קלאסטרינג

KMEANS, CMEANS, GAP STATISTICS

יהלי צופים
נועם סטולרו
מנחה: ד״ר עופר שיר.

תוכן עניינים

מבוא:

ניתוח אשכולות הינו המשימה של קיבוץ אובייקטים לקבוצות, כך שהאובייקטים הנמצאים באותה קבוצה דומים זה לזה יותר מאשר לאובייקטים השייכים לקבוצות אחרות.

קלאסטרינג הינו הלחם והחמאה של תחום ה Data Science.

בתחומים רבים כגון: למידה חישובית, זיהוי תבניות, ניתוח תמונה, איסוף נתונים, ביו אינפורמטיקה ודחיסת מידע, יש כמות אדירה של מידע. על מנת שנוכל לעבוד עם כמות כזו של מידע, עלינו לבצע עיבוד מקדים. אחד הפתרונות הפופולריים הינו – ניתוח אשכולות באופן בלתי מונחה (unsupervised).

מטרה:

מטרת הפרויקט, הינה מימוש אלגוריתמים אשר מציעים פתרון לבעיה המורכבת של ניתוח אשכולות, תוך כדי החלטה על מדד הדמיון לאובייקטים השונים.

את המימושים בחרנו לכתוב בשפת python עקב השימוש ההולך וגובר בשפה ומשום שהשפה הינה שפת קוד פתוח ולכן הקוד שכתבנו יכול להיות לשימוש לכל אחד.

רקע:

מהלך העבודה:

קשיים:

כאשר ניגשנו למטלת מימוש אלגוריתם ה-kmeans, ניצבו בפנינו הקשיים הבאים:

כיצד קובעים דמיון?

בשביל להחליט מה יהיה מדד הדמיון, יש להכיר את המידע אותו הולכים לנתח, קשה מאוד למצוא מאפיינים אשר לפיהם ניתן לקבוע דמיון בצורה חד משמעית לכל האובייקטים.

למשל, בפרחים נרצה אולי להשוות בין המאפיינים הבאים:

- צבעים
- מספר עליי כותרת
- עובי גבעול
- גובה
- שורשים.

ואילו בבני אדם:

- צבע עיניים
- גובה
- משקל
- צבע עור
- מידת נעליים

לאחר שנקבע מדד הדמיון, נדרשת שיטה לקביעת שייכות לקבוצה, ישנם מספר דרכים לקבוע שייכות בין אובייקטים, לדוגמא: מרכז משותף, צפיפות. כל שיטה בעלת יתרונות וחסרונות ובהתאם ייבחר האלגוריתם. כאמור אנו נשתמש בשייכות לפי מרכז משותף, אובייקט ישויך לקבוצה אשר אובייקט המרכז שלה הוא הדומה לו ביותר, כמו כן, המדד הדמיון שלנו יהיה מרחק אוקלידי.

בעיה נוספת הנה מימד נתונים גבוה.

כאשר מימד הנתונים גבוה אי אפשר להציג את המידע באופן ויזואלי, כמו כן זמן העיבוד גדל. ייתכן גם כי ישנם מאפיינים אשר חשיבותם פחותה ורק יכולים להפריע. זו בעיה שהיקפה חורג מגבולות ולכן לא ניכנס אליה, נציין כי השתמשנו באלגוריתם ה PCA (principle component analysis) .

הבעיה האחרונה אשר עלולה לעלות הנה הבעיה של קביעת מספר האשכולות.

כאשר לא ידוע דבר על הנתונים, לא ניתן לקבוע מראש כמה קבוצות קיימות לנתונים. בהמשך נציג פתרון לבעיה זו.

kmeans:

מימוש:

אנו רוצים למצוא k אשכולות באופן איטרטיבי על פי מרחק אוקלידי וצמצום הפונקציה הבאה:

$$J = \sum_{k} \sum_{i=1}^{n} N_{k,i} \cdot \|x_i - \mu_k\|^2$$

כאשר:

$$N_{k,i} = \begin{cases} 1 & , \quad if\ x_i \in C_k \\ 0 & , \quad if\ x_i \notin C_k \end{cases}$$

למעשה הפונקציה הזו הנה סכום המרחקים הפנימיים של כל קבוצה, כמו כן אובייקט יכול לתרום אך ורק לקבוצה אליה הוא שייך.

