Uczenie maszynowe

Marcin Jarczewski
Mikolaj Szawerda

7 grudnia 2023

Spis treści

Streszczenie

Projekt polega na zaimplementowaniu inkrementacyjnej indukcji reguł. Zbiór reguł ma ulegać przebudowie na podstawie sekwencyjnie nadchodzących porcji danych lub pojedynczych przykładów.

1 Opis projektu

1.1 Indukcja reguł

Zadanie indukcji reguł polega na wyznaczeniu zbioru reguł. Pojedyczna reguła składa się z części warunkowej - kompleksu determinującego pokrywanie danego przykładu i części decyzyjnej - przyporządkującej klasę na podstawie rezultatu pokrywania. Reguł, które w części decyzyjnej mają taką samą klasę są połączone alternatywą. Cechą wyznaczonych reguł powinna być maksymalna ogólność - w pokrywanych przykładach przez kompleks powinna dominować jedna klasa.

1.2 Przebudowa zbioru reguł

Zadaniem przebudowy reguł jest dynamiczne generowanie i usuwanie już wygenerowanych reguł w zależności od sekwencyjnie pojawiających się danych. Celem przebudowy jest wyindukowanie nowej wiedzy, utwierdzenie już stworzonej wiedzy oraz adekwatna zmiana aktualnej wiedzy. Badane algorytmy powinny więc pamiętać niepokryte/źle sklasifikowane przykłady i w przypadku pojawienia się statystcznie znaczącej próbki dokonać odpowiednich akcji na zbiorze reguł.

1.3 Zasady przebudowy

Dodatkowo przebudowywanie ustalonego zbioru reguł, będzie odbywać się w określony sposób:

1. Nowy przykład jest sklasyfikowany poprawnie, wtedy nic nie robimy
2. Nowy przykład nie jest pokryty, wtedy tworzymy nową regułę, bazując na wszystkich niepokrytych przykładach.
3. Nowy przykład jest błędnie sklasyfikowany, wówczas mamy dwie możliwości:
   - jeśli są to przypadki pojedyncze, tj ich liczba nie przekracza $X\%$ wszystkich przypadków, to ignorujemy je
   - w przeciwnym przypadku, usuwamy wszystkie reguły, które błędnie zaklasyfikowały dane przypadki i przebudowujemy zbiór reguł dla nowo niepokrytych danych.

2 Opis algorytmów

Wykorzystamy dwa algorytmy implementujące podejście sekwencyjnego pokrywania, które były podane na wykładzie. W każdym z nich dodatkowo przetestujemy metody rozstrzygania reguł pokrywających ten sam przykład.

2.1 CN2

Algorytm dla zadanego zbioru trenującego generuje reguły, na zasadzie specjalizacji kompleksów. Reguła zostaje utworzona poprzez iteracyjne generowanie kompleksów z aktualnie utworzonych(poczynając od najbardziej ogólnej), konkretyzując po każdych możliwych wartościach selektorów. Po każdej iteracji wybierane jest N najlepszych kompleksów. Szeregowanie reguł odbywa się na podstawie wartości entropii $-\sum_i p_i log_2(p_i)$ i faktu czy reguła jest statystycznie znacząca - na podstawie testu χ^2

```
%D - zbior trenujacy
%S - mozliwe wartosci selektorow
cn2(D):
    RULES = []
    while BEST_CPX is null or D is empty:
        BEST_CPX = find_best_complex(D)
        if BEST_CPX is not null:
            D' = examples from D covered by BEST_CPX
            D = D \ D'
            CLASS = most common class in D'
            RULES += BEST_CPX
    ret RULES
find_best_complex(D):
    STAR = ?
    BEST_CPX = null
    while STAR is not empty:
        NEW_STAR = {(x and y) where x in STAR, y in S} - all possible specializations of
        NEW_STAR = NEW_STAR \ STAR
        for COMPLEX in NEW_STAR:
            if COMPLEX statistically significant AND COMPLEX > BEST_CPX:
                BEST_CPX = COMPLEX
        while len(NEW_STAR) > MAX_CPX_LEN:
            NEW_STAR = NEW_STAR \ wors_from(NEW_STAR)
        STAR = NEW_STAR
    return BEST_CPX
```

