

1 Część I

1.1 Opis programu

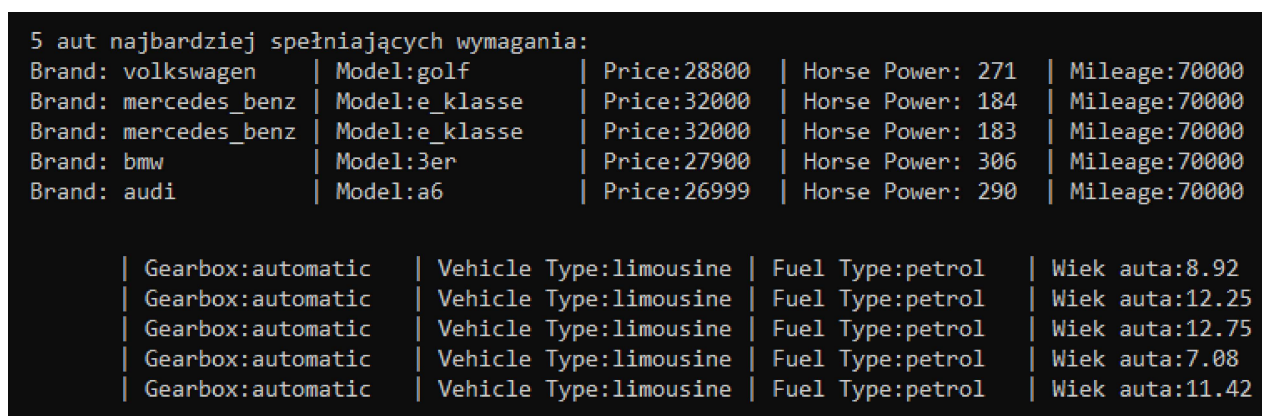
Program ten jest aplikacją konsolową, który na podstawie bazy danych (cleanedCar.csv) oraz zapytań do użytkownika będzie znajdował najbardziej odpowiednie egzemplarze aut. Bazować on będzie na algorytmie k-najbliższych sąsiadów (KNN).

1.2 Instrukcja obsługi

Program można uruchomić bezpośrednio z pliku auta_uzywane_knn.py lub z konsoli języka Python. Po uruchomieniu do dyspozycji użytkownika ukaże się menu wraz z dostępnymi opcjami.

1. **Start** - Program rozpoczyna przeprowadzenie ankiety na podstawie której znajduje i wyświetla najbardziej pasujące modele aut.
2. **Pokaż informacje** - Na ekranie zostaje wyświetlona zawartość pliku informacyjnego.
9. **Zakończ program** - Zamknięcie programu bez wykonywania dodatkowych czynności.

Podczas przeprowadzania ankiety nie ma możliwości powrotu do poprzednich pytań, a w przypadku błędu można powrócić do głównego menu poprzez podanie wartości -1.



5 aut najbardziej spełniających wymagania:

Brand: volkswagen	Model: golf	Price: 28800	Horse Power: 271	Mileage: 70000
Brand: mercedes_benz	Model: e_klasse	Price: 32000	Horse Power: 184	Mileage: 70000
Brand: mercedes_benz	Model: e_klasse	Price: 32000	Horse Power: 183	Mileage: 70000
Brand: bmw	Model: 3er	Price: 27900	Horse Power: 306	Mileage: 70000
Brand: audi	Model: a6	Price: 26999	Horse Power: 290	Mileage: 70000

Gearbox: automatic	Vehicle Type: limousine	Fuel Type: petrol	Wiek auta: 8.92
Gearbox: automatic	Vehicle Type: limousine	Fuel Type: petrol	Wiek auta: 12.25
Gearbox: automatic	Vehicle Type: limousine	Fuel Type: petrol	Wiek auta: 12.75
Gearbox: automatic	Vehicle Type: limousine	Fuel Type: petrol	Wiek auta: 7.08
Gearbox: automatic	Vehicle Type: limousine	Fuel Type: petrol	Wiek auta: 11.42

Rysunek 1: Przykładowe wyniki programu.

1.3 Dodatkowe informacje

Wymagania:¹

- Python 3.8.5
- Biblioteki: Pandas 1.1.3, pathlib
- Baza danych z odpowiednią nazwą w lokalizacji programu.

¹Wymagania zostały utworzone w oparciu o platformę, na której program był napisany i testowany.

2 Część II

2.1 Teoria

Algorytm k-najbliższych sąsiadów jest jednym z algorytmów klasyfikacji nieparametrycznej, która została zapoczątkowana przez Evelyn Fix'a i Joseph Hodges'a już w 1951 roku, a rozwijana dalej przez Thomas Covea'a. Algorytm ten jest używany w klasyfikacji i regresji.