אופן הפעולה:

1. הכנס k רצוי.
2. הגרל k מרכזים באופן אקראי.
3. שייך כל אובייקט למרכז הקרוב ביותר.
4. חשב מחדש את כל המרכזים.
5. חזור על צעדים 3 ו 4 עד שהמרכזים אינם משתנים.

שלב 4 של האלגוריתם מבוצע בצורה הבאה:

עבור כל קבוצה, נחשב את הממוצע של המאפיינים ולפיו נבחר את האובייקט הקרוב ביותר לממוצע.

kmeans++:

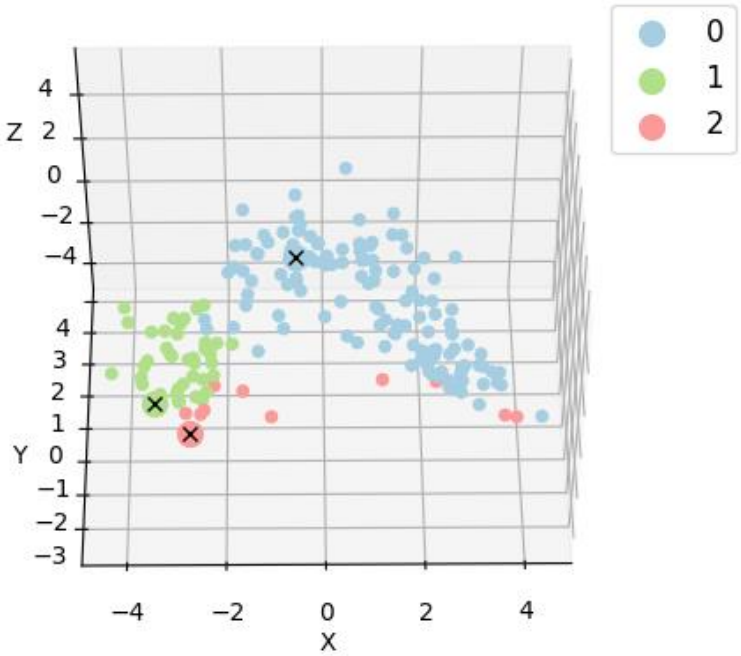
בשלבים הראשונים של הפרויקט נתקלנו בבעיות של זמן ריצה גבוה ותוצאות שגויות. חיפשנו דרכים לשפר את הדברים הללו ומצאנו שכאשר בוחרים מרכזים ראשוניים רחוקים יותר אז התוצאות וזמן הריצה משתפרים. התוספת של בחירת המרכזים הראשוניים באופן מושכל נקראת kmeans++.

אופן הפעולה:

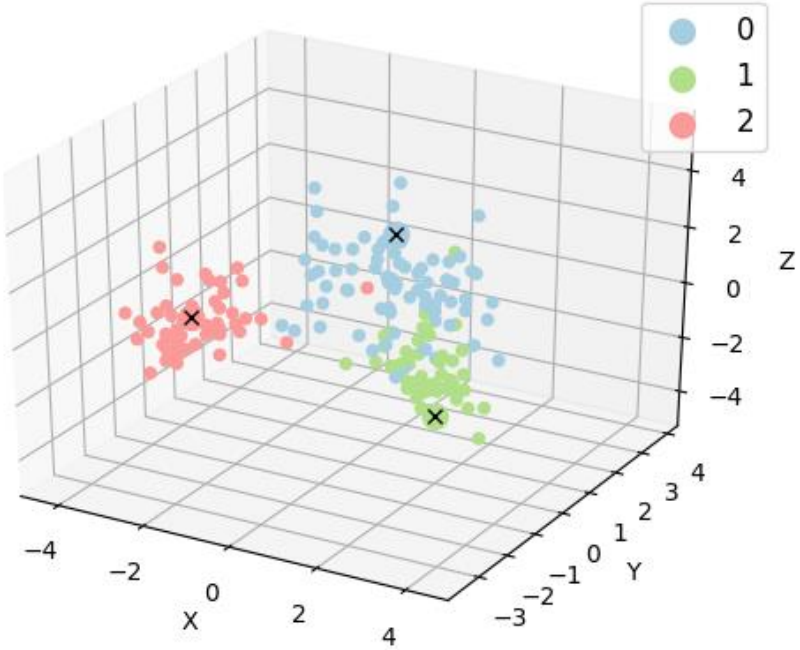
1. בחר מרכז.
2. משקל את שאר האובייקטים לפי מרחקם מהמרכזים שנבחרו.
3. בחר מרכז נוסף באופן הסתברותי לפי המשקל - משקל גבוה יותר הסתברות גבוהה יותר.
4. חזור על 2-3 K פעמים.

