**הרצאה 5**

Adaboost במקרים של בעיות רגרסיה

האלגוריתם הרגיל של adaboost לסיווג

Text, letter

Description automatically generated

מחשבים עבור כל תצפית המשקל שלה חלקי המשקל של שאר התצפיות כדי לדעת מה ההסתברות לקבל את תצפית זו.

האלגוריתם לבעיות רגרסיה

Text

Description automatically generated

D-הטעות המקסמילית של הרגרסור

עבור כל תצפית מחשבים את השגיאה עבור תצפית זו ומחלקים בD על מנת לנרמל

ואז מחשבים ממוצע של השגיאות עבור כל הנקודות-

ככל שהשגיאה יותר קטנה נקבל קטנה יותר

נעדכן את המשקל של התצפיות באמצעות חישוב

*A picture containing text, clipart

Description automatically generated*

*זאת אומרת שככל שהשגיאה יותר גדולה המשקל יורד פחות*

*המשקל של כל מסווג הוא לוג של 1 חלק הבטא של מסווג זה*

*Diagram, schematic

Description automatically generated*

*ככל שהרגרסור יותר טוב- המשקל שלו יותר גבוה*

*Text

Description automatically generated*

*כיוון שממוצע מאוד מושפע מערכים קיצוניים שרחקוים מהממוצע, מחשבים במקום חציון ממושקל על מנת להחליט מיהו הרגרסור הטוב יותר*

*לוקחלText

Description automatically generated*

*לוקחים את כל הרגרסורים ומסדרים אותם מהקטן לגדול על פי משקלים*

*ואז הרגרסור החציון הוא הרגרסור שמשקלו הוא חצי מסכום המשקלים של כל הרגרסורים*

*Text

Description automatically generated*

לאחר מכן באמצעות הרגרסור שנבחר, מחשבים את ההסתברות לקבל כל תצפית, מחשבים את הממוצע וסטיית התקן לכל התצפיות.

**Clustering:**

בקלאסטרינג אין data מסווג, מסווגים את הנתונים לקבוצות כאשר בכל קבוצה הרשומות הן בעלות ערכים דומים. לדוגמה אנשים בסופר שקונים דברים דומים.

משמש בדרך כלל לניתוח נתונים ראשוני

**Graphical user interface, text, application

Description automatically generated**

**Diagram

Description automatically generated**

בצד שמאל אנחנו רואים נתונים של כוכבים, קיבלנו 3 קבוצות: ענקים אדומים, רגילים וגמדים לבנים.

לאחר שחילקנו לקבוצות, ניתן לנתח כל קבוצה בנפרד, לנתח מה המשמעות הפיזקלית של כל קבוצה.

בדוגמה בצד ימין ניתן לראות את הגובה והמשקל של נשים בצבא האמריקאי. מדדו לגבי נשים מהצבא האמריקאי כל מיני נתונים, באמצעות K-means ביצעו קלאסטרינג. לכל קלאסטר יצרו סוג של מדים אשר מתאים לקלאסטר זה.

**K Means:**

Graphical user interface, text, application

Description automatically generated

K מסמן את מספר הקלסטרים שאנו רוצים רוצים לקבל

האלגוריתם עובד באמצעות בחירה רנדומלית של K נקודות

לאחר מכן משייכים כל אחת מנקודות הנתונים לאחד מK הקלאסטרים, באמצעות חישוב המרחק לכל מרכז קלאסטר ושיוך לקלאסטר עם המרחק הכי קטן למרכזו.

ואז מחשבים את המרכז של כל קלאסטר באמצעות ממוצע הערכים של כל קלאסטר

מחשבים שוב עבור כל נקודה את המרחק שלהם ממרכז של כל קלאסטר ואז אם יש מרחק קצר יותר בקלאסטר אחר אז מעבירים את הנקודה לקלאסטר זה.

חוזרים על תהליך זה עד שנקודות מפסיקות לזוז בין קלאסטרים.

