מערכות לומדות תרגיל 4

שם: אסף חייק ברוך **ת.ז:** 206783441

שם: בן בנוז **ת.ז:** 207570573

בתרגיל זה הוצבו בפנינו 3 בעיות:

1. מציאת קואליציה יציבה כך שיותר מ-51% מהמצביעים הצביעו למפלגות המרכיבות אותה והמצביעים שהצביעו להם קרובים יחסית זה לזה אבל שונים מאוד ממצביעי מפלגות האופוזיציה
2. זיהוי תכונות שעל ידי מניפולציה שלהן נוכל להכריע את מנצחת הבחירות
3. זיהוי תכונות שעל ידי שינוי שלהן נוכל ליצור קואליציה יציבה כרצוננו.

דרכי השגת הפיתרון לבעיות:

ראשית כפי שצוין בתרגיל השתמשנו ב-generative models וב-clustering models כדי להגיע לפתרונות הרצויים, לצורך התרגיל גם עשינו שימוש במודל מהתרגיל הקודם עם היפר פרמטרים מסוימים שהראה את הperformance הטוב ביותר בקשר לבעיית חיזוי תוצאות הבחירות (היסטוגרמת התוצאה הסופית), וכמובן השתמשנו במידע המעובד מהתרגילים הקודמים.

ראשית את בעיית הקמת הקואליציה היציבה תקפנו ב-2 דרכים:

א. דרך clustering

ראשית אנחנו משתמשים ב-2 שיטות clustering כדי לפלג את המצביעים של כל מפלגה ל-3 קלאסטרים לשם כך השתמשנו בשיטות ה clustering הבאות:

*MiniBatchKMeans- כלומר אלגוריתם kmeans שנלמד בהרצאה (עם שינוי קטן שמוסיף מהירות ושלא משנה תוצאות באופן משמעותי)

*BayesianGaussianMixture – גרסא יותר יציבה של GMM

לאחר מכן השתמשנו ב-clusters שכל שיטה הוציאה בתור labels לאימון מודל על ה-features המקוריים, בעצם אימנו 2 מודלים מסוג RandomForestClassifier, כאשר כל מודל מתאמן על תכונות המצביעים בתור features ועל הקלאסטרים שכל שיטה הוציאה בתור labels (זה על כל הקלאסטרים שמצאנו אצל המפלגות השונות ביחד).
ההיפר פרמטרים שבחרנו הם אלה שמצאנו בתרגיל הקודם.

לאחר מכן השתמשנו בפונק' predict_proba של כל אחד מהמודלים על סט המצביעים כך שהמודלים מחזירים את ההסתברות של כל מצביע להיות שייך לכל אחד מהקלאסטרים, והשתמשנו בdot product בין וקטורי ההסתברויות של כל 2 מצביעים כמדד לקרבת 2 המצביעים זה לזה, ועל ידי כך יצרנו מטריצת קישוריות שהיא בעצם מטריצה בממדים n_samples*n_samples כך שמקום הi,j בה יש את רמת הקרבה (dot product) של המצביע ה-i למצביע ה-j (המטריצה הנ"ל סימטרית).
לפני מכפלת הוקטורים החלטנו לאפס את כל ההסתברויות שקטנות מערך מסוים (בחרנו 0.3) בכדי לחסוך בזמן ריצה ולוודא שהדימיון נחשב רק החל מערכים שמביעים רמת ביטחון סבירה של השייכות.

כעת נשתמש במטריצת הקרבה בין samples כדי לבצע אלגוריתם clustering סופי שיחלק את המצביעים למצביעי קואליציה ולמצביעי אופוזיציה, אלגוריתם ה clustering שהשתמשנו בו הוא AgglomerativeClustering , זה בעצם אלגוריתם שמאתחל כל דוגמא לקלאסטר משלה ולאחר מכן מאחד קלאסטרים לפי המרחק האוקלידי ביניהם לפי מטריקה מסויימת (אנחנו בחרנו אחת שמצמצמת variance של כל קלאסטר), הוא משתמש במטריצת הקרבה שהגדרנו כדי להעריך עד כמה הדוגמאות נחשבות דומות בנוסף למטריקה הרגילה (השימוש הוא כמטריצת connectivity בין הדוגמאות). הפלט הוא 2 קלאסטרים שונים.
מתוך הקלאסטר הגדול מהשניים ניקח את כל המפלגות שיותר מ-75% ממצביעיהן בו, והן יהיו הקואליציה בעוד שהשאר יהיו באופוזיציה.