השימוש ב Kmeans++ הוביל לשיפור גדול בזמן הריצה ובנכונות התוצאות. (ראה גרף 1 ו- 2).

השוואה בין kmeans ל kmeans++:



גרף 1 תוצאת Kmeans.



גרף 2 תוצאת Kmeans++.

מימוש Gaps-Statistics:

כאשר סיימנו את מימוש ה k-means והצלחנו למצוא k אשכולות אשר דומים ביותר זה לזה, נגשנו למצוא פתרון לבעיית מציאת מספר האשכולות המתאים לקבוצת נתונים.

על כן, הגדרנו פונקציית מטרה אותה רצינו למקסם:

$$J = \log_2 Exp_n(w_k) - \log_2 w_k$$

$$w_k = \sum_k \sum_{i=1}^{n} N_{k,i} \cdot \|x_i - \mu_k\|^2$$

w_k - סכום המשקלים הפנימיים של הקלסטרים.

$Exp_n(w_k)$ - נקבע ע"י שיטת מונטה קרלו.

משמעות הפונקציה: חישוב ההפרש בין כל w_k לערך הצפוי עבור ה *K* הספציפי. את הערך הצפוי אנו מחשבים בעזרת שיטת מונטה קרלו, שאותה נסביר בהמשך.

אופן פעולה:

1. נגדיר טווח של *k* חשודים.
2. לכל *k* נחשב את $Exp_n(w_k)$.
3. לכל *k* נחשב את w_k.
4. נחזיר את ה *k* עבור *J* מקסימלי.

בתום החישוב, נבחר את ה k עבורו ההפרש הוא הגדול ביותר. (ראה גרף 4.)

מימוש Monte-Carlo:

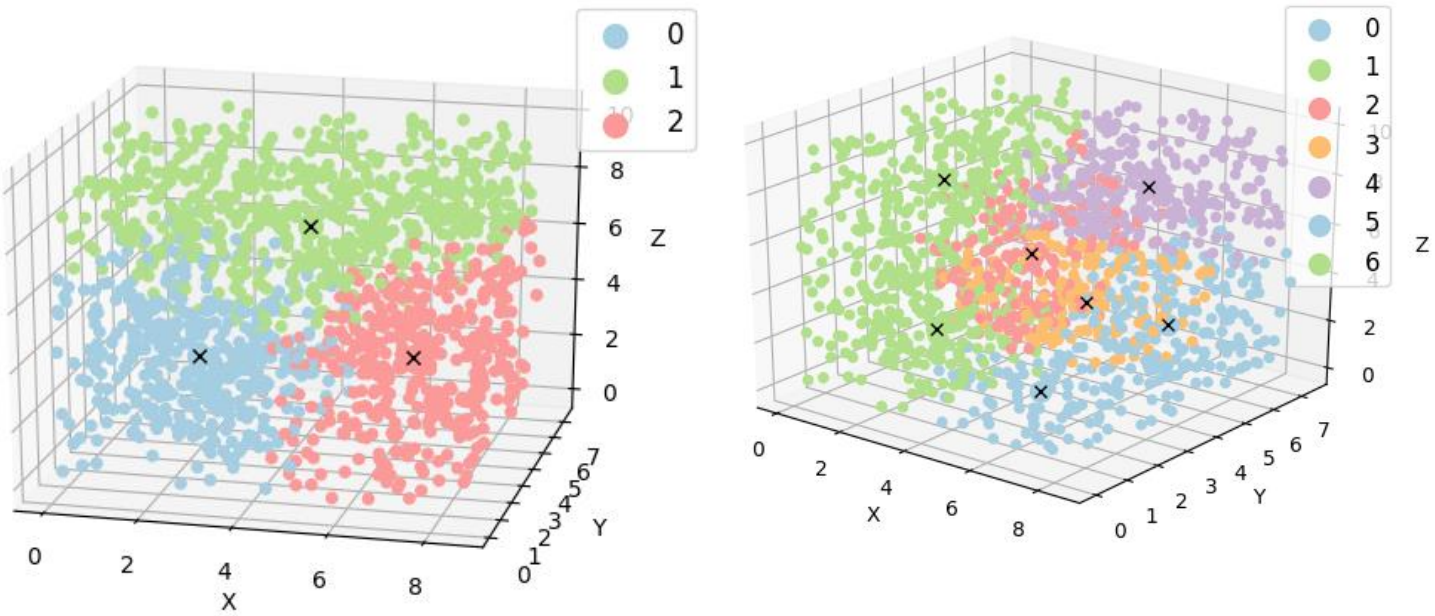
שיטת מונטה קרלו הינה שיערוך לבעיות חישוביות באמצעות מספרים רנדומליים.

