דו"ח לפרויקט מעשי

מגיש : גלעד X ת.ז: X קורס : מבוא לבינה מלאכותית – X838

הקדמה:

גילוי נאות : ההבדל בין ההוראות בתחילת וסוף המשימה ("בנו אלגוריתם למידה" לעומת " מומלץ להשתמש בשיטות הכלליות וה"כבדות "ובעיקר ברשתות נוירונים") גרם לי לקצר במוח , אז :

העבודה נעשתה ונכתבה מתוך תפיסת העולם שלי , עצם העובדה שישנן מברגות חשמליות ואקדחי מסמרים - לא מבטלת את הצורך בידע והניסיון של שימוש במברג ופטיש בגרסאותיהן הבסיסיות , במיוחד אם מעוניינים לפתוח מפעל לייצור מברגות .ידע שכזה נרכש בזיעה ופציעות באצבעות .

בעבודה זו כתבתי מספר שיטות בסיסיות (כמובן בהדרכה והיעזרות במידע נגיש) , שיחקתי איתן קצת לטובת היכרות (בשונה משימוש קודם במתודות כתובות מחבילות מוכרות) .

התעניינתי בלבחון שיטות בסיסיות אלו אל מול שיטות מחבילות קיימות .

בשלב הבא השתמשתי בSCIKIT עבור שיטות מתקדמות יותר שלא הייתי מספיק לדבג בשנה הקרובה אם הייתי כותב אותן בעצמי (על אף שנעשו ניסיונות שעלו בזמן יקר , אך תרמו להבנה).

ולבסוף עברתי לשימוש כמעט מלא בחבילות לצורך הצגה והשוואות בין שיטות שונות ופרמטרים שונים.

הקוד נשלח במחברת ג'ופיטר , בניסיון לחסוך צורך בהרצת הקוד והנוחות שבהרצת תאים ולא קוד שלם , כמעט בכל שלב ישנו תיאור בראש התא בצורת הערה (#) של מה מבוצע בו או מטרתו , ורוב הפלט בעל כותרות מודפסות בין השלבים של הפלט .

1. מסדי נתונים :

א. פינגווינים – דומה לאירוסים (לא בהתפלגות אחידה) – עבור המודלים הבסיסיים , השתמשתי בקובץ השני (האוריגינל) , מצורף בZIP .
https://www.kaggle.com/datasets/samybaladram/palmers-penguin-dataset-extended?select=palmerpenguins_original.csv

ב. ספרות – MNIST HANDWRITTEN DIGITS IN CSV , עבור בדיקת שיפור יעילות וזיהוי בעזרת רשתות נוירונים ושילוב עם שיטות שלמדנו.
https://www.kaggle.com/datasets/oddrationale/mnist-in-csv

מסד נתונים	סוג נתונים	דגימות	פיצ'רים
פינגווינים קלסיפיקציית שייכות ל3 מינים ע"פ מדדים פיזיים . וחיזוי ל2 ע"י שיטות לינאריות לצורך תרגול.	מספריים – אורך ורוחב של כנף ומקור , משקל. מילוליים - שמות זנים , איים ומין	344 , סוננו ל334 בעקבות ערכים חסרים.	6 לכל דגימה , בשלב מסוים סוננו חלק לבדיקה.
ספרות קלסיפיקציה של "תמונה" לספרה הכתובה בה .	מספריים - ייצוג במערך של תמונות 28*28 , שחור לבן , כאשר כל שורה היא מערך שכזה ששוטח.	70,000 (מחולקים לאימון ומבחן)	784 , כל עמודה היא גוון הפיקסל 0-255

2. שיטות : מתוך רצון להבין טוב יותר את הבסיס , כתבתי מספר שיטות בסיסיות ושיחקתי איתן אל מול נתונים מומצאים (BLOBS וכדומה) , וכן בחרתי במסד הנתונים של הפינגווינים להשוואה בין השיטות הבסיסיות יותר . לאחר מכן הוספתי מסד נתונים MNIST של תמונות לצורך ניסיון בזיהוי תמונה ובדיקת הגבולות והיתרונות של רשתות נוירונים אל מול אלגוריתמים אחרים שחלקם נכשלים לחלוטין במטלה .

