## Lab7C (5 p, pytanie 1p) 14.01.2022

Termin odesłania **21.01.2022 godz. 14.15** na platformie **Ms Teams** ( we właściwym zespole **lab** przypisanym dla przedmiotu **Programowanie Matematyczne** ). **Opóźnione** przesłanie rozwiązania zadania będzie rozliczane zgodnie z regulaminem przedmiotu.

**Rozwiązanie zadania** tj. wszystkie źródłowe **m-pliki, raport** ( *obowiązkowy, zawierający oświadczenie o samodzielności* ) w formacie **zip** o nazwie `pm7c_swojeimie_swojenazwisko.zip`

---

**Raport** (plik **pdf**) powinno być w formacie **A4** i powinno obejmować:
Dane studenta (imię, nazwisko, grupa, data)
Treść zadania (postać rozwiązywanego problemu)
Opis kroków przekształcania zadania, krótki opis algorytmu
Ciekawe przykłady obliczeniowe ( również dodatkowo wskazane w treści zadania )
Analizę (omówienie) wyników obliczeniowych, testów
Ponadto należy załączyć:
Kody źródłowe wszystkich funkcji/procedur i skryptów (**brak** kompletu jest traktowany jak **brak** przesłania zadania w terminie)

---

Napisz **skrypt**, w którym proszę wykonać całe zadanie i wywołać odpowiednie funkcje.

Zaproponuj plik **dane.m**, który utworzy tablicę $\boldsymbol{P} = [\boldsymbol{p^1}, \boldsymbol{p^2}, \dots, \boldsymbol{p^m}]$ losowo wygenerowanych $\boldsymbol{m}$ punktów (w $R^3$), kolejne **kolumny** tablicy zawierają współrzędne wylosowanych punktów ( tablica $\boldsymbol{P}$ wymiaru ***3xm***, przykładowe **losowe dane** mogą zawierać wylosowane punkty o współrzędnych. z [-200, 200], *może inne?* ).
Funkcja zwraca również ***x0*** dla wygenerowanych danych, jest to **punkt startowy** dla własnego algorytmu ***IPM***.

**Problem minimalnej kuli zawierającej punkty P**

**ZP1**

Znaleźć parametry optymalnej kuli: promień $r \in R$ oraz środek $s \in R^3$, co można zapisać w postaci zadania:

$$\begin{cases} \min\limits_{x=(s,r)} r \\ \|p^i - s\| \le r, \quad i = 1, \dots, m \end{cases}$$

**ZP2**

$z = r^2$

Problem można zapisać w postaci:

$$\begin{cases} \min_{x=(s,r)} z \\ \|p^i - s\|^2 \le z, \quad i = 1, \dots, m \end{cases} \qquad \text{czyli} \qquad \begin{cases} \min_{x=(s,r)} z \\ (p^i)^T p^i - 2(p^i)^T s + s^T s \le z, \quad i = 1, \dots, m \end{cases}$$

**ZD**

**Problem ZP2** można rozwiązać, sprowadzając do rozwiązania **ZD** ( ***zadania programowania kwadratowego*** ):

$$\max_{y \in \Omega_y} \left( -\boldsymbol{y}^T \boldsymbol{P}^T \boldsymbol{P} \boldsymbol{y} + \sum_{i=1}^m \boldsymbol{p}_i^T \boldsymbol{p}_i \boldsymbol{y}_i \right)$$

$$\Omega_y: \begin{cases} \sum_{i=1}^m \boldsymbol{y}_i = \boldsymbol{1} \\ \boldsymbol{y} \ge \boldsymbol{0} \end{cases}$$

Dla $\bar{\boldsymbol{y}}$ będącego RO, $\boldsymbol{s} = \boldsymbol{P}\bar{\boldsymbol{y}} = \sum_{i=1}^m \boldsymbol{y}_i \boldsymbol{p}_i$, natomiast $\boldsymbol{r} = \sqrt{\boldsymbol{fval}}$, gdzie *fval* – wartość funkcji **ZD**.

