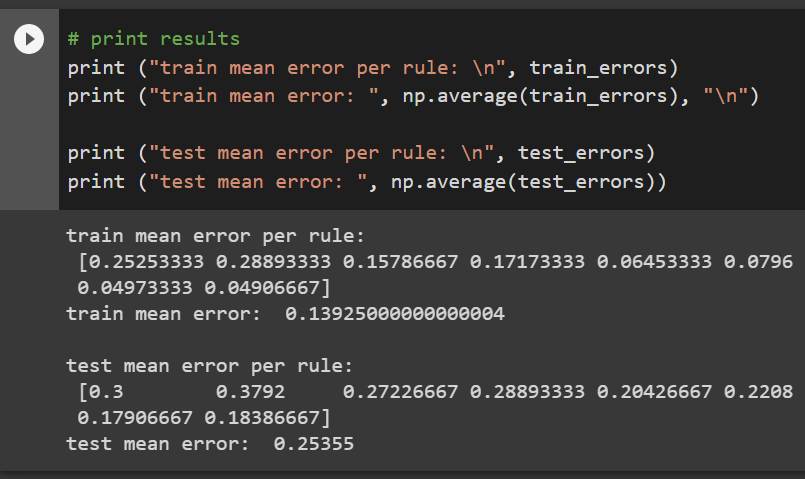
# למידת מכונה - מטלה 3 - חלק תיאורטי

מאיר נזרי – 312237563

איתן ליכטמן – 205463920

\* נפגשנו ועשינו את המטלה ביחד.

לאחר הרצת הקוד אלו התוצאות שקיבלנו. השתמשנו בגרסה Python 3.9.

1. **ניתוח התוצאות**:

* ממוצע הטעות של שימוש בחוק אחד בלבד על ה-train הוא 0.25. הטעות היא מצד אחד לא גבוהה במיוחד כיוון שכל חוק הוא הטוב ביותר על ה-train שממנו נבחר, אך מצד שני הטעות גם לא קטנה מדי כיוון שכל חוק הוא קו ישר ואילו המסווג האופטימלי הנדרש כאן הוא מלבן. על ה-test ממוצע הטעות הוא קצת יותר גבוה כיוון שכל חוק נבחר לפי הטעות המינימלית על ה-train ולא ה-test.
* עבור שני חוקים, ממוצע הטעות גם ב-train וגם ב-test עולה. הסיבה לכך לדעתי היא, שלאחר שנבחר חוק אחד באיטרציה הראשונה, באיטרציה השנייה משקל כל הנקודות שהחוק הראשון טעה עליהם עולה משמעותית, ולכן הרבה פעמים החוק השני שיבחר יהיה החוק ההופכי לחוק הראשון. בחירת החוק השני בדרך זו תגרום לכך שהמודל יטעה על הרבה נקודות שהחוק הראשון סיווג נכון.
* מכאן והלאה, ככל שמוסיפים יותר חוקים יש שיפור גם ב-train וגם ב-test. זוהי התנהגות הגיונית לעיקרון לפיו אלגוריתם Adaboost עובד. אמנם נשים לב שהשיפור הגדול ביותר הוא כאשר משתמשים ביותר מ-4 חוקים. הסיבה לכך היא שכאשר משתמשים ב-4 חוקים ומעלה, כאשר כל חוק מייצג קו ישר, החוקים יכולים ליצור מעין צורה של מלבן שהוא המסווג האופטימלי עבור הדאטה שלנו.
* באופן כללי נשים לב שיש פער לא קטן בין הטעויות על ה-train לטעויות על ה-test. תופעה זו נפוצה בכל המודלים של למידת מכונה, ונובעת מכך שהמודל מנסה למזער את הטעות על ה-train ולא על ה-test.

1. **האם יש Overfitting?**

למרות הפער הגדול בין הטעויות על ה-train ל-test והעקרון שככל שהחוק נהיה מסובך יותר המימד VC גדל, ניתן לראות בתוצאות שלנו שככל שמוסיפים חוקים ל-Adaboost הטעות על ה-test פוחתת, ולכן **אין כאן Overfitting**. הסיבה לכך היא שכאן הוספת חוקים מורידה משמעותית את הטעות על המדגם, ואזי למרות שהמימד VC גדל, החוק הסופי שמתקבל הוא טוב יותר על ה-test.

קישור Jupiter Notebook:

<https://colab.research.google.com/drive/1_O04i66euo5hCGh7gdJucZsv_p4DrtaO?usp=sharing>