Klasyfikacja statystyczna jest to rodzaj algorytmu, którego zadaniem jest przydzielenie obserwacji statystycznej na podstawie danych atrybutów. Klasycznie najbliższe obserwacje biorą udział w głosowaniu, którego wynik jest wynikiem predykcji.

Najbliższa obserwacja sprowadza się do minimalizacji pewnej metryki (np. Minkowskiego) czyli znalezienie najmniejszej odległości pomiędzy wektorami.

Odległość Minkowskiego jest to uogólniona miara odległości między punktami w przestrzeni Euklidesowej, która wyraża się wzorem:

$$L_m(x, y) = \left(\sum_{i=1}^n |x_i - y_i|^m \right)^{1/m}$$

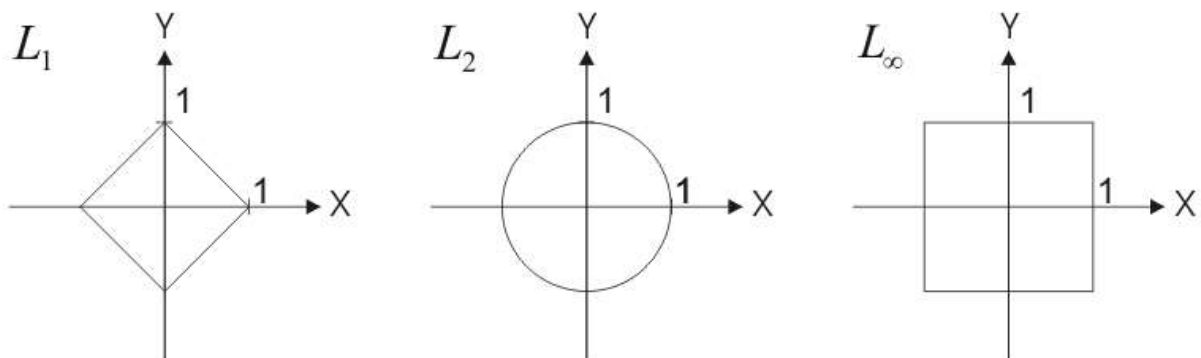
Gdzie:

x - pierwszy wektor,

y - drugi wektor,

n - wymiar wektorów (ilość parametrów),

m - parametr odpowiedzialny za kształt w przestrzeni.



Rysunek 2: przykłady parametru m w \mathbb{R}^2

2.2 Przykład obliczeniowy

Poniżej zostanie przedstawiony przykład obliczenia odległości przy użyciu metryki Minkowskiego w przestrzeni euklidesowej dla pojedynczej próbki danych.

Dane wejściowe: (v1, v2, m)

- v1** - przygotowana przez użytkownika próbka danych,
- v2** - rekord z bazy danych,
- m** - parametr odpowiedzialny za kształt w przestrzeni.

Wartości próbki v1:

- Znormalizowany wiek auta : 0.108162100456621
- Znormalizowany przebieg : 0.4666666666666667
- Znormalizowana moc silnika : 0.28
- Znormalizowana cena : 0.10275862068965518

Wartości rekordu z bazy danych v2:

- Znormalizowany wiek auta : 0.052226
- Znormalizowany przebieg : 0.166667
- Znormalizowana moc silnika : 0.42
- Znormalizowana cena : 0.152759

Dla każdego elementu jest obliczana wartość bezwzględna z ich różnicy, która jest później podnoszona do kwadratu.

$$|v1_i - v2_i|^2$$

Wiek: 0.003128847334293196

Przebieg: 0.08999980000011111

Moc silnika: 0.019599999999999999

Cena: 0.0025000379311783593

Sumujemy wszystkie powyższe wartości.

$$\sum_{i=1}^n |v1_i - v2_i|^2$$

Suma: 0.11522868526558265

Wynikiem jest pierwiastek z obliczonej sumy.

$$\left(\sum_{i=1}^n |v1_i - v2_i|^2\right)^{1/2}$$

Dane wyjściowe: (Odległość Minkowskiego)

- **0.33945350972641697**

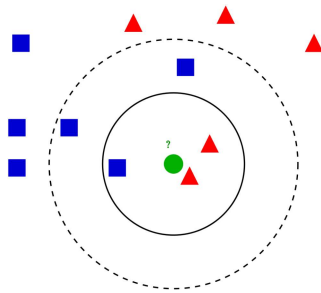
2.3 Algorytm

Klasyczna wersja algorytmu k -najbliższych sąsiadów posiada założenia:

- Dany jest zbiór uczący posiadający obserwacje, gdzie każda zawiera wektor zmiennych objaśniających oraz zmienną wynikową,
- Dana jest obserwacja zawierająca wektor zmiennych objaśniających dla której chcemy prognozować zmienną wynikową.

