נושאים מתקדמים בלמידת מכונה

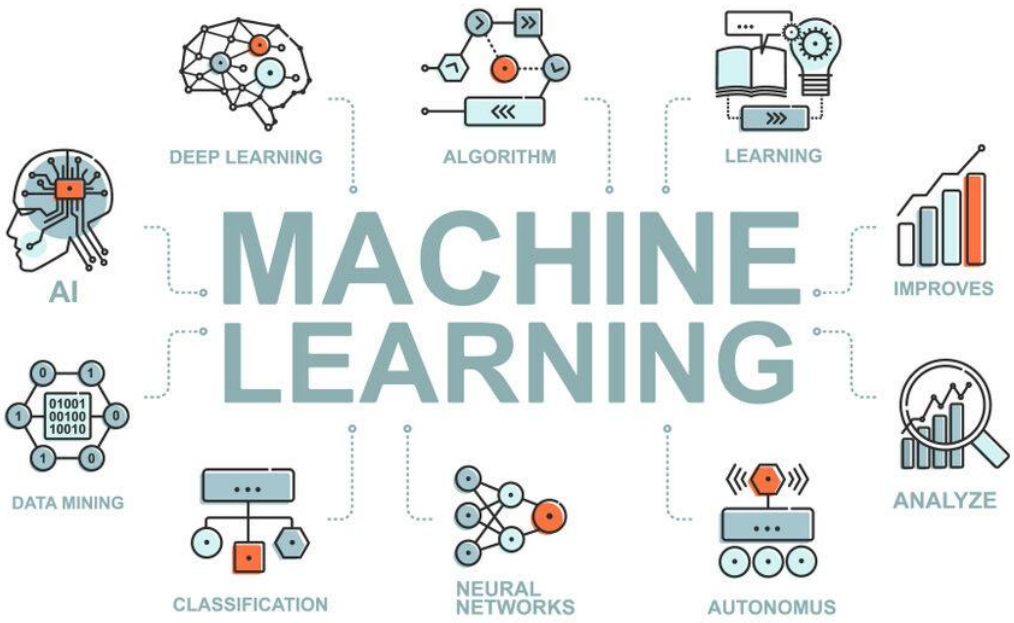
דוח סופי

מרצה: ד"ר חג'ג' חן

<u>מגישים:</u>

מרינה נז'לסקי

עדן כהן



1.1 מבוא

הפרויקט מתמקד בניתוח נתוני קווי נסיעות האוטובוסים במדינת ישראל בשנת 2024. במסגרת הפרויקט נעשה שימוש בלמידה מפוקחת לצורך סיווג קווים למטרופולינים לפי תכונות נבחרות, וכן בלמידה לא מפוקחת לצורך קיבוץ נתוני קווים בהתאם למאפיינים דומים.

1.2 בעיות ויעדים

הבעיה המפוקחת:

מערכת התחבורה הציבורית מורכבת ממסלולים ואוטובוסים רבים העוברים בין מטרופולינים שונים, ולכל מסלול מאפיינים ייחודיים כמו עיר מוצא, עיר יעד, סוג שירות, זמני פעילות, ותנועת נוסעים.
במצב כזה, השיוך של הנתונים למטרופולין מסוים יכול להיות מורכב, במיוחד אם מדובר במסלולים עם מאפיינים חופפים בין אזורים גיאוגרפיים שונים.
המטרה היא לפתח מודל לסיווג המטרופולין של כל קו בהתבסס על מאפייניו, ובעזרת זה יהיה ניתן להבין איזו ניהול מיטבי לכל מטרופולין.

בעיה לא מפוקחת:

במערכת התחבורה הציבורית, אוטובוסים פועלים בתנאים משתנים: שעות עומס שונות, רמות תפעול שונות, ותנועת נוסעים משתנה לאורך היום. ישנו קושי להבין אילו אוטובוסים מתאימים לאותן משימות או מאפיינים, מה שיכול להוביל לבזבוז משאבים ואי-התאמות בשירות.
המטרה בקיבוץ היא לזהות קבוצות או אשכולות של אוטובוסים שחולקים מאפיינים דומים על סמך נתונים קיימים, ולהשתמש בתובנות אלו לשיפור התפעול וההתאמה. חשיבות של דיוק הקיבוץ תוביל להבנה באפיון כל קבוצת קווי אוטובוסים שנמצא וממה נובעים ההבדלים בין אותן הקבוצות.

