

# למידת מכונה - פרויקט גמר

## 1 הקדמה

### 1.1 הגדרה- ML

אלגוריתמים לינאריים אשר לומדים את הדאטא תוך כדי הרצה. כלומר האלגוריתם **משנה את עצמו**, מתקן את המשקל שהוא נותן לפקטורים שונים ומשתכלל מנסיעה לנסיעה. כלומר, ה-Input נשאר לכאורה קבוע, אבל ה-Output משתנה.

ה-ML התפתח לשם שיפור סיבוכיות זמן ריצה.

### 1.2 משפט (מנורו חישוביות)

כל בעיה ניתנת לחישוב ב-3 שכבות (הוכחה לכך שאפשר לפתור כל בעיה באמצעות 3 שכבות ב-DL).

### 1.3 השערות

אלגוריתמי ML לא יתפקדו פחות טוב על תמונות כיוון שהם לא לומדים **קשרים מרחביים**. בניגוד ל-DL שמשמש ברשת CNN הלומדת את המידע החבוי בתמונה גם **ברמה הלוקאלית** וגם **ברמה הגלובלית**. קונבולוציה היא פונקציה לא לינארית המאפשרת לנו ללמוד מידע בצורה שונה.

יכול להיות שאם נעלה את רמת המורכבות של אלגוריתמי ML (מספר נוירונים ברשת, מספר איטרציות, מרחב loss וכדומה) נוכל להגיע לאותה רמת הדיוק של רשת CNN.

## 2 מטרת הפרויקט

אנו שואפים לענות על שתי שאלות:

1. האם אלגוריתמים מסורתיים של למידת מכונה מסוגלים לסווג תמונות למרות מגבלותיהם?

2. כיצד אלגוריתמי חילוף תכונות מתפקדים טוב יותר כמעבדי קדם?

בניסוי הראשון: נעצב מחדש ונרמל תמונות.

בניסוי השני: נזין את הנתונים שחולצו ונבצע תהליך של דחיסת נתונים.

כדי לבדוק את שאלת המחקר שלנו, נשתמש באלגוריתמים הנפוצים ביותר של למידת מכונה ונשווה את התוצאות שלהם עם רשת למידה עמוקה פשוטה.

## 3 ניסוי ראשון: DL vs. ML ללא Data preprocessing

### 3.1 תוצאות

כפי שחזינו DL נותן תוצאות טובות יותר מ-ML.

בניגוד להשערה שלנו אלגוריתמי ML נתנו תוצאות קרובות, יחסית, וטובות.

## 3.2 מסקנות

שאלת השאלה: "מהו סף מורכבות הדאטא עבורה אלגוריתמי ML יצליחו להגיע לתוצאה טובה?"

התשובה לשאלה זו, אולי, תוביל להבנה טובה יותר של הקשר בין אלגוריתמי DL ואלגוריתמי ML לסיווג תמונות.

## 4 ניסוי שני: DL vs. ML עם Data preprocessing ל-Data

### 4.1 הגדרה- AutoEncoder

מקודד אוטומטי מפחית את הממדיות של הנתונים על ידי התמקדות באזורים רלוונטיים. בתהליך זה, נתוני קלט נדחסים, מקודדים ולאחר מכן נבנה מחדש כפלט כדי להפחית רעש. שלב הפענוח משחזר את הנתונים המקודדים, כאשר הפלט מייצג את הקלט באופן שווה.

### 4.2 הגדרה- PCA

PCA מחליץ את הפיצ'רים החשובים ביותר, מקטין את ממד הנתונים וכך משפר את יכולת הפרשנות וממזער את אובדן המידע.

בחרנו להשתמש ב-PCA (אלגוריתם דטרמיניסטי) ולא ב-JL (אלגוריתם הסתברותי- רנדומלי).

### 4.3 תוצאות

PCA נותן תוצאות טובות יותר מ-AutoEncoder בכל האלגוריתמים.

תוצאה זו מפתיעה אותנו כיוון ש-AutoEncoder הוא אלגוריתם מבוסס אופטימיזציה וציפינו שהוא יצליח יותר מ-PCA שהוא אלגוריתם דטרמיניסטי.

### 4.4 מסקנות

אנו מייחסים את תוצאות אלו לעובדה ש-PCA מבודד את הפיצ'רים החשובים כך שהם **בלתי תלויים** במודל, אלא תלויים בדאטא בלבד. לעומת זאת AutoEncoder (DL) הינו אלגוריתם אופטימיזציה ולכן מבודד את הפיצ'רים החשובים כך שהם **תלויים** במודל עצמו ולא בדאטא בלבד.

עוד הסבר יכול להיות לכך ש-AutoEncoder לא הספיק להתכנס במספר האיטרציות הנתון.

## 5 נקודות לשיפור

### 5.1 ניסוי שני

נוכל לבדוק מהו היחס בין מספר הפרמטרים שכל רשת פולטת (דוחסת) לדיוק המודל, נוכל להציג תוצאות אלו באמצעות גרף.