2.2 AQ

Algorytm opiera się na generowaniu kompleksów, które pokrywają losowo wybrany przykład pozytywny i nie pokrywają przykładów negatywnych. Reguła utworzona zostaje z kompleksu, który pokrywa największą ilość przykładów pozytywnych i najmniejszą ilość przykładów negatywnych.

```
%P - przykłady pozytywne
%N - przykłady negatywne

aq(P, N)
    COVER = ?
    while COVER != P % dopoki cover nie pokrywa wszystkich pozytywnych
        SEED = x where x in P and x not in COVER
        STAR = star(SEED, N) % zbior kompleksow, ktore pokrywaja SEED, ale nie N
        BEST = max(STAR)
        COVER += BEST
    ret COVER
star(SEED, NEG)
    star = null
    while NEG*star != null
        neg = x where x in NEG and x in star
        star -> modify complex such that SEED in star and neg not in star
        star -> leave only most general complexes
        while len(star) < MAX_CPX
            star = star \ worst_from(star)
    return star
```

3 Zbiory danych

W ramach projektu, będziemy korzystać z dwóch zbiorów:

3.1 Bank Marketing

3.1.1 Opis atrybutów

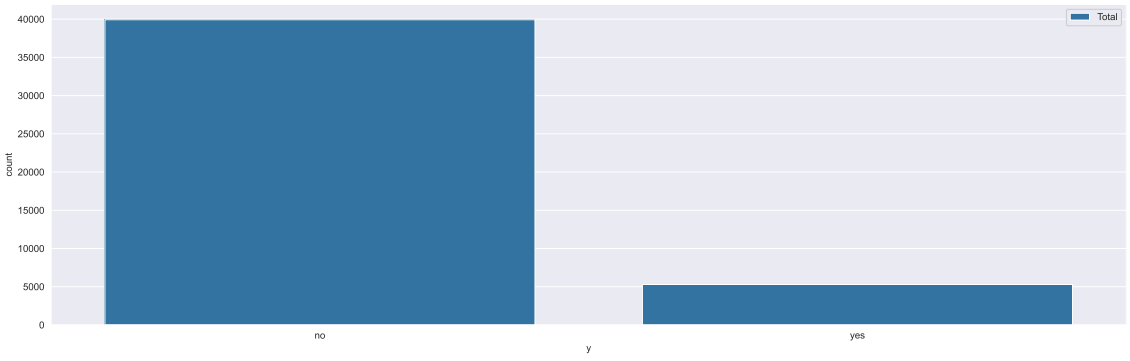
Opisujący dane klientów portugalskiego banku w trakcie prowadzenia telefonicznej kampanii reklamowej, której celem było zachęcenie klientów do skorzystania z lokaty. Zadaniem klasyfikacji w zbiorze jest przewidzenie na podstawie cech, czy klient weźmie lokatę. Zbiór składa się z cech:

- age (liczba)
- job : rodzaj zatrudnienia: "admin.","unknown","unemployed","management","housemaid","entrepreneur","student" "blue-collar","self-employed","retired","technician","services")
- marital : stan cywilny (kategorie: "married","divorced","single"; note: "divorced"means divorced or widowed)
- education (kategorie: "unknown","secondary","primary","tertiary")
- default: czy posiada kredyt? (binanry: "yes","no")
- balance: średni roczny stan konta, w euro (liczba)
- housing: czy posiada nieruchomość? (binarny: "yes","no")
- loan: czy ma własną lokatę? (binarny: "yes","no")
- contact: rodzaj kontaktu (kategorie: "unknown","telephone","cellular")
- day: dzień miesiąca, ostatniego kontaktu (liczba)

