

Haberman's Survival Data Set

פרויקט גמר – למידת מכונה

מגישים: אורי חנונוב – 204558399, אולגה מאזו - 314106766

תיאור המאגר:

מאגר הנתונים כולל מקרים ממחקר שנערך בין השנים 1958 - 1970 בבית החולים בילינגס באוניברסיטת שיקגו על הישרדותם של חולים שעברו ניתוח לסרטן השד, המאגר נכתב בשנת 1999.

המאגר מורכב מ-3 מאפיינים: גיל המטופל בזמן הניתוח, שנת הניתוח, מס' תאים חיוביים לסרטן השד.

בנוסף, המאגר מכיל מאפיין סיווג (סטטוס הישרדות) – האם המטופל ישרוד 5 שנים הקרובות או לא. מאפיין זה מקבל את הערך 1/2 בהתאם לשלושת המאפיינים הקודמים.

1 – המטופל ישרוד 5 שנים או יותר

2 – המטופל לא ישרוד עד 5 שנים

נתונים כלליים על המאמר:

גיל החולים משתנה בין 30 ל- 83 עם חציון 52.

למרות שהמספר המרבי של בלוטות הלימפה החיוביות שנצפו הוא 52, כמעט 75% מהמטופלים סובלים מפחות מ-5 בלוטות לימפה חיוביות וכמעט 25% מהמטופלים אינם בעלי בלוטות לימפה חיוביות

מערך הנתונים מכיל 306 רשומות.

נתונים אלו התקבלו ע"י הפקודה - `print(Haberman.describe())`:

	age	year_of_treatment	positive_lymph_nodes	survival_status_after_5_years
count	306.000000	306.000000	306.000000	306.000000
mean	52.457516	62.852941	4.026144	1.264706
std	10.803452	3.249405	7.189654	0.441899
min	30.000000	58.000000	0.000000	1.000000
25%	44.000000	60.000000	0.000000	1.000000
50%	52.000000	63.000000	1.000000	1.000000
75%	60.750000	65.750000	4.000000	2.000000
max	83.000000	69.000000	52.000000	2.000000

תיאור הפרויקט:

חילקנו את נתוני המאגר ל-2 חלקים: X ו- Y , כאשר X מכיל את שלושת המאפיינים ו- Y מכיל את מאפיין הסיווג. את חלקים אלו חילקנו ל-2 קבוצות: קבוצת אימון וקבוצת טסט (כאשר קבוצת האימון מכילה אחוז מסוים שאנו בוחרים מתוך המאגר). חלוקה זו מתבצעת ע"י `train_test_split` כאשר `test_size` מייצג את גודל קבוצת הטסט (30% מכלל המאגר במקרה שלנו).

את קבוצת האימון אנו שולחים כל פעם לטכניקה אחרת – מקבלים את המכונה המאומנת ואיתה בודקים את קבוצת הטסט ובהתאם מפיקים את המסקנות ואחוזי ההצלחה של המכונה.

שלחנו את הנתונים לכל טכניקה 300 פעמים וחישבנו את הממוצע שמתקבל (כלומר את אחוז ההצלחה הממוצע של המכונה). כדי לעשות את זה יצרנו מערך שמכיל את כל הטכניקות ואז בדקנו כל טכניקה 300 פעמים עם קבוצת אימון רנדומלית אחרת וחישבנו את ממוצע ההצלחה שמתקבל ממכונה זו.

במהלך כתיבת התוכנית השתמשנו כמה פעמים בפונקציה `predict` כדי לבדוק שהסיווג שמתקבל ע"י המכונה תואם לסיווג שאמור היה להתקבל (לפי מאגר הנתונים).

המטרה המרכזית של המכונות היא לחזות על פי שלושת המאפיינים של האדם האם הוא ישרוד יותר או פחות מ-5 שנים לאחר שיעבור את הניתוח.

הטכניקות שהשתמשנו בהן ללמידת המכונה:

1. **Ada Boost** - כאשר כל פעם הוא עושה 50 סיבובים של הטכניקה, מגדיל את משקלי הנקודות שהיו טעות ובכך גורם בסיבוב הבא להסתברות גבוהה יותר שנקודות אלו יבחרו.

