למידה חישובית ממידעית תרגול 10 – UnSupervised Learning

הדבר המרכזי ב-**unsupervised learning** הוא שאין לנו label, לכן יהיה עלינו ללמד על הדאטה: מבנה, **דמיון**. לכן, נרצה להסיק מהו המבנה של הדאטה ה-unlabeled.

Clustering

חלוקת הדגימות ה-unlabeled לתתי קבוצות של קלאסטרים. דפוסים בתוך קלאסטר יהיו מאוד דומים, ודפוסים מתוך קלאסטרים שונים יהיו מאוד שונים. אלגוריתם הקלאסטרינג ימצא קלאסטרים, תתי קבוצות של הדאטה, גם אם לכאורה אין באמת קלאסטרים בדאטה. ומטרתו למדל את המבנה השוכן בחובו של מרחב הפיצ'רים של הדאטה. לכן אין שיטה "טובה ביותר" למצוא את הקלאסטרינג הטוב ביותר, אבל נוכל להגיד שבהינתן הנחות התכנסנו לתוצאה הטובה ביותר.

יש להגדיר באלגוריתם קלאסטרינג איך מודדים דמיון, מהי פונקציית המטרה וכו'...

מדידת דמיון

- עלינו לדעת איך למצוא דמיון, נוכל למשל נוכל לדבר על מרחק – ככל שדגימות קרובות יותר כך הן יותר דומות.
- מרחק חייב לקיים את התכונות הבאות
  - Non-negativity:
    - $d(x_1, x_2) \geq 0$
  - Identity of indiscernible:
    - $d(x_1, x_2) = 0 \Leftrightarrow x_1 = x_2$
  - Symmetry:
    - $d(x_1, x_2) = d(x_2, x_1)$
  - Triangle inequality:
    - $d(x_1, x_2) \leq d(x_1, x_3) + d(x_3, x_2)$
- ראינו סוגי מרחקים שונים: מנהטן, אוקלידי וכו'.
- <u>הגישה הפשוטה:</u> נתחיל בדגימה שמסומנת unclustered ונכניס אותו לתוך קלאסטר חדש. נכניס לקלאסטר הזה את כל הדגימות בעלות מרחק (מדד הדמיון) שנמוך מאיזשהו threshold לאחד מהאינסטנסים בקלאסטר – נחזור על כך עד שלא יהיו דגימות שנוכל להסיף לקלאסטר. שאין להן cluster. נחזור על שני הצעדים עד שלא יהיו דגימות לא מסומנות.

<u>איך נדע אם החלוקה לדגימות היא הטובה ביותר עבור k קלאסטרים כלשהם?</u>

- נמקסם את המרווח (המרחקים) בין הקלאסטרים
- נמזער את המרחקים בין הדגימות בתוך קלאסטר ספציפי

K-Means

- Centroid-based clustering, קלאסטרים מיוצגים על ידי וקטור מרכזי, שהוא לא בהכרח חלק מהדאטה סט.
- בהינתן קבוצת דגימות (x1,…,xn) כאשר xi הוא וקטור d ממדי.
- אלגוריתמים Centroid-based פועלים במטרה לחלק את n הדגימות ל-k (קטן שווה מ- n) קלאסטרים {C1,…,Ck} כך שממזערים את מרחקי הריבועים בתוך כל קלאסטר:

$$\sum_{i=1}^{k} \sum_{x \in C_i} \|x - C_i\|^2$$

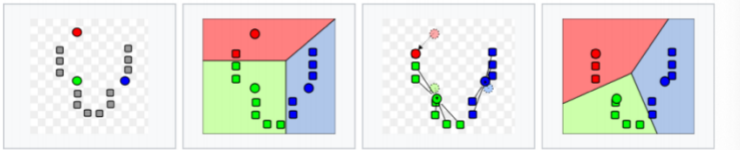
<u>הפתרון הטוב ביותר</u>: יש k^n / k! דרכים לחלק את n האיברים ל-k קבוצות ואם נחפש גם אחר k טוב ביותר נקבל בערך: $\sim \sum_{k=1}^{n} \frac{k^n}{k!}$

לכן מציאת פתרון אופטימלי לבעיית אופטימיזציה זו היא מורכבת גם עבור k=2 לכן בדרך כלל משתמשים בגישות / אלגוריתמים יוריסטיים.

