Las losowy z naiwnym klasyfikatorem bayesowskim w zadaniu klasyfikacji

Hubert Rutkowski i Adam Szumada

Temat

Las losowy z naiwnym klasyfikatorem bayesowskim (NBC) w zadaniu klasyfikacji. Postępujemy tak jak przy tworzeniu lasu losowego, tylko co drugi klasyfikator w lesie to NBC. Jeden z klasyfikatorów (NBC lub drzewo ID3) może pochodzić z istniejącej implementacji.

Interpretacja tematu projektu / zadania

Badania prowadzone w ramach realizacji projektu mają na celu zdobycie fundamentalnej wiedzy w zakresie klasycznych algorytmów uczenia maszynowego, w szczególności lasu losowego, drzewa decyzyjnego oraz naiwnego klasyfikatora Bayesa. W ramach projektu zaimplementowana zostanie jedna z modyfikacji klasycznego lasu losowego, w której na przemian stosowane będą dwa klasyfikatory: drzewo decyzyjne ID3 oraz naiwny klasyfikator bayesowski (NBC). W zadaniu umożliwiono wykorzystanie istniejącej już implementacji dla jednego z klasyfikatorów, więc w celu realizacji drzewa ID3 posłużymy się gotową implementacją z biblioteki.
W ramach badań wykonana zostanie seria eksperymentów, które pozwolą miarodajnie ocenić działanie zmodyfikowanego w ten sposób algorytmu. Przede wszystkim przebadane zostanie zachowanie algorytmu w różnych przypadkach, w tym dla zbiorów danych o różnym charakterze czy zmianie parametrów.

Środowisko implementacji i biblioteki (techstack)

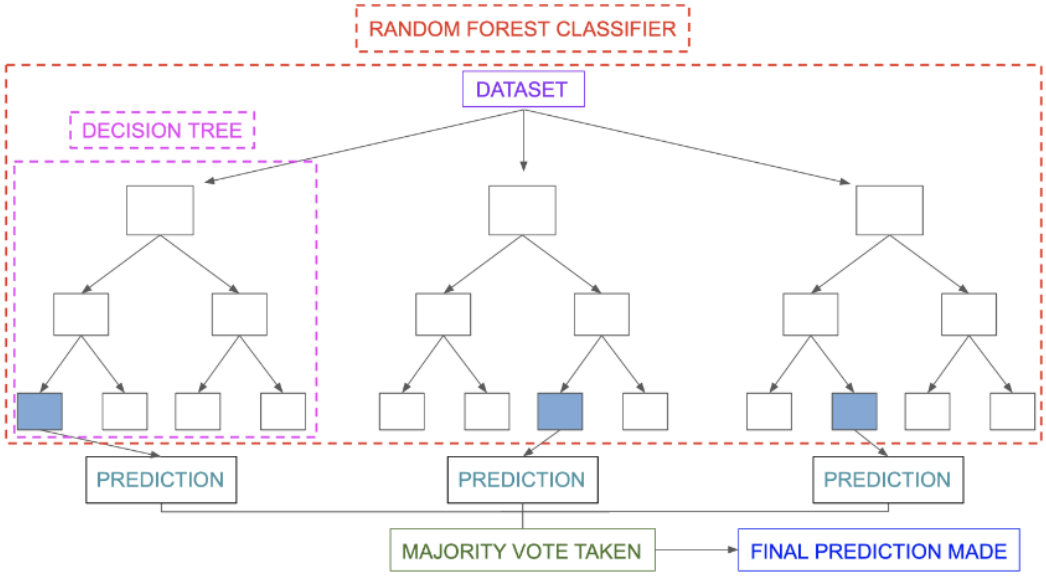
Implementacja algorytmu i badania zostaną przeprowadzone przy pomocy jednego z najpopularniejszych, darmowych języków programowania zwanego Python. Ponadto programy będą działały pod kontrolą systemu Ubuntu Linux 22.04.
Przy okazji używania Pythona potrzebne będą również specjalne biblioteki umożliwiające realizację projektu. Najprawdopodobniej zastosowane zostaną pandas, numpy, sklearn, id3. Niemniej jednak z nieznanych jeszcze w tym momencie powodów zastrzega się możliwość zastosowania innych bibliotek. W takim przypadku zostanie to odnotowane w dokumentacji końcowej wraz z odpowiednim uzasadnieniem.