2. **SVM** – יש כמה סוגי `kernel` שניתן לשלוח למכונה, כל סוג חותך את הנתונים בדרך אחרת. אנחנו שלחנו את: `linear`, `poly`, `rbf` and `sigmoid`.

3. **Decision Tree** - המטרה היא ליצור מודל שמנבא את ערכו של משתנה יעד על ידי למידת כללי החלטה פשוטים הנגזרים מתכונות הנתונים.

4. **KNN** – חיפוש שכך קרוב, קבלת הסיווג לפי השכנים הקרובים. שלחנו כל פעם 1 עד 9 שכנים (אי זוגיים) כדי לראות אם התוצאה משתפרת ובנוסף שלחנו את זה עם שני סוגי מרחקים שונים: `l2 = Euclidean distance`, `l1 = Manhattan distance`.

אתגרים שנתקלנו בהם:

*כשהתחלנו לעבוד על מאגר הנתונים הבנו שאנחנו צריכים לחלק את הנתונים ל-2 חלקים: X ו- Y כך ש- X יכיל את המאפיינים (הפיצ'רים) ו- Y יכיל את מאפיין הסיווג, ולאחר מכן להשתמש בספריות של `sklearn` שבעזרתן אפשר להשתמש בכל הטכניקות שציינו ולעבד את התוצאות.

הקושי שלנו היה בעיקר להבין איך לבצע את החלוקה כמו שצריך כדי שהכל יעבוד. ישבנו על זה די הרבה זמן, קראנו באינטרנט וניסינו, עד שהחלוקה הצליחה. ברגע שהבנו איך עושים את החלוקה ואיך להתעסק איתה בספריות העבודה על מאגר הנתונים התחילה להיות ברורה יותר.

בנוסף, הבנת הספרייה `sklearn` ושימוש נכון בה ובפונקציות שלה גם כן היה מאתגר מכיוון שזו ספרייה עם הרבה אפשרויות שאנחנו לא הכי מכירים.

תוצאות:

```
AdaBoost for train: 81.4018691588785
AdaBoost for test: 72.61956521739131
SVMLLinear for train: 74.43457943925233
SVMLLinear for test: 72.7173913043478
SVMPoly for train: 74.7570093457944
SVMPoly for test: 72.93478260869561
SVMRbf for train: 73.98598130841121
SVMRbf for test: 73.08695652173913
SVMSigmoid for train: 73.39252336448595
SVMSigmoid for test: 73.84782608695649
DecisionTree for train: 98.58411214953274
DecisionTree for test: 65.69565217391302
```

```
KNN1ManhattanDistance for train: 98.58878504672901
KNN1ManhattanDistance for test: 66.78260869565216
KNN3ManhattanDistance for train: 83.07476635514018
KNN3ManhattanDistance for test: 68.91304347826085
KNN5ManhattanDistance for train: 79.82710280373833
KNN5ManhattanDistance for test: 72.13043478260866
KNN7ManhattanDistance for train: 78.35514018691592
KNN7ManhattanDistance for test: 73.3369565217391
KNN9ManhattanDistance for train: 77.67289719626169
KNN9ManhattanDistance for test: 74.14130434782606
KNN1EuclideanDistance for train: 98.63084112149532
KNN1EuclideanDistance for test: 68.39130434782605
KNN3EuclideanDistance for train: 84.07009345794395
KNN3EuclideanDistance for test: 70.45652173913041
KNN5EuclideanDistance for train: 79.54672897196257
KNN5EuclideanDistance for test: 71.40217391304344
KNN7EuclideanDistance for train: 77.92990654205607
KNN7EuclideanDistance for test: 72.79347826086952
KNN9EuclideanDistance for train: 77.96261682242991
KNN9EuclideanDistance for test: 74.1630434782609
```

בתוצאות אלו קבוצת האימון הייתה 70% מכלל המאגר (חלוקה זו מתבצעת ע"י הפונקציה `train_test_split` שלקוחה מתוך הספרייה `sklearn`). התוצאות מראות את אחוזי ההצלחה על קבוצות האימון וקבוצות הטסט (בכמה מקרים מתוך הקבוצות הצלחנו לתת סיווג נכון – האם הבן אדם ישרוד או לא).

