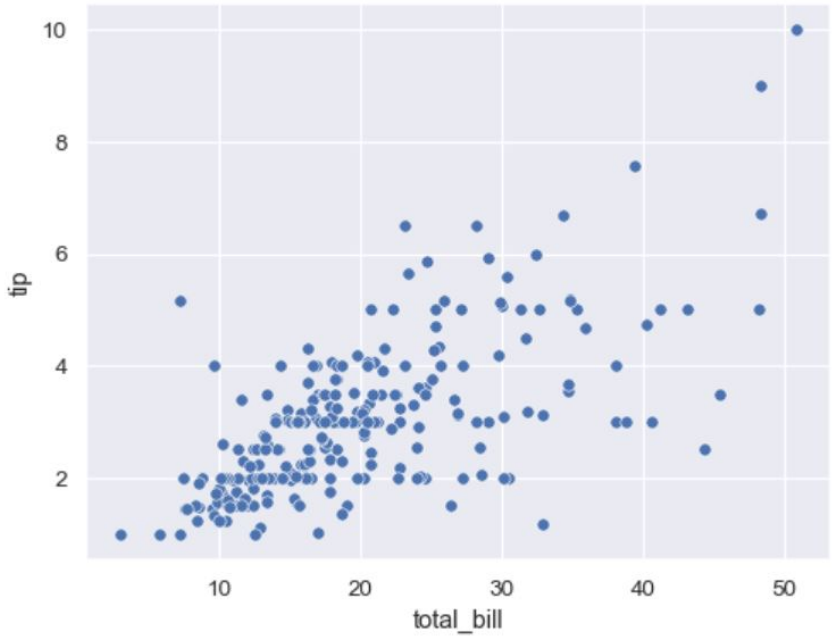
Spis treści

1. Zapoznanie się z biblioteką seaborn

Biblioteka Seaborn jest biblioteką języka Python, bardzo podobna do matplotlib, i na niej też zbudowana. Cechuje się ekstremalną prostotą w użyciu. Zawiere wiele wbudowanych typaów wykresów oraz predefiniowane style, które nie wymagają ręcznego dostosowywania. Ze względu, na jej łatwość użycia i ekstensywne możliwości w zastosowaniu, jest częstym wyborem analityków danych, w wizulizowaniu zagmatwanych zbiorów danych.

1.1. Przykład



2. Wprowadzenie do strony dane.gov

Strona https://dane.gov.pl/pl jest oficjalną platformą udostępniającą dane publiczne w Polsce. Strona udostępnia ogrom danych, z wielu obszarów na przykład z dziedzin: Energetyka, Transport, Nauka i Technologia, Zdrowie etc. Dane udostępniane są głównie przez instytucje publiczne, lecz zdarzają się zestawy danych udostępnione przez firmy prywatne.

2.1. Wykorzystanie

Dane można wykorzystać, w celach komercyjnych lub edukacyjnych w zależności od przypisanej licencji użytkowania. Zdecydowana większość, zezwala na dowolne wykorzystanie danych, ponad połowa całkowicie zrzeka się praw autorskich. Pozostałe wymagają co najwyżej 'Uznania autorstwa', czyli utwór trzeba odpowiednio oznaczyć i podać link do licencji.

2.2. Api

Korzystanie z Api jest bardzo skomplikowane i zazwyczaj zbędne. W łatwy sposób można dane pobrać wprost ze strony, klikając w link. Wykorzystanie Api, jest zależne od dostawcy danych i nie zawsze możliwe. Posiada również ograniczenia w postaci limitów zapytań i opatrzone jest wyjątkowo skomplikowaną dokumentacją, która szczególnie zniechęca do korzystania z takiego rozwiązania.

2.3. Jakość danych

Możliwe jest zweryfikowanie jakości danych, ponieważ każdy zasób ma zaznaczony poziom otwartości danych, co według serwisu oznacza: "...że dane są lepiej przygotowane do dalszego przetwarzania"", co przekłada się na ich jakość. Serwis również oferuje podglądy danych w postaci widoku tabelarczynego, wykresów oraz map, pozwala to na własną ocenę jakości danych.

3. Wybór danych

Do tego projektu został wybrany Zbiorczy-raport-dzienny-2024-05-01-0400csv.csv Jest to zbiór danych, z najważniejszymi informacjami na temat zasobów udostępnianych w serwisie dane.gov. Został wybrany ze względu na relatywnie duży rozmiar i przejrzystość, w porównaniu do innych zasobów, które można odnaleźć w tym serwisie. Został pobrany wprost ze strony w formacie csv.

4. Pierwszy etap pipeline'u ML

Pierwszym etapem, każdego pipeline'u Machine Learning jest EDA (Exploratory Data Analysis) oraz FE (Feature Engineering). EDA polega na zrozumieniu, pobranego zbioru danych. Badanie wymiarów (ilość kolumn i wierszy) zasobu, typów danych, brakujących wartości. Przede wszystkim polega na zapoznaniu się ze znaczeniem poszczególnych wartości, zrozumieniu schematów ukrytych w danych oraz wyszukiwaniu outlierów. FE to proces wybierania, manipulowania i przekształcania surowych danych w cechy, które można wykorzystać w uczeniu nadzorowanym.

