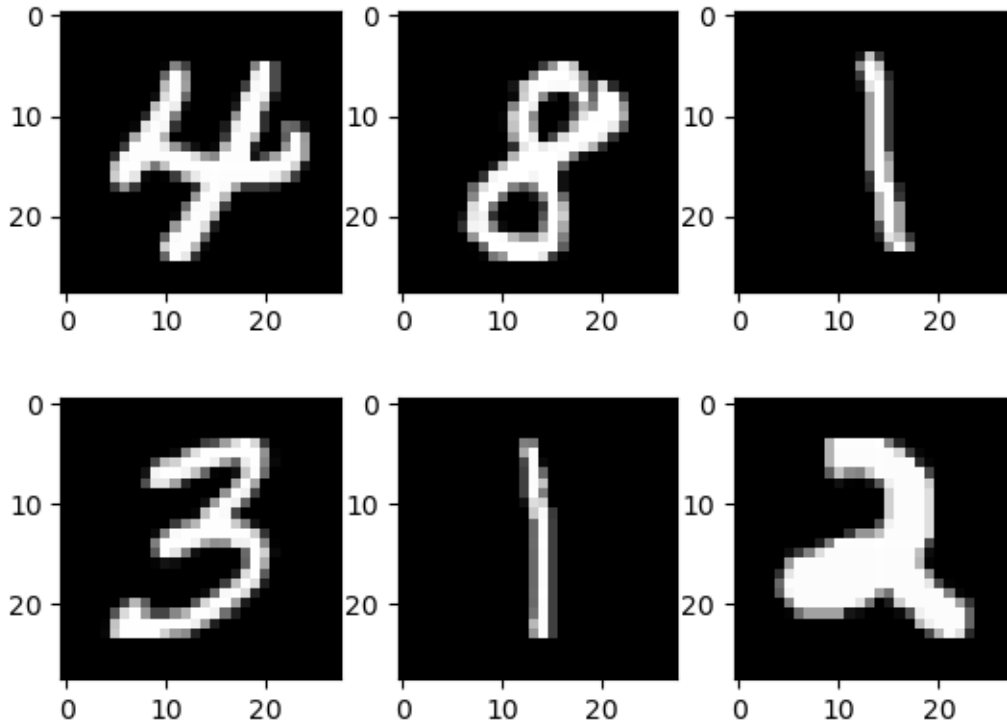


מסמך מלווה ללמידת מכונה על MNIST:

מגיש - איתי גלילי

הנחות היסוד שקבעתי לתהליך בניית המודל:

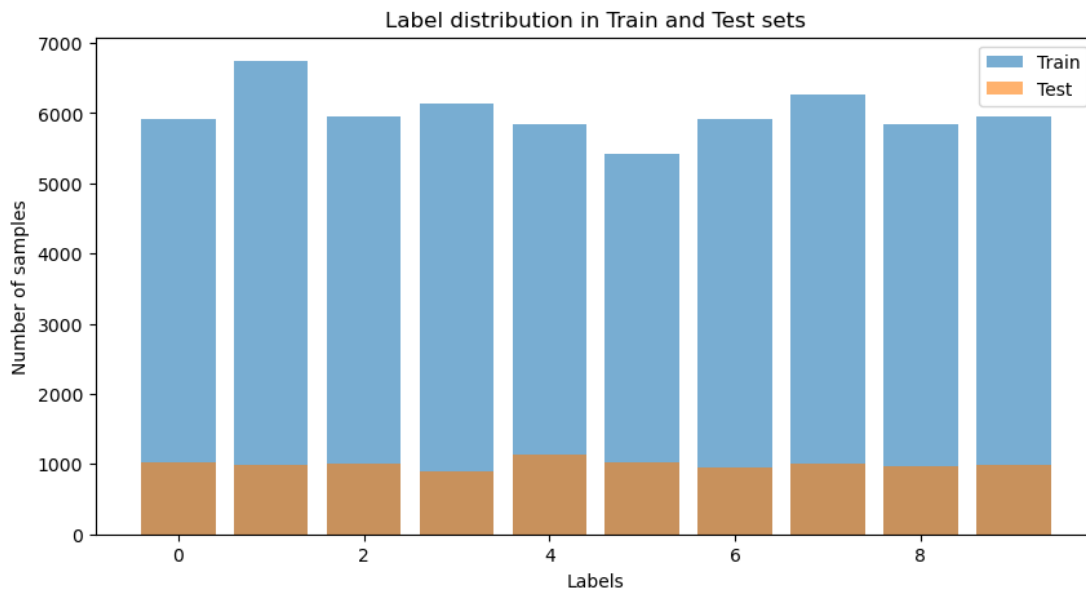
1. מודל ANN, כמתבקש בדרישות הפרוייקט.
2. שכבת קלט – 28×28 נירונים



