

# Drugi Projekt Labowy

## Metody Numeryczne 2022/2023, grupy 1 i 2

Termin oddania: do 22 grudnia 2022 włącznie

W ramach swoich obowiązków służbowych Syzyf musi rozwiązywać wiele Liniowych Zadań Najmniejszych Kwadratów postaci

$$\min_x \|A\mathbf{x} - \mathbf{b}\|_2^2, \quad (\star)$$

gdzie  $A$  jest kolumnowo regularną macierzą, a  $\mathbf{x}$  i  $\mathbf{b}$  są oczywiście wektorami o stosownych wymiarach. Przez długi czas jego praca była o tyle monotonna, że macierz  $A$  pozostawała taka sama, w zadaniu  $(\star)$  zmieniały się jedynie wektory  $\mathbf{b}$ . Ostatnio jednak przełożeni Syzyfa zaczęli często dodawać kolumny do jego macierzy, a czasem nawet (choć sporadycznie) te kolumny mu zabierają. Bardzo to Syzyfa frustruje, gdyż wszystko musi wtedy przeliczać od nowa, a to długo trwa. Syzyf chciałby efektywnie radzić sobie z takimi sytuacjami i zwrócił się do Ciebie z prośbą o pomoc.

Stwórz w Pythonie klasę LZNK, implementującą metody przedstawione poniżej. Przy opisie wymagań kosztowych zakładamy, że liczba wykonań operacji dodawania i odejmowania kolumn jest  $O(m)$ , w związku z czym „co do rzędu” wymiary macierzy pozostają bez zmian.

- `__init__(A:ndarray)->LZNK`  
Konstruktor. Powinien działać w czasie  $O(nm^2)$ , gdzie  $n$  i  $m$  ( $n \geq m$ ) to wymiary (odpowiednio liczba wierszy i kolumn) macierzy  $A$  (reprezentowanej jako `numpy.ndarray`).
- `addcol(self, c:ndarray)`  
Metoda aktualizuje obiekt klasy LZNK i ma odpowiadać dodaniu kolumny  $c$  na koniec „obecnej” macierzy  $A$ . Koszt aktualizacji powinien być  $O(nm)$ .
- `delcol(self, i:int)`  
Metoda aktualizuje obiekt klasy LZNK i ma odpowiadać usunięciu  $i$ -tej kolumny z „obecnej” macierzy  $A$ . Koszt aktualizacji powinien być  $O(m^2)$ .
- `lstsq(self, b:ndarray)->ndarray`  
Metoda rozwiązuje LZNK  $(\star)$  z „obecną” macierzą  $A$  (kolumnowo pełnego rzędu) i zadany wektorem  $\mathbf{b}$ . Powinna działać w czasie  $O(nm)$  i być możliwie stabilna numerycznie (w szczególności, preferowane jest rozwiązywanie Liniowego Zadania Najmniejszych Kwadratów poprzez skorzystanie z rozkładu QR, a nie np. układu równań normalnych – ta preferencja znajdzie odzwierciedlenie w punktacji).

Odnosnie do kosztu pamięciowego, rozwiązanie nie powinno przekraczać (co do rzędu) kosztu trzymania samej macierzy  $A$  (tzn. ma być  $O(nm)$ ). Można korzystać wyłącznie z funkcji wbudowanych w Pythona, jak również tych dostępnych w pakiecie `numpy`. Skorzystanie z innych pakietów może (nie musi) skutkować zmniejszeniem punktacji (jeśli dany pakiet ułatwiałby część czysto implementacyjną, to najpewniej sprawdzający nie będzie miał nic przeciwko – dla pewności, warto to wcześniej skonsultować, najlepiej na forum zadania). Opisanie wyżej metody nie muszą sprawdzać poprawności przyjmowanych argumentów (np. pełnego rzędu kolumnowego macierzy  $A$  czy zgodności wymiarów).

Do pliku tekstowego zawierającego implementację opisaną wyżej klasy (oraz wszystkich pomocniczych klas i funkcji, jeśli takie będą wykorzystywane) należy dołączyć plik `pdf` z (choćby zgrubnym) opisem rozwiązania (i uzasadnieniem spełniania przedstawionych wymogów kosztowych). Rzecz jasna, projekt ma bardzo duży związek z zadaniem 2. z ostatniej pracy domowej i tym bardziej, przynajmniej w kwestii operacji dodawania kolumn, opis może być powierzchowny.

## Ocena

Za zadanie można otrzymać 5 punktów. Ta pula jest rozdzielana między konkretne metody w następujący sposób:

- `__init__()`: 1 punkt
- `addcol()`: 2 punkty
- `delcol()`: 2 punkty

W każdym z powyższych pod uwagę brany jest wymagany koszt obliczeniowy i to, czy nie naruszany jest narzut pamięciowy ( $O(nm)$ ). Jeśli po wywołaniu danej metody nie działa poprawnie metoda `lstsq()` (co będzie weryfikowane przy użyciu losowych macierzy  $A$ , wektorów  $\mathbf{b}$  oraz `numpy.linalg.lstsq`), za daną część nie uzyskuje się punktów.

W stosunku do powyższego opisu, sprawdzający zastrzega sobie prawo zmniejszenia oceny o najwyżej punkt (z powodów, które będzie musiał wyjaśnić w komentarzu); w grę wchodzi również zmniejszanie o ujemną liczbę punktów.