1) KNN - מבוסס מרחק אוקלידי בין נקודות , בוחר בהצבעת רוב מהלייבלים הקרובים ביותר ומשייך לדגימה החדשה.
2) KMEANS - איטרטיבי , מנסה לשייך נתונים למספר האשכולות שקיבל , ע"ב ממוצע האשכול , מזיז בכל חזרה את מרכזי האשכולות ל נקודה שהינה ממוצע הערכים באשכול ומסווג מחדש את האשכולות עד אשר אין שינוי .
3) Linear SVM - מנסה למצוא מפריד לינארי שמרחקו מ2 הקבוצות המתקבלות הוא מקסימלי , מפריד קשיח (בהנחה שניתן לפצל את הדגימות) .
4) SVM with KERNEL TRICK – לא לינארי , משתמש בהעלאת מימדים בעזרת שינוי הנתונים תחת קרנל לבחירתנו – פולינומי , גאוס וכו' , מסוגל ללמוד היפותזות מורכבות יותר אך דורש שימוש מפריד רך , עבורו ננסה להקטין את כמות השגיאות אל מול הגדלת גודל המפריד .
5) PERCEPTRON – מודל בסיסי , עם פונקציית הפעלה מסוג מדרגה , עובד כמו מפריד לינארי במספר ממדים בהתאם לכמות המשקולות וההטיות שנבחרות עבורו . לא יציג יתרון גם אל מול אינסוף העתקים של עצמו באינסוף שכבות שכן בגלל פונקציית המדרגה נקבל קשרים לינארים בכל השכבות . במקרה שלנו משתמש בפונקציית עדכון שמעלה את ערכה של משקולת והטיה עבור הכניסות שלא היו מתחת לסף המדרגה .
6) PCA – שיטת לטרנספורמציה של המידע , מ מימד גבוה לנמוך יותר ע"י פירוק לערכים סינגולריים (SVD) סידורם לפי חשיבות (כמות המידע שיש בהם) והשארת מספר הערכים המבוקש . יעיל גם לצורך ייצוג מידע שכזה ב2-3 ממדים .
7) ANN – רשת של נוירונים ,מסודרת בשכבות , עם פונקציות הפעלה שונות בין השכבות אשר שומרות על המידע בת"ל , העלאת ממדים של הנתונים ויכולת ללמוד היפותזות מסובכות . במקרה של מודל "מתקדם" יותר (RNN), משתמשת בפעפוע לאחור ופונקציית הפסד בשילוב עם גרסאות שונות של גרדיאנט דיסנד (GD/SGD) – חיפוש כיוון השינוי המקסימלי ושינוי משקלים והטיות של השכבות השונות על ידי נגזרות חלקיות של המידע שהתקבל בקצה (התחזית).
8) Random Forest – מבוסס עצי החלטה , יוצר יער של עצי החלטה שונים שחושבו על כלל הפיצ'רים בסדר שונה , בהינתן דגימה חדשה – מחליט ע"פ רוב .
9) Naïve bayes - גאוסייני , מבוסס על חוק בייס , מנסה להתאים את הדגימה ע"י חישוב ההסתברויות הידועות מראש , ובחינה הקטגוריה הנכונה בהסרת התלות בין הפיצ'רים השונים .
10) K-fold cross validation - שיטה לפיצול ובדיקה חוזרת של נתחים שונים מהנתונים בתור אימון ומבחן .

בחירת השיטות נעשתה על בסיס עניין בכתיבתן ,תרגול כללי ורצון לבדוק האם ניתן להתחרות בשיטות החזקות והמודרניות יותר .

בהתבסס על ניתוח הנתונים , ברור ש RNN תצליח בקלות יחסית במשימת החיזוי , אך כפי שהבנתי את המטלה , הרעיון הוא להתנסות בבניה / שינוי של מודלים ו"עלויות" שונות של האימון של המודלים כתוצאה משינוי בפרמטרים , פונקציות "ענישה" וכו' .