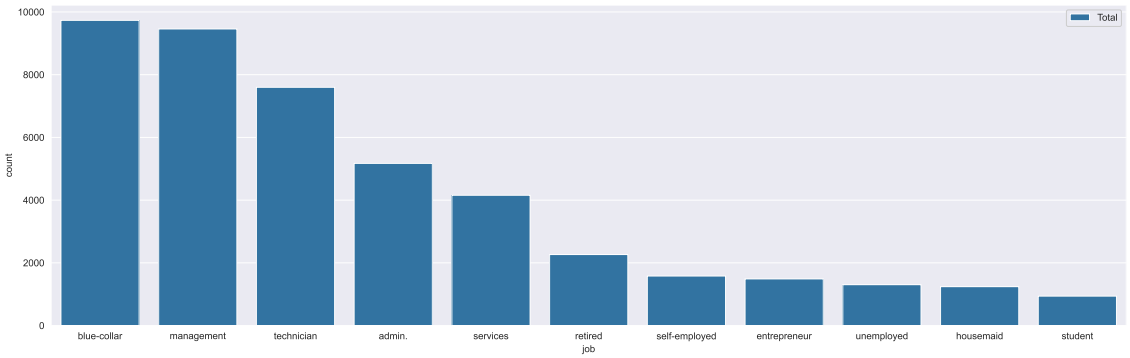
- month: miesiąc, ostatniego kontaktu (kategorie: ”jan”, ”feb”, ”mar”, ..., ”nov”, ”dec”)
- duration: długość trwania rozmowy, w sekundach (liczba)
- campaign: liczba kontaktów przeprowadzonych w ramach ostatniej kampani reklamowej (liczba, zawiera ostatni kontakt)
- pdays: liczba dni, które minęła od poprzedniej kampani (liczba, -1 oznacza brak wcześniejszego kontaktu)
- previous: liczba kontaktów przeprowadzonych przed tą kampanią dla danego klienta
- poutcome: wynik kampanii (kategorie: ”unknown”,”other”,”failure”,”success”)

3.1.2 Analiza danych

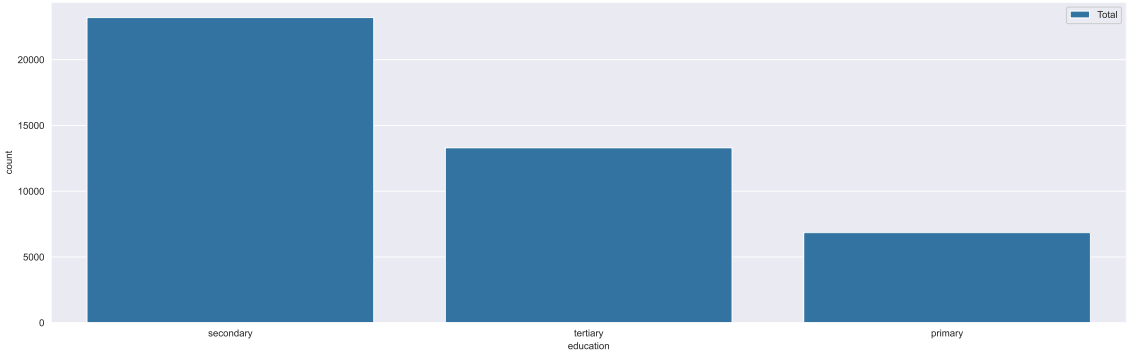
Atrybut	Liczba brakujących wartości
age	0
job	288
marital	0
education	1857
default	0
balance	0
housing	0
loan	0
contact	13020
day	0
month	0
duration	0
campaign	0
pdays	0
previous	0
poutcome	36959
y	0