אנו ניעזר בשיטה זו על מנת למצוא ערך צפוי לפונקציית המשקל של קבוצת נתונים עבור *K*.

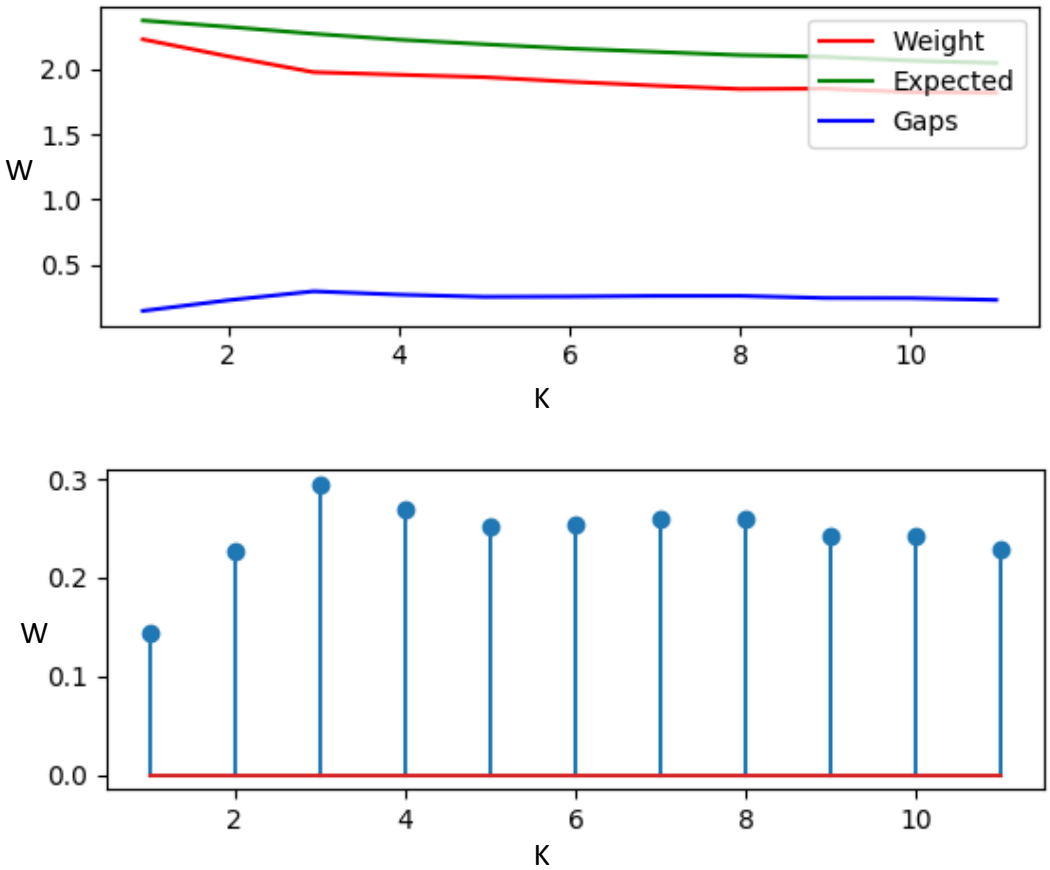
אופן הפעולה:

1. נגריל אובייקטים באופן אחיד בגבולות מרחב הנתונים.
2. נריץ *Kmeans* על הנתונים שהוגרלו.
3. נחשב משקלים פנימיים עם האשכולות שקיבלנו.
4. נבצע את 1-3 *n* איטרציות.
5. נחזיר את ממוצע המשקלים הפנימיים.

את השלבים 1-5 נעשה לכל *k* בטווח שנרצה לבדוק. (ראה גרף 3.)



גרף 3 מונטה קרלו: חישוב משקלים פנימיים של 3 ו- 7 קלסטרים.



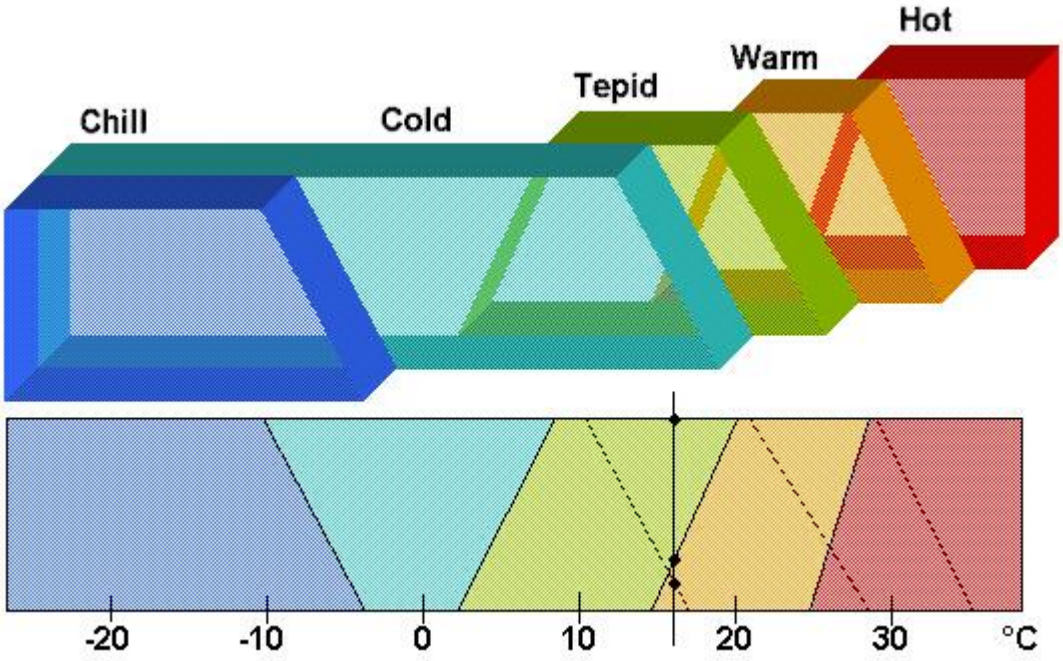
גרף 4 Gap Statistics: משקל מצופה, משקל מצוי והפרשם.

מימוש Cmeans:

לוגיקה עמומה:

עולם המחשבים מתואר באמצעות המספרים הבינאריים. כלומר, בעזרת לוגיקה של כן ולא, אבל העולם בו אנו חיים, איננו רק שחור ולבן, לכן, על מנת לנסות להתאים את עולם המחשבים והתוכנה ל"עולם האמיתי", היה צורך בצורת חשיבה אחרת. כזו שמקבלת את העובדה שמשפט הוא לא 1 או 0. (ראה תרשים 1)

בלוגיקה עמומה ערך האמת יכול לקבל כל ערך בתחום $[0,1]$.



