



Optimalizace (DÚ3)

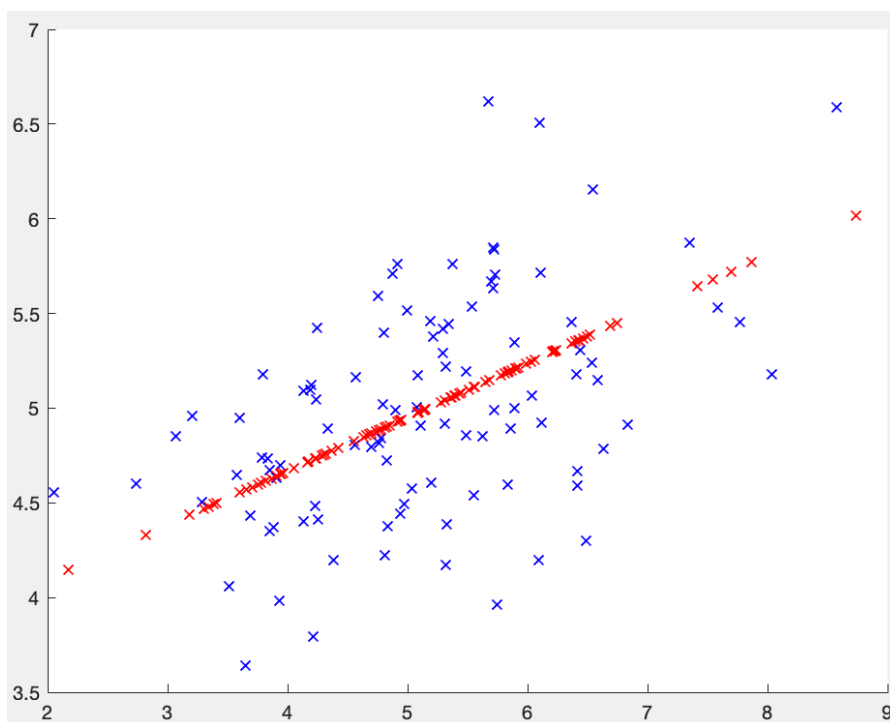
1. Část: Aproximace bodů přímkou

Úkol 1.1)

Zadání úlohy: Zobrazte do jednoho obrázku zadané body a_1, \dots, a_m (modře) a jejich kolmé projekce $\tilde{a}_1, \dots, \tilde{a}_m$ na nalezenou přímku (červeně).

Výstup: graf.

Řešení: obr. 1



obr. 1

Úkol 1.2)

Zadání úlohy: Jaký je součet čtverců kolmých vzdáleností bodů k nalezené přímce?

Výstup: číslo.

Řešení:

Součet čtverců kolmých vzdáleností bodů k nalezené přímce lze spočítat pomocí jednoduchého vzorce pro vzdálenost dvou bodů

$$|AB| = \sqrt{(x_B - x_A)^2 + (y_B - y_A)^2}.$$

Tento vzorec použijeme pro matici původních bodů a matici projekcí na přímku. Poté vše sečteme. Součet tedy výjde

$$sum_squares = 24.2419.$$

Úkol 1.3)

Zadání úlohy: Nalezněte požadovanou přímku ve dvou různých reprezentacích:

$$\{y \in \mathbb{R}^2 \mid y^T x = \alpha\} = \{y_0 + ts \mid t \in \mathbb{R}\}$$

Výstup: vektory $x \in \mathbb{R}^2, y_0 \in \mathbb{R}^2, s \in \mathbb{R}^2$ a číslo $\alpha \in \mathbb{R}$.

Řešení:

Vektory x (normálový vektor přímky) a s (směrový vektor přímky) dostáváme přímo z matice V po rozkladu SVD, kde

$$s = \begin{bmatrix} -0.9617 \\ -0.2740 \end{bmatrix}$$

a

$$x = \begin{bmatrix} -0.2740 \\ 0.9617 \end{bmatrix}.$$

Dále vektor y_0 získáme jako

$$y_0 = matice_strednich_hodnot^T - projekce * matice_strednich_hodnot^T$$

a y_0 tedy vyjde

$$y_0 = \begin{bmatrix} -0.9296 \\ 3.2633 \end{bmatrix}.$$

Nakonec vypočítáme α , kterou získáme jako normálový vektor krát bod na přímce (použijeme například již vypočítanou projekci bodu $[0,0]$ na přímku, tedy y_0), tedy

$$\alpha = -3.3931.$$

2. Část: Komprese sekvence z motion capture

Úkol 2.1)

Zadání úlohy: Minimalizujte kritérium (1 - viz. zadání) za podmínky, že body $\tilde{a}_1, \dots, \tilde{a}_m$ leží v afinním podprostoru dimenze r . Výsledkem bude matice \tilde{A} s řádky $\tilde{a}_1^T, \dots, \tilde{a}_m^T$. Provedte pro sekvenci “Chůze” a pro pět různých hodnot $r \in \{1, 2, 5, 10, 15\}$.

