המכללה האקדמית תל-חי
החוג למדעי המחשב

קלאסטרינג

KMEANS, CMEANS, GAP STATISTICS

מגישים:
יהלי צופים 304952898
נועם סטולרו 201581683
מנחה: ד"ר עופר שיר

תוכן עניינים

מבוא 3
מהלך העבודה..... 4
מטרה 4
רקע 4
שיטות עבודה 5
מימוש kmeans 5
מימוש kmeans++ 6
מימוש Gaps-Statistics..... 6
השוואה בין kmeans ל kmeans++ 7
מימוש Monte-Carlo 8
מימוש Cmeans..... 9
לוגיקה עמומה 9
חלוקה רכה 9
האלגוריתם 10
דוגמת ריצה – יינות..... 11
סיכום 13

רשימת איורים

איור 1 תוצאות Kmeans.....7
איור 2 תוצאות Kmeans++.....7
איור 3 מונטה קרלו : חישוב משקלים פנימיים של 3 ו- 7 קלסטרים 8
איור 4 Gap Statistics : משקל מצופה, משקל מצוי והפרשם.....9
איור 5 רמות טמפורטורה, לא ניתן להספק בחם או קר.....10

מבוא

בעבודה זו עסקנו במימוש אלגוריתמי קלאסטרינג Kmeans ו- Cmeans. כבר בתחילת העבודה שמנו לב כי על מנת לממש כלי אשר ניתן יהיה לעבוד איתו, נדרש לממש מספר שיטות נוספות, ביניהן Gap-Statistics ו- Monte-Carlo.

בסופו של דבר הצלחנו לממש את שני האלגוריתמים וליצור כלים אשר ניתן להשתמש בהם ליצירת קלאסטרים, הכלים מאפשרים ליצור קלאסטרים למאגר נתונים ואף להציג אותם בצורה ויזואלית.
במקרים שבדקנו הגענו לתוצאות דיוק של מעל 95%.

מהלך העבודה

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 לשמש כל אחד.

בעולם של היום ישנה כמות אדירה של מידע בכל נושא. לעיתים אנו נדרשים לעבוד עם מאגרי מידע גדולים מאוד שאין אנו יודעים עליהם יותר מידי. במקרים כאלו, כאשר כלי מבצע את החלוקה לקבוצות, יש לו משמעות מכרעת בעבודה יעילה ומשמעותית עם הנתונים הנמצאים במאגרי המידע.

לכן בחרנו בפרויקט זה ליצירת כלי אשר מבצע את הפעולה של ניתוח אשכולות.

רקע

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

האלגוריתם Kmeans, אותו בחרנו לממש הינו שיטה נפוצה בתחום של ניתוח אשכולות אשר מחלקת את הנתונים לקבוצות לפי מרחק אוקלידי. מכיוון שהחלוקה ש Kmeans מחלק את הנתונים בצורה בינרית לקבוצות, שייך או לא שייך, בחרנו לממש אלגוריתם נוסף, Cmeans.

האלגוריתם Cmeans, דומה מאוד ל- Kmeans, פרט לעבודה שהחלוקה לקבוצות אינה בינרית, אובייקט מקבל מטריצת שייכות וכך ניתן לראות כמה הוא שייך לכל קבוצה.

בשני האלגוריתמים, פרט למסד הנתונים, ישנו צורך להזין את מספר האשכולות אותם נצפה למצוא בנתונים. מכיוון שלא תמיד נדע מהו המספר, נרצה שיטה אשר תעזור לנו למצוא אותו, לשיטה זו קוראים Gap-Statistics.

בכדי להבין כיצד Gap-Statistics עובד, נגדיר מהם סכום משקלים פנימיים. משקל פנימי של קבוצה הינו, סכום המרחק האוקלידי של כל האובייקטים השייכים לקבוצה מהמרכז של הקבוצה, סכום המשקלים הפנימיים הינו סכום כל המשקלים הפנימיים של הקבוצות.