מסקנות:

- ניתן לראות שמספר השכנים בטכניקת ה-KNN משפיע על אחוזי ההצלחה של המכונה. ככל שמגדילים את מספר השכנים במכונה, אחוזי ההצלחה עולים. כלומר אנשים שדומים אחד לשני (מבחינת המאפיינים) יקבלו את אותו סיווג – האם ישרדו או לא ישרדו לאחר 5 שנים. כאשר $k=9$ תוצאות המכונה היו טובות ביותר וסוג המרחק המחושב אוקלידי או מנהטן לא בהכרח השפיע על התוצאה, התוצאות די דומות מבחינת שינוי המרחקים.

- מכמה הרצות של התוכנית ראינו כי אין טכניקה שיותר או פחות טובה **משמעותית** משאר הטכניקות האחרות. רוב הטכניקות מצביעות על 70% הצלחה (כאשר קבוצת האימון 70%), אם כי ניתן לראות שחיפוש שכן קרוב כאשר $k=9$ היא הטכניקה עם הכי הרבה אחוזי הצלחה על הטסט. זה קורה כנראה מכיוון שאנחנו בוחנים שם אנשים שמאפייניהם (X) די דומים אחד לשני ולכן גם הסיכוי שלהם לשרוד יהיה אותו הדבר.

- לדעתנו, סיבות אפשריות שאין טכניקה טובה ביותר: המאגר יחסית קטן (מכיל 306 נתונים) ומספר המאפיינים שהמכונה לומדת מהם הוא 3. (אם היו יותר מאפיינים היו למכונה יותר אפשרויות ללמוד לסווג).

- בדקנו מה אחוזי ההצלחה על קבוצת האימון מול אחוזי ההצלחה על קבוצת הטסט, כלומר בדקנו האם יש `overfitting`. ניתן לראות כי התוצאות על מדגם האימון הרבה יותר טובות מהתוצאות על מדגם הטסט, אחוז השגיאה באימון הרבה יותר נמוך במכונות הבאות:

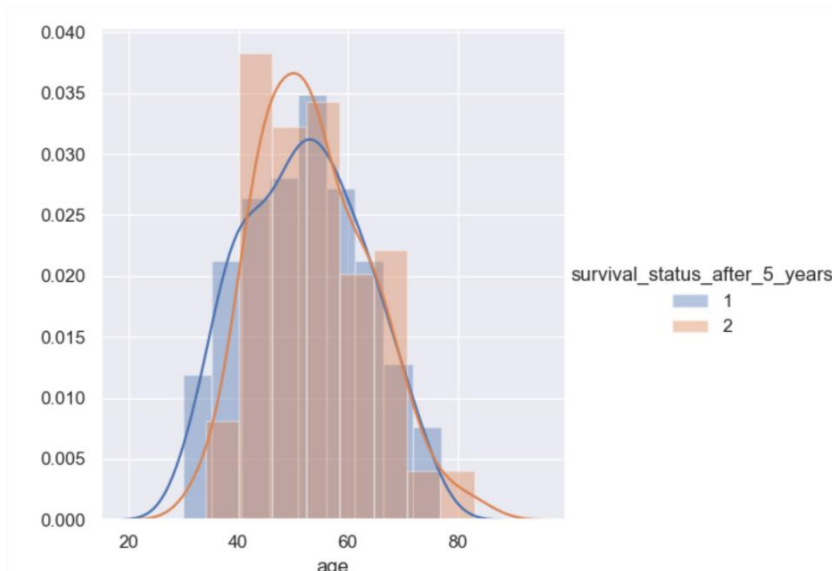
Adaboost, Decision Tree, KNN(with $k = 1,3,5$)

- מבחינת SVMs הם כולם בערך יצאו 72/73 אחוזי הצלחה, כלומר אין השפעה גבוהה יותר מדי לסוג ה `kernel` שבורחים.

