Lab3-Analizator_wyników

Oskar Kamiński s25020 gr. Ćw. 12c

1) Eksploracja i wstępna analiza danych.

Po wczytaniu danych można zauważyć następujące rzeczy:

1. Dane dzielą się na 15 kolumn tj.:
    a. Kolumny numeryczne:
        i. Rownames – index kolumny w datasecie.
        ii. Score – liczba zdobytych punktów (wartość przewidywana).
        iii. Unemp – wartość bezrobocia
        iv. Wage – płaca godzinowa
        v. Distance – dystans między miejscem zamieszkania a uczelnią liczonych w 10 milach
        vi. Tuition – średni koszt czesnego w USD
        vii. Education – ilość lat spędzonych podczas nauki
    b. Kolumny kategoryczne:
        i. Gender - płeć
        ii. Ethnicity – pochodzenie. Wyróżnia się 3 wartości: „afam” (African-American), „hispanic”, „other”
        iii. Fcollege – zmienna sprawdzająca czy matka ucznia zdała college
        iv. Mcollege – zmienna sprawdzająca czy ojciec ucznia zdał college
        v. Home – zmienna zawierająca informacje czy rodzina posiada dom.
        vi. Urban – zmienna sprawdzająca czy szkoła znajduję się w terenie miejskim.
        vii. Income – zarobki rodziny. Przyjmuje wartości „low” jeśli zarobki są mniejsze niż 25 000 USD rocznie. W przeciwnym wypadku przybiera wartość „high”
        viii. Region – zmienna sprawdzająca rejon. Przyjmuje wartość „West” lub „other”
2. Dataset posiada 4739 rekordów z czego wszystkie rekordy są uzupełnione.

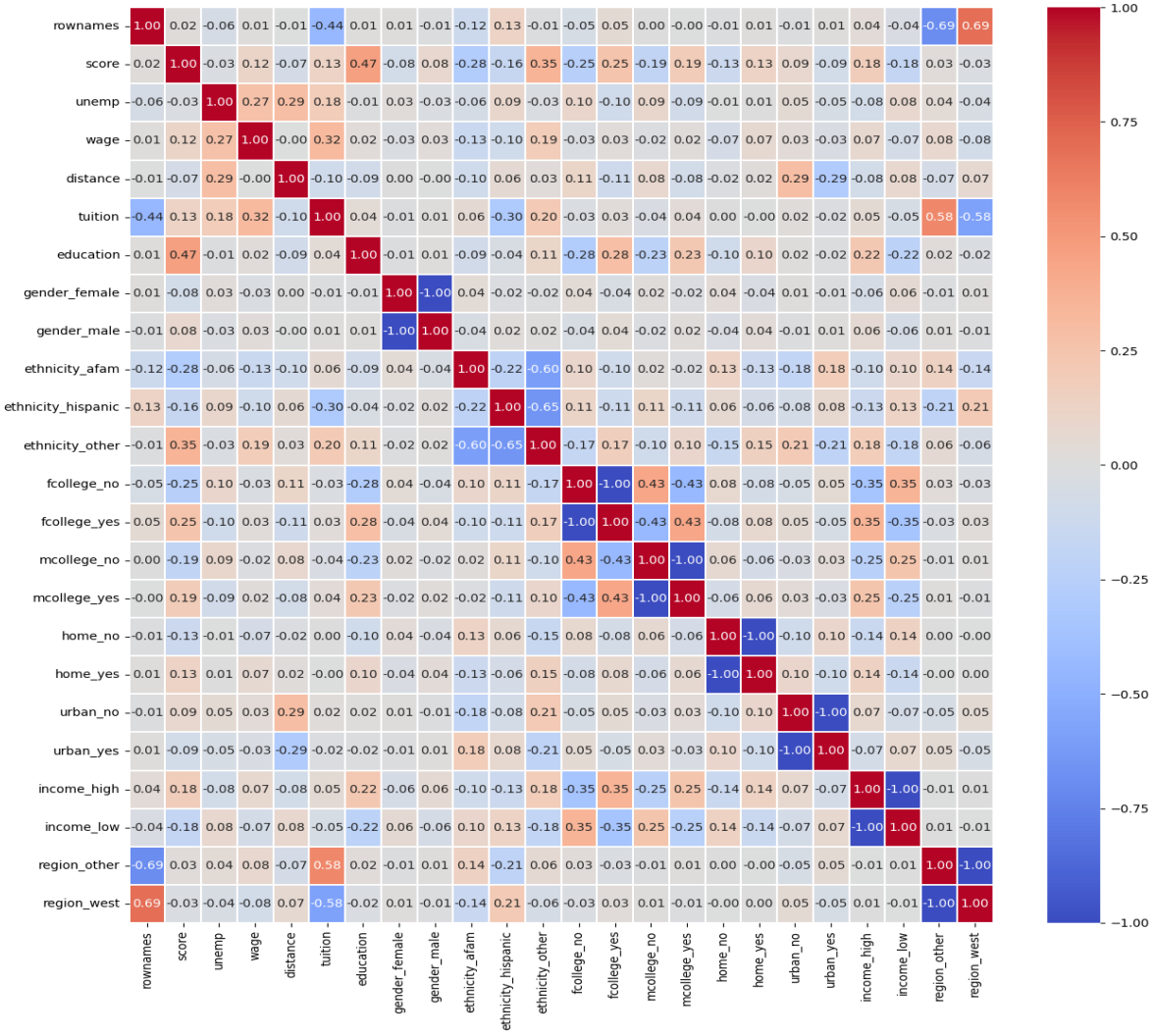
Jednakże w celu zabezpieczenia się przed niechcianymi problemami tj. brak wartości w niektórych kolumnach przygotowałem następujące rozwiązanie:

1. Usunięcie rekordu w przypadku gdy ma on za mało uzupełnionych danych. (Usunięcie kolumn które mają mniej niż 5 uzupełnionych kolumn).
2. Następnie imputacja brakujących wartości w następujący sposób:

a. Brakujące dane numeryczne zostaną uzupełnione średnią wartością.
b. Brakujące dane kategoryczne zostaną uzupełnione najczęściej występującą wartością.

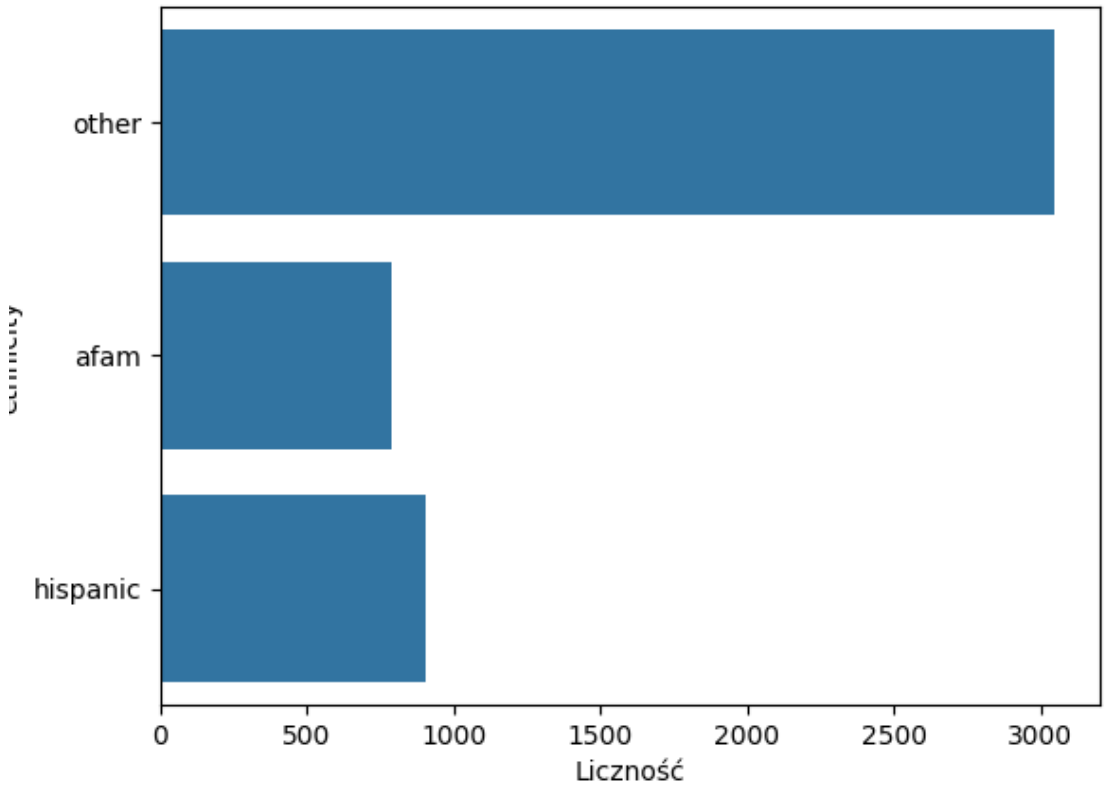
Poniżej przestawiam następujące wykresy

a. Wykres korelacji macierzy

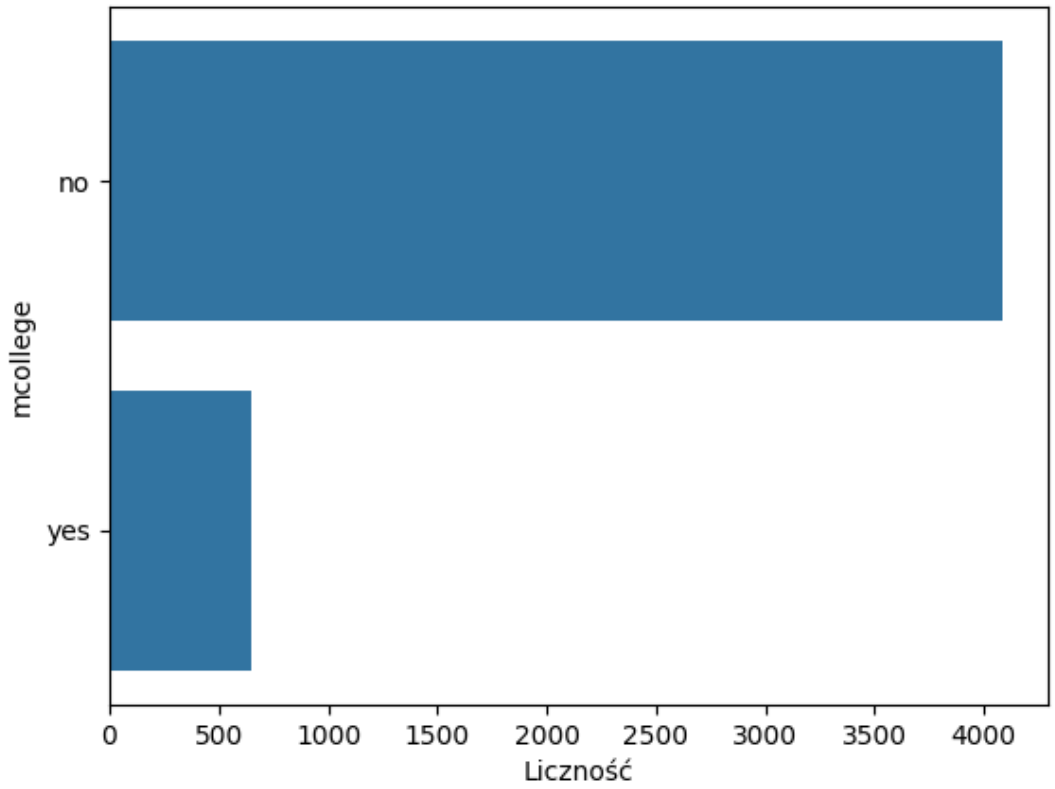