השיטה Gap-Statistics, הינה שיטה לשערוך מספר הקבוצות אשר ניתן לחלק בהם את הנתונים, השיטה משתמשת בעובדה שההפרש בין סכום המשקלים הפנימיים של כל הקבוצות בפועל לבין סכום המשקלים הפנימיים של הקבוצות המצופה יהיה הגדול ביותר.

שיטות עבודה

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

1. כיצד קובעים דמיון?

בכדי לקבוע 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 עלי כותרת, עובי הגבעול, גובה, שורשים. ואילו בבני אדם: צבע עיניים, גובה, משקל, צבע עור, מידת נעליים.

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

2. 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).

3. קביעת מספר האשכולות:

כאשר לא ידוע דבר על הנתונים, לא ניתן לקבוע מראש כמה קבוצות קיימות לנתונים. בחרנו להשתמש בשיטת Gap-Statistics.

מימוש kmeans

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 הינה 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.

מימוש 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).

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:

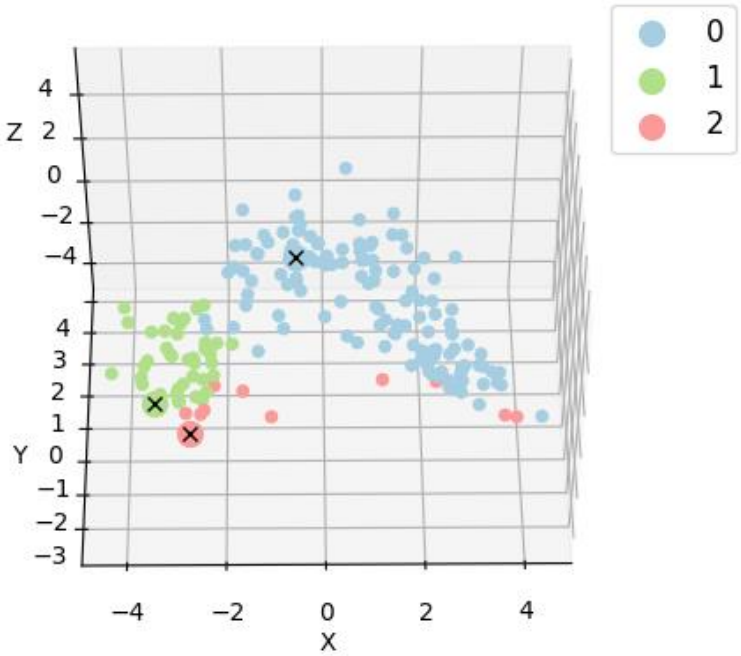
$$J = \log_2 Exp_n(w_k) - \log_2 w_k$$

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

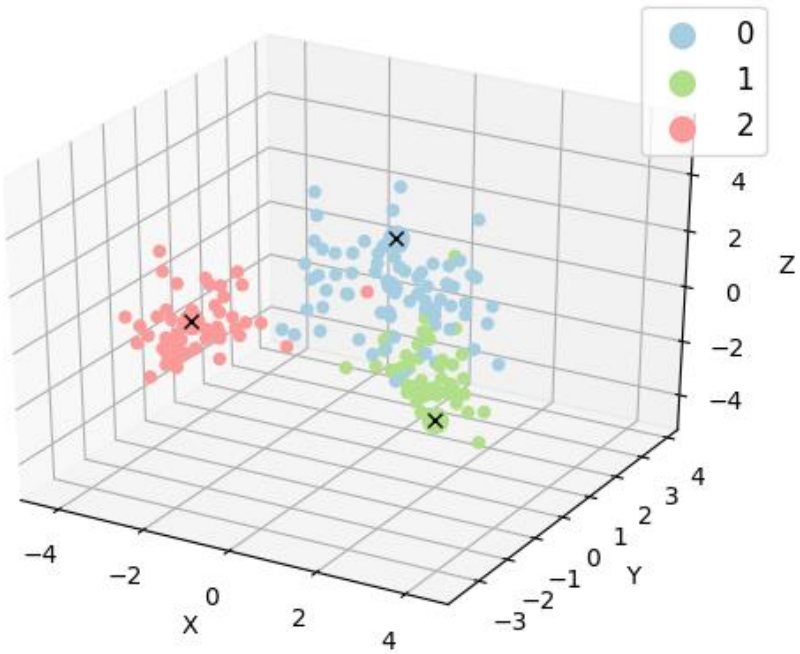
w_k - סכום המשקלים הפנימיים של הקלסטרים.

