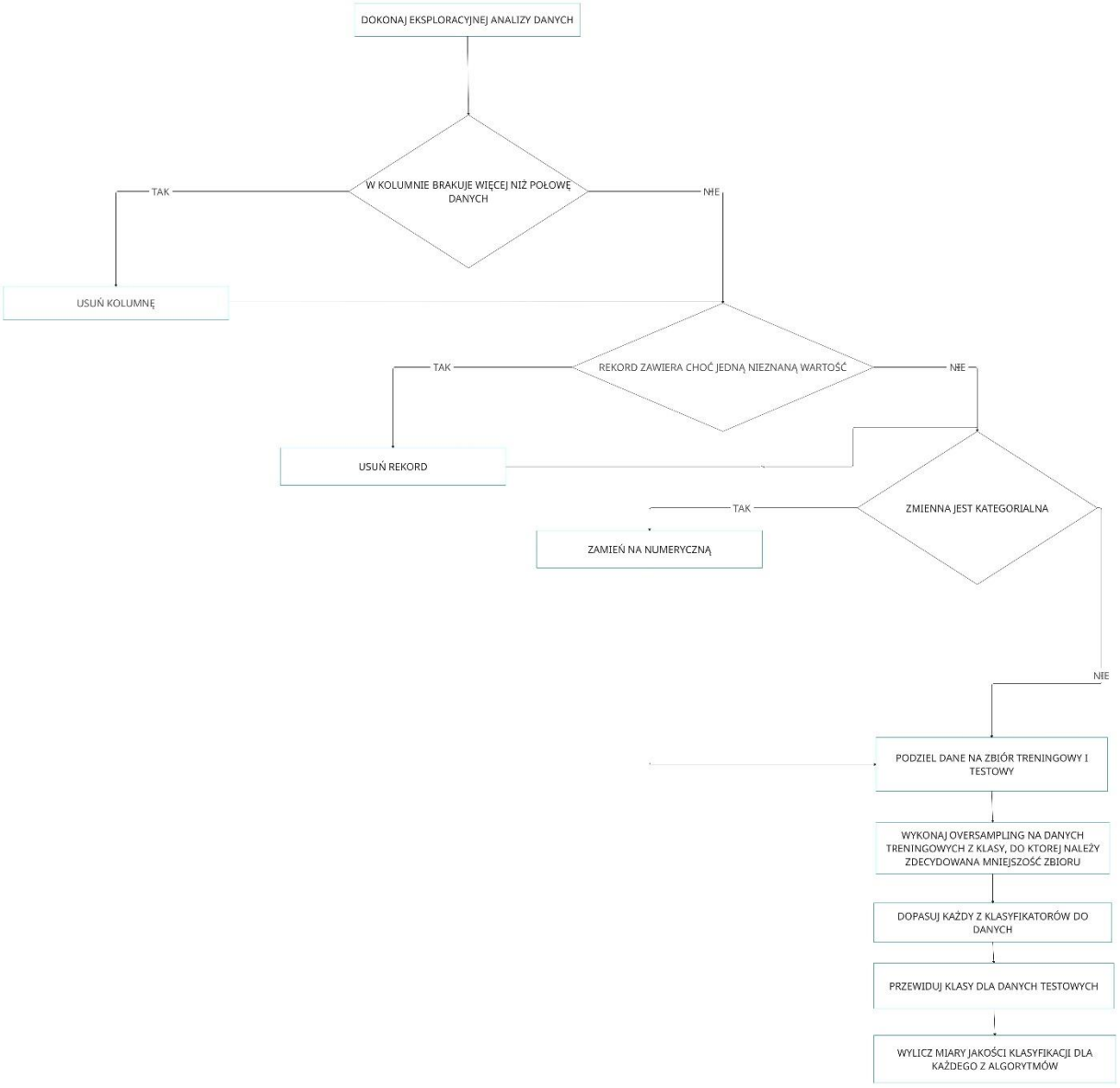
W opisywanym poniżej projekcie, korzystać będę ze zbioru danych dotyczącego klientów banku, który zawiera dane zarówno o samych klientach takie jak ich stan cywilny, wykształcenie, pożyczki czy też średnie roczne saldo, jak i o przeprowadzonych z nimi rozmowach telefonicznych. Zbiór pochodzi ze strony kaggle.com i ma 45211 rekordów i 17 cech.

Postaram się stworzyć model klasyfikacji, który będzie przewidywał czy dany klient zainwestuje w lokatę krótkoterminową. W tym celu dopasuję do danych treningowych różne klasyfikatory, które potem wykorzystam do predykcji klas dla danych testowych i porównam ze sobą wyniki za pomocą miar jakości klasyfikacji. Projekt ten, pozwoli lepiej zrozumieć algorytmy klasyfikacji, stwierdzić który z nich sprawdza się najlepiej w danej sytuacji, a także dowiedzieć się jak przeprowadzić obróbkę i przygotowanie danych, w celu osiągnięcia jak najlepszych wyników.