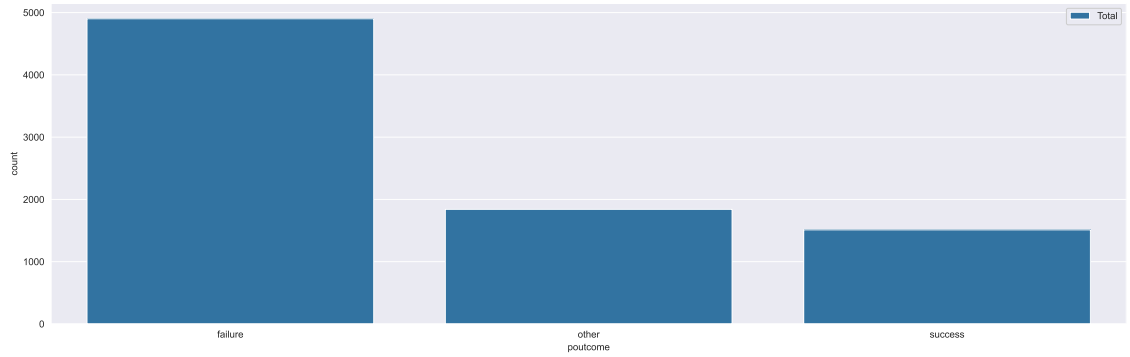
Rysunek 1: Rozkład klasy



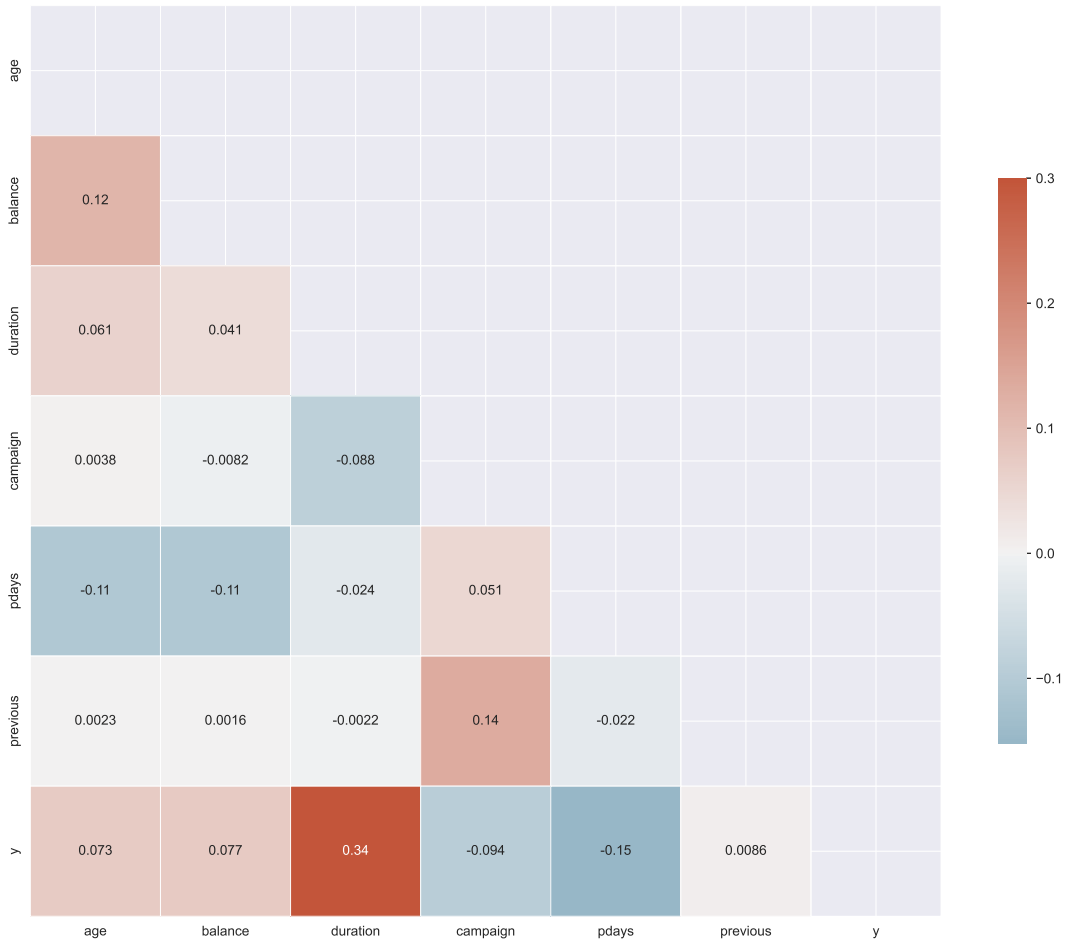
Rysunek 2: Rozkład wartości atrybutu *job*