Výstup: tabulka s optimálními hodnotami pro zadané hodnoty.

Řešení:

r	$\sum_{i=1}^m \tilde{a}_i - a_i ^2$
1	$4.6166 \cdot 10^8$
2	$1.6925 \cdot 10^8$
5	$1.0453 \cdot 10^7$
10	$1.1982 \cdot 10^6$
15	$2.5626 \cdot 10^5$

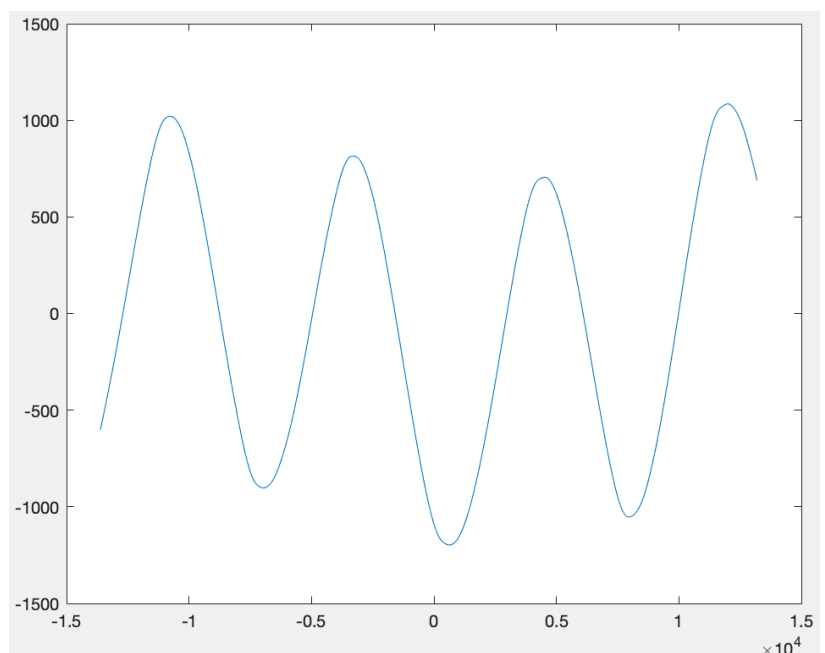
Úkol 2.2)

Zadání úlohy: Výsledné body vyjádřete jako lineární kombinaci báze vektorů. Pro $r = 2$ nakreslete sekvenci vektorů y_1, \dots, y_m jako trajektorii v rovině (po sobě jdoucí body spojte usečkou). To samé udělejte i ve třírozměrném prostoru, tedy pro $r = 3$. Provedte pro sekvence “Tanec Makarena” a “Chůze”.

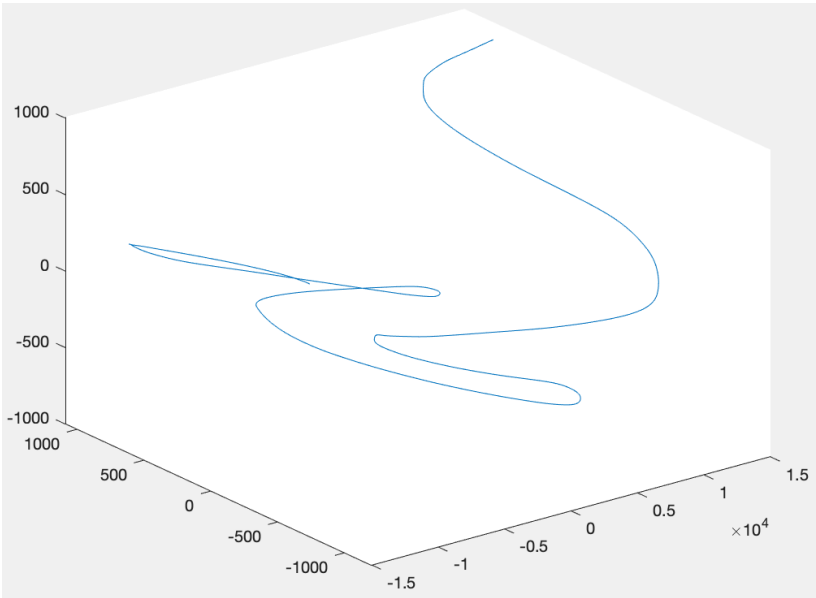
Výstup: dva grafy zobrazující 2D trajektorii a dva grafy zobrazující 3D trajektorii.