2. Opis przeprowadzonych badań

Pierwszym krokiem do osiągnięcia tego celu jest eksploracyjna analiza danych. Dzięki funkcjom wbudowanym w bibliotece pandas staram się dowiedzieć jak wyglądają dane zawarte w zbiorze przez zobaczenie jakie wartości znajdują się w każdej z kolumn, jakiego typu są to dane, a także ile znajduje się tam duplikatów i nieznanych wartości. Po eksploracji danych wiem już, które kolumny powinnam usunąć z powodu zbyt dużej ilości brakujących danych- w kolumnie ‘poutcome’ opisującej poprzedni skutek kampanii marketingowej brakuje aż 36959 danych, co stanowi większość zbioru. W przypadku kolumn ‘contact’(opisującej sposób kontaktu z klientem), ‘education’(stopień edukacji), ‘job’ (praca) te braki są zdecydowanie mniejsze- 13020, 1857 i 288, więc usuwam jedynie rekordy, w których występują brakujące dane. Dzięki sprawdzeniu, jaki typ danych znajduje się w kolumnach wiem , które zmienne powinnam zamienić- jeśli zmienna jest kategorialna zmieniam ją na numeryczną. W przypadku zmiennych, które przyjmują wartości ‘tak’ lub ‘nie’, zmieniam je na 0(w przypadku ‘nie’) i 1(w przypadku ‘tak’). Jeśli zmienna przyjmuje więcej niż 2 wartości zamieniam ją przy użyciu funkcji get_dummies z biblioteki pandas, która każdą kategorię zamienia na osobną kolumnę zawierającą zmienną binarną (0 kiedy dana obserwacja nie należy do tej kategorii, 1 kiedy należy). Wykonuję standaryzację danych, aby wszystkie dane były w tej samej skali, a także aby cechy o większej wariancji nie miały dominującego wpływu na proces klasyfikacji. Dzielę zbiór na dane testowe i treningowe, aby móc wypróbować nauczony model na innych danych. Potem testuję różne algorytmy klasyfikacji dopasowując je do danych treningowych i sprawdzając jak sobie radzą na danych testowych używając różnego rodzaju miar jakości klasyfikacji takich jak precyzja, czułość f1 i dokładność.

SCHEMAT DZIAŁANIA

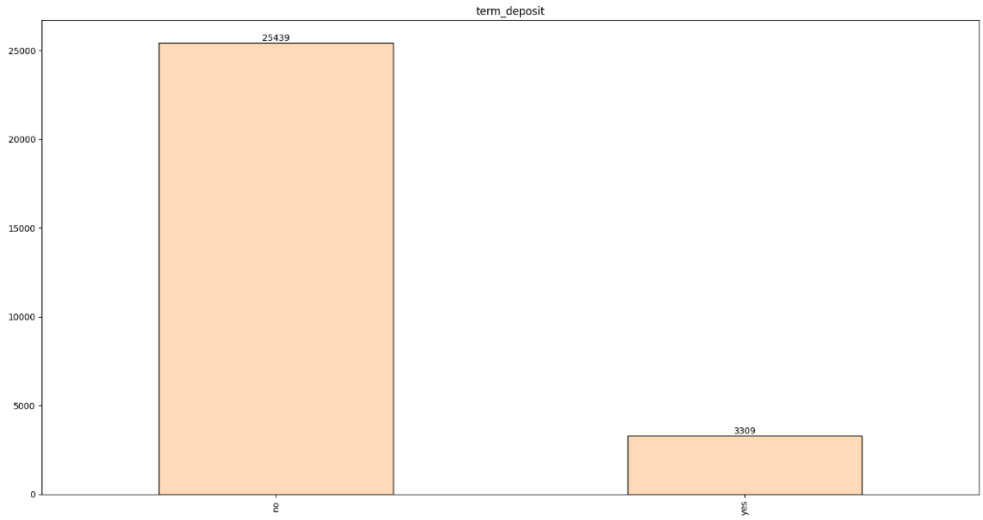


3. Wyniki

3.1 ILE NIEZNANYCH WARTOŚCI W KOLUMNIE
age : 0
job : 264
marital : 0
education : 1690
default : 0
balance : 0
housing : 0
loan : 0
contact : 12776
day : 0
month : 0
duration : 0
campaign : 0
pdays : 0
previous : 0
poutcome : 36085
term_deposit : 0