2.1 מערך הנתונים והמשתנים

מערך הנתונים "נסועה בקווי אוטובוס שנת 2024" נאסף מאגר המידע הפתוח: Data.gov.il. מערך הנתונים מכיל מידע על כל קווי האוטובוס בישראל בשנת 2024 ומכיל 53 תכונות לכל רשומה (סה"כ 10679 רשומות) הכולל מאפייני הקו, אזור פעילות הקו, מסלול הקו ומידע על נוסעי הקו.

2.2 הכנת הנתונים

1. **ניקוי נתונים:**

- על מנת לאתגר את הניתוח ואת מודל, הסרת עמודות קטגוריאליות הדומות לעמודת המטרופולין לפי ערך Cramer's V > 0.5 שזהו מדד סטטיסטי שמעריך את עוצמת הקשר בין שתי משתנים קטגוריאליים, כשהערך נע בין 0 (אין קשר) ל-1 (קשר חזק מאוד).
- הסרת עמודות מיותרות כמו RouteID ו-RouteName- עמודות אלה מכילות נתונים ייחודיים (מפתח) והן לא רלוונטיות למטרת הניתוח.
- הורדת שורות עם ערכים חסרים בעמודות BusType, AVGPassengersPerWeek ו-OperatingCostPerPassenger. הסרת שורות שבהן -Bus Size הוא 'לא מוגדר'.
- המרה של עמודות יום עבודה, שבת, שישי ותחנות ייחודיות לערכים בינאריים (שורות עם ערך יקבלו 1 שורות ללא ערך יקבלו 0).
- החלפת ערכים בעמודת BusType ('בינעירוני ממוגן אבן' ו-'בינעירוני ממוגן ירי' ל-'בינעירוני').
- טיפול בערכים בעברית- תרגום ערכים בעמודת Metropolin לאנגלית עבור נוחות העבודה.

2. **הנדסת תכונות (Feature Engineering)**

הוספת תכונות חדשות על סמך חישובים מעמודות אחרות:

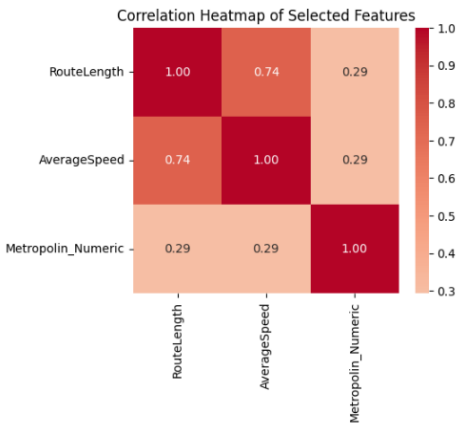
- חישוב עלות תפעול לנסיעה (OperatingCostPerRide)= עלות נסיעה לנוסע*מספר הנוסעים השבועי/מספר הנסיעות השבועי
- חישוב מרחק ממוצע בין תחנות (AVGdistanceBetweenStations)= אורך המסלול/ מספר תחנות במסלול

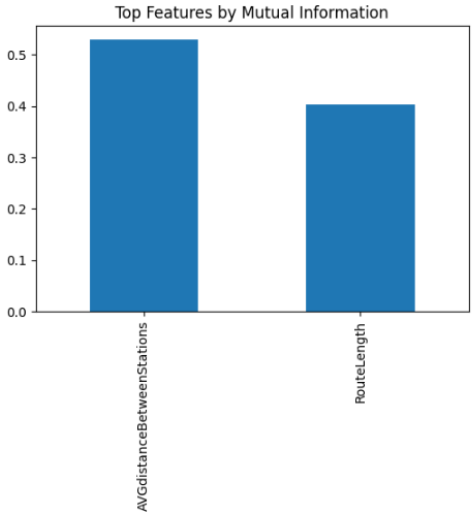
3. **Encoding**

- קידוד מספרי לערכי Metropolin באמצעות מספר סידורי.
- קידוד לערכי OperationSince לפי תדירות ההופעה.
- Label Encoding לעמודת MaxRidership.
- One hot Encoding לעמודות RouteParticular, BusType, BusSize, RouteType ו-ServiceType.