4.1. Pobranie danych i zapoznanie się z ich opisem

Na rysunku 4.1 widać wczytane nieprzetworzone dane. Oryginalnie ramka danych składała się z 29 kolumn i 37679 wierszy.

```
df = pd.read_csv('Zbiorczy_raport_dzienny_2024_05_01_0400csv.csv')
print('Wymiary ramki danych:',df.shape)
df.head(5)
```

```
Wymiary ramki danych: (37679, 29)
```

	Id zasobu	Link zasobu	Nazwa	Typ	Format	Formaty po konwersji	Data utworzenia zasobu	Data modyfikacji zasobu	Stopien otwartosci	Zasob posiada dane wysokiej wartosci	...	Link zbioru	Data utworzenia zbioru danych	Data modyfikacji zbioru danych	Liczba obserwujacych	Id instytucji
0	9.0	https://dane.gov.pl/pl/dataset/590/resource/9	Diagnoza dla Programu Polska Cyfrowa 2014-2020	plik	pdf	NaN	2018-07-27 09:04:51.406852+00:00	2019-08-28 09:30:21.627283+00:00	1.0	Nie sprecyzowano	...	https://dane.gov.pl/pl/dataset/590	2018-07-27 07:03:15.690783+00:00	2024-03-19 19:15:18.454848+00:00	0	131

Rysunek 4.1: *Pobranie i wyświetlenie danych*

Opisy danych podzielono na 3 podzbiory: Zasobów, Zbiorów danych i Instytucji, w celu uzyskania większej przejrzystości w dalszej analizie.

Opis danych zasobów:

- **Id zasobu**: Unikalne id, w kontekście ML bezużyteczne.
- **Link zasobu**: Link, również bezużyteczny w ML. Przydatny jako dodatkowa informacja.
- **Nazwa**: Jeśli nie będą się powtarzać, może zostać uznana jako ID, informacja przydatna dla nas, dla modelu ML nie bardzo.
- **Typ**: Może być przydatny do modelu, pod warunkiem, że będzie zawierał inne wartości niż 'plik'.

- **Format**: Podobnie do 'Typ', może być przydatne.
- **Formaty po konwersji**: Nie jestem pewien, czym jest 'konwersja'.
- **Data utworzenia zasobu**: Może być przydatna, jeśli sposób nadawania klasy 'Stopień otwartości' zmieniał się w czasie.
- **Data modyfikacji zasobu**: Podobnie z 'Data utworzenia zasobu', modyfikacja zasobu mogła wpłynąć na TARGET.
- **Stopień otwartości**: Wybrałem stopień otwartości jako TARGET, ponieważ jest klasą nadawaną przez.
- **Zasób posiada dane wysokiej wartości**: Ta zmienna wskazuje, czy dany zasób zawiera dane o wysokiej wartości. Może być użyteczna do identyfikacji zasobów, które są szczególnie cenne lub istotne dla użytkowników.
- **Zasób posiada dane dynamiczne**: Określa, czy zasób zawiera dynamiczne dane, które mogą ulegać zmianom w czasie. Jest to przydatna informacja, jeśli chcesz rozróżnić zasoby statyczne od tych, które są aktualizowane regularnie.
- **Zasób posiada dane badawcze**: Ta zmienna wskazuje, czy dany zasób zawiera dane badawcze, co może być istotne dla osób poszukujących źródeł informacji do badań naukowych lub analiz.
- **Zasób zawiera wykaz chronionych danych**: Informuje, czy zasób zawiera wykaz danych, które są objęte ochroną lub ograniczeniami dostępu. Jest to istotna informacja dla osób odpowiedzialnych za zarządzanie danymi wrażliwymi lub prywatnymi.
- **Liczba wyświetleń**: Ta zmienna wskazuje liczbę wyświetleń danego zasobu. Może być przydatna do oceny popularności zasobu lub stopnia zainteresowania danymi przez użytkowników.
- **Liczba pobrań**: Określa liczbę pobrań danego zasobu. Podobnie jak liczba wyświetleń, może być używana do oceny popularności lub stopnia wykorzystania danych.

Opis danych zbiorów danych:

- **Id zbioru danych**: Unikalne identyfikatory zbiorów danych. WAŻNE: Zbiór Danych, oznacza grupę Zasobów.
- print(df['Id zasobu'].nunique()) -> 37601
- print(df['Id zbioru danych'].nunique()) -> 2800
- **Zbiór danych posiada dane wysokiej wartości**: Podobnie jak w przypadku zasobów, ta zmienna wskazuje, czy dany zbiór danych zawiera dane o wysokiej wartości.

– **Zbiór danych posiada dane dynamiczne**: Analogicznie do kolumny 10, ta zmienna określa, czy dany zbiór danych zawiera dynamiczne dane, które mogą ulegać zmianom w czasie.

– **Zbiór danych posiada dane badawcze**: Podobnie jak w kolumnie 11, ta zmienna wskazuje, czy dany zbiór danych zawiera dane badawcze.

– **Link zbioru**: Link do konkretnego zbioru danych. Może być używany do bezpośredniego dostępu do danych lub udostępniania linku do innych użytkowników.

– **Data utworzenia zbioru danych**: Informuje o dacie utworzenia danego zbioru danych. Może być użyteczna do śledzenia historii danych lub analizy zmian w czasie.

– **Data modyfikacji zbioru danych**: Określa datę ostatniej modyfikacji zbioru danych. Jest to istotna informacja w kontekście aktualności danych i ich ewentualnych zmian.

– **Liczba obserwujących**: Informuje o liczbie użytkowników, którzy obserwują dany zasób lub zbiór danych. Może być używana do oceny zainteresowania danych przez społeczność.

Opis danych instytucji:

– **Id instytucji**: Unikalne identyfikatory instytucji lub organizacji odpowiedzialnych za udostępnianie danych.

– **Link instytucji**: Link do strony internetowej instytucji lub organizacji. Może być używany do uzyskania dodatkowych informacji o źródle danych.

– **Tytuł**: Tytuł instytucji.

– **Rodzaj**: Rodzaj instytucji.

– **Data utworzenia instytucji**: Informuje o dacie utworzenia instytucji lub organizacji odpowiedzialnej za udostęnanie danych

– **Liczba udostepnionych zbiorow danych**: Określa liczbę innych zbiorów danych udostępnionych przez tę samą instytucję lub organizację. Może być użyteczne do oceny aktywności udostępniania danych przez daną jednostkę.