K-Means

- Initialize randomly the k-means $\mu_1, \cdots, \mu_k$
- Repeat
  - For each instance
    - Assign it to the nearest cluster w.r.t its mean μ_i
  - Re-computes μ_i for each cluster
- Until no change in $\mu_1, \cdots, \mu_k$ (or any other stopping condition)
- Return $\mu_1, \cdots, \mu_k$

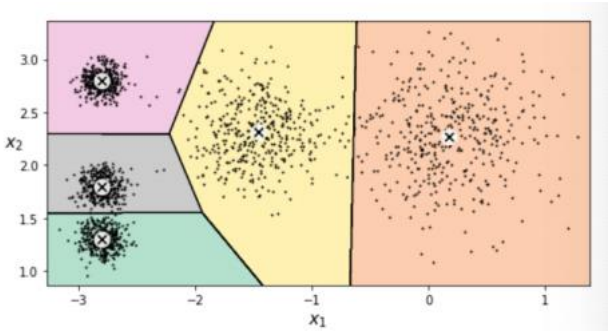
* Usually uses simple Euclidean distance in feature space

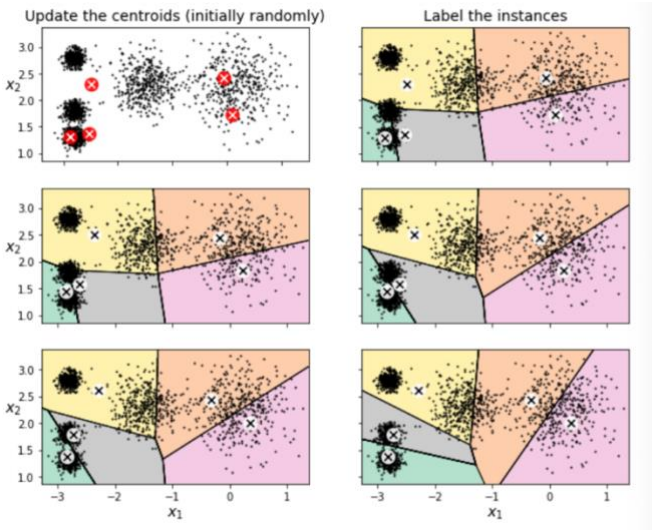


1. k=3 initial "means" are randomly generated within the data domain
2. k clusters are created by associating every observation with the nearest mean
3. The centroid of each of the k clusters becomes the new mean
4. Steps 2 and 3 are repeated until convergence has been reached

דוגמה למימוש K-Means Clustering על דיאגרמת Voronoi:

דוגמאות קוד מופיעות בתרגול





להלן הפונקציה שנרצה למזער:

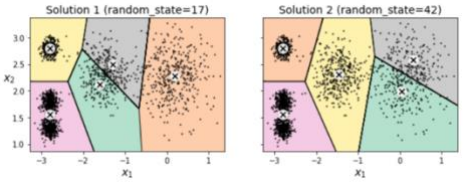
$$minimize \sum_{i=1}^{K} \sum_{x \in D_i} \|x - \mu_i\|^2$$

ונוכל לכתוב אותה גם באופן הבא (כאשר מיו-סי-איי היא התוחלת של הקלאסטר ש-xi שייך אליו:

$$minimize \frac{1}{m} \sum_{i=1}^{m} \|x_i - \mu_{c_i}\|^2$$

בכל שלב הפונקציה $\frac{1}{m}\sum_{i=1}^{m}\|x_i - \mu_{c_i}\|^2$ **מתמזערת באופן הבא:**

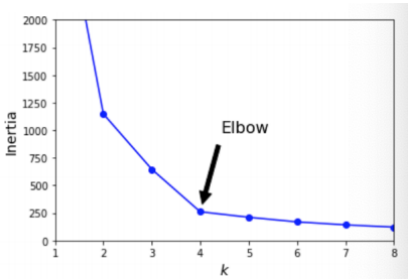
- בשלב ההשמה של דגימה xi לקלאסטר: אם xi לא משנה את הקלסטר – אין שינוי. אם xi משנה את הקלסטר שאליו הוא שייך, הוא משויך לקלסטר הקרוב ביותר – נפחית.
- בשלב החישוב מחדש של התוחלת מיו: הממוצע יניב min square error – ולכן נפחית.
- לכן תהיה התכנסות, אבל לא בטוח לאופטימום.



דוגמה להתכנסות למינימום לוקאלי: נבצע שני איתחולים במקום שונה ונקבל תוצאות שונות, כלומר התקבלה התכנסות במינימום לוקלאלי – ונרצה להימנע ממצב זה.
לכן נגדיר **אינרטיה**.

אינרטיה / inertia: כדי לבחור את המודל הטוב ביותר עלינו לשערך את הביצועים של ה-k-means. לצערנו, קלאסטרינג היא משימה לא מפוקחת אז אין לנו ערכי target. אבל נוכל לפחות למדוד את המרחקים בין כל דגימה לצנטרויד שלה. אינטריה היא סוג של מטריקה והיא מחשבת את סכום המרחקים המרובעיים בין כל training instance לבין הצנטרויד הקרוב ביותר אליה.

