חיזוי של דירוג מסעדה חדשה
לפי מסעדות קיימות

יקיר אוזן

315009019

uzan.yakir@campus.technion.ac.il

שרי יצחק

319011490

sarayyitzhak@campus.technion.ac.il

תוכן עניינים:

שער 1
תוכן עניינים 2
מבוא 3
תיאור הפתרון המוצע לבעיה 4
איסוף הנתונים 4
מקורות המידע 4
אופן איסוף וקבלת הנתונים 6
הכנת מסד הנתונים המרכזי 8
חיבור מסד הנתונים הראשי עם שאר מסדי הנתונים 8
השלמת ערכים חסרים במסד הנתונים המרכזי 9
אופן חישוב הדירוג המשוקלל 10
הסרת תכונות במסד נתונים המרכזי 12
התכונות במסד הנתונים המרכזי 13
האלגוריתם 14
שדרוג פעולת האלגוריתם 14
חישוב ערך MSE 16
חיזוי דירוג המסעדה 16
תיאור המערכת 17
שרטוט המערכת 18
תהליך האימון 18
תהליך החיזוי 18
תיאור הניסויים 19
היפר-הפרמטרים 19
המדדים 19
ביצוע הניסויים 19
תוצאות הניסויים של דיוק תוצאות החיזוי עבור קבוצת המבחן 20
תוצאות הניסויים של ערכי MSE עבור קבוצת המבחן 24
תוצאות הניסויים עבור קבוצת המבחן 24
תוצאות הניסויים של דיוק תוצאות החיזוי עבור קבוצת האימון 25
תוצאות הניסויים של ערכי MSE עבור קבוצת האימון 29
תוצאות הניסויים עבור קבוצת האימון 29
סיכום 30
קשיים במהלך הפרויקט 30
כיוונים להמשך המחקר 30
ביבליוגרפיה 31

מבוא:

נכון לשנת 2020, בממוצע נסגרות בישראל 3000 מסעדות בשנה. המחזור השנתי של תעשיית המסעדות עמד על 30 מיליארד שקל (בשנת 2019), והוא מעסיק 203 אלף עובדים. לרוב, המסעדות ובתי הקפה לא "שורדים" את השנה הראשונה, פעמים רבות זה נגרם עקב חוסר התאמה של המסעדה לאזור או לאוכלוסייה. דבר זה גורם לבעלי מסעדות הפסדים כספיים רבים.

לפי הכתבה: *"תוך חצי שנה: ממסעדה לא כשרה למסעדה כשרה – ובחזרה"*, ראינו שהמסעדה "פדרו" באילת שינתה את אופי המסעדה (כשרות) שלה ולאחר חצי שנה החזירה חזרה את אופי המסעדה. "פדרו" החזירו את אופי המסעדה הקודם מכיוון שמסעדה כשרה לא התאימה לאופי האנשים באזור. המסעדה "פדרו" אופיינה במספר סוגי היין הרב שיש במסעדה והיווה חלק ניכר מהכנסות המסעדה.
*"**אז נכון שהשקענו** במהלך של ההפיכה לכשרים **כסף**, אבל זה הביא לנו קהל שלא הייתי נחשף אליו אם לא הייתי כשר."*
*"...אבל אחרי הכל, קהל אוכלי הכשר זה קהל שרובו לא ישקיע כסף בארוחות כאילו אין מחר. **בקהל הלא כשר יש אנשים שאוהבים לשתות ולאכול ולשתות יין יקר, ובקהל הכשר – פחות**".*
*"...היה לנו במתכונת הלא כשרה בערך 80-70 סוגי יין. אבל פתאום הרגשנו שגם מגוון היין ירד וגם **הקהל שמגיע לא ממש מתעניין בזה**."*

כאמור בכתבה המסעדה לא נסגרה בסופו של דבר, אך איבדה כסף רב, משאבים ואולי גם אנשים שאיבדו עניין במסעדה עקב החלטה שגויה.

כתבה נוספת: *"שגב משנה קונספט: ממסעדה כשרה למסעדה לא כשרה".*
"רק שנה מאז שפתח את המסעדה הכשרה הראשונה שלו בבאר שבע, משנה השף שגב משה את הקונספט והופך אותה למסעדה לא כשרה."
"המתחם בקושי עובד באמצע השבוע, אנחנו עסק כלכלי ואין שום היגיון להפעיל שם מסעדה כשרה"
*"גם עם זה שאנחנו אמנים אנחנו אנשי עסקים, ואנחנו מבינים **שההחלטה הייתה שגויה**."*
*"העסק צריך להתקיים בזכות עצמו במובן הכלכלי, ואתה כמסעדן **צריך להתאים את עצמך לשטח**."*

נפתור בעיה זו על ידי חיזוי דירוג המסעדה לפני פתיחתה או בזמן קיומה של המסעדה, על פי מאפיינים שידועים מראש. כך יוכל הבעלים להחליט האם כדאי לו לפתוח את המסעדה, או האם הוא נדרש לעשות שינויים בסגנון המסעדה, מיקום המסעדה וכו'.

תיאור הפתרון המוצע לבעיה:

בפרויקט זה נתמודד עם הבעיה בתור בעיית למידה. בחרנו להשתמש בעץ רגרסיה. הרעיון להשתמש בעץ החלטה הוא שיש המון שאלות שניתן לשאול על מנת למצוא מכנה משותף בין מסעדות מצליחות למסעדות שאינן מצליחות. על ידי שאלות אלו ניתן לבנות עץ החלטה וכך נוכל לסווג בצורה מדויקת את המסעדות לפי המאפיינים שלהם. היתרון של עץ החלטה על פני מסווגים אחרים הוא שהמסווג ניתן לפירוש על ידי בני אדם, ניתן לראות את החלטות האלגוריתם בכל שלב ולהסיק את מידת החשיבות של כל תכונה. בנוסף, בעץ החלטה ניתן להשתמש בדוגמאות עם תכונות חסרות.

איסוף הנתונים:

החלק החשוב ביותר בפרויקט מסוג זה הוא המידע שאנו משתמשים, מסד הנתונים שעליו אלגוריתם הלמידה שלנו מסתמך. במידה ומקור הנתונים שלנו יהיה לא אמין או לא מגוון אז המסווג שיצא לנו יהיה לא מוצלח.

את הנתונים אספנו עבור הערים: חיפה, הקריות, נשר וטירת הכרמל.

<u>מקורות המידע:</u>

1. Google – עיקר המידע שלנו, ממקור זה אנו מקבלים מידע שימושי שיש למסעדות. ממקור זה אנו מקבלים את התכונות הבאות:
   - מזהה המסעדה לפי גוגל (place id).
   - שם המסעדה (name).
   - רמת מחיר (price level).
   - כתובת (city, street, street number).
   - מיקום גיאוגרפי (geo location).
   - שעות פתיחה לפי ימים (activity hours).
   - האם ניתן לאכול במקום (dine in).
   - האם ניתן לבצע משלוחים (delivery).
   - האם ניתן לבצע איסוף עצמי (takeout).
   - האם ניתן להזמין מקום (reservable).
   - האם מגישים בירה (serves beer).
   - האם מגישים יין (serves wine).
   - האם מגישים ארוחות בוקר (serves breakfast).
   - האם מגישים בראנץ' (serves brunch).
   - האם מגישים ארוחות צהרים (serves lunch).
   - האם מגישים ארוחות ערב (serves dinner).
   - האם מגישים אוכל טבעוני (serves vegetarian food).
   - האם יש נגישות לנכים (wheelchair accessible entrance).
   - האם ניתן לבצע איסוף בצד המדרכה (curbside pickup).
   - האם פתוח בשבת (open on Saturday).
   - האם יש אתר (website).

בנוסף לתכונות אלו, אנו מקבלים גם את הדירוגים (הסיווג) של המסעדות:

- דירוג (rating).
- מספר מדרגים (user ratings total).

בעזרת שני תיוגים אלו אנו יוצרים את הדירוג המשוקלל (יתואר בהמשך) של המסעדה.

2. Google Places – ממקור זה אנו מקבלים את המיקומים של המסעדות והחנויות.

ממקור זה אנו מקבלים את התכונות הבאות:

- מזהה המקום לפי גוגל (place id).
- מיקום גיאוגרפי (geo location).
- סוג, מסעדה/חנות (type).

3. Rest – ממקור זה אנו מקבלים את סוג המסעדה והאם היא כשרה.

ממקור זה אנו מקבלים את התכונות הבאות:

- מזהה המקום לפי rest (id).
- שם המסעדה (name).
- סוג המסעדה (type).
- האם היא כשרה (kosher).
- רמת מחיר (price level).
- כתובת (city, address).

4. Gov – מאגרי המידע הממשלתיים – ממקור זה אנו מקבלים את המיקומים של כל תחנות האוטובוס.

ממקור זה אנו מקבלים את התכונות הבאות:

- מזהה התחנה לפי gov (station id).
- מיקום גיאוגרפי (geo location).

5. Cbs – הלשכה המרכזית לסטטיסטיקה – ממקור זה אנו מקבלים את המצב הסוציו-אקונומי ואת אחוז החרדים לפי רחובות.

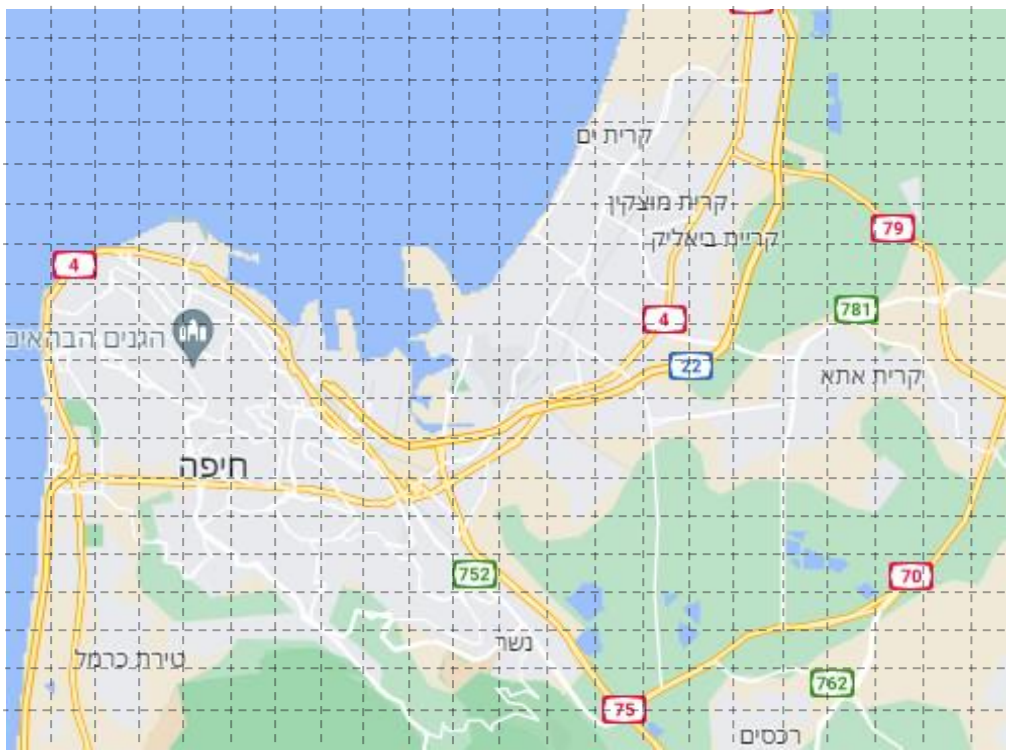
ממקור זה אנו מקבלים את התכונות הבאות:

- עיר (city).
- רחוב (street).
- אחוז החרדים (percent of religious).
- מצב סוציו-אקונומי (socio-economic cluster, rank, value).

אופן איסוף וקבלת הנתונים:

מכל מקור מידע אספנו את הנתונים בצורה שונה, דרכי התמודדות שונות וכן קשיים שנאלצנו לפתור עבור מקרים מסוימים.

