Przetwarzanie i analiza danych w systemie SAS®

Kolokwium nr 2 - 26.01.2015

- Zapisać plik `szablon.sas` pod nazwą `nazwisko.sas` (gdzie `nazwisko` to nazwisko piszącego kolokwium). Na początku pliku należy wpisać w komentarzu własne imię i nazwisko.
- Rozwiązania zadań należy wpisywać do pliku `nazwisko.sas`. Plik należy we własnym interesie często zapisywać.
- Rozpakowane pliki z danymi wejściowym mają być umieszczone w bibliotece o nazwie *KOLO*.
- Wszelkie tworzone przez siebie zbiory sasowe należy umieszczać w bibliotece *WORK*.
- Tworzone programy powinny działać poprawnie bez żadnych zmian dla dowolnych zbiorów o takiej samej strukturze (tj. o takich samych zmiennych i ich atrybutach), jak wymienione w treści zadań. W szczególności, rozwiązania będą testowane na zbiorach różnych od podanych.
- Po zakończeniu pracy, należy wpisać do górnego paska Eksploratora Windows nazwę katalogu

  `\\secundus\upload\sas`

  a następnie skopiować i wkleić do tego katalogu stworzony przez siebie podczas rozwiązywania zadań plik `nazwisko.sas` (tylko ten plik, bez tworzonych zbiorów sasowych). We własnym interesie należy poczekać na ustne potwierdzenie odbioru pliku.
- Powodzenia!

1. (13 pkt.) Dla danego $n \in \mathbb{N}$ i danej biblioteki *bib*, niech zb_1, zb_2 oznaczają nazwy dowolnych zbiorów z *bib* liczących dokładnie n obserwacji, a zm_i niech oznacza dowolną zmienną numeryczną ze zbioru zb_i, $i = 1, 2$. Napisać dwuparametrowe makro $\%z1(bib, n)$, które dla wszystkich $(zb_1, zm_1) \neq (zb_2, zm_2)$ obliczy normę l_1 różnicy zmiennych $zm_1 - zm_2$ traktowanej jako wektor z $\mathbb{R}^n$ i zapisze ją do zmiennej o nazwie *norma* w zbiorze **WORK.normy** . Zbiór **WORK.normy** ma mieć pięć zmiennych o nazwach, kolejno: *zbior_1*, *zmienna_1*, *zbior_2*, *zmienna_2*, *norma*. W każdym wierszu zbioru **WORK.normy**, zmienne *zbior_i* mają przyjmować wartości dane przez zb_i, a zmienne *zmienna_i* wartości dane przez zm_i. Można zakładać, że w zbiorach z biblioteki *bib* nie ma brakujących danych.

2. (12 pkt.) Dane są zbiory o strukturze takiej jak **KOLO.b1**, **KOLO.b2** i **KOLO.grupy**. Powiemy, że *id* ze zbioru **KOLO.b1** lub **KOLO.b2** należy do grupy g (patrz zbiór **KOLO.grupy**), jeśli odległość euklidesowa wektora o współrzędnych równych kolejnym wartościom zmiennych *z1-z20* odpowiadających danemu *id*, od wektora o współrzędnych równym kolejnym wartościom zmiennych *sg1-sg20* odpowiadających danemu g, jest mniejsza niż analogiczna odległość przy jakiejkolwiek innej wartości g.

   Używając języka 4GL znaleźć liczbę tych *id*, które występują zarówno w **KOLO.b1**, jak i w **KOLO.b2**, oraz w obydwu tych zbiorach należą do tej samej grupy. Można założyć, że każde z rozpatrywanych id należy do tylko jednej grupy. Można także założyć, że liczba grup jest znana.

3. (8 pkt) Dane są trzy zbiory sasowe o strukturze takiej jak **KOLO.klienci**, **KOLO.auta** i **KOLO.wypozyczenia** (patrz spakowany plik). W szczególności zakłada się, że zmienne o nazwach zaczynających się od *id* jednoznacznie identyfikują klientów, samochody i wypożyczenia.

   Używając języka SQL znaleźć, o ile istnieją, marki tych samochodów, które były wypożyczane przez co najmniej połowę klientów wypożyczalni.

4. (7 pkt.) Załóżmy, że macierz $A \in \mathbb{R}^{4\times 4}$ i wektor $b \in \mathbb{R}^4$ dane są w postaci zbiorów sasowych o strukturze takiej, jak **KOLO.macierz** i **KOLO.wektor** (patrz spakowany plik). Napisać program, który, w przypadku gdy A jest odwracalna, zapisze rozwiązanie równania $Ax = b$ do zbioru sasowego. Jeśli A nie jest odwracalna, program ma znaleźć wartość własną A o największej co do modułu części rzeczywistej.