$Exp_n(w_k)$ - 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.



איור 1 תוצאות Kmeans



איור 2 תוצאות Kmeans++

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

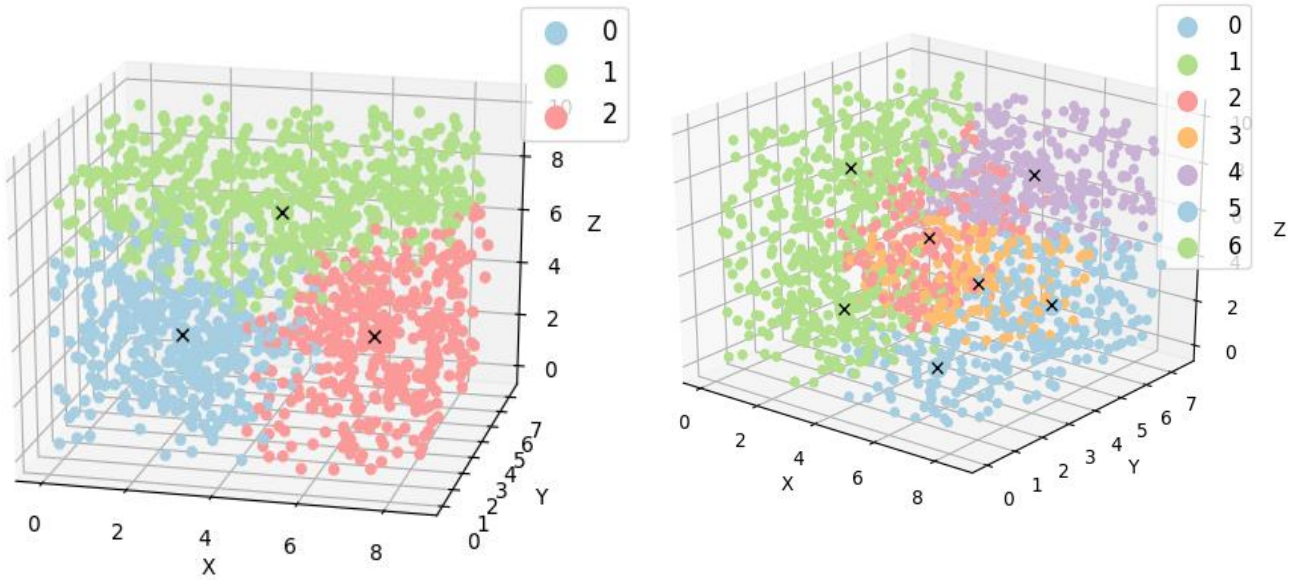
שיטת מונטה קרלו הינה שיערוך לבעיות חישוביות באמצעות מספרים רנדומליים.

