Projekt 1 DATA MINING 2020

Celem projektu jest implementacja metody selekcji zmiennych opartej na warunkowej informacji wzajemnej CMIM (Conditional Mutual Information Maximization). Dokładny opis zadania znajduje się na końcu tekstu.

Specyfikacja:

- Program zapisujemy w pliku *nazwisko.R* lub *nazwisko.py* bez polskich znaków. W środku umieszczamy własną implementację zadanego algorytmu wraz z ewentualnymi funkcjami pomocniczymi. W pierwszej linijce kodu umieszczamy informację, czy zaimplentowany algorytm jest w wersji normalnej (wtedy proszę napisać *normalny*), czy szybkiej (*szybki*). W kolejnych liniach kodu umieszczamy ładowanie potrzebnych pakietów, a następnie implementacje funkcji głównej i ewentualnie funkcji pomocniczych. Szablon dla R znajduje się na stronie. Proszę zwrócić uwagę w jaki sposób należy załadować wymagane pakiety.
- Zakładamy, że w danych nie ma braków danych. Wektor odpowiedzi y jest dyskretną zmienną objaśniającą. Każda kolumna ramki danych X odpowiada zmiennej objaśniającej. Wszystkie zmienne objaśniające są dyskretnymi zmiennymi numerycznymi. Zakładamy, że liczba zmiennych jest większa niż 1.
- Liczba naturalna *kmax* ma być mniejsza bądź równa liczbie zmiennych objaśniających.

Opis projektu

Poniższy opis opiera się na założeniach poczynionych w specyfikacji. Należy sprawdzać, czy zmienne przekazywane zaimplementowanej funkcji spełniają wymagania.

Należy zaimplementować funkcję *CMIMselection(X, y, kmax)* gdzie:

- X - macierz, której i-ty wiersz jest wektorem wartości zmiennych objaśniających dla i-tej obserwacji (wymiaru n na p),
- y - wektor odpowiedzi, zmienna grupująca przyjmująca skończenie wiele wartości, której i-ta wartość odpowiada i-tej obserwacji (o długości n),
- *kmax* - liczba naturalna oznaczająca maksymalną liczbę kroków, któe ma wykonać algorytm.

Funkcja ma zwracać listę zawierającą następujące elementy:

- S - wektor o długości *kmax* zawierający indeksy wybranych zmiennych,
- *score* - wektor o długości *kmax* zawierający wartości kryterium dla wybranych zmiennych.

W implementacji nie można korzystać z gotowych funkcji wyznaczających wartość kryterium CMIM!!!

Ocena projektu będzie uwzględniała trzy składowe:

- poprawność algorytmu (8pkt)
- czas działania funkcji (8pkt)
- obsługa błędów i wyjątków oraz przejrzystość kodu (4pkt)

Czas działania funkcji zostanie przetestowany na zbiorze, który nie jest udostępniony. Za wersję zwykłą algorytmu można dostać maksymalnie 4pkt, za wersję szybką 8pkt.

Spawdzenie poprawności odbędzie się poprzez wykonanie poleceń (na ostatniej stronie można znaleźć wyniki działania poniższych funkcji):

```
library(mlbench)
data("BreastCancer")
```

```r
obserwacje_bez_NA <- complete.cases(BreastCancer)
X <- BreastCancer[obserwacje_bez_NA , -c(1, 11)]
y <- BreastCancer[obserwacje_bez_NA , 11]

CMIM_sel <- CMIMselection(X=X, y=y, kmax=9)

CMIM_sel$S
CMIM_sel$score
```

Dodatkowo działanie funkcji zostanie przetestowane na jeszcze na innych zbiorach danych, których nie podajemy.

Spawdzenie czasu działania odbędzie się poprzez uśrednienie czasu działania funkcji na wybranym zbiorze danych. Uwzględniane będą tylko poprawne implementacje.

Projekty realizowane są samodzielnie! Niesamodzielne prace skutkują zerową liczbę punktów!

Projekty należy dosłać do wtorku 21.04.2020, do godziny 23:59 na adres m.lazecka@mini.pw.edu.pl.