3. חלוקת הנתונים :

מתוך עניין , כתבתי מספר גרסאות של פונקציות חלוקה , לפני בחירת מסד הנתונים . כמובן שברצון לייצר כלי חלוקה אפקטיבי עבור מספר סוגי נתונים , נתקלתי בקושי בהכללת הפונקציה ולכן פשוט שיניתי אותן בהתאם לכל אחד ממסדי הנתונים .
למטרת סיווג דגימות ,הפונקציה מקבלת מסד נתונים או כתובת , יחס חלוקה רצוי (0.7 לאימון כברירת מחדל) ו זרע רנדומלי.
לאחר מכן ,מעבירה לתוך מערך NUMPY , בודקת את התפלגות המטרות (TARGETS) , כל עוד אנחנו קרובים להתפלגות אחידה , מבצעת ריצה רנדומלית על דגימות בהתאם לכמות המטרות ומוסיפה לקבוצת המבחן .
את שאר הדגימות מעתיקה לקבוצת האימון . מערבבת את שתיהן, ופולטת 2/3 מערכים בתצורת – מערך וקטורים לאימון, תוצאות לאימון, מערך וקטורים לבדיקה/ולידציה ותוצאות.

א. פינגווינים – התפלגות לא אחידה , בוצעו בדיקות לבדוק אם החלוקה משפיעה על התוצאות . בהתחלה , הפונקציה ביצעה תת-דגימה (UNDERSAMPLING) כך שבמערך האימון יש התפלגות אחידה וכן בקבוצת המבחן , ובוולידציה התקבלו שאר הדגימות . בהמשך נבדקה חלוקה לפי אחוזים בלבד ללא התחשבות בהתפלגות .

ב. ספרות – המאגר הכיל עשרות אלפי דגימות , לצורך ייעול , הפונקציה קיבלה מס' דוגמאות רצוי וחילקה לשני מערכי אימון ותרגול בגודל דומה בהנחה שניתן להשתמש בשאר הדגימות או במערך הנוסף בקובץ לצורך ולידציה .

4. פונקציות סיכון והערכת ביצועים :

א. ACCURACY – מכיוון שמדובר בבעיית תיוג של תצפיות בהתפלגות דומה ,אין מחיר גבוהה לשגיאות בכיוון מסוים , בחרתי להתבסס על כמות ה"קליעות" לעומת "זריקות לסל" , מדד בסיסי שנועד לבדוק האם השיטות שכתבתי עובדות בכלל ואם כן – לקבל קנה מידה ליעילותן עבור מסדי נתונים שונים . גרסה שלי מופיעה בקוד כ "accTest" .

ב. מטריצת "בלבול" – Confusion matrix – מחבילת SCIKIT , מראה תיוגים אל מול האפשרויות בצורת טבלה , מאפשרת לראות בעין את הטעויות שהתרחשו וכך לזהות הטיות של השיטה לאחר האימון לטעות בין תיוגים מסוימים . (לדוגמה לבלבל בין שתי ספרות דומות בגלל חוסר איזון בנתוני האימון)

ג. דו"ח קלסיפיקציה – classification report – מאותה חבילה , נותן את כל המדדים העיקריים להערכת ביצועים :

1) ריקול – כמות התיוגים הנכונים חלקי מספר הדוגמאות בעלות התיוג הנ"ל עבור כל תיוג אפשרי .
2) F1 – מדד משולב של דיוק וריקול , ממצע את ערכיהם יחד , ועוזר למציאת איזון בין שני המדדים במקביל .
3) צפייה במדד ההפסד שלי מתעדכן עם כל חזרה על מחזור אימון 😊

ד. סיכון : ברשת הנוירונים השתמשתי במזעור לוג שלילי , שכן היא מתאימה למשימת תיוג דגימות ונבחרה כבת הזוג של הסופטמקס ביציאה מהרשת , הרי המטרה היא להגדיל את ההערכה להסתברויות הרצויות .

5. אובר פיט :

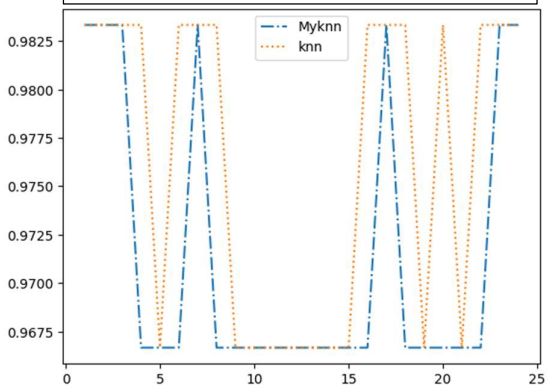
התקבל במהלך אימון על הפינגווינים , כאשר הנתונים בקבוצת המבחן והאימון הראו התפלגות קיצונית מאוד (מס ערכים בודדים של זן פינגווין 1 והמון מהשאר) , בהערכה , המודל לא ידע לתייג את הפינגווינים הללו וניסה לתייגם בקבוצות האחרות .
נפתר על ידי חלוקה יותר מאוזן ששומרת על התפלגות יותר מאוזנן בקבוצת האימון.

פינגווינים :

פעולות שבוצעו על הנתונים:

א. ניתוח הנתונים , הסרת ערכים חסרים , תרגום ערכים לא מספריים למספריים .
ב. פיצול לקבוצות אימון מבחן ו ולידציה .
ג. בדיקת מטריצת קו וריאנס , לבחירת הנתונים היעילים ביותר לסיווג , עמודת המין ייצרה בעיה , הוחלט להשוות עם ובלי וביצוע נרמול לנתונים עבור KNN.

ניסוי 1 :דיוק מול מספר שכנים



ניסויים :

1. בדיקת דיוק ומהירות ריצה – KNN שלי מול חבילה.
הרצת 24 איטרציות אימון שונות
כתלות במספר שכנים נבחר (1-25).
השוואת זמני אימון ACCו על קבוצת מבחן .
השוואת 3 שכנים , על קבוצת ולידציה .

תוצאות : האלגוריתם שלי נחות משמעותית (ביצועית).

בבדיקה מול סט ולידציה שניהם זהים – 90% דיוק.

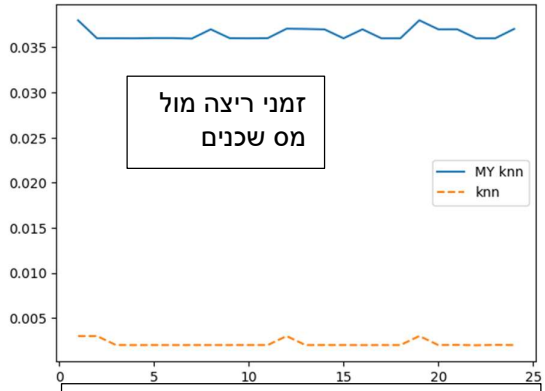
תוצאה הגיונית .

עלולה לנבוע משימוש שלי במתודה פחות יעילה.

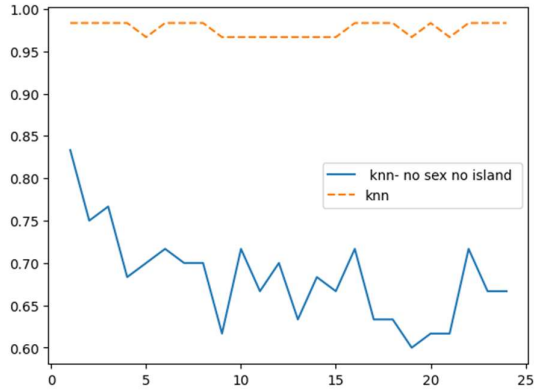
לאחר נרמול – שיפור משמעותי של השיטה שלי לעומת

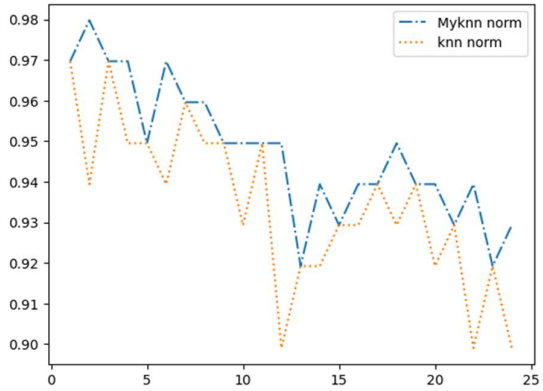
השיטה מהספרייה , אך בעלות בזמן מטורפת (פי 33).

