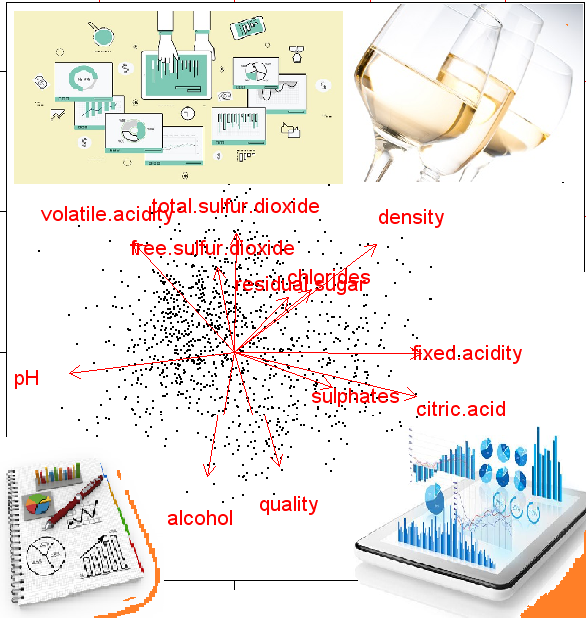
06/06/2019

|  |
| --- |
| מטלת מנחה 22 – פרויקט גמר | חנן |



|  |  |
| --- | --- |
| חנן וולט | מטלת מנחה (ממ"ן) 22 – פרויקט גמר |

תוכן עניינים

[1. **חוקי הקשר** 2](#_Toc10751602)

[א. בחירת שני אלגוריתמים של חוקי הקשר 2](#_Toc10751603)

[ב. מציאת כל הקבוצות התדירות בסט הנתונים 3](#_Toc10751604)

[ג. הצגת חוקי ההקשר החזקים 5](#_Toc10751605)

[ד. הרצת שני האלגוריתמים של חוקי הקשר ודיווח תוצאות ההרצה 5](#_Toc10751606)

[ה. ניתוח השוואתי בין תוצאות שני האלגוריתמים והסקת מסקנות 6](#_Toc10751607)

[2. **ניתוח אשכולות** 7](#_Toc10751608)

[א. הגדרה מהו ניתוח אשכולות 7](#_Toc10751609)

[ב. הגדרת מדדי איכות לאשכולות 7](#_Toc10751610)

[ג. בחירת שתי גישות לניתוח אשכולות 8](#_Toc10751611)

[ד. תיאור שלבי ניתוח האשכולות עבור שתי הגישות 9](#_Toc10751612)

[ה. דיווח תוצאות הניתוחים 10](#_Toc10751613)

[ו. ניתוח השוואתי של התוצאות והסקת מסקנות 10](#_Toc10751614)

[3. **סיכום ומסקנות** 11](#_Toc10751615)

# חוקי הקשר

## בחירת שני אלגוריתמים של חוקי הקשר

נבחר בשני האלגוריתמים המרכזיים שנלמדו בקורס במסגרת חוקי הקשר: Apriory ו-FP-growth –

1. **Apriory**
   1. **תיאור**: אלגוריתם זה מוצא קבוצות תדירות באורכים הולכים וגדלים ע"י בניית קבוצות המועמדות כפוטנציאליות להיות קבוצות שכיחות, בדיקתן מול תנאי הסף של התמיכה המינימאלית, ויצירת קבוצות שכיחות על סמך אלו המועמדות שעמדו בתנאי הסף. פעולות האלגוריתם היא איטרטיבית כאשר גודל הקבוצות השכיחות שנמצאות מתרחב באופן אינקרמנטלי עם ריצת האלגוריתם: באיטרציה הראשונה נמצאות הקבוצות בנות איבר יחיד (הפריטים האינדיווידואליים השכיחים) ולכל k>1, באיטרציה ה-k מבוצע חיפוש של קבוצות שכיחות בגודל k ע"י בניית צירופים (join) בין קבוצות מועמדים בגודל k-1 שנמצאו באיטרציה קודמת כקבוצות שכיחות הנבדלות ביניהן בתכונה אחת בלבד, וע"י כך האלגוריתם בונה קבוצה חדשה בגודל k, שנבדקת מול תנאי הסף של התמיכה המינימאלית ומצורפת לפלט הקבוצות השכיחות במידה ועומדת בתנאי הסף. בסיום ריצת האלגוריתם ומציאת הקבוצות השכיחות, האלגוריתם מפיק את חוקי ההקשר החזקים, דהיינו אלו שעומדים בנוסף לתנאי סף של התמיכה גם בתנאי הסף של הביטחון. קיימים גם מדדים נוספים שניתן לשלב לסינון/בחירת החוקים המשקפים את מידת הקשר (Correlation) שבין התכונות/פריטים שבחוק.
   2. **ניתוח**: מספר תתי-הקבוצות של קבוצת פריטים/איברים בגודל  הינה , לפיכך, באופן תיאורטי, חיפוש ממצה של האלגוריתם על תתי-הקבוצות האפשריות בגדלים ידרוש זמן ריצה אקספוננציאלי (מעריכי). בפועל לעומת זאת, האלגוריתם חוסך זמן ריצה יקר של בדיקת כל תתי-הקבוצות האפשריות תוך הסתמכות על עקרון ה-'אפריורי': עקרון זה מקיים את תכונת האנטי-מונוטוניות: "אם קבוצה אינה שכיחה, אזי גם כל קבוצה המכילה אותה אינה שכיחה". לפיכך, מרגע שהאלגוריתם מצא שצירוף מסוים אינו שכיח, זה חוסך את בדיקת כל הקבוצות המכילות צירוף זה. טקטיקה זו מזכירה במקצת את טקטיקת התכנון הדינאמי, המסתמכת על תוצאות ביניים משלבי חישוב קודמים כדי להסיק תוצאות עוקבות במאמץ מופחת.
   3. **נימוק הבחירה:** ראשית, אלגוריתם זה פשוט (Straight forward), מה שמקל את ההבנה ומעקב. שנית, הודות לעקרון האפריורי לעיל בתוספת אופטימיזציות נוספות כגון שימוש בעץ גיבוב, חלוקת האוכלוסייה למחיצות ודגימות אקראיות האלגוריתם מגיע לרמת סקלביליות טובה ותומך גם בבסיסי נתונים גדולים במיוחד.
2. **FP-growth**
   1. **תיאור**: להבדיל מאפריורי, אלגוריתם זה חוסך את הצורך ביצירת קבוצות מועמדים ובדיקתן. הוא עושה זאת בעזרת מבנה נתונים ייעודי הנקרא 'עץ תבניות-שכיחות' (FrequentPattern-tree או בקיצור FP-tree). כל צומת בעץ מייצג פריט, והעץ במכלול משקף את תתי-הקבוצות השכיחות כמסלולים בעץ, כאשר רמת התדירות של פריט היא ביחס הפוך לעומק שלו בעץ: ככל שפריט תדיר יותר, הוא יופיע ברמה נמוכה יותר בעץ (קרוב יותר לשורש) וככל שפריט נדיר יותר הוא יופיע ברמה עמוקה יותר (קרוב לעלים). אופן ייצוג זה מאפשר לייצג תתי-קבוצות משותפות ע"י תתי-מסלולים. בכדי לנהל מעקב אחר רמת השכיחות של פריט במסלול מסוים, מתחזקים בכל צומת מונה עם מספר המופעים של הפריט במסלול. בכדי לקבל את המונה הכולל של פריט בכלל העץ, מתחזקים גם טבלת הצבעות מהפריטים לצמתים בעץ, ומכל צומת יוצרים רשימה מקושרת של הצבעות בין צמתים המייצגים מופעים של הפריט בהסתעפויות שונות, וע"י כך אפשר לסרוק את הרשימה ולקבל את המנייה הכוללת של פריט בקבוצה. בניית העץ נעשית מלמעלה-למטה, ושחזור הקבוצות השכיחות וכללי ההקשר החזקים מבוצעים בסיום בניית העץ מלמטה-למעלה, ע"י בניית תתי-עצים מותנים לפריטים השכיחים והתמקדות במסלולים המובילים אליהם.
   2. **ניתוח**: ייצוג הקבוצות המשותפות כמסלולים משותפים בעץ מאפשר לכווץ את גודל הייצוג של בסיס הנתונים לייצוג קומפקטי וחסכוני. בנוסף לחיסכון בזיכרון, החיסכון בצורך לעבור על קבוצות הולכות וגדלות של מועמדים תוך שימוש בעץ FP והסתמכות על עקרון האפריורי חוסך גם זמן ריצה ניכר בכמה סדרי גודל.
   3. **נימוק הבחירה:** קומפקטיות בזיכרון ושיפור משמעותי בזמן ריצה (האלגוריתם מסתפק בסריקת מסד הנתונים פעמיים בלבד וחוסך את הצורך ביצירת מועמדים בכל שלב).