ב. דרך generative models

בדרך זו השתמשנו ב generative models הבאים:

- LinearDiscriminantAnalysis - מפריד לינארי (LDA).
- QuadraticDiscriminantAnalysis - מפריד ריבועי (QDA).

ראשית אימנו כל אחד מהמודלים הללו על סט המצביעים שלנו כאשר המפלגה לה הצביעו היא label.

לאחר מכן נייצר מכל אחד מהם קואליציה בצורה הבאה:

1. נשתמש במדד Jensen-Shannon שהינו מדד לקרבה בין שתי התפלגויות, במילים אחרות נמדוד קרבה בין מצביעי מפלגות על ידי קרבת ההתפלגות שלהם.

   לגבי Jensen-Shannon:

כל אחד מהמודלים הנ"ל שומר אצלו חישוב של ממוצע הדוגמאות של כל מחלקה ואת השונות של כל סט המידע עליו התאמן.

נשתמש במידע זה כדי לדגום סט דוגמאות כל פעם של מצביעים עבור כל 2 מפלגות ונשתמש בהם כדי לחשב את מדד Jensen-Shannon לגבי הקרבה בין הקבוצות.

2. לאחר שחישבנו את המדד עבור כל שתי מפלגות נתחיל לאחד מפלגות שמצביעיהם קרובים זה לזה עד אשר נקבל קואליציה כאשר נעצור לאחר שקיימת קבוצת מפלגות שיש לה יותר מחצי מהמצביעים והאיחוד הבא לא ישפר את מדד Davies-Bouldin שלה.
   Davies-Bouldin הוא מדד עבור קלאסטרים שמציג את רמת הקרבה של דוגמאות שנמצאות באותו קלאסטר וכמה דוגמאות מקלאסטרים שונים שונות זו מזו, וכך נמשיך להוסיף עוד מפלגות לקואליציה עד אשר לא נקבל שיפור במדד.
3. נחזיר קואליציה עבור כל אחד מה- הgenerative models .

לבסוף נעבור על כל הקואליציות שקיבלנו מכל השיטות וננקד את טיב הפירוק לקואליציות על פי מדד Davies-Bouldin וניקח את הטובה ביותר.

בתוצאות שלנו יצא כי GMM הוציא את הקואליציה הכי יציבה, עם ניקוד Davies-Bouldin של 4.77. הקואליציה מורכבת מהמפלגות הבאות:
Browns, Purples, Violets, Greens, Whites, Oranges, Pinks, Blues, Greys, Turquoises

אנחנו ממליצים להריץ את קובץ ה-main איך שהוא בהגשה בכדי לראות את פילוג הקלאסטרים, המפלגות והקואליציות השונות בתלת מימד לפי ערכי PCA.

מציאת התכונות שיקבעו את מנצחת הבחירות ואת הקואליציה

שינוי המנצחת בבחירות על ידי מניפולציה של תכונות הבוחרים

את הבעיה הזו פתרנו על ידי בדיקת שינוי התכונות לטווחים שונים על התפלגות קולות הבחירות שהמודל שלנו מהתרגיל הקודם זוכר, כלומר מהתרגיל הקודם יש ברשותנו מודל שחוזה בדיוק רב את התפלגות הקולות בין המפלגות בבחירות, נאמן אותו על סט המצביעים ואז נשנה את התכונות של המצביעים לטווחים מסוימים שאנחנו בודקים ונבדוק מה המודל חוזה על סט המצביעים הנ"ל לגבי התפלגות הקולות בין המפלגות, ולאחר מכן נבדוק את השינוי שחל בין תוצאות האמת (היסטוגרמת ההצבעה למפלגות של סט המידע המקורי) ולפיו נראה מה השפעתו על תוצאות הבחירות.
בתוך הטווחים הגרלנו ערכים אקראיים באופן אחיד, ולקחנו את הממוצע של החיזוי מתוך 50 הגרלות שונות.

בתוצאות האמת מפלגת הסגולים אמורה לנצח עם כרבע מכלל הקולות.
סיכום של התוצאות נמצא בקובץ observations.txt.

