

SYSTEM EKSPERCKI - INSTRUKCJA UŻYTKOWNIKA

1. Przygotowanie przed uruchomieniem programu

Przed uruchomieniem programu należy zainstalować dodatkowe biblioteki zewnętrzne. Jest to realizowane za pomocą programu pip. Informacje o bibliotekach są zawarte w pliku 'requirements.txt'. Aby je zainstalować należy uruchomić terminal w folderze, w którym ten plik się znajduje i wywołać komendę:

```
python3 -m pip install -r requirements.txt
```

Po wykonaniu tej czynności program powinien działać poprawnie.

2. Przyjęte założenia

W realizacji programu przyjęto następujące założenia:

- bazy reguł muszą być zapisane w formacie csv
- żadne pola (zarówno nagłówki, jak i wartości) w bazie nie mogą zaczynać się lub kończyć spacją.
- nagłówki pól nie powinny nazywać się 'Quit' - jest to polecenie zarezerwowane do przerywania tworzenia bazy danych od zera
- użytkownik poprawnie wpisuje informacje (zachowuje odpowiednią wielkość znaków i nie popełnia błędów pisowni), w szczególności należy zwrócić uwagę na poprawność podczas uzupełniania bazy danych.
- użytkownik zna wartości, jakie może mieć zmienna (np. numeryczne, True / False, tekst) i podaje odpowiednie typy przy uzupełnianiu bazy danych.

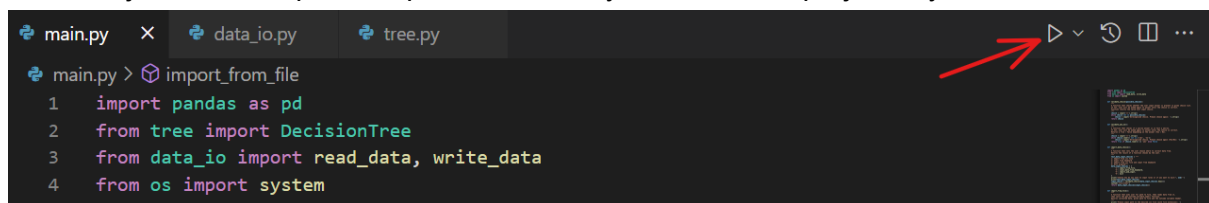
3. Uruchomienie programu

Program można uruchomić na 2 sposoby:

- przez zwykłe uruchomienie go w Visual Studio Code
- przez wpisanie następującego polecenia w wierszu poleceń:

```
python3 main.py
```

*lokalizacja w wierszu poleceń powinna wskazywać na folder projektowy



Uruchomienie programu w Visual Studio Code

```
pwysocki@DESKTOP-GLS68AI:~/Nauka_Linuxa/vscode/drzewo_decyzyjne$ python3 main.py
```

Uruchomienie programu w terminalu

4. Wskazówki w obsłudze programu

Zostaną omówione niektóre sytuacje w działaniu programu.

A) Importowanie bazy reguł z pliku - należy podać ścieżkę do pliku wraz z jego rozszerzeniem. Jeżeli z jakiegoś powodu nie uda się odczytać zawartości pliku, użytkownik zostanie o tym poinformowany i poproszony o podanie nowej ścieżki.

```
Given path: './datasets/' doesn't lead to a file.
Please give the correct path with the file name and its extension.
>>|
```

Przykład: Podanie ścieżki do folderu, a nie do pliku

```
You do not have permission to open file under path:
./datasets/test.csv
Please specify path to another file.
>>|
```

Przykład: Podanie ścieżki do pliku, do którego odczytania nie ma się uprawnień

```
No file found under path:
./datasets/non_existing_file.csv
Please check if file exists and make sure to specify its extension or specify another file.
>>|
```