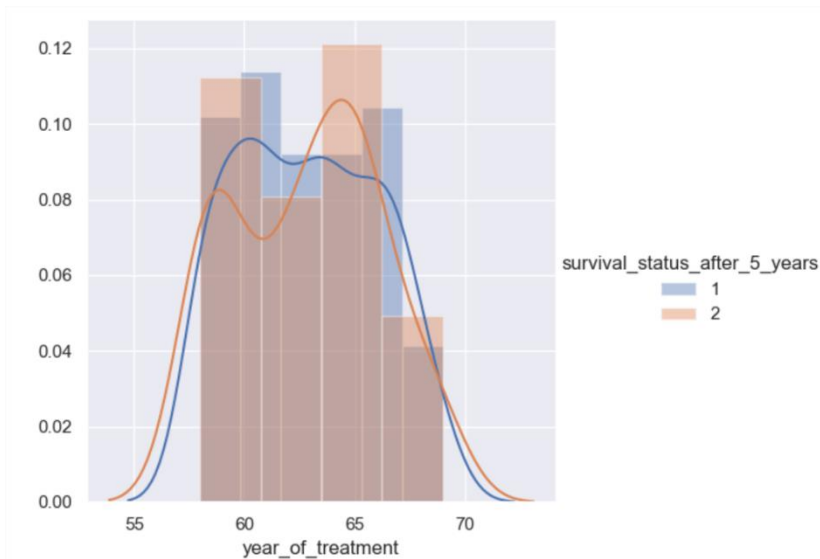
*הגדלנו את קבוצת האימון להיות 90% ממאגר הנתונים (במקום 70%) וראינו שאחוזי ההצלחה נשארים בערך אותו הדבר (עומדים על בערך 70% הצלחה).

דיאגרמות (אנליזות):

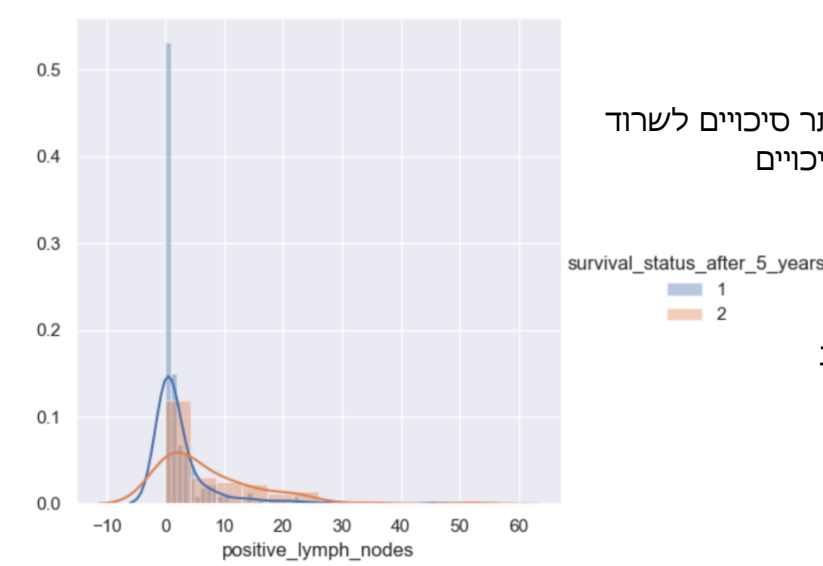
*דיאגרמה זו מציגה את גיל המטופל מול סיכויי ההישרדות שלו



*דיאגרמה זו מציגה את שנת הניתוח של המטופל מול סיכויי ההישרדות שלו



*ניתן לראות שעד בערך 4 תאים יש לאדם יותר סיכויים לשרוד מאשר לא, ואם מדובר על יותר מ-4 תאים הסיכויים שלא ישרוד גבוהים יותר.



*ניתן לראות כי למאפיינים גיל המטופל ושנת הניתוח יש פחות השפעה על אחוזי ההישרדות של האדם ולמספר התאים הסרטניים יש השפעה הרבה יותר גדולה.