1. Google – השתמשנו ב- API של גוגל, שבו מבקשים מסעדות/חנויות לפי מיקום גיאוגרפי. הקושי שנתקלנו בו היה שלכל שאילתת API, מספר המקומות שקיבלנו מהשאילתה היה מוגבל (60), לכן לא יכולנו לבקש את כל המסעדות שיש בחיפה (למשל) בשאילתה אחת. התמודדנו עם קושי זה על ידי מספר רב של בקשות. יצרנו רשת של נקודות על פני המפה ולכל נקודה ביצענו שאילתת API. המרחק בין הנקודות היה נמוך יחסית וקבוע (1 קילומטר).



2. Rest – את הנתונים אנחנו משיגים על ידי web scraping. לכל עיר אנו רושמים בשורה של כתובת ה- URL את הקישור המתאים לאתר Rest, משם אנו לוקחים את תוצאות החיפוש ומחלצים את הנתונים שאנו רוצים לאסוף.

3. Gov – את הנתונים אנחנו משיגים על ידי בקשות HTTPS ל- API של האתר Gov. לכל עיר אנו מבצעים שאילתת HTTPS, מקבלים את הנתונים הרלוונטיים של אותה העיר ומחלצים את הנתונים שאנו רוצים לאסוף.

4. Cbs – את הנתונים אנו מורידים מהאתר של Cbs בצורת xlsx. אנו מבצעים 2 מיפויים על מנת לאסוף את הנתונים.
לכל אזור (שמכיל מספר רחובות) יש קוד שנקרא "סמל אזור סטטיסטי".
במהלך השנים, שונו סמלי האזור הסטטיסטיים, לכן יש צורך לבצע המרה ביניהם.
הנתונים שאנו משיגים ממקור זה מכילים את סמל האזור הסטטיסטי הישן והאקסל שמכיל את המעבר מסמל אזור סטטיסטי לרחובות משתמש בסמל האזור הסטטיסטי החדש. השתמשנו באקסל שמכיל מיפוי בין סמל אזור סטטיסטי ישן לאזור סטטיסטי חדש.
כלומר סה"כ יש 4 אקסלים:

- אחוז החרדים לפי סמל אזור סטטיסטי ישן.
- מצב סוציו-אקונומי לפי סמל אזור סטטיסטי ישן.
- טבלת המרה מסמל אזור סטטיסטי ישן לסמל אזור סטטיסטי חדש.
- רחובות לפי סמל אזור סטטיסטי חדש.

דוגמה:

מחוז	סמל יישוב	שם יישוב	אזור סטטיסטי	סך הכל אוכלוסייה באזור הסטטיסטי	מזה: חרדים	
					מספרים מוחלטים	אחוזים
חיפה	4000	חיפה	811	3,828	1,012	26.4

לפי השורה לעיל ניתן לראות שבאזור הסטטיסטי 811 שבעיר חיפה יש 26.4 אחוז חרדים.

סמל א"ס 2011	סמל א"ס ייחודי 2011	סמל ישוב 2022	שם ישוב 2022	סמל א"ס 2022	סמל א"ס ייחודי 2022
0811	40000811	4000	חיפה	0721	40000721

ניתן לראות, שסמל אזור סטטיסטי 811 (הישן) השתנה ל- 721 (החדש).

שם יישוב	סמל א"ס מ	סמל יישוב	א"ס	רחובות עיקריים
חיפה	40000721	4000	721	דורי יעקב רא"ל, שד טרומפלדור, לוחמי חרות ישראל, הגליל, מימון הרב

לסיום, הרחובות שנמצאים באזור הסטטיסטי 721 הם דורי יעקב רא"ל, שד טרומפלדור, לוחמי חרות ישראל, הגליל ומימון הרב.

לסיכום קיבלנו שברחובות הללו יש 26.4 אחוז חרדים.

הכנת מסד הנתונים המרכזי:

עד כה, יצרנו 5 מסדי נתונים: Google, Google Places, Rest, Gov, Cbs.

כעת, נרצה לחבר את כולם למסד נתונים אחד מרכזי.

<u>חיבור מסד הנתונים הראשי (גוגל) עם שאר מסדי הנתונים:</u>

1. Google Places – אנו מוסיפים לכל מסעדה את מספר החנויות ואת מספר המסעדות שיש לה באזור לפי רדיוס (100 מטרים ו- 500 מטרים). לכל מסעדה, אנו סופרים את המקומות שהמרחק שלהם מהמסעדה קטן מ- 100 מטרים ו- 500 מטרים בעזרת המיקום הגיאוגרפי.
אחוז התאמה: 100%

2. Rest – אנו מוסיפים לכל מסעדה האם היא כשרה ואת סגנון המסעדה לפי הכתובת והשם שלה. עבור כל מסעדה במסד הנתונים של גוגל, אנו בודקים את כתובתה. נחפש את המסעדות עם כתובת דומה במסד הנתונים של Rest, ולאחר מכן נבדוק אם יש מסעדה עם שם זהה. במידה ולא מצאנו מסעדות עם כתובת דומה, נבצע חיפוש רק לפי השם.
במידה ויש התאמה אנו מוסיפים את סוג המסעדה והאם היא כשרה.
ביצענו התאמה בדרך זו מכיוון ששמנו לב שישנם שמות מעט שונים עבור אותה מסעדה בין מסדי הנתונים.
אחוז התאמה של האם היא כשרה: 55%
אחוז התאמה של סוג המסעדה: 43.3%

3. Gov – אנו מוסיפים לכל מסעדה את מספר תחנות האוטובוס שיש לה באזור לפי רדיוס (100 מטרים ו- 500 מטרים). לכל מסעדה, אנו סופרים את תחנות האוטובוס שהמרחק שלהם מהמסעדה קטן מ- 100 מטרים ו- 500 מטרים בעזרת המיקום הגיאוגרפי.
אחוז התאמה: 100%

4. Cbs – אנו מוסיפים לכל מסעדה את אחוז החרדים ואת המצב הסוציו-אקונומי שיש באזור לפי כתובת. לכל מסעדה, אנו מחפשים לפי הכתובת את השורה המתאימה במסד הנתונים הנ"ל ולפי זה אנו מתאימים את אחוז החרדים ואת המצב הסוציו-אקונומי.
אחוז התאמה של אחוז החרדים: 27.7%
אחוז התאמה של המצב הסוציו-אקונומי: 39.6%

<u>השלמת ערכים חסרים במסד הנתונים המרכזי:</u>

השלמת ערכים חסרים הוא שלב חשוב בתהליך עיבוד המידע. ערכים חסרים פוגעים בשלמות הנתונים ומקשים על ניתוח התוצאות. בנוסף, ערכים חסרים עלולים לגרום להטיה של תוצאות האלגוריתם. כמו כן, השלמת ערכים חסרים משפרת את הביצועים של המודל בתהליך האימון.

כפי שניתן לראות, אחוז ההתאמה של מסדי הנתונים Cbs ו- Rest הם מועטים יחסית, נרצה להשלים את הערכים החסרים בעזרת ידע כללי על התכונות.

1. Rest –

- כידוע, מסעדה שפתוחה בשבת בהכרח לא כשרה. לכן, ניתן לדעת בעזרת התכונה, open on Saturday, האם המסעדה בהכרח לא כשרה. כלומר, אם ערך תכונה זו הוא True אז ערך התכונה Kosher יהיה False.
  אחוז התאמה של האם היא כשרה: 74% (+19%)

- לפי שם המסעדה ניתן לדעת, בחלק מהמקרים, את סוג המסעדה. למשל מסעדות שיש להם בשם את המילה "בורגר" (כנראה) יהיו מסעדות המבורגרים. מילים נוספות ששייכנו לסוגים שונים:
  - "בורגר" – מסעדת המבורגרים
  - "פיצה" – פיצריות
  - "קפה" – בתי קפה
  - "חומוס" – חומוסייה

  אחוז התאמה של סוג המסעדה: 51% (+7.7%)

2. Cbs – במסעדות שאנו לא יודעים את אחוזי החרדים/המצב הסוציו-אקונומי, השלמנו את החסר בעזרת ממוצע הערכים של המסעדות שאנו יודעים את הנתונים הנ"ל לפי מרחק (בדקנו את המסעדות במרחק של 500 מטרים).
   אחוז התאמה של אחוז החרדים: 68.4% (+40.7%)
   אחוז התאמה של המצב הסוציו-אקונומי: 74.4% (+34.8%)

3. Google – במסעדות שאנו לא יודעים את רמת המחיר לפי גוגל, אנו משלימים את החסר בעזרת רמת המחיר שיש לנו לפי Rest, לכן אם מצאנו התאמה של מסעדה עם Rest ויש עבורה את התכונה (רמת המחיר) אז אנו מוסיפים את הערכים הללו.
   אחוז הערכים המלאים לפני: 36.3%
   אחוז הערכים המלאים אחרי: 42.5% (+6.2%)

כעת, נשלים חלק מהערכים החסרים הנותרים על ידי שיטות מוכרות:

1. Google –

- ערכים בוליאנים אנו נשלים עם הערך הקבוע False. אנו מניחים שערכים בוליאנים ריקים הם ככל הנראה False, אחרת בעל המסעדה (או לקוחות שמזינים את הערכים הללו בגוגל) היה משלים את זה עם הערך True. למשל עבור השאלה "האם המסעדה עושה משלוחים?" אם המסעדה אכן הייתה עושה משלוחים, ככל הנראה בעל המסעדה היה רושם True, אחרת או שהוא רושם False או שהוא ישאיר ערך זה ריק.

הערה: את הערך "פתוח בשבת" אנו יצרנו בעזרת התכונות של שעות פתיחה/סגירה של המסעדה. למעשה, חישבנו את הערך הזה בעזרת שעות הסגירה ביום שישי ושעות הפתיחה ביום שבת. במידה ואין שעות פתיחה/סגירה למסעדה, השארנו את הערך הזה ריק. לכן את הערך (הבוליאני) הזה אנו לא נשלים עם הערך הקבוע False כי אנו לא יודעים את שעות הפתיחה/סגירה (נתמודד עם זה בהמשך).

- רמת מחיר ושעות פתיחה/סגירה אנו נשלים בעזרת הערך הממוצע של כל המסעדות.

2. Cbs – אחוז החרדים ומצב סוציו-אקונומי אנו נשלים בעזרת ערך הממוצע של כל המסעדות. לאחר השלמת הערכים לפי מרחק, נשלים את הערכים החסרים הנותרים לפי ממוצע של כל המסעדות.

הערה: חלק מהתכונות עדיין מכילות ערכים ריקים ולא נשלים אותן, והן "פתוח בשבת" כפי שהוסבר לעיל, "סוג המסעדה" ו- "האם כשר". בחרנו לא להשלים ערכים ריקים של תכונות אלו מכיוון שאנו לא יודעים מהו הערך האופטימלי והעדפנו לא למלא ערכים שגויים מכיוון שאלו פוגעים באיכות המודל ובתוצאותיו. נתמודד עם ערכים ריקים באלגוריתם עצמו כפי שיתואר בהמשך.

<u>אופן חישוב הדירוג המשוקלל (הסיווג):</u>

כאמור, יש לנו את התכונות הבאות (ממסד הנתונים של גוגל):

- דירוג
- מספר מדרגים

בעזרתן אנו יוצרים את הדירוג המשוקלל של המסעדה.
הדירוג המשוקלל יהיה מספר בין 10 ל- 100.

