עבודת גמר – סיכום

שיפור מחברת מסמסטר קודם

בסמסטר הקודם עבדתי על דאטה שנקרא 'wine quality' המטרה שלי הייתה לסווג יין ל-2 קטגוריות – טוב ולא טוב. בסמסטר שעבר בעזרת 'Decision Tree' הגעתי ל-accuracy של **90.25%**.

חשבתי שיהיה מעניין לעשות השוואה של 'Random Forest', עליו למדנו בסמסטר הזה, ל-' Decision Tree' וכך לנסות לשפר.

בנוסף בסמסטר הקודם ניסיתי לצמצם עמודות שחשבתי שהן לא רלוונטיות למודל ולא הצלחתי לשפר את הדיוק בדרך זו, אז חשבתי לנסות לצמצם את הממדים בעזרת PCA.

לשמחתי, שני הדברים האלה שיפרו את המודל וכך הצלחתי להגיע למודל יותר קומפקטי (**9 ממדים במקום 11**) עם דיוק גבוה יותר – **91.25%**.

Fashion-mnist

זהו דאטה יחסית פשוט שמגיע מסודר בצורה התחלתית טובה.

צמצמתי את הממדים **מ-784 ל-120** בעזרת PCA ואימנתי מספר מודלים שהותאמו עם הפרמטרים הטובים ביותר עבורם.

לבסוף הגעתי לכך שהמודל הטוב ביותר היה 'Voting Classifier' שהיה בעצם אנסמבל של כל המודלים:

- ☒ KNeighbors Classifier
- ☒ XGB Classifier
- ☒ Random Forest Classifier
- ☒ Logistic Regression

וכך הגעתי לדיוק של: **89.37%**.

בנוסף, ניסיתי לשפר את התוצאה בעזרת שימוש ב-Kmeans וחיפוש ערכים עבורו בעזרת Grid Search אך זה לא שיפר לי את התוצאה ולכן לא עשיתי בזה שימוש במודל הסופי.

Cat VS Dog

כאן קיבלנו 2 קבצים עם תמונות אחד ל-test ואחד ל-train. תחילה ניקיתי מהדאטה תמונות לא רלוונטיות. לאחר מכן טענתי את התמונות ושיניתי להם את הגודל לגודל אחיד ואת הגוונים לגווני אפור (בכדי להקל על המודל) ולבסוף הכנסתי כל תמונה כשורה אחת בטבלה הסופית, ואת הקטגוריה הגדרתי כך – חתול=0, כלב=1.

לאחר חיפוש ארוך אחר המודלים והפרמטרים הטובים ביותר, הגעתי לכך שהמודל הטוב ביותר היה 'Voting Classifier' שהיה בעצם אנסמבל של המודלים:

- ☒ KNeighbors Classifier
- ☒ Extra Trees Classifier
- ☒ XGB Classifier
- ☒ Random Forest Classifier
- ☒ Logistic Regression

כל זה יחד עם PCA שהפחית את הממדים מ-**4,096** ל-**45** ממדים בלבד.

Test :

כיוון שהקטגוריות של הכלבים והחתולים בטסט לא מפורסמות, עשיתי 2 סוגי טסטים (כמו שדיברנו במייל):

1. פיצול 10% מה-train לטובת טסט, ובדיקה עליו בסוף המחברת. והגעתי לדיוק של **64.4%**.
2. טסט על הדאטה טסט המקורי, ושימוש בתחרות פתוחה בקאגל בכדי לקבל 'ציון' למודל על הנתונים מהדאטה-טסט.

בקאגל, כמו שדיברנו, הציון מתקבל ב-Log Loss (שזה בעצם אומדן לכמה רחוק ה- prediction probability מהלייבל האמיתי). קיבלתי Log Loss של **0.71072** (כמובן שב-Log Loss אנחנו רוצים לקבל מספר כמה שיותר נמוך).

בנוסף היו לי עוד 2 רעיונות לשיפור המודל שלא צלחו:

1. שימוש ב-Kmeans – בפעול זה לא שיפר את המודל.
2. עיבוד תמונה – רציתי לחתוך מכל תמונה את אזור הפנים של הכלב/חתול ולמחוק את שאר התמונה, וכך המודל שלי יתמקד רק בהבדלים המהותיים בין החיות ולא ברקע של התמונה, שתופס הרבה מקום (ונתונים) ולא נותן הרבה הבנה האם זה כלב או חתול.

בפועל לאחר ניסיונות רבים ושימוש בספרייה open-CV הבנתי שהידע שלא בנושא לא מספיק לצורך ביצוע המשימה הזו ולכן אני לא יודע אם זה היה יכול לשפר את המודל או לא, אך מעניין לבדוק את זה בהמשך מתי שאשפר את היכולות שלי בעיבוד תמונה.

Hands classification

פה מה שקיבלנו היה נתונים של סרטונים שמוצגים כשורות בטבלאות רבות כך שכל פריים מוצג ב-2 שורות, אחת של יד ימין ואחת של שמאל.
המטרה הייתה לסווג את מצבי תנועת הידיים מבין 3 המצבים הקיימים.

לצורך עיבוד הנתונים, החלטתי לכתוב פונקציות שיפתחו את כל הקבצים וייצרו לי דאטה-פריים אחד גדול של כל הנתונים.

תחילה, חיברתי את כל השורות של יד ימין ויד שמאל. במצב 'alone' חיברתי את יד שמאל עם יד ימין שנועדה לחיבור לכל מצבי ה- 'alone', במצב זה כאשר היה צורך הכפלתי את שורות יד ימין כדי שיהיה מספיק שורות בשביל לשים שורה מימין מול שורה משמאל (מותר להכפיל כי במצב זה אין קשר בין הידיים).

לאחר מכן, בכדי ליצור סוג של 'סרטון' של פריימים, לקחתי פריים אחד מכל 5 פריימים (בדילוגים) וחיברתי מהפריימים שנבחרו חמישיות של פריימים לשורות ארוכות עם 5 פריימים ו-2 ידיים.
לבסוף קיבלתי דאטה עם 180 עמודות.

חלוקת הנתונים ל- train and validation:

כיוון שהניסוי הוא מוגבל ולא רציתי לבזבז נתונים על וולידציה ולא להשתמש בהם בצורה ישירה לאימון המודל, ומצד שני קרוס-וולידציה לא הייתה עובדת פה טוב (מפורט בהמשך), בחרתי להוציא אדם אחד בתור וולידציה ובסוף לאחר בחירת המודל והפרמטרים האידיאליים לאחד חזרה את הטריין והוולידציה לצורך אימון מחודש של המודל האידיאלי לפני בדיקה שלו על הטסט הסופי.

בחרתי להפריד את אחד האנשים בתור וולידציה, כיוון שאם הייתי עושה split רגיל (או קרוס וולידציה) הייתי יכול לקבל בוולידציה פריים קרוב לפריים אחר שב-train וכך הייתה נוצרת סוג של דליפת נתונים והוולידציה לא הייתה נותנת לי תמונת מצב אמיתית לגבי המודל שלי.

המודל הטוב ביותר היה 'Voting Classifier' שהיה בעצם אנסמבל של המודלים:

- ☒ KNeighbors Classifier
- ☒ Extra Trees Classifier
- ☒ XGB Classifier
- ☒ Random Forest Classifier
- ☒ Logistic Regression

כל מודל עם הפרמטרים האידיאליים שלו, יחד עם PCA שהפחית את הממדים מ-**180** ל-**45** ממדים.
לבסוף עם כל זה קיבלתי דיוק של **89.3%**.