4. **סינון תכונות עבור Classification:**

- חישוב Mutual Information לבחירת תכונות - תכונות עם ערך 0.4<mi score נבחרו. (Mutual Information מודד את התלות והמידע המשותף בין משתנים, ו-MI Score משמש לבחירת תכונות רלוונטיות במודלים לפי ערך התלות ביניהן).
- חישוב מקדם המתאם בעזרת Person Correlation לבחירת תכונות עם ערך מתאם (המייצג קשר לינארי) גבוה 0.2 (קשר חיובי) או נמוך מ 0.2- (קשר שלילי) עם Metropolin_Numeric





5. **סינון התכונת עבור Clustering:**

- בדיקת מספר שורות לפי מטרופולין וסינון הדאטה ל-"Gush Dan"=Metropolin"
- סטנדרטיזציה של הנתונים המספריים.
- ביצוע PCA להורדת מימד הנתונים: המחשת תרומתם של המשתנים על שני רכיבי הPCA וסינון המשתנים עם תרומה של פחות מ0.2, על מנת להגיע לאחוז שונות כולל של 70%.
- יצירת דאטא-פריים חדש עם משתנים חשובים לצורך Clustering.

3.1 מתודולוגיה

למידה מפוקחת:

בחרנו להשתמש בכ-5 מודלי סיווג אשר מציעים יכולות שונות להתמודד עם מגוון אתגרים בסיווג, בהתאם לאופי הנתונים ומורכבות הבעיה. השונות ביניהם מאפשרת גמישות רבה במציאת הפתרון האופטימלי לבעיה ולנתונים:

- **Random Forest -** זהו מודל המבוסס על ידי כמה עצי החלטה, כאשר כל עץ מבצע סיווג באופן עצמאי והתוצאה הסופית היא החלטה משולבת מכל העצים כאשר התוצאה הסופית תהיה על פי החלטת ה"רוב". שיטה זו התאימה לנו מאחר ושיטה זו עמידה בפני Overfitting ושיטה וזו מבצעת סיווג בצורה מהירה.
- **Gradient Boosting -** זהו מודל המייצר עץ החלטה אחד בכל פעם, כאשר הוא מתמקס בטעויות שנעשו על ידי העצים הקודמים ומתקן אותם. שיטה זו התאימה לנו מאחר ושיטה זו אמורה לשפר משמעותית תוצאות על ידי תיקון טעויות.
- **Logistic Regression -** זהו מודל לינארי שמבצע חישוב של סכום לינארי של הפיצ'רים וממיר את התוצאה להסתברות באמצעות פונקציית לוגיסטית (סיגמואיד). שיטה זו התאימה לנו מאחר שמודל זה מאפשר אפשרות לפירוש קל של תוצאות המודל, מאחר ומדובר במודל לינארי שמסביר את הקשרים בצורה ברורה גם כשהסיווג כולל יותר מ2 קבוצות.
- **SVC -** זהו מודל המתמקד ביצירת גבול החלטה שמחלק את הנתונים בצורה אופטימלית, כך שיש מרווח גדול ככל האפשר בין נקודות הנתונים לגבול ההחלטה המפריד בין הקטגוריות. שיטה זו התאימה לנו מאחר שהוא מתאים לבעיות סיווג במקרים של גבול החלטה ברור, במיוחד כשיש מימד גבוה.
- **KNN -** זהו מודל המתבסס על חישוב המרחקים בין הדגימות הקיימות ומסווג את הדוגמה החדשה לפי הקטגוריה של ה-K השכנים הקרובים ביותר לה. שיטה זו התאימה לנו מאחר ולא דורשת הנחות מוקדמות על הנתונים, כמו לינאריות או לא לינאריות, והיא גמישה ויכולה להתאים לבעיות סיווג מרובות קטגוריות.

בנוסף, לכל מודל ביצענו התאמת סקלארים.הגענו למסקנה כי לKNN הסקילינג המתאים ביותר הוא MinMax ואילו לכל השאר Standard (ידוע שהמודלים RandomForest וGradient Boosting לא חייבים הגדרת סקלאר אבל למען האחידות השתמשנו בסקלאר זה).