תרשים 1 רמות טמפורטורה, לא ניתן להספק בחם או קר.

חלוקה רכה:

בניתוח אשכולות נהוג לחלק את האובייקטים כך שכל אובייקט שייך לאשכול מסוים בלבד. אך כמו שנאמר קודם, לא תמיד נרצה שזה יהיה המצב. חלוקה רכה לאשכולות, נותנת לכל אובייקט מידת שייכות לכל אשכול. מידת השייכות הינה ערך בתחום $[0,1]$. כאשר סך השייכויות של איבר הינו 1.

Cmeans:

מטרתנו הינה מציאת k אשכולות באופן איטרטיבי על פי מרחק אוקלידי וצמצום הפונקציה:

$$J = \sum_{j=1}^{k} \sum_{i=1}^{n} z_{ij} \left\| x_j - \mu_i \right\|^2$$

כאשר:

$$z_{ij} = \frac{e^{-\beta \left\| x_j - \mu_i \right\|^2}}{\sum_{l=1}^{k} e^{-\beta \left\| x_j - \mu_l \right\|^2}}$$

β – פרמטר עבור פונקציית השייכות.

הפונקציה אותה אנו רוצים לצמצם היא בעצם המרחקים של כל אובייקט מכל אחד מהמרכזים עם משקולת על פי מידת השייכות שלו לאשכול של אותו מרכז.

אופן הפעולה:

1. הגרל k מרכזים.
2. עבור כל אובייקט, חשב את מידת השייכות שלו עבור כל מרכז.
3. חשב מחדש את כל המרכזים.
4. חזור על צעדים 2 ו 3 עד שהמרכזים אינם משתנים.

שלב 3 של האלגוריתם מבוצע כך:

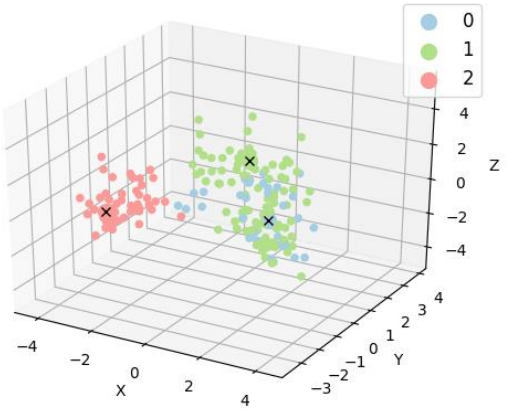
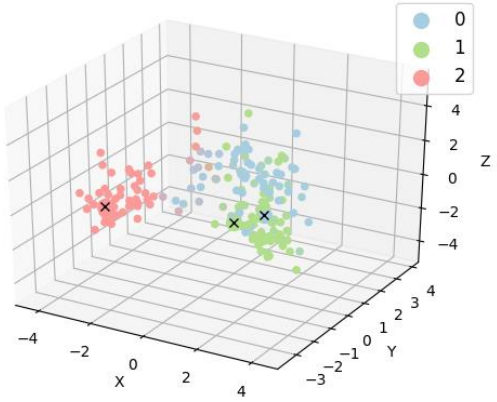
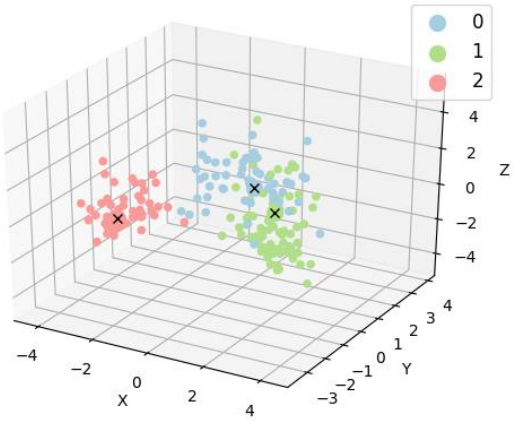
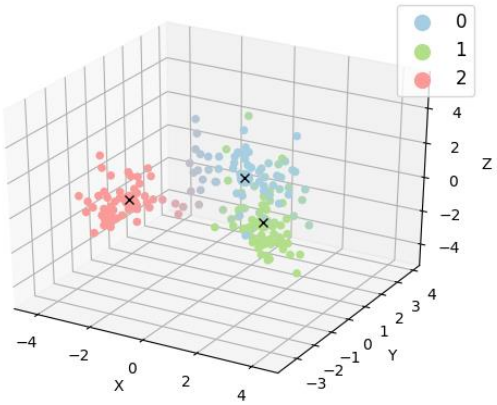
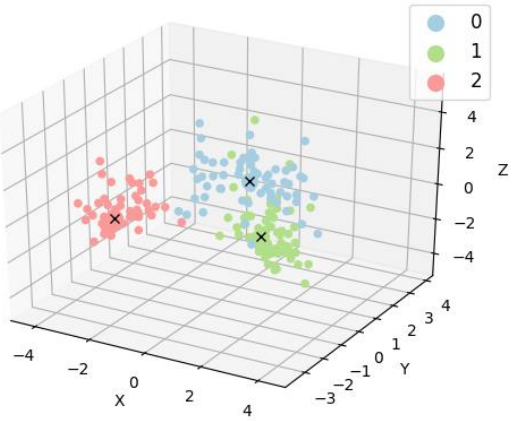
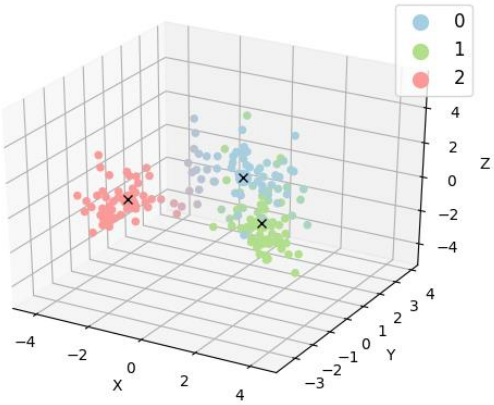
עבור כל אשכול, נחשב ממוצע משוקלל של המאפיינים של כל קבוצת הנתונים. המשקל שיינתן לכל אובייקט בקביעת המרכז החדש, הינו לפי מידת השייכות שלו לאותו אשכול.

דוגמת ריצה – יינות.

לפנינו מאגר נתונים של יינות:

- תיאור: ניתוח כימי של יינות איטלקיים העשויים משלושה זנים.
- מאפיינים שונים: 13 (אחוז אלכוהול, חומצה מאלית, מגנזיום וכו'...).
- מספר אשכולות צפוי: 3.

יש לציין כי איננו מבינים גדולים ביינות ולכן ניתוח אשכולות יכול לעזור לנו לקבל מידע על הנתונים על היינות הללו.

KMEANS++:	CMEANS:
	
	
	

תרשים 2 הפעלת האלגוריתם על מאגר נתונים של יינות.

סיכום:

נוכחנו לראות בעבודתנו כי קלאסטרינג הינו כלי מאוד שימושי, המאפשר עיבוד מקדים של נתונים ללא צורך בידע מקדים עליהם.

למרות ששיטה זו מאוד נפוצה וממומשת בשפות רבות, ביניהן python ו- R, על מנת לקבל תוצאות טובות אשר ניתן לעבוד איתן, יש צורך להכיר טוב את האלגוריתמים השונים של קלאסטרינג ובכך גם לבחור את הכלי המתאים ביותר למאגר הנתונים אשר יביא לתוצאות הרצויות.

במהלך מימוש אלגוריתם ה Kmeans התוודענו לחשיבות בחירת המרכזים הראשונית ולהשפעה שלה על זמן הריצה ותוצאותיה. כמו כן את החשיבות של שיערוך מספר הקלאסטרים המצויים במאגר הנתונים.

ראינו גם כי ישנם שיטות אשר מנסות לחקות את מצב "טבעי" יותר ומחלקות את הנתונים לרמות שייכות לקלאסטרים, כמו למשל האלגוריתם Cmeans.