Opis używanych algorytmów

Klasyfikacja w lesie losowym

Las losowy jest to maszynowa metoda uczenia się algorytmów klasyfikacyjnych, składająca się z drzew decyzyjnych. Drzewa decyzyjne opierają się na cechach losowych i każde z nich tworzy predykcję dla danych wejściowych. W wyniku działania tego algorytmu wybierana jest predykcja występująca najczęściej. Zastosowanie lasu losowego pozwala na zgromadzenie danych od wielu niezależnych od siebie modeli. W praktyce przekłada się to na większą wiarygodność wyników niż w przypadku użycia pojedynczego drzewa decyzyjnego.

Las losowy jest algorytmem zespołowym tzn. łączy kilka technik uczenia maszynowego w jeden spójny model w celu zmniejszenia wariancji (technika bagging), błędu systematycznego i ogólnej poprawy przewidywań. Modele uczone w lesie losowym, wykorzystują do nauki przykłady, których cechy wybierane są w sposób losowy dla każdego modelu. To pozwala uzyskać większą różnorodność modeli.



Źródło: https://www.section.io/engineering-education/introduction-to-random-forest-in-machine-learning/

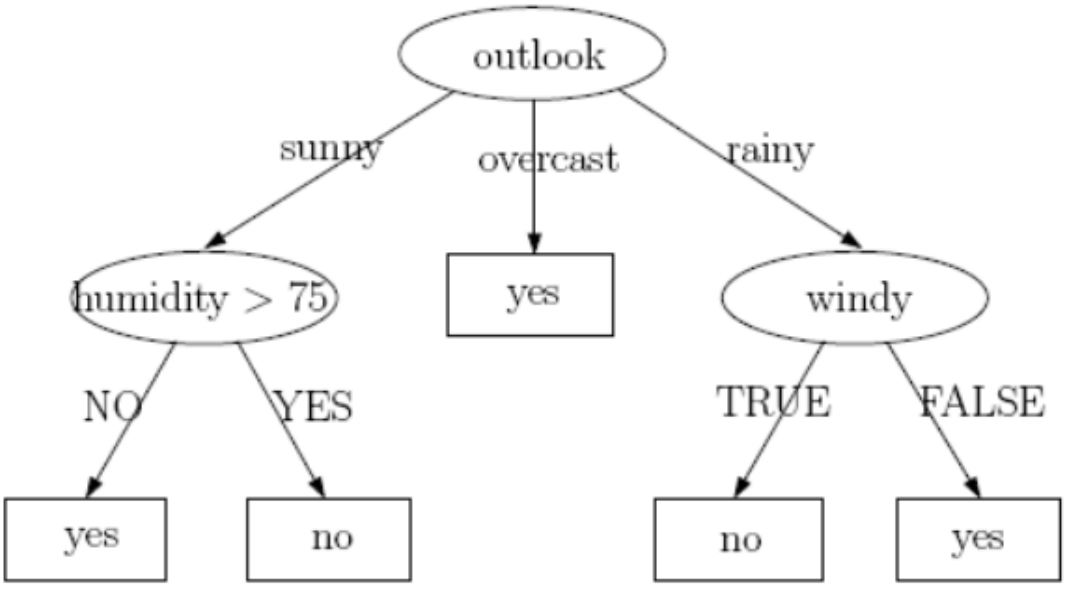
Na powyższym zdjęciu przedstawione są trzy drzewa decyzyjne i wygenerowane przez nie predykcje. Następnie wybierana jest finalna predykcja występująca najczęściej.