למידה לא מפוקחת:

בחלק זה השתמשנו בנתונים המקודדים וביצענו סינון למטרופולין יחיד שהוא גוש דן המכיל 1164 רשומות (סינון הנתונים נועד על מנת להקל על מציאת האשכולות).
תחילה ביצענו סטנדריזציה לנתונים בעזרת StandardScaler כדי להביא את הנתונים לאותה סקאלה. זה חשוב במיוחד כאשר יש המון פיצ'רים בעלי ערכים שנמצאים בטווחים שונים. לדוגמה, אם עמודה אחת נמדדת בטווח של 0-1 ואחרת בטווח של מאות, התוצאה של PCA עלולה להיות מוטה לטובת המשתנים עם הערכים הגדולים יותר.

לאחר מכן, מכיוון ויש לנו מספר גדול מאד של תכונות, השתמשנו בשיטת ה-PCA אשר מצמצמת את מימדי הנתונים המורכבים בעזרת ניתוח שונות, כדי לזהות את הווקטורים בהם הנתונים משתנים בצורה מירבית ולבחור אותם כרכיבים עיקריים (Principal Components). בחרנו 2 רכיבים עיקריים על מנת שנוכל להציג את התוצאה בצורה ויזואלית שנוחה להסקת מסקנות.

תחילה קיבלנו שונות של 40% בין הווקטורים, ועל מנת לעלות את אחוז השונות, סיננו את המשתנים שפחות משפיעים על הווקטורים, והגדרנו שכל משתנה שתורם פחות מ0.21, יוסר מהניתוח. בצורה זו העלנו את השונות בין הווקטורים לכ-70%.

מודלי Clustering:

KMeans- בחרנו להשתמש במודל זה כדי לחלק את הנתונים לאשכולות. KMeans משתמש באלגוריתם של חישוב מרכזי האשכולות וממקסמת את המרחק בין המרכזים. בכדי לקבוע את מספר האשכולות האופטימלי השתמשנו בכמה קריטריונים:

- **Elbow Method:** אלגוריתם זה לוקח בחשבון את נקודת השבירה שבה יש שינוי חד בערכי Within-Cluster Sum of Squares. נקודה זו היא האינדיקציה למספר האשכולות האופטימלי.
- **Silhouette Score:** ציון זה מודד את איכות הקבוצים בכך שהוא בודק עד כמה הפרטים קרובים למרכז האשכול שלהם וכמה הם מרוחקים מאשכולות אחרים. ערכים גבוהים יותר מעידים על קבוצים טובים יותר.

Gaussian Mixture Model (GMM)- מודל זה מאפשר קבוץ של נתונים במבנים לא לינאריים ומסתמך על הנחת עבודה לפיה הנתונים מגיעים מתערובת של כמה התפלגויות גאוסיאניות.

- BIC, AIC- קריטריונים אלו משמשים לבחירת המודל האופטימלי ע"י מיזעור הפרמטרים ושמירה על המודלים הפשוטים ביותר.
- Silhouette Score: נחשב גם במקרה של GMM כדי לוודא את איכות הקבוץ.
- Elbow Method (Second Derivative of Log Likelihood): זוהה מספר האשכולות האופטימלי על ידי חישוב ההפלה השנייה של ה-log likelihood, המצביעה על נקודת השינוי החדה בתוצאה.