Należy napomknąć, że Stopień otwartości odnosi się do stopnia, w jakim dane są udostępniane w sposób dostępny, zrozumiały i wykorzystywalny przez komputery oraz ludzi, zgodnie z pięciogwiazdkowym schematem opracowanym przez Sir Tima Berners-Lee. Im wyższy stopień otwartości, tym dane są bardziej dostępne, ustrukturyzowane i połączone, co ułatwia ich interpretację i analizę. Dokładne wyjaśnienie znajduje się w tym artykule: [1]

4.2. Do czego można użyć danych w kontekście Uczenia Maszynowego

Na podstawie ’Liczby wyświetleń’ i ’Liczby pobrań’ (zmienne w czasie), można stworzyć feature określający popularność zasobu i wykorzystać w uczeniu nadzorowanym. Jako zmienne zależne wykorzystując zmienne nie zmieniające się w czasie, pozwoli to na przewidywanie popularności / skuteczności, przyszłych zasobów danych.

4.3. EDA i FE

Jest bardzo dużo kolumn (aż 29). Na początku, należy wyodrębnić cechy nieistotne w kontekście tworzenia modelu ML, by ułatwić sobie dalszą analizę i zredukować wymiar pracy. Jeśli będziemy dokonywać uczenia nadzorowanego, z pewnością nieistonymi kolumnami będą:

– Id zasobu

– Link zasobu

– Nazwa

– Link zbioru

– Link instytucji

– Tytuł

– Data utworzenia instytucji

Pozbyto się tych kolumn, przedstawione na Rysunku 4.2

```
columns_to_drop = ['Id zasobu','Link zasobu','Nazwa','Link zbioru','Link instytucji','Tytul','Data utworzenia instytucji']
df = df.drop(columns_to_drop,axis=1)
```

Rysunek 4.2: *Usuwanie zbędnych kolumn*

Poza kolumną 'Formaty po konwersji', jest bardzo mało brakujących danych Rysunek 4.3 (mniej niż 1 procent). Więc nie należy przejmować się powodami ich nieobecności. Można skategoryzować je jako missing completely at random (MCAR). Następnie usunąć rekordy z brakującymi danymi.

```
df.isna().mean()
```

```
Typ                                        0.002070
Format                                     0.005706
Formaty po konwersji                       0.812468
Data utworzenia zasobu                     0.002070
Data modyfikacji zasobu                    0.002070
Stopien otwartosci                         0.002070
Zasob posiada dane wysokiej wartosci       0.002070
Zasob posiada dane dynamiczne              0.002070
Zasob posiada dane badawcze                0.002070
Zasob zawiera wykaz chronionych danych     0.002070
Liczba wyswietlen                          0.002070
Liczba pobran                              0.002070
Id zbioru danych                           0.001115
Zbior danych posiada dane wysokiej wartosci 0.001115
Zbior danych posiada dane dynamiczne       0.001115
Zbior danych posiada dane badawcze         0.001115
Data utworzenia zbioru danych              0.001115
Data modyfikacji zbioru danych             0.001115
Liczba obserwujacych                       0.000000
Id instytucji                              0.000000
Rodzaj                                     0.000000
Liczba udostepnionych zbiorow danych       0.000000
dtype: float64
```

Rysunek 4.3: *Udział brakujących wartości*

Kolumnę 'Formaty po konwersji' uznano jako missing not at random (MNAR), ponieważ brakujące dane są spowodowane brakiem konwersji danych, co sugeruje istnienie jasno widocznego mechanizmu przyczynowo-skutkowego. Należy podmienić wartości NaN na wartości 'unchanged'. Jedynym problemem jest to, że niektóre 'Formaty po konwersji' są nieatomowe, posiadają dwie lub więcej wartości, np. 'csv, json-ld'.

Właściwie to Format po konwersji, jest aktualnym Formatem, więc można podmienić Wartości Format, tymi aktulanymi. Oczywiście pomijając wartości 'unchanged'. Teraz można pozbyć się kolumny 'Format po Konwersji'. Ponownie redukując liczbę cech. Przedstawione na Rysunek 4.4

```
df['Format'] = df.apply(lambda row: row['Formaty po konwersji'] if row['Formaty po konwersji'] != 'unchanged' else row['Format'], axis=1)

df = df.drop('Formaty po konwersji',axis=1)
df.head(5)
```

Rysunek 4.4: *Aktualizacja formatu i redukcja kolumn*

Nastepnie zbadano kardynalność cech, zaczęto od sprawdzenia poprawności typów danych, widoczne na Rysunek 4.5

```
df.info()
<class 'pandas.core.frame.DataFrame'>
Int64Index: 37464 entries, 0 to 37600
Data columns (total 21 columns):
 #   Column                                      Non-Null Count  Dtype
---  ------                                      --------------  -----
 0   Typ                                         37464 non-null  object
 1   Format                                      37464 non-null  object
 2   Data utworzenia zasobu                      37464 non-null  object
 3   Data modyfikacji zasobu                     37464 non-null  object
 4   Stopien otwartosci                          37464 non-null  float64
 5   Zasob posiada dane wysokiej wartosci        37464 non-null  object
 6   Zasob posiada dane dynamiczne               37464 non-null  object
 7   Zasob posiada dane badawcze                 37464 non-null  object
 8   Zasob zawiera wykaz chronionych danych      37464 non-null  object
 9   Liczba wyswietlen                           37464 non-null  float64
 10  Liczba pobran                               37464 non-null  float64
 11  Id zbioru danych                            37464 non-null  float64
 12  Zbior danych posiada dane wysokiej wartosci 37464 non-null  object
 13  Zbior danych posiada dane dynamiczne        37464 non-null  object
 14  Zbior danych posiada dane badawcze          37464 non-null  object
 15  Data utworzenia zbioru danych               37464 non-null  object
 16  Data modyfikacji zbioru danych              37464 non-null  object
 17  Liczba obserwujacych                        37464 non-null  int64
 18  Id instytucji                               37464 non-null  int64
 19  Rodzaj                                      37464 non-null  object
 20  Liczba udostepnionych zbiorow danych        37464 non-null  int64
dtypes: float64(4), int64(3), object(14)
memory usage: 6.3+ MB
```

