תרגיל בית 5 מבוא למערכות לומדות

דוח עבודה

מגישים:

אביב כספי – 311136691

יקיר יהודה - 205710528

<u>שלבי עבודה:</u>

1. **Pre-processing:**

דבר ראשון שעשינו היה ביצוע עיבוד ראשוני לדאטה שלנו.
מימוש חלק זה נמצא בקובץ preprocessing.py.
את הפעולות הבאות ביצענו על הדאטה הישן (הסט שמכיל את הלייבלים):

א. המרה לone-hot של הפיצ'רים הנומינליים ללא חשיבות לסדר :
מתוך הפיצ'רים שנבחרו היה עלינו לשנות רק את 'Most_Important_Issue' ל-one-hot.

ב. המרת כל הפיצ'רים הקטגוריאליים למספריים:
על מנת שנוכל לעבוד על הדאטה, עלינו לשנות את כל הפיצ'רים שיהיו מספריים.

ג. תיקון ערכים שליליים בפיצ'רים:
בדומה לתרגיל קודם, גם כעת ישנו פיצ'ר בעל ערכים שליליים למרות שלא ייתכן כי הערכים של הפיצ'ר יהיו שליליים.
בצורה דומה לתרגיל הקודם, התמודדנו עם בעיה זו על ידי ביצוע abs על הפיצ'ר הנ"ל. (כמו בתרגיל קודם, זיהינו כי התפלגות הדאטה השלילי מתאימה בדיוק להתפלגות הפיצ'ר בערכים החיובים, לכן החלטנו כי ייתכן כי בזמן הדגימה שונה הסימן בטעות, לכן החלטנו רק לבצע ערך מוחלט על הדאטה הנ"ל)

ד. לאחר מכן ביצענו פיצול לדאטה שלנו:
בדומה לתרגיל הקודם בחרנו את הפיצול הבא – train(70%), val(15%), test(15%)

ה. מחיקת outliers:
בשלב זה מצאנו את הדוגמאות בהן ערך הפיצ'ר במרחק גדול מ4.5 סטיות תקן מהממוצע של הפיצ'ר, והצבנו NAN במקומות אלה.
פעולה זו ביצענו רק על סט האימון, והשתמשנו בלייבלים של הדאטה על מנת למצוא outliers יחסית ללייבל.

ו. Imputation :
השלמת ערכים חסרים ביצענו בנפרד לסט האימון ולסט הולידציה והבדיקה -
עבור סט האימון בחרנו בשיטה הנקראת Bootstraping שלמדנו בכיתה.
בשיטה זו, עבור כל ערך חסר, השלמנו על ידי כך שדגמנו מתוך הדאטה שלנו עם חשיבות ללייבל, ערך חדש.
עבור סט הולידציה והבדיה בחרנו לטפל בצורה שונה.
תחילה לקחנו את סט האימון, ועבור כל פיצ'ר חישבנו את ממוצע הפיצ'ר עבור ערכים נומריים ואת הערך הנפוץ ביותר עבור ערכים נומינליים (ללא חשיבות ללייבל).
לאחר מכן, השלמנו את סט הולידציה והבדיקה בעזרת הערכים שחושבו מסט האימון.

ז. Scaling:
עבור פיצ'רים יוניפורמים ביצענו scaling לטווח [1 -1]
עבור פיצ'רים נורמלים ביצענו נורמליזציה עם ממוצע 0 וסטיית תקן 1.

עבור הסט החדש, עליו צריך לבצע חיזוי ביצענו את אותם הפעולות כמו שביצענו על ה test set בחלק זה, כלומר:

- המרה ל-onehot של הפיצ'ר Most_important_issue
- המרת הפיצ'רים הקטגוריאליים למספריים.
- תיקון ערכים שליליים בפיצ'רים.
- Imputation לפי מידע מסט האימון.
- Scaling לפי מידע מסט האימון.