Algorytm polega na:

- Znalezieniu odległości pomiędzy poszukiwaną obserwacją, a każdą obserwacją ze zbioru uczącego na podstawie wybranej metryki,
- Wyborze k najbliższych sąsiadów,
- Przeprowadzeniu głosowania na podstawie zmiennych wynikowych,
- Zwrócenie wygranego w głosowaniu jako wyniku algorytmu.



Dla $k=3$ (mniejszy okrąg), zielona kropka zostanie zakwalifikowana do czerwonych trójkątów. Gdy $k=5$ (większy okrąg) – do niebieskich kwadratów.

Rysunek 3: Przykład klasyfikacji KNN

W przypadku tego programu, algorytm KNN po wyliczeniu odległości nie przeprowadza głosowania, ale zwraca tyle wyników ile życzył sobie użytkownik. W obliczaniu odległości brane pod uwagę są jedynie parametry takie jak:

- cena pojazdu,
- moc silnika w koniach mechanicznych,
- przebieg pojazdu,
- wiek pojazdu.

Pozostałe zmienne, które nie są wartościami numerycznymi, takie jak typ paliwa, są używane, aby zawęzić zakres poszukiwań zgodnie z podanymi preferencjami użytkownika.

Data:*sample* - próbka danych*x_org* - baza aut*k* - ilość zwracanych wyników (sąsiadów)*min_b* - minimalna cena*max_b* - maksymalna cena**Result:***results* - tablica zawierająca nie więcej niż *k* najlepiej dopasowanych aut

Na podstawie próbki i budżetu wyodrębnij dane z bazy aut;

x_lim = Ograniczona baza aut na podstawie próbki i budżetu;*distances* = [];*results* = [];**for** *i* **in** *range*(0, *len*(*x_limited*)) **do**| Oblicz odległość Minkowskiego i dodaj do *distances***end****for** *i* **in** *range* (0, *k*) **do**| **for** *j* **in** *range* (*i*, *len*(*distances*)) **do**| | **if** *distances*[*i*] > *distances*[*j*] **then**| | | Zamień *distances*[*i*] i *distances*[*j*] miejscami;| | | Zamień rekord *i* i rekord *j* w bazie aut miejscami;| | **end**| **end****end****if** *x_lim* **is not empty** **then**| **if** *len*(*x_lim*) < *k* **then**| | **for** *i* **in** *range* (0, *len*(*x_lim*)) **do**| | | Do *results*[] dopisz auto z bazy o pozycji *i*-tej;| | **end**| **end****else**| **for** *i* **in** *range*(0, *k*) **do**| | Do *results*[] dopisz auto z bazy o pozycji *i*-tej;| | **end**| **end****end****Algorithm 1:** KNN bez głosowania, zwracający *k* najlepszych wyników.

2.4 Baza danych

<https://www.kaggle.com/aman2457/usedcarsdatasetenglish50000>

Baza danych zawierająca dane o autach jest przechowywana w pliku cleanedCar.csv, który jest wczytywany wraz ze startem programu. W całej bazie jest 42867 aut, natomiast program pracuje jedynie na w pełni wypełnionych rekordach których jest 32874.

Index	price	vehicleType	gearbox	powerPS	model	kilometer	fuelType	brand	notRepairedDamage	Age
0	5000	bus	manual	158	andere	150000	lpg	peugeot	no	16.25
1	8500	limousine	automatic	286	7er	150000	petrol	bmw	no	23.5
2	8990	limousine	manual	102	golf	70000	petrol	volkswagen	no	11.58
3	4350	small car	automatic	71	fortwo	70000	petrol	smart	no	13.5
6	1990	limousine	manual	90	golf	150000	diesel	volkswagen	no	23
7	590	bus	manual	90	megane	150000	petrol	renault	no	23.58
9	5299	small car	automatic	71	fortwo	50000	petrol	smart	no	10.75

Rysunek 4: Przykładowe rekordy z bazy danych

- Index - klucz główny,
- price - cena auta,
- vehicleType - rodzaj nadwozia:
 - bus, limousine, small car, kombi, coupe, suv, cabrio, others
- gearbox - rodzaj skrzyni biegów:
 - manual, automatic
- powerPS - moc wyrażona w koniach mechanicznych
- model - nazwa modelu samochodu
 - andere - inne
- kilometer - aktualny przebieg pojazdu
- fuelType - rodzaj paliwa
 - petrol, diesel, lpg, hybrid, electric, cng, others
- brand - marka samochodu
- notRepairedDamage - auto zawiera nienaprawione uszkodzenie
 - yes, no
- age - wiek pojazdu