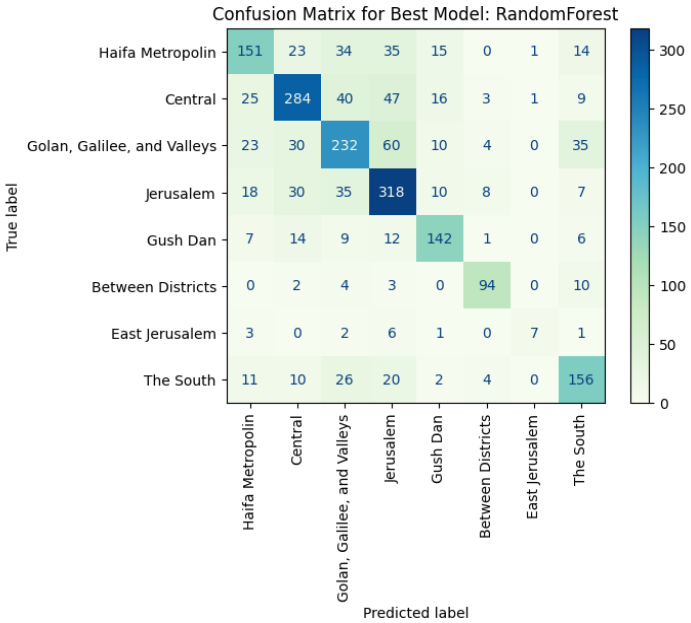
4. ניסויים ותוצאות:

למידה מפוקחת:

בתחילת הניסוי, הנתונים חולקו לסט אימון וסט בדיקה ביחס של 80:20 באמצעות הפונקציה train_test_split. חלוקה זו נועדה לאפשר למודל ללמוד מהנתונים ולבדוק את ביצועיו על נתונים חדשים שלא שימשו באימון.

המודל שהשיג את הביצועים הטובים ביותר היה RandomForest עם דיוק ממוצע של 71.8% לפי הצלבה (Cross-Validation). אלגוריתם זה הראה יציבות והצליח להבחין בצורה טובה בין המטרופולינים השונים.

נעשה שימוש ב-Cross-Validation עם 5 קיפולים (fold=5), המבטיח הערכה יציבה על פני קיפולים שונים של הנתונים. הבחירה ב-5 קיפולים שומרת על מהירות חישוב ומספקת מדידה אובייקטיבית יותר לביצועים(כלומר, לצורך העניין 10 יכולים להגדיל את הדיוק אך דורשים יותר זמן חישוב ואילו 3 עלולים להביא לחוסר יציבות בתוצאה).



- **דיוק (Precision):** המודל הצליח להימנע מטעויות רבות בזיהוי המטרופולינים "גוש דן" (5) ו"בין מחוזי" (6), עם דיוק של 80% ו- 74% בהתאמה.
- **רגישות (Recall):** רגישות גבוהה הייתה במטרופולינים "בין מחוזי" (6) -85%, אך נמוכה מאוד במטרופולין "מזרח ירושלים"(7)- 30%, מה שמעיד על קושי בזיהוי המטרופולין הזה.
- **מדד F1:** מדד זה, שמאזן בין דיוק ורגישות, היה 68% במטרופולין "ירושלים (4), מה שמעיד על איזון סביר בתחזיות עבורו.

דיוק כולל

הדיוק הכולל של המודל על סט הבדיקה היה 67%, אך הוא משתנה בין המטרופולינים השונים. הממוצע המאקרו (Macro Avg) היה 65%, והשוקלל 66% (Weighted Avg), דבר המעיד על ביצועים פחות טובים במטרופולינים בעלי דוגמאות מועטות.

ניתוח הצלחה ומגבלות

- הצלחות: המודל הראה ביצועים טובים במטרופולינים "גוש דן" ו"בין מחוזי" עם דיוק ורגישות גבוהים.
- מגבלות: המודל התקשה בזיהוי המטרופולין "ירושלים", עם רגישות נמוכה מאוד (30%).

השוואת ביצועים בין המודלים

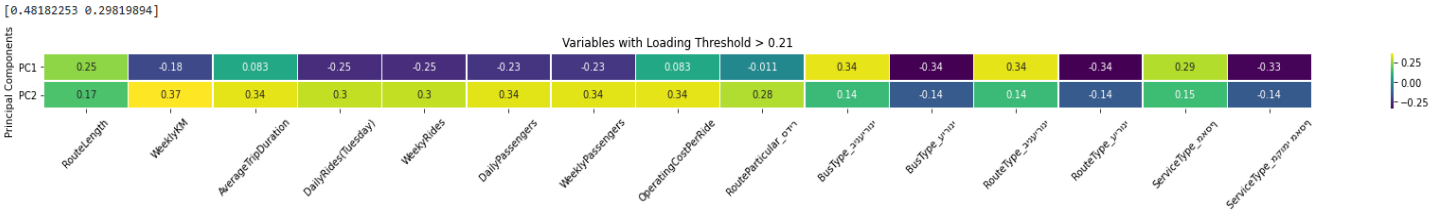
ביצועי המודלים השונים משתנים בהתאם למורכבות ולמבנה הנתונים:

- **Random Forest:** הראה יציבות גבוהה ודיוק טוב בזיהוי המטרופולינים עם מאפיינים ברורים. הוא מספק איזון טוב בין מורכבות לביצועי מודל, מתקדם עם סוגים שונים של נתונים, יציבות גבוהה ויכולת להתמודד עם נתונים לא לינאריים בצורה טובה יותר ממודלים אחרים.
- **Logistic Regression**: השיג דיוק נמוך יחסית עקב הנחת הקווים הלינאריים במידע שלא מתאים למורכבות הנתונים. מתאים יותר לנתונים לינאריים, לכן פחות מתאים במידה והמידע יותר מורכב.
- **SVC**: דיוק בינוני, אך רגיש מאוד להבדלים בסקלות בין המאפיינים, מה שמחייב סקלור מדויק. מתאים במיוחד לזיהוי גבולות ברורים בין קטגוריות בנתונים בעלי מימד גבוה, אך דורש סקלור נכון.
- **KNN**: סבל מהערכת מרחקים בין הדגימות, דבר שדורש סקלור מדויק של המידע. מודל תלוי על חישובי מרחקים, המושפעים מקרבה יחסית בין נקודות בנתונים וסקלור נכון.

למידה לא מפוקחת:

PCA

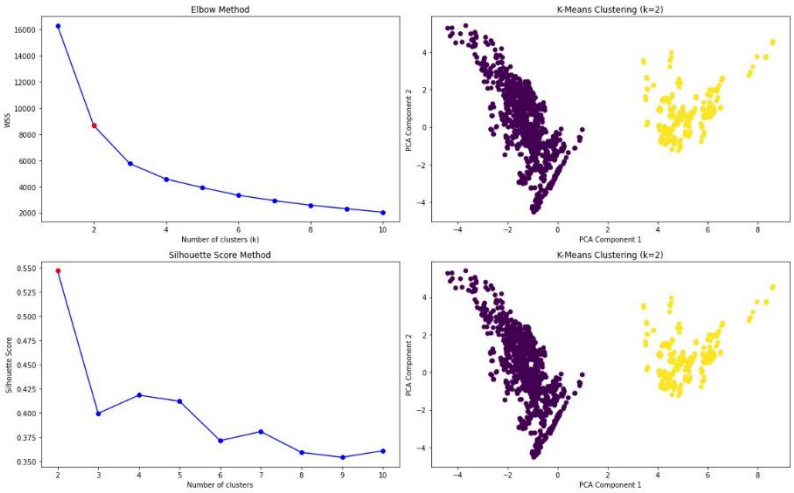
בניתוח השונות הראשון בו נלקחו כל המשתנים התקבלה שונות מוסברת של 38% [0.256,0.128]. אחוז שונות גבוה יותר מאזן בין שמירה על מידע משמעותי מהנתונים לבין הפחתת המורכבות והרעשים, מה שמשפר את יעילות החישובים ואת ביצועי המודלים. על מנת להגיע לאחוז שונות מוסברת שגדול מ70%, ביצענו מספר ניסויים להתאמת סף השפעה מסויים, כדי לסנן את כל המשתנים שתורמים הכי פחות לשונות הווקטורים, ומצאנו שסף השפעה של 0.2 גורם לשונות של 77% [0.481,0.298].



קיבלנו 2 רכיבים עיקריים:

- **PC1** - משקף את המשתנים: RouteLength (אורך מסלול), BusType_בינעירוני (סוג אוטובוס בינעירוני), ו- RouteType_בינעירוני (סוג קו בינעירוני)
- **PC2** - משקף את המשתנים: OperatingCostPerRide (עלות תפעול לנסיעה), WeeklyKM (קילומטרים שבועיים), ו- DailyPassengers (נוסעים יומיים)

K-Means



כדי למצוא את מספר האשכולות האופטימלי, רשמנו אלגוריתם שמציג את מדדי ה Elbow, Silhouette בצורה גרפית, ובוחר באופן אוטומטי את התוצאה הטובה ביותר (הנקודה האדומה בגרפים השמאליים).