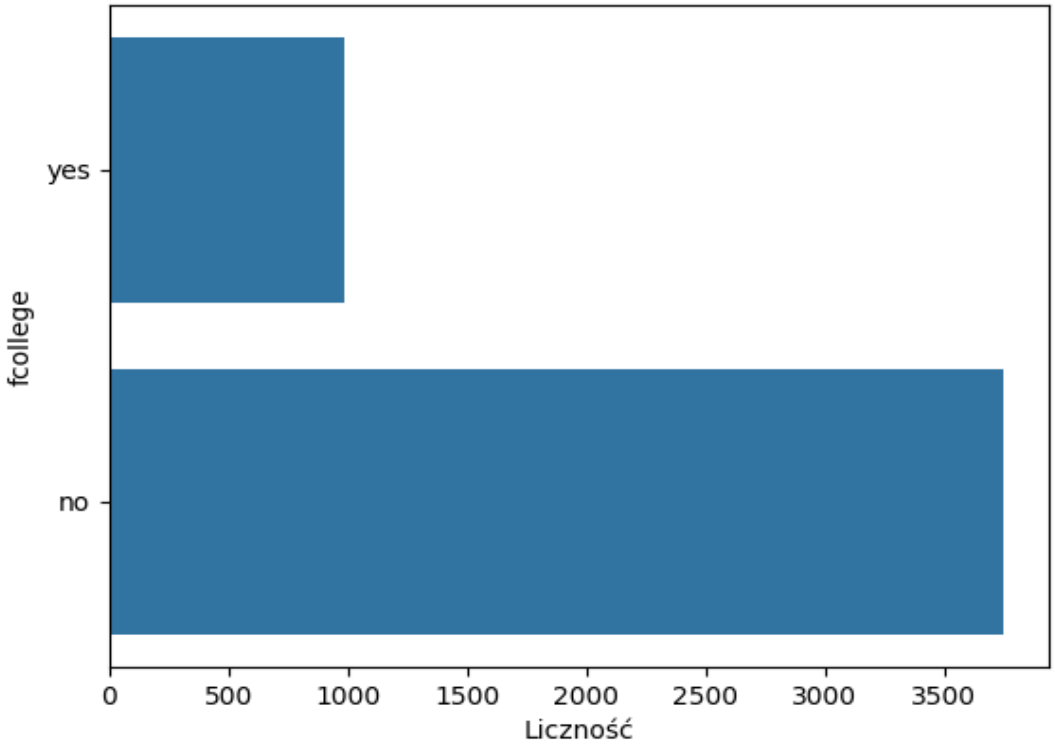
Z wykresu można wywnioskować, że największą korelacje z „score” mają: education (0.47) oraz ethnicity_other (0.35). Praktyczny brak powiązań z „score” mają następujące kolumny: rownames(0.02), unemp(-0.03), distance(-0.07), region_west(-0.03), urban_yes(-0.09), ethnicity_hispanic(-0.16), income_low(-0.18), fcollege_no(-0.25), ethnicity_afam(-0.28). W wielu z powyżej wymienionych przypadków wynika z dominacji w swoich kategoriach.