Rysunek 3: Rozkład wartości atrybutu *education*



Rysunek 4: Rozkład wartości atrybutu *poutcome*



Rysunek 5: Korelacje między atrybutami ciągłymi

3.2 Adult

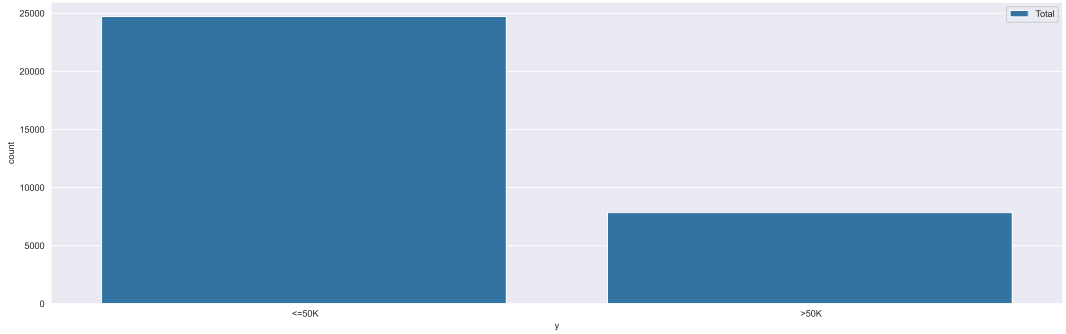
3.2.1 Opis atrybutów

Zbiór danych zawierający zestaw cech osób dorosłych w celu przewidzenia czy dana osoba zarabia więcej czy mniej niż 50 tys. dolarów rocznie. W skład cech wchodzą:

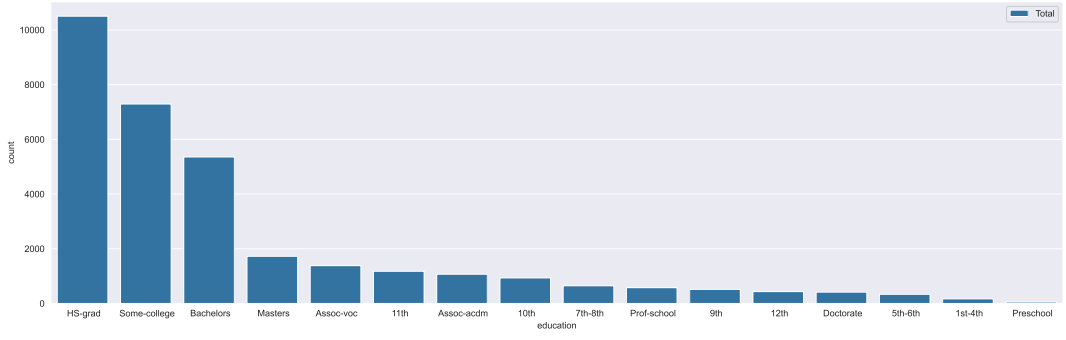
- age: atrybut ciągły.
- workclass: stan zatrudnienia - kategorie: Private, Self-emp-not-inc, Self-emp-inc, Federal-gov, Local-gov, State-gov, Without-pay, Never-worked.
- fnlwgt: atrybut ciągły - wartość wyznaczona na podstawie charakterystyki demograficznej(osoby z podobną charakterystyką posiadają zbliżone wartości tej cechy)
- education: wykształcenie - kategorie: Bachelors, Some-college, 11th, HS-grad, Prof-school, Assoc-acdm, Assoc-voc, 9th, 7th-8th, 12th, Masters, 1st-4th, 10th, Doctorate, 5th-6th, Preschool.
- education-num: atrybut ciągły - liczba lat edukacji
- marital-status: stan cywilny - kategorie: Married-civ-spouse, Divorced, Never-married, Separated, Widowed, Married-spouse-absent, Married-AF-spouse.
- occupation: zawód - kategorie: Tech-support, Craft-repair, Other-service, Sales, Exec-managerial, Prof-specialty, Handlers-cleaners, Machine-op-inspct, Adm-clerical, Farming-fishing, Transport-moving, Priv-house-serv, Protective-serv, Armed-Forces.
- relationship: relacja - kategorie: Wife, Own-child, Husband, Not-in-family, Other-relative, Unmarried.
- race: rasa - kategorie: White, Asian-Pac-Islander, Amer-Indian-Eskimo, Other, Black.
- sex: płeć - kategorie: Female, Male.
- capital-gain: atrybut ciągły.
- capital-loss: atrybut ciągły.
- hours-per-week: atrybut ciągły. - tygodniowa liczba godzin pracy
- native-country: kraj pochodzenia - kategorie: United-States, Cambodia, England, Puerto-Rico, Canada, Germany, Outlying-US(Guam-USVI-etc), India, Japan, Greece, South, China, Cuba, Iran, Honduras, Philippines, Italy, Poland, Jamaica, Vietnam, Mexico, Portugal, Ireland, France, Dominican-Republic, Laos, Ecuador, Taiwan, Haiti, Columbia, Hungary, Guatemala, Nicaragua, Scotland, Thailand, Yugoslavia, El-Salvador, TrinadadTobago, Peru, Hong, Holand-Netherlands.