## מציאת כל הקבוצות התדירות בסט הנתונים

נשתמש כעת בשני האלגוריתמים שנבחרו לעיל, ונמצא בעזרתם את כל הקבוצות השכיחות תוך תחת ההנחה שתנאי הסף מוגדרים כדלהלן –

* תמיכה מינימאלית (Minimum Support): 40%.
* ביטחון מינימאלי (Minimum Confidence): 60%.

נעבוד עם סט הנתונים המטוייב, המתקבל לאחר הסרת רשומות בעלי ערכים חריגים/קיצוניים נרמול המאפיינים להתפלגות נורמלית סטנדרטית:



בסט נתונים זה יש  רשומות. מתקיים –



לפיכך בכדי שקבוצה תוגדר כשכיחה (תמיכה מינימאלית), כל פריט (מאפיין, ערך) שמוכל בה צריך להופיע בסט הנתונים לפחות פעמים. להלן כמות המופעים (שכיחות אבסולוטית) של רמת איכות היין (המשתנה שאותו אנו מעוניינים לחזות) בסט הנתונים –

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| רמת איכות היין | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | סכום כולל |
| מספר המופעים | **0** | **0** | **13** | **142** | **1371** | **2114** | **874** | **171** | **5** | **0** | **4690** |

מכאן נובע, שהקבוצות הפוטנציאליות היחידות העשויות להוות קבוצות שכיחות הן קבוצות המכילות רמת איכות יין '6'. בפועל, עבור כל יתר המאפיינים אין אף ערך החוזר על עצמו  פעמים או יותר, לכן הפעלת האלגוריתמים "על-עיוור" במצב הנוכחי תוכל לכל היותר להניב קבוצה תדירה טריוויאלית בעלת איבר בודד של רמת איכות יין = 6, וזהו, ללא אף כללי הקשר בפלט. יתר על כן, אף אם סט הנתונים היה מאפשר להפיק כללי-הקשר חזקים, כללים אלו רלוונטיים אך ורק לרמת איכות אחת מתוך כלל ה-10 (רמה 6), ואנחנו מעוניינים לחזות באופן כללי את רמת איכות היין, ולא להתמקד רק בחיזוי/סיווג יינות לרמת איכות '6'. מכאן שעלינו לשנות את הגישה ולא להפעיל את האלגוריתמים "על עיוור".

דרך אחת להתמודד עם המצב היא לבצע [מסע של דיסקרטיזציה](מסע%20של%20דיסקרטיזציה.mp4), ולחלק את המאפיינים ל-bin-ים ע"פ מקטעים (אינטרוולים), במטרה לקבץ קבוצות קטנות לקבוצות גדולות יותר שיהיו גדולות דיין מבחינת תנאי הסף של התמיכה המינימאלית. הדבר דומה לאיחוד מפלגות קטנות במערכת בחירות, הרצות יחד ברשימה משותפת בכדי לעבור את אחוז החסימה. בכדי להביא לידי כך ש-bin מסוים יהיה בכלל פוטנציאלי להשתתפות בקבוצה תדירה (או באנלוגיה - בכדי שקבוצה מאוחדת שכזו אכן תעבור את אחוז החסימה), הוא צריך להכיל כאמור לכל הפחות  רשומות.

**מצד אחד, אם נבצע חלוקה שוות עומק/שכיחות**, דהיינו חלוקה המקצה לכל מקטע מספר רשומות מאוזן פחות או יותר, נבחין כי כל חלוקה לשלושה bin-ים או יותר תניב כ- רשומות לכל היותר בכל מקטע, ומאחר ש- חלוקה שכזו למעשה לא תעמוד בתנאי סף ותניב פלט ריק! לעומת זאת, חלוקה ל-2 bin-ים בלבד בחלוקה שוות עומק אמנם תניב קבוצות גדולות דיין, אולם לאור מנעד הערכים הרחב בסט הנתונים, חלוקה שכזו תהיה כל כך גסה ותביא למסקנות כלליות מידי שאין בהם שום תועלת.