Przykład: Podanie ścieżki do nieistniejącego pliku

B) Tworzenie bazy reguł od podstaw - należy podać nazwy przynajmniej 2 zmiennych, przy czym wskazane jest podanie większej ilości, gdyż jedna z nich zostanie potem wybrana jako zmienna wynikowa. Dodatkowo, mała liczba zmiennych może negatywnie wpłynąć na skuteczność programu. Zmienna nie może nazywać się Quit, gdyż jest to polecenie zarezerwowane do przerywania dodawania zmiennych. Nazwy zmiennych nie mogą się również powtarzać.

```
Please input at least 2 different variable names.
Please input the variable name:
>>Quit
Variable cannot be named "Quit". Try again.
Please input the variable name:
>>|
```

Przykład: Podanie 'Quit' jako nazwy zmiennej

```
Please input at least 2 different variable names.
Please input the variable name:
>>Rower
Please input the variable name:
>>Lalka
Please input the variable name or type "Quit" to end adding:
>>Pies
Please input the variable name or type "Quit" to end adding:
>>Lalka
Error - this variable name has already been given. Please input another variable name:
>>Quit|
```

Przykład: Powtórzenie nazwy zmiennej

C) Nauka / uzupełnienie bazy danych - jeżeli wprowadzona reguła już istnieje w bazie, użytkownik zostanie poproszony o wprowadzenie innej reguły. Dodatkowo, jeżeli wprowadzona reguła jest sprzeczna z już istniejącą, użytkownik zostanie o tym poinformowany, a następnie zapytany, czy chce zamienić już istniejącą na nową.

```
Please input variable "Rower" value:
>>
Variable value cannot be empty! Please input again:
>> █
```

Przykład: Próba nie wprowadzenia wartości

```
Error - rule already in database!:
  Outlook Temperature Humidity      Wind PlayGolf
10   Sunny           Mild   Normal   Strong      Yes

Please input another rule.
```

Przykład: Podanie reguły, która już znajduje się w bazie

```
Now please input at least 2 rules.
Rule no. 1:
Please input feature "Outlook" value
>>Sunny
Please input feature "Temperature" value
>>Mild
Please input feature "Humidity" value
>>Normal
Please input feature "Wind" value
>>Strong
Please input feature "PlayGolf" value
>>No
Warning - rule with the same values but different outcome found!
The rules are:
  Outlook Temperature Humidity      Wind PlayGolf
5   Sunny           Mild   Normal   Strong      Yes
10  Sunny           Mild   Normal   Strong      No
Would you like to replace the old rule (the higher one)? [Yes/No]
>> █
```

Przykład: Podanie reguły sprzecznej z tą znajdującą się już w bazie

D) Wybór z listy możliwych odpowiedzi - należy zwrócić szczególną uwagę na pisownię, w szczególności na rozmiar liter i ilość spacji między wyrazami

```
Please choose the outcome variable from variables listed below:
['Outlook', 'Temperature', 'Humidity', 'Wind', 'Play Golf']
>>pLay gOLf
Unrecognized choice. Please choose again: Play      Golf
Unrecognized choice. Please choose again: Play Golf █
```

Przykład: Podanie odpowiedzi bez uwzględnienia wielkości znaków oraz z nieprawidłową ilością spacji między wyrazami

E) Pytania typu tak / nie - jeżeli pojawi się takie pytanie, użytkownik zostanie poinformowany, że powinien wskazać jedną z tych odpowiedzi

```
The suggested decision is: Yes.  
Is it correct? [Yes/No]  
>>maybe  
Unrecognized choice. Please choose again [Yes/No]: sure  
Unrecognized choice. Please choose again [Yes/No]: True  
Unrecognized choice. Please choose again [Yes/No]: Yes
```

Przykład: Podanie odpowiedzi niebędącej Tak / Nie

F) Wyświetlenie struktury drzewa decyzyjnego - jest to dodatkowa funkcja, którą należy odblokować bezpośrednio w kodzie programu, usuwając komentarz z odpowiedniego polecenia. Należy usunąć znak “#” przy poleceniu `tree.printer()` w funkcji `make_decision()` w pliku `main.py`. Pozwala ona przyjrzeć się strukturze utworzonego drzewa.

```
Outlook: ?  
  Overcast – Yes  
Rainy: Wind: ?  
  Strong – No  
  Weak – Yes  
Sunny: Humidity: ?  
  High – No  
  Normal – Yes
```