3.2.2 Analiza danych

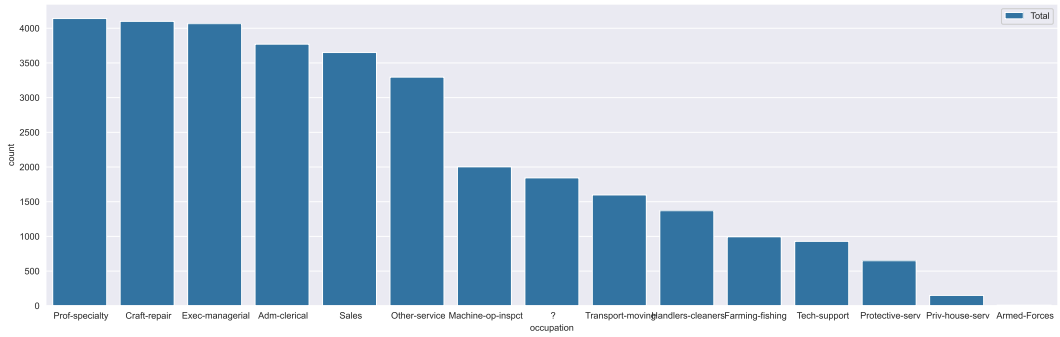
Atrybut	Liczba brakujących wartości
age	0
workclass	1836
fnlwgt	0
education	0
education-num	0
marital-status	0
occupation	1843
relationship	0
race	0
sex	0
capital-gain	0
capital-loss	0
hours-per-week	0
native-country	583
y	0