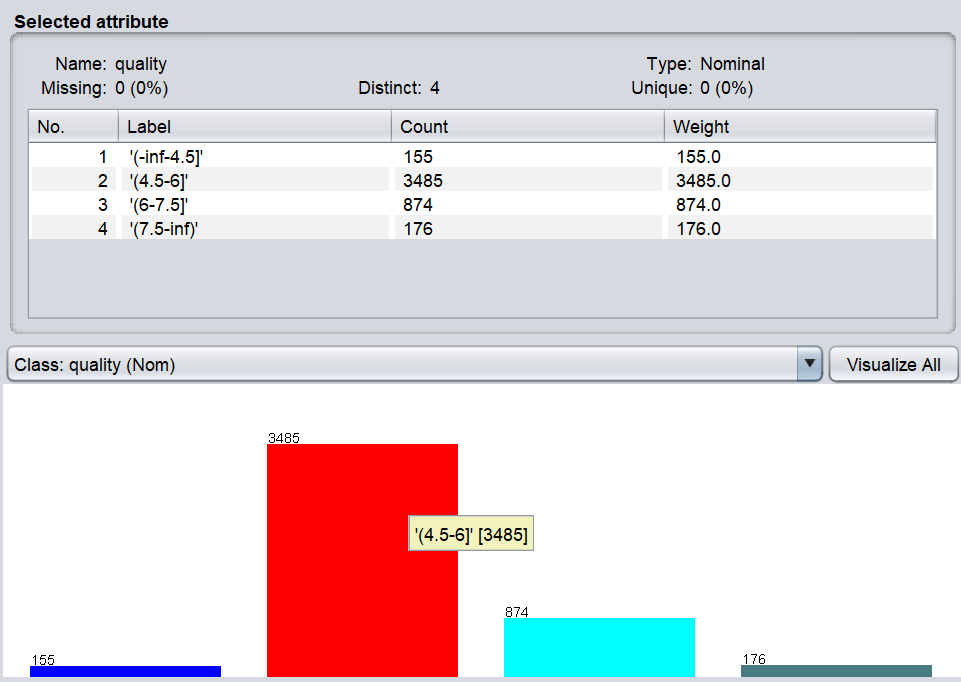
**מצד שני, בחלוקה שוות רוחב**, דהיינו חלוקה למקטעים ברוחב קבוע ללא תלות בהתפלגות הנתונים, מאחר שמלכתחילה רמת איכות 6 היא היחידה העומדת בתנאי סף, ומאחר שגם אם נקבץ את כל הרמות משמאלה או מימין עדיין לא נגיע ל- רשומות, כל קבוצה תדירה תהיה סביב הרדיוס של רמת איכות 6, ולכן ממילא תועלת הפלט הפוטנציאלי היא מזערית. לצורך המחשה, נריץ את אלגוריתם אפריורי על סט הנתונים לאחר שחולק ל-4 bin-ים בחלוקה שוות רוחב. להלן סט הנתונים לאחר ביצוע דיסקרטיזציה ע"י חלוקה לארבעה אינטרוולים שווי-רוחב –



עבור סט זה, פלט תוצאות ההרצה של אלגוריתם אפריורי עבור תמיכה מינימאלית של 40% וביטחון מינימאלי של 60% כולל 12 קבוצות תדירות בנות איבר בודד (דהיינו 12 פריטים אינדיבידואלים שכיחים) מבניהן רק אחת מכילה פריט של משתנה המטרה (רמת איכות היין) ו-24 קבוצות תדירות בנות 2 איברים, מתוכן רק 8 קבוצות מכילות את משתנה המטרה של רמת איכות היין. להלן הקבוצות התדירות המכילות את משתנה המטרה של רמת איכות היין לפי סדר עולה של מספר הפריטים (מאפיינים וערכיהם) בכל קבוצה –

|  |  |  |  |
| --- | --- | --- | --- |
| # | גודל הקבוצה | תוכן הקבוצה (הפריטים: לכל פריט: מאפיין וערך) | מספר מופעים אבסולוטי |
| 1 | 1 | quality='(4.5-6]' | 3485 |
| 2 | 2 | fixed acidity='(0.25-0.5]' quality='(4.5-6]' | 2088 |
| 3 | 2 | volatile acidity='(0.25-0.5]' quality='(4.5-6]' | 1978 |
| 4 | 2 | citric acid='(0.25-0.5]' quality='(4.5-6]' | 2278 |
| 5 | 2 | residual sugar='(-inf-0.25]' quality='(4.5-6]' | 1898 |
| 6 | 2 | chlorides='(0.25-0.5]' quality='(4.5-6]' | 2389 |
| 7 | 2 | total sulfur dioxide='(0.25-0.5]' quality='(4.5-6]' | 2295 |
| 8 | 2 | pH='(0.25-0.5]' quality='(4.5-6]' | 2358 |
| 9 | 2 | sulphates='(0.25-0.5]' quality='(4.5-6]' | 2190 |

באופן לא מפתיע, הפלט מכיל רק פריט אחד של רמת איכות, דהיינו בכל הפריטים עם מאפיין רמת איכות יש אותו ערך של רמת האיכות. זה לא מפתיע מאחר שזה הפריט היחיד שעומד בתנאי סף לאחר החלוקה לאינטרוולים (הפריט שיצא בפלט הוא זה ששייך לאינטרוול באדום) –



אלגוריתם FP-growth ב-Weka אינו מציג את הקבוצות התדירות בפלט, אלא רק את חוקי ההקשר, לכן בסעיף זה הוצגו הקבוצות מאפריורי בלבד. רשימת הקבוצות המלאה קיימת בקובץ שלהלן –



## הצגת חוקי ההקשר החזקים

כעת נציג את חוקי ההקשר החזקים המתקבלים מ-2 האלגוריתמים, כאשר בחוקים "חזקים" אנחנו מתייחסים להגדרה מתחתית עמוד 142 במדריך הלמידה, דהיינו חוקים המקיימים את אילוצי ה-'Minimum Support-Confidence Framework', ללא תלות במדדי ה-Correlation המוסיפים מידע בדבר כמה חוקים אלו מעניינים או מועילים.