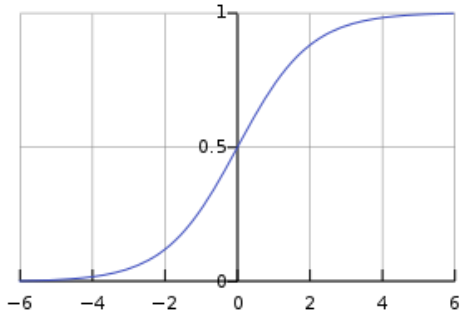
הדירוג נע בין 1 ל- 5.
מספר המדרגים נע בין 1 לבין 10,000 (עבור הנתונים שאספנו).
רצינו לשקלל את שתי התכונות כדי להבדיל, למשל, בין מסעדות עם דירוג 5 ומספר מדרגים גבוה לבין מסעדות עם דירוג 5 ומספר מדרגים נמוך, דבר שמעיד על אמינות הציון ופופולאריות המסעדה.
עשינו זאת על ידי נירמול מספר המדרגים וכך יצרנו ערך שמבטא את אחוז האמינות של הדירוג. פונקציית הנירמול שבחרנו היא סיגמואיד מכיוון שרצינו שערכים גבוהים יהיו קרובים לאחד וערכים נמוכים (חיוביים) יהיו קרובים לחצי (מכיוון שאנו לא יודעים האם הדירוג אמין, כי יש מעט דירוגים, אז 50 אחוז שהדירוג מייצג את המסעדה ו- 50 אחוז שהוא לא).

נסמן:

$$\sigma(n) = \frac{1}{1 + e^{-n}}$$

כאשר n הוא מספר המדרגים.

נשים לב שהערכים החיוביים הראשונים של הסיגמואיד הם:



$\sigma(1) = 0.73$
$\sigma(2) = 0.88$
$\sigma(3) = 0.95$
$\sigma(4) = 0.98$

קיבלנו שעבור מספר מדרגים נמוך יחסית אנחנו מגיעים כמעט לאחד (שמייצג 100% אמינות).

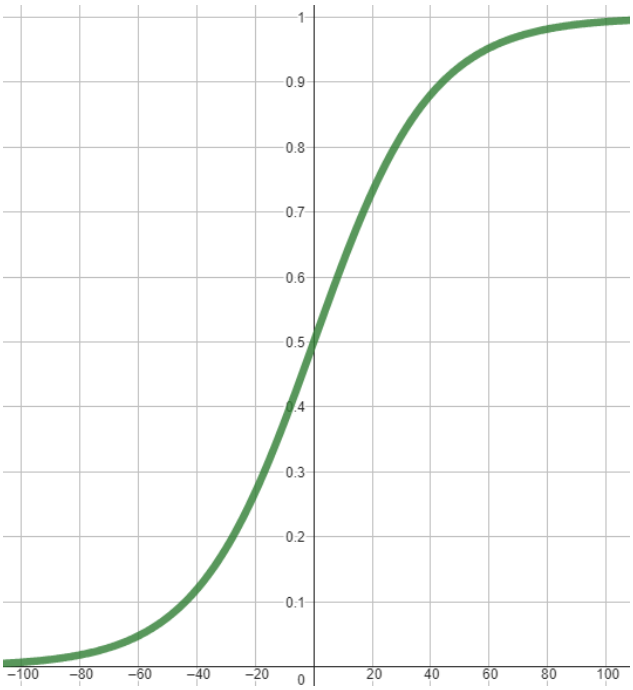
נפתור בעיה זו על ידי הקטנת המקדם שיש באקספוננט וכך "נשטח" את הסיגמואיד, נקבל:

$$\sigma(n) = \frac{1}{1 + e^{-0.05n}}$$

כאשר n הוא מספר המדרגים.

נשים לב לכמה ערכים שיש לסיגמואיד החדש שיצרנו:

$\sigma(1) = 0.51$
$\sigma(2) = 0.52$
$\sigma(10) = 0.62$
$\sigma(20) = 0.73$
$\sigma(50) = 0.92$
$\sigma(80) = 0.98$
$\sigma(100) = 0.99$
$\sigma(1000) \approx 1$



כלומר קיבלנו שעבור ערכים נמוכים אנחנו קרובים לחצי (יותר מהפונקציה הקודמת) ועבור ערכים גבוהים (יחסית) אנחנו קרובים לאחד. בנוסף, נשים לב, שעבור 80 מדרגים ועבור 1000 מדרגים, אחוז האמינות כמעט 1 בשני המקרים. זה מאוד הגיוני, כלומר אם יש 80 מדרגים אז כנראה שהדירוג של המסעדה אמין בדומה לדירוג של מסעדה עם 1000 מדרגים. כלומר בחרנו את המקדם 0.05- באופן כזה, שנקבל את ההתנהגות הזו, עבור ערכים בין 1 ל- 100 נקבל התנהגות של סיגמואיד ועבור ערכים שגבוהים מ- 100 נקבל כמעט את הערך 1.

כעת נציג את הפונקציה שמחשבת את הדירוג המשוקלל של מסעדה:

$$f(x, n) = \sigma(n) \cdot x \cdot 20$$

כאשר x הוא הדירוג ו- n הוא מספר המדרגים.

נשים לב שהדירוג נע בין 1 ל- 5, לכן כאשר נכפיל ב- 20 נקבל ערך שנע בין 20 ל- 100.
לאחר הכפלה בסיגמואיד נקבל ערך שנע בין 10 ל- 100.

הסרת תכונות במסד נתונים המרכזי:

אנו בחרנו להסיר תכונות עם קורלציה גבוהה. הסרת תכונות אלו יכולה למנוע Overfitting, כך שבמקום לתת יותר דגש לתכונות עם קורלציה גבוהה, תהיה יותר התחשבות בתכונות עם מידע רלוונטי יותר. בנוסף, מכיוון שאנו משתמשים בעץ החלטה, יש משמעות גבוהה להסרת תכונות מכיוון שזה יגרום לשיפור בחירת השאלות (הפיצול). כמו כן, הסרת תכונות עם קורלציה גבוהה תגרום לשיפור ביצועים ויעילות חישובית.

1. שעות פעילות – נשים לב לפי הטבלאות, שיש קורלציה גבוהה בין שעות הפעילות של ימים שני, שלישי, רביעי וחמישי, לכן נרצה לבחור רק אחד מהימים הללו (נבחר את יום רביעי, מכיוון שערך הקורלציה שלו עם כל אחד בממוצע גבוה יותר משאר הימים).

	sunday_open	monday_open	tuesday_open	wednesday_open	thursday_open	friday_open	saturday_open
sunday_open	1.00000	0.58027	0.57793	0.59934	0.53493	0.17126	0.19440
monday_open	0.58027	1.00000	0.89800	0.92959	0.80651	0.34523	0.29834
tuesday_open	0.57793	0.89800	1.00000	0.96713	0.84525	0.37068	0.31466
wednesday_open	0.59934	0.92959	0.96713	1.00000	0.87213	0.38417	0.30679
thursday_open	0.53493	0.80651	0.84525	0.87213	1.00000	0.36185	0.27978
friday_open	0.17126	0.34523	0.37068	0.38417	0.36185	1.00000	0.29150
saturday_open	0.19440	0.29834	0.31466	0.30679	0.27978	0.29150	1.00000

	sunday_close	monday_close	tuesday_close	wednesday_close	thursday_close	friday_close	saturday_close
sunday_close	1.00000	0.65011	0.66591	0.66525	0.63560	0.28290	0.33484
monday_close	0.65011	1.00000	0.92771	0.93329	0.87208	0.48027	0.51974
tuesday_close	0.66591	0.92771	1.00000	0.97526	0.91170	0.51513	0.55860
wednesday_close	0.66525	0.93329	0.97526	1.00000	0.91832	0.52242	0.55990
thursday_close	0.63560	0.87208	0.91170	0.91832	1.00000	0.53406	0.55725
friday_close	0.28290	0.48027	0.51513	0.52242	0.53406	1.00000	0.55333
saturday_close	0.33484	0.51974	0.55860	0.55990	0.55725	0.55333	1.00000

2. יש קורלציה גבוהה (0.97) בין שתי התכונות הבאות:
   a. האם מגישים בירה (serves beer).
   b. האם מגישים יין (serves wine).

   כלומר בהרבה מסעדות שמגישים בירה מגישים גם יין ולהיפך.
   לכן נבחר רק תכונה אחת (נבחר שרירותית את התכונה: האם מגישים בירה).

3. מצב סוציו-אקונומי – ניתן לראות לפי הטבלה שיש קורלציה גבוהה בין התכונות השונות שקשורות למצב הסוציו-אקונומי. לכן נבחר בתכונה: socio-economic rank מכיוון שערך הקורלציה שלו עם השאר בממוצע גבוה יותר משאר התכונות.

	socio-economic index value	socio-economic rank	socio-economic cluster
socio-economic index value	1.00000	0.99135	0.98534
socio-economic rank	0.99135	1.00000	0.99196
socio-economic cluster	0.98534	0.99196	1.00000

Fun Fact: גילינו שיש קורלציה יחסית גבוהה (0.5) בין מספר האוטובוסים באזור לבין אחוז החרדים באזור.

התכונות במסד הנתונים המרכזי:

שם	הסבר מילולי	סוג/ערכים אפשריים
dine_in	האם ניתן לאכול במקום	[True, False]
delivery	האם ניתן לבצע משלוחים	[True, False]
reservable	האם ניתן להזמין מקום	[True, False]
serves_beer	האם מגישים בירה	[True, False]
serves_breakfast	האם מגישים ארוחות בוקר	[True, False]
serves_brunch	האם מגישים בראנץ'	[True, False]
serves_dinner	האם מגישים ארוחות ערב	[True, False]
serves_lunch	האם מגישים ארוחות צהרים	[True, False]
serves_vegetarian_food	האם מגישים אוכל טבעוני	[True, False]
takeout	האם ניתן לבצע איסוף עצמי	[True, False]
wheelchair_accessible_entrance	האם יש נגישות לנכים	[True, False]
curbside_pickup	האם ניתן לבצע איסוף בצד המדרכה	[True, False]
website	האם יש אתר	[True, False]
price_level	רמת המחיר	[1, 2, 3]
sunday_open	שעת פתיחה בראשון (בדקות)	[0 - 1440]
sunday_close	שעת סגירה בראשון (בדקות)	[0 - 1440]
wednesday_open	שעת פתיחה ברביעי (בדקות)	[0 - 1440]
wednesday_close	שעת סגירה ברביעי (בדקות)	[0 - 1440]
friday_open	שעת פתיחה בשישי (בדקות)	[0 - 1440]
friday_close	שעת סגירה בשישי (בדקות)	[0 - 1440]
saturday_open	שעת פתיחה בשבת (בדקות)	[0 - 1440]
saturday_close	שעת סגירה בשבת (בדקות)	[0 - 1440]
open_on_saturday	האם פתוח בשבת	[True, False]
geo_location	מיקום גיאוגרפי	(float, float)
percent of religious	אחוז החרדים באזור	[0 - 100]
socio-economic rank	מצב סוציו-אקונומי באזור	float
store_100	מספר חנויות באזור ברדיוס של 100 מטרים	int
store_500	מספר חנויות באזור ברדיוס של 500 מטרים	int

rest_100	מספר מסעדות באזור ברדיוס של 100 מטרים	int
rest_500	מספר מסעדות באזור ברדיוס של 50 מטרים	int
bus_station_100	מספר תחנות אוטובוס באזור ברדיוס של 100 מטרים	int
bus_station_500	מספר תחנות אוטובוס באזור ברדיוס של 500 מטרים	int
type	סוג המסעדה (35 אופציות)	str
kosher	האם היא כשרה	[True, False]
grade	הדירוג המשוקלל (הסיווג) של מסעדה	[10 - 100]

האלגוריתם:

האלגוריתם שאיתו בחרנו לעבוד הוא עץ רגרסיה. בחרנו בעץ רגרסיה מכיוון שקל להבין את תוצאות הפלט של האלגוריתם (העץ), אינטואיטיבי לפרש אותו. בנוסף, התכונות שלנו מקיימות את הנדרש עבור שאלה בינארית ולכן קל לפצל אותן. כמו כן, עץ החלטה בוחר מטבעו את התכונות החשובות ביותר בראש העץ.