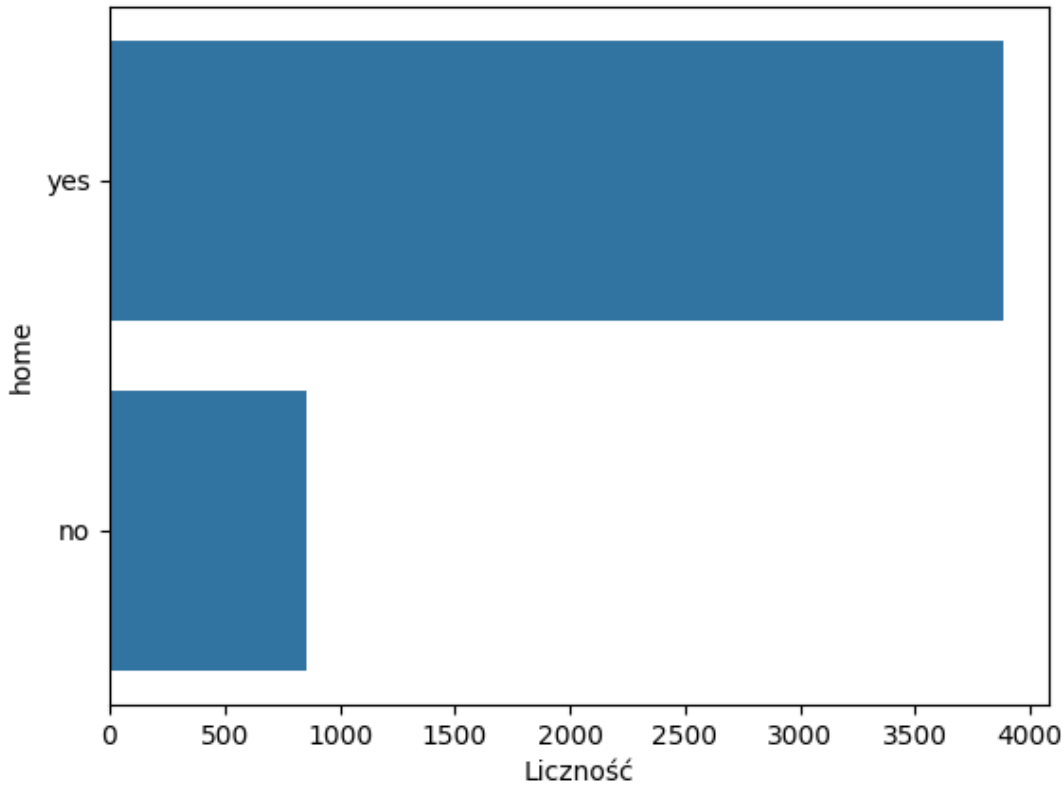
Takim przykładem jest chociażby ethnicity_other który dominuje na poniższym załączonym wykresie.