3. שכבת פלט – 10 נירונים
4. על כן, תהליך המעבר בין השכבות צריך להיות במגמת צמצום נירונים בין שכבה לשכבה.
5. **קריטריון להפסד (Loss Function):** הקוד עושה שימוש ב-CrossEntropyLoss - שהוא קריטריון נפוץ עבור בעיות סיווג.
6. Hyperparameters:
 - a. Learning rate נמדדו האופציות 0.01 או 0.001.
 - b. Batch size
 - c. Num of epochs
 - d. פונקציית אקטיבציה – relu

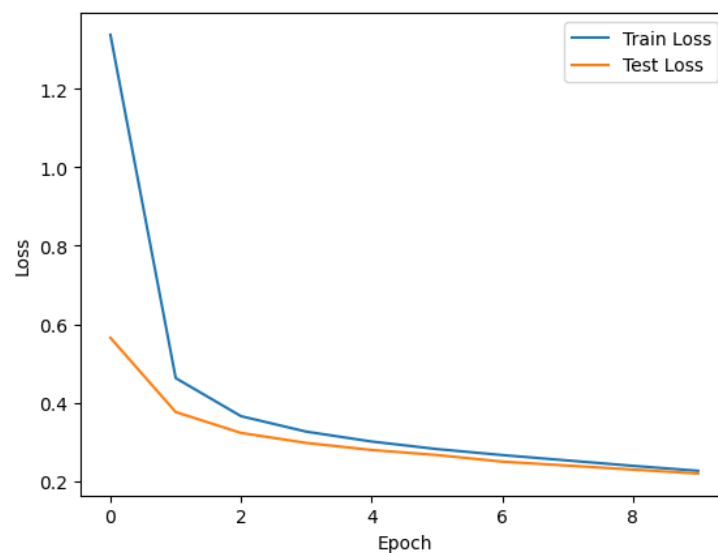
תהליך הלמידה:

1. בדיקת תקינות MNIST, טרם הרצת המודלים, נבדוק שיש לנו מספיק דוגמאות לאמן את המודל, נבדוק גם את החלוקה בין הtrain לtest ובין labels הגיונית:



2. רשת מס' 1:

השכבות:	קצב למידה:	number of epochs
784	0.01	10
128	dropout	
64	NO	
10		

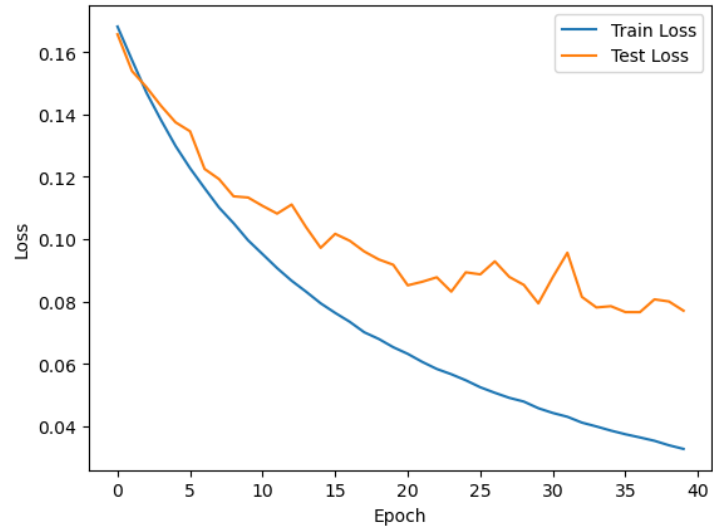


לקחים –

1. להגדיל את כמות המחזורים ליצירת למידה איכותית יותר.

3. רשת מס' 2:

השכבות:	קצב למידה:	number of epochs
784	0.01	40
128	dropout	
64	NO	
10		



לקחים:

1. אפשר לראות שמאוד מהר הtest מתקשה לרדת מ0.08.