<u>שדרוג פעולת האלגוריתם:</u>

כפי שראינו בשלב בניית מסד הנתונים, קיימים ערכים חסרים בתכונות מסוימות. במהלך האלגוריתם, אנו מצפים שלא יהיו לנו ערכים ריקים מכיוון שלא ניתן לענות על שאלה עם ערך ריק. השלמנו חלק מהערכים בעזרת ידע כללי על התכונות, ממוצעים וערכים קבועים, אך עדיין יש לנו ערכים ריקים. במקום איבוד דוגמאות של מסעדות שחסרים להן ערכים בתכונות מסוימות, ניסינו להתמודד עם ערכים ריקים תוך ריצת האלגוריתם.

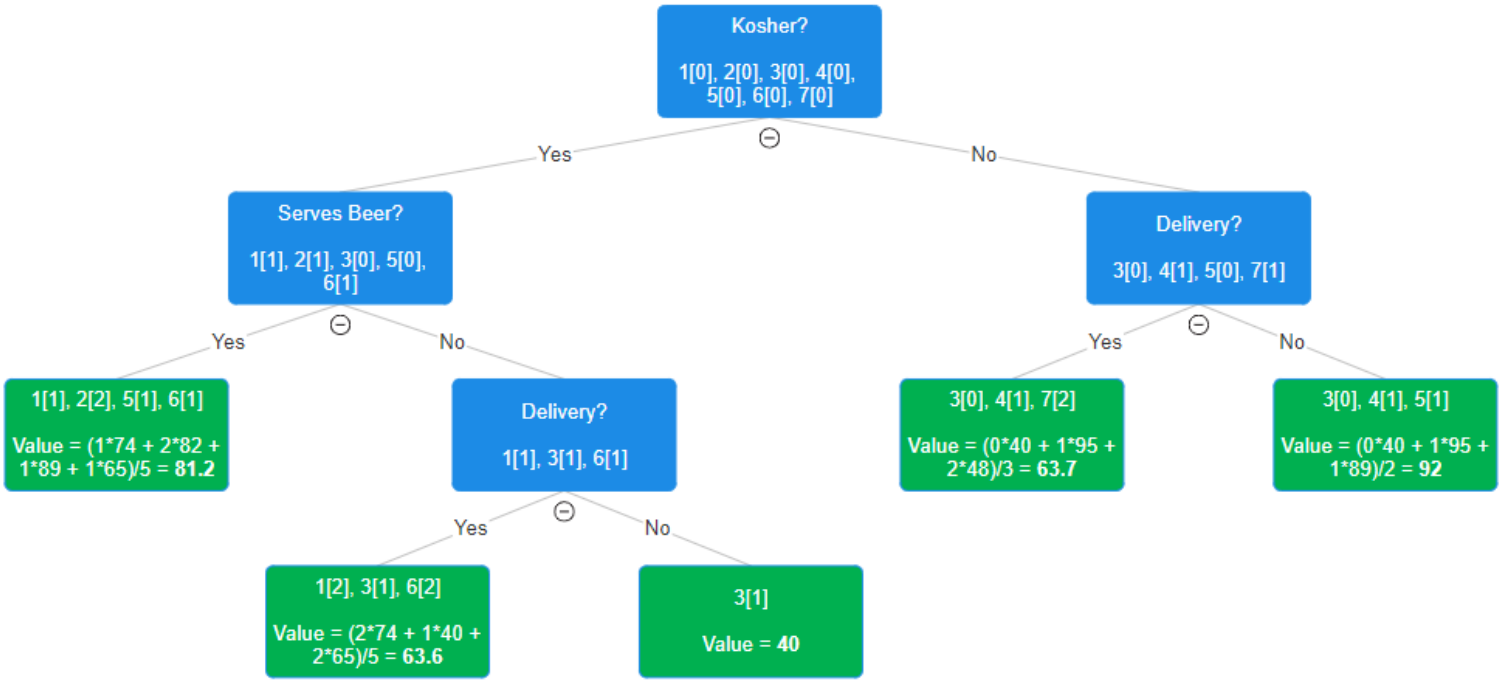
במהלך בניית עץ ההחלטה, כאשר בוחרים תכונה שלפיה מפרידים את הנתונים (בעזרת שאלה בינארית), אם תהיה דוגמה שערך התכונה שלה הוא ריק אנו נוסיף את הנתונים של אותה דוגמה לשני הצמתים (לצומת שעונה "כן" ולצומת שעונה "לא"). אחרת, נוסיף את הנתונים של אותה דוגמה לצומת הרלוונטי. לדוגמה: נניח שהתכונה/שאלה שלפיה מפרידים את הנתונים כעת היא "האם המסעדה כשרה?". עבור מסעדה שערך הכשרות שלה הוא ריק, נשים את המסעדה הנ"ל בשני הצמתים שנובעים מצומת שאלה זה.

כעת, אנו נוסיף בכל שלב משקלים המייצגים את כמות השאלות שנענו לכל דוגמה במהלך ריצת העץ. במידה וענינו "כן" או "לא" נוסיף אחד למשקל של מסעדה זו, אחרת (הערך ריק) לא נוסיף משקל. לבסוף, כשנגיע לעלה, נחשב את ערך העלה על ידי ממוצע משוקלל שיש לכל מסעדה. נדגים:

נניח שנתון לנו מסד הנתונים הנ"ל:

Index\Feature	Kosher	Serves Beer	Delivery	Grade
1	True	(blank)	True	**74**
2	True	True	True	**82**
3	(blank)	False	(blank)	**40**
4	False	True	(blank)	**95**
5	(blank)	True	False	**89**
6	True	(blank)	True	**65**
7	False	(blank)	True	**48**

נניח שהעץ שנוצר הוא:



כאשר צמתים פנימיים בצבע כחול ועלים בצבע ירוק. השאלות מופיעות בשורה הראשונה והמשקל של כל מסעדה מופיע בתוך סוגריים מרובעים לצד האינדקס של המסעדה. הערכים בעלים מחושבים לפי ממוצע משוקלל בהתאם לדירוג המסעדה והמשקל שלה.

נסתכל על חישוב ערך העלה השמאלי ביותר: בהתחלה המשקל של כל מסעדה הוא 0 כמו שניתן לראות בראש העץ. לאחר השאלה הראשונה, "האם המסעדה כשרה?", המסעדות באינדקס 1,2,6 ענו "כן" והמסעדות באינדקס 3,5 עם ערך ריק. לכן, המשקל של מסעדות 1,2,6 עלה ב-1 והמשקל של מסעדות 3,5 נשאר 0. לאחר השאלה השנייה, "האם המסעדה מגישה בירה?", המסעדות שענו "כן" הם באינדקס 2,5 ולכן המשקלים שלהם הם 2,1 בהתאמה והמסעדות 1,6 עם ערך ריק עבור שאלה זו, לכן המשקל שלהם נשאר 1. כעת, נחשב את ערך העלה הנ"ל בהתאם למשקלים: $\frac{1\cdot 74+2\cdot 82+1\cdot 89+1\cdot 65}{5} = 81.2$.

חישוב ערך MSE:

החישוב מבוסס על ממוצע מרחק הערכים מהממוצע בריבוע. כאמור, כשיש ערכים ריקים אנו מכניסים את הדוגמה לשני הצמתים הבנים, אך בעת ביצוע חישוב ה- MSE, אנו מכניסים לחישוב את הערכים הלא ריקים בלבד. כלומר, רק הדוגמאות שענו על השאלה הבינארית. ננסה להבין מדוע דרך זו עדיפה על ידי דוגמה. נחשב את ערכי ה- MSE של הבנים של הצומת השמאלי (Serves Beer?) מהדוגמה לעיל בשתי דרכים:

- ללא ערכים ריקים:
  צד שמאל מכיל את שורות 2,5 וצד ימין מכיל את שורה 3. נתחיל מצד שמאל:
  הממוצע של הערכים הוא 85.5. ערך ה- MSE הוא 12.25. צד ימין: יש רק ערך אחד לכן ערך ה- MSE הוא 0.
  ערך ה- MSE המשוקלל הוא 8.1667.

- עם ערכים ריקים:
  צד שמאל מכיל את שורות 1,2,5,6 וצד ימין מכיל את שורות 1,3,6. נתחיל מצד שמאל: הממוצע של הערכים הוא 77.5. ערך ה- MSE הוא 88.25. צד ימין: הממוצע של הערכים הוא 59.67. ערך ה- MSE הוא 206.89.
  ערך ה- MSE המשוקלל הוא 132.95.

ניתן לראות שכאשר הוספנו את הערכים הריקים, ערך ה- MSE קפץ מאוד ולכן יתכן ולא יבחר את השאלה הנ"ל בעקבות זאת, למרות שבלי הערכים הריקים ערך ה- MSE נמוך ולכן יתכן שהשאלה תהיה השאלה הכי טובה בנקודה זו. בנוסף, כשמחשבים את ערך ה- MSE עם ערכים ריקים הוא עלול לעודד שאלות שמכילות הרבה ערכים ריקים (פחות אינפורמטיביות) לעומת שאלות עם פחות ערכים ריקים. לכן בחרנו בתהליך בניית העץ שלנו לחשב את ערך ה- MSE ללא ערכים ריקים.

חיזוי דירוג המסעדה:

ראשית, אנו מוסיפים תכונות על סמך הנתונים שהמשתמש הזין, כגון אחוז החרדים שיש באזור המסעדה שלו, כמה מסעדות יש באזור המסעדה וכו'. לאחר מכן, על מנת לחזות את דירוג המסעדה, אנו מריצים את עץ ההחלטה שיצרנו עם נתוני המסעדה החדשה. לבסוף כשמגיעים לעלה אנו מחזירים את הערך שרשום בו. במידה ויש ערך ריק בתכונה שהעץ משתמש בה בריצת האלגוריתם, אנו מחשבים את שני המסלולים של הצומת ולבסוף מחזירים ממוצע של שני ערכי העלים שמתקבלים.

תיאור המערכת:

המערכת שבנינו נכתבה בשפת *Python*. במערכת זו מבוצעים עיקר הדברים שציינו בחלק "תיאור הפתרון המוצע לבעיה".

המערכת מתבססת על בניית מסדי הנתונים, הרכבת מסד נתונים מרכזי, למידה, יצירת עץ החלטה וחיזוי דירוג המסעדה של הלקוח.

<u>בניית מסדי הנתונים:</u>

- Google – אספנו את הנתונים של גוגל בעזרת הספרייה *googleplaces*, על ידי מספר שאילתות כפי שתואר בחלק הקודם.

- Rest – אספנו את הנתונים מהאתר *rest.co.il* על ידי web scraping בעזרת הספרייה *selenium*.

- Gov – אספנו את הנתונים מהאתר *data.gov.il* על ידי בקשות HTTPS בעזרת הספרייה *urllib*.

- Cbs – הורדנו את הנתונים מהאתר *cbs.gov.il* וחילצנו את הנתונים כפי שתואר בחלק הקודם.

<u>הרכבת מסד נתונים מרכזי:</u>

חיברנו את כל מסדי הנתונים עם מסד הנתונים של גוגל כפי שתואר בחלק הקודם. נעזרנו בספרייה *textdistance* למדידת מרחק בין מילים ובספרייה *mpu* למדידת מרחק בין 2 נקודות גיאוגרפיות. בנוסף, השלמנו ערכים חסרים, חישבנו את הדירוג המשוקלל והסרנו תכונות כפי שתואר בחלק הקודם.

<u>למידה:</u>

בשלב זה למדנו את ההיפר-פרמטרים (יתואר בהמשך) הנותנים את תוצאות הדיוק הטובות ביותר. לאחר מכן, השתמשנו בהיפר-פרמטרים שמצאנו ביצירת המסווג (עץ רגרסיה).