בשני האלגוריתמים התקבל סט חוקי-הקשר זהה בפלט, המונה 43 חוקי הקשר-חזקים, מתוך חסם עליון של 100 שהוגדר בקלט. מתוך 43 החוקים-החזקים, נציין את אלו הכוללים את משתנה המטרה (רמת איכות היין הלבן) בצד ימין של החוק, דהיינו בצד הנגרר, הסיפא. יש 8 חוקים כאלו. להלן רשימת החוקים עם ציון מספר הרשומות התומכות בחוק, מדד הביטחון ומדד התלות (lift) מציין את מידת הקשר שבין הפריטים בחוק -

1. volatile acidity='(0.25-0.5]' 2555 ==> **quality** ='(4.5-6]' 1978 <conf:(0.77)> **lift:(1.04)**
2. [pH='(0.25-0.5]'=1]: 3048 ==> [**quality** ='(4.5-6]'=1]: 2358 <conf:(0.77)> **lift:(1.04)**
3. sulphates='(0.25-0.5]' 2831 ==> **quality** ='(4.5-6]' 2190 <conf:(0.77)> **lift:(1.04)**
4. fixed acidity='(0.25-0.5]' 2818 ==> **quality** ='(4.5-6]' 2088 <conf:(0.74)> lift:(1)
5. chlorides='(0.25-0.5]' 3229 ==> **quality** ='(4.5-6]' 2389 <conf:(0.74)> lift:(1)
6. total sulfur dioxide='(0.25-0.5]' 3179 ==> **quality** ='(4.5-6]' 2295 <conf:(0.72)> **lift:(0.97)**
7. citric acid='(0.25-0.5]' 3168 ==> **quality** ='(4.5-6]' 2278 <conf:(0.72)> **lift:(0.97)**
8. residual sugar='(-inf-0.25]' 2774 ==> **quality** ='(4.5-6]' 1898 <conf:(0.68)> **lift:(0.92)**

מחוקים אלו, אנו למדים על מידות קשר בין צירופי ערכים של מאפיינים שונים לרמת איכות יין באינטרוול של רמת איכות הנעה מ-4.5 עד 6 (בפועל רמות האיכות המקוריות היו בדידות והכילו ערכים שלמים בלבד), לכן נוכל למעשה לחדד ולהעלות את רמת הדיוק לאינטרוול של רמת איכות בין 5 ל-6.

מדדי ה-lift בירוק (ערכים גבוהים מ-1) מצביעים על מידת קוריאלציה חיובית, מדדי ה-lift בשחור (ערכים השווים ל-1) מצביע על חוסר תלות בין הפריטים, ומדדי ה-lift באדום (ערכים נמוכים מ-1) מצביעים על מידת קוריאלציה שלילית.

כך למשל, אנחנו למדים מ-3 החוקים הראשונים שערכים בטווח הבינוני-נמוך שבין 0.25 ל-0.5 עבור חומציות נדיפה, רמת חומציות וגופרה מחזקים את ריכוז רמת איכות היין סביב המנעד הבינוני של 5-6 בלבד. לעומת זאת, מחוקים 4-5 אנחנו למדים שחומציות יציבה/סטטית וכלוריד עם אותם טווחי ערכים אינם משפיעים על רמת האיכות, ואילו 3 החוקים האחרונים לעיל מלמדים על מגמה שלילית, למשל סוכר שאריתי בטווח הנמוך של עד 0.25 דווקא מרחיק את רמת האיכות מהמנעד שבין 5 ל-6. עם זאת, מניתוח זה גרידא אנחנו לא יודעים להגיד האם רמת הרחקה זו מונוטונית באיזשהו כיוון (לטובה/לרעה מבחינת רמת איכות היין או שמא גם וגם, כלומר לא לכאן ולא לכאן).

בנוסף לחוקים אלו, קיימים חוקים נוספים שאינם כוללים את משתנה המטרה, ועל אלו ניתן להסיק חוקים נוספים תוך שימוש בטרנזיטיביות, להלן רשימת החוקים המלאה:



## הרצת שני האלגוריתמים של חוקי הקשר ודיווח תוצאות ההרצה

בשביל להריץ את האלגוריתם של **FP-growth** היה צורך לבצע תחילה "בינאריזציה" ע"י המרת כל מקטע לעמודה נפרדת המקבלת ערך '1' עבור רשומות בעלי ערך של מקטע זה ו-'0' אחרת (קיים ב-Weka פילטר NominalToBinary  המבצע זאת אוטומטית עבור המשתנים שאינם משתנה המטרה, ועבור משתנה המטרה הטרנספורמציה בוצעה ידנית ב-CSV). להלן סט הנתונים המעודכן לאחר המרת האינטרוולים לעמודות בייצוג בינארי –



להלן הפלטים של 2 ההרצות - אפריורי ו-FP-growth (מומלץ להציגן עם Notepad++):

**** 

## ניתוח השוואתי בין תוצאות שני האלגוריתמים והסקת מסקנות

**לגבי השוואה** - שני האלגוריתמים הניבו אותו סט של כללי הקשר-חזקים, כך שמבחינת תוכן הפונקציונליות שהם מספקים שניהם שקולים.

(הערה: אמנם ב-Weka אלגוריתם של אפריורי סיפק בפלט הריצה בנוסף לחוקי-הקשר גם את הקבוצות השכיחות, אולם זוהי עובדה תלוית מימוש (feature) – גם מהאלגוריתם של FP-growth ניתן לשחזר את הקבוצות התדירות, כפי שבוצע במסגרת ממ"ן 12).

לאור האמור לעיל, המדד הרלוונטי להשוואה בין 2 האלגוריתמים הוא זמן הריצה, אולם פירוט של זמן הריצה ב-Weka אינו זמין בפלט עבור 2 אלגוריתמים אלו.

כמו כן, מבחינת זמן אבסולוטי ממשי, על סט הנתונים לעיל שניהם סיימו לרוץ בשבריר שנייה כך שקשה באמת להשוות ללא איזשהו כלי חיצוני דמוי Profiler שעוקב אחר הריצות מקרוב. עם זאת, אנחנו יודעים שאלגוריתם אפריורי נוהג לעבור על מסד הנתונים שוב ושוב בכל איטרציה, כך שלמרות שבוצעו רק 3 איטרציות (נמצאו קבוצות שכיחות בגודל 2 לכל היותר) כל איטרציה סט נתונים של כ-4-5K רשומות כמו הסט הנידון גוררת עוד אלפי פעולות עיבוד לכן מטבע הדברים שאלגוריתם FP-growth עדיף בזמן ריצה.