Przykład: Wyświetlona struktura drzewa decyzyjnego dla danych z pliku `playgolf_data.csv`

G) Ograniczenie liczby zadawanych pytań - jest to dodatkowa funkcja, którą należy odblokować bezpośrednio w kodzie programu, usuwając komentarz z odpowiedniego polecenia. Należy usunąć znak “#” przy poleceniu `self._max_tree_depth = len(self.variables)` w konstruktorze klasy `DecisionTree` w pliku `tree.py`. Opcja ta pozwala na zredukowanie maksymalnej liczby pytań do ilości zmiennych reguły. Jest ona domyślnie wyłączona, gdyż korzystanie z niej może wiązać się ze zmniejszeniem prawdopodobieństwa prawidłowej decyzji.

5. Dostarczone bazy reguł

Do programu został dostarczony folder ‘datasets’ zawierający następujące bazy:

A) plik `diabetes.csv` - decyzją jest zmienna ‘Has Diabetes’. Baza zawiera informacje o pacjentach, na podstawie których można stwierdzić, czy ktoś ma cukrzycę. Zawiera następujące dane liczbowe: liczba ciąż, poziom glukozy, ciśnienie krwi, grubość skóry, poziom insuliny, BMI pacjenta, funkcja pedi (prawdopodobieństwo cukrzycy ze względu na występowanie przypadków w rodzinie), wiek.

B) plik drug200.csv - decyzją jest zmienna 'Drug'. Baza zawiera informacje o pacjentach, na podstawie których można stwierdzić, jaki typ leku należy mu podać. Zawiera następujące dane: wiek (liczbowe), płeć (M / F), ciśnienie krwi (HIGH / NORMAL / LOW), poziom cholesterolu (HIGH / NORMAL / LOW), stosunek zawartości Sodiu do Potasu we krwi (liczbowe).

C) plik iris.csv - decyzją jest zmienna 'Type'. Baza zawiera informacje o wymiarach irysów, na podstawie których można stwierdzić, do jakiego typu należy badany irys. Zawiera następujące dane liczbowe: długość płatków, szerokość płatków, długość działki kielicha, szerokość działki kielicha.

D) plik playgolf_data.csv - decyzją jest zmienna 'Play Golf'. Baza zawiera informacje o warunkach atmosferycznych, na podstawie których można stwierdzić, czy gra w golfa jest możliwa. Zawiera następujące dane: pogoda (Overcast / Sunny / Rainy), temperatura (Hot / Mild / Cool), wilgotność (High / Normal), wiatr (Strong / Weak)

E) plik wine_quality.csv - decyzją jest zmienna 'quality'. Baza zawiera informacje o cechach win, na podstawie których przypisuje się im stopień jakości. Zawiera następujące zmienne numeryczne: stała kwasowość, lotna kwasowość, zawartość kwasu cytrynowego, zawartość cukru resztkowego, zawartość chlorków, wolny dwutlenek siarki, całkowity dwutlenek siarki, gęstość, pH, zawartość siarczanów, stężenie alkoholu

F) plik zoo.csv - decyzją jest zmienna 'animal name'. Baza zawiera informacje o cechach szczególnych zwierząt, na podstawie których można odgadnąć dane zwierze. Zawiera następujące dane: królestwo (Mammal / Bird / Reptile / Fish / Reptile / Bug / Invertebrate), liczba nóg (numeryczne), a także zmienne typu True / False: ma włosy, ma pióra, składa jaja, produkuje mleko, jest zwierzęciem lotnym, jest zwierzęciem wodnym, jest drapieżnikiem, ma zęby, ma kręgosłup, oddycha, jest jadowite, ma płetwy, ma ogon, jest domowe, jest rozmiaru kota

Paweł Wysocki