גם בשיטת **Elbow** וגם במדד **Silhouette**, המספר האופטימלי של אשכולות שנמצא הוא **2**.

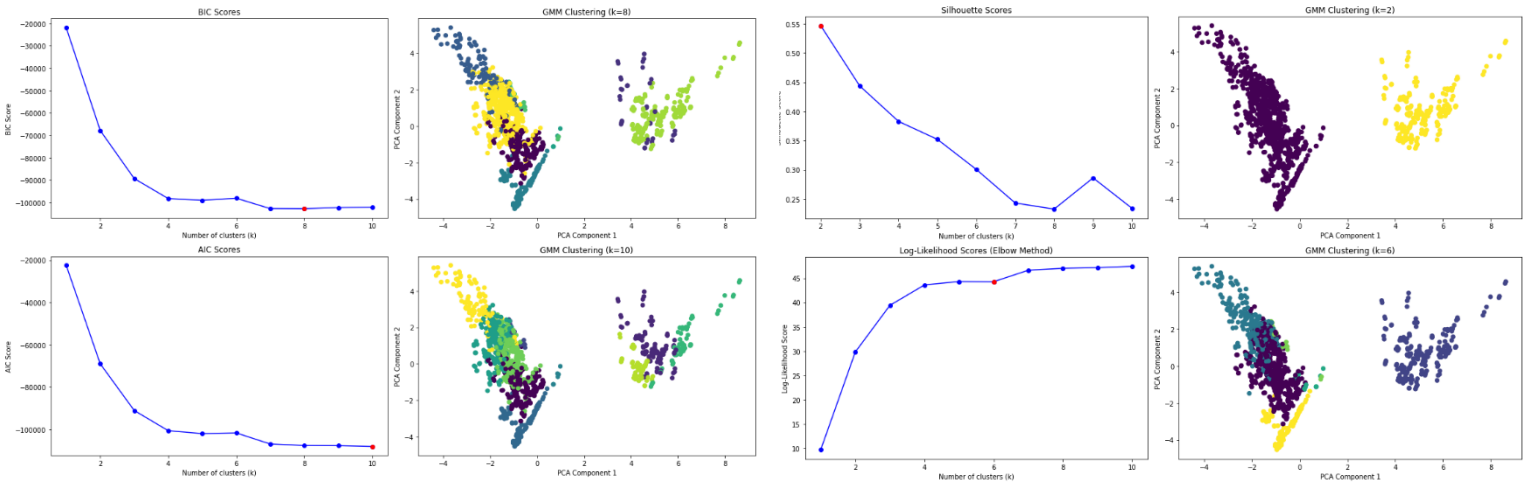
GMM

התוצאה עבור AIC היא 10 אשכולות עם חפיפה AIC. מדד זה שואף למצוא את המודל הטוב ביותר מבחינת הסבר הנתונים תוך שמירה על פשטות, אך החפיפה בין האשכולות מעידה על מודל מורכב.

התוצאה עבור BIC היא 8 אשכולות עם חפיפה BIC. דומה ל AIC, אך עם קנס חזק יותר עבור מודלים מרובי פרמטרים. גם כאן, החפיפה מצביעה על מודל מורכב.

Silhouette Score מראה תוצאה של 2 אשכולות עם הפרדה ברורה. מדד זה בודק את איכות הקיבוץ לפי אחידות בתוך האשכול והפרדה ביניהם, וציון גבוה מעיד על קיבוץ איכותי.

Elbow Method (Log-Likelihood) מצביע על 6 אשכולות. שיטה זו בודקת שינוי בסבירות המודל לעומת מספר האשכולות. התייצבות בלוג-לייקליוד ב-6 אשכולות מעידה על הנקודה שבה שיפור המודל מתחיל להתייצב, כלומר זהו המספר האופטימלי של אשכולות.