2. הורדת עמודות "מין" ו"אי" ובדיקת ביצועי KNN רק על בסיס נתוני מדידות : משקל , אורך/גובה מקור , אורך כנף.
תוצאות : ירידה משמעותית ביכולת החיזוי .
בבדיקה מול סט ולידציה – 69% דיוק.
שיפור : ע"י נרמול הנתונים .



ניסוי 2 : ללא עמודות "מין" ו "אי"





ניסוי 1 – שיפור ע"י נרמול נתונים .

3. KMEANS (שלי) לצורך תיוג פינגווינים בלי עמודות מין ואי:
תוצאה: כושל לחלוטין – 60% ~ דיוק , לא מתאים לנתונים .
שיפור : ע"י נרמול הנתונים - 80% .

4. השוואה בין מספר שיטות (מהספרייה)
בעזרת KFOLD קרוס ולידציה – עבור דאטה עם תת-דגימה
ועבור חלוקות שונות לקבוצות מבחן ואימון.
תוצאות למטה.
ניתן לראות כי כולם השתפרו במעבר לדגימות אקראיות עבור
נתוני האימון .

uniform split	percent split	sklearn split
KNN:, 0.668132, (0.111071)	KNN:, 0.821277, (0.045831)	KNN:, 0.790217, (0.066250)
NB:, 0.954396, (0.050719)	NB:, 0.957447, (0.013457)	NB:, 0.974275, (0.028630)
SVM:, 0.508242, (0.053001)	SVM:, 0.765957, (0.038061)	SVM:, 0.734783, (0.102050)
rfc:, 0.984615, (0.030769)	rfc:, 0.987234, (0.017021)	rfc:, 0.965761, (0.025316)

5. השוואת שיטות בעזרת תוצאות מחיזוי קבוצת המבחן :

שיטה	ACCURACY	לאחר נרמול
RF	0.99	פוגע בביצועים
gNB	0.97	פוגע בביצועים
KNN	0.69	0.97
KMEANS	0.60	0.80
Svm	0.65	0.80
mlp	מזעזע	0.95

כפי שניתן לראות עבור נתונים מועטים (יחסית) מהסוג הנתון לנו השיטות היעילות ביותר היו עצי החלטה וסיווג בייסיאני , אך גם רשת נוירונים עם 3 נוירוני פלט הצליחה לא רע , כמו כן גם שיטה בסיסית כמו KNN עובדת יותר טוב .

<u>ספרות MNIST: - במחברת קולאב (אומן על 4T GPU של גוגל)</u>

עבור מסד הנתונים הזה , התנסיתי בלנסות ולשפר דיוק ע"י שינוי פרמטרים ברשתות נוירונים בהשוואה לרשת הקונבולוציה מהתרגול (ACC 98.8%) , והרשת הרגילה (ACC 90.0%) .

הנתונים הגיעו מסודרים ומחולקים לקבוצות ניסוי – מבחן .הקוד ברובו מהתרגול .

התנסיתי בלעלות על רמת הדיוק שהוצגה בתרגול בעזרת שינוי מבנה הרשת ושינוי פרמטרים.

בסופו של דבר , ע"י שינוי הרשת לרשת פשוטה יותר – כניסה – כגודל הקלט (784) אמצע 374 ויציאה של 10 עם ReLU בכניסה וסופט מקס ביציאה , וקביעת 5 סבבי אימון וקצב למידה 0.072

הגעתי לתוצאה של 0.9668% על קבוצת הבדיקה . תוצאה זו מעניינת כי השיפור ב זמן האימון לעומת אימון רשת קונבולוציה שעולה עליה רק ב2.2% דיוק (כאשר בהרצות אחרות גם הגיע לאותה רמת דיוק) , היא 20% ממשך האימון הנדרש , תוצאה שכזו משמעותית בהינתן מטלה שאין בה מחיר כבד לטעות , כמו במשימה הנוכחית .

אין סיבה לסבך עבור מטלה פשוטה , במיוחד לא כשניתן לעשות כיוונונים ולהוסיף עוד שיטות כחלק מהתהליך עבור משימות מסובכות יותר .