Opis metody

Entropia i informacja wzajemna

Metoda CMIM oparta jest na teorioinformacyjnych miarach zależności pomiędzy zmiennymi. Podstawową miarą zależności pomiędzy dwiema zmiennymi dyskretnymi X i Y o nośnikach odpowiednio $\mathcal{X}$ i $\mathcal{Y}$ jest informacja wzajemna wyrażająca się wzorem

$$I(X,Y) = \sum_{x\in\mathcal{X}, y\in\mathcal{Y}} p_{X,Y}(x,y) \log \frac{p_{X,Y}(x,y)}{p_X(x)p_Y(y)},$$

gdzie gdzie $p_{X,Y}$ jest gęstością łączną zmiennych X i Y, zaś p_X, p_Y odpowiednimi gęstościami brzegowymi. Informację wzajemną zmiennych X i Y można zapisać również jako różnicę entropii

$$I(X,Y) = H(X) - H(X|Y),$$

gdzie $H(X)$ oznacza entropię zmiennej losowej X, zaś $H(X|Y)$ oznacza entropię warunkową X pod warunkiem, że znamy Y i wyraża się wzorem

$$H(X|Y) = -\sum_{x\in\mathcal{X}, y\in\mathcal{Y}} p_{X,Y}(x,y) \log p_{X|Y}(x,y)$$

Intuicyjnie informacja wzajemna mówi nam, o ile zostanie zredukowana niepewność zmiennej X, jeśli będziemy znać zmienną Y. Informacja wzajemna jest równa 0, gdy zmienne są niezależne, zaś wartość maksymalną równą $H(X)$ osiąga dla $X = Y$, ponieważ wtedy w Y zawarta jest pełna informacja o X.

Analogicznie definiujemy warunkową informację wzajemną:

$$I(X,Y|Z) = H(X|Z) - H(X|Y,Z),$$

gdzie X i Y są jak wyżej, a Z jest dyskretną zmienną o nośniku $\mathcal{Z}$. Warunkowa informacja wzajemna jest miarą mówiącą o tym, o ile znajomość zmiennej Y redukuje niepewność X przy danej zmiennej Z.

Do liczenia informacji wzajemnej i warunkowej informacji wzajemnej można wykorzystać pakiet `infotheo`.

Więcej informacji można znaleźć w książce: Cover, T. M. and Thomas, J. A. (1990). *Elements of Information Theory.* John Wiley, New York.

CMIM

CMIM jest krokową metodą selekcji, tzn. zaczynamy od pustego zbioru indeksów wybranych zmiennych S i w każdym kroku będziemy do niego dołączać jedną zmienną. Niech Y będzie zmienną objaśnianą, a X_i dla $i = 1, 2, \ldots, p$ zmiennymi objaśniającymi. S_k oznaczać będzie zbiór indeksów wybranych zmiennych do kroku k włącznie z k-tym, $\nu(k)$ indeks zmiennej wybranej w k-tym kroku.

- Krok 1

$$\nu(1) = \arg\max_{i \in \{1,2,\ldots,p\}} \hat{I}(Y, X_i)$$
$$S_1 = \nu(1)$$

- Krok k-ty

$$\nu(k) = \arg\max_{i \in \{1,2,\ldots,p\} \setminus S_{k-1}} \min_{j \in S_{k-1}} \hat{I}(Y, X_i | X_j)$$
$$S_k = S_{k-1} \cup \nu(k)$$

Minimalizację warunkowej informacji wzajemnej $\hat{I}(Y, X_i|X_j)$ po $j \in S_{k-1}$ można rozumieć w ten sposób, że szukamy dla X_i (czyli ustalonego kandydata do zbioru S_k) takiej zmiennej X_j spośród już wybranych, która sprawia, że informacja jaką o Y wnosi X_i przy założeniu, że znamy X_j jest najmniej istotna. Następnie chcemy wybrać taką zmienną X_i, której minimalna warunkowa informacja wzajemna będzie największa.

Więcej informacji można znaleźć w artykule: *Fast Binary Feature Selection using Conditional Mutual Information Maximisation* F. Fleuret, JMLR (2004).

CMIM wersja szybka

Dokładny opis znajduje się w artykule podanym wyżej na stronach 1540-1541.

Wynik przykładowego wywołania

```
CMIM_sel$S
```

```
## [1] 2 6 1 8 7 5 3 4 9
```

```
round(CMIM_sel$score, 4)
```

```
## [1] 0.4868 0.1000 0.0769 0.0708 0.0631 0.0523 0.0456 0.0428 0.0192
```