<u>יצירת עץ רגרסיה:</u>

יצרנו אלגוריתם הבונה את עץ ההחלטה בהתאם למה שכתוב בחלק הקודם (שימוש במשקלים עבור תכונות עם ערכים חסרים). כמו כן, השתמשנו בחישוב MSE כפי שתואר.

<u>חיזוי דירוג המסעדה של הלקוח:</u>

יצרנו אלגוריתם המריץ את הנתונים של הלקוח על עץ ההחלטה שיצרנו, לבסוף מחזיר את הדירוג של המסעדה (כפי שתואר בחלק הקודם).

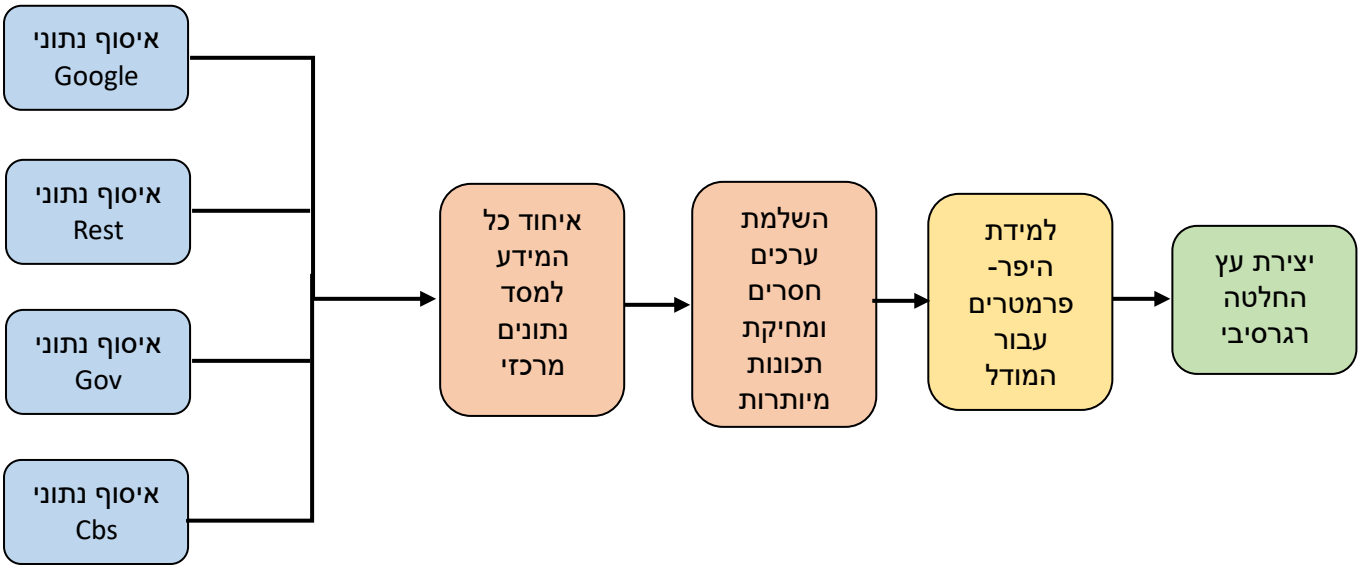
<u>אפשרויות נוספות:</u>

- כאשר בעל המסעדה ממלא את נתוני המסעדה שלו, הוא יכול לבחור באופציה מיוחדת של "מצא את סוג המסעדה המתאים ביותר". במצב זה, האפליקציה מריצה את עץ ההחלטה מספר פעמים, בכל פעם עם אפשרות אחרת לסוג המסעדה. לבסוף, מחזירה ללקוח את סוג המסעדה עם הציון הטוב ביותר (בהתאם לשאר הנתונים של הלקוח).

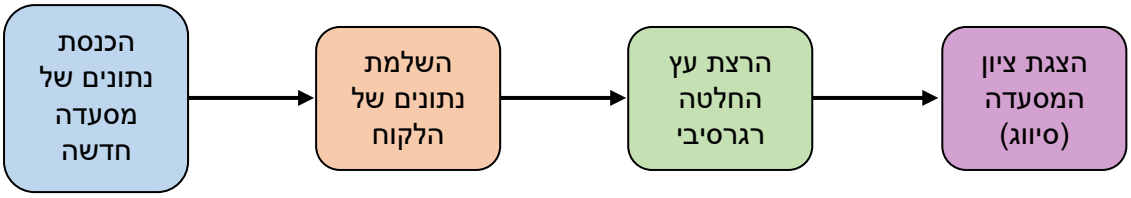
- כאשר בעל המסעדה מקבל את הציון של המסעדה שלו, הוא מקבל בנוסף מידע סביב אזור המסעדה שלו. למשל, מצב סוציו-אקונומי, אחוז הדתיים, מספר תחנות אוטובוס ועוד. כמו כן, האפליקציה מציגה לבעל המסעדה את מיקום דירוג המסעדה שלו ביחס למסעדות באזור וכן כמה מסעדות, עם סוג כמו שלו, יש באזור.

שרטוט המערכת:

<u>תהליך האימון:</u>



<u>תהליך החיזוי:</u>



תיאור הניסויים:

בשלב זה ביצענו ניסויים וכך למדנו את היפר-הפרמטרים הטובים ביותר עבור עץ ההחלטה שלנו. הניסויים שעשינו בוצעו על ידי *cross_validation*. בעזרת שיטה זו נוכל להעריך את הביצועים של המודל.

<u>היפר-הפרמטרים:</u>

1. min for pruning – המספר המינימלי של דוגמאות הנדרש לפיצול צומת פנימי. אם נגיע לצומת שיש בו מספר דוגמאות שקטן ממספר זה, נהפוך צומת זה לעלה בעץ.
   ערכים: [3, 10, 25, 50, 65].

2. max depth – העומק המירבי של העץ.
   ערכים: [3, 5, 8, 10, 12, 15].

3. min samples leaf – האחוז המינימלי של דוגמאות הנדרש כדי שצומת יהיה עלה בעץ. כלומר, תהיה נקודת פיצול בצומת מסוים רק אם בכל צד בפיצול (ימין ושמאל) יהיו לפחות האחוז הנ"ל ממספר הדוגמאות בצומת.
   ערכים: [0.1, 0.15, 0.2, 0.25].

<u>המדדים:</u>

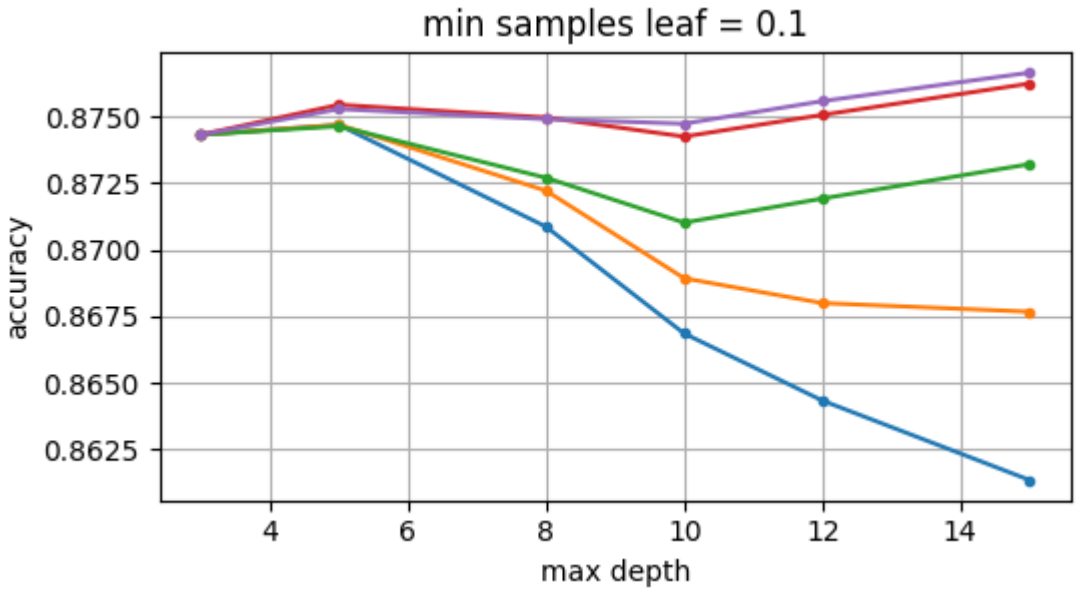
בדקנו את אחוז ההצלחה של תוצאות החיזוי עבור קבוצת המבחן ועבור קבוצת האימון. בדקנו זאת על ידי ממוצע שיערוך השגיאה של תוצאות החיזוי מהדירוג האמיתי. בעזרת מדד זה נוכל לדעת את טיב האלגוריתם (כלומר כמה עץ ההחלטה צודק עם היפר-הפרמטרים שנתונים לו). בנוסף, בדקנו את ערך ה- MSE (mean squared error) של תוצאות החיזוי עבור קבוצת המבחן ועבור קבוצת האימון. בעזרת מדד זה נוכל להעריך את ההכללה של המודל, וכן האם אנו מקבלים overfitting. במידה ונקבל ערכים גבוהים של MSE, נדע שהיפר-פרמטרים אלו לא טובים עבור המודל.

<u>ביצוע הניסויים:</u>

מכיוון שיש 3 היפר-פרמטרים חילקנו את הגרפים בצורה הבאה: לכל ערך של min samples leaf יצרנו גרף בפני עצמו, כאשר לכל גרף יש עקומה לכל ערך של min for pruning, לבסוף ערכי ציר x הם הערכים של max depth וערכי ציר y הם הערכים של המדדים (דיוקים או MSE).

צבעי העקומות של הערכים של min for pruning בכל הגרפים הם:

תוצאות הניסויים של דיוק תוצאות החיזוי עבור קבוצת המבחן:



min for pruning = 10

valid acc	max depth
0.874325	3
0.874699	5
0.87222	8
0.86893	10
0.867997	12
0.867672	15

min for pruning = 3

valid acc	max depth
0.874325	3
0.874693	5
0.870867	8
0.866854	10
0.864333	12
0.861346	15

min for pruning = 50

valid acc	max depth
0.874325	3
0.875453	5
0.874974	8
0.874256	10
0.87507	12
0.876249	15

min for pruning = 25

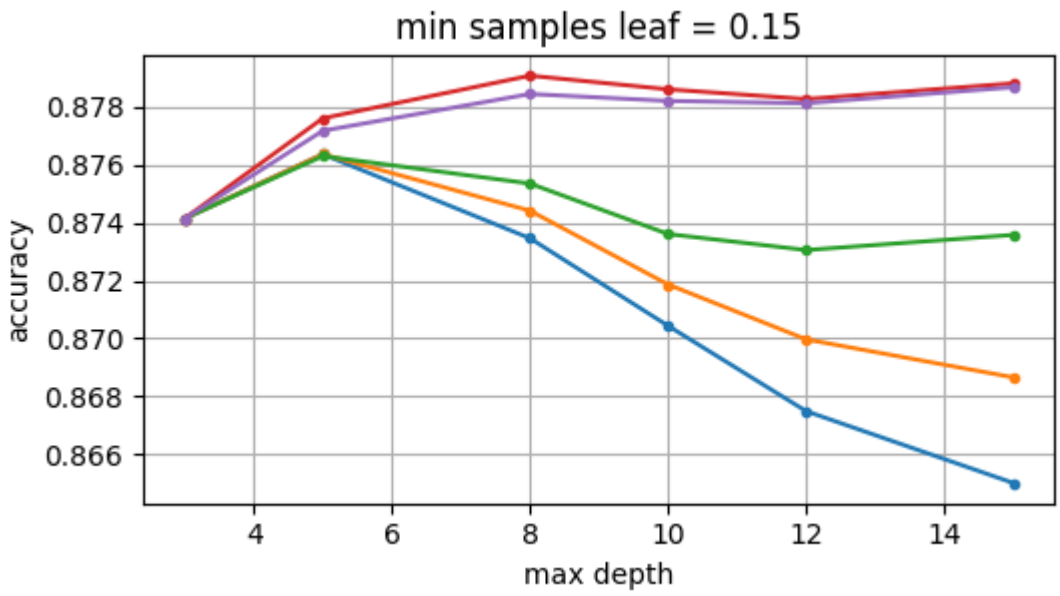
valid acc	max depth
0.874325	3
0.874645	5
0.872707	8
0.87102	10
0.871925	12
0.873219	15