הערה: בדוח הפלט של אפריורי מצוין שבוצעו 60 מחזורים עד לקבלת סט החוקים-החזקים ברמות תמיכה וביטחון המבוקשות, אולם מספר המחזורים הרב נובע הן מהמימוש של Weka (באלגוריתם הקלאסי הפרמטרים נקבעים מראש פעם אחת ואינם משתנים ממחזור למחזור), הן מהדלתא הקטנה שנקבעה לשינוי בין המחזורים (0.01) וכן אין זה אומר שבהרצה של FP-growth לא בוצעו 60 מחזורים גם כן (ע"פ התיעוד של weka גם שם מבוצעת סדרת מחזורי ריצה עם פרמטר הולך וקטן).

**לגבי הסקת מסקנות** –

אמנם קיבלנו כמה חוקי הקשר מעניינים באשר להשפעה על רמת האיכות, אך בשל סט הנתונים הלא מאוזן שמוטה בעיקר לרמות איכות בינוניות, חוקי ההקשר החזקים שהצלחנו למצוא רק תורמים ליכולת שלנו להבין מה עלול לגרום ליין להיות מסווג כבינוני (או ליתר דיוק בטווח שבין 4.5 ל-6), אולם אין אנחנו יכולים להסיק מכך גורם לו להיות לא בינוני, כלומר אנחנו לא יודעים מחוקי הקשר אלו בלבד מה יגרום לרמת איכות להיות גבוהה יותר מ-6 או נמוכה מ-4.5, שינויים בצד הגורר של כל חוק עלולים לגרום או לעלייה או לירידה ברמת האיכות, ואת זה אנחנו עוד לא יודעים מהניתוח הקיים.

בניסיון לקבל תובנות על התפלגות כלל הערכים, ניתן אולי לנסות לקחת דגימה אקראית או דגימה שוות מאוזנת שוות התפלגות על מנת שיינתן משקל נוסף גם לרמות איכות שבשוליים ("פריפריה") וע"י כך לנסות לגלות קשרים מעניינים, בצד הידיעה שבשל הדגימה וההטיה מלכתחילה הקשרים אולי יהוו חוקים לא חזקים מבחינת אמינות ו/או ביטחון מכלל האוכלוסייה.

# ניתוח אשכולות

## הגדרה מהו ניתוח אשכולות

ניתוח אשכולות (Cluster analysis) הוא תהליך של חלוקת רשומות (תצפיות) בסט נתונים לתתי-קבוצות, באופן כזה שבכל תת-קבוצה יימצאו רשומות הדומות זו לזו (ע"פ רשימת המאפיינים/ממדים בכל רשומה), אך שונות מרשומות בתתי-הקבוצות האחרות. ניתוח זה מאפשר לסווג קבוצות בעלי מכנה משותף מסוים המבדיל אותן מיתר הקבוצות.

להבדיל מתהליך הסיווג הסטנדרטי, שהינו תהליך לימוד מפוקח של למידה מתוך דגימות/דוגמאות, ניתוח אשכולות הוא תהליך של לימוד לא מפוקח: הקבוצות (אשכולות) שיימצאו לאחר החלוקה אינן ידועות מראש, לא מבחינת כמות הקבוצות ולא מבחינת התוכן שלהן ו/או הקשר/שוני שביניהן. מכאן שניתוח אשכולות יכול לסייע בגילוי דברים נסתרים ולא צפויים כגון תופעות חריגות או מכנים משותפים מעניינים שלא מתאפשר לגלות אותם בתהליך סיווג אקטיבי סטנדרטי שבו אנחנו נדרשים מראש לציין את תגיות המחלקות/תתי-קבוצות שאליהן אנו מעוניינים לסווג.

תהליך ניתוח אשכולות יכול לשמש כאמור לסיווג, וזאת במגוון יישומים (ראייה ממוחשבת וזיהוי פנים בפרט, מיפוי גיאוגרפי, מודיעין עסקי ועוד) אך גם לניתוח ראשוני של הנתונים לקבלת תמונת מצב כללית וכן כשלב עיבוד מקדים לאלגוריתמי כריית מידע אחרים.

## הגדרת מדדי איכות לאשכולות

לתהליך ניתוח אשכולות יש מגוון מדדי איכות, בין היתר – סקלביליות, יכולת התמודדות עם נתונים רועשים, נתונים רב-ממדיים מטיפוסים שונים, יכולת הסתגלות של המודל לנתונים חדשים מבלי להצטרך ללבנות את האשכולות מחדש ועוד.

למדידת האיכות של האשכולות עצמם, דהיינו מדידת איכות של חלוקה מסוימת אד-הוק בפועל לאחר מעשה, משתמשים בין היתר במדדי האיכות הבאים –

1. **הומוגניות ושלמות** - עד כמה דומים זה לזה העצמים בכל אשכול ועד כמה הם שונים מעצמים באשכולות אחרים? ככל שגובר הדמיון והמשותף בתוך האשכול ושונה מעצמים מחוץ לאשכול החלוקה יותר איכותית.
2. **מגמתיות**  - האם ישנה איזושהי מגמה או תופעה לא טריוויאלית שניתן ללמוד עליה מהחלוקה? חלוקה טובה תניב מבנים לא אקראיים המלמדים אותנו על תובנות משמעותיות שלא ידענו עליהן קודם טרום החלוקה.

## בחירת שתי גישות לניתוח אשכולות

מבין הגישות השונות לניתוח אשכולות - שיטת חלוקה, שיטות מבוססות היררכיה, שיטות מבוססות צפיפות, ושיטות מבוססות סריג (grid), נבחר בגישת החלוקה (Partitioning) באלגוריתם k-means, ובגישת מבוססת צפיפות (Density) – באלגוריתם DBSCAN) –

1. **גישה מבוססת חלוקה - אלגוריתם k-means**: באופן כללי, גישות חלוקה מפזרות את העצמים בין מחיצות (partitions) זרות, כאשר החלוקה מבוססת על פונקציות מרחק/דמיון. לאלגוריתמים כאלו לרוב נדרש לציין מראש את k, מספר האשכולות המבוקש כפרמטר קלט.

