שיטות אלגבריות בהנדסת נתונים – תרגיל בית 4

שאלה 1

תהי $A \in \mathbb{R}^{nxn}$, נתון פירוק ה-SVD הקומפקטי שלה: $A = U_r \Sigma_r V_r^T$. הוכיחו את הטענות הבאות:

1. אם A מוגדרת אי שלילית אז $\forall i = 1,2, \dots ,\dots$ מתקיים: $A^i = U_r \Sigma_r^i U_r^T$.
2. אם A לא סינגולרית (כלומר $rank(A) = n$) אז $A^{-1} = V_r \Sigma_r^{-1} U_r^T$.

שאלה 2

תהי A מטריצה ממשית $m \times n$ ונתבונן בבעית האופטימיזציה הבאה:

$$\max_{B \in R^{m \times n}: \|B\|_2 \leq 1} Tr(AB^\top)$$

נסמן את פירוק ה-SVD הקומפקטי של A: $A = U_r \Sigma_\mathrm{r} V_r^T$.

הוכיחו כי הערך האופטימלי של בעיית האופטימיזציה הנ״ל הוא $Tr(\Sigma_r)$ וכי הוא מתקבל עבור המטריצה $B^* = U_r V_r^\top$.

תזכורת: ראינו בתרגיל בית כי עבור $A \in \mathbb{R}^{mxn}$ מתקיים:

$$\max_{u \in \mathbb{R}^m, v \in \mathbb{R}^n, \|u\|_2 = \|v\|_2 = 1} u^T A v = \sigma_1(A)$$

שאלה 3

יהיו $x_1, \dots, x_m \in \mathbb{R}^n$ נקודות דאטה, כך ש- $\bar{x} = \frac{1}{m} \sum_{i=1}^m x_i \neq 0$.

נסמן ב- $X \in \mathbb{R}^{n \times m}$ את המטריצה $X = (x_1, \dots, x_m)$,

וב-$\tilde{X} \in \mathbb{R}^{n \times m}$ את המטריצה הממורכזת $\tilde{X} = X - \bar{x}$.

נניח כי U_k היא המטריצה המכילה את k הרכיבים העיקריים הראשונים של המטריצה $\tilde{X}$, ואילו V_k היא המטריצה המכילה את k הרכיבים העיקריים הראשונים של המטריצה שאינה ממרוכזת X.

הוכח \ הפרך:

$$U_k U_k^T \tilde{x}_i = z_i - \bar{z}, \qquad \forall i: z_i = V_k V_k^T x_i$$

<u>שאלה 4</u>

שימו לב: לפני תחילת העבודה על תרגיל זה, קראו את המדריך המצורף "מדריך לעבודה עם תמונות בפייתון".

לצורך תרגיל זה, בחרו תמונה צבעונית כלשהי. ניזכר כי תמונה צבעונית מיוצגת על ידי שלוש מטריצות, אחת לכל גוון (אדום, ירוק וכחול).

עבור מספר ערכי k, מצאו את הקירוב הטוב ביותר מדרגה k של המטריצה האדומה, הירוקה והכחולה בהתאמה המרכיבות את התמונה, והציגו את התמונה המתקבלת מקירוב זה. נסו לקבוע מהו ערך ה-k המינימלי עבורו התמונה המתקבלת נראית כמעט זהה לתמונה המקורית.

מה היחס בין הערך המינימלי הנ"ל לבין הדרגה של המטריצות המקוריות?

לכל ערך של k כתבו את השגיאה היחסית (נזכר כי השגיאה היחסית של דירוג מדרגה k של מטריצה A נתונה על ידי: $\frac{\|A-A_k\|_F^2}{\|A\|_F^2}$, כאשר A_k היא הקירוב הטוב ביותר מדרגה k). בחלק זה ניתן להתייחס לשגיאה המתקבלת לכל ערך k עבור אחת ממטריצות הצבע ולא כולן.

האם ערכי השגיאה מסכימים עם ערך ה-*k* המינימלי שמצאתם קודם לכן? כלומר, האם ניתן לראות ירידה חדה בערכי השגיאה בסביבות ה-*k* המינימלי?

יש להגיש:

1. קובץ *py* של הקוד שכתבתם.
2. את התמונות שיצרתם עבור ערכי ה-*k* השונים (לכל תמונה כתבו את ערך ה-*k* ואת ערך השגיאה), כמו גם את התמונה המקורית.
3. את התשובות לשאלות שנשאלו.

<u>שאלה 5</u>

שימו לב: לפני תחילת העבודה על תרגיל זה, קראו את המדריך המצורף "מדריך לעבודה עם תמונות בפייתון", המתאר גם את השימוש בדאטה סט CIFAR10.

<u>אלגוריתם KNN:</u>

נזכר כי k-Nearest Neighbors עובד באופן הבא: נתון סט אימון של m נקודות: $(x_i, y_i) \in \mathbb{R}^n \times \{1,2,..,q\}, i = 1, \dots, m$. כך שלכל דוגמא בסט האימון, x_i הוא וקטור הפיצ'רים, ו-y_i הוא התיוג של הנקודה (אחד מבין q תיוגים אפשריים).