3.2 WARTOŚCI W KOLUMNIE DLA DANYCH KATEGORIALNYCH
job : ['admin.', 'blue-collar', 'entrepreneur', 'housemaid', 'management', 'retired', 'self-employed', 'services', 'student', 'technician', 'unemployed', 'unknown']
marital : ['divorced', 'married', 'single']
education : ['primary', 'secondary', 'tertiary', 'unknown']
default : ['no', 'yes']
housing : ['no', 'yes']
loan : ['no', 'yes']
contact : ['cellular', 'telephone', 'unknown']
month : ['apr', 'aug', 'dec', 'feb', 'jan', 'jul', 'jun', 'mar', 'may', 'nov', 'oct', 'sep']
poutcome : ['failure', 'other', 'success', 'unknown']
term_deposit : ['no', 'yes']

3.3 ROZKŁAD KLAS



MIARY JAKOŚCI KLASYFIKACJI DLA KAŻDEGO Z ALGORYTMÓW:

a) K-najbliższych sąsiadów

Czułość	0.607871720116618
Precyzja	0.3601036269430052
F1	0.45227765726681124
Dokładność	0.8243478260869566
Tablica pomyłek	[[4323 741] [269 417]]

b) Drzewo decyzyjne

Czułość	0.4139941690962099
Precyzja	0.44724409448818897
F1	0.4299772899318698
Dokładność	0.8690434782608696
Tablica pomyłek	[[4713 351] [402 284]]

c) Naiwny Bayes

Czułość	0.4897959183673469
Precyzja	0.40481927710843374
F1	0.4432717678100264
Dokładność	0.8532173913043478
Tablica pomyłek	[[4570 494] [350 336]]

d) GradientBoosting (Wzmocnienie gradientowe)

Czułość	0.8338192419825073
Precyzja	0.41210374639769454
F1	0.5515911282545806
Dokładność	0.8382608695652174
Tablica pomyłek	[[4248 816] [114 572]]

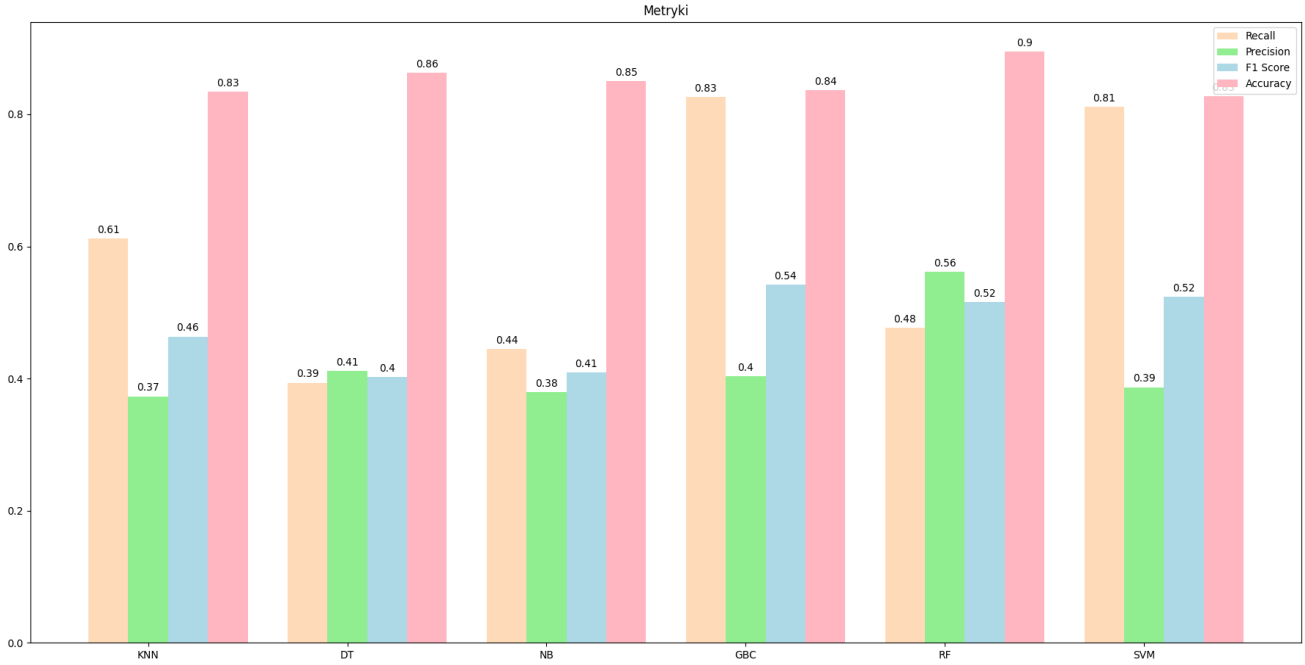
e) Losowy las

Czułość	0.4679300291545189
Precyzja	0.5631578947368421
F1	0.5111464968152866
Dokładność	0.8932173913043479
Tablica pomyłek	[[4815 249] [365 321]]

f) Maszyna wektorów nośnych

Czułość	0.8338192419825073
Precyzja	0.411214953271028
F1	0.5507944150216658
Dokładność	0.8377391304347827
Tablica pomyłek	[[4245 819] [114 572]]