ספציפית לאלגוריתם k-means המממש את גישת החלוקה, מתוך סך n הנקודות (רשומות/עצמים) נבחרות תחילה k נקודות אקראיות כנגד k האשכולות המבוקשים, וכל אחת מיתר n-k הנקודות ממופה לאחת מאשכולות אלו בהתאם לפונקציית המרחק/דמיון (כל נקודה ממופה לזו שהכי קרובה אליה מבין k הנקודות שנבחרו באקראי). לאחר חלוקה זו, בכל אשכול מחושבת נקודת המרכז (נקודות כובד משקל /ממוצע האשכול) - centroid, ביחס לכלל הנקודות הנמצאות כעת באשכול, ובהתאם, מבוצע חישוב מחדש של פונקציית המרחק/דמיון בין כל הנקודות לנקודות הכובד המעודכנות, ושוב הנקודות מחולקות לאשכולות לפי נקודת המרכז הקרובה אליהן ביותר. תהליך זה נמשך עד אשר הקבוצות מתייצבות, נקודות המרכז בכל אשכול מתכנסות ולא ומקובעות, או עד אשר מגיעים לתנאי עצירה מוגדר מראש (למשל חסם עליון על מספר איטרציות), המוקדם מבניהם.

אלגוריתם זה אינו אופטימאלי בהכרח בחלוקה, שכן מבוצעים בחירות רנדומאליות בהתחלה, ויש גישות שונות למדידת מרחק/מיון. עם זאת, האלגוריתם הזה טבעי ופשוט להבנה, וביצועיו טובים מאוד:  עבור  נקודות,  אשכולות ו- איטרציות, כך שברוב המקרים, זמן הריצה לינארי, על אף שהבעיה הכללית של מזעור המרחקים בין הנקודות היא NP-קשה, דהיינו כלל לא ידוע על פתרון פולינומיאלי עבורה, על כן, על אף הפתרון כאן אינו בהכרח מניב את החלוקה האופטימאלית, הוא מניב פתרון סביר בזמן סביר ריצה מצוין, ולכן נבחר בו.

1. **גישת מבוססת צפיפות - אלגוריתם DBSCAN**: גישות מסוימות לניתוח אשכולות רגישות לרעשים, למשל נקודת קיצון חריגה ב-k-means תשפיע משמעותית על נקודת המרכז של האשכול המכיל את הנקודה החריגה, וע"י כך החלוקה כולה תושפע. בהתאם, גישות שכאלה מצליחות למצוא רק מבנים בצורה קמורה. לעומת גישות מבוססות צפיפות אינן רגישות לרעשים וטובות בלמצוא קשרים במבנים לא טריוויאליים לאו דווקא קמורים, על בסיס תכונות צפיפות בין נקודות בסט הנתונים.

ספציפית לאלגוריתם DBSCAN, קובעים מראש פרמטר קלט  המהווה רדיוס מינימאלי לשכנות/סמיכות בין 2 נקודות, ו- (MinPts) מספר הנקודות המינימאלי בסביבת  של נקודה , בכדי שתהווה נקודת ליבה: נקודות ליבה הן נקודות הנמצאות במרכז צפוף דיו, באופן שמרוכזות סביבן ברדיוס ה- שנקבע  לכל הפחות. האלגוריתם סורק את הנקודות בסט הנתונים ומסווג אותן לנקודות ליבה ונקודות "רעש". כאשר נמצאת נקודת ליבה חדשה, יוצרים ממנה אשכול חדש, ונקודות בסביבת ה- שלה מצורפות לאשכול שלה. ממשיכים לסווג ולשייך את הנקודות עד אשר כל הנקודות סווגו/שויכו.

הגדרת המינוח של נקודת ליבה יחד עם מספר מינימאלי של נקודות בשכנות ורדיוס הסביבה מבטיחים שנקודות קיצון ורעשים יישארו כאלה ולא יגררו אחריהם את האשכול כולו.

הבחירה שלי בגישה זו ובאלגוריתם DBSCAN בפרט היא בשל היכולת שלו למצוא את הצורות הלא טריוויאליות ולכפר על הרגישות לרעשים שיש ל-k-means, וכן בשל זמן הריצה של האלגוריתם, שאף הוא בדומה ל-k-means יחסית סביר ( או לכל היותר ) וכן בשל סט הנתונים הנידון, שמכיל לא מעט מרכזי צפיפות שגישה זו אולי תוכל לפענח.

## תיאור שלבי ניתוח האשכולות עבור שתי הגישות

1. **הכנת הנתונים**

באופן כללי, רב-ממדיות בנתונים מהווה אתגר רציני לתהליכי כריית מידע בכלל ולניתוח אשכולות בפרט: ככל שהממד גבוה יותר, החלוקה נעשית "מלוכלכת" יותר, מתקבלים מבנים יותר אקראיים ופחות משמעותיים וברורים. לפיכך במטרה לייעל ולשפל את תהליך ניתוח האשכולות, נבדוק האם ניתן למזער את ממד סט הנתונים ע"י בדיקת מידת ההשפעה של המאפיינים השונים על סיווג רמת האיכות. לאחר בדיקת מספר מדדים (מדד ג'יני, רווח אינפורמטיבי ועוד) באחד המדדים התקבל פלט המצמצם את הממד משמעותית –



כמו כן, נמיר את משתנה איכות היין לנומינלי, וננרמל את כל יתר המאפיינים לטווח שבין 0 ל-1 בכדי שפונקציית המרחקים/דמיון לא תייחס משקל עודף למשתנים מסוימים על פני אחרים, מה שעשוי לגרום חריגות בשל הבדלים בסולמות מדידה. להלן סט הנתונים המעודכן לאחר צמצום הממדים ונרמול. נשתמש בו לניתוח האשכולות ב-2 הגישות –



1. **פרמטרים וערכיהם** 
   1. **אלגוריתם k-means:**

הפרמטר העיקרי עבור האלגוריתם הקלאסי של k-means, הוא הערך של k, מספר האשכולות המבוקשים לחלוקה (בהתאם לפרמטר זה נקבעות k הנקודות הראשונותשיוצרו את הקלאסטרים ההתחלתיים).