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

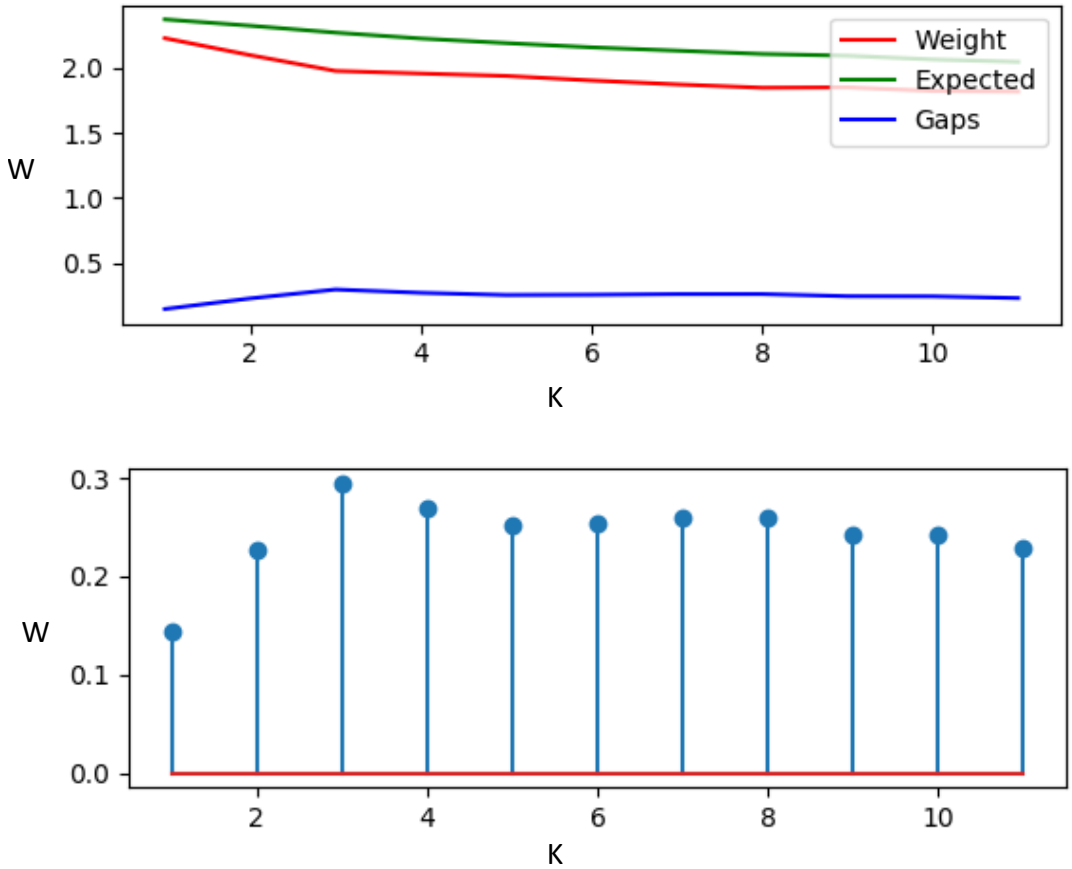
אופן הפעולה:

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.)