2. **מציאת ערכי hyperparameters עבור כל מודל:**

השלב הבא היה לאמן מודלים על סלט האימון ולבחור את המודל הטוב ביותר, על מנת לבצע זאת תחילה חיפשנו עבור כל מודל את ההיפר-פרמטרים הטובים ביותר עבורו על ידי ביצוע cross validation .
מודלים שבחרנו לבדוק הינם:
MLPClassifier, RandomForestClassifier, SVC, VotingClassifier

בחרנו להתמקד במודלים שלמדנו בכיתה וכאלה שהביאו ביצועים טובים בתרגיל בית הקודם.
בנוסף, בחרנו להוסיף מודל של ועדה, אשר ישתמש בשלושת המודלים על מנת לבצע החלטה.
מודל זה בחנו בשתי שיטות, אחת hard voting והשנייה soft voting. כלומר, המודל הראשון יבצע הצבעה בין שלושת המודלים הפנימיים ויחליט על פי רוב בהצבעה, והמודל השני יבצע הצבעה ממושקלת על פי הביטחון של כל מודל בהחלטה שלו, על ידי שימוש בפונקציה predict_proba.
היפר הפרמטרים הטובים ביותר עבור כל מודל שמצאנו הינם:

RandomForestClassifier: n_estimators: 220, min_samples_split: 6
SVC: kernel: 'rbf'
MLPClassifier: 'hidden_layer_sizes': (50,100), 'max_iter': 1000, ‘activation’: ‘tanh’, ‘learning_rate’='adaptive'

3. **אימון המודלים והערכת ביצועים:**

בשלב זה ביצענו אימון לכל אחד מהמודלים שציינו, עם ההיפר פרמטרים שנמצאו בסעיף הקודם.
כל מודל אימנו על סט האימון, וביצענו הערכת ביצועים על סט הולידציה לפי accuracy_score.
בנוסף למודלים הנ"ל, אימנו את מודלי הvoting בשימוש בשלושת המודלים הנ"ל עם הפרמטרים שנמצאו.

לבסוף הסתכלנו על כל התוצאות שקיבלנו ובחרנו בצורה ידנית את המודל המוצלח ביותר.
תוצאות האימונים שקיבלנו עבור המודלים הינם:

RandomForestClassifier : 92.2%
SVC : 89.267%
MLPClassifier : 92.867%
VotingClassifier - hard voting: 92.267%
VotingClassifier – soft voting: 93.93%

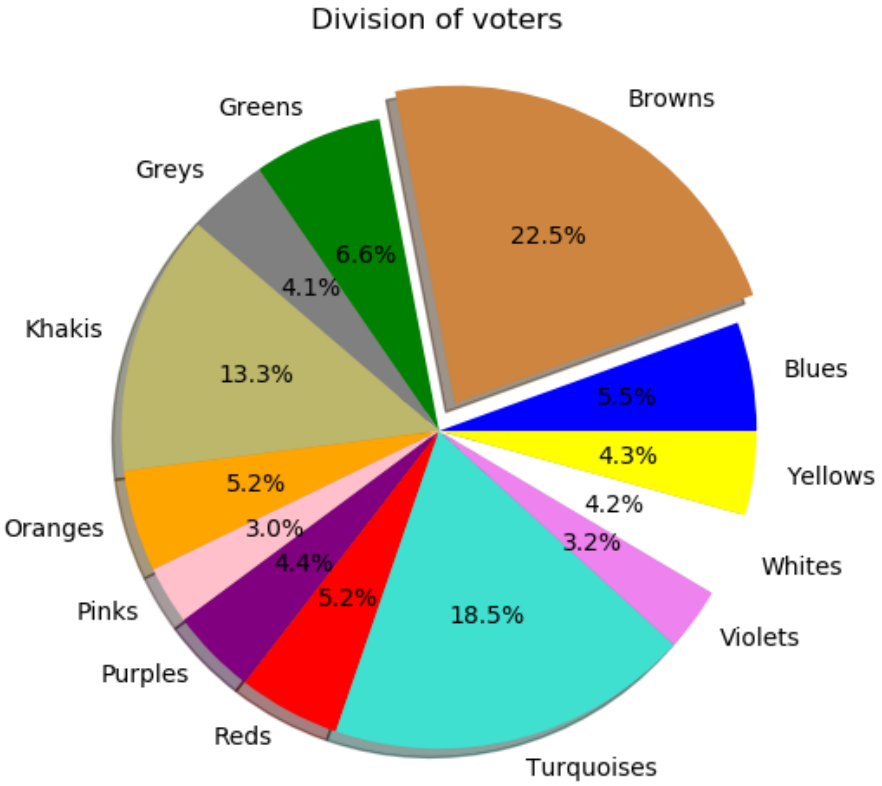
המודל שקיבל את הביצועים הטובים ביותר היה המודל VotingClassifier – soft voting עם כמעט 94% דיוק על סט הולידציה.
לכן בחרנו להמשיך עם מודל זה ולבצע בעזרתו את החיזוי.