איפיון האשכולות

תחילה נאפיין את רכיבי הPCA שקיבלנו:

- ב-PC1 ערכים גבוהים למשתנים כמו RouteLength, BusType_בינעירוני, וRouteType_בינעירוני, מצביעים על קווים ארוכים ובינעירוניים.
- ב-PC2 ערכים גבוהים למשתנים כמו OperatingCostPerRide ו-WeeklyKM, מצביעים על קווים עם קילומטרג' שבועי גבוהה יותר שיכולים להיות יקרים יותר, ועם עלויות תפעול גבוהות.

על מנת לאפיין את האשכולות, נבחר במדדים שהצביעו על 2 אשכולות, מכיוון שאנו רואים הפרדה ברורה וניתן יהיה לאפיין בקלות את האשכולות.

- **אשכול 1 (הסגול)-** תחום על ציר PC1 בין (1 , 5-) ו-PC2 בין (5,5-) , מצביע על כך שמדובר בקווים עירוניים בגוש דן (עירוניים בדרך כלל יהיו בעלי אורך מסלול קטן יותר) וייתכן גם עלויות תפעול נמוכות יותר ותחבורה עם תדירות גבוהה.
- **אשכול 2 (הצהוב) -** תחום על ציר PC1 בין (3,8) ו-PC2 בין (2,4-) מצביע על קווים בינעירוניים בגוש דן (כיוון שיש כאן אורך מסלול גבוה וייתכן שעלות התפעול גבוהה יותר).

5. סיכום ומסקנות

בפרויקט זה עבדנו בשיתוף פעולה מלא, והתמקדנו בניתוח נתוני מסלולי אוטובוס לשנת 2024 תוך שימוש בטכניקות למידת מכונה מפוקחת ובלתי מפוקחת. במסגרת הלמידה המפוקחת, בחרנו להסיר עמודות הדומות לעמודת המטרופולין עם ערך Cramer's V > 0.5 כדי לצמצם את מספר התכונות ולהתמקד במידע המשמעותי ביותר. התמודדנו עם נתונים חסרים באמצעות השלמה בממוצעים או ערכים תדירים. בנוסף, ביצענו קידוד משתנים קטגוריאליים בשיטות Label Encoding ו One-Hot Encoding,-ולאחר מכן בחרנו בעזרת Feature importance analysis וגם Pearson Correlation את התכונות המתאימות ביותר לסיווג המטרופולין כמו Route Length, AVG distance Between Stations ו Average Speed. הפעלנו מודל Random Forest שהשיג דיוק של 72% בסיווג קווי האוטובוס לפי אזור המטרופולין. בניסיון לשפר את התוצאות, השתמשנו גם באלגוריתמים נוספים כמו Logistic Regression, SVC, Gradient Boosting ו KNN,-אך אלו הניבו תוצאות פחות טובות במדדי הערכה שונים.

במסגרת הלמידה הבלתי מפוקחת, ביצענו צמצום מימדים באמצעות PCA כדי לסנן ולדרג את התכונות לפי חשיבותן ולהתמקד באלו המהותיות ביותר. לאחר מכן, ביצענו קיבוץ מסלולי אוטובוס באמצעות האלגוריתמים K-Means ו GMM-על מנת לאפיין ולזהות אשכולות עם מאפיינים משותפים. הבחירה במספר האשכולות האופטימלי נעשתה באמצעות קריטריונים כמו Elbow Method ו-Silhouette Score.

כיווני מחקר עתידיים

ישנם מספר כיוונים עתידיים שניתן לחקור ולפתח על בסיס הממצאים והלמידה מהפרויקט הנוכחי:

1. להמשיך ולשפר את הדיוק במודל הסיווג תוך בחינת מודלים נוספים או הרצת נוספים (Hyperparameter Tuning) על הדאטה הקיים.
2. למצות תכונות חדשות ושיפור תהליך Feature Engineering במטרה לשפר את הביצועים של המודל הקיים.
3. לבחון גם טכניקות מתקדמות יותר של למידת מכונה עמוקה (Deep Learning) כדי לבדוק אם ניתן להשיג שיפור נוסף בביצועי המודלים.
4. להשתמש בממצאים והאשכולות שהתגלו במחקר כדי לייעל את תפעול התחבורה הציבורית תוך התאמת השירות לפי הנתונים על הביקוש והיצע במסלולים מסוימים.