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



איור 4 Gap Statistics: משקל מצופה, משקל מצוי והפרשם

מימוש Cmeans

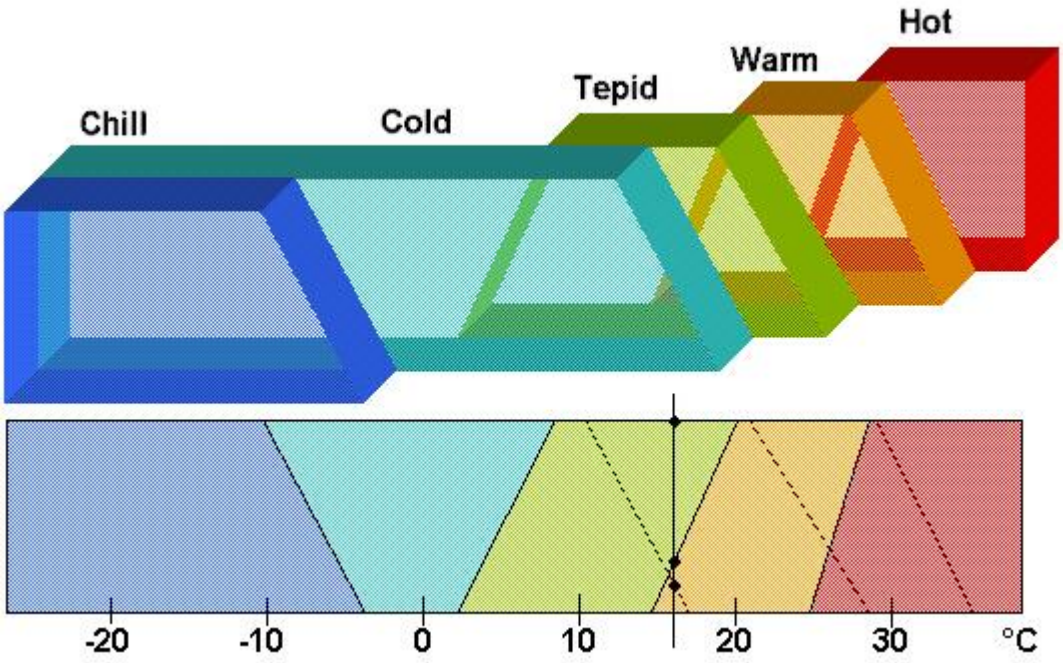
לוגיקה עמומה

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 $[0,1]$.

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.