היפר-פרמטרים הטובים ביותר

Min samples leaf = 0.1
Min for pruning = 65
Max depth = 15
Valid accuracy = 0.8766

min for pruning = 65

valid acc	max depth
0.874325	3
0.87529	5
0.874913	8
0.87474	10
0.875587	12
0.876658	15



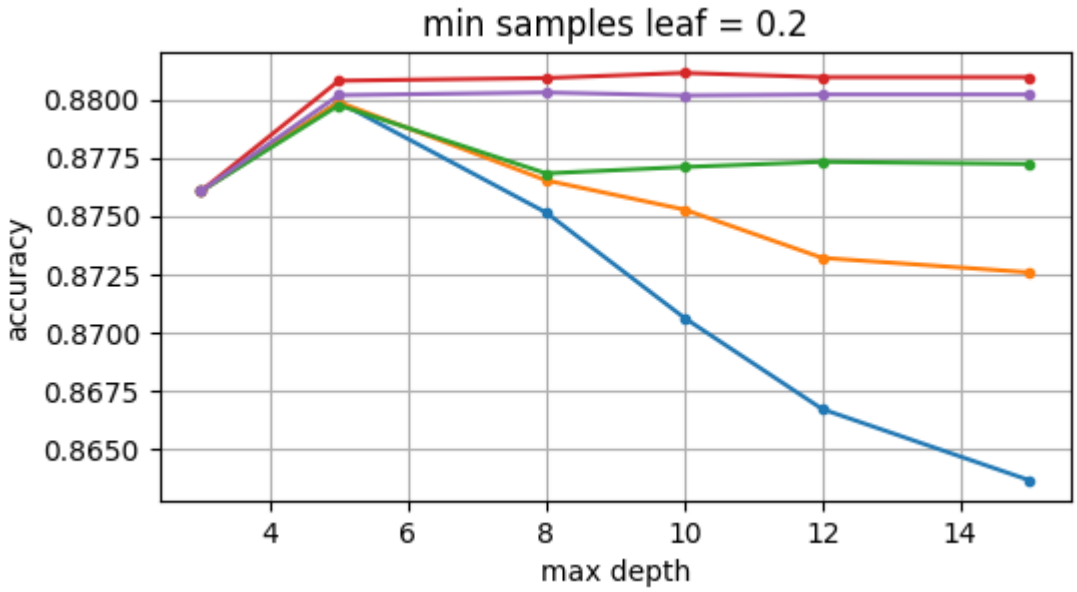
min for pruning = 10	
valid acc	max depth
0.874144	3
0.876359	5
0.87441	8
0.871862	10
0.869967	12
0.868654	15

min for pruning = 3	
valid acc	max depth
0.874144	3
0.876359	5
0.873457	8
0.870435	10
0.867487	12
0.865007	15

min for pruning = 50	
valid acc	max depth
0.874144	3
0.8776	5
0.879072	8
0.8786	10
0.878264	12
0.878806	15

min for pruning = 25	
valid acc	max depth
0.874144	3
0.876295	5
0.875347	8
0.873604	10
0.873051	12
0.873574	15

היפר-פרמטרים הטובים ביותר	min for pruning = 65	
Min samples leaf = 0.15 Min for pruning = 50 Max depth = 8 Valid accuracy = 0.879	valid acc	max depth
	0.874144	3
	0.877169	5
	0.878439	8
	0.8782	10
	0.87812	12
	0.878673	15



min for pruning = 10

valid acc	max depth
0.876066	3
0.879884	5
0.876534	8
0.875284	10
0.873213	12
0.872593	15

min for pruning = 3

valid acc	max depth
0.876066	3
0.879884	5
0.875157	8
0.870652	10
0.86674	12
0.863692	15

min for pruning = 50

valid acc	max depth
0.876066	3
0.880799	5
0.880914	8
0.881132	10
0.88095	12
0.88095	15

min for pruning = 25

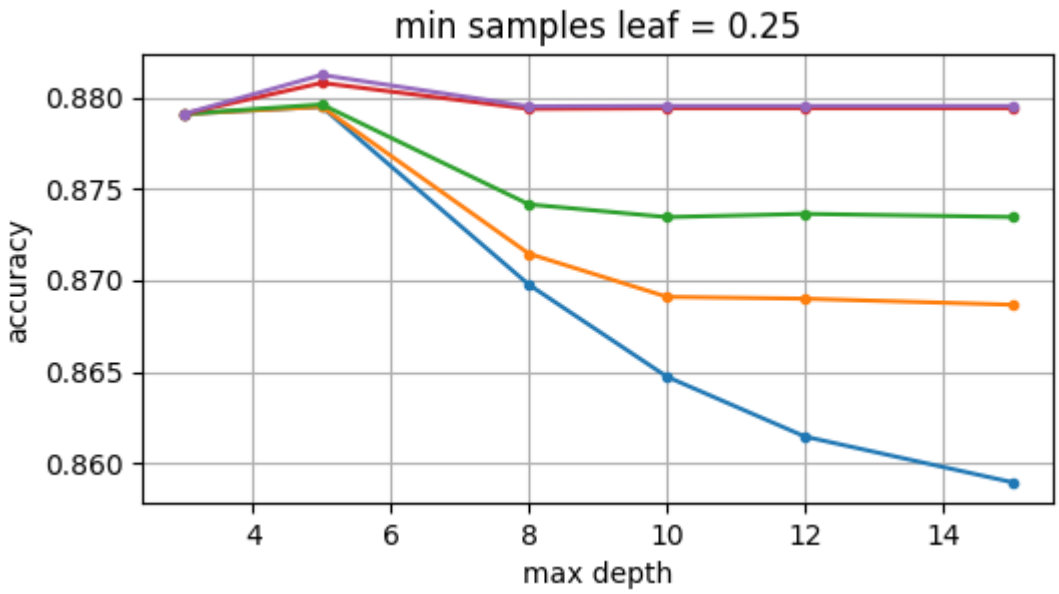
valid acc	max depth
0.876066	3
0.879748	5
0.876831	8
0.877105	10
0.877327	12
0.877235	15

היפר-פרמטרים הטובים ביותר

Min samples leaf = 0.2
Min for pruning = 50
Max depth = 10
Valid accuracy = 0.881

min for pruning = 65

valid acc	max depth
0.876066	3
0.880181	5
0.880294	8
0.880168	10
0.880209	12
0.880209	15



min for pruning = 10	
valid acc	max depth
0.879099	3
0.879486	5
0.871458	8
0.869102	10
0.869001	12
0.868671	15

min for pruning = 3	
valid acc	max depth
0.879099	3
0.879486	5
0.869765	8
0.864734	10
0.861462	12
0.85897	15

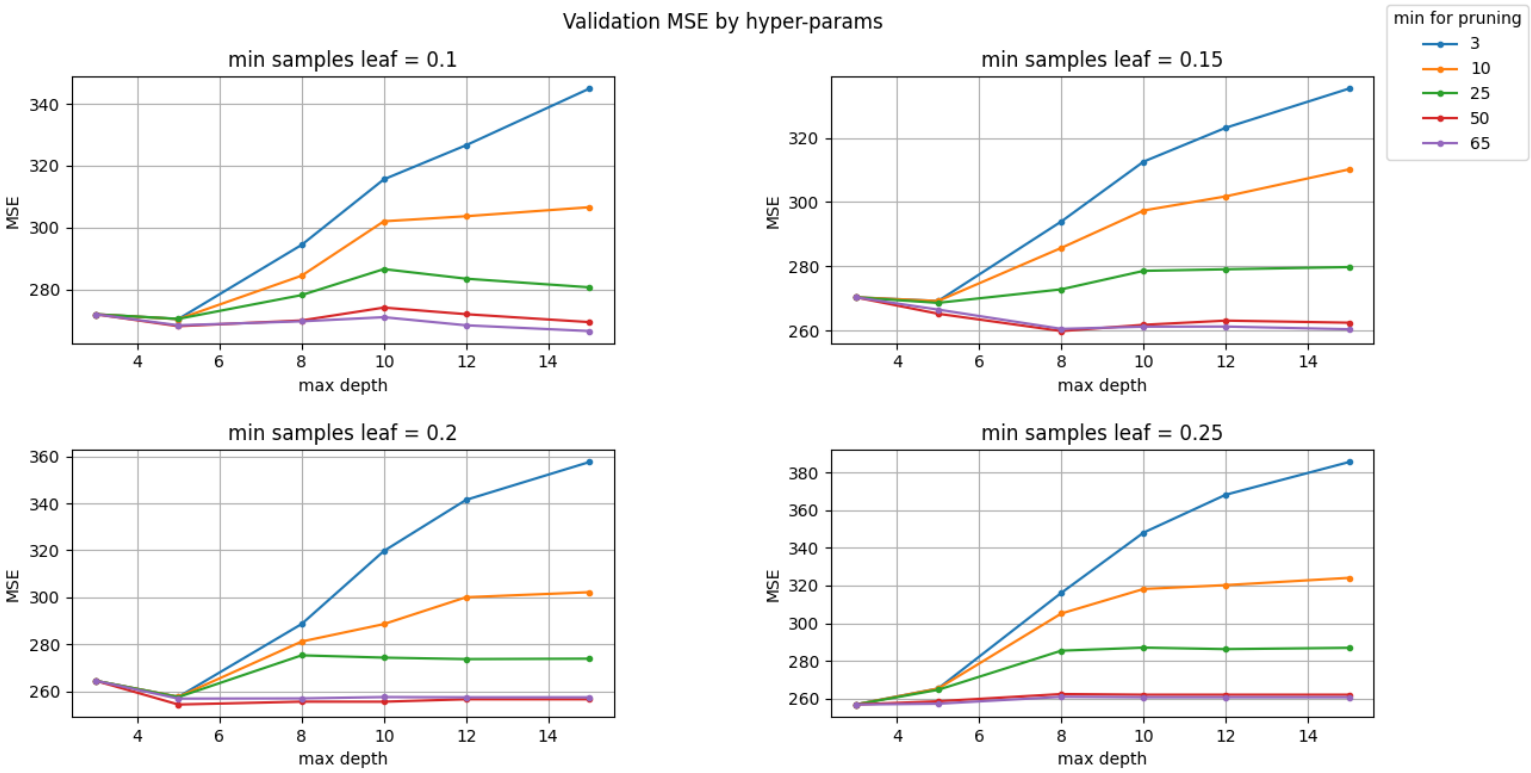
min for pruning = 50	
valid acc	max depth
0.879099	3
0.880795	5
0.879378	8
0.879411	10
0.879411	12
0.879411	15

min for pruning = 25	
valid acc	max depth
0.879099	3
0.879619	5
0.874155	8
0.87346	10
0.873638	12
0.873466	15

היפר-פרמטרים הטובים ביותר
Min samples leaf = 0.25 Min for pruning = 65 Max depth = 5 Valid accuracy = 0.881

min for pruning = 65	
valid acc	max depth
0.879099	3
0.88123	5
0.879513	8
0.879538	10
0.879538	12
0.879538	15

תוצאות הניסויים של ערכי MSE עבור קבוצת המבחן:



תוצאות הניסויים עבור קבוצת המבחן:

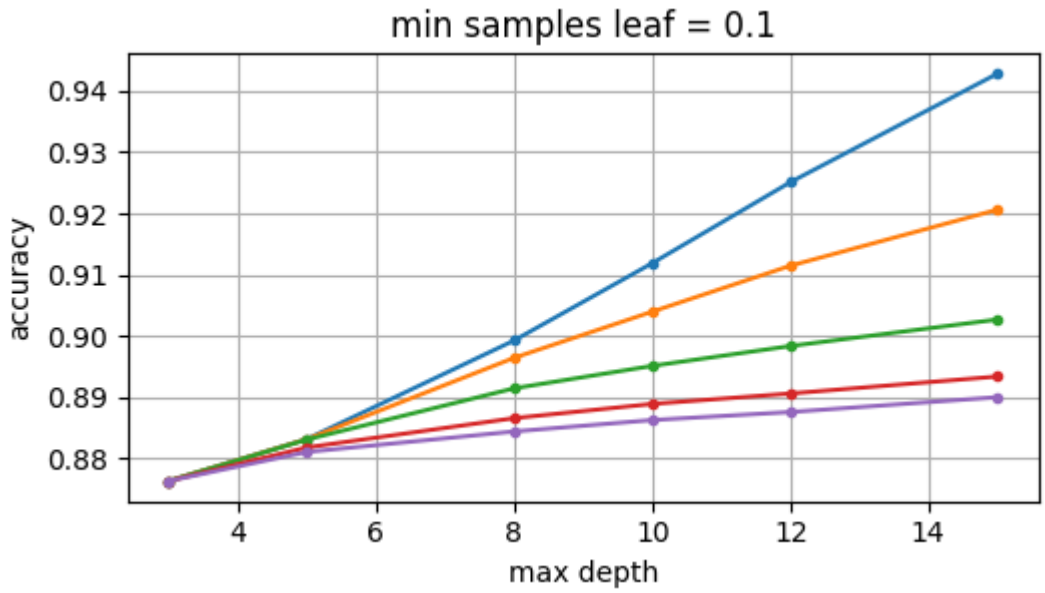
היפר-הפרמטרים הטובים ביותר שקיבלנו בסך הכל הם:

Min samples leaf = 0.2	**Min samples leaf = 0.25**
Min for pruning = 50	**Min for pruning = 65**
Max depth = 10	**Max depth = 5**

קיבלנו 2 אופציות של היפר-פרמטרים (הדיוק שלהם קרוב מאוד עד כדי מאית אחוז). אנו נבחר את האופציה עם עומק העץ הגבוה יותר על מנת לקבל עץ יותר מורכב, עם יותר עלים, יותר חיזויים. מכיוון שאנו בונים עץ רגרסיה אנו מעוניינים ביותר עלים כדי לקבל שונות גדולה יותר.

נשים לב שככל שאנו מגדילים את היפר-הפרמטר min for pruning אנו מקבלים תוצאות טובות יותר, כלומר אם גודמים ענפים בשלבים מוקדמים יותר בתהליך בניית העץ מקבלים אחוז דיוק גבוה יותר. עבור ערכים קטנים אנו מקבלים overfitting כי מנסים להגיע לעלים עם מעט דוגמאות בעוד עבור ערכים גבוהים אנו מתגברים על רעשים ומקבלים חלוקה טובה יותר. בנוסף, קיבלנו שככל שאנו מגדילים את היפר-הפרמטר max depth בשילוב עם min for pruning נמוך יחסית אנו מקבלים ירידה באחוז הדיוק. כמו קודם, יש נטייה לקבל overfitting כאשר עומק העץ גבוה ועלים עם מספר דוגמאות קטן יחסית.

תוצאות הניסויים של דיוק תוצאות החיזוי עבור קבוצת האימון:



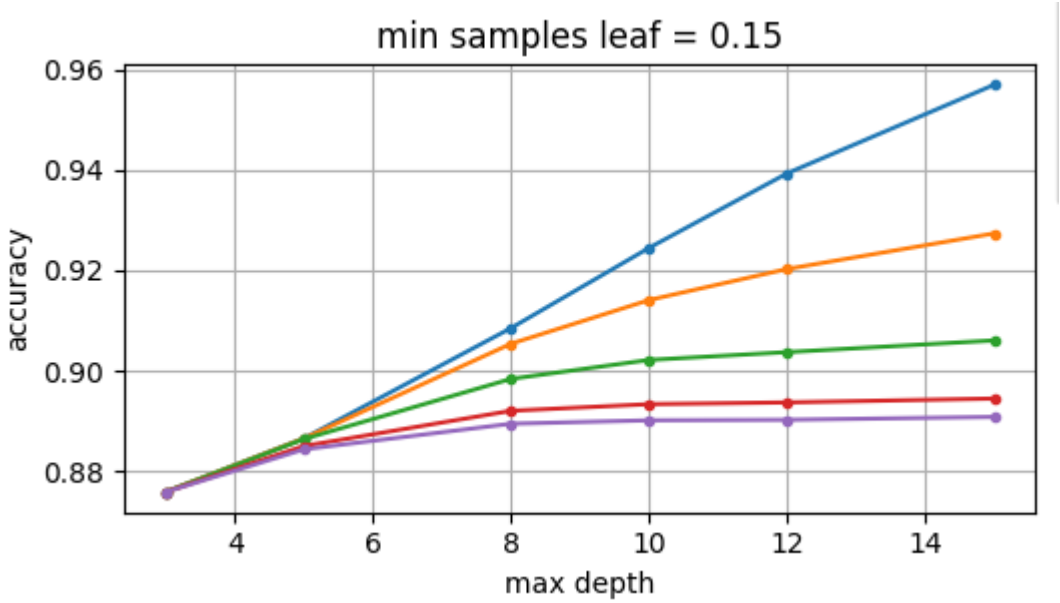
min for pruning = 10	
train acc	max depth
0.876367	3
0.883186	5
0.896452	8
0.903985	10
0.911477	12
0.920625	15

min for pruning = 3	
train acc	max depth
0.876367	3
0.883193	5
0.899283	8
0.911891	10
0.925102	12
0.942817	15

min for pruning = 50	
train acc	max depth
0.876367	3
0.881839	5
0.886616	8
0.888938	10
0.890632	12
0.89341	15

min for pruning = 25	
train acc	max depth
0.876367	3
0.883122	5
0.891478	8
0.895138	10
0.898381	12
0.902709	15

היפר-פרמטרים הטובים ביותר	min for pruning = 65	
Min samples leaf = 0.1 Min for pruning = 3 Max depth = 15 Valid accuracy = 0.942	train acc	max depth
	0.876367	3
	0.881099	5
	0.884444	8
	0.886305	10
	0.887613	12
	0.890035	15



min for pruning = 10	
train acc	max depth
0.875822	3
0.886496	5
0.905378	8
0.914094	10
0.920287	12
0.927385	15

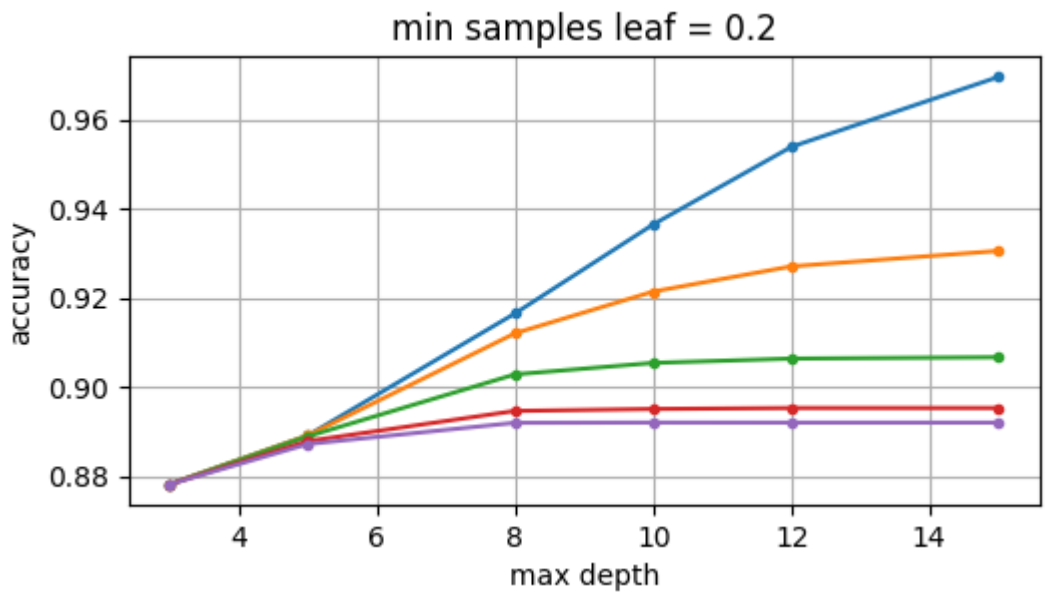
min for pruning = 3	
train acc	max depth
0.875822	3
0.886496	5
0.908498	8
0.924497	10
0.939363	12
0.957036	15

min for pruning = 50	
train acc	max depth
0.875822	3
0.884977	5
0.892011	8
0.893332	10
0.893682	12
0.89445	15

min for pruning = 25	
train acc	max depth
0.875822	3
0.886366	5
0.898371	8
0.90215	10
0.903708	12
0.906039	15

היפר-פרמטרים הטובים ביותר
Min samples leaf = 0.15 Min for pruning = 3 Max depth = 15 Valid accuracy = 0.957

min for pruning = 65	
train acc	max depth
0.875822	3
0.884321	5
0.889435	8
0.890114	10
0.890216	12
0.890792	15



min for pruning = 10

train acc	max depth
0.87825	3
0.889175	5
0.912072	8
0.921411	10
0.92706	12
0.930532	15

min for pruning = 3

train acc	max depth
0.87825	3
0.889175	5
0.916517	8
0.936445	10
0.953819	12
0.969499	15

min for pruning = 50

train acc	max depth
0.87825	3
0.887914	5
0.894678	8
0.895115	10
0.89533	12
0.89533	15

min for pruning = 25

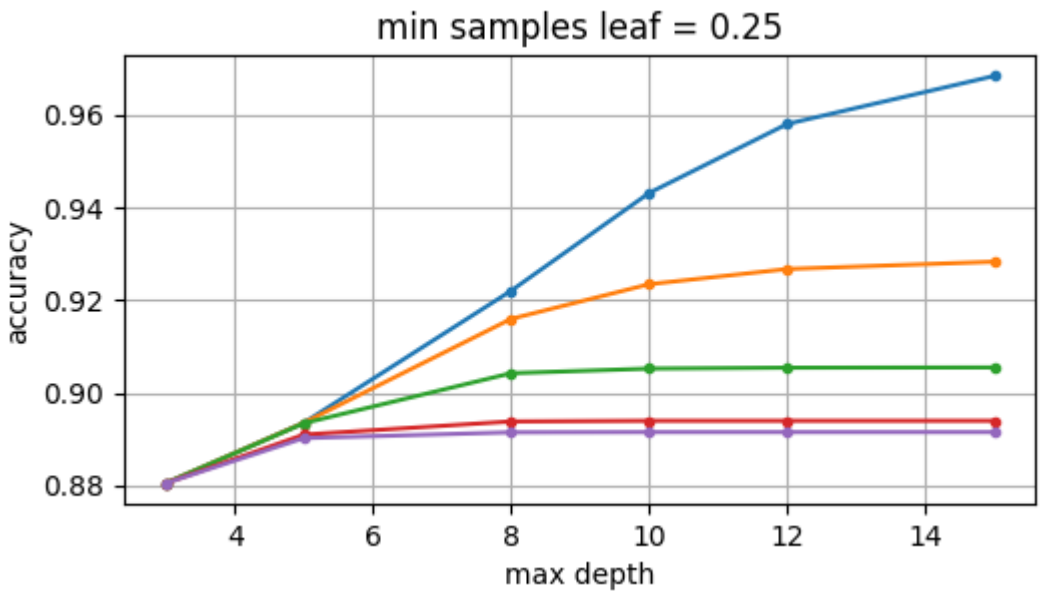
train acc	max depth
0.87825	3
0.888977	5
0.902868	8
0.905434	10
0.906402	12
0.90671	15