בוצעו מספר ניסיונות הרצה עם ערכי k שונים, וכן השוואה אל מול האלגוריתם x-means ב- Weka המהווה ניסיון שיפור ל- k-means, ובין היתר כולל קביעה אוטומטית של ערך ה-k האופטימאלי מבין טווח ערכים נתון. המסקנה מהבדיקות וההשוואה אל x-means התוצאות המיטביות מתקבלות עבור k=2, דהיינו 2 אשכולות.

פרמטר חשוב נוסף באלגוריתם זה הוא פונקציית המרחק שעל בסיסה נקבעת מידת הקירוב והשיוך בהתאם של כל נקודה לאשכול הכי קרוב אליה. בגרסת Weka אצלי הייתה תמיכה רק בפונקציות המרחק של מנהטן ושל אוקלידס. על אף הפוטנציאל של המרחק האוקלידי להגברת השגיאה במקרה של רעשים רמת הדיוק שלו בהשוואה למנהטן הייתה טיפה טובה יותר, לפיכך נבחר בפונקציית מרחק האוקלידית.

פרמטרים אחרים שהינם טכניים בעיקרם כגון ערך ה-seed הראשוני לבחירה הרנדומאלית של k הנקודות הושארו עם ערכי ברירת המחדל.

* 1. **אלגוריתם DBSCAN:**

לאלגוריתם זה יש לציין את 2 הפרמטרים העיקריים הבאים –

** - גודל מינימאלי של רדיוס מינימאלי לשכנות/סמיכות בין 2 נקודות** – במטרה לקבץ יחד את הנקודות הקרובות באמת הערך מוזער משמעותית מברירת המחדל של weka (0.9), ואכן בכל ניסיון מזעור של  דיוק הסיווג לפי נקודות שחולקו לאותו אשכול עלה, עם זאת, החל מאיזשהו סף מסוים, מגיעים ל-Threshold. מאותה נקודה ואילך הדיוק אמנם ממשיך לעלות, אך מצד שני, מרוב שה- קטן יותר ויותר נקודות לא עומדות בסף קירבה של אף נקודת ליבה, וכתוצאה מכך מסווגות כנקודות "רעש" ולא נכללות באף אשכול. מאזן טוב שנמצא פחות או יותר המשפר את הדיוק מחד אך לא מסווג יותר מידי רשומות כ-"רעש" הוא ערך של 0.3.

** (MinPts) מספר הנקודות המינימאלי בסביבת  של נקודה , בכדי שתהווה נקודת ליבה**. גם כאן יש Tradeoff: מצד אחד אפשר להגדיל את המספר, במטרה לקבל אשכולות עם יותר "בשר" ולא חלוקות לכמה נקודות אזוטריות בודדות בקיצון, עם זאת, הגדלה יתר על המידה עלולה להוביל לכך שכל הנקודות יסווגו לאשכול או שניים לכל היותר וכל היתר יסווגו כרעש. מאחר שבסט הנתונים יש אכן גם קבוצות רשומות בודדות של יינות טובים מאוד או גרועים מאוד, נשאיר את ערך הפרמטר כברירת המחדל של Weka (6).

כמו גן גם לאלגוריתם זה נדרש לציין את **פונקציית המרחק** המבוקשת כפרמטר. להבדיל מאלגוריתם k-means, באלגוריתם זה ב-Weka בגרסה המותקנת אצלי יש תמיכה גם בפונקציות המרחק של [מינקובסקי](https://en.wikipedia.org/wiki/Minkowski_distance) (הכללה של מנהטן ואוקלידס עבור חזקה m כלשהי) וגם של [צ'בישב](https://he.wikipedia.org/wiki/%D7%9E%D7%A8%D7%97%D7%A7_%D7%A6%27%D7%91%D7%99%D7%A9%D7%91). בהשוואה מבניהם דווקא פונקציית המרחק של מנהטן הניבה את התוצאות המדויקות ביותר על כן נבחר בה.

