**ממן 11 חלק א'**

**נושא הממן: סיווג ספרות**

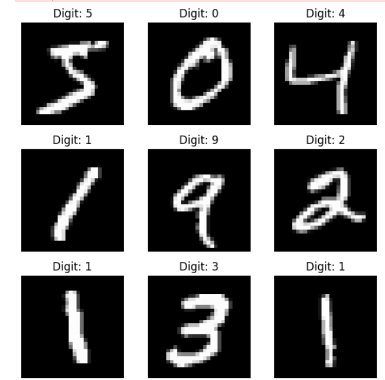
**מטרה:** מטרת הממן היא לאמן את המחשב בזיהוי ספרות אשר נכתבו בכתב יד

**השיטה:** שימוש ב pocket algorithm לסיווג בינארי. נאמן מודלים שונים להבחין בין ספרה מסוימת לכל שאר הספרות. לאחר מכן כשנראה לחזות ספרה חדשה נבדוק כל אחד מהמודלים מול הספרה ונבחר את התוצאה בעל האחוז ביטחון הגבוהה ביותר.

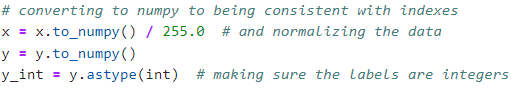
**אלגוריתמים מרכזיים:** שימוש ב pocket algorithm ו PLA

**ספריות שבהם השתמשתי:**

* Numpy – ספריה מתמטית אשר מיועדת לחישובים על מטריצות וקטורים וכו'...
* Matplotlib – ספריה שנועדה לסרטוט של הגרפים השונים
* Sklearn – ספריה שממנה מגיעים הנתונים ומאפשר לבצע מניפולציה על הנתונים.

**הנתונים:** תמונות של מספרים בגודל 28X28 פיקסלים ב gray scale ולכל אחד תווית אשר אומרת איזה מספר מיוצג בתמונה:

**הכנת המידע:** הכנת מידע בוצעה ע"י העברה של כל המידע למטריצות numpy (מגיע בתור pandas series) ונרמול של הפיקסלים לערכים של 0-1 במקום 0-255

תמונה שמכילה טקסט, גופן, צילום מסך

התיאור נוצר באופן אוטומטי

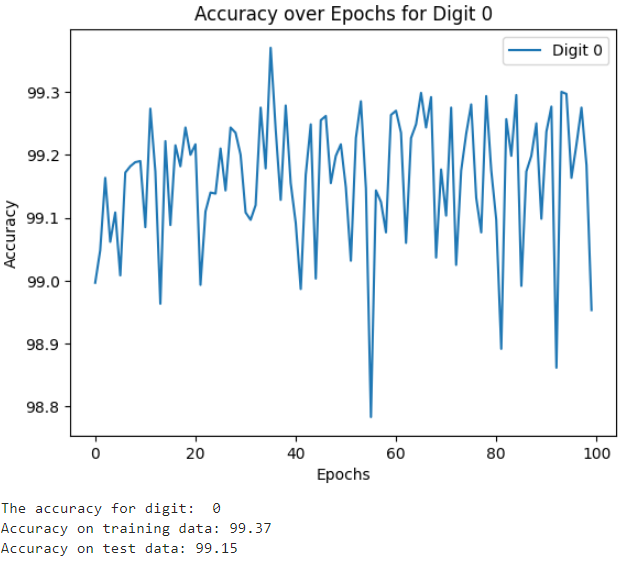
המידע חולק ל 60 אלף תמונות אימון ו 10 אלף תמונות מבחן כפי שהתבקש בממן. בנוסף היה צורך להוסיף את איבר ההטיה.

**האלגוריתם:** לב האלגוריתם הוא ה PLA אשר מנסה לבצע חיזוי על פי סט משקלים. מבצעים מכפלה פנימית בין המשקלים ל וקטור הנתונים שלנו במידה והתוצאה היא חיובית ממש אז ככל הסיווג נכון אחרת התוצאה שלילית. לשם כך נעדכן את סט המשקלים שלנו על ידי הכלל הבא:

W(t+1) = w(t) + x(t) \* y(y) מה שגורם למשקולות "ללכת בכיוון הנכון"