Rysunek 4.5: *Typy danych (Dtype)*

Zmienne określające Daty, mają obecnie typ object, należy zamienić go na typ datetime. Zmienne określające Id, są zapisane jako int64 lub float64. Stopień otwartości jako float64. Należy zamienić je na typ str Rysunek 4.6, żeby dalej zostały uznane jako zmienna kategoryczna (Dtype == object).

```
df["Data utworzenia zasobu"] = pd.to_datetime(df["Data utworzenia zasobu"])
df["Data modyfikacji zasobu"] = pd.to_datetime(df["Data modyfikacji zasobu"])
df["Data utworzenia zbioru danych"] = pd.to_datetime(df["Data utworzenia zbioru danych"])
df["Data modyfikacji zbioru danych"] = pd.to_datetime(df["Data modyfikacji zbioru danych"])

df["Stopien otwartosci"] = df["Stopien otwartosci"].astype(str)
df["Id zbioru danych"] = df["Id zbioru danych"].astype(str)
df["Id instytucji"] = df["Id instytucji"].astype(str)
```

Rysunek 4.6: *Zamiana typów*

Wygląda na to, że większość zmiennych posiada niską kardynalność cech, Rysunek 4.7 Właściwie tylko Id's, co było do przewidzenia mają wysoką, oraz Format.

```
categorical_columns = [col for col in df.columns if df[col].dtype == 'object']
for column_name in categorical_columns:
    unique_values = len(df[f'{column_name}'].unique())
    # Uznałem 10 jako granicę mocy zbioru pomiędzy wysoką a niską
    cardinality = 'wysoka' if unique_values > 10 else 'niska'
    print(f'\nLiczba etykiet zmiennej {column_name}: {unique_values}, Moc zbioru: {cardinality}')
```

```
Liczba etykiet zmiennej Typ: 3, Moc zbioru: niska

Liczba etykiet zmiennej Format: 29, Moc zbioru: wysoka

Liczba etykiet zmiennej Stopien otwartosci: 6, Moc zbioru: niska

Liczba etykiet zmiennej Zasob posiada dane wysokiej wartosci: 3, Moc zbioru: niska

Liczba etykiet zmiennej Zasob posiada dane dynamiczne: 3, Moc zbioru: niska

Liczba etykiet zmiennej Zasob posiada dane badawcze: 3, Moc zbioru: niska

Liczba etykiet zmiennej Zasob zawiera wykaz chronionych danych: 2, Moc zbioru: niska

Liczba etykiet zmiennej Id zbioru danych: 2745, Moc zbioru: wysoka

Liczba etykiet zmiennej Zbior danych posiada dane wysokiej wartosci: 3, Moc zbioru: niska

Liczba etykiet zmiennej Zbior danych posiada dane dynamiczne: 3, Moc zbioru: niska

Liczba etykiet zmiennej Zbior danych posiada dane badawcze: 3, Moc zbioru: niska

Liczba etykiet zmiennej Id instytucji: 357, Moc zbioru: wysoka

Liczba etykiet zmiennej Rodzaj: 3, Moc zbioru: niska
```

Rysunek 4.7: *Kardynalność kategorycznych cech*

Zredukowano kardynalność zmiennej Format, pozbywając się nie-atomowych etykiet. Ponieważ, w kolumnie powinny znajdować się dane niepodzielne, jest to też ważne w kontekście uczenia maszynowego, pozbyto się wieloznaczności informacji, co powinno pozytywnie wpłynąć na nauczanie. Ilość nie-atomowych etykiet stanowi zaledwie mniej niż jeden procent naszych danych, więc redukcja ich do wartości 'csv', nie powinna być szkodliwa. Cały proces przedstawiony na Rysunek 4.8

```
nie_atomowe = [string for string in df["Format"].unique() if ',' in string]
print(nie_atomowe)
```

```
['json-ld, csv', 'csv, json-ld', 'json-ld, json-ld, csv, csv']
```

```
print(round(len(df[df['Format'].isin(nie_atomowe)]) / len(df),3),"%")
```

```
0.099 %
```

```
df['Format'] = df.apply(lambda row: 'csv' if row['Format'] in nie_atomowe else row['Format'], axis=1)
```

```
unique_values = len(df['Format'].unique())
cardinality = 'wysoka' if unique_values > 10 else 'niska'
print(f'\nLiczba etykiet zmiennej Format: {unique_values}, Moc zbioru: {cardinality}')
```