## דיווח תוצאות הניתוחים

להלן פלטי תוצאות ניתוח האשכולות לסיווג רמת האיכות ב-2 השיטות שלעיל –

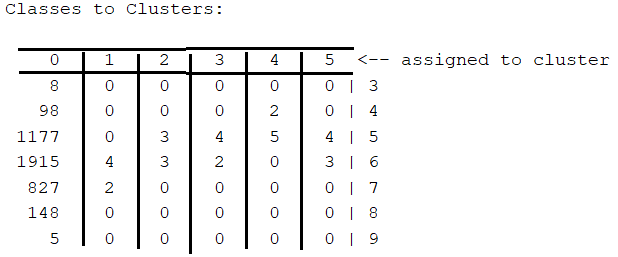


## ניתוח השוואתי של התוצאות והסקת מסקנות

להלן השוואה של אחוזי דיוק הסיווג מבין 2 המודלים –

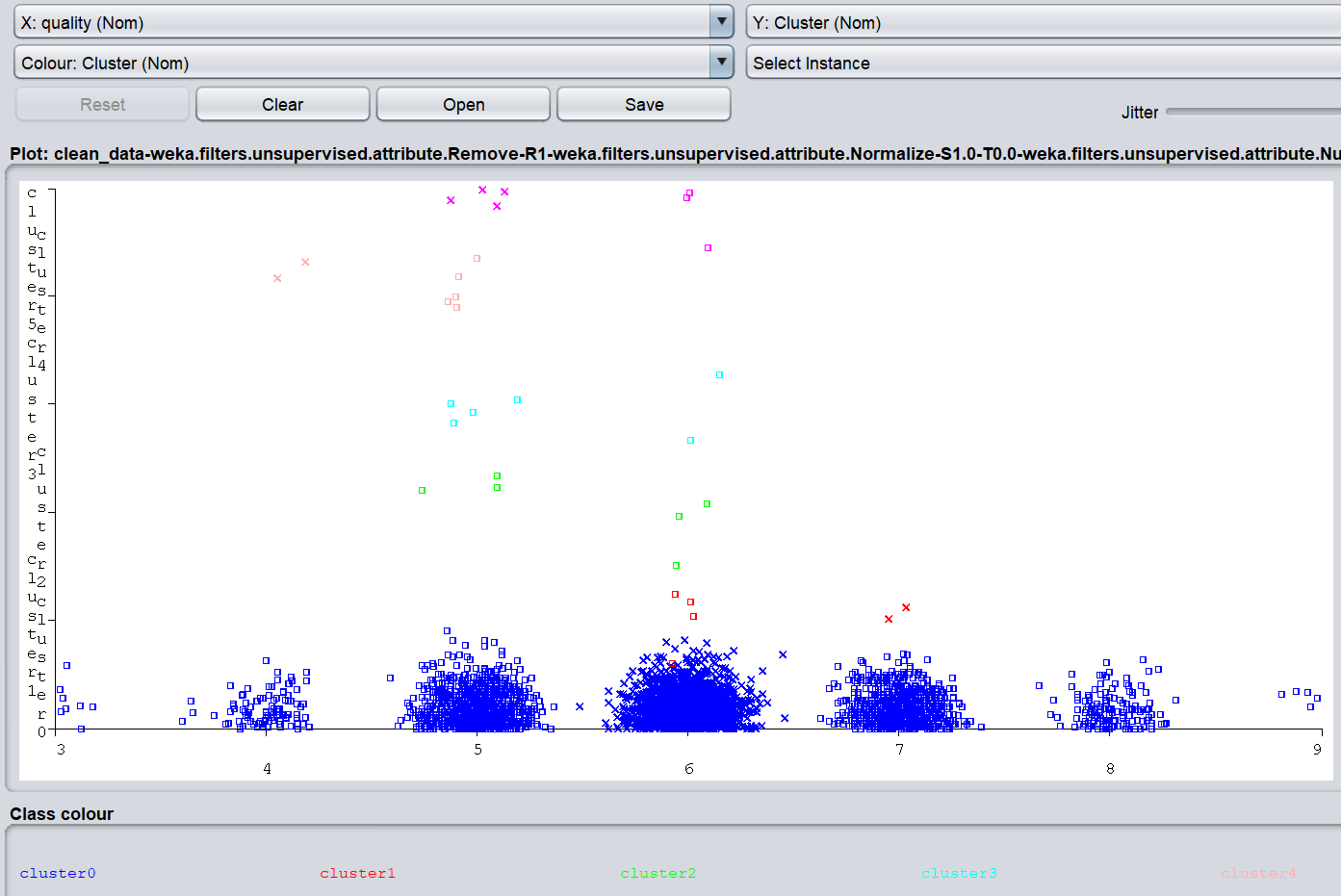
|  |  |  |
| --- | --- | --- |
| מודל | % רשומות שסווגו באופן שגוי | % רשומות שסווגו נכון |
| k-means | 55.5224 | 44.48 |
| DBSCAN | 48.7633 | 51.2367 |

כפי שניתן לראות, המודל של DBSCAN הניב תוצאות טובות יותר ממודל k-means. יתר על כן: כזכור ממטלה 21, חוק הרוב (ה-Base line"") מניב דיוק של 45% בסט הנתונים הקיים. המשמעות היא שמודל k-means שנוצר כלל לא תורם לניתוח ביחס למצב ההתחלתי. לפיכך נתמקד במודל של DBSCAN שכן עבר את סף חוק הרוב וננסה להסיק ממנו מסקנות. ובכן, למרבית הצער, על אף השיפור היחסי אל מול חוק הרוב, החלוקה הקיימת לאשכולות לא תורמת יותר מידי. להלן פילוח סטטיסטי של התפלגות רמות האיכות השונות לאשכולות –



בעמודה הימנית ביותר רשומות רמות האיכות, שורת הכותרת מציינת את מספרי הקלאסטרים, ותוכן הטבלה מכיל את מספר הרשומות מכל רמת איכות שסווגו לכל קלאסטר. ניתן לראות שכמעט ללא תלות ברמת האיכות, כמעט כל הרשומות מרוכזות בקלאסטר הימני ביותר (קלאסטר מספר 0). באשכול זה יש רשומות מכל רמות האיכות, אז לא ניתן להסיק מזה שום דבר לגבי סיווג רמת האיכות. הרשומות הבודדות שנמצאות באשכולות אחרים מהווים כמות כל כך מזערית ביחס לאוכלוסייה כולה כך שגם היינו יכולים להסיק מזה תובנה כלשהי, האמינות שלה הייתה מוטלת בספק.

ניתן לראות זאת גם באופן גראפי, מתקדמים לאורך ציר רמת האיכות ונשארים על אותו אשכול מספר אפס (הכחול) למעט החריגים שלא תורמים יותר מידי –



# סיכום ומסקנות

מבין כל המודלים לסיווג רמת איכות שהשתמשנו בהם במסגרת המטלות, המודל שהניב את התוצאות הטובות ביותר עבור סט הנתונים הקיים הינו עץ החלטה במימוש של אלגוריתם CART, עם אחוז דיוק של 55.23%. לפיכך, בהינתן רשומה חדשה עם פרמטרי מדידות של יין הוא מסווג את רמת האיכות שלה באופן הטוב ביותר מבין האחרות. יחד עם זאת, מודל זה קשה לניתוח ולא ברור הלוגיקה שהוא מפעיל לסיווג בשל ריבוי הסתעפויות ורמות עומק.

לעומת אי-הבהירות של המודל הטוב ביותר שהופק, עם חוקי ההקשר ואסוציאציות מצאנו, קיבלנו כמה חוקים חזקים שברור בדיוק מה הם אומרים וכיצד הם משפיעים על רמת האיכות. למשל – אנחנו יודעים ברמת ביטחון של 68% שבשל הקוריאלציה השלילית שבין סוכר שאריתי נמוך לרמת איכות יין בינונית-מינוס, אם נכין יין עם ריכוז סוכר שאריתי נמוך מאוד רמת האיכות לא תהיה לא בינונית. יחד עם זאת, על אף הבהירות של החוק, אנחנו לא יודעים להסיק מכך ישירות האם ההשפעה תהיה לטובה או לרעה על רמת האיכות, כלומר על אף בהירות חוקי ההקשר, תחום הערכים שאליהם הוא מתייחס די מצומצם ולא שופך אור על התמונה הגדולה.

מסקנות: סט הנתונים הקיים די רועש ומוטה, המעט שהופק ממנו למטרת הסיווג וחיזוי תורם רק במעט מעבר לברירת המחדל הטריוויאלית של חוק הרוב. בכדי להשיג תוצאות טובות יותר כדאי אולי לבצע דגימות נוספות יותר מאוזנות ופחות מוטות.