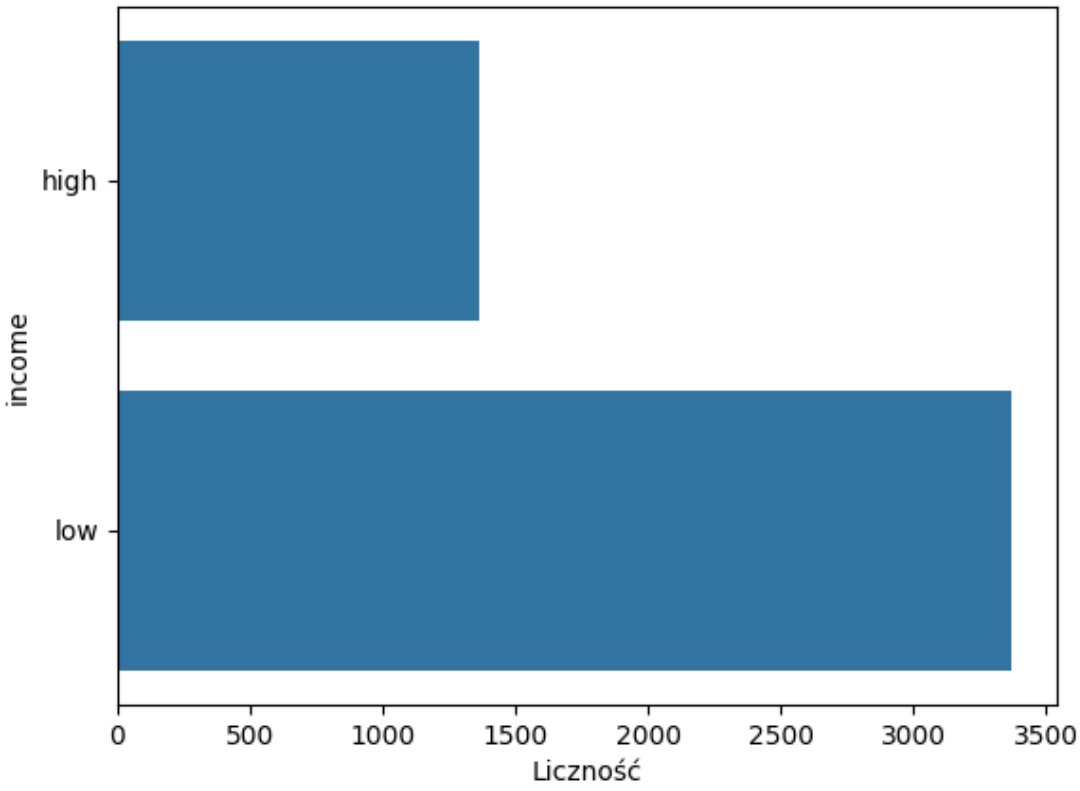


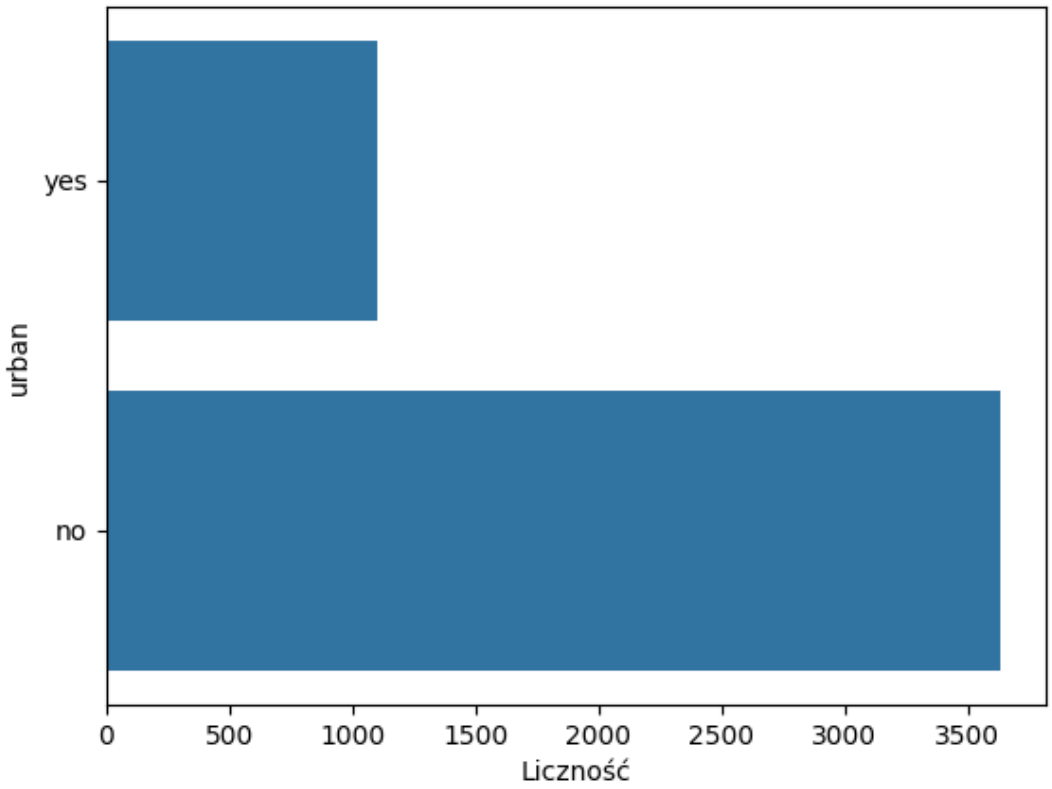
Poniżej załączam również resztę wykresów kategorycznych. Na większości z nich widać znaczącą dominację jednej kategorii (wyjątkiem jest kolumna „gender”)