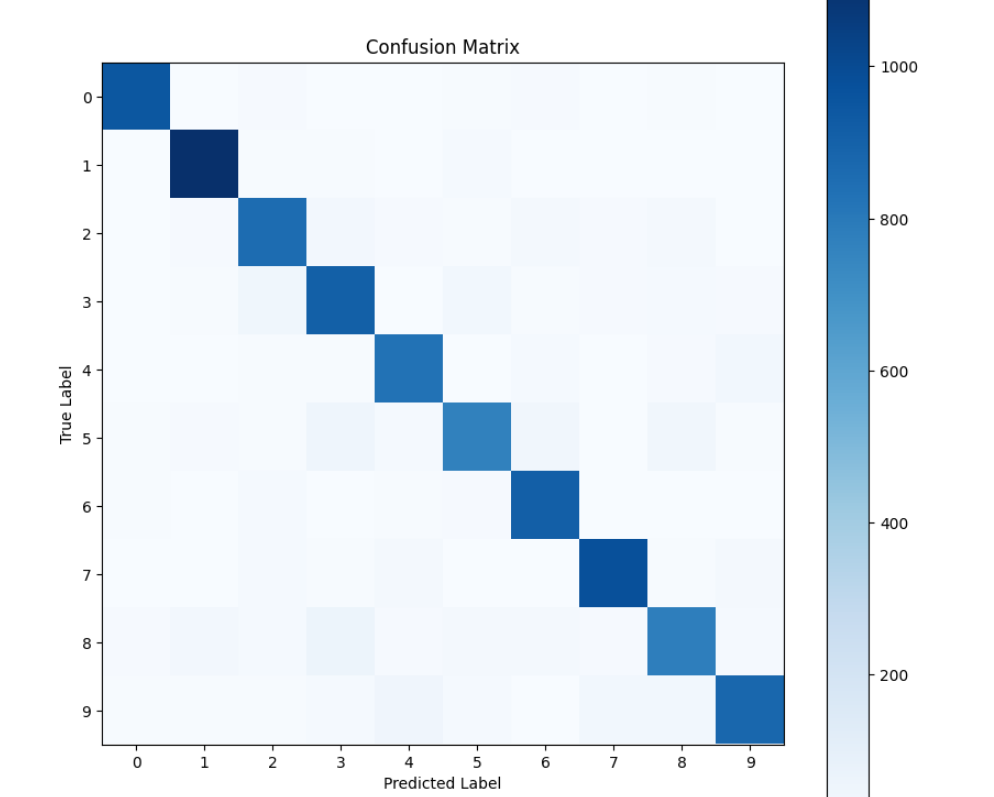
הבעיה עם האלגוריתם הנ"ל ייתכן כי אנחנו פוגעים בסיווגים האחרים ואם ה data הוא לא linearly separable אז לא מובטחת לנו התכנסות.   
הפתרון: שימוש ב pocket algorithm הרעיון הוא כל פעם שנעדכן את סט המשקולות נבדוק אותו על ה data אם הוא נותן לנו אחוזי סיווג גבוהים יותר נשמור אותו בצד (בכיס) ואם אז נישאר עם הקוד.

**ריצת התוכנית:** ניתן לראות כי עבור כל מסווג מתכנס באופן מהיר למדי. לאחר 100 epoches כבר מגיעים לאחוזי סיווג מרשימים פר סיפרה. לדוגמא הספרה 0:



ובאופן כללי האלגוריתם שלנו מציג אחוזי ניבוי מרשימים של כ 90%





עפ"י ה confusion matrix ניתן לראות כי האלגוריתם שלנו מנבא סה"כ באופן דיי טוב את הספרה 0 נראה כי רוב הטעויות נופלות בין 3 ו 5 טעות זו הגיונית כי 3 ו 5 ספרות שהחלק התחתון שלהם דיי וזה והחלק העליון של 3 הוא מעוגל ו ב 5 הוא ישר מה שיכול לגרום לטעויות בזיהוי. ובין 8 ל 5 מסיבה דומה.

סה"כ למרות השימוש במסווג בינארי שפחות מותאם לבעיה של סיווג ספרות לדעתי הגענו לאחוזי סיווג גבוהים. כשמדובר בספרות 0 1 ו 2 ניתן לומר כמעט בוודאות מוחלטת שהסיווג היה נכון ועבור ספרות כמו 3 5 ו 8 הייתי ממליץ לבצע בדיקה נוספת ע"י אלגוריתם אחר