**סוגיות לדיון**

1. מה הקשר בין KMeans ובין GMM?

שני האלגוריתמים נותנים cluster-ים דומים, ולמעשה נוכל לחשוב על GMM כעל מעין הכללה של KMeans, כאשר אין דרישה ל-cluster-ים כדוריים, וניתן לקבל גם cluster-ים בעלי צורה אליפטית. ב-GMM אנו מקבלים גם פרשנות הסתברותית, שניתן למצוא לה שימושים רבים.

1. הכללה אל מחוץ לסט האימון

איך נקבל פרדיקציה לנקודות שלא הופיעו בסט האימון?

* 1. ב-KMeans ההכללה אל מחוץ לסט האימון היא פשוטה באופן יחסי – עבור דוגמא חדשה, ה-cluster שלה הוא זה שמרכזו קרוב אליה ככל האפשר. אכן, sklearn תומך באפשרות הזאת.
  2. ב-DBSCAN לא ברור איך לבצע את ההכללה הזו. פתרון נאיבי יהיה לבחור ב-cluster הכי קרוב. אולם cluster-ים של DBSCAN הם באופן כללי פחות "כדוריים", ושימוש בקריטריון מרחק פשוט הוא פחות הגיוני. בנוסף, עלינו לחשב לכל נקודה את המרחק מכל נקודה בכל cluster כדי לקבוע לאיזה cluster לשייך אותה, כלומר המשימה הרבה יותר מסובכת חישובית. אכן, sklearn אף לא תומך באפשרות ל-predict ב-DBSCAN.
  3. ב-GMM העבודה דומה לעבודה ב-KMeans, אולם לפונקציית המרחק ניתן להוסיף פרשנות הסתברותית, ולהשתמש ב-likelihood כדי לקבוע את הגאוסיאן אליו נשייך את הדוגמא. גם כאן sklearn תומך ב-predict (וגם ב-predict\_proba).

1. מציאת אנומליות

האם יש דרך טבעית להגדיר אנומליות?

* 1. ב-KMeans אין דרך טבעית להגדיר אנומליות. ניתן לנסות לנסח מדדים הקשורים למרחק של נקודה ממרכזי כל ה-cluster-ים, להתייחס לפיזור של ה-cluster-ים השונים ועוד.
  2. ב-DBSCAN אנומליות מוגדרות באופן טבעי. אכן, זהו אחד מה-class-ים שחוזרים.
  3. ב-GMM ניתן להגדיר אנומליות בצורה יחסית טבעית, באמצעות פונקציית ה-likelihood שמתקבלת בו (דרך פונקציית score\_samples למשל).

1. מטריקה מותאמת

פעמים רבות אנו רוצים לבצע clustering על מטריקה שאיננה אוקלידית. למשל, עבור דאטה רב מימדי פעמים רבות נוח לעבוד עם מטריקת cosine similarity. איך נוכל לבצע clustering במטריקה זו? נוכל, למשל, להטיל את כל הוקטורים על ספירת היחידה, ואז מרחק ה-cosine similarity ביניהם לא ישתנה, אבל המרחק האוקלידי יהיה כפל ב- של מרחק ה-cosine. באופן זה נוכל להשתמש ב-KMeans, או ב-GMM, ולקבל clustering עם מטריקת cosine similarity באופן אפקטיבי.

אולם, ב-DBSCAN אנו בפועל נדרשים רק ליכולת לחשב מרחקים, כך שניתן לעבוד בו עם כל מטריקה שהיא. בפרט ניתן לתת ל-DBSCAN מטריצה של מטריקה מחושבת מראש. באופן זה נוכל להפעיל clustering על מרחבים בהם המטריקה במובהק אינה אוקלידית – למשל, מרחק עריכה בין מילים.

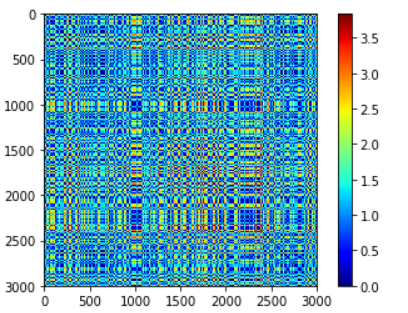
1. מדוע לא ניתן להשתמש במטריקה כלשהי עבור KMeans? איפה האלגוריתם יכשל?

לא ניתן להכליל את חישוב המרכז של cluster למטריקה כללית באופן נוח. במטריקה האוקלידית הנקודה הקרובה ביותר לכל הנקודות האחרות היא גם הממוצע של כולן.

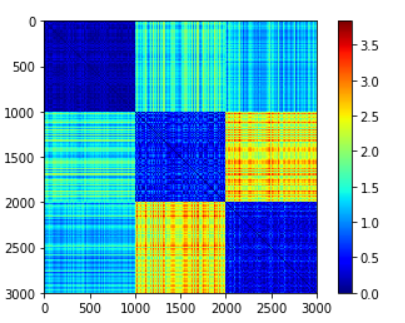
1. איך בודקים האם clustering עבד?

ניתן להשתמש בשיטות מבוססות מדד כלשהו, ולחפש ברך בגרף. למשל, ניתן לחשב את סכום המרחקים בתוך כל cluster לעומת המרחקים בין cluster-ים.

גישה נוספת יכולה להיות בחינה של מטריצת המרחקים. עבור מידע כלשהו, מטריצת המרחקים בין כל שתי דוגמאות רועשת מאוד, ונראית, למשל, כך:



אולם אם נסדר את הנקודות שלנו בסדר הנקבע ע"י ה-cluster-ים (כלומר, קודם כל הנקודות מ-cluster 0, אחר כך כל הנקודות מ-cluster 1 וכן האלה) נקבל מטריצה מסודרת בהרבה:



בחינה של מטריצות שכאלה יכולה לאפשר לנו לקבל מושג על טיב ה-clustering גם במימדים גבוהים, כאשר לא ניתן להציג את ה-cluster-ים.

1. עבודה במימדים גבוהים

במימדים גבוהים האינטואיציה שלנו על האופן בו מתנהגים מרחקים עשויה להיכשל. בפרט ישנם כמה אפקטים מעניינים מאוד במימדים גבוהים שישפיעו עלינו בהקשר זה:

1. התפלגויות "סבירות" נעשות דומות מאוד להתפלגויות על ספירה. כתרגיל, ניתן להגריל נקודות מגאוסיאן n מימדי ולבחון כיצד מתנהגת התפלגות המרחקים של הנקודות מתוחלת הגאוסיאן כתלות ב-n. ניתן לשים לב כי סטיית התקן של המרחקים קטנה, וכי המרחק מהתוחלת הולך ונעשה קבוע.
2. משום כך, למשל, מרכז ההתפלגות, או מרכז ה-cluster, פעמים רבות כלל לא יהווה ייצוג של נקודה מה-cluster. אם נרצה להשתמש במרכז ה-cluster כוקטור מייצג עבור ה-cluster נאלץ "לדחוף אותו" אל ה-cluster.