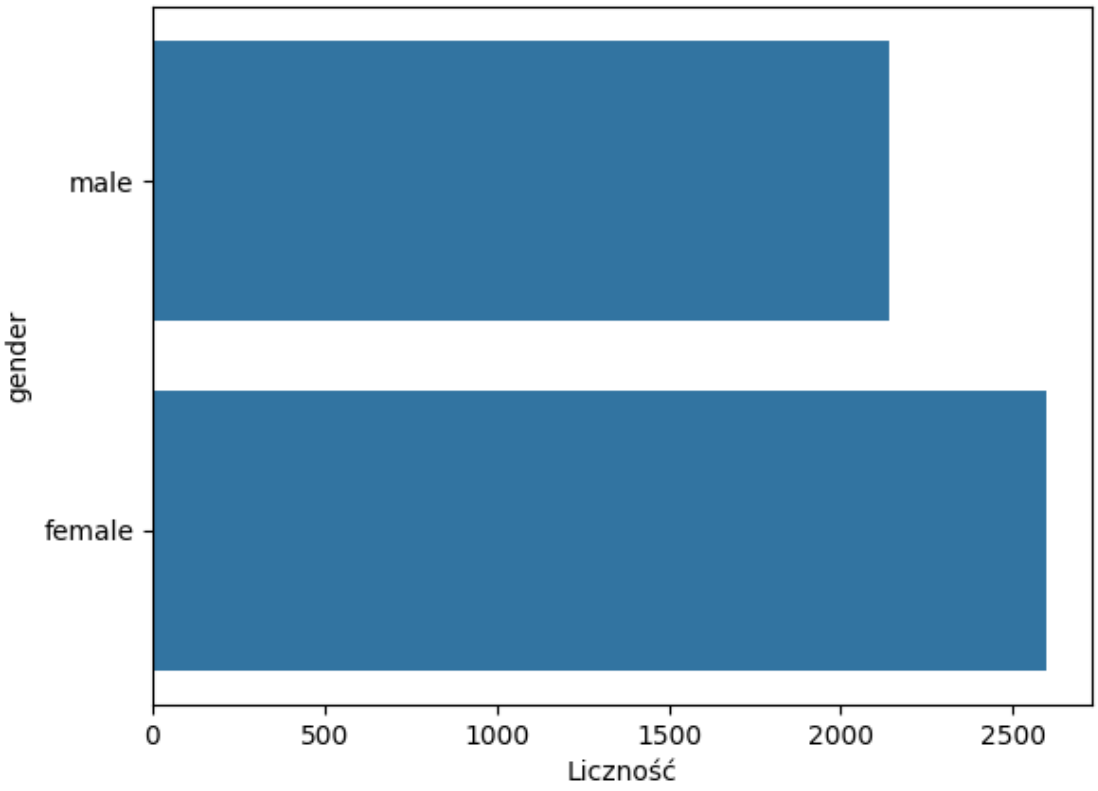

mcollege
no
yes
0
500
1000
1500
2000
2500
3000
3500
4000
Liczność


fcollege
yes
no
0
500
1000
1500
2000
2500
3000
3500
Liczność

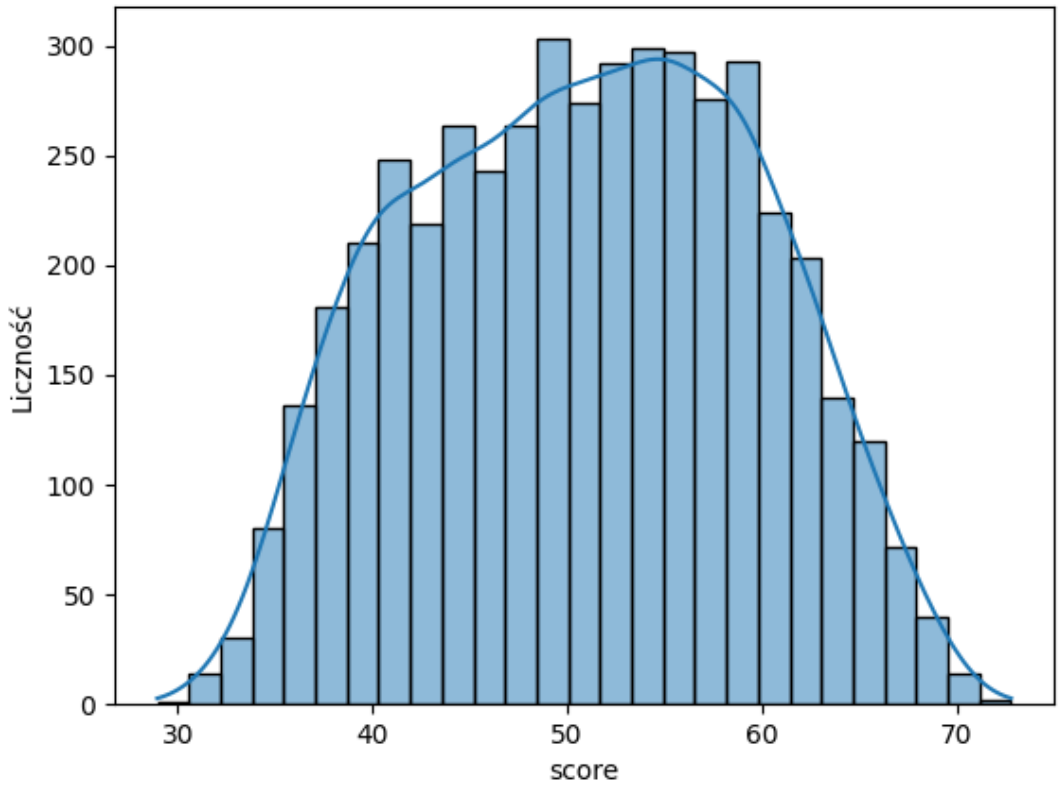

yes
no
home
0
500
1000
1500
2000
2500
3000
3500
4000
Liczność


high
low
income
0
500
1000
1500
2000
2500
3000
3500
Liczność


yes
no
urban
0
500
1000
1500
2000
2500
3000
3500
Liczność


male
female
gender
0
500
1000
1500
2000
2500
Liczność

b. Wykres rozkładu dla zmiennej „score”



Z powyższego wykresu wynika, że najwięcej zdobytych punktów mieści się w przedziale od 40 do 60.

2) Inżynieria cech i przygotowanie danych

W celu przygotowania danych, podjęte zostały następujące czynności:

1) Usunięcie kolumny „rownames” ze względu iż posiada same unikalne rekordy. Dodatkowo negatywnie wpływa na wyszukiwane dane.

2) Podzielenie na kolumny numeryczne i kategoryczne.

3) Zdekodowanie danych kategorycznych poprzez użycie OneHotEncoder na wartości liczbowe.

4) Standaryzacja danych numerycznych (z wyłączeniem kolumn „score”) przez przekształcenie ich na rozkład o średniej 0 i odchylenie standardowe 1.

Po tych krokach, uzyskany zbiór został podzielony na zbiór treningowy i testowy w podziale 80/20.

3) Wybór i trenowanie modelu

Zdecydowałem się na wybór algorytmu drzew losowych z następujących powodów:

- Jest w stanie wykrywać zależności zarówno liniowe jak i nieliniowe.