Drzewo decyzyjne (algorytm ID3)

Algorytm ID3 stosowany jest w celu budowy drzew decyzyjnych.
Same drzewa decyzyjne mają hierarchiczną, drzewiastą strukturę. Są to etykietowane drzewa, w których każdy wewnętrzny węzeł odpowiada przeprowadzeniu pewnego testu na wartościach atrybutów. Z każdego takiego węzła wychodzi tyle gałęzi ile jest możliwych rezultatów (wyników) testu. Każdy liść zawiera decyzję o klasyfikacji danego przykładu.

Przykład tablicy decyzyjnej i drzewa decyzyjnego, które jej odpowiada:

a_1	a_2	a_3	a_4	dec
outlook	temp.	humid.	windy	play
sunny	85	85	FALSE	no
sunny	80	90	TRUE	no
overcast	83	86	FALSE	yes
rainy	70	96	FALSE	yes
rainy	68	80	FALSE	yes
rainy	65	70	TRUE	no
overcast	64	65	TRUE	yes
sunny	72	95	FALSE	no
sunny	69	70	FALSE	yes
rainy	75	80	FALSE	yes
sunny	75	70	TRUE	yes
overcast	72	90	TRUE	yes
overcast	81	75	FALSE	yes
rainy	71	91	TRUE	no



Źródło: https://edu.pjwstk.edu.pl/wyklady/adn/scb/wyklad10/w10.htm

Analiza przytoczonego drzewa decyzyjnego:
Jak widać na przykładzie korzeń dotyczy atrybutu outlook, z którego wychodzą 3 gałęzie (3 możliwe wartości): sunny, overcast, rainy. Następnie w węzłach drzewa zauważyć można kolejne atrybuty, a od nich w gałęziach wartości jakie mogą przyjąć. Dla windy przyjmowane wartości można określić jako binarne (0 – FALSE, 1 – TRUE). Natomiast dla humidity istnieje wiele różnych wartości, które można kategoryzować względem np. wartości większej od X (tutaj 75). Jeżeli chodzi o liście to znajdują się tam możliwe rezultaty kategoryzacji (różne klasy), które w analizowanym przykładzie oznaczają decyzję (tak lub nie).

Predykcja na podstawie drzewa decyzyjnego zachodzi poprzez przejście przez całe drzewo od korzenia aż do liścia zgodnie z wartościami atrybutów odwiedzonych węzłów.

Drzewo decyzyjne ID3 jest zachłannym algorytmem klasyfikacji, który konstruowany jest rekurencyjnie. Wejście algorytmu stanowią:

- zbiór testujący,
- zbiór atrybutów kategorycznych.

Schemat działania algorytmu:

1. Oryginalny zestaw danych jest węzłem głównym.
2. Poszukiwanie atrybutu , który najlepiej podzieli dane trenujące. Najczęściej kryterium wyboru to minimalna wartość entropii lub maksymalna wartość przyrostu informacji.
3. Zbiór danych zostaje podzielony według wybranego atrybutu w celu uzyskania podzbiorów danych.
4. Algorytm powtarza się w każdym podzbiorze, biorąc pod uwagę tylko niewybrane wcześniej atrybuty.

Kryteria wyboru atrybutu:

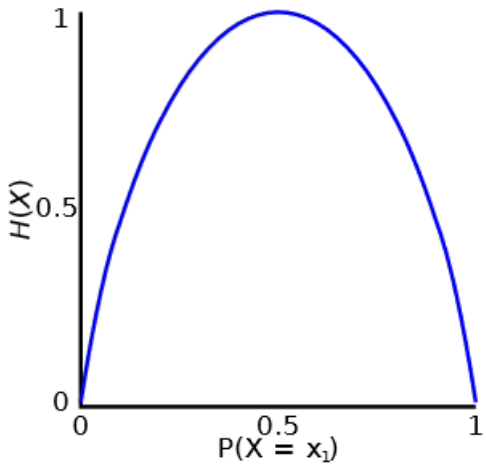
- Entropia:
  Jest to miara losowości przetwarzanych informacji (miara zróżnicowania danych). Wysoka entropia przekłada się bezpośrednio na dużą trudność wyciągnięcia wniosków z informacji.
  Entropia dla zbioru X można opisać następującym równaniem:

$$H(S) = -\sum_{x \in \mathrm{X}} p(x) \cdot log_2(p(x))$$

, gdzie:

- S – aktualny zbiór danych, dla którego liczona jest entropia
- X – zbiór klas w zbiorze S
- p(x) – stosunek liczby elementów z klasy x do elementów w zbiorze S

Dla uproszczenia zrozumienia działania entropii przeanalizujemy przypadek tylko dwóch możliwości wartości (x_1 i x_2) zmiennej losowej X, dla różnych prawdopodobieństw wylosowania x_1. W takiej sytuacji wykres entropii prezentuje się następująco:



Entropia osiąga minimum (0), gdy któraś z wartości zmiennej losowej X ma prawdopodobieństwo wylosowania równe 1 oraz wartość maksymalną (1) przy takim samym prawdopodobieństwie obydwu wartości ($p(x_1) = p(x_2) = 0.5$).

- Przyrost informacji:
  Jest to miara kryterium wyboru ściśle związana z entropią. Dokładniej jest to różnica entropii przed i po rozbiciu zbioru danych przy pomocy atrybutu d.
  Można go zatem opisać następującym równaniem:

$$H_X(S) = H_X(S) - H_{X,d}(S, d)$$

Naiwny klasyfikator bayesowski

Naiwny klasyfikator bayesowski jest statystycznym klasyfikatorem opartym na twierdzeniu Bayesa. Jest to jedna z metod uczenia maszynowego wykorzystywana do rozwiązywania problemu sortowania i klasyfikacji. Naiwny klasyfikator bayesowski zakłada, że wartości atrybutów w klasach są niezależne.

Twierdzenie Bayesa

Twierdzenie Bayesa pozwala obliczyć prawdopodobieństwo wystąpienia zdarzenia, biorąc pod uwagę prawdopodobieństwo wystąpienia innego zdarzenia, które już wystąpiło. Matematycznie, twierdzenie Bayesa wyrażane jest w następujący sposób:

$$\mathrm{P(A|B)} = \mathrm{P(B|A)} \cdot \frac{\mathrm{P(A)}}{\mathrm{P(B)}}$$

Outlook	Temperature	Humidity	Windy	Play football
Rainy	Hot	High	False	No

Tabela 1. Fragment przykładowego zbioru danych

Analizując zaprezentowany powyżej fragment zbioru danych, założenia przyjęte przez algorytm mogą być następujące:

- żadna para cech nie jest współzależna, przykładowo: wysoka wilgotność nie ma związku z brakiem wiatru,
- każda z cech ma taką samą wagę, tj. znając samą temperaturę i wietrzność algorytm nie jest w stanie trafnie przewidzieć wyniku.

Podsumowując, fundamentalnym założeniem naiwnego Bayesa jest to, że każda cecha ma niezależny i równy wkład w wynik. Warto jednak zaznaczyć, że w świecie rzeczywistym założenie o niezależności przyjęte przez algorytm zwykle nie jest poprawne. Mimo to, w praktyce często zwraca pożądane wyniki.
Dla tak przygotowanych danych można zastosować twierdzenie Bayesa w postaci:

$$P(y|X) = \frac{P(X|y) \cdot P(y)}{P(X)}$$

, gdzie:

- y – jest to klasa zmiennej,
- X - jest wektorem wartości atrybutów gdzie: X = (x_1, x_2, …, x_n), gdzie kolejne wartości x są pojedynczymi atrybutami.

Jest to postać twierdzenia wykorzystywana na potrzeby uczenia maszynowego. Dla fragmentu zaprezentowanego w Tabeli 1 podajemy przykład wektora cech i odpowiadającej mu klasy (etykietowania):
X = (Rainy, Hot, High, False)
y = No.

W tym przypadku $P(y|X)$ oznacza prawdopodobieństwo nie grania w piłkę biorąc pod uwagę konkretne parametry pogodowe.