Diagram

Description automatically generated

A picture containing logo

Description automatically generated

-מרכז הקלאסטר

Argmin זה חישוב המרחקים של נקודה ממרכז הקלסטרים השונים ומחזיר את המרכז במרחק המינימלי

- מרכז הקלאטסר החדש הקרוב ביותר

לדוגמה אם רוצים לדעת מה הערך המינמלי במערך אז משתמשים בargmin כדי לדעת מה האינדקס של ערך זה.

*Text

Description automatically generated*

*לכל נקודה פחות מרכז הקלאסטר שלה- חישוב המרחק של הנקודה ממרכז הקלאסטר שלה.*

*מבצעים זאת עבור כל נקודה וסוכמים. הסכום הוא הציון של K-means זה.*

*לכן ככל שהציון יהיה נמוך יותר הk-means ביצע קלאסטרים טובים יותר.*

*ככל שנגדיל את K הציון ירד, השאלה כמה משמעותית הירידה.*

*מנסים K שונים ומסתכלים על הציונים, בוחרים את הK עם הציון הכי טוב, שאחריו הירידה בK כבר מאוד קטנה.*

*על מנת למצוא את המרחק המינימלי בין נקודות מחשבים את הסכום של המרחקים ביניהם, גוזרים את המשוואה שיצאה ומשווים ל0. כך נקבל את סכום הערכים של כל הנקודות חלקי מספר הנקודות-ממוצע. לכן מרכז כל קלאסטר הוא הממוצע של המרחקים בין הנקודות בקלאסטר*

*Text

Description automatically generated with low confidence*

*עבור כל נקודה יש K אפשרויות של קלאסטרים להיות בהן, לכן עבור כל הנקודות יש אפשרויות.*

*הפתרון האופטימלי הוא לחשב את הציון עבור על אפשרויות אלו, אך זו סיבוכיות זמן מאוד גבוהה, לכן משתמשים בK-means כדי לצמצם את מספר האפשרויות.*

Graphical user interface, text, application, chat or text message

Description automatically generated

*נקודות הן לא תמיד מספרים, לכן צריך למצוא דרך לתרגם את הנתונים הלא מספריים, כך שנקודות בעלות ערכים דומים יהיו קרובות.*

*ניתן לתת משקל שונה לערכים שונים.*

*נתונים קטגוריליים- כמו סוגי גלידות, ניתן להגיד האם זו אותו גלידה או לא, אין מרחק יחסי, המרחק הוא 0 או 1. דרך אחת להתמודד עם זה היא ליצור מערך באורך מספר האפשרויות הקיימות, כאשר כל תא מייצג ערך, אם הנקודה בעלת ערך זו אז יש בתא זה 1, בכל השאר 0. כך המרחק יהיה 0 או 1.*

*משתנים ברמות- יש סדר בין האופציות.*

*מספרים כאשר ה0 לא מוגדר טוב- לדוגמה בטמפרטורות.*

*0 כאשר יש לו משמעות מיוחדת, כמו שהערכים מתחילים ממנו- גיל, משקל, אורך*

*פרהיינט וצלסיוס- אין ל0 משמעות מיוחדת, אין משמעות מיוחדת למרחק מ0*

*לכן בדרך כלל מסתכלים על אינטרוולים של טמפרטורות(טווח).*

*חשוב להבין מה זה דמיון בנתונים שאנחנו בודקים, ועל פי זה להתאים את פונקציית המרחק שאנו בודקים, על מנת לבדוק את הדמיון בין הנקודות.*

***מרחק בין 2 אובייקטים חייב לקיים את התנאים הבאים:***

***Text, letter

Description automatically generated***

* *המרחק בין 2 אובייקטים הוא 0 אם ורק אם 2 הנקודות זהות*
* *מתקיים אי שיוויון המשולש*
* *המרחק גדול או שווה ל0*
* *המרחק בין x לy שווה למרחק בין y לx*

***מרחק מנהטן:***

***Text, letter

Description automatically generated***

במנהטן הרחובות הם ממזרח למערב על פי מספרים, והשדרות הן מצפון לדרום, לכן המרחק שלך מנקודה אחת לאחרת הוא ההפרש בין מספר השדרות למספר הרחובות, בערך מוחלט.