בצורה זו שמנו לב לתכונות הבאות:

אם ברצוננו לחזק את מפלגת הסגולים בתור המנצחת נדאג לאחד מהדברים הבאים:

- שהתכונה avg_enviermental_importance תהיה נמוכה כי כאשר היא בטווחים נמוכים היא זוכה ל-45% מכל הקולות.
- שהתכונה Avg_government_satisfaction תהיה בטווח נמוך או ממוצע בין 0 ל-700 כי אז היא זוכה בין 23% ל-44% מהקולות ומנצחת בבחירות.
- שהתכונה Avg_monthly_expense_on_pets_or_plants תהיה בטווחים 0 עד 500 כי אז היא זוכה בין 30% ל-38% מכל הקולות ומנצחת בבחירות
- שהתכונה Yearly_ExpensesK תהיה בכל טווח שגדול מ-4000 כי אז היא לוקחת בין 20 ל-45% מכל הקולות ומנצחת.
- שהתכונה number_of valued_kneset_members תהיה 6 ומעלה כי אז היא מקבלת בין 25% ל-44% מהקולות ומנצחים בבחירות
- שהתכונה Weighted_education_rank תהיה בין הערכים 14 ל-700 כי אז היא מנצחת עם 25% עד 38% מכלל הקולות.

אם אנחנו רוצים להחליף את מנצחת הבחירות למפלגת החאקים נדאג לאחד מהדברים הבאים:

- שהתכונה Avg_government_satisfaction תהיה בטווח גבוה של בין 900 ל1000 שכן אז הם זוכים ב-51% מקולות המצביעים ומנצחים את הבחירות בוודאות
- שהתכונה Avg_monthly_expense_on_pets_or_plantsn תהיה בטווח גבוה של 500 ומעלה כי אז היא זוכה ב-23% עד 32% מהקולות ומנצחת בבחירות.

אם אנחנו רוצים להחליף את מנצחת הבחירות למפלגת החומים נדאג לאחד מהדברים הבאים:

- שהתכונה Yearly_ExpensesK תהיה בטווח ממש נמוך בין 3000 ל-4000, שכן אז היא זוכה בבחירות עם 20% מהקולות.

אם אנחנו רוצים להחליף את מנצחת הבחירות למפלגת הלבנים נדאג לאחד מהדברים הבאים:

- שהתכונה number_of valued_kneset_members תהיה בטווח של בין 0 ל-5 שכן אז היא מנצחת בבחירות עם 30% עד 45% מכלל הקולות.

אם אנחנו רוצים להחליף את מנצחת הבחירות למפלגת הכתומים נדאג לאחד מהדברים הבאים:

- שהתכונה Weighted_education_rank תהיה בטווח ממש גבוה של 700 ומעלה כי אז היא זוכה עם 22% עד 30% אחוזים מכלל הקולות.

יצירת קואליציה חזקה יותר על ידי מניפולציה של תכונות הבוחרים

את הבעיה הזו פתרנו על ידי אימון מודל randomForest על סט המצביעים המקורי , ולאחר מכן נתנו לו לחזות את תוצאות הבחירות על סט מצביעים שביצענו עליו מניפולציה, ואז השתמשנו בהתפלגות הקולות שהוא חזה כדי לייצר קואליציה לפי שיטת ה-clustering שהצגנו בסעיף א עם שימוש בשיטת הclustering -BayesianGaussianMixture , לאחר מכן בחנו את גודל ויציבות הקואליציה שנבחרה לפי מדד Davies-Bouldin.

לאחר בחינת התכונות כפי שתוארה קודם הגענו למסקנות הבאות לגבי האפשרות להרכבת קואליציה יציבה כרצוננו:

```
# change the data
changed_data["Avg_environmental_importance"] = changed_data["Avg_environmental_importance"] * 0.8
changed_data["Avg_education_importance"] = changed_data["Avg_education_importance"] * 1.2  # less
changed_data["Avg_monthly_expense_on_pets_or_plants"] = changed_data[
                                                        "Avg_monthly_expense_on_pets_or_plants"] * 0.4  # less
changed_data["Yearly_ExpensesK"] = changed_data["Yearly_ExpensesK"] * 0.8
```

כפי שניתן לראות בתמונה הנ"ל הורדנו ב20% את רמת ההתחשבות בסביבה ואת ההוצאות השנתיות, והורדנו ב60% את ההוצאה החודשית הממוצעת על חיות או צמחים, והעלנו את רמת הדאגה לחינוך ב20%, תכונות שרשום לידן בהערה less הם תכונות שהשפיעו במידה קטנה יותר על חוזק הקואליציה אבל בכל זאת יותר משאר התכונות שלא שינו את חוזק הקואליציה בצורה משמעותית.

הקואליציה המקורית שלנו הייתה עם ניקוד 4.77 והכילה את:

Browns, Purples, Violets, Greens, Whites, Oranges, Pinks, Blues, Greys, Turquoises

בעוד שלאחר המניפולציות קיבלנו ניקוד 3.63 ואת המפלגות:

Yellows, Oranges, Pinks, Purples, Khakis, Greens, Greys, Reds, Violets