Inteligencja obliczeniowa – stud.niestacj.

Laboratorium 4: Czyszczenie danych.

Prawie zawsze w większych bazach danych są jakieś błędy i nieścisłości. Wymagane jest wówczas czyszczenie (data cleansing/cleaning) i obróbka. Z reguły najpierw trzeba naprawić bazę danych (znaleźć i usunąć błędy w jej strukturze, złe rekordy, złe dane lub ich zakres czy typ). Nastepnie mając działającą bazę, należy sprawić by była spójna i logiczna (np. zająć się brakującymi danymi).

Na dzisiejszych laboratoriach dalej zamjmujemy się bazą danych z irysami, ale ściągniemy jej „popsutą wersję" (`dirty_iris.csv`, w załączniku) i postaramy się ją naprawić za pomocą różnych narzędzi diagnostyczno-naprawczych.

W zadaniach będziemy korzystać z przystępnie napisanego samouczka o czyszczeniu danych dostępnego pod adresem:
https://cran.r-project.org/doc/contrib/de_Jonge+van_der_Loo-Introduction_to_data_cleaning_with_R.pdf
Zapoznaj się zwłaszcza z rozdziałem 3.1.4.

Zadanie 1 (Znajdujemy błędy)
Wczytaj bazę z irysami `dirty_iris.csv` (załączone). Następnie wykonaj polecenia:

a) Załaduj bazę danych do R.
```
dirty.iris <- read.csv("dirty_iris.csv", header=TRUE, sep=",")
```
Wyświetl ją i przyjrzyj się rekordom. Sporo rekordów ma brakujące dane (NA = Not Avaialable). Policz ile ze wszystkich 150 rekordów jest pełnych (ma wszystkie dane). Wykorzystaj do tego funkcje:
`nrow` – liczenie wierszy w tabeli
`subset` – filtrowanie tabeli
`is.finite` – sprawdzanie, które dane są liczbami skończonymi
Poprawna odpowiedź: 95.

b) Chcemy stworzyć zestaw reguł, które sprawdzą czy tabela z irysami jest poprawna. Wykorzystamy do tego paczkę `editrules` (patrz: https://cran.r-project.org/doc/contrib/de_Jonge+van_der_Loo-Introduction_to_data_cleaning_with_R.pdf strona 35, 36). Zainstaluj i załaduj tę paczkę i dodaj regułę, że długość działki kielicha nie może być dłuższa niż 30 cm:

```
install.packages("editrules")
library(editrules)
E <- editset(c("Sepal.Length <= 30"))
E
```

Następnie sprawdzamy ile wierszy tabeli nie spełnia tej reguły.
```
ve <- violatedEdits(E, dirty.iris)
ve
```

Można wyświetlić też :
```
summary(ve)
plot(ve)
```

c) Popraw editset `E`, tak aby miał dodatkowo następujące reguły (możesz je wpisać ręcznie w

komendzie, lub wczytać z pliku komendą `editfile`).

- Ostatnia kolumna zawiera tylko wartości: setosa, versicolor, virginica.
- Wszystkie numeryczne wartości muszą być dodatnie.
- Petal.Length musi być minimum dwa razy większe niż Petal.Width
- Sepal jest dłuższy Petal.

d) Sprawdź za pomocą komendy `violatedEdits`, ile każda z reguł została złamana. Ile danych jest całkowicie poprawnych? Które irysy mają za długie płatki względem ich szerokości?

Zadanie 2 (Usuwamy błędy)
Wykryliśmy błędy, a teraz pora je poprawić. Jeszcze raz wróć do samouczka
https://cran.r-project.org/doc/contrib/de_Jonge+van_der_Loo-Introduction_to_data_cleaning_with_R.pdf
i przejdź do rozdziału 3.2.

Korzystając z paczki `deducorrect` (opisana w podrozdziale 3.2.1).

- Zamień niedodatnie wartości z `Petal.Width` na wartość `NA` korzystając z komendy `correctWithRules`.
- Zastąp wszystkie inne nieporawne dane wykryte w zadaniu 1 etykietką NA.

Zadanie 3 (Wypełniamy luki z błędami sensownymi danymi)
Pozbyliśmy się błędnych danych, ale zostaliśmy z etykietkami NA. Byłoby dobrze, gdybyśmyw ich miejscu mieli liczby. Bezsensowne byłoby wpisywanie jakichkolwiek liczb, ale jest kilka technik, które przynajmniej w przybliżeniu uzupełniają je dobrymi wartościami. Przetestujemy dwie z nich.

a) Wszystkie puste dane z danej kolumny zastąp wartością średnią z kolumny (rozdziałó 3.3.1).
Zastosuj tę technikę i zapisz wyniki w tabeli `clean.iris.mean`.

b) Najbliżsi sąsiedzi (k-Nearest Neighbor) – rozdział 3.3.3. Gdy irys ma brakujące dane, szukamy k najbardziej podobnych irysów do niego, mających wartości najbardziej zbliżone. Na ich podstawie wyliczana jest brakująca wartość.
Zastosuj tę technikę i zapisz wyniki w tabeli `clean.iris.knn`.