**1 pkt**

Zdefiniuj odpowiednie macierze dla **funkcji celu** oraz dla **ograniczeń**.
Rozwiązać problem **ZD** za pomocą funkcji `quadprog` dla różnych plików danych (**m=2,3,10,100,200,** ***więcej?***)
W **optimoptions** ustaw:
**ConstraintTolerance: 1.0000e-10**
**OptimalityTolerance: 1.0000e-10**

Gdy **exitflag** jest równe **1**, to podaj znaleziony wektor **y** , oraz wyznacz *środek kuli* **s**, *promień* **r** .
Wywołaj: **rysuj3d(P,s,r)**

**1 pkt** ***(opis w raporcie)***

- Napisz funkcję Lagrange'a $\boldsymbol{L}(\boldsymbol{s}, \boldsymbol{z}, \lambda)$ dla **ZP2** ( $\lambda$ - mnożniki Lagrange'a )
- Napisz komplet **WKT** dla **ZP2**, zaobserwuj jakie zależności z nich wynikają?
- Jeśli **ZP2** odpowiada postaci: $\mathbf{min}_{(s,z)}\, \mathbf{max}_{\lambda \geq 0}\, \boldsymbol{L}(\boldsymbol{s}, \boldsymbol{z}, \lambda)$, to uzasadnij podaną powyżej postać **ZD** przekształcając je do $\mathbf{max}_{\lambda \geq 0}\, \mathbf{min}_{(s,z)}\, \boldsymbol{L}(\boldsymbol{s}, \boldsymbol{z}, \lambda)$. Czym są zmienne $\boldsymbol{y}$ w **ZD**?
  **Uzasadnij dokładnie**: skąd otrzymujemy wzór **maksymalizowanej funkcji** w **ZD**, skąd otrzymujemy **ograniczenia** w **ZD**, skąd wynikają wzory na wyznaczenie $\boldsymbol{s}$ oraz $\boldsymbol{r}$.

**3 pkt**

Proszę rozwiązać **problem ZD** *(i porównać z powyższymi wynikami* ***quadprog****-a)* za pomocą **własnej funkcji** wykorzystującej **algorytm punktu wewnętrznego** ***IPM*** dla odpowiedniego **zadania kwadratowego**

W raporcie (***obowiązkowo***) należy podać **opis algorytmu** wykorzystanego w **swojej implementacji** oraz **uzasadnić** wszystkie podejmowane **kroki**.
Uzasadnij, jak wyznacza się **kierunek**, jak wyznacza się **krok** oraz **kluczowe parametry** (np. punkt startowy, warunek stopu, itp).

**`[RO, f_opt, exitflag, it, LL_eqlin, LL_lower]=IPM(parametry);`**
**`e=?`** parametr definiujący dokładność obliczeń ( zbieżność, itp.), np**. e=1e-6?**
**`x0`** – punkt startowy
**`MAX_IT = ?`**
**`M = ?`**

**`it`** – liczba iteracji
**`LL`** – obliczone **mnożniki Lagrange'a** ( *które zmienne w algorytmie IPM stanowią mnożniki Lagrange'a?*)

Jeśli istnieje rozwiązanie zadania (**`exitflag=1`**), to zbadaj skuteczność algorytmu (*może wykres?):*
**`norm(x-RO)`**

Jeśli istnieje rozwiązanie zadania (**`exitflag=1`**), to wyświetl **mnożniki Lagrange'a** oraz **porównaj** je z uzyskanymi z **`quadprog`**.

**TESTY**

- Należy przeprowadzić **testy algorytmu** dla różnych wartości $\boldsymbol{m}$ (*jak duże?)* w celu zbadania liczby iteracji, której wymaga algorytm, by uzyskać rozwiązanie z podaną dokładnością.
- Oczekiwana jest **wysoka skuteczność** własnej implementacji IPM (dopracuj parametry algorytmu).

**Opis testów**
**Wnioski**