PORÓWNANIE WYNIKÓW NA WYKRESIE



4. Analiza

Dokładność większości tych klasyfikatorów jest podobna, mieszcząca się w granicach 0.83-0.90, co oznacza, że w każdym z przypadków 83 do 90% danych jest przyporządkowywana do właściwej klasy. Pozostałe parametry są zdecydowanie bardziej zróżnicowane. Precyzja informująca o tym, jaka część osób, która według naszych predykcji miała zainwestować w lokatę krótkoterminową faktycznie w rzeczywistości ją wzięła jest dosyć niska – waha się w granicach 0.37-0.56. Czułość, która mówi nam o tym jaka część osób, która faktycznie zainwestowała w lokatę została przez nas przypisana do tej klasy ma bardzo rozbieżne wartości w zależności od użytego algorytmu klasyfikacji- od najgorzej w tej kategorii wypadającego algorytmu drzewa decyzyjnego, dla którego jest ona równa 0.39, aż do najlepszego wzmocnienia gradientowego aż 0.83. Wskaźnik F1, który może być dobrą miarą jakości modelu, która łączy w sobie precyzję i czułość, bo jest on ich średnią harmoniczną mieści się w przedziale od 0.4 (dla drzewa decyzyjnego) do 0.54(dla wzmocnienia gradientowego).

5. Wnioski

Ciężko jest wybrać jedną uniwersalną miarę, która określi jednoznacznie jakość klasyfikatora. Wiele zależy od rozwiązywanego problemu, w tym przypadku, w którym próbujemy przewidzieć inwestycję klientów w lokatę, dokładność nie jest najważniejszą miarą. Mimo, że w każdym klasyfikatorze jest dość wysoka, to jednak zdecydowana większość osób nie zainwestuje w lokatę – wiemy to z początkowej eksploracyjnej analizy naszego zbioru. Stąd też ogólny procent poprawnie sklasyfikowanych przypadków nie będzie czymś na podstawie czego będziemy mogli właściwie ocenić jakość modelu. Precyzja jest już przydatniejszą miarą, ponieważ wiedza o tym jaka część naszych predykcji o dokonaniu inwestycji jest słuszna może okazać się przydatna. Niestety w wyżej przedstawionym przypadku przy użyciu żadnego z algorytmów nasze przewidywania dotyczące mniej licznej z klas nie były do końca precyzyjne z powodu małej ilości danych z tej grupy, co powodowało konieczność wykonania oversamplingu, który nie jest zdolny w 100% zastąpić rzeczywistych danych. W takim niezbalansowanym zbiorze niska precyzja jest dość częstym zjawiskiem. Aby poprawić precyzję powinno się zebrać więcej danych z klasy mniejszościowej, wtedy model lepiej nauczyłby się ją klasyfikować, co poskutkowałoby zwiększeniem tego parametru. Dlatego precyzja również nie jest do końca dobrym dla naszego problemu kryterium porównawczym. Miarą, na którą powinniśmy najbardziej zwrócić uwagę jest czułość, bo ona mówi nam jaką część osób, która tej inwestycji dokonała faktycznie została przez nasz model przypisana do właściwej klasy, co jest najbardziej użyteczne w rozpatrywanej sytuacji. Dużo lepiej jest wyłapać jak najwięcej z pozytywnych przypadków, nawet kosztem tego, że w tej klasie znajdzie się więcej tych fałszywie pozytywnych, bo w przeciwnym razie możemy stracić potencjalnych klientów. Możemy też użyć wskaźnika F1 jeśli jednak chcemy wybrać algorytm, który ma najlepszy związek precyzji i czułości. W obu tych przypadkach do naszej predykcji najlepiej sprawdził się algorytm wzmocnienia gradientowego. Wybór najbardziej odpowiedniego algorytmu może być różny w zależności od potrzeb, natomiast w przedstawionym powyżej przypadku najlepiej sprawdzić się powinno wzmocnienie gradientowe, ponieważ powinno „wyłapać” ze zbioru danych jak najwięcej danych osób, które zainwestują w lokatę co gwarantuje nam wysoka czułość, a jego dokładność również jest na dosyć dobrym poziomie.

Powyższe rozważania pokazują, że miary jakości mogą być bardzo zróżnicowane i to, który algorytm wybierzemy może się różnić w zależności od tego co chcielibyśmy osiągnąć. Chcąc rozwiązać rożne problemy musimy zwracać uwagę na inne aspekty i ocenić, które z nich pozwolą nam osiągnąć zamierzony cel.