Podstawiając za X i rozwijając za pomocą reguły łańcuchowej, otrzymujemy postać:

$$P(y|x_1, x_2, \dots, x_n) = \frac{P(x_1|y) \cdot P(x_2|y) \cdot \dots \cdot P(x_n|y) \cdot P(y)}{P(x_1) \cdot P(x_2) \cdot \dots \cdot P(x_n)}$$

Ostatecznie, prawdopodobieństwo wystąpienia klasy y pod warunkiem wystąpienia atrybutów X obliczane jest za pomocą powyższego wzoru. Tworząc model klasyfikatora liczone jest prawdopodobieństwo danego zestawu danych wejściowych dla wszystkich możliwych wartości klasy y i wybierany jest wynik z największym prawdopodobieństwem.

Przykład:
Załóżmy na potrzeby przykładu, że atrybuty „Outlook” oraz „Temperature” mogą przyjmować 3 wartości, natomiast „Humidity”, „Windy” oraz „Play Football” po 2 wartości. Dodatkowo zakładamy, że Tabela 1. składa się z 14 różnych przypadków.
Obliczmy prawdopodobieństwo gry w piłkę w dniu dzisiejszym zakładając, że zbiór atrybutów wygląda następująco:
today = (Sunny, Hot, Normal, False) = (Outlook = Sunny, Tempertatue = Hot, Humidity = High, Windy = False)

P(Yes|today) = P(Sunny|Yes) · P(Hot|Yes) · P(Normal|Yes) · P(NoWind|Yes) · P(Yes) · 1/P(today)
P(No|today) = P(Sunny|No) · P(Hot|No) · P(Normal|No) · P(NoWind|No) · P(No) · 1/P(today)

Zakładając, że rozkład wyżej ujętych prawdopodobieństw wygląda następująco:
P(Sunny|Yes) = 2/9, P(Hot|Yes) = 2/9, P(Normal|Yes) = 6/9, P(NoWind|Yes) = 6/9,
P(Yes) = 9/14
oraz

P(Sunny|No) = 3/5, P(Hot| No) = 2/5, P(Normal| No) = 1/5, P(NoWind| No) = 2/5,
P(No) = 5/14

Ponieważ P(today) jest takie samo w obu przypadkach, możemy je pominąć, a obliczenia będą wyglądały następująco:
P(Yes|today) = 2/9 · 2/9 · 6/9 · 6/9 · 9/14 ≈ 0.0141
P(No|today) = 3/5 · 2/5 · 1/5 · 2/5 · 5/14 ≈ 0.0068

Jako, że P(Yes|today) + P(No|today) = 1, po normalizacji prawdopodobieństwa wyglądają następująco:
P(Yes|today) = 0.0141 / (0.0141 + 0.0068) ≈ 0.675
P(No|today) = 0.0068 / (0.0141 + 0.0068) ≈ 0.325

Jako, że prawdopodobieństwo gry w piłkę w dniu dzisiejszym jest większe niż prawdopodobieństwo nie grania, predykcja zwróci wartość „Yes”.

W zależności od typu rozkładu cech (ciągły bądź dyskretny), w celu klasyfikacji cechy na podstawie prawdopodobieństw przynależności do danej klasy wykorzystywane są różne klasyfikatory wykorzystujące twierdzenie Bayesa.

Gaussowski naiwny klasyfikator Bayesa – w przypadku tego klasyfikatora, zakłada się że wartości powiązane z cechami są ciągłe i mają rozkład zgodny z rozkładem Gaussa.

$$P(x_i \mid y) = \frac{1}{\sqrt{2\pi\sigma_y^2}} \exp\left(-\frac{(x_i - \mu_y)^2}{2\sigma_y^2}\right)$$

Zmienne o charakterze dyskretnym mogą być klasyfikowane przy pomocy wielomianowego naiwnego klasyfikatora Bayesa. W tym przypadku rozkład jest zwykle parametryzowany w postaci wektorów $\theta_y = (\theta_{y1}, \ldots, \theta_{yn})$ dla każdej klasy „y”, gdzie „n” jest liczbą atrybutów, a θ_{yi} jest prawdopodobieństwem $P(x_i|y)$ wystąpienia cechy „i” w próbce należącej do klasy y.