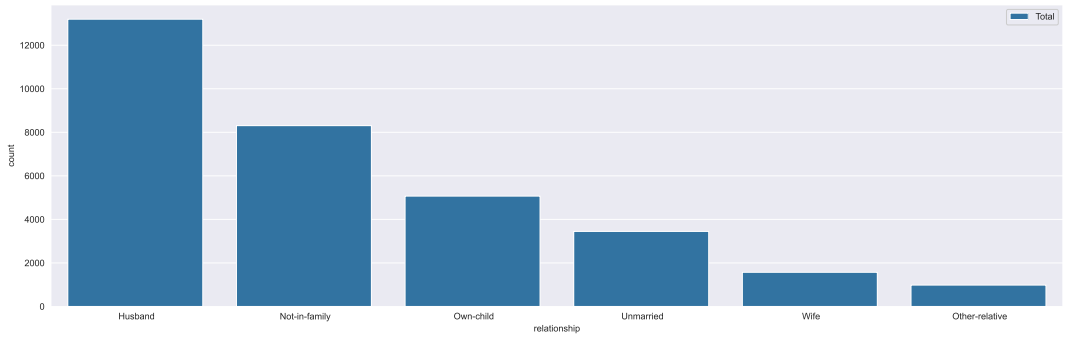
Rysunek 6: Rozkład klasy



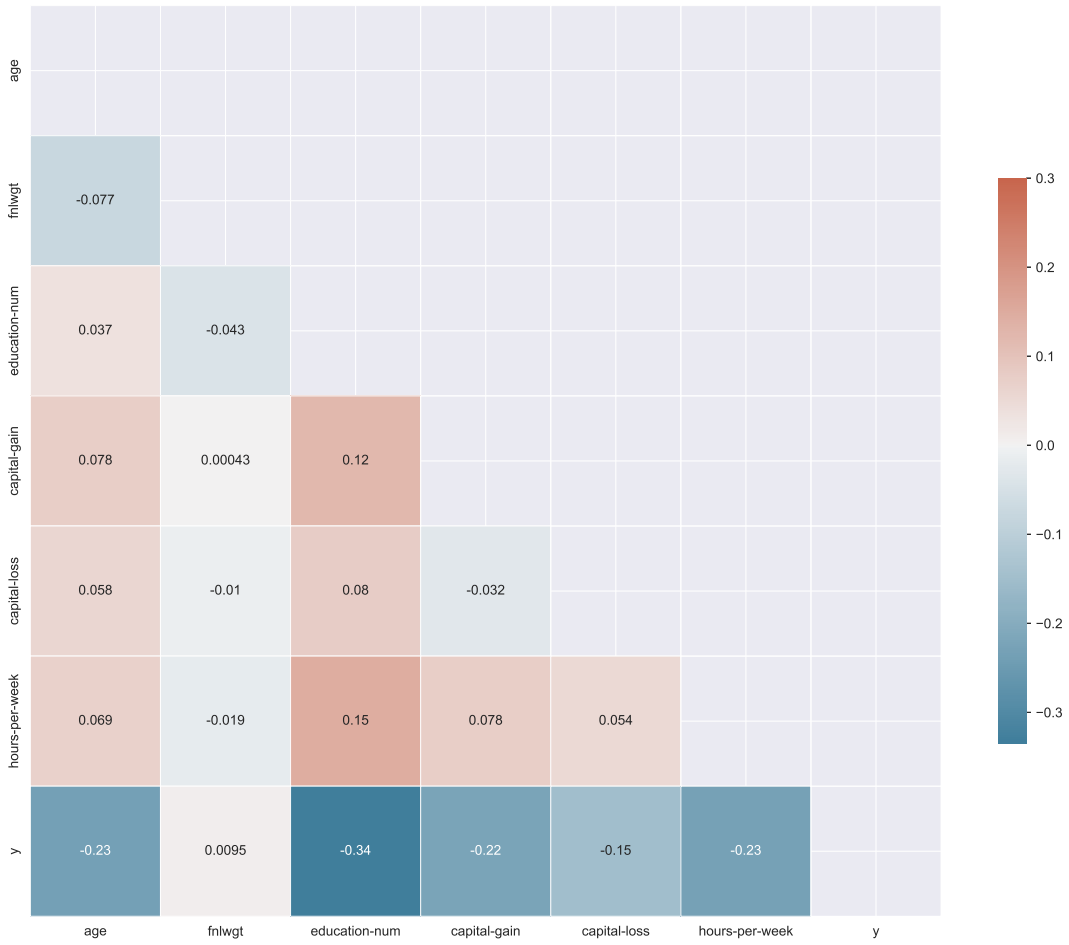
Rysunek 7: Rozkład wartości atrybutu *education*



Rysunek 8: Rozkład wartości atrybutu *occupation*



Rysunek 9: Rozkład wartości atrybutu *relationship*



Rysunek 10: Korelacje między atrybutami ciągłymi

4 Plan eksperymentów

Zgodnie z celem projektu, zadaniem eksperymentów będzie zbadanie poprawności implementacji dynamicznej przebudowy reguł, a także porównanie dwóch algorytmów w środowisku dwóch różnych zbiorów danych zadania klasyfikacji binarnej oraz wybranych sposobów rozstrzygania ostatecznej odpowiedzi modelu z ustalonego zbioru reguł.

W celu porównania algorytmów i możliwych do zmiany części algorytmu podzielimy nasz zbiór

na zbiory: uczący, testowy i walidacyjny. W ramach zbioru testowego chcemy zaobserwować zmianę powstających reguł poprzez zliczanie ile razy reguły zostały przebudowywane. Odbędzie się to poprzez porównanie statystyk klasyfikacji przykładów do reguł przed i po przebudowie. Dodatkowo będziemy mogli porównać ile procent przykładów jest dobrze sklasyfikowanych.

W celu zapewnienia zaobserwowania przebudowy reguł wykonamy osobny eksperyment, w którym modelelowi będą podawane sekwencyjnie dane loswej wielkości z góry ustalonym rozkładem - który będzie miał intuicyjną interpretację zmiennej przewidywanej. Następnie po wyczerpaniu przez model pierwszych porcji danych, podane zostaną kolejne porcje o przeciwstawnej interpretacji w ilości teoretycznie zmuszającej model do całkowitej przebudowy reguł.

Tak jak było wspomniane w punkcie 1.3 planujemy przetestować różne wartości parametru X, który oznacza procent błędnie sklasyfikowanych przypadków testowych. Będzie ona wynosić od 5% do 15%, jednak dokładne wartości zostaną ustalone w trakcie implementacji.