תמונה שמכילה טקסט, גופן, צילום מסך, ירוק

תוכן שנוצר על-ידי בינה מלאכותית עשוי להיות שגוי.

****

**Data Preparation Report**

**רום קדוש – 207691841**

**סנדרה זייגרמכר -206987521**

**תוכן עניינים**

1. **בחירת הנתונים ............................................................................. 2-3**
2. **ניקוי הנתונים ................................................................................ 4-7**
3. **יצירת הנתונים ................................................................................ 7-8**
4. **שילוב הנתונים ................................................................................. 8**
5. **עיצוב הנתונים.................................................................................9-11**
6. **ניתוח נתונים ראשוני.........................................................................11-18**
7. **נספחים..........................................................................................19-21**
8. **בחירת הנתונים**

הפרויקט עוסק בניתוח תאונות דרכים במטרה לזהות דפוסים מרכזיים ולהסיק תובנות שיוכלו לסייע בשיפור הבטיחות בדרכים. במסגרת הפרויקט אנו מנתחים נתונים של תאונות דרכים בארץ בין השנים 2022 -2023. באמצעות שימוש בכלים ושיטות מתקדמות מתחום מדעי הנתונים נוכל להבין כיצד גורמים שונים, כגון תנאי הדרך, מזג האוויר, סוג התאונה והמהירות המותרת ועוד, מהווים השפעה על חומרת התאונה.

בחירת שיטת הסינון – התמקדות בעמודות (תכונות הנתונים):

בפרויקט זה בחרנו בגישה של סינון וניתוח עמודות מסוימות (Attributes Selection) במקום סינון שורות. הסיבה לכך שבחרנו את השיטה הזו היא כי באמצעותה נוכל לזהות את המשתנים המשפיעים ביותר על תאונות הדרכים, ולכן ישנה חשיבות להתמקד במידע הרלוונטי ביותר מבחינת תוכן (עמודות) ולא בכמות הרשומות (שורות).

במקום לעבוד עם כל המשתנים הזמינים, בחרנו בקפידה את העמודות החשובות ביותר (Features Selection) בשביל להימנע ממידע לא רלוונטי שעלול לפגוע ברמת דיוק הניתוח. המשתנים שנבחרו מספקים תובנות על סוג הדרך, חומרת התאונה, מהירות מותרת, סוג התאונה, תנאי מזג האוויר, תאורה, רמזורים תקינות הדרך ועוד מספר מאפיינים קריטיים אשר כולם משחקים תפקיד חשוב בהבנת הגורמים המרכזיים לתאונות דרכים.

**המאפיינים שבחרנו מתוך הdataset:**

סוג דרך, סמל ישוב, רחוב 1,רחוב 2, שנת תאונה, חודש תאונה, יום/לילה, יום בשבוע, מרומזר/לא מרומזר, חומרת התאונה, סוג תאונה, מהירות מותרת, תקינות הדרך, סימון/תמרור, תאורה, בקרה בצומת (קיים רק בסוג דרך 1,3), מזג אויר, מצב פני הכביש.

**רלוונטיות של המשתנים שנבחרו למטרות כריית הנתונים**

בפרויקט זה, מטרת כריית הנתונים היא לנתח תאונות דרכים עם נפגעים תוך זיהוי מגמות, גורמים תורמים וגורמים מסייעים למניעת תאונות דרכים עתידיות. המשתנים שנבחרו מספקים מידע קריטי על מאפייני התאונה, תנאי הדרך, סוג התאונה, חומרתה, מיקומה, תנאי מזג האוויר ועוד. המשתנים הללו נבחרו בקפידה כדי לתמוך בניתוח מעמיק של הקשרים בין הגורמים השונים ולזהות דפוסים חוזרים שיכולים לשמש לגיבוש המלצות בטיחות.

להלן מספר דוגמאות מתוך סך המאפיינים שבחרנו:

* "חומרת התאונה" היא משתנה קריטי המאפשר הבנה של סוגי התאונות והשלכותיהן.
* "סוג תאונה" מאפשר סיווג של התאונה לפי אופי ההתנגשות, כגון תאונת חזיתית, התנגשות ברכב חונה, דריסה ועוד.
* "מהירות מותרת" מהווה אינדיקציה לגבי תנאי הדרך והשפעת המהירות על חומרת התאונה.
* "תנאי הדרך" ו"תנאי מזג האוויר" יכולים לספק קשרים אפשריים בין מצב הכביש/מזג האוויר לבין שכיחות התאונות.
* "סוג הדרך" (עירוני, בין-עירוני) עוזר להבין כיצד תשתית הדרך משפיעה על בטיחות הנהגים ועוברי האורח בדרכים.
* "מרומזר/לא מרומזר" מאפשר לבדוק האם צמתים עם רמזורים מפחיתים תאונות.

המשתנים הללו מאפשרים חקר רב-ממדי שיסייע לנו להבין אילו גורמים הכי משפיעים על מקרי תאונות בדרגות חומרה שונות, ואילו תנאים עלולים לצמצם תאונות דרכים.

**איכות הנתונים והשפעתה על תקפות התוצאות**

איכות הנתונים הינה קריטית לדיוק המסקנות. ישנם מספר קשיים שעלולים להופיע בפרויקט:

* נתונים חסרים – במקרים מסוימים, לא כל השדות ימולאו במלואם. לדוגמה, אם "מהירות מותרת" או "מצב פני הכביש" חסרים, זה עלול לפגוע בדיוק הניתוח.
* שגיאות הקלדה/קידוד – אם יש בעיות בקידוד המשתנים (למשל קידודים שונים לאותו סוג תאונה), ייתכן שיידרש ניקוי וסטנדרטיזציה של הנתונים.
* ערכים קיצוניים – אם יש ערכים קיצוניים ולא הגיוניים (למשל מהירות מותרת של 500 קמ"ש), יש צורך להסירם או לטפל בהם באופן מושכל.
* לפני ביצוע הניתוח, נדרש לבצע בקרת איכות נתונים הכוללת השלמת ערכים חסרים, בדיקת חריגים, ואיחוד קידודים.

**שחזור ושיפור הנתונים החסרים**

במקרים בהם ניתקל בנתונים חסרים נוכל לטפל בהם במספר דרכים:

* השלמה על סמך נתונים דומים – לדוגמה, אם חסרה מהירות מותרת ברחוב מסוים, ניתן לבדוק ערכים דומים מאותו אזור, אפשר לבצע חישוב מתמטי של הממוצע באותו אזור וכו׳.
* הסרת נתונים חלקיים – אם המידע חסר בשיעור גבוה מאוד, ייתכן שיהיה עדיף להסיר רשומות לא שלמות על מנת למנוע הטיה של המודל ופגיעה בטיב התוצאות.
* המרת ערכים חסרים לערכים ניטרליים – לדוגמה, אם תנאי הדרך חסרים, ניתן לשים אותם תחת קטגוריית "לא ידוע" במקום להסיר את הרשומה.

**מגבלות אתיות או חוקיות בשימוש בנתונים**

הנתונים אינם מכילים פרטים מזהים כמו שמות נהגים או מספרי רכב, ולכן אין מגבלות פרטיות. עם זאת, תמיד חשוב לשים לב למשתנים שעלולים להוות בעיה בהקשרים רגולטוריים, כמו מגבלות על הצגת נתונים ספציפיים לאזורים מסוימים או סיווג אזורים על סמך מדדים רגישים.

**שוני בין בחירת הנתונים להיפותזות שהוגדרו בשלבי ההבנה הראשוניים**

בשלב הראשוני הונחו מספר השערות, כמו ההשפעה של מהירות מותרת על חומרת התאונה או הקשר בין תנאי מזג האוויר לבין סוג התאונה. כעת, אחרי בחינת הנתונים, ייתכן שחלק מההנחות יצטרכו להיות מותאמות על בסיס המידע הזמין. לדוגמה, אם יש חוסר נתונים משמעותי באחד מהמשתנים, ייתכן שלא ניתן יהיה להסיק ממנו מסקנות חד משמעיות, במידה ואכן ניתקל בבעיות מהסוג הזה נעשה התאמות על מנת לשמור על הנתונים בשלמותם ונוכל להסיק עליהם תובנות ברמת דיוק גבוהה.

1. **ניקוי נתונים**

בפרויקט זה אנו עוסקים בניתוח תאונות דרכים תוך התמקדות בגורמים מרכזיים שעשויים להשפיע על רמת הסיכון בכבישים שונים בישראל. על מנת להבטיח שהנתונים שלנו יהיו אמינים, מדויקים ורלוונטיים למטרות הניתוח, ביצענו מספר שלבים הכרחיים בעיבוד הנתונים. כל שלב בתהליך הותאם כך שיספק תשתית איכותית ואחידה לניתוח סטטיסטי ובניית מודלים מתקדמים.

**שלבי העבודה**

**1. הורדת קבצי הנתונים לשנים 2022-2023**

הנתונים המקוריים נלקחו ממאגרי המידע הפתוחים של מדינת ישראל (datagov) , והכילו פרטים על תאונות דרכים עם נפגעים לאורך השנים 2022 ו-2023. קבצים אלו מכילים מספר רב של עמודות, חלקן עם מידע קריטי לניתוח שלנו, וחלקן אינן רלוונטיות או מכילות ערכים חסרים רבים.

**2. איחוד הנתונים לקובץ מרכזי אחד**

על מנת שנוכל לבצע השוואות בין השנים ולאפשר ניתוח מגמות והתנהגות לאורך זמן, היה הכרחי לאחד את שני קובצי השנים 2022 ו-2023 לקובץ נתונים אחד. האיחוד בוצע תוך שמירה על עקביות במבנה הנתונים, כך שכל עמודה בקובץ המאוחד מייצגת את אותו המידע בשתי השנים עם מבנה טבלאי זהה על מנת שנוכל לבצע עיבודים שונים על הנתונים בהמשך, בנוסף, ביצענו המרה של הקבצים מcsv לxsls בשביל שבהמשך הניתוחים הפורמט ייתמך בצורה טובה יותר עם תוכנות ומערכות שונות.

**3. בחירת העמודות הרלוונטיות לפרויקט**

מכיוון שהקבצים המקוריים כללו עשרות משתנים, רבים מהם לא היו רלוונטיים למטרות הניתוח שלנו. לכן, בחרנו בקפידה את העמודות המרכזיות שישמשו אותנו לאורך הפרויקט, תוך התמקדות בפרמטרים המשפיעים ישירות על תאונות הדרכים, כגון:

* מאפייני הדרך (סוג דרך, מהירות מותרת, תקינות הדרך, תאורה, סימון ותמרור)
* מאפייני התאונה (חומרת התאונה, סוג התאונה)
* תנאי הסביבה (מזג אוויר, מצב פני הכביש)
* מיקום גיאוגרפי (קורדינטות X ו-Y, סמל יישוב, רחוב 1 ורחוב 2)
* זמן האירוע (שנת תאונה, חודש, יום בשבוע, יום/לילה)
* תהליך הבחירה בוצע מתוך הבנה כי לכל אחד מהמשתנים הללו עשויה להיות השפעה מכרעת על רמת הסיכון בכבישים, ולכן יש לכלול אותם בניתוח.

**4. טיפול בבעיית הרחובות החסרים והמרת קורדינטות**

במהלך בחינת הנתונים, נתקלנו בבעיה משמעותית בעמודות "רחוב 1" ו-"רחוב 2". עמודות אלו הכילו מידע מקודד על שמות הרחובות והכילו ערכים חסרים רבים, כך שלחלק ניכר מהתאונות לא היה שיוך לרחוב מסוים. למאגר הנתונים לא נמצא מילון מיפוי רשמי שמאפשר לפענח את הקודים, מה שהפך את העמודות לפחות שימושיות. המידע על הרחובות היה חיוני ביותר לפרויקט, שכן ניתוח הסיכונים מבוסס בין היתר על זיהוי כבישים ואזורים מועדים לפורענות.

לכן, החלטנו להשתמש בדרכים חלופיות על מנת לקבל את המידע המדויק באמצעות הקורדינטות X ו-Y שקיבלנו כתכונות רלוונטיות במאגר נתונים שלנו ביצענו המרה שלהן על מנת לקבל כתובות גיאוגרפיות מדויקות. נתקלנו באתגר הראשון והמרכזי - הקורדינטות בקובץ הנתונים היו מקודדות לפי מערכת רשת ישראל החדשה (ITM), שאינה גלובלית. לכן, נדרשה המרה למערכת גלובלית סטנדרטית (WGS84 - רוחב ואורך גיאוגרפי).

ביצענו את ההמרה באמצעות ספריית Geopandas ו-Geopy וPyproj בפייתון, שאפשרו לנו למפות את הנתונים לכתובות מדויקות. פעולה זו אפשרה לנו לשחזר את הרחובות החסרים, ולבצע ניתוח מרחבי מדויק יותר. קוד המרת הקורדינטות מ ITMל WGS84נמצא בסוף הדו״ח תחת ״נספחים״.

**5. סינון והסרת נתונים לא רלוונטיים מתוך העמודה החדשה שייצרנו (street) באמצעות קוד:** תוך עבודה על הנתונים החדשים שיצרנו, קיבלנו גם ערכים חדשים מספריים שמסמלים את מספרי הכביש, שמות של רחובות בעברית, שמות של רחובות בערבית ושפות זרות.

לאחר שהמרנו את הקורדינטות לרחובות, ביצענו ניקוי נוסף של המידע:

* הסרת רחובות ומספרי כבישים שנכתבו בערבית – מאחר והנתונים שהתקבלו היו בשפות שונות (עברית, ערבית), הוחלט להסיר את הרחובות בערבית בשביל למנוע כפילות ובעיות ניתוח עתידיות שעלולות ליצור בלבול ואי דיוק.

במהלך עיבוד הנתונים, נתקלנו בערכים רבים של "Unknown" כאשר ניסינו להמיר קואורדינטות לשמות רחובות. כדי לשפר את איכות הנתונים ולצמצם חוסרים, החלטנו ליישם קוד המאתר את המחוז של כל קואורדינטה, כך שגם במקרים בהם לא ניתן לשלוף שם עיר מדויק, עדיין נוכל לסווג את הנתון לרמת מחוז. זה יסייע לנו לשמור על מידע גיאוגרפי רלוונטי, לזהות אזורים בסיכון גבוה ולבצע ניתוחים סטטיסטיים מבלי לאבד רשומות.

כתוצאה מכך הצלחנו לשמור על 1,628 רשומות ולהסיר רק 58.

מחיקת הרשומות הלא רלוונטיות (שפות זרות וערכים חסרים) מסך הרשומות במאגר המאוחד לא היווה שינוי משמעותי בכמויות הנתונים, סה״כ כמות הרשומות שלנו הייתה 19,237 - לאחר ההסרה נשארנו עם 19,044 רשומות.

**6**. **מחיקת עמודות הרחובות ושימוש בקורדינטות הגיאוגרפיות**

לאחר שביצענו את המרת הקורדינטות מרשת ישראלית מקומית (ITM) למערכת גיאוגרפית גלובלית (WGS84), החלטנו להסיר את עמודות "רחוב 1" ו-"רחוב 2" והקורדינטות הישנות (x,y) והחדשות שיצרנו (longitude ו latitude) .

במהלך בחינת הנתונים, זיהינו כי עמודות אלו סובלות משני כשלים עיקריים:

* חוסר משמעותי בנתונים – מספר רב של שורות הכילו ערכים חסרים, מה שהפחית את המהימנות של מידע זה לצורך הניתוח, לאחר שפתרנו את הסוגייה הזו גילינו כי אין לנו צורך בעמודות הישנות.
* קידוד בלתי נגיש – לא נמצא מילון מיפוי מקיף שיכול לאפשר פענוח של שמות הרחובות מתוך הקודים המופיעים בקובץ. מצב זה הפך את הנתונים בעמודות אלו לכמעט בלתי ניתנים לשימוש אנליטי.

בהתאם לכך, החלטנו להסיר לחלוטין את עמודות "רחוב 1" ו-"רחוב 2", ולבסס את הניתוחים שלנו על עמודת רחובות חדשה, שנוצרה באמצעות המרת הקורדינטות הגיאוגרפיות.

**מחקר והשוואה של קורדינטות מקוריות וחדשות**

לאחר ההמרה, ביצענו מחקר לבדיקת התאמה בין הקורדינטות המקוריות (X, Y) לבין הקורדינטות החדשות שהופקו. נבדקו מקרים שבהם ההמרה סיפקה כתובות מדויקות, והצלחנו למפות רחובות בהתאם למיקומם הגיאוגרפי בפועל. זוהו אזורים שבהם ההמרה לא הייתה חד-משמעית, ולכן בוצע סינון נוסף של נתונים עם חוסר ודאות גבוהה.

לבסוף, לאחר תהליך זה, הקובץ הסופי מכיל עמודת רחובות חדשה ומדויקת, המבוססת על נתוני הקורדינטות הגיאוגרפיות המתוקנות. עמודה זו תחליף את העמודות הישנות ותשמש כבסיס לכל ניתוח מרחבי שנבצע בהמשך הפרויקט.

**7. יצירת קובץ הנתונים הסופי**

לאחר כל תהליכי הניקוי והעיבוד, התקבל קובץ נתונים אחיד, הכולל את כל המידע הנדרש לניתוח, עם קורדינטות מתוקנות ונתונים אמינים יותר. קובץ זה ישמש אותנו לכל אורך חיי הפרויקט לצורך ביצוע אנליזות, זיהוי מגמות וסיכונים, ויישום אלגוריתמים מתקדמים.

**מסקנות והשלכות להמשך הפרויקט**

לאחר שביצענו תהליך מעמיק של עיבוד, ניקוי ושיפור הנתונים, אנו יכולים להסיק מספר מסקנות עיקריות שישפיעו על המשך העבודה בפרויקט:

* שיפור איכות הנתונים והפחתת רעשים – באמצעות שלבי העיבוד השונים, הבטחנו שהנתונים המשמשים לניתוח יהיו מדויקים, נקיים ומסודרים, תוך הסרת נתונים לא רלוונטיים או כאלו שעלולים להטות את המסקנות. הסרת הרחובות הכתובים בערבית, כמו גם נתוני Unknown, צמצמה את הרעש בנתונים והובילה למידע איכותי ומהימן יותר.
* חיזוק היכולת לבצע ניתוחים מבוססי מיקום והסקת מסקנות מבוססות נתונים – המעבר לשימוש בקורדינטות גיאוגרפיות (WGS84 במקום ITM) אפשר לנו לייצר תובנות מבוססות-מיקום בצורה מדויקת יותר. כעת, ניתן למפות באופן אמין את המיקום של תאונות הדרכים ולנתח אזורים בעלי רמת סיכון גבוהה, תוך שימוש ברחובות המדויקים שנוצרו לאחר המרת הקורדינטות. העובדה שהנתונים כעת כוללים מידע מדויק על מיקום התאונות מאפשרת לחזות מוקדי סיכון עתידיים ולתת תובנות חשובות לגורמים האחראים על בטיחות בדרכים. מודלים מבוססי למידת מכונה, שיורצו בשלב מתקדם יותר, יתבססו על נתונים מדויקים יותר, מה שיאפשר ניבוי אמין של אזורים בעלי פוטנציאל גבוה לתאונות.
* יצירת בסיס נתונים אחיד וסטנדרטי – תהליך עיבוד הנתונים כלל הסרת חוסר אחידות בקידוד, התאמת נתונים לפורמט אחיד והמרת ערכים לפורמטים סטנדרטיים. תהליך זה חיזק את היכולת שלנו להשוות בין השנים השונות ולבצע אנליזות עומק על מגמות בתאונות הדרכים בישראל.
* תמיכה בניתוחים גיאוגרפיים מתקדמים – העובדה שאנו כעת עובדים עם קורדינטות גיאוגרפיות תקניות ומדויקות תאפשר לנו לבצע ניתוחים מבוססי GIS, כמו יצירת מפות חום, זיהוי מקטעי כביש מסוכנים, וביצוע חיתוכים מרחביים בין גורמים שונים (כגון מזג אוויר, סוגי כבישים, רמת תאורה, ועוד).

1. **יצירת נתונים חדשים**

בפרויקט שלנו, נדרשנו לבנות נתונים חדשים על מנת לשפר את איכות המידע ולבצע ניתוחים גיאוגרפיים מתקדמים. שלב זה כלל גזירת מאפיינים חדשים מתוך המידע הקיים.

1. המרת קורדינטות למערכת גלובלית (ITM -> WGS84)

השתמשנו בספריית Pyproj להמרת נקודות הקורדינטות ממערכת ITM למערכת WGS84 (EPSG:4326), שהיא הסטנדרט הבינלאומי למפות ומערכות GIS.

תהליך זה בוצע בקוד הראשון על ידי הפונקציה Transformer.from\_crs("EPSG:2039", "EPSG:4326"), אשר ביצעה את ההמרה של X ו-Y לרוחב ואורך גיאוגרפי (Latitude, Longitude). לבסוף, צירפנו את הערכים החדשים לקובץ הנתונים ושמרנו אותו עם הקורדינטות הגלובליות החדשות.

2. הפקת נתוני רחובות באמצעות הקורדינטות

לאחר שהמרנו את הקורדינטות לפורמט גלובלי, ביצענו שליפת שמות רחובות בהתבסס על הנתונים הגיאוגרפיים. לצורך כך, השתמשנו בספריית geopy, הכוללת את היכולת לבצע Geocoding הפוך – כלומר, קבלת שם רחוב מתוך קורדינטות. תהליך זה בוצע ע"י הספרייה Nominatim, שהיא מנוע GIS חינמי המשתמש ב-OpenStreetMap (OSM) כדי לתרגם קורדינטות לשמות רחובות.

הקוד עבר תהליך אופטימיזציה משמעותי על ידי:

* הפחתת קריאות מיותרות – שמרנו בזיכרון את הכתובות שכבר הומרו, כך שנקודה גיאוגרפית שחוזרת על עצמה לא תישלח לשירות פעם נוספת.
* Rate Limiting – מאחר ול-Nominatim יש הגבלת גישה, השתמשנו בפונקציה RateLimiter, שאפשרה לבצע שליפות בצורה מבוקרת עם שהייה של שנייה בין כל קריאה כדי למנוע חסימה.
* שמירה זמנית – כל 500 רשומות נשמרו לקובץ זמני כדי למנוע איבוד מידע במקרה של הפסקת הריצה.
* עדכון קובץ הנתונים הסופי – הכתובות שהופקו נוספו למערך הנתונים תחת עמודה חדשה בשם "street", והקובץ נשמר מחדש.

3. הפקת נתוני מחוזות באמצעות קואורדינטות

בנוסף לשליפת שמות רחובות, זיהינו צורך להשלים מידע גיאוגרפי גם עבור רשומות שבהן לא הצלחנו לאתר כתובת מדויקת – מצבים שבהם התקבל הערך "Unknown". על מנת לשמר את ערך המידע ולבצע ניתוחים ברמה גאוגרפית רחבה יותר, יצרנו עמודה נוספת בשם "district", אשר מציינת את שם המחוז שבו נמצאות הקואורדינטות של כל רשומה.

לצורך כך השתמשנו באותה הגישה של Reverse Geocoding דרך ספריית geopy ומנוע Nominatim, אשר מבוסס על שירותי OpenStreetMap (OSM). במקום לחפש שם רחוב, חילצנו את שם המחוז (שדה "state" במבנה הכתובת) מתוך הקואורדינטות.

התהליך בוצע תוך הקפדה על יעילות ומהימנות, וכלל את אותם שיפורים שביצענו בשליפת הרחובות.

* הפחתת קריאות כפולות על ידי שמירה בזיכרון של קואורדינטות שכבר עובדו.
* הגבלת קצב הפניות (Rate Limiting) למניעת חסימה על ידי Nominatim.
* שמירה הדרגתית של תוצאות לקובץ זמני, למניעת אובדן מידע במקרה של עצירה לא מתוכננת.
* עדכון הקובץ הסופי בעמודה חדשה "district" עם שם המחוז המתאים.

באמצעות צירוף זה, הצלחנו להעשיר את המידע הקיים גם כשחלק מהכתובות אינן זמינות, ובכך לאפשר ניתוחים לפי אזורים גיאוגרפיים, זיהוי מוקדי סיכון ברמה מחוזית, והסקת מסקנות כלל ארציות, כך בעצם מנענו אובדן מידע ושימרנו את כמות הרשומות.

1. **שילוב נתונים**

בפרויקט שלנו היה צורך קריטי בשילוב נתונים ממקורות שונים, תוך שמירה על אחידות, דיוק ומהימנות הנתונים. שלב זה היה הכרחי לצורך יצירת מאגר מידע שלם שיאפשר ניתוחים אנליטיים איכותיים, חיזוי מגמות, וזיהוי אזורי סיכון.

**שיטת העבודה שנבחרה:**

תהליך שילוב הנתונים בוצע באמצעות חיבור (Appending) בהתאם למטרות השונות של הפרויקט.

**חיבור נתונים (Appending):**

ביצענו שילוב של שני מאגרי נתונים מרכזיים – נתוני תאונות הדרכים לשנים 2022 ו-2023. כל קובץ הכיל רשומות ועמודות בעלות מבנה זהה, אך עבור שנים שונות. חיבור הנתונים נעשה על פי עקרון הוספת רשומות (Appending), כלומר, צירוף הנתונים יחדיו לכדי קובץ אחיד, תוך הוספת עמודת זיהוי (SHNET\_TEUNA) לציון שנת האירוע. תהליך זה איפשר ניתוח רוחבי והשוואתי בין השנים, במטרה לזהות מגמות ושינויים לאורך זמן.

**אתגרים והתמודדות**

אחידות העמודות - חיבור הנתונים דרש התאמה מלאה של שמות העמודות והמבנה בכל קובץ.

חוסר בערכים - חלק מהעמודות הכילו ערכים חסרים, לכן טיפלנו בהם, כתוצאה מכך אין ערכים חסרים במאגר הנתונים שלנו.

1. **עיצוב נתונים**

בשלב זה, טרם הפעלת המודלים האנליטיים, אנו בודקות את מבנה הנתונים, מוודאות את התאמתם לדרישות האלגוריתמיות, ומבצעות שינויים נדרשים לשיפור איכות הנתונים. מטרת שלב זה היא להפוך את הנתונים לפורמט שיאפשר ניתוח מדויק, חיזוי אמין והסקת מסקנות מבוססות.

רשימת משימות ופעולות שביצענו:

* ניקוי והכנת הנתונים – הסרת נתונים לא עקביים, טיפול בערכים חסרים, הסרת ערכים לא רלוונטיים.
* המרת קורדינטות – הפיכת קורדינטות מקומיות (X, Y) למערכת גיאוגרפית גלובלית (Latitude, Longitude).
* שילוב נתונים חיצוניים – שימוש במידע משלים ממקורות נוספים.

**סקיצה ראשונית למודלים**

המודלים שנבחרו לפרויקט מבוססים על שיטות חיזוי מתקדמות, ניתוח אשכולות, הבנה של סדרות זמן והסבר תוצאות חיזוי. כל אחד מהמודלים ממלא תפקיד שונה ומשלים את האחרים, כך שהשילוב ביניהם מאפשר תמונה רחבה, מדויקת ומעמיקה של הגורמים המשפיעים על תאונות דרכים. רצינו להשתמש במודלים חדשניים ומורכבים על מנת למקסם את תוצאות החיזוי ולדייק את תוצאות החיזוי.

**1. XGBoost – חיזוי חומרת התאונה**

XGBoost (Extreme Gradient Boosting) הוא אחד המודלים המתקדמים ביותר ללמידה מפוקחת בתחום הסיווג והחיזוי. הוא מבוסס על עצים מחוזקים (Boosted Trees) ומנצל אלגוריתמים של גרדיאנט בוסטינג כדי לשפר את הדיוק ולהקטין את שגיאות החיזוי.

יתרונות מרכזיים:

* ביצועים גבוהים מאוד במקרים של נתונים מסובכים ורבים.
* יכולת להתמודד עם ערכים חסרים ולנחש נתונים שאינם קיימים.
* הבנה של משתנים חשובים והשפעתם על התוצאה.

תרומת המודל לפרויקט:

* זיהוי הגורמים המשפיעים ביותר על חומרת התאונה (כגון מהירות מותרת, סוג הדרך, מזג אוויר).
* חיזוי האם תאונה תהיה קטלנית, קשה או קלה, בהתבסס על מאפייני התאונה.
* שימוש ב-SHAP להסבר תחזיות – מאפשר להבין מדוע התאונה דורגה ברמת חומרה מסוימת.

**2. DBSCAN – זיהוי אזורים מסוכנים לתאונות**

DBSCAN (Density-Based Spatial Clustering of Applications with Noise) הוא אלגוריתם למידה לא מפוקחת המשמש לזיהוי אשכולות (Clusters) של תאונות במרחב הגיאוגרפי. בניגוד לשיטות כמו K-Means, DBSCAN לא דורש לקבוע מראש את מספר הקבוצות, אלא מזהה באופן דינמי אזורים בעלי ריכוז גבוה של תאונות.

יתרונות מרכזיים:

* מתמודד היטב עם נתונים מרחביים ומפוזרים.
* יודע לזהות נקודות "חריגות" ולסמן אזורים בעלי ריכוז גבוה של תאונות.
* לא רגיש לרעש – מסוגל לזהות אשכולות גם כאשר יש נתונים מבודדים.

תרומת המודל לפרויקט:

* איתור מוקדי סיכון בתנועה, תוך שימוש בקורדינטות הגיאוגרפיות של התאונות.
* זיהוי קשרים בין אזורי סכנה לסוגי הכבישים, השלטים והתמרורים במקום.
* מתן המלצות לרשויות התחבורה/ בעלי עניין על מקומות בהם יש צורך בשיפור התשתית.

**3. LSTM – חיזוי מגמות תאונות לאורך זמן**

LSTM (Long Short-Term Memory) הוא סוג של רשת נוירונים רקורסיבית (Recurrent Neural Network - RNN), המתמחה בסדרות זמן. בשונה ממודלים רגילים, LSTM שומר מידע על אירועים מהעבר כדי לבצע תחזיות מדויקות יותר על המגמות העתידיות.

יתרונות מרכזיים:

* מתמודד עם סדרות זמן מורכבות.
* לוקח בחשבון הקשרים ארוכי טווח ולא רק נתונים נקודתיים.
* מאפשר זיהוי דפוסים חוזרים בתאונות, למשל עלייה בתקופות מסוימות של השנה.

תרומת המודל לפרויקט:

* ניתוח מגמות תאונות לאורך זמן והבנה של תבניות חוזרות.
* חיזוי מספר התאונות בעתיד בהתבסס על נתוני שנים קודמות.
* זיהוי דפוסים עונתיים – האם יש יותר תאונות בתקופות מסוימות (למשל חורף, חגים, שעות עומס).

**4. K-Means Clustering – זיהוי דפוסי תאונות לפי סוגי כבישים**

K-Means הוא אלגוריתם אשכולות קלאסי בלמידה לא מפוקחת, המאפשר לקבץ נתונים לפי דמיון ביניהם. האלגוריתם מחלק את הנתונים למספר קבוצות מוגדר מראש כך שהנקודות בתוך כל קבוצה יהיו קרובות זו לזו.

יתרונות מרכזיים:

* מאפשר חלוקה ברורה של הנתונים לקבוצות.
* מבצע ניתוח קבוצתי בצורה יעילה ומהירה.
* מתאים במיוחד לניתוח דפוסי תאונות על פי מאפיינים שונים.

תרומת המודל לפרויקט:

* זיהוי סוגי כבישים בהם יש יותר תאונות ושיוכם לקבוצות.
* הבנה אילו גורמים משותפים משפיעים על קבוצות מסוימות של תאונות.
* ניתוח האם סוג הדרך, מהירות מותרת, והתמרורים משפיעים על כמות התאונות.

**5. Logistic Regression – ניתוח ההסתברות לתאונה חמורה**

רגרסיה לוגיסטית היא מודל למידה מפוקחת המשמש לחיזוי הסתברות של מאורע מסוים. היא אידיאלית לניתוח בעיות של סיווג בינארי או רב-ערכי, ולכן מתאימה במיוחד לחיזוי חומרת התאונה.

יתרונות מרכזיים:

* קל לפרש את התוצאות ולהבין את תרומת כל משתנה.
* מודל יציב שמתמודד היטב עם כמויות גדולות של נתונים.
* מאפשר חיזוי הסתברותי – מה הסיכוי שתאונה תהיה קטלנית או חמורה.

תרומת המודל לפרויקט:

* חיזוי ההסתברות שחומרת תאונה תהיה קטלנית או קשה בהתבסס על גורמים סביבתיים.
* הבנת התרומה של כל משתנה לדרגת החומרה – למשל, האם מהירות מותרת גבוהה מגבירה את הסיכוי לתאונה קטלנית.
* יצירת מודל פשוט ואמין שניתן לשלבו עם מודלים אחרים לצורך הסבר מעמיק יותר.

**6. ניתוח נתונים חקרני (Exploratory Data Analysis - EDA)**

בשלב זה בוצע ניתוח ראשוני וחקרני לנתונים המשולבים של השנים 2022–2023, במטרה להבין את המבנה הכללי של המידע, לזהות דפוסים, לבדוק קורלציות בין מאפיינים שונים ולבצע סינון של נתונים חריגים או לא רלוונטיים. ניתוח זה חיוני לצורך התאמת הנתונים למודלים של למידת מכונה ולגיבוש תובנות מקדימות שישפיעו על המשך העבודה בפרויקט.

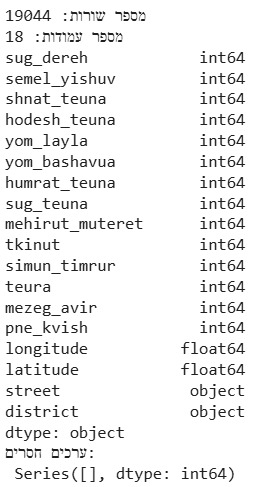
**משתנה המטרה (Target Variable)**

בחרנו בעמודה "חומרת התאונה" (humrat\_teuna) כמשתנה המטרה (target variable) משום שהיא מייצגת את הערך שאותו נרצה לחזות ולנתח – האם התאונה הייתה קלה, קשה או קטלנית. הבחירה נעשתה בהתאם למטרת הפרויקט: להפחית את שיעור התאונות החמורות והקטלניות בישראל על ידי איתור מוקדי סיכון והבנה אילו תנאים מביאים לתאונות חמורות.

פעולות שבוצעו במסגרת EDA:

* איחוד שנים לקובץ אחיד – הנתונים אוחדו משתי שנות דיווח (2022 ו-2023) לצורך ניתוח מגמות רחב ומבוסס יותר.
* המרת קואורדינטות למיקום גיאוגרפי – העמודות X ו-Y הומרו לקואורדינטות latitude ו-longitude, מה שאפשר ליצור עמודת רחוב מדויקת (street) באמצעות reverse geocoding.
* ניקוי נתונים – הסרה של ערכים חסרים או חשודים, כמו רחובות בערבית או ערכים מסוג Unknown, כדי להבטיח נתונים נקיים לניתוח.
* תיאור משתנים – יצירת טבלאות תדירות וסטטיסטיקה תיאורית למשתנים מרכזיים כמו: יום בשבוע, סוג הדרך, מהירות מותרת, מזג אוויר, תקינות דרך ועוד.
* ויזואליזציות מקדימות – גרפים שונים נותחו ונבנו כדי לזהות מגמות ודפוסים (לדוגמה: תאונות לפי יום בשבוע, חומרת תאונות לפי מזג אוויר, אזורים גיאוגרפיים מסוכנים וכו').
* בדיקת קשרים בין משתנים – נבדקו מתאמים בין משתנים (מטריצות קורלציה) והושם דגש על הקשרים בינם לבין חומרת התאונה. לדוגמה: בדיקת הקשר בין מהירות מותרת לחומרת התאונה.

**מבנה קובץ הנתונים הסופי:**



התמונה מציגה את מבנה מסגרת הנתונים (DataFrame) כפי שהודפס לאחר טעינה ועיבוד בסביבת Google Colab באמצעות ספריית pandas, שהיא ספרייה מרכזית לעבודה עם נתונים טבלאיים בפייתון.

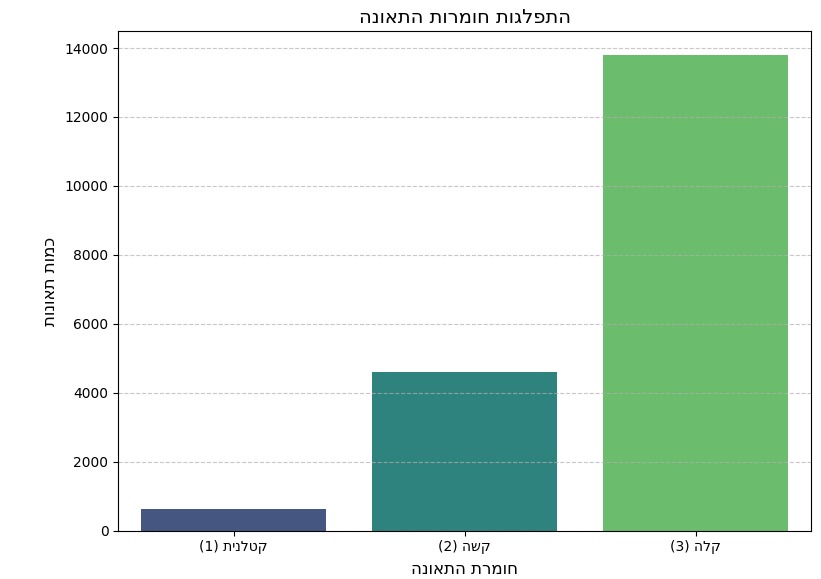
ראשית, ניתן לראות את מספר השורות בקובץ – כלומר, מספר הרשומות הכולל בדאטה במקרה שלנו: 19,044 שורות כאשר כל שורה מייצגת אירוע תאונה בודד עם כלל מאפייניו.

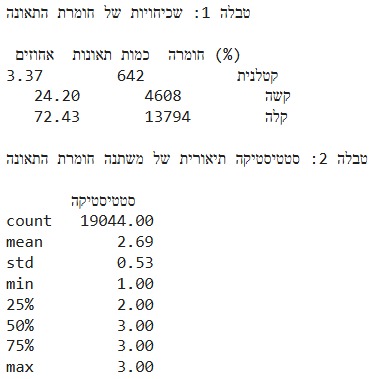
בנוסף, מוצג גם מספר העמודות – כלומר, מספר המשתנים שיש על כל תאונה, ובמקרה זה 18 עמודות שונות.

בהמשך מופיעה רשימה של שמות העמודות יחד עם סוג הנתונים (dtypes) של כל משתנה, מה שמאפשר לזהות את סוג המידע שכל עמודה מחזיקה.

* עמודות מסוג מספרים שלמים – לרוב משמשות למשתנים מקודדים.
* עמודות מסוג float64  – מתארות ערכים רציפים, במקרה הזה קואורדינטות גיאוגרפיות.
* עמודות מסוג object  – מייצגות טקסטים/מחרוזות, למשל שמות רחובות (street) או מחוזות (district)שהתווספו בתהליך.

בסיום הפלט, יש שורת בדיקה שמראה את כמות הערכים החסרים (Null Values) בעמודות – ולפי הפלט: לא נמצאו עמודות עם ערכים חסרים, דבר המעיד על כך שהנתונים נקיים ומוכנים לניתוח ראשוני.

**נתונים סטטיסטיים עבור משתנה המטרה ״חומרת התאונה״:**



טבלה 1: שכיחויות של חומרת התאונה

טבלה זו מציגה את התפלגות רמות חומרת התאונה על פי שלוש קטגוריות: קלה, קשה, ו-אחרים. מתוך 19,044 התאונות:

72.43% מהן היו תאונות קלות, המהוות את הרוב המוחלט – 13,794 תאונות.

24.20% היו תאונות קשות – 4,608 מקרים.

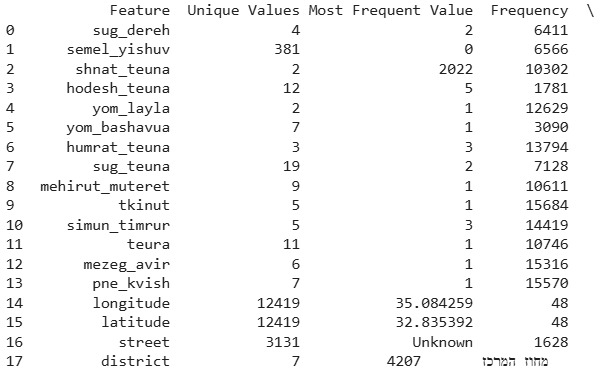
3.37% הוגדרו כקטלניות – 642 תאונות.

מסקנה: מרבית התאונות אינן חמורות. עם זאת, כ-1 מתוך כל 4 תאונות מוגדרת כקשה – נתון מדאיג שראוי לבחינה מעמיקה יותר (למשל: באילו תנאים או מקומות יש שיעור גבוה יותר של תאונות קשות).

טבלה 2: סטטיסטיקה תיאורית של חומרת התאונה

טבלה זו מספקת תיאור מספרי של העמודה חומרת התאונה, המיוצגת כאן בצורה מקודדת מספרית:

* מספר התאונות count: 19,044 – אין ערכים חסרים.
* ממוצע mean: 2.69 – קרוב מאוד ל-3, מה שמעיד שרוב הערכים הם ברמת חומרה נמוכה (נניח שחומרה קלה מקודדת כ-3).
* סטיית תקן std: 0.63 – מראה פיזור בינוני סביב הממוצע.
* מינימום: 1, מקסימום: 3 – כלומר, יש 3 רמות חומרה מוגדרות.
* הרבעון הראשון (25%) הוא 2, החציון (50%) הוא 3, והרבעון העליון (75%) גם הוא 3 – שוב, חיזוק לכך שיותר מ-50% מהתאונות הן מהדרגה הנמוכה ביותר (קלה).

**נתונים סטטיסטיים עבור כלל המשתנים:**

בהתבסס על הנתונים שלנו קיבלנו טבלה המסכמת את מאפייני הנתונים במסד הנתונים שלנו – לאחר עיבוד ראשוני. בטבלה זו נבחנה כל עמודה (Feature) במסד הנתונים מבחינת שלושה פרמטרים מרכזיים:

* מספר הערכים הייחודיים שקיימים בעמודה (Unique Values)
* הערך השכיח ביותר באותה עמודה (Most Frequent Value)
* מספר הפעמים שהערך השכיח מופיע (Frequency)

לצורך הבנה מדויקת יותר, ביצענו הצלבה עם "מילון הקידוד" אשר מפרש את הערכים המספריים לטקסט.

**sug\_dereh – סוג הדרך:**

יש 4 ערכים שונים כאשר הערך השכיח הוא 2, שהופיע 6,411 פעמים.

לפי המילון, ערך 2 מתייחס לדרך עירונית או בינעירונית.

**semel\_yishuv – קוד יישוב:**

מופיעים 381 ערכים ייחודיים, משקף פיזור רחב של התאונות ביישובים שונים.

**shnat\_teuna – שנת התאונה:**

קיימות רק 2 שנים (2022, 2023), כשהשנה השכיחה היא 2022 (10,302 מופעים). מצביע על כך שרוב הנתונים נאספו בשנת 2022.

**humrat\_teuna – חומרת התאונה:**

3 רמות חומרה (כנראה: קלה, קשה, קטלנית), כשהשכיח ביותר הוא 3, עם 13,794 מופעים. לפי המילון, 3 מציין "קלה", כלומר רוב התאונות הן לא חמורות.

**sug\_teuna – סוג התאונה:**

19 ערכים שונים. הערך השכיח ביותר הוא 2, עם 7,128 מקרים.

לפי המילון, 2 מציין תאונה בין כלי רכב, שהיא סוג התאונה הנפוץ ביותר בנתונים.

**mehirut\_muteret – מהירות מותרת:**

9 ערכים שונים. הערך השכיח ביותר הוא 1, עם 11,328 מופעים.

ערך 1 לפי המילון מייצגת מהירות של 50 קמ"ש (שכיחה בדרכים עירוניות).

**simun\_timrur – סימון ותמרור:**

5 ערכים שונים. הערך 1 מופיע 14,419 פעמים – ככל הנראה מעיד על נוכחות תמרורים תקניים.

**mezeg\_avir – מזג אוויר:**

6 ערכים שונים. הערך השכיח הוא 1, מופיע 15,316 פעמים – כנראה מציין "מזג אוויר נאה".

**longitude / latitude – קואורדינטות:**

קיימים 12,419 ערכים ייחודיים – מעיד על פריסה גיאוגרפית רחבה של האירועים.

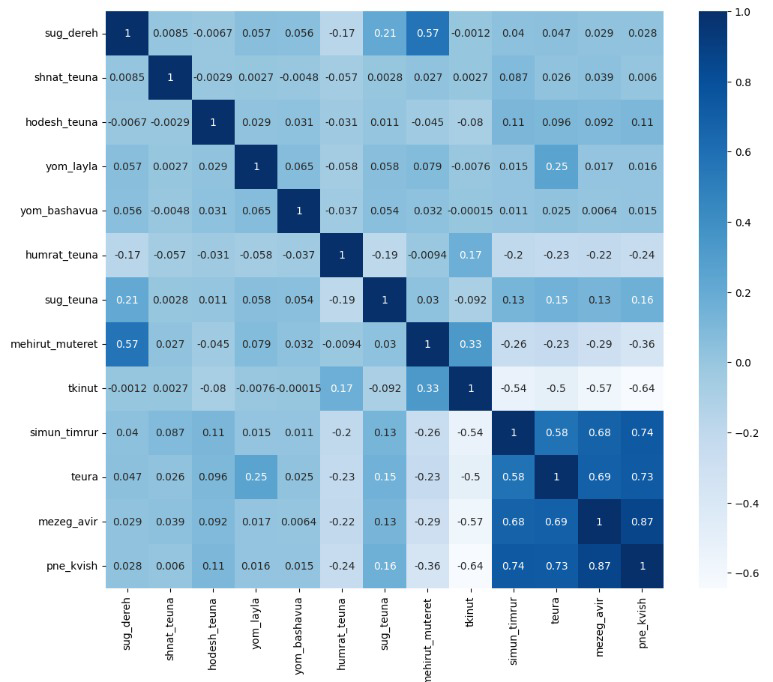
הערך השכיח ביותר מופיע רק 48 פעמים – כלומר אין מוקדים ש"מושכים" את רוב התאונות.

**street – שם רחוב:**

3,131 שמות רחוב שונים.

**district – מחוז:**

7 ערכים שונים (בהתאם לחלוקה המנהלית בישראל). המחוז השכיח הוא מחוז המרכז עם 4,207 מופעים.

**מטריצת הקורלציה:**

**תובנות עיקריות מהקורלציה:**

1. קורלציות חזקות מאוד (מעל 0.7):

simun\_timur, teura, mezeg\_avir, ו-pne\_kvish מראים ביניהם קשרים חזקים מאוד:

* simun\_timur ו-pne\_kvish: 0.74
* mezeg\_avir ו-pne\_kvish: 0.87
* teura ו-mezeg\_avir: 0.73

ייתכן שעמודות אלו נגזרות אחת מהשנייה או שהן תלויות זו בזו בקונטקסט של תנאי הדרך או מזג האוויר בזמן התאונה.

1. קורלציות שליליות משמעותיות:

* tkinut מראה גם קורלציה שלילית עם mezeg\_avir ו-simun\_timur.
* tkinut עם pne\_kvish בקורלציה של -0.64

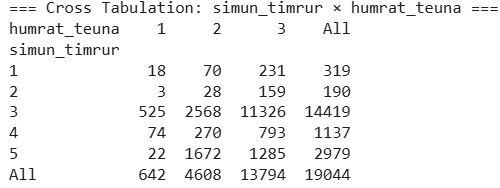
ייתכן שכאשר הכביש לא תקין, תנאי פני הכביש מחמירים.

1. עמודות עם קשר חלש או אפסי:

עמודות כמו shnat\_teuna, hodesh\_teuna, ו-yom\_layla מראות קורלציות מאוד חלשות (קרוב ל-0) עם רוב העמודות — כלומר אין קשר ליניארי מובהק בינן לבין שאר המאפיינים.

1. קורלציות בינוניות:

mehirut\_muteret עם sug\_dereh – קורלציה של 0.57 מצביעה על קשר בינוני: סוג הדרך כנראה משפיע על המהירות המותרת.

**קשרים בין משתנה המטרה לשאר המאפיינים בטבלת שכיחויות**:

הטבלה המוצגת היא טבלת שכיחויות אשר בוחנת את הקשר בין משתנה המטרה ״חומרת התאונה״ לבין משתנה מסביר אחר - ״סימון ותמרור במקום התאונה״. טבלה זו מאפשרת להבין כיצד התפלגות רמות החומרה של התאונה משתנה בהתאם לרמת סימון הדרך. כל תא בטבלה מציג את מספר התאונות שהתרחשו עבור צירוף מסוים של שני משתנים אלו. בסך הכול, מדובר ב־19,044 תאונות אשר חולקו לפי 5 דרגות של סימון (1 עד 5), ולפי 3 דרגות חומרה (1 – קטלנית, 2 – קשה, 3 – קלה).

תובנות:

| **קוד (simun\_timrur)** | **תיאור הקטגוריה** | **ניתוח** |
| --- | --- | --- |
| **1** | סימון לקוי / חסר | שיעור גבוה יחסית של תאונות קשות (5.6%) – מה שמעיד על חשיבות הסימון הפיזי על פני התמרור בלבד. |
| **2** | תמרור לקוי / חסר | שיעור נמוך יחסית של תאונות קשות (1.6%). ייתכן שמדובר בכמות מקרים קטנה או שהתמרור עצמו פחות משפיע מאשר סימון ממשי. |
| **3** | אין ליקוי | הקטגוריה הנפוצה ביותר – כ־76% מהתאונות. עם זאת, גם כאשר אין ליקוי, התרחשו תאונות קשות (כ־3.6%), מה שמעיד כי תמרור לבדו אינו מונע תאונות. |
| **4** | לא נדרש תמרור | שיעור גבוה יחסית של תאונות קשות (6.5%) – ייתכן שבמקומות בהם לא נדרש תמרור, נוצרות סיטואציות מפתיעות עבור הנהגים. |
| **5** | לא ידוע | מופיעה רבות אך עם שיעור נמוך של תאונות קשות (0.7%). מדובר בקטגוריה עם חוסר דיווח מדויק או נתונים חסרים ולכן אנו נתייחס אליה בזהירות. |

**נספחים**

1. קישור לקולאב עבור הגרפים (חלק 6):

[**https://colab.research.google.com/drive/1ZcayZ8SdVLoTYZvUW8\_aBHhXt6Ffp7rj?usp=sharing**](https://colab.research.google.com/drive/1ZcayZ8SdVLoTYZvUW8_aBHhXt6Ffp7rj?usp=sharing)

1. הקוד הבא טוען קובץ CSV עם קואורדינטות במערכת ישראלית שקיבלנו בקובץ הנתונים (EPSG:2039) וממירה אותן למערכת גיאוגרפית עולמית (WGS84 - EPSG:4326) באמצעות הספרייה pyproj ומוסיף את עמודות Latitude  ו־Longitude  החדשות לקובץ נתונים חדש.

import pandas as pd

from pyproj import Transformer

# Use the correct CRS transformation

transformer = Transformer.from\_crs("EPSG:2039", "EPSG:4326", always\_xy=True)

# Load the entire CSV file

file\_path = "/Users/x\_punker\_x/Downloads/2022\_1.csv"

data = pd.read\_csv(file\_path)

# Convert the ITM X (Northing) and Y (Easting) coordinates to Latitude and Longitude

coordinates = data.apply(

lambda row: transformer.transform(row["X"], row["Y"]), axis=1, result\_type="expand"

)

coordinates.columns = ["Latitude", "Longitude"]

# Merge the converted coordinates back into the original DataFrame

data = pd.concat([data, coordinates], axis=1)

# Save the modified dataset back to the original file (keeping all columns)

data.to\_csv(file\_path, index=False)

print(f"Converted coordinates saved to {file\_path}")

1. הקוד הבא טוען קובץ Excel עם קואורדינטות (latitude, longitude), ממיר אותן לשמות רחובות באמצעות Geocoding הפוך באמצעות geopy ו־Nominatim (ספריות) מוסיף את שמות הרחובות לקובץ הנתונים החדש, ושומר את הקובץ המעודכן בפורמט Excel ​​ - (הקוד לא נכנס בדף אחד אבל נמצא בתיבת טקסט באופן מלא למטה). בנוסף, השתמשנו באותו הקוד על מנת לחלץ את שם המחוז גם כן, ההבדל ביניהם הוא שקוד 1 שולף שמות רחובות, מתאים לניתוח ברמת מיקרו לאיתור כתובת מדויקת לעומת קוד 2 שולף שמות מחוזות, מתאים לניתוח ברמת מאקרו ובעצם מבוסס על חלוקה מנהלית של המדינה.

במהלך פיתוח תהליך המיפוי הגיאוגרפי, ביצענו מספר בדיקות קריטיות על מנת להבטיח את שלמות הנתונים, מניעת קריסות ותמיכה בריצות חוזרות. הקוד נבנה כך שיבדוק מראש את תקינות הקואורדינטות – כולל בדיקה האם הערכים חסרים (NaN), לא סופיים או בלתי ניתנים להמרה. במקרים שבהם זוהתה שגיאה בעת ביצוע Geocoding הפוך, ביצענו טיפול בחריגות שמנע את עצירת התהליך והחזיר ערך ברירת מחדל של "Unknown". כמו כן, שילבנו מנגנון השהיה (RateLimiter) כדי לעמוד בדרישות הגישה אל השרת Nominatim . בנוסף, הקוד נשמר לריצות ארוכות באופן מבוקר: כל 500 קואורדינטות שעובדו, נשמרו בקובץ זמני עם ההתקדמות עד אותה נקודה. כך יכלנו להמשיך את ההרצה גם אם הופסקה באמצע – מבלי לאבד מידע שכבר נאסף.  
התהליך כולו של הרצת נמשך כ־4 שעות (סה״כ 2 הרצות), בשל היקף הנתונים הגדול והצורך בהשהיה בין כל קריאה לשירות המיפוי – מה שהדגיש את החשיבות של טיפול זהיר, מבוקר ועמיד לשגיאות.

ההבדל ברמת הקוד:

**קוד שמחזיר שמות רחובות (street):**

# שליפת שם רחוב

return address.get('road', 'Unknown') # road = שם הרחוב

**קוד שמחזיר שמות מחוזות (district):**

# שליפת שם מחוז

return address.get('state', 'Unknown') # state = שם המחוז

הקוד המלא בעמוד הבא...

import pandas as pd

from geopy.geocoders import Nominatim

from geopy.extra.rate\_limiter import RateLimiter

from functools import partial

import math

# Replace with your actual file name

df = pd.read\_excel('/Users/x\_punker\_x/FinalYear/sandraly/input.xlsx')

print("Excel file loaded successfully.")

geolocator = Nominatim(user\_agent="my\_geocoding\_app")

reverse\_geocode = RateLimiter(partial(geolocator.reverse, language='he'), min\_delay\_seconds=1)

unique\_coords = df[['latitude', 'longitude']].drop\_duplicates()

unique\_coords = unique\_coords[

unique\_coords['latitude'].notna() &

unique\_coords['longitude'].notna() &

unique\_coords['latitude'].apply(math.isfinite) &

unique\_coords['longitude'].apply(math.isfinite)

]

total\_coords = len(unique\_coords)

print(f"Processing {total\_coords} unique coordinates...")

def get\_street\_for\_unique(row, counter=[0], results={}):

counter[0] += 1

lat, lon = row['latitude'], row['longitude']

try:

location = reverse\_geocode((lat, lon))

street = location.raw['address'].get('road', 'Unknown') if location and 'address' in location.raw else 'Unknown'

results[(lat, lon)] = street

print(f"Processed {counter[0]}/{total\_coords} ({(counter[0]/total\_coords)\*100:.1f}%)")

if counter[0] % 500 == 0: # Save every 500 coordinates

temp\_df = df.copy()

temp\_df['street'] = temp\_df.apply(

lambda r: results.get((r['latitude'], r['longitude']), 'Unknown'), axis=1)

temp\_df.to\_excel('/Users/x\_punker\_x/FinalYear/sandraly/temp\_output.xlsx', index=False, engine='openpyxl')

print("Saved temporary output to 'temp\_output.xlsx'.")

return street

except Exception as e:

print(f"Error at {counter[0]}/{total\_coords}: {e}")

return 'Unknown'

unique\_coords['street'] = unique\_coords.apply(get\_street\_for\_unique, axis=1)

coord\_to\_street = unique\_coords.set\_index(['latitude', 'longitude'])['street'].to\_dict()

df['street'] = df.apply(

lambda row: coord\_to\_street.get((row['latitude'], row['longitude']), 'Unknown')

if pd.notna(row['latitude']) and pd.notna(row['longitude'])

and math.isfinite(row['latitude']) and math.isfinite(row['longitude'])

else 'Unknown',

axis=1

)

df.to\_excel('/Users/x\_punker\_x/FinalYear/sandraly/output.xlsx', index=False, engine='openpyxl')

print("Processing complete. Output saved to 'output.xlsx'.")