Plan eksperymentów

Biorąc pod uwagę fakt wykorzystywania liczb pseudolosowych przez rozważane algorytmy, wyniki podlegające analizie będą zagregowane z co najmniej 25 uruchomień.

Zgodnie z wymogami projektowymi, wyniki zostaną przedstawione w postaci:

- średniej,
- odchylenia standardowego,
- najlepszego oraz najgorszego wyniku.

Zbiory danych nie posiadają osobnych zbiorów testowych, zatem ewaluacja modelu dokonywana będzie z wykorzystaniem metody k-krotnej walidacji krzyżowej (k zostanie dobrane tak aby zoptymalizować czas badań i dokładność wyników). Metoda ta polega na podziale zestawu danych na k podzespołów i następnie wybieraniu kolejno jednego z nich jako zestawu danych testowych, a reszta jako zestaw danych treningowych. Ocena modelu stanowi średnią arytmetyczną wszystkich k testów.

Dokonane zostanie porównanie wyników badanego algorytmu z wynikami losowego lasu z klasyfikatorami ID3 z gotowej implementacji.

Dokonana zostanie ewaluacja modelu z modyfikacją parametrów drzew decyzyjnych:

- maksymalna głębokość,
- minimalna liczba przykładów w węźle, aby można było dokonać podziału.

Dokonana zostanie ewaluacja modelu z modyfikacją parametru procentowego udziału losowanych przykładów.

Dokonana zostanie ewaluacja modelu w zależności liczby klasyfikatorów wchodzących w skład lasu losowego.

Wszelkie eksperymenty zostaną odpowiednio udokumentowane i z każdego z badań zostaną wyciągnięte obiektywne wnioski oparte na krytycznej analizie, wraz z uzasadnieniami zaobserwowanych zależności. Natomiast cała implementacja zawarta zostanie w plikach z kodem.

Sposób ewaluacji (miary jakości)

W ramach ewaluacji rezultatów uzyskanych podczas eksperymentów dokonane zostanie wyliczenie następujących metryk:

- Dokładność (accuracy)
  Określa ona procent poprawnie sklasyfikowanych przypadków w stosunku do wszystkich przypadków w zbiorze danych.
- Tabele pomyłek (confusion matrix)
  Przedstawia liczbę poprawnych przewidywań w porównaniu z liczbą błędnych przewidywań. Dla klasyfikatora binarnego oznacza to liczbę prawdziwych negatywów i prawdziwych pozytywów (poprawne przewidywania) w porównaniu z liczbą fałszywych negatywów i fałszywych pozytywów (błędne przewidywania).
- F1 Score
  Jest to miara służąca do oceny wydajności modelu w zadaniach klasyfikacji. Uwzględnia ona zarówno precyzję jak i czułość modelu.
  Precyzja – stosunek poprawnie sklasyfikowanych pozytywnych przypadków do wszystkich pozytywnych przypadków
  Czułość – stosunek poprawnie sklasyfikowanych pozytywnych przypadków do wszystkich rzeczywiście pozytywnych przypadków.

Zbiory danych

Ogólne informacje

Implementacja algorytmu przebadana zostanie na 4 zbiorach danych. Zbiory te różnią się pomiędzy sobą charakterem, ilością rekordów, ilością kolumn, ilością klas oraz tematyką.
Dobór odpowiednich zbiorów był jedną z kluczowych wyzwań postawionych na tym etapie. Musiały one stanowić dane kategoryczne lub musiała istnieć możliwość ich przekształcenia. Spowodowane to było przede wszystkim faktem, że drzewo ID3 ma ograniczone możliwości działania, gdyż znajduje zastosowanie właśnie jedynie przy danych kategorycznych.
W projekcie zastosowania znajdą zbiory danych ze strony Kaggle.