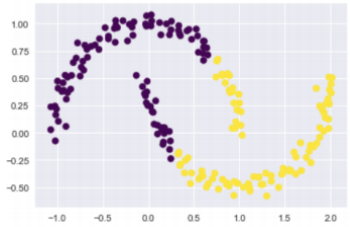
איתחולים רבים / multiple initializations: גישה לפרון בעיית ההתכנסות למינימום לוקאלי הינה להריץ את אלגוריתם k-means מספר פעמים עם איתחולים רנדומליים שונים בכל ריצה, ולבחור את הפתרון שממזער את האינרטיה. (בשקופית 35 מדגימים בקוד)



מציאת המספר האופטימלי של קלאסטרים: לא תמיד נוכל לקחת את הערך k אשר ממזער את האינרטיה מכיוון שהוא הולך וקטן ככל שנגדיל את k. ככל שיש יותר קאלסטרים, כך כל דגימה תהיה יותר קרובה לצנטרויד הקרוב ביותר אליה, ולכן האינרטיה תלך ותקטן. אבל, נוכל להציג את האינרטיה כפונקציה של k ולנתח את הגרף – כך שנשאף לבחור את ה-k בו מופיע "מרפק", בו השינוי באירנטיה מפסיק להיות משמעותי. בדוגמה הנ"ל יש מרפק עבור k=4, המשמעות היא שפחות קלאסטרים מ-4 יהיו לא טובים, ויותר מ-4 לא יעזרו יותר מידי ועשויים לחצות קלאסטרים ל-2.

Hard Clustering vs. Soft Clustering

- **Hard** – נניח כי כל דגימה ניתנת השמה "קשה" (כלומר חד משמעית), לקלאסטר אחד בדיוק. כל דגימה יודעת בדיוק לאן היא שייכת. גישת Hard היא זו שהשתמשנו בה עד עכשיו.
- **Soft** – נותן הסתברויות שדגימה תהיה שייכות לכל אחד מהקלאסטרים (למשל עבור דגימה xi ועבור 3 קלאסטרים, יהיה לה 70% להשתייך לקלאסטר A, 25% להשתייך לקלאסטר B ו-5% להשתייך לקלאסטר C).
- <u>איך אפשר להשתמש בגישת ה-Soft ב-k-means?</u> עם מרחקים. למשל, לכל דגימה נחשב וקטור מרחקים מכל התוחלות, ננרמל את הווקטור ונשתמש בממוצע המשקולות כדי לחשב את התוחלות החדשות.



<u>בעיה נוספת שיכולה לעלות הינה</u>: ההנחה כי הנקודות יהיו קרובות ביותר למרכז הכובד שלהן מאשר לתוחלות אחרות בין k-means clusters, היא הנחה לינארית. לכן, שיטת k-means אינה מתאימה עבור גילוי clusters עם צורות שאינן קעורות או עם clusters בעלי גדלים שונים. בנוסף לכך, האלגוריתם רגיש לרעש ול-outlier data points מכיוון שמספר קטן של data כזו יכול להשפיע על ערך התוחלת.

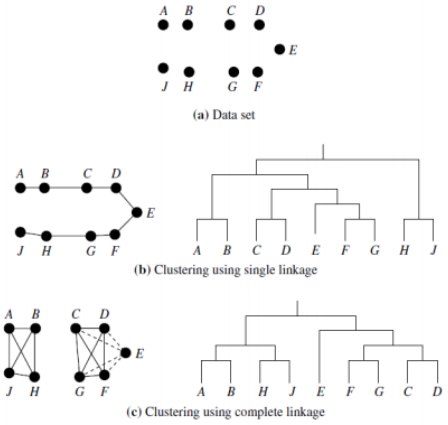
<u>נפתור זאת על ידי קלאסטרינג היררכי / Hierarchial Clustering</u>:

שתי הגישות העיקריות לקלאסטרינג היררכי הינן agglomerative ו-divisive.

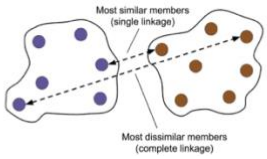
- ב-**agglomerative clustering** אנחנו מתחילים עם כל דגימה כקלאסטר וממזגים את הזוג הקרוב ביותר של קלאסטרים עד שנגיע לקלאסטר בודד. זהו תהליך איטראטיבי שמסתכם בצעדים הבאים: נחשב את מטריצת המרחקים על כל הדגימות, נייצג כל נקודת דאטה כקלאסטר שהוא סינגלטון, נמזג את שני הקלאסטרים הטובים ביותר בהתבסס על ה-linkage criterion הנבחר, נעדכן את מטריצת הדמיון/המרחקים, נחזור על צעדים 2-4 עד שנגיע לקלאסטר יחיד.
- ב-**divisive hierarchial clustering** אנחנו מתחילים עם קלאסטר יחיד שמכיל את כל הדגימות שלנו ובאופן איטראטיבי מפצלים אותו לקלאסטרים קטנים יותר עד שנישאר עם קלאסטר לכל דגימה.