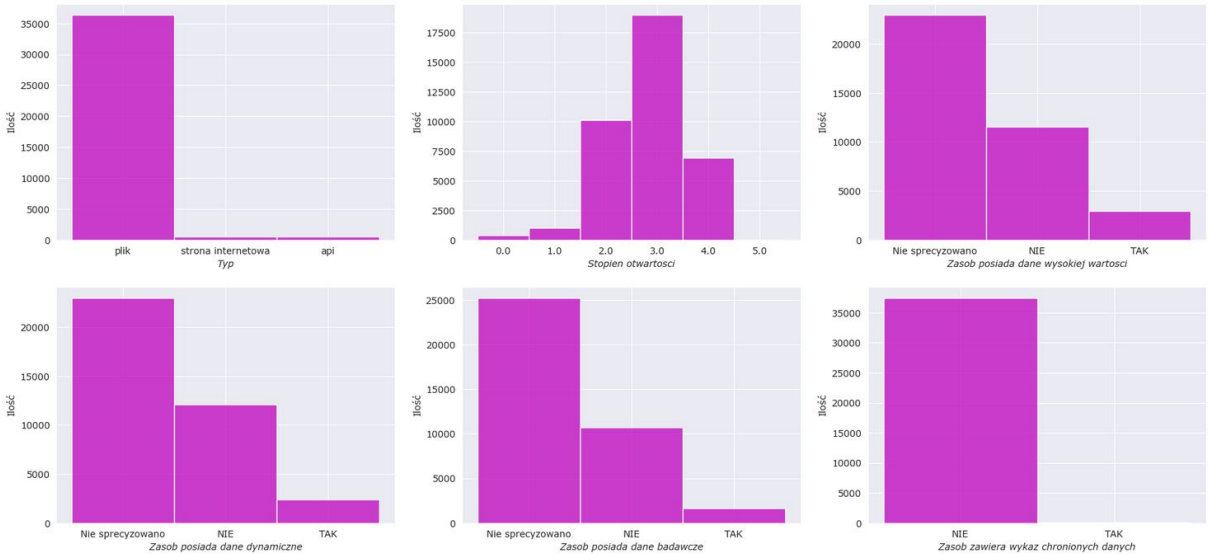
```
Liczba etykiet zmiennej Format: 26, Moc zbioru: wysoka
```

Rysunek 4.8: *Redukcja kardynalności formatu*

Wizulizacja danych kategorycznych

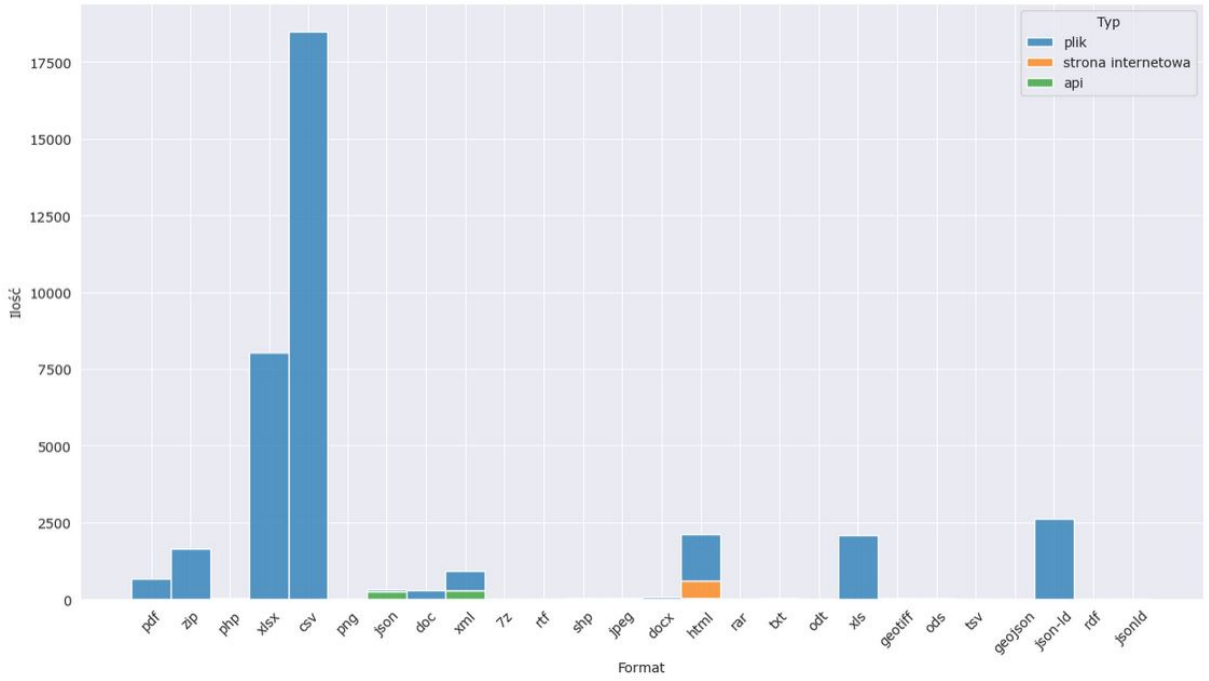
Rozpoczęto od przedstawienia danych dla zmiennych z podzbioru: Zasoby, o niskiej mocy zbioru

Jak widać na załączonym Rysunku 4.9, dominującym Typem zasobów jest 'plik'. Najczęściej spotykane stopnie otwartości to 2, 3 i 4. Warto zauważyć, że Stopień otwartości równy 0, nie istnieje w oficjalnej skali stworzonej przez Sir Tima Berners-Lee [1], co sugeruje, że autorzy serwisu nie do końca zalecają się do oficjalnego schematu. Rozkłady dotyczące, dynamiczności, badawczych oraz wysokiej wartości zasobów są bardzo podobne, zapewne te same zasoby pokrywają się wartościami w tych trzech kolumnach. Praktycznie żadne zasoby nie zawierają danych chronionych.



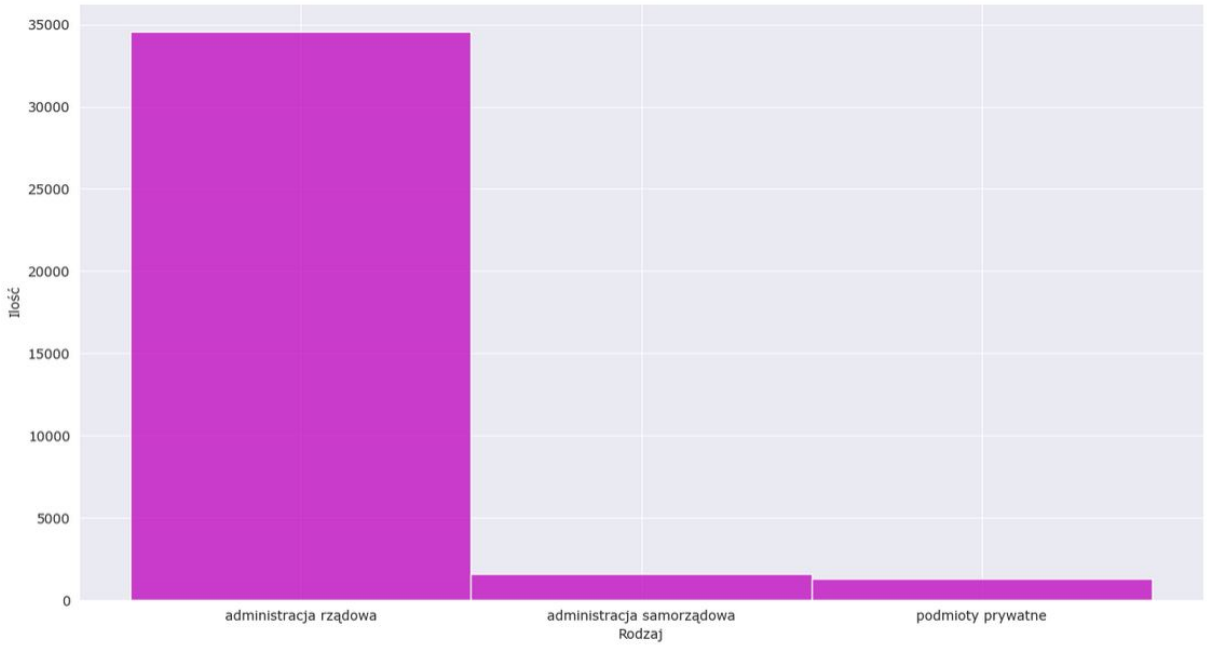
Rysunek 4.9: *Rozkłady zmiennych kategorycznych podzbioru: 'Zasoby', o niskiej mocy zbioru*

Rozkład zmiennej Format 4.10, dodatkowo pogrupowany po typie zasobu, uwidacznia, że dominującymi formatami są csv, który stanowi ponad połowę zasobów oraz xlsx (format excelowy).Taki nierównomierny rozkład etykiet nie jest najlepszy w kontekście uczenia maszynowego. Ponownie widać, że dominującym typem zasobu jest plik.



Rysunek 4.10: *Rozkład zmiennej Format, pogrupowanej po typie zasobu*

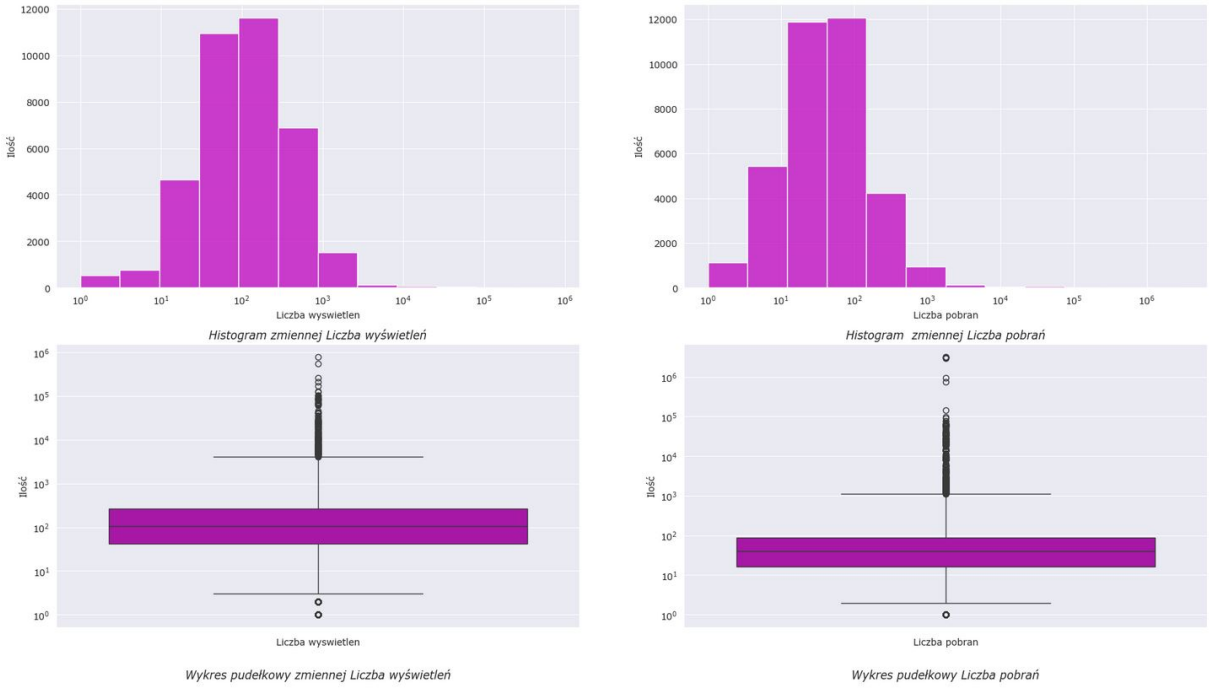
Tak jak wspominano we wstępie, zdecydowana większość zasobów pochodzi z instytucji rządowych, co widać na Rysunku 4.11, niewielki procent pochodzi od podmiotów prywatnych.



Rysunek 4.11: *Rozkład zmiennej Rodzaj Instytucji*

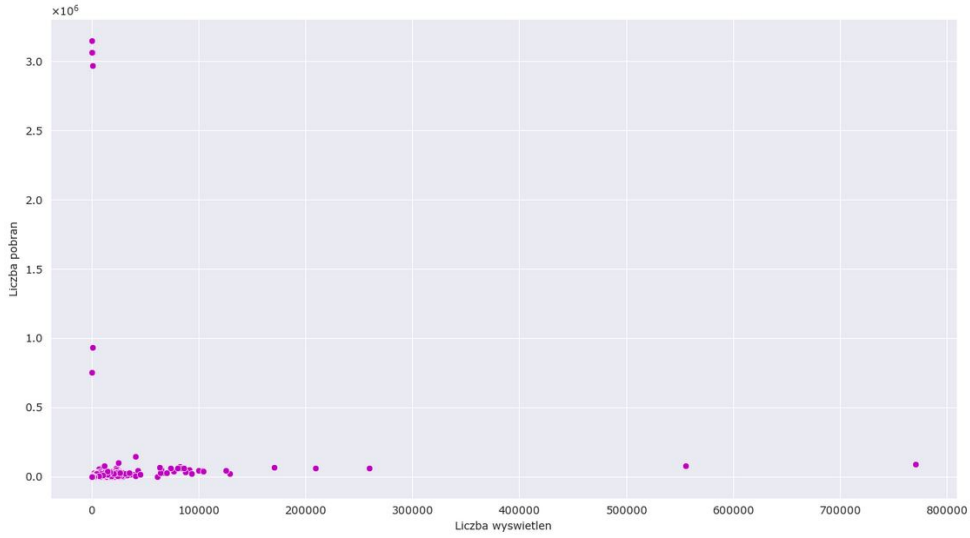
Wizulizacja danych numerycznych

Zbadano rozkłady 4.12, zmiennych liczba wyświetlneń oraz pobrań. Są to jedne z ciekawszych informacji w naszym zestawie danych, rzuca się w oczy iż przypominają lekko przesunięty w lewo rozkład normalny. Widać wiele outlierów, które są większe nawet od 3 rzędy wielkości od większości danych.



Rysunek 4.12: *Rozkłady zmiennych numerycznych podzbioru: 'Zasoby', w skali logartymicznej*

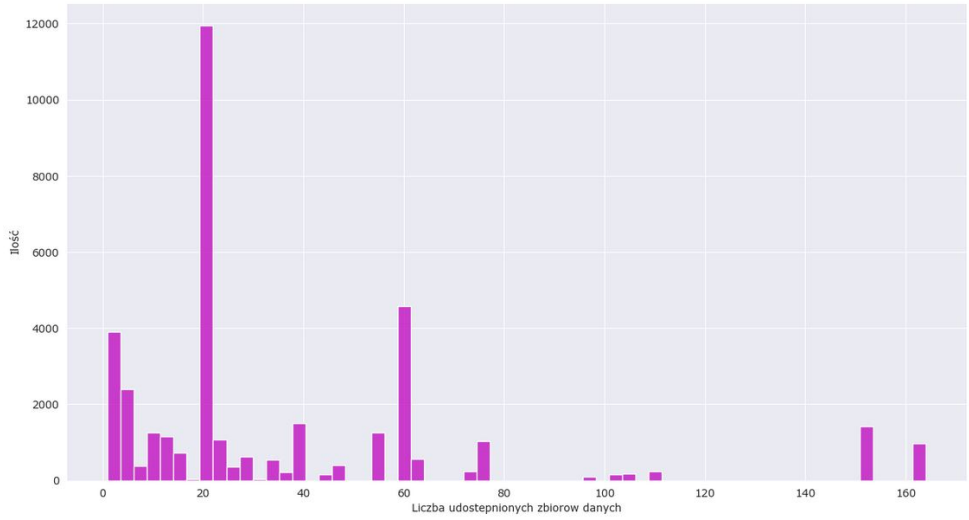
Nie występuje żadna zależność liniowa, pomiędzy liczbą wyświetleń a liczbą pobrań 4.13. Co ciekawe prawie jedna czwarta badanych zasobów, ma większą ilość pobrań od ilości wyświetleń, co można by uznać za anomalię, choć wcale nie wykluczone jest żeby na jednej sesji pobrać zasób wielokrotnie. Podejrzano również, że wyświetlenia mogą nie być czasem rejestrowane, ponieważ istnieją zasoby, których liczba wyświetleń wynosi 0, a zostały pobrane. Anomaliami na pewno jest 7 rekordów, których liczba pobrań, jest w milionach a liczba wyświetleń bliska jest zeru. Tych rekordów się pozbędziono.



Rysunek 4.13: *Wykres punktowy zależności Liczby pobrań od Liczby wyświetleń*

Ilość udostępninych danych 4.14, przez konkretną instytucję, może przekładać się na jej renomę i

doświadczenie, więc może to być istotna informacja w kontekście prognowozowania popularności zasobu.



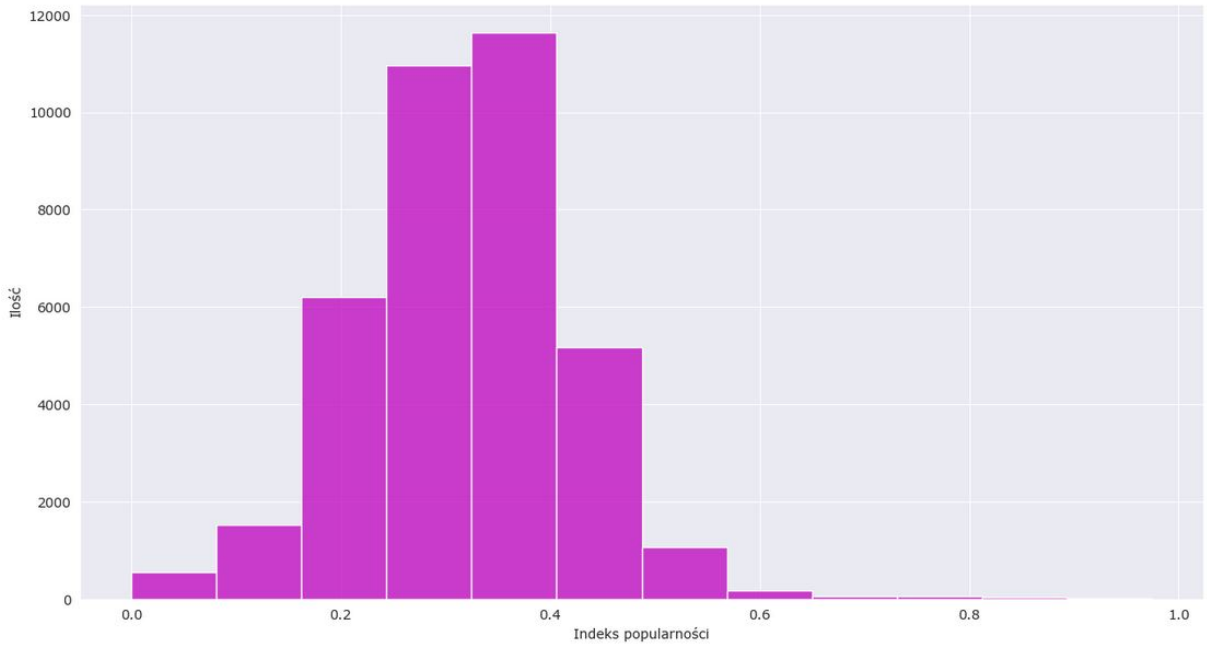
Rysunek 4.14: *Wykres punktowy zależności Liczby pobrań od Liczby wyświetleń*

Dokonano usunięcia anomalii 4.15. Wspomnianych przy omawianiu wykresu 4.13. Dodatkowo stworzono 2 nowe cechy, tj. logarytmy liczby pobrań i liczby wyświetleń 4.15. Na podstawie których stworzono feature: 'Indeks popularności' 4.15. Skalowanie logarytmami zostało dokonano, ze względu na nierównomierny rozkład wartości dla Indeksu przed skalowaniem. Nadano również wagę: 2 dla Liczby pobrań i 1 dla Liczby wyświetleń, ponieważ pobrania są ważniejsze w aspekcie 'popularności', ze względu na naturę tych cech.

```
df = df[df['Liczba pobran'] < 500000]
df['pobrania_log'] = np.log(df['Liczba pobran'] + 1)
df['pobrania_log']
df['wyswietlenia_log'] = np.log(df['Liczba wyswietlen'] + 1)
df['wyswietlenia_log']
df['Indeks popularności'] = (df['pobrania_log'] * 2 + df['wyswietlenia_log']) / (df['pobrania_log'].max() * 2 + df['wyswietlenia_log'].max())
```

Rysunek 4.15: *Wykres punktowy zależności Liczby pobrań od Liczby wyświetleń*

Indeks popularności, przypomina rozkład normalny 4.16, co zazwyczaj jest pozytwyną rzeczą w prognozowaniu.



Rysunek 4.16: *Rozkład Indeksu popularności zasobu*

4.4. Target

Jako kolumnę TARGET wybrano 'Indeks popularności' jest to specjalnie przygotowany, znormalizowany i zważony Indeks, o dobrym rozkładzie, które mówi o tym jak bardzo dany zasób stał się popularny. Dla konkrentego zasobu jest to wartość zmienna w czasie, lecz możemy go prognozować w oparciu o zmienne statyczne, czyli cechy nadawane mu przy tworzeniu. Takie prognozy, mogą być przydatne dla autorów zasobów, którym zależy na optymalnym dobraniu parametrów.

4.5. Features

Na podstawie wcześniej przeprowadzonej analizy, można stwierdzić, że features, które mogą być dobre do wyznaczania 'Indeksu popularności' są:

- Typ - określa, sposób dostępu do zasobów, jest ważny dla osób, które zamierzają pobrać zasób
- Format - w zależoności od formatu, dostępny jest podgląd danych, co wpływa na jego popularność
- Zasob posiada dane wysokiej wartosci - nazwa zachęca do analizy, i może być kusząca dla osób szukających ciekawych danych
- Rodzaj - podmioty prywatne, mogą oferować ciekawsze lub po prostu inne dane niż instytujce publiczne
- Liczba udostepnionych zbiorow danych - określa doświadczenie danej instytucji oraz nakład wydanych zasobów

Ewentualnie można by uwzględnić, Id Instytucji, ponieważ niektóre mogą mieć większą renomę, niż zwykle i mogą być chętniej przeglądane.

5. Wnioski

Poprzez EDA, ustalono, że liczba wyświetleń i pobrań są istotnymi cechami, które mogą być wykorzystane do prognozowania popularności zasobów.

Warto zauważyć, że FE nie jest procesem jednorazowym. W miarę postępu analizy danych i eksperymentów z modelami, istnieje możliwość powrotu do tego etapu, aby dodać nowe cechy, usunąć zbędne lub zmodyfikować istniejące w celu poprawy jakości modelu.

Choć analiza i prognozowanie popularności zasobów, prawdopodobnie nie jest priorytetowa dla serwisu rządowego, to narzędzie do określania parametrów udostępnianych danych w celu zwiększenia zasięgów mogłoby mieć ciekawe zastosowanie dla prywantych podwykonawców, działających na takich serwisach jak Kaggle.

Wtedy można by brać pod uwagę więcej parametrów, na przykład rozmiar pliku, ilość kolumn, tematyka zasobu itp.

Bibliografia

[1] 5 poziomów otwartości danych — od pdf do lod. *Medium*, 2021.