איור 5 רמות טמפורטורה, לא ניתן להספק בחם או קר

האלגוריתם

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

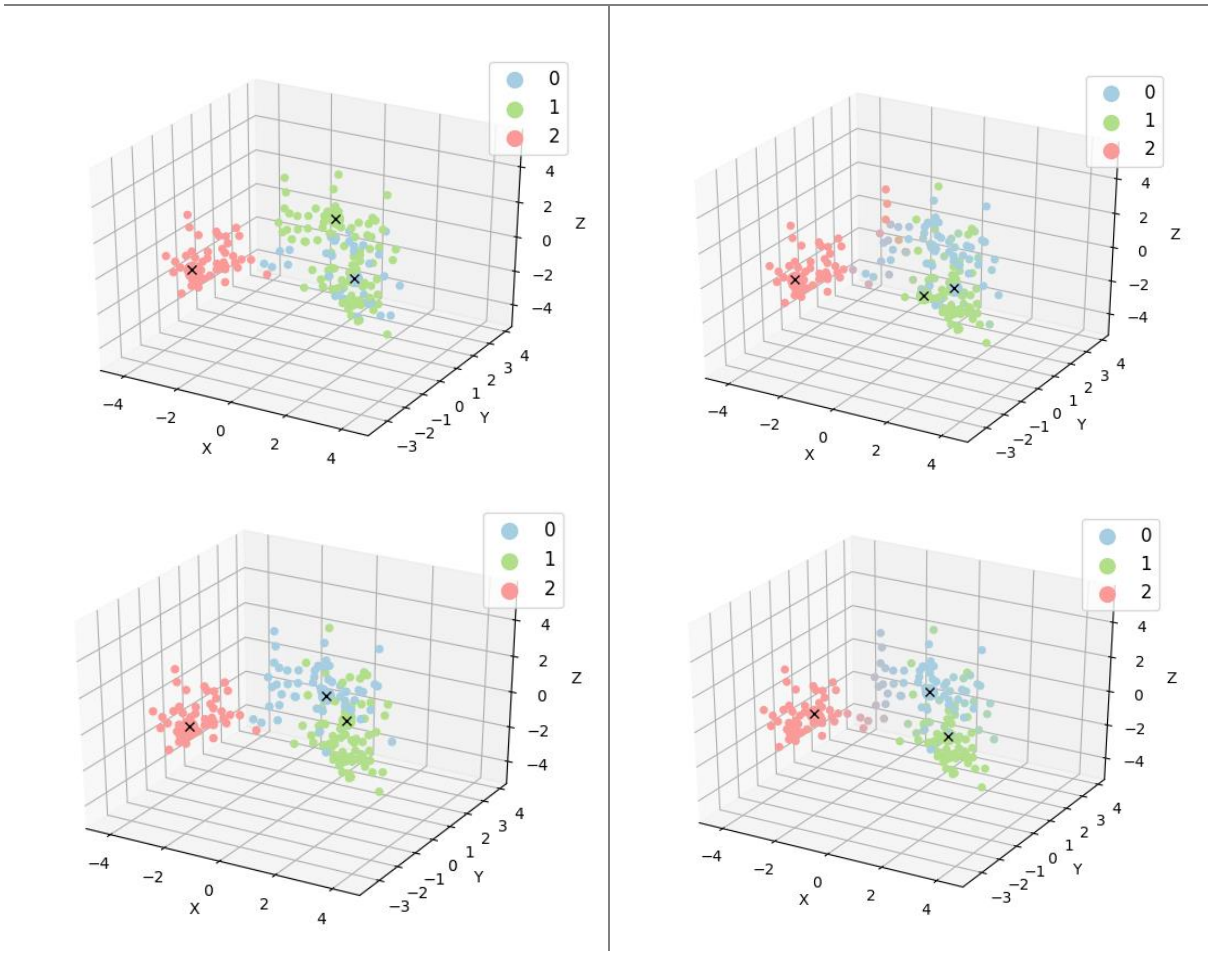
דוגמת ריצה – יינות

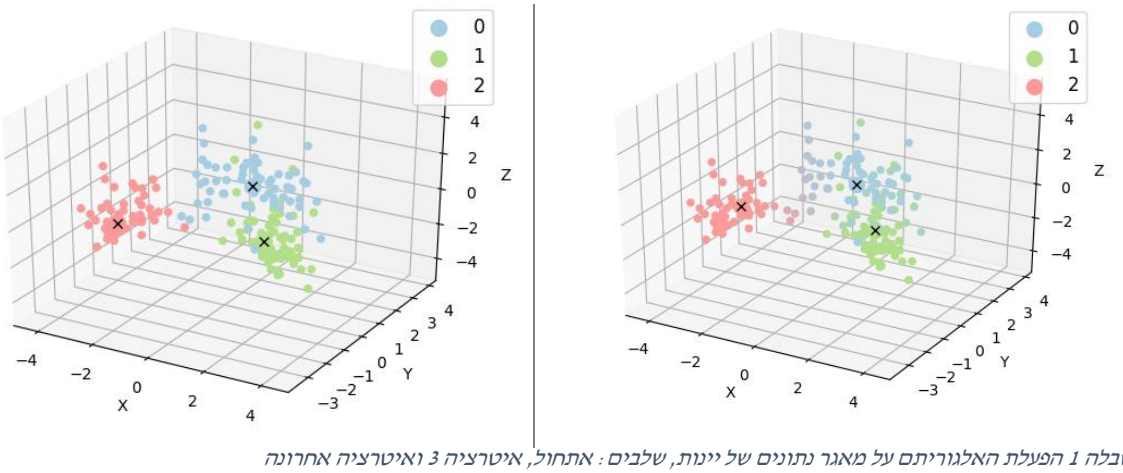
לפנינו מאגר נתונים של יינות:

- 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 נתוני היינות הללו.

CMEANS: **KMEANS++:**





טבלה 1 הפעלת האלגוריתם על מאגר נתונים של יינות, שלבים : אתחול, איטרציה 3 ואיטרציה אחרונה

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

Bibliography

Bauckhage, C. (2015, October 30). Retrieved from ResearchGate: https://doi.org/10.13140/rg.2.1.3582.6643

Hastie, T., Tibshirani, R., & Friedman, J. (2001). The Elements of Statistical Learning. In *The Elements of Statistical Learning Data Mining, Inference, and Prediction* (Vol. II, pp. 501-511). Stanford, California: Springer.

Tibshirani, R., Walther, G., & Hastie, T. (2001). Estimating the number of clusters in a data set via the gap statistic. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 63(2), 411–423. https://doi.org/10.1111/1467-9868.00293