Linkage Measure

- ה-linkage criterion קובע באיזה מרחק להשתמש מבין קבוצה של דגימות.
- האלגוריתם ימזג זוגות של קלאסטרים שממזער את הקריטריון הזה.
- מדידות linkage נפוצות:



- **Single linkage** uses the minimum of the distances between all observations of the two sets
- **Complete linkage** uses the maximum distances between all observations of the two sets
- **Ward** minimizes the variance of the clusters being merged
- **Average** uses the average of the distances of each observation of the two sets



האאוטפוט הראשי של קלאסטרינג היררכי הינו **דנדוגרם / Dendogram** שמראה את מערכת היחסים ההיררכיאלית בין קלאסטרים.

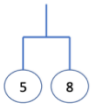
דוגמה:

נתונה לנו הדאטה הבאה: {(1,2), (4,8), (3,9), (7,3), (4,3), (2,4), (5,2), (3,5), (2,5), (6,6)}

נריץ את האלגוריתם agglomerative hierarchial clustering סינגל לינקז', ומנהטן דיסטנס.

Point	X1	X2
0	1	2
1	4	8
2	3	9
3	7	3
4	4	3
5	2	4
6	5	2
7	3	5
8	2	5
9	6	6

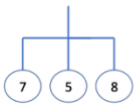
Point	Dist 0	Dist 1	Dist 2	Dist 3	Dist 4	Dist 5	Dist 6	Dist 7	Dist 8	Dist 9
0										
1	9									
2	9	2								
3	7	8	10							
4	4	5	7	3						
5	3	6	6	6	3					
6	4	7	9	3	2	5				
7	5	4	4	6	3	2	5			
8	4	5	5	7	4	1	6	1		
9	9	4	6	4	5	6	5	4	5	

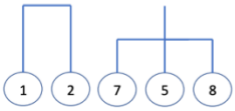


	Dist 0	Dist 1	Dist 2	Dist 3	Dist 4	Dist 5,8	Dist 6	Dist 7	Dist 9
0									
1	9								
2	9	2							
3	7	8	10						
4	4	5	7	3					
5,8	3	5	5	6	3				
6	4	7	9	3	2	5			
7	5	4	4	6	3	1	5		
9	9	4	6	6	5	5	5	4	

המשך
הדוגמה:

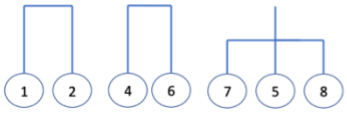
	Dist 0	Dist 1	Dist 2	Dist 3	Dist 4	Dist 7,5,8	Dist 6	Dist 7
0								
1	9							
2	9	2						
3	7	8	10					
4	4	5	7	3				
7,5,8	3	4	4	6	3			
6	4	7	9	3	2	5		
9	9	4	6	6	5	4	5	



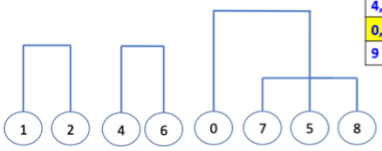


	0	1,2	3	4	7,5,8	6	9
0							
1,2	9						
3	7	8					
4	4	5	3				
7,5,8	3	4	6	3			
6	4	7	3	2	5		
9	9	4	6	5	4	5	

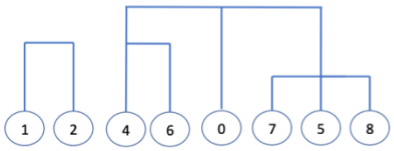
	0	1,2	3	4,6	7,5,8	9
0						
1,2	9					
3	7	8				
4,6	4	5	3			
7,5,8	3	4	6	3		
9	9	4	6	5	4	



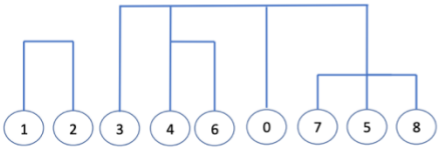
	1,2	3	4,6	0,7,5,8	9
1,2					
3	8				
4,6	5	3			
0,7,5,8	4	6	3		
9	4	6	5	4	



	1,2	3	4,6,0,7,5,8	9
1,2				
3	8			
4,6,0,7,5,8	4	3		
9	4	6	4	



	1,2	3,4,6,0,7,5,8	9
1,2			
3,4,6,0,7,5,8	4		
9	4	4	



The final dendrogram is: