**מבוא למערכות לומדות – תרגיל 2**

# **תרגיל מחייב – נקודות ראויות לציון**

* היות וסדר הפעולות חשוב, מצוינות פה הפעולות לפי סדר הופעתן בקוד:

1. **Data Splitting:**
   1. בצענו פיצול לשם שמירת הסטים המקוריים בקובץ נפרד.
   2. שמרנו את אינדקסי הפיצול על-מנת שנוכל להמשיך לעבוד על ה-dataFrame המלא (ללא פיצול) ולפצל לאותן הקבוצות בעתיד.
2. **Type/Value modification:**
   1. הפכנו את כל הערכים למיספריים בצורה אוטומטית (על-גבי כל ה-data, ללא חלוקה לסטים).
3. **Identify and set the correct type of each attribute :**
   1. הסרנו ערכים שליליים מהמערכת על-ידי הפיכתם ל-NaN, מתוך הנחה כי כל הערכים אמורים להיות אי-שליליים.
4. **Imputation:**
   1. פצלנו את ה-data ל-train/validation/test בשנית.
   2. השלמנו לכל אחד מהסטים את הערכים החסרים בצורה הבאה:
      1. לערך float השתמשנו ב-median של train.
      2. לערך str השתמשנו ב-most\_common של train.
5. **Outlier Detection:**
   1. השתמשנו ב-LocalOutlierFactor מהחבילה sklearn על-מנת לזהות outliers ב-train ויצרנו קבוצה חדשה של train ללא ה-outliers בשם train\_without\_outliers בה השתמשנו בשלבים הבאים.
6. **Normalization:**
   1. לכל עמודה ב-train\_without\_outliers, השתמשנו ב-NormalTest מהחבילה SciPy על מנת לבדוק האם איברי העמודה בעלי התפלגות נורמלית.
   2. עמודות שמתפלגות נורמלית נורמלו על-ידי z-score, והאחרות על-ידי min-max לכל אחד מהסטים הדרושים.
7. **Feature Selection:**
8. הרצנו VarianceThreshold מ-sklearn (?) בצורה שמרנית על train\_without\_outliers, ושמרנו ב- train\_without\_outliers רק את הפיצ'רים הנבחרים.
9. בצורה דומה, הרצנו על train\_without\_outliers את Relief ולאחריו את SFS.
10. יצרנו וקטור עם ערכי 0 לכל שאר התכונות והרצנו Feature Importance Forest, SelectFwe לפי chi2 ו-SelectKBest לפי mutual\_info\_classifier, כולם מ-sklearn על-גבי מה שנשאר מ- train\_without\_outliers. כל אחד מה-wrappers הללו החזיר 1 עבור ה-features שהיו חשובים לדעתו, ולבסוף בחרנו ב-features שקבלו לפחות 2 קולות.