בנוסף, נתון סט מבחן $z_1, \dots, z_l \in \mathbb{R}^n$, עבורו לא קיים תיוג, שאותו נרצה לתייג לאחד מהתיוגים האפשריים.

כעת, בהינתן פרמטר k (הקובע בכמה שכנים להשתמש בתיוג), אלגוריתם *KNN* מוצא את k הנקודות הקרובות <u>בסט האימון</u> ביותר לנקודה נתונה <u>בסט המבחן</u> z_i, נסמן נקודות אלו על ידי: $x_1^{(i)}, \dots, x_k^{(i)}$. כעת, האלגוריתם יתייג את הנקודה z_i לפי התיוג הנפוץ ביותר מבין הנקודות $\left(x_1^{(i)}, y_1^{(i)}\right), \dots, \left(x_k^{(i)}, y_k^{(i)}\right)$.

שגיאת התיוג של האלגוריתם נמדדת על ידי שיעור התיוגים הלא נכונים. כלומר, בהינתן התיוגים האמיתיים $y_1, \ldots, y_m$, והתיוגים שניתנו על ידי האלגוריתם $\widehat{y_1}, \ldots, \widehat{y_m}$ שגיאת התיוג נמדדת על ידי:

$$error \coloneqq \frac{1}{m}\sum_{i-1}^{m} 1_{y_i \neq \widehat{y_i}}$$

<u>שימוש ב-*PCA* עבור *KNN*:</u>
כאמור, אופן הפעולה של אלגוריתם *KNN* מתבסס על חישוב מרחקים בין וקטורי פיצ'רים. כפי שראינו ~~בכיתה~~ בזום, אם נסמן את *s* הרכיבים העיקריים של הדאטה על ידי $u_1, \ldots, u_s$ של מטריצת הדאטה $X = (x_1, \ldots, x_m)$ (כלומר – דאטה בעמודות), ונסמן את המטריצה $U = (u_1, \ldots, u_s)$ אז הטלת הנקודות בסט האימון וסט המבחן על *s* הרכיבים העיקריים נתונה על ידי:

$$\widehat{x_i} = UU^T x_i \in \mathbb{R}^n, i = 1, \ldots, m$$
$$\widehat{z_j} = UU^T z_j \in \mathbb{R}^n, j = 1, \ldots, p$$

ראינו כי עבור ערכי *s* גדולים מספיק יתקיים $\widehat{x_i} \approx x_i$.
בנוסף, ראינו כי מתקיים: $\left\|\widehat{x_i} - \widehat{z_j}\right\|_2 = \left\|U^T x_i - U^T z_j\right\|_2$, כלומר שהמרחק בין הנקודות $\widehat{x_i}, \widehat{z_j} \in \mathbb{R}^n$ זהה למרחק בין הנקודות $U^T x_i, U^T z_j \in \mathbb{R}^s$.

<u>המשימה שלכם:</u>
כתבו קוד בפייתון המקבל את הפרמטרים *k* (מספר השכנים) ו-*s* (מספר הרכיבים העיקריים), ומתייג נקודות לפי אלגוריתם *KNN*, כאשר הוא משתמש ב-5 החלקים הראשונים של *CIFAR10* בתור סט אימון, ובחלק האחרון בתור סט המבחן.

1. המירו את התמונות ל*grayscale*.
2. מצאו את *s* הרכיבים העיקריים של הדאטה לפי *PCA*.
3. הפעילו את אלגוריתם ה-*KNN* על הדאטה לאחר ההטלה כדי לתייג את הנקודות בסט המבחן.
4. חשבו את שגיאת האימון כפי שתוארה לפני כן.

השתמשו במספר ערכי *k* ובמספר ערכי *s* ותייצרו את הטבלה הבאה: לכל ערך *k*, כתבו את שגיאת התיוג על <u>סט המבחן</u> כשמשתמשים בערכי *s* שונים כמו גם כמשתמשים בדאטה המלא ללא *PCA*.

בנוסף, ענו על השאלות הבאות:
האם לדעתכם *PCA* אפקטיבי עבור *KNN*? כלומר, האם קיים ערך $s \ll n$ עבורו שגיאת התיוג דומה לשגיאת התיוג על הדאטה המקורי? האם הערך של *s* תלוי במספר השכנים *k*?

הגישו:
1. את הקוד שכתבתם.
2. את הטבלה שתוארה לפני כן ואת התשובות לשאלות.

<u>דגש למימוש:</u>
יש לממש את האלגוריתם כך שכאשר משתמשים ב-*PCA* עם הפרמטר *s*, כל חישובי המרחק בין נקודות נעשים בסיבוכיות של $O(s)$ ולא ב-$O(n)$.
בנוסף, יש לממש את האלגוריתם *KNN* ולא להשתמש בספריות מוכנות.