היפר-פרמטרים הטובים ביותר

Min samples leaf = 0.2
Min for pruning = 3
Max depth = 15
Valid accuracy = 0.969

min for pruning = 65

train acc	max depth
0.87825	3
0.887265	5
0.892025	8
0.892078	10
0.892065	12
0.892065	15



min for pruning = 10	
train acc	max depth
0.88045	3
0.8935	5
0.915937	8
0.923435	10
0.926721	12
0.928295	15

min for pruning = 3	
train acc	max depth
0.88045	3
0.8935	5
0.922002	8
0.943137	10
0.958046	12
0.968422	15

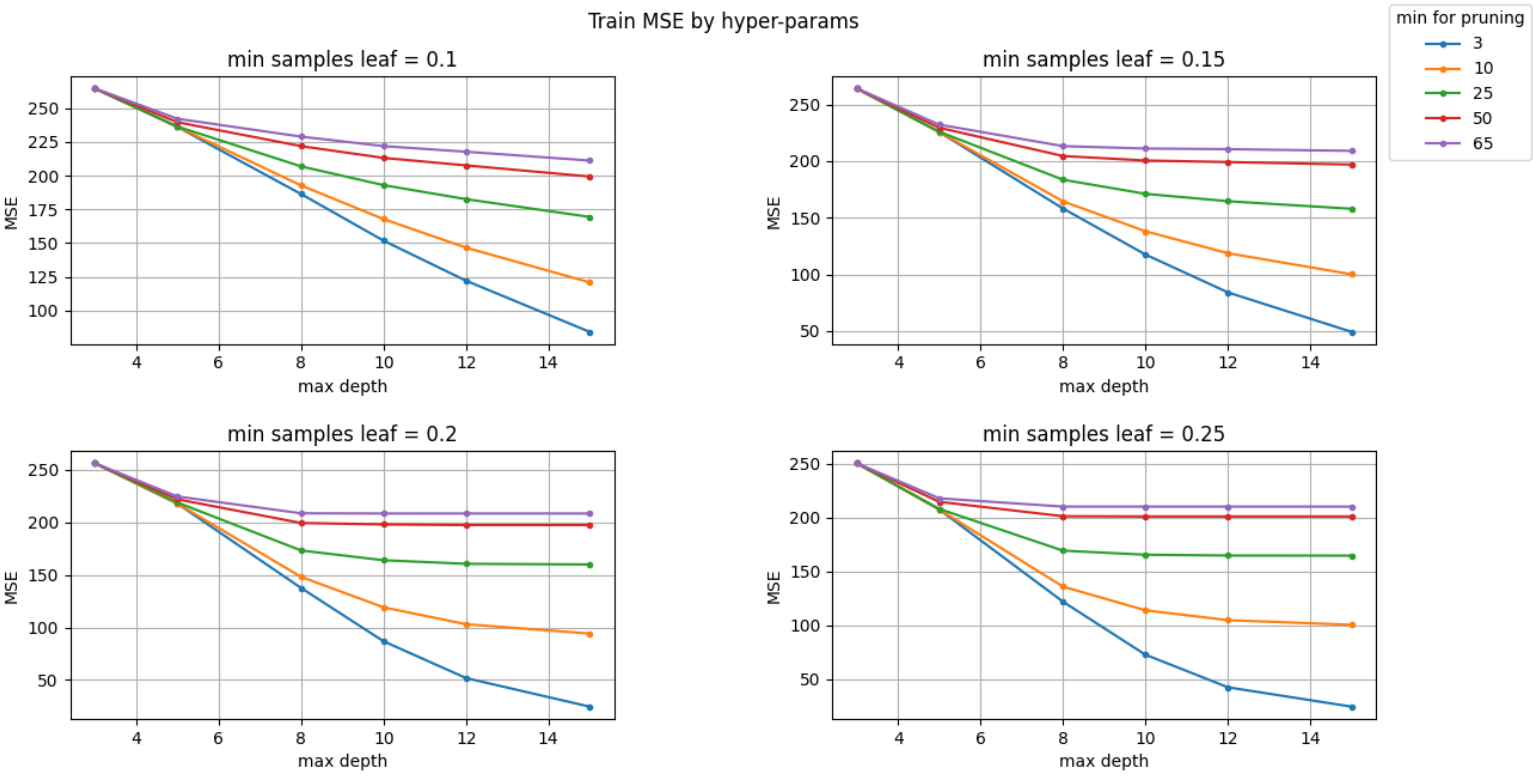
min for pruning = 50	
train acc	max depth
0.88045	3
0.890949	5
0.893799	8
0.893939	10
0.893939	12
0.893939	15

min for pruning = 25	
train acc	max depth
0.88045	3
0.893406	5
0.90421	8
0.905213	10
0.905406	12
0.905465	15

היפר-פרמטרים הטובים ביותר
Min samples leaf = 0.25 Min for pruning = 3 Max depth = 15 Valid accuracy = 0.968

min for pruning = 65	
train acc	max depth
0.88045	3
0.890199	5
0.891448	8
0.891496	10
0.891496	12
0.891496	15

תוצאות הניסויים של ערכי MSE עבור קבוצת האימון:



תוצאות הניסויים עבור קבוצת האימון:

היפר-הפרמטרים הטובים ביותר שקיבלנו בסך הכל הם:

Min samples leaf = 0.2

Min for pruning = 3

Max depth = 15

נשים לב שככל שאנו מגדילים את היפר-הפרמטר max depth אנו מקבלים תוצאות טובות יותר. מכיוון שככל שעומק העץ גבוה יותר כך אנו מקבלים יותר overfitting, ולכן כאשר אנו בודקים עבור קבוצת האימון אנו מקבלים אחוזי דיוק גבוהים יותר. בנוסף, ככל שהיפר-הפרמטר min for pruning קטן יותר כך אחוזי הדיוק גבוהים יותר. כמו קודם, אנו מקבלים פחות דוגמאות בעלים ולכן overfitting גבוה יותר.

המסקנה שלנו בסוף חלק זה היא שהמסווג עם היפר-הפרמטרים הללו:

Min samples leaf = 0.2

Min for pruning = 50

Max depth = 10

נותן את התוצאות הטובות ביותר (88% דיוק) מבין שאר האפשרויות ולכן אלו היפר-הפרמטרים שאיתם נעבוד בבניית עץ ההחלטה שלנו.

סיכום:

מטרתנו בפרויקט זה הייתה ללמוד מסווג עץ רגרסיה אשר חוזה דירוג מסעדה חדשה. במהלך הפרויקט התמודדנו עם בניית מסדי נתונים מכמה מאגרי מידע שונים ומיזוגם למסד נתונים אחד גדול. בנוסף, בחרנו לסנן תכונות עם חשיבות נמוכה יותר ולהשלים מידע חסר על סמך מידע כללי על התכונות ובעזרת שיטות נוספות. כמו כן, למדנו את המסווג שבעזרתו אנו חוזים את הדירוג של המסעדה החדשה.

<u>קשיים במהלך הפרויקט:</u>

במהלך הפרויקט גילינו קשיים הן בשלבים הראשונים (בניית מסד הנתונים) והן בשלבים המתקדמים (יצירת המסווג):

- הבאת הנתונים דרך גוגל הייתה מאתגרת מכיוון שלכל שאילתה קיבלנו לכל היותר 60 מסעדות לכן היינו צריכים לפתור זאת על ידי מספר שאילתות ולאפיין זאת על ידי אזורים גיאוגרפיים.
- איחוד כל מקורות המידע למסד נתונים אחד גדול. לכל מקור מידע היה מזהה ייחודי משלו ללא קשר למזהים הייחודיים של המקורות האחרים (כאשר אפילו השם והכתובת לא בהכרח זהים במקורות שונים), לכן היינו צריכים למצוא פתרון איך לחבר את כולם למסד נתונים אחד. פתרנו זאת על ידי מיקום גיאוגרפי או על ידי שם מסעדה דומה וכתובת מקורבת.
- נתקלנו במסעדות עבורן היה מספר מדרגים נמוך, וקשה להסתמך על דירוג זה (לא נרצה להסתמך על חוויה ספציפית אלא על מכלול של דעות). רצינו לתת למסעדות אלו פחות חשיבות מאשר מסעדות עם הרבה מדרגים. פתרנו זאת על ידי חישוב ציון המסעדה עם פונקציה שמשקללת את מספר המדרגים ביחד עם הדירוג של המסעדה וכך קיבלנו סיווגים מדויקים יותר ומגוונים יותר.
- ערכים ריקים במסד הנתונים שלנו גרם לקושי בהרצת האלגוריתם (בניית העץ), השתמשנו במשקלים בזמן בניית העץ כדי להתאים את האלגוריתם לתכונות עם ערכים ריקים וכך ניתן לקבוע אילו שאלות מתאימות יותר לצמתים.

<u>כיוונים להמשך המחקר:</u>

- שימוש במסווגים נוספים – בפרויקט התמקדנו בעיקר במסווג של בניית עץ רגרסיה. ייתכן וקיימים מסווגים אחרים שיתנו תוצאות מדויקות יותר או בעל ביצועים גבוהים יותר מאשר המסווג שבו השתמשנו. לכן, רצוי מאוד בהמשך המחקר לבחון מסווגים אחרים ולבדוק אותם מול מסד הנתונים שלנו ולהשוות מול המסווג שהשתמשנו בו.
- שימוש באזורים גיאוגרפים נוספים – בפרויקט בחרנו לבנות את מסד הנתונים על בסיס אזור חיפה והסביבה. אבל, על ידי שינויים מינוריים ניתן להשתמש באזורים נוספים. כך נוכל לבנות מסד נתונים גדול יותר או לבנות לכל אזור את מסד הנתונים שלו וליצור לו מסווג אופטימלי לאזור שלו בלבד. בשני המקרים נשפר את יכולות הפרויקט ונקבל פרספקטיבה רחבה יותר.
- הכנסת תפריט למסד הנתונים – בפרויקט יש לנו תכונה של סוג המסעדה. אך תכונה זו כללית מידי ולא יורדת לפרטים של איזה סוגי מנות נוכל למצוא במסעדה. ניתן להשתמש במקור נתונים נוסף וכך להרחיב את מסד הנתונים שלנו לתכונות נוספות (למשל במסעדה בשרית נוכל לדעת האם מגישים קבב, פרגיות, המבורגר, סטייקים ו/או כנפיים וכו). בנוסף, נוכל לקבל היבט נוסף על מחירי המנות ולתת להם משמעות נוספת במסד הנתונים המרכזי. התמודדות נוספת במקרה זה היא חילוץ התפריט מאתרי מסעדות, כלומר יש הרבה מסעדות שהתפריט מופיע כתמונה או קובץ pdf לכן נאלץ לחלץ את התפריט ממנו (ניתן לעשות זאת למשל על ידי deep learning).

- שימוש בביקורות על מסעדה – לכל מסעדה שאנו מקבלים מגוגל אנו יכולים לחלץ את הביקורות שהשאירו לקוחות. בעזרת הביקורות הללו אנו יכולים לחלץ תכונות נוספות על המסעדה, למשל, האם מגישים מנות ללא גלוטן? האם יש כיסאות לתינוקות? רמת הניקיון? שירות העובדים? וכו'. אומנם המסעדן שישתמש באפליקציה לא יכול לדעת את התשובות לתכונות הללו, אבל הוא יכול לראות כמה תכונות אלו ישפיעו על דירוג המסעדה שלו ויוכל לשנות אותן בהתאם.

ביבליוגרפיה:

תוך חצי שנה: ממסעדה לא כשרה למסעדה כשרה - ובחזרה – כתבה מאתר ynet.

שגב משנה קונספט: ממסעדה כשרה למסעדה לא כשרה – כתבה מאתר ynet.

https://www.google.com/maps - Google Maps.