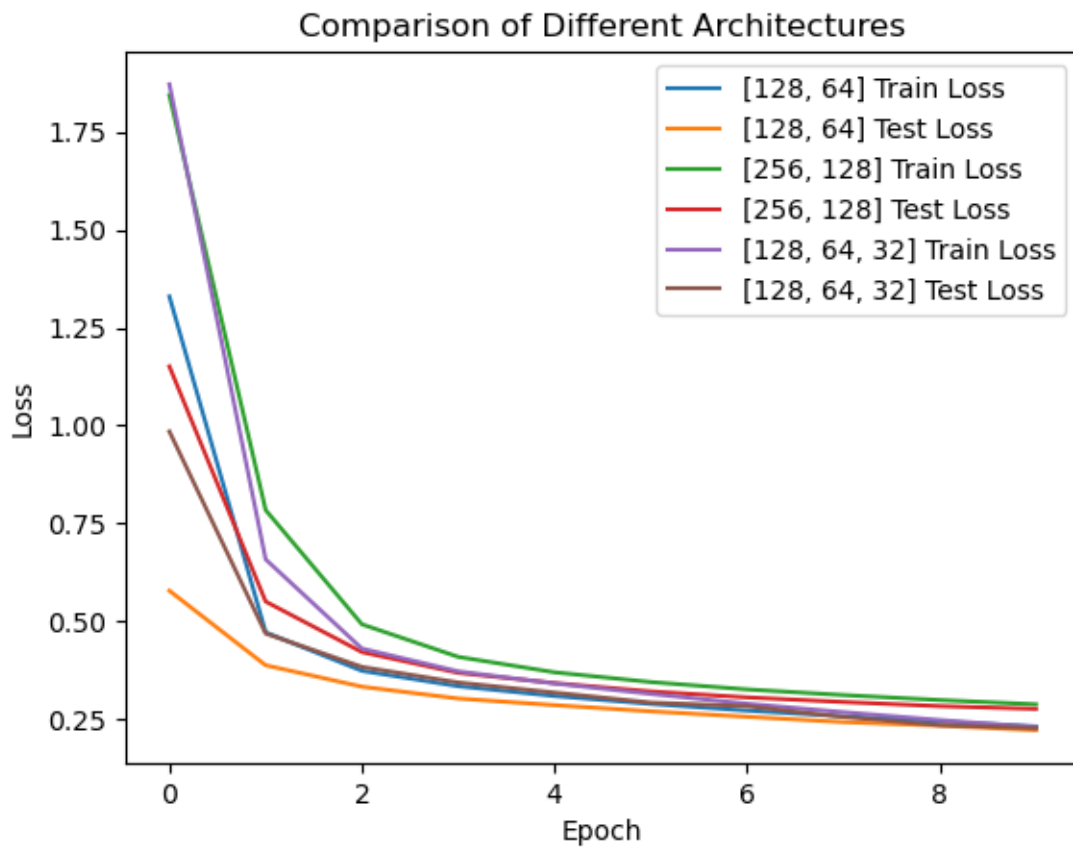
לאחר מכאן, ניסיתי לבצע מספר וריאציות של שינויים בספר וגודל השכבות ושינוי בגודל קצב הלמידה:

4. רשת מס' 3

השכבות:	קצב למידה:	number of epochs
784	0.005	10
256	Dropout	
128	NO	
10		

5. רשת מס' 4

השכבות:	קצב למידה:	number of epochs
784	0.01	40
128	Dropout	
64	NO	
32		
10		



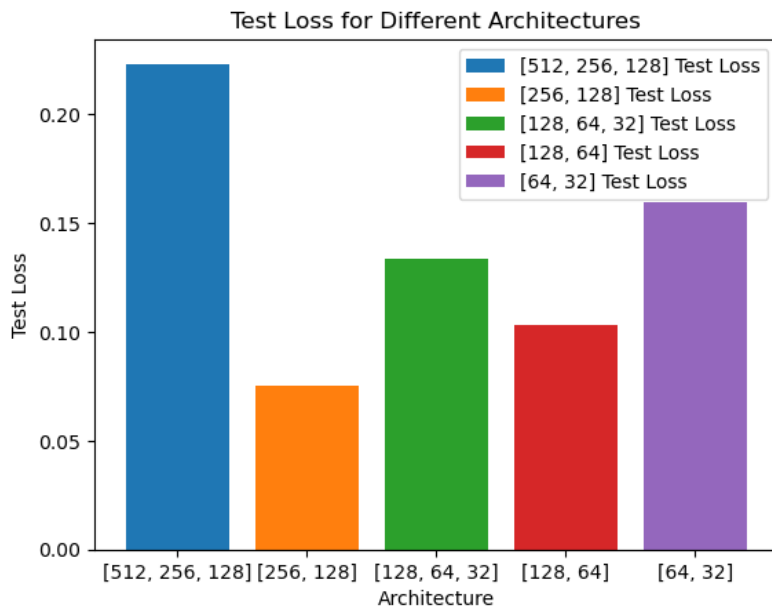
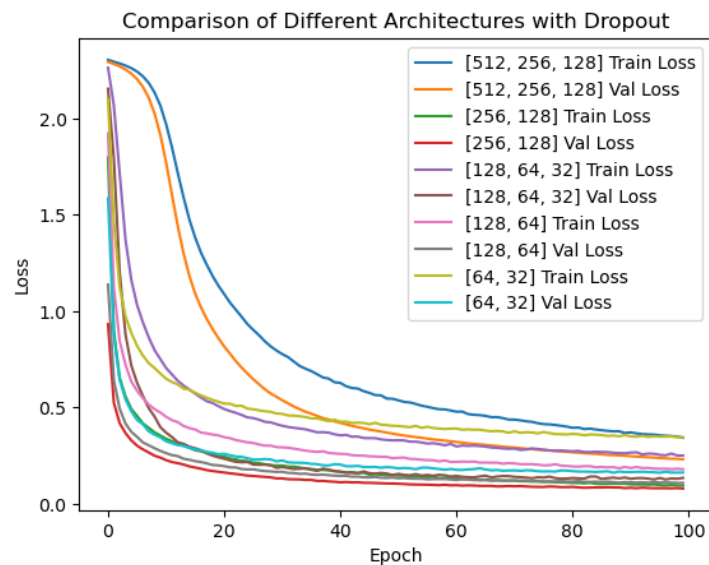
לקחים:

1. ניתן לראות שהבדל בין סוגי השכבות נותן דיוקים שונים (למידה באיכות שונה).

6. הרשתות הבאות (שמהוות את ההרצה הרצינית הראשונה למודל):

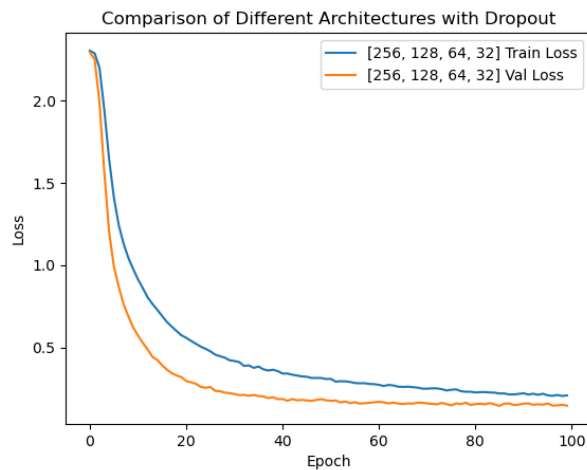
```
architectures = [  
    {'layers': [512, 256, 128], 'learning_rate': 0.001, 'dropout': 0.5},  
    {'layers': [256, 128], 'learning_rate': 0.01, 'dropout': 0.5},  
    {'layers': [128, 64, 32], 'learning_rate': 0.01, 'dropout': 0.5},  
    {'layers': [128, 64], 'learning_rate': 0.01, 'dropout': 0.5},  
    {'layers': [64, 32], 'learning_rate': 0.01, 'dropout': 0.5}  
]
```

עם כמות epochs – 100 לכל רשת, כמו כן, חילקתי את הדאטה עם ולידאציה ובחנתי לאחר מכאן את הtest:



הרשת בעלת השכבות 784,256,128,10 עם dropout וקצב למידה 0.01 עם תוצאות של 0.075!

7. רשת עם יותר שכבות (10,32,64,128,256,784):



loss גבוה של 0.158 (פחות טוב מהמודלים הקודמים).

מסקנות ולמידה מהתהליך:

1. לחלק את הtrain עם ולידציה. לא היה נכון להשתמש בtest בתוך התהליך.
2. הגדרת early stopping מאפשר להימנע מבזבז זמן של ריצה שמתחילה להיכנס ל overfitting.
3. להבא, לנסות פונקציות אקטיבציה נוספות.
4. להגדיר את פונקציות הלמידה עם פרמטרים משתנים, ולהימנע מכתובת קוד מיותרת שיוצרת סתם בלאגן (פעם אחת לאתחל אבסטרקטית את המודל, ואז לרוץ עם הגדרות ה hyper parameters בלבד).
5. להימנע מלהריץ מספר מודלים יחד, זה מעמיס על היכולת לנתח בגרפים.

קישור למחברת:

<https://github.com/ItayGalili/MNIST/blob/main/deeplarning1.ipynb>