Wytypowane zbiory danych (wraz z opisem)

1. **Glass Classification**
   Zbiór danych pochodzi z niemieckiego Central Reaserch Establishment. Badania nad klasyfikacją typów szkła zostały zmotywowane śledztwami kryminalistycznymi. Ponieważ szkło zostawione na miejscu zbrodni może być użyte jako dowód w sprawie, pod warunkiem, że zostanie poprawnie zidentyfikowane.
   Ilość instancji: 214
   Ilość atrybutów: 10.
   Atrybuty to:
   - RI - współczynnik załamania światła,
   - pierwiastki chemiczne (wchodzące w skład tlenków): Na – sód, Mg – magnez, Al – aluminium, Si – krzem, K – potas, Ca – wapń, Ba – bar, Fe – żelazo
   - Typ szkła

   (Wartości atrybutów będących pierwiastkami chemicznymi to jednostki miary wyrażane jako procent wagowy danego pierwiastka w odpowiednim tlenku – są to floaty)
   Wyróżnia się tu 6 klas (6 typów szkła)
   Brak osobnego zbioru testowego, więc zastosowana zostanie metoda k-krotnej walidacji krzyżowej.
   Link do zbioru danych: https://www.kaggle.com/datasets/uciml/glass/data

2. **Predicting divorce**
   Zbiór danych dotyczący korelacji odpowiedzi na konkretne pytania związane z relacjami w parach na możliwość rozwodu w przyszłości.
   Algorytm ma na celu wytypowanie czy para rozwiedzie się w przyszłości. Predykcja ta opiera się na zbiorze danych zawierającym pytania, na które odpowiedź to liczby w skali 0 - 4, gdzie 0 to waga najniższa, a 4 najwyższa.
   Zbiór danych zawiera 54 atrybuty - pytania. Zbiorem klas jest odpowiedź „tak” lub „nie”.
   Brak osobnego zbioru testowego, więc zastosowana zostanie metoda k-krotnej walidacji krzyżowej.
   Link do zbioru danych:
   https://www.kaggle.com/datasets/csafrit2/predicting-divorce

3. **Loan Approval Prediction Dataset**
   Zbiór danych dotyczący zatwierdzania pożyczki to zbiór parametrów wykorzystywanych do określenia zdolności finansowej osoby. Zbiór zawiera 13 atrybutów, które wpływają na to czy status pożyczki będzie miał wartość „approved" bądź „rejected", a więc zbiór zawiera 2 klasy.
   Brak osobnego zbioru testowego, więc zastosowana zostanie metoda k-krotnej walidacji krzyżowej.
   Link do zbioru danych:
   https://www.kaggle.com/datasets/architsharma01/loan-approval-prediction-dataset

4. **COVID 19 Prediction Binary Classification Problem**
   Zbiór danych służących do predykcji COVID-19 na podstawie diagnozy opartej o symptomy.
   Charakteryzuje się różnymi grupami atrybutów:
   - Informacje o osobie: płeć, wiek, data testu
   - Symptomy (wartości binarne): kaszel, gorączka, ból gardła, krótszy oddech, ból głowy)
   - Pozostałe informacje (wartości binarne): czy wiadomo o kontakcie z osobą zarażoną
   - Wynik (wartości binarne): pozytywny lub negatywny wynik na Coronę

   W zbiorze znajduje się ponad 279 tysięcy rekordów.
   Brak osobnego zbioru testowego, więc zastosowana zostanie metoda k-krotnej walidacji krzyżowej.
   Link do zbioru danych:
   https://www.kaggle.com/datasets/rohitudageri/covid-19-prediction-binary-classification-problem/data

Uwagi końcowe

W uzasadnionych przypadkach badacze zastrzegają sobie prawo do modyfikacji założeń i ustaleń poczynionych w niniejszym dokumencie, stanowiącym wstępną dokumentację. Jeżeli do takowego odstępstwa dojdzie zostanie to odnotowane i uzasadnione w końcowym sprawozdaniu z przeprowadzonych badań.