דו"ח מסכם : שם הפרויקט – "נושמים לרווחה"

הסבר על ה dataset :

1. **שנה** – עמודה המתארת את השנה בה נאסף הנתון.
2. **מחוז** – עמודה המתארת את המחוז.
3. **רשות** – עמודה המתארת את מספר הרשות המקומית.
4. **שם רשות –** עמודה שמתארת את שם הרשות המקומית.
5. **מספר משפחות מקבלות סיוע –** עמודה שמתארת את מספר המשפחות שקיבלו סיוע.
6. **סך הכל שימושים** – עמודה שמתארת את מספר השימושים הכללי.
7. עבור כל סוג סיוע:
   - **הכמות של סלי המענים** – עמודה המתארת את מספר סל המענים בתחום מסוים.
   - **הוצאה כספית** – עמודה שמתארת את הסכום הכולל שהוקצה לסיוע בתחום מסוים.

(דוגמת סיוע בתחום "איבזור הבית והתקנה:"

- מספר סל המענים בסוג סיוע איבזור הבית והתקנה
- סך הכל הוצאה עבור איבזור הבית והתקנה בש"ח)

לסיכום: לכל תחום יש שתי עמודות, אחת שמתארת את כמות סלי המענים (הכמות) והשנייה שמתארת את הסכום הכספי שהוקצה לו (ההוצאה).

הסיבה שבחרנו בדאטה : היא ענתה על כל ההנחיות שקיבלנו והנושא נראה מעניין מעבר למספרים.

Section 1 – Introduction

הצגת הבעיה וחשיבותה : **supervised** הבעיה שמועלת בעבודה זו נוגעת לניתוח נתונים בתחום הסיוע הכלכלי והחברתי לאזרחים. הנתונים כוללים את הסלים השונים של סוגי הסיוע (כגון סיוע לאיבזור הבית, חינוך והשכלה, בריאות, תעסוקה, תחבורה ועוד) והשפעתם על המחוזות השונים במדינה. חשיבות העבודה נובעת מהצורך בהבנת התפלגות הסיוע והשפעתו על הצרכים השונים של האוכלוסייה בכל מחוז. יישום נכון של נתונים אלו יכול לסייע בקבלת החלטות מדיניות, ניהול משאבים והבנת האתגרים החברתיים-כלכליים שמציב כל מחוז.
unsupervised בחרנו לבחון את הקלאסטרינג לפי ניתוח סלים ולא לפי חלוקה גיאוגרפית משום שחלוקה גיאוגרפית לא בהכרח משקפת את דפוסי הצריכה האמיתיים של המשפחות. בעוד שחלוקה גיאוגרפית מתבסס על מקום המגורים בלבד, ניתוח הסלים מאפשר לנו להבין את ההוצאות של המשפחות בתחומים שונים (כמו חינוך, בריאות, דיור ועוד), מה שמספק תובנות עמוקות יותר על הצרכים וההתנהגויות שלהן. כך, קיבוץ המשפחות לקבוצות על פי דפוסים פנימיים בהוצאות יכול לעזור לזהות קבוצות הומוגניות יותר ולהתאים את הסיוע בצורה מדויקת יותר.

הבעיה נבחרה משום שמדובר בתחום שיכול להשפיע על איכות החיים של האזרחים ויכול להוביל לתובנות חשובות על חלוקה יעילה של סיוע ממשלתי.

מטרות : מטרת העבודה היא לבצע ניתוח על הנתונים שנמסרו, תוך זיהוי הקשר בין הסיוע שהתקבל בכל מחוז לבין הצרכים של האוכלוסייה, כפי שמיוצגים בעמודות של הסלים השונים. העבודה תתמקד בניתוח השפעת הסלים הספציפיים,

כמו חינוך, בריאות, תעסוקה ועוד, תוך סיווג המחוזות וניתוח ההתפלגות הכלכלית והחברתית של הסיוע. במקביל, ניתוח זה יספק הבנה לגבי המחוזות הזקוקים לסיוע מיוחד ובאיזה תחום יש להשקיע יותר משאבים, מה שיסייע בהחלטות ציבוריות ואסטרטגיות למדיניות סוציאלית.

Section 2 - Dataset and Features

פירוט תהליך הסריקה והורדת הנתונים

הנתונים שנעשה בהם שימוש נאספו מאתר אינטרנט. השיטה בה נעשה שימוש כדי לאסוף את הנתונים הייתה הורדה ישירה של הקובץ בפורמט CSV מהאתר, ולא באמצעות API או קריאה חיה של נתונים.

צעדי ההכנות, בחירת הפיצ'רים והרציונליות מאחורי הבחירות :

א. שנה אחת – בחרנו לעבוד עם שנה אחת (2019) כדי למנוע כפילויות בנתונים. החלטה זו עוזרת להבטיח שהמודל לא ימצא את עצמו מתמודד עם נתונים מרובים מאותה שנה, מה שמפשט את העבודה ומונע בעיות של חפיפות.

ב. ערכים חריגים – הערים "בענה" ו"חצור" מולאו על ידי ערים דומות להן מ2021 (כדיי להימנע מערכים חריגים בשנה של הקורונה) עם דימיון מבחינה כלכלית חברתית וגאוגרפית.

ג. בחירת הפיצ'רים – במהלך ניתוח הנתונים, גילינו שלכל תחום עזרה קיימות שתי עמודות – "מספר הסלים" ו-"סכום ההוצאה הכוללת". עם זאת, מצאנו כי סטיית התקן והממוצע בשתי העמודות היו שווים, דבר שהוביל אותנו למסקנה כי לא ניתן להפיק מידע מגוון נוסף מהן לצורך הניתוח. בעקבות זאת, החלטנו להתמקד בעמודה אחת בלבד, ובחרנו בעמודת "מספר הסלים", בשל נוחותה ויתרונותיה בעיבוד וניתוח הנתונים.

ד. הורדת נושא – פעילות רשותית. הנתונים בתחום זה היו חריגים במידה רבה, והם לא נראו כנתונים שהוזנו בצורה תקינה או הגיונית. לא היה ניתן להשלים את הערכים החסרים, שכן מספרם היה גבוה מאוד ולא הייתה אפשרות להשוותם לערכים אחרים, מה שהקשה על ביצוע חישובים או תיקונים בעזרת נוסחאות. לכן, החלטנו להסיר את הנתונים הללו מהניתוח.

Section 3- Methodology

תיאור האלגוריתם והטכניקות שיישמנו + תיאור מה נעשה ומדוע

למידת מכונה מפוקחת :

האלגוריתם שהשתמשנו בו הוא Random Forest, Logistic Regression, SVM, KNN, ו-Gradient Boosting, בגלל שהנתונים שלנו כוללים מספר קטגוריות לא מאוזנות, ואנו צריכות למצוא את המודל שיעבוד הכי טוב עם מאפיינים כמו משתנים רבים, חוסר איזון, ונתונים מורכבים.

הטכניקות שיישמנו הן SMOTE ו-GridSearchCV, בגלל ש SMOTE עוזר לאזן את הקטגוריות בנתוני האימון, ו -GridSearchCV מאפשר לנו למצוא את הפרמטרים האופטימליים לכל מודל על מנת להשיג את הביצועים הטובים ביותר. כל התהליך כולל גם Cross-Validation (CV) כדי להבטיח שהמודלים ייבחנו בצורה הוגנת ונכונה על פי נתונים שונים, ובכך למנוע overfitting ולשפר את הכללה של המודלים.

לאחר התוצאה, שמנו לב שעדיין יש מחוזות שלפעמים לא נקלטו על ידי חלק מן המודלים אז הוספנו עוד טכניקה להשוואה - תוצאות והשוואה על ידי מטריצת בלבול .

למידת מכונה לא מפוקחת :

האלגוריתמים שנבחרו הם KMeans ו:Agglomerative Clustering-

KMeans : מתאים לנתונים שלנו בגלל יעילותו בעיבוד דאטה גדול ויכולת יצירת חלוקה ברורה של המשפחות לקבוצות לפי הוצאות, מה שמסייע בהבנת דפוסי הצריכה של משפחות בקבוצות שונות.

Agglomerative Clustering : מתאים בשל גמישותו במבנה, ללא הצורך להגדיר את מספר הקבוצות מראש, והיכולת שלו להתמודד עם קלאסטרים בעלי צורות לא כדוריות, מה שיכול להוביל לגילוי מבנים פנימיים בנתונים שלא היו נראים בעבודה עם KMeans .

הטכניקות :

- השוואת אלגוריתמים :כדי לבחור את האלגוריתם המיטבי, השווינו את תוצאות KMeans ו Agglomerative-Clustering.
- :Silhouette Scoresהשתמשנו במדד זה כדי לקבוע את מספר הקבוצות האופטימלי.
- ויזואליזציה :בדקנו את חלוקות הקבוצות באמצעות גרפים כדי לוודא שהחלוקה ברורה ומתאימה לנתונים.

ההיגיון בבחירה היה לאפשר השוואה בין אלגוריתמים בעלי יתרונות שונים ולהתאים את החלוקה למאפייני הנתונים.

Section 4- Experiments and Results

<u>בחירת הפרמטרים</u>

למידת מכונה מפוקחת : הגענו לפרמטרים אלו באמצעות ,GridSearchCV שמאפשרת למצוא את הערכים האופטימליים לכל מודל על ידי חיפוש ממוקד בשדה הפרמטרים האפשריים.

```
Best parameters for Logistic Regression: {'model__C': 100, 'model__solver': 'saga'}
Best cross-validation score: 0.7447

Best parameters for Random Forest: {'model__max_depth': None, 'model__min_samples_split': 2, 'model__n_estimators': 50}
Best cross-validation score: 0.8225

Best parameters for SVM: {'model__C': 100, 'model__gamma': 'scale', 'model__kernel': 'rbf'}
Best cross-validation score: 0.8515

Best parameters for K-Nearest Neighbors: {'model__n_neighbors': 1, 'model__p': 1, 'model__weights': 'uniform'}
Best cross-validation score: 0.8102

Best parameters for Gradient Boosting: {'model__learning_rate': 0.1, 'model__max_depth': 3, 'model__n_estimators': 300}
Best cross-validation score: 0.8285
```

- מודל Logistic Regression : הערך הגבוה של C = 100 שיפר את הדיוק (0.7447) על ידי הפחתת כמות טעויות הסיווג, מה שגרם למודל להיות פחות סלחני כלפי טעויות.

- מודל Random Forest : השארת max_depth = None אפשרה למודל להעמיק את העצים ולהתאים לדפוסים מורכבים יותר בנתונים, דבר שתרם לדיוק גבוה.(0.8225)
- מודל SVM : שילוב של C = 100, kernel = 'rbf' ו gamma = 'scale'-שיפר את הביצועים (0.8515) על ידי הגברת יכולת המודל להתמודד עם מבנה לא לינארי של הנתונים ודפוסים מורכבים.
- מודל K-NN : "בהגדרת n_neighbors = 1, המודל הסתמך רק על השכנה הקרובה ביותר, מה שהגביר את הרגישות לשינויים קטנים בנתונים, וכתוצאה מכך הדיוק היה 0.8102".
- מודל Gradient Boosting : הגדרת learning_rate = 0.1 תרמה לדיוק של 0.8285 על ידי שמירה על איזון בין מהירות הלמידה ויציבות המודל, מנעה overfitting והביאה לתוצאה מדויקת יותר.

למידת מכונה לא מפוקחת :

מדובר בבחירת פרמטרים, אך אלה פרמטרים של מספר הקלאסטרים שנבדקים במודלים.

הקוד בודק את ערך ה silhouette score-עבור מספר קלאסטרים שונים ומבצע השוואה כדי למצוא את מספר הקלאסטרים עם הציון הגבוה ביותר עבור כל אלגוריתם. הבחירה מבוצעת בהתבסס על הציון הגבוה ביותר של.silhouette score

<u>מודלי הערכה ולמה הם נכונים לבעיה שלנו + הסבר המדדים המשעמעותיים</u>

למידת מכונה מפוקחת :

דיוק (Accuracy) **:** המדד הזה מתאים כיוון שהוא הכלי הכללי ביותר להערכת ביצועים בסיווג. עבור בעיות סיווג, כמו שלנו, יש חשיבות להעריך עד כמה המודל מצליח להקצות את הדוגמאות לקטגוריות הנכונות, והדיוק מספק תמונה כוללת על תפקוד המודל.

דיוק חיובי (Precision) : המדד חשוב במיוחד כאשר יש אי-איזון בין הקטגוריות, כלומר כאשר יש קטגוריות נדירות או קטגוריות עם הרבה יותר דוגמאות מאחרות Precision .עוזר לנו לוודא שהמודל לא חזה קטגוריות בצורה שגויה, בכך שהוא מתמקד בכמה מהתחזיות החיוביות היו נכונות.

זכירה (Recall) : קריטי כאשר יש צורך לוודא שהמודל לא מתעלם מקטגוריות נדירות, ושמצליח לזהות את כל המקרים. במקרה של קטגוריות פחות נפוצות Recall ,עוזר לוודא שהמודל לא "פוספס" את המקרים החשובים.

ציון F1 – Score **:** מדד זה מתאם במיוחד כאשר יש חשיבות לאיזון בין Precision ו Recall,-ובמיוחד כאשר מדובר בבעיות עם קטגוריות לא מאוזנות F1-Score .מספק מדד מאוחד שמזג את שני המדדים האלה, ומאפשר לנו להעריך את המודל בצורה מאוזנת יותר.

מטריצת בלבול (Confusion Matrix) **:** המטריצה מאוד מתאימה לבעיות של סיווג, שבהן המטרה היא להקצות קטגוריה בצורה נכונה. היא מציגה את היחסים בין התחזיות של המודל לבין הקטגוריות האמיתיות, ומאפשרת להבין בצורה ברורה כיצד המודל מבצע סיווג נכון ושגוי.

למידת מכונה לא מפוקחת :

Silhouette Score **:** מדד זה משמש להעריך את איכות הקיבוצים (clusters) שנוצרו על ידי אלגוריתמים של סיווג לא מפוקח .(Unsupervised Learning) הוא מודד את ההתאמה של הדגימות לקבוצות השונות, כלומר עד כמה כל נקודת נתון "מתאימה" לקבוצה שלה לעומת קבוצה אחרת.

מדדים ויזואליים: לצורך הערכת דיוק הקלאסטרינג השתמשנו בכמה מדדים ויזואליים. השתמשנו בגרף Silhouette Plot כדי להעריך את איכות ההפרדה בין הקלאסטרים, וב PCA-(ניתוח רכיבים עיקריים) כדי להוריד את ממדי הנתונים ולהציג את הקלאסטרים במרחב דו-ממדי. הדמיה זו עוזרת להבחין בהפרדה הברורה בין הקבוצות, ומאפשרת הערכה ויזואלית של איכות החלוקה, כמו גם זיהוי חפיפות בין הקלאסטרים.

<u>ממצאים, הצלחות, מגבלות ותוצאות בלתי צפויות</u>

למידת מכונה מפוקחת :

ממצאים :

א. המודל SVM : הוא המודל עם הביצועים הטובים ביותר על פי תוצאות ה ,Cross Validation-אך יש לו בעיות בהערכת הקטגוריות "ירושליים ותל אביב והמרכז"

ב. המודלים Random Forest ו - Gradient Boosting :הם מודלים מבטיחים עם ביצועים יחסית טובים.

ג. מודלים כמו K-Nearest Neighborsו SVM : המודלים הללו הציגו דיוק נמוך, במיוחד כשמדובר בקטגוריות בעלות תמיכה קטנה יותר, דבר שמצביע על קושי בהכללת המידע.

בשל הסיבות המפורטות בסעיפים א' ו-ג,'נבדקו המודלים גם בעזרת מטריצת בילבול על מנת לבחון את הממצאים בשיטה שונה.

ד. מודל Logistic Regression : המודל הצליח לזהות כראוי 10 תצפיות של קטגוריה 0 (10,0,0,1). בנוסף, זיהה תצפיות אחרות בצורה פחות מדויקת. הדירוג הסופי של המודל הוא 70, כלומר מדובר במודל עם ביצועים טובים יחסית.

ה. מודל Random Forest : המודל הצליח לאתר בצורה מדויקת 11 תצפיות של קטגוריה 0. היו טעויות בכמה קטגוריות אחרות, אך המודל לא היה רע באופן כללי.

ו. מודל SVM : ביצועים דומים ל - . המודל הצליח לאתר 10 תצפיות של קטגוריה 0 בצורה נכונה.

ז. מודל K-Nearest Neighbors : המודל מציג כמה טעויות יותר בהשוואה לאחרים, במיוחד בקטגוריות 1 ו – 3. היו כמה טעויות בקטגוריה 3

ח. מודל Gradient Boosting : יש זיהוי טוב עבור קטגוריה 0 (11 נכונות).היו טעויות בעיקר בקטגוריה 1.

סיכום: הממצא המרכזי הוא שמודל ה- Logistic Regressionהוא המודל עם הביצועים הטובים ביותר מבין כל המודלים, עם ציון של 70%.

מגבלות ותוצאות בלתי צפויות : החלוקה הלא מאוזנת של הקטגוריות והדאטה הקטנה יצרו קשיים, כפי שניתן לראות בממצאים. לצורך התמודדות עם הבעיות הללו, ניסינו להשתמש בטכניקות כמו SMOTE, stratify=y, ושימוש ב balanced-במודלים שאפשרו זאת (אך לבסוף החלטנו להפסיק את השימוש בהן כיוון שראינו שהתוצאות לא השתפרו). - הפיצ'רים (המאפיינים או המשתנים) שבנתונים אינם כוללים ייצוג מלא או מספיק של מחוזות תל אביב וירושלים. זה עשוי להעיד על כך שאין הבדל משמעותי בהתנהגות או בהתנהלות של מחוזות אלו ביחס למחוזות אחרים.

למידת מכונה לא מפוקחת :

ממצאים:

הבחנה בין קלאסטרים :באמצעות אלגוריתם ,KMeans הצלחנו להבחין בין קבוצות שונות של משפחות לפי דפוסי הצריכה שלהן. הקבוצות היו מבוססות על הוצאות בתחומים שונים כמו חינוך, בריאות, דיור ועוד. **תוצאות ויזואליות :**גרף ה-Silhouette Plot וPCA-אפשרו לנו להעריך את איכות הקלאסטרים וראו שהחלוקה הוגדרה היטב. המהות הויזואלית של הקלאסטרים הייתה ברורה, אך היו מקרים של חפיפות קלה בין קבוצות מסוימות . **תוצאות כמותיות :**המדדים כמו Silhouette Scoreהצביעו על כך שבחלק מהמקרים KMeans ,סיפק תוצאות טובות יותר מבחינת ההפרדה בין הקלאסטרים, אם כי היו מקרים שבהם המדד לא שיקף בדיוק את איכות ההפרדה שנראה בגרפים**.**

מגבלות:

הנחות על הנתונים: אחת המגבלות המרכזיות הייתה שהנתונים לא היו תמיד הומוגניים, מה שיכול להשפיע על תוצאות הקלאסטרינג. למשל, היו קבוצות ששיתפו דפוסים דומים אך לא תמיד הייתה הבדל ברור ביניהם.

תלות בבחירת מספר הקלאסטרים: קביעת מספר הקלאסטרים האידיאלי לא תמיד היתה חד משמעית, והייתה תלויה בבחירה של פרמטרים שונים כמו מספר הקלאסטרים וסוג האלגוריתם.

תוצאות בלתי צפויות:

הבדלים בין אלגוריתמים: הבחנתי שלמרות ש-KMeans נתן תוצאות טובות יותר מבחינה ויזואלית, אלגוריתם Agglomerative Clustering הציג לעיתים ציון Silhouette Score גבוה יותר, דבר שמפתיע כי בדרך כלל אנחנו מצפים מההפרדה הוויזואלית להיות מתואמת לציון זה.

קלאסטרים חופפים: למרות שהייתה ציפייה לקבוצות מופרדות היטב, היו מקרים בהם חפיפות בין קלאסטרים התרחשו, במיוחד כאשר מדובר בקבוצות בעלות דפוסים צריכה דומים במידה רבה.

<u>שוני בין האלגוריתמים</u>

למידת מכונה מפוקחת :

- המודל SVM : הראה תוצאה טובה יותר בחיפוש חוצה-ולידיישן, אך כיוון שמודל זה עלול להיתקל בבעיות של overfitting, ייתכן שהוא לא התאים בצורה טובה לנתוני הבדיקה.

- מודלי Gradient Boosting ו - Random Forest : היו יעילים בעבודה עם נתונים מאוזנים אך לא הצליחו להתמודד עם כל המחלקות בצורה שווה.
- מודל Logistic Regression : הציג ביצועים יציבים ביותר והיה המודל שמנע בעיות של.overfitting
- מודל Random Forest : מצוין במחלקות מסוימות, אך יש טעויות רבות במחלקה **south**, דבר שמוריד את ביצועיו הכוללים.

לסיכום SVM, היה המודל בעל הציון הגבוה ביותר במבחן חוצה-ולידיישן, אך לא עמד בציפיות כאשר הוכנס לנתוני הבדיקה. על מנת לשפר את הביצועים, לכן ניסינו להתמקד במודלים כמו Random Forestאו Logistic Regression, שהם בעלי ביצועים יציבים יותר.

במטריצת הבילבול היה ניתן לראות שהמודל של רגרסיה לוגיסטית קיבל את הציון הגבוה ביותר.

למידת מכונה לא מפוקחת : כמו שנאמר קודם לעיתים ציון ה Silhouette Score היה גבוהה יותר באלגוריתם אחד בעוד תוצאות הגרפים הויזואליים הראו שהאלגוריתם השני נותן חלוקה טובה יותר, בסוף בגלל שההבדל לבציון המספרי היה לרוב מינורי בחרתי להשתמש בחלוקה הויזואלית המודייקת שנראתה שהביאה חלוקה טובה ומופרדת יותר.

Section 5 - Conclusion and Discussion

<u>תרומות והשלכות של כל חבר בפרוייקט:</u>

	יובל	**ענבל**
1	בחירת הדאטה וסידור הדאטה	בחירת הדאטה סידור הדאטה
2	למידת מכונה מפקוחת (בעיקר)	למידת מכונה לא מפוקחת (בעיקר)
3	תדריך	תדריך
4	מצגת	מצגת
5	עידוד וחיזוק ענבל	עידוד וחיזוק יובל

היינו ממשכיות לחקור אם יש עוד דרכים להתעלות מעל חוסר איזון בעמודת ה-Y, התעסקות עם דאטה קצרת שורות וקורולציות נמוכות בין הפיצ'רים לעמודת החיזוי. כמו כן בדיקה של כל הפיצ'רים ותרומתם לחיזוי. שיטה שונה בכל מודל לדיוק התוצאות. לבחון את התוצאות יותר לעומק ואת השפעתו של הנירמול על החיזוי, שימוש בטכניקות לאיזון :כמו oversampling או undersamplingכדי לאזן את הקטגוריות. וכו..

כמו כן, Anomaly Detection - זיהוי אנומליות היה יכול להיות שימוש מעניין כיוון שהוא כלי שימושי במקרה של דאטה קצרה ולא מאוזנת ב Y-(המשתנה התלוי) כמו אצלנו. אנומליה מתייחסת למקרים שבהם הדפוסים בנתונים שונים באופן מובהק מהשאר, ולפיכך ייתכן שנוכל לנצל את זה כדי לזהות דפוסים או תצפיות חריגות שעשויות להיות בעלות ערך.

יובל בר-און – ת.ז. 318227261 : https://github.com/YuvalBaron1997/Course-Project---Advanced-Topics-in-Machine-Learning-

ענבל אקרמן – ת.ז. 322522996 https://github.com/InbalAkerman/Advanced-Topics-in-Machine-Learning