Řešení: obr. 2 (chůze 2D), obr. 3 (chůze 3D), obr. 4 (makarena 2D), obr. 5 (makarena 3D)

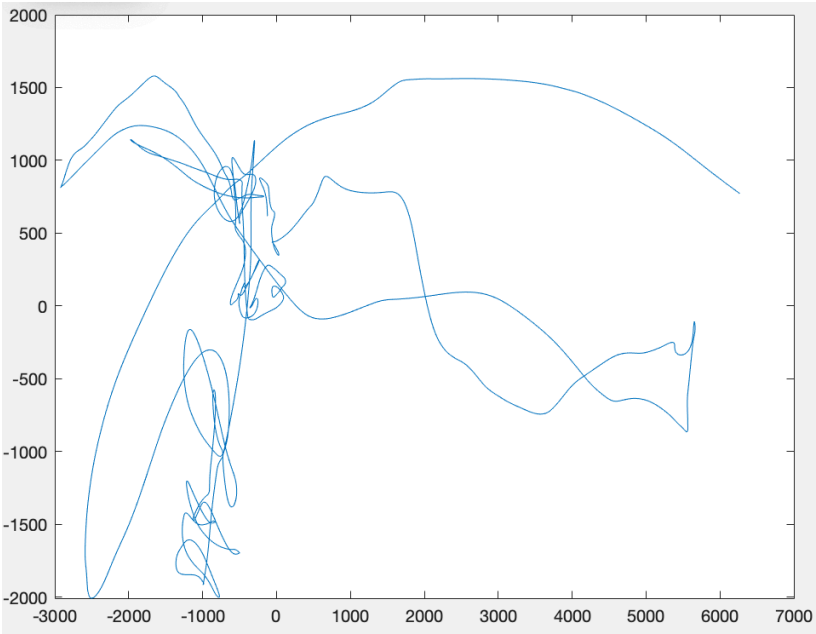
obr. 2 (chůze 2D)



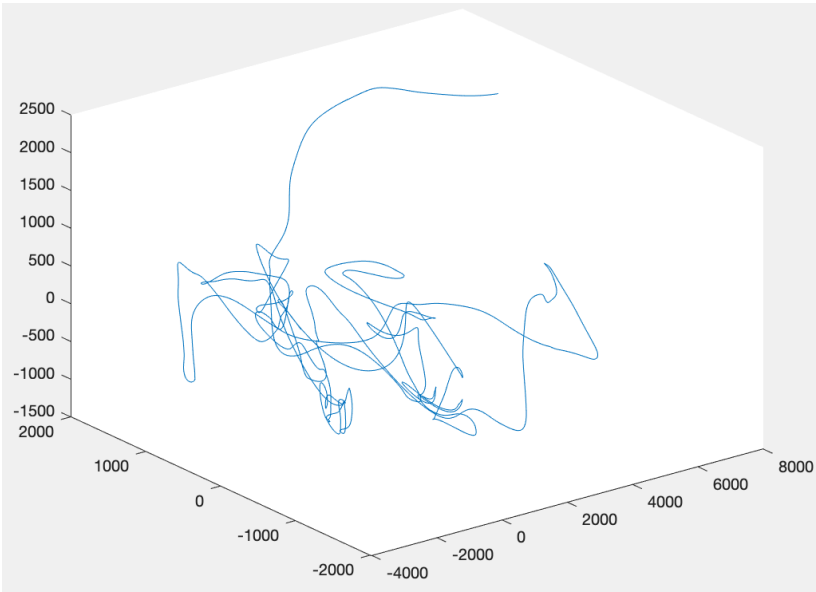
obr. 3 (chůze 3D)



obr. 4 (makarena 2D)



obr. 5 (makarena 3D)



Úkol 2.3)

Zadání úlohy: Uvažujte že postva dělá čistý translační pohyb, tj. konfigurace bodů se nemění a jejich souřadnice se pohybují po přímce. Jaká je minimální dimenze podprostoru, aby aproximační chyba byla nulová?

Výstup: číslo a stručné teoretické odůvodnění.

Řešení: K nulové aproximační chybě nám stačí podprostor dimenze **1**, jelikož pohyb probíhá pouze na přímce. Tedy k vyjádření pohybu nám stačí vektor, který bude udávat polohu bodů vůči přímce.

Úkol 2.4)

Zadání úlohy: Dejme tomu, že bychom chtěli spočítat optimální chybu aproximace (1 - viz. zadání) pro různé hodnoty $r \leq n$. Dostali bychom tedy n čísel, z nichž poslední by bylo nulové. Jaký vztah mají tato čísla k singulárním číslům? Jako inspiraci si vykreslete singulární čísla seřazená do grafu.

Výstup: vzorec.

Řešení:

Hodnost původní matice je n . Hodnost aproximované matice je r . Platí tedy $r \leq n$. Když si vypíšeme singulární čísla zjistíme, že se prvních r čísel rovná, tedy suma po r -té singulární číslo rozdílu kvadrátu bude 0. Budeme-li pokračovat v sumaci kvadrátů singulárních čísel původní matice od $r+1$ do n , dostaneme hledanou aproximační chybu:

$$\sum_{i=1}^r ||s_i - \tilde{s}_i||^2 = 0$$

$$\sum_{i=r+1}^n s_i^2 = err.$$