**מבוא לבינה מלאכותית – רטוב 3**

**חלק ב':**

שאלה 3 חלק 2:

כשאנחנו משווים אלגוריתמים ושיטות עבודה, אנחנו רוצים שכמה שיותר מההבדל במדידות הביצועים יבוא מהאלגוריתמים עצמם, ולא ממקורות חיצוניים, כגון הבדל בחילוק הקבוצות אימון/בחינה.

שאלה 5 חלק 2:

שאלה 5 חלק 3:

ביצועים אופטימליים התקבלו ב

שאלה 5 חלק 4:

מקסימום מתקבל ב ומינימום ב ניתן להסביר זאת עם Overfitting. ב יש יותר מידי רעש, כי כל דוגמה, גם הרועשות קובעות את הסיווג באזור מסוים. ניתן גם לראות מגמה של ירידה בדיוק ככל ש K גדל, שזה בגלל שאנחנו מתייחסים לשכנים רחוקים שאין להם באמת שום דימוי לקלט הנבדק.

שאלה 7 חלק 2:

מבין שתי הניסויים שהרצנו בשאלה זו, האלגוריתם של ID3 הניב תוצאות מדויקות יותר, אבל גם הוא היה פחות מוצלח מה 5-NN שהוא היה הגרסה המדויקת של KNN לפי הניסוי בשאלה 5.

**חלק ג**

כדי למצוא את המסווג הטוב ביותר ערכנו סדרת ניסויים שכללו הרצות רבות על ה dataset כדי לראות אילו פרמטרים מניבים את התוצאה הטובה ביותר.

הפרמטרים שנבדקו הם צמצום דוגמאות, מספר הקבוצות לחלוקה ומספר השכנים.

עבור dataset מלא שלא צומצם כלל ועבור מספר קבוצות לחלוקה, הרצנו את כל הפרמוטציות המחלקות את ה dataset לקבוצות אימון ומבחן.

עליה של פרמטר מספר השכנים יתבצע עד וכאשר זיהינו שהגענו למקסימום, כלומר הלולאה הופסקה כאשר זוהתה ירידה בדיוק.

אילו הם הממצאים(עבור דיוק, כך ששורות הם מספר השכנים ועמודות הם מספר החלוקה לקבוצות):

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 |  |
| 0.125 | 0.12 | 0.119 | 0.119 | 0.12 | 0.12 | 0.119 | 0.119 | 1 |
| 0.964 | 0.944 | 0.957 | 0.934 | 0.939 | 0.92 | 0.945 | 0.935 | 3 |
| 0.954 | 0.952 | 0.957 | 0.952 | 0.97 | 0.94 | 0.963 | 0.945 | 5 |
|  | 0.96 | 0.958 | 0.952 | 0.95 | 0.944 | 0.96 | 0.949 | 7 |
|  | 0.952 | 0.958 | 0.957 |  | 0.952 |  | 0.953 | 9 |
|  |  | 0.957 | 0.975 |  | 0.952 |  | 0.9599 | 11 |
|  |  |  | 0.957 |  | 0.956 |  | 0.953 | 13 |
|  |  |  |  |  | 0.956 |  |  | 15 |
|  |  |  |  |  | 0.952 |  |  | 17 |

הדיוק הטוב יותר שנמדד עבור ניסוי זה הוא עבור 6 קבוצות ו11 שכנים 0.975.

ניסוי דומה הורץ הפעם עם אלגוריתם לצמצום דוגמאות כפי שנלמד בכיתה:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 |  |
| 0.91 | 0.12 | 0.119 | 0.119 | 0.12 | 0.12 | 0.119 | 0.119 | 1 |
| 0.954 | 0.944 | 0.923 | 0.904 | 0.90 | 0.892 | 0.877 | 0.871 | 3 |
| 0.963 | 0.92 | 0.937 | 0.928 | 0.944 | 0.92 | 0.913 | 0.873 | 5 |
| 0.954 |  | 0.93 | 0.946 | 0.925 | 0.924 | 0.946 | 0.911 | 7 |
|  |  |  | 0.928 |  | 0.908 | 0.922 | 0.9 | 9 |

הדיוק הטוב יותר שנמדד עבור ניסוי זה הוא עבור 9 קבוצות ו5 שכנים 0.963.

לאחר פענוח הממצאים החלטנו לבחור כמסווג knn ללא צמצום דוגמאות עם פרמטר חלוקה של 6 ומספר שכנים 11.

**הסבר על הקבצים:**

Aux\_functions.py – קובץ עם פונקציות עזר: מרחק אוקלידי ונרמול וקטור.

Knn\_classifier.py – מחלקה היורשת מהמחלקה אבסטרקטית לפי דרישת התרגיל.

Knn\_factory.py – מחלקה היורשת מהמחלקה האבסטרקטית לפי דרישת התרגיל.

Validation.py – קובץ עם פונקציות עזר נוספות הקשורות לחלוקה לקבוצות, כתיבה לקבצים, ו evaluate.

Id3Classifier.py – קובץ המכיל מסווג של ID3 אשר יורשת גם מהמחלקה האבסטרקטית abstract\_classifer\_factory כדי שיוכל להתאים כפרמטר לפונקצייה evluate.

PerceptronClassifer.py - קובץ המכיל מסווג של Perceptron אשר יורשת גם מהמחלקה האבסטרקטית abstract\_classifer\_factory כדי שיוכל להתאים כפרמטר לפונקצייה evluate.

Competition\_factory.py – מחלקה של המסווג איתו בדקנו את קבוצת המבחן.