- Jest odporny na przeuczenie.
- Umożliwia tuningowanie hiperparametrów z pomocą GridSearchCV w celu optymalizacji modelu.
- Nie ma problemu z radzeniem sobie z danymi kategorycznymi
- Jest w stanie radzić sobie z cechami odstającymi.
- Ma dobrą skalowalność i jest w stanie równolegle przetwarzać drzewa co powoduje przyspieszenie treningu.

4) Ocena i optymalizacji modelu

Poniżej przedstawiam wyniki R2, MSE i MAE dla modelu drzew losowych:

```
R2: 0.2673747738665674
MSE: 0.7185683202578315
MAE: 0.6782519655882582
```

Powyższe wyniki są niesatysfakcjonujące. Pokazują znaczące ograniczenie przewidywań zmiennej „score”. W tym celu postanowiłem zoptymalizować model drzew losowych poprzez tuning hiperparametrów oraz użycia walidacji krzyżowej (cv = 5).

W trakcie optymalizacji najlepsze hiperparametry to:
max_depth = 5

Max_features = None

N_estimators = 50 lub 200 (częsta zmiana hiperparameterów)

Po przećwiczeniu modelu uzyskałem następujące wyniki:

```
R2: 0.3321170873346857
MSE: 0.6550682198259652
MAE: 0.6585745252741696
```

Wniosek: Po optymalizacji modelu drzew losowych, jakość modelu wzrosła względem poprzedniej iteracji drzew losowych". Można zauważyć znaczący wzrost wartości R2 aż o 7%. Mimo to model nadal ma słabą zdolność do przewidywania zmiennej „score". Ma też nie za dużą wartości błędów MSE i MAE.