4. אימון מודל נבחר וחיזוי המשימות:

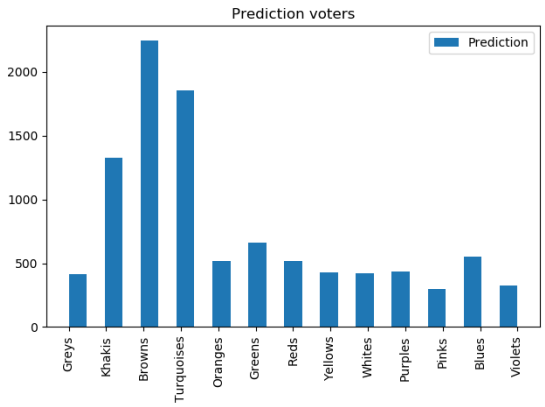
לאחר בחירת המודל המוצלח ביותר, ביצענו אימון שלו בעזרת סט האימון והולידציה יחדיו (בחרנו להשתמש בסט הולידציה לאימון מפני שכעת אין צורך לבצע ולידציה נוספת).
הדבר האחרון שביצענו לפני חיזוי המשימות, היה בדיקת יכולת ההכללה של המודל שלנו על סט test ששמרנו מתחילת התרגיל.
תוצאות החיזוי עבור סט זה :
Test Error : 0.04400000000000004, Train Error : 0.009176470588235341
כלומר קיבלנו עבור סט האימון דיוק של 99.08% ועבור סט הבדיקה דיוק של 95.6%.

כעת עברנו לביצוע משימות החיזוי:

א. חיזוי פילוג ההצבעה לכל מפלגה לפי הסט החדש והמפלגה המנצחת:



ניתן לראות בגרף את פילוג המצביעים לכל מפלגה בסט החדש.



בנוסף הינה הנתונים שיצרו את הגרף :

מפלגה	מספר מצביעים	אחוז הצבעה
Blues	554	5.54%
Browns	**2247**	**22.47%**
Greens	659	6.59%
Greys	413	4.13%
Khakis	1328	13.28%
Oranges	518	5.18%
Pinks	296	2.96%
Purples	438	4.38%
Reds	520	5.2%
Turquoises	1852	18.52%
Violets	323	3.23%
Whites	424	4.24%
Yellows	428	4.28%

ניתן לראות כי Browns ניצח עם 22.47% מהקולות.

2. **מציאת קואליציות אפשריות על ידי מודל clustering:**

על מנת לבצע חלק זה, בחרנו בשני מודלים של clustering אשר בעזרתם ננסה לבנות קואליציה טובה.
המודלים שבחרנו הינם: GaussianMixture , KMeans.
שלב 1: מציאת מספר הקלאסטרים האופטימלי עבור כל מודל.
שלב זה ביצענו על ידי ביצוע CV על מספר הקלאסטרים שכל מודל מחפש, ובנינו פונקציית ניקוד אשר מנקדת כל קלאסטר שהוצע על ידי המודל ולפי כך בחרנו את הערך הטוב ביותר.
עבור אימון המודלים השתמשנו בסט האימון וסט הולידציה כאחד.
פונקציית הניקוד – בהינתן מודל מאומן, וסט ולידציה, תחילה ביצענו חיזוי עבור כל נקודה בדאטה, לאיזה מהקלאסטרים היא שייכת.
לאחר מכן עבור כל קלאסטר, עברנו על כל המפלגות שחזינו כי ישנם מצביעים הנמצאים באותו קלאסטר. עבור כל מפלגה בדקנו כמה אחוז מכלל המצביעים למפלגה זו נמצא בתוך הקלאסטר, אם אחוז זה גדול מ60% (כלומר רוב המצביעים למפלגה נמצאים בקלאסטר זה) הוספנו אחוז זה לציון הקלאסטר. (אם אחוז המצביעים למפלגה מסוימת היה קטן מ60% התעלמנו ממפלגה זו כי אינה שייכת לקלאסטר).
פונקציית ניקוד זו, בעצם מקשרת בין הקלאסטרים למפלגות הבעיה שלנו, כלומר ככל שהניקוד גדול יותר, כך יותר מפלגות שייכות ברובן לקלאסטר מסוים. לדוגמא אם בחנו אלגוריתם מסוים עם 3 קלאסטרים וקיבלנו כי מפלגה A נמצא בצורה שווה בכל קלאסטר (30% בכל קלאסטר), מודל זה לא יקבל ניקוד עבור המפלגה הנ"ל. אך אם מפלגה B שייכת בעיקר לקלאסטר מסוים (מעל 60%), המודל יקבל ציון לפי הגודל היחסי של המפלגה השייך לקלאסטר.

ביצענו בדיקה זו עבור מספר קלאסטרים הנע בין 2-7 ולבסוף קיבלנו כי שני המודלים מקבלים ניקוד אופטימלי עבור 2 קלאסטרים, לכן המשכנו עם פרמטרים אלה.

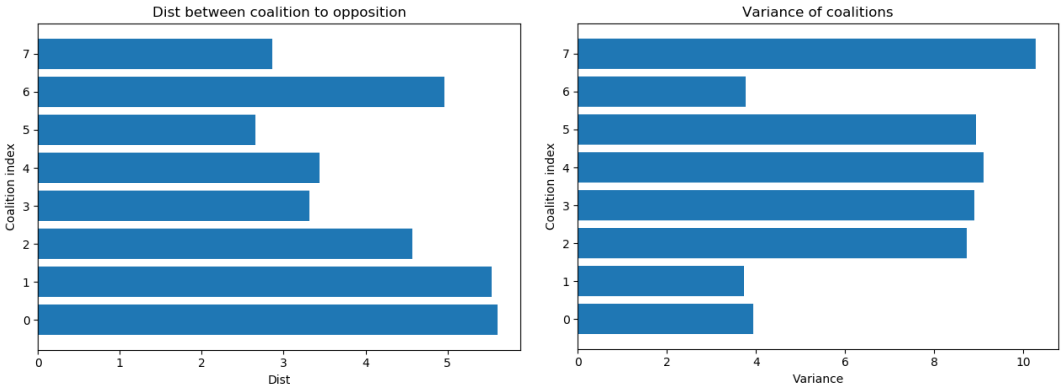
שלב 2: מציאת קואליציות אפשריות על ידי המודלים שמצאנו.
כעת לאחר שמצאנו את הפרמטרים המתאימים לכל מודל, השתמשנו במודל החיזוי מהסעיפים הקודמים (Voting classifier) על מנת לסווג את הסט החדש שקיבלנו.

כעת יש לנו את הסט החדש בעל התוויות שחזינו עבור כל מצביע.
בשלב זה עבור כל מודל שמצאנו ביצענו:
עבור כל קלאסטר שהמודל מצא –
השתמשנו בסט שבנינו עם החיזוי ומצאנו את כל המפלגות אשר אחוז שייכות המצביעים לכל מפלגה בקלאסטר גדול מ 85% (בחרנו ערך זה מפני שהוא גבוה מספיק כדי להגיד שרוב גדול של מצביעים למפלגה שייכים לקלאסטר זה)
לאחר מכן ספרנו את גודל סך המצביעים לכל מפלגות אלה, אם גודל זה גדול שווה ל51%, הוספנו את המפלגות הנ"ל כקואליציה לרשימת הקואליציות האפשריות.

בסוף שלב זה קיבלנו רשימה של קואליציות אפשריות, כך שכל קואליציה מכילה מספר מפלגות שסך המצביעים הכולל למפלגות אלה גדול מ51%.

שלב 3: מציאת קואליציה הומוגנית ביותר ושונה ביותר מהאופוזיציה.
בשלב זה עבור כל קואליציה שמצאנו בשלב הקודם, חישבנו את שונות הפיצ'רים של מצביעיה (המצביעים של הקואליציה – מצביעים לכל אחת מהמפלגות בקואליציה)
ובנוסף חישבנו את המרחק של ממוצע הפיצ'רים של הקואליציה לממוצע הפיצ'רים של האופוזיציה. (כלומר כמה שונה האופוזיציה מהקואליציה)
ביצענו מיון על כל אחת מהרשימות הנ"ל (שונות ומרחק), והסתכלנו על התוצאות, כך שהרשימה של השונות ממוינת מהקטן לגדול (ערך קטן יותר – קואליציה הומוגנית יותר) והמרחק ממוין מהגדול לקטן (ערך גדול יותר – קואליציה שונה מהאופוזיציה יותר).
לאחר הסתכלות על התוצאות בחרנו בקואליציה אשר מיקומה היה בראשונים בכל רשימה, כך שערך השונות שלה קטן והמרחק מהאופוזיציה גדול.

הגרפים שקיבלנו עבור השונות והמרחק של הקואליציות:



מתוך רשימת הקואליציות הבאות:

מפלגה	2, 3, 4, 5, 6, 8, 9, 11, 12	2, 3, 4, 5, 6, 8, 9, 11	0, 1, 2, 4, 6, 7, 9, 10, 11, 12	0, 1, 2, 4, 6, 7, 9, 10, 11	0, 1, 2, 4, 6, 7, 9, 10	0, 1, 2, 4, 6, 7, 9	2, 3, 4, 5, 6, 8, 9	0, 1, 2, 4, 6, 7
אינדקס	0	1	2	3	4	5	6	7

כאשר:

'Violets' - 10
'Browns' - 1
'Greens' - 2
'Whites' - 11
'Reds' - 8
'Turquoises' - 9
'Purples' - 7
'Greys' - 3
'Blues' - 0
'Khakis' - 4

'Oranges' - 5
'Pinks' - 6
'Yellows' - 12

הקואליציה שנבחרה היא קואליציה 0

['Greens', 'Greys', 'Khakis', 'Oranges', 'Pinks', 'Reds', 'Turquoises', 'Whites', 'Yellows']

אשר מכילה 9 מפלגות מתוך 13.
בעלת מרחק מקסימלי ושונות כמעת מינימלית.

הערות:
בחרנו בקואליציה זו מכמה סיבות:
- היא בעלת המרחק המקסימלי מהאופוזיציה, אשר גורם לכך שהיא כמה שיותר שונה מהאופוזיציה ולכן יותר יציבה (פחות סיכוי שמפלגה מסוימת מתוך הקואליציה תרצה להיות באופוזיציה)
- בעלת שונות כמעט מינימלית אך גדולה יותר מבחינת כמות מצביעים מאשר הקואלציות עם השונות הקטנה יותר, כלומר בכך שהגדלנו את מספר המפלגות בקואליציה בתמורה לגדילה קטנה בשונות, מאפשרת לקואליציה להיות יותר יציבה, מפני שבמקרה בו אחת המפלגות תחליט לפרוש מהקואליציה, עדיין יהיה מספיק מפלגות על מנת להחזיק בשלטון.

נשים לב כי הקואליציה אינה מכילה את המפלגה המנצחת של הבחירות, אך גודלה עדיין 64.38% בגלל שמכילה את המפלגה השנייה בגודלה, ולכן מספיקה להקמת הקואליציה.