מרחק אוקלידי בין 2 נקודות הוא אלכסון בין 2 נקודות, אך בעיר כמו מנהטן נדרש ללכת ממזרח למערב ורק אז מצפון לדרום לדוגמה.

אפשר להגדיר כל P שרוצים

Text, letter

Description automatically generated

ככל שהP הולך וגדל ההשפעה של מרחק גדול הוא יותר משמעותי.

לכן אם יש דאגה לערכים זבליים, עדיף לעבוד עם P קטן יותר כדי שערכים קיצוניים יהיו פחות משמעותיים.

אם אז נקבל 0, אחרת נקבל 1. זאת אומרת שעבור 2 וקטורים X וY אם הL0 שלהם שווה 7, אז יש 7 ערכים לא שווים בין וקטורים אלו.

לכן לפי L0 אין משמעות לערך ההפרש, הוא רק בודק אם יש הפרש, ככל שעולים בL (L1,L2,L3..) אז ההפרש הוא יותר משמעותי.

עבור נעלה בחזקת . לכן זה פונקציית מקסימום- מחשבים את כל ההפרשים בערך מוחלט, רואים מי ההפרש הכי גדול, הוא יהיה התוצאה של .

*C -מרכז הקלאסטר, בL0 יהיה השכיח של הנקודות.*

*ניתן להשתמש בL השונים לחלוקה לקלאסטרים בצורה הכי מדויקת, לנתונים שעובדים עליהם.*

*אם אז בודקים את הרחק של נקודה משאר הנקודות בערך מוחלט- לכן נקבל לכל קורדינטה את החציון שלה. ואז מרכז הקלאסטר יהיה לפי חציונים*

*דרך נוסת לחשב מרחק היא co side similarity- זווית בין 2 וקטורים*

*Chart

Description automatically generated*

*מה שחשוב זה הקשרים בין הנקודות והיחסים ביניהם.*

*הדרך לחשב את מרחקים אלו היא לנרמל את הוקטורים.*

*כך נקבל וקטור שסכום הערכים שלו הוא 1 ואז יהיה ניתן לחשב את המרחק בין הוקטורים*

*A picture containing text

Description automatically generated*

*גירסה אחרת היא לחשב לפי קוסינוס*

*Text

Description automatically generated with low confidence*

*אם Ai וBi זהים נקבל 1*

**

*במקרה בו יש 2 משתנים עם סקאלת ערכים מאוד שונה, נדרש לארגן את הוקטורים בצורה שיהיה ניתן לעבוד איתם.*

*שיטה 1: scaling- נקבל ערכים בין 0 ל1*

*הבעיה בשיטה זו היא ערכים חריגים, לדוגמה במשכורת המנכל מרוויח הרבה יותר מעובדיו ולכן הוא יהיה באזור ה1 וכל שאר העובדים מאוד קרוב ל0*

*על מנת לפתור את בעיה זו ניתן להשתמש בערכי אחוזונים ולא בכל תצפית בנפרד.*

*שיטה 2: להתייחס לנתונים כהתפלגות נורמלית. כך ניתן לחשב את צפיפות ההסתברות על פי ערך מסוים, לפי טבלת Z.*

*ניתן לגזום את הקצוות של הערכים החריגים-מאוד גבוהים או מאוד נמוכים, על מנת לפתור את בעיית הערכים החריגים.*

*שיטה 3:שימשו במשקלים על מנת להתייחס לחשיבות כל תכונה*

*לתת משקל לכל תכונה על פי החשיבות שלה ולחשב את המרחב בין הערכים של התכונות.*