פרויקט סיום בקורס "תכנות מדעי בשפת פייתון

מגישה: עדי יערי

ת.ז: 302704752

קישור לסרטון הסבר:

https://drive.google.com/drive/folders/1d68VesL3l6SpaUP6eSVpeA_56GcRLvUb?usp=sharing

הדאטה סט שקיבלתי מכיל ממוצע דירוגים של משתמשים עבור קטגוריות שונות.
שלוש העמודות הראשונות הן מאפייני המשתמש; קבוצת גיל, סטטוס ומין.
שאר העמודות מכילות את הקטגוריות מקומות פנאי.
העמודה שמכילה דירוג מוזיאונים היא עמודת ה target ומייצגת את הדירוג האחרון שניתן ע"י משתמש עבור קטגוריה זו.

Initial data analysis

1. קראתי את קובץ ה csv והכנסתי את המידע לתוך data frame.
בהדפסת ()info אפשר לראות:
- 24 עמודות; 4 עמודות של object 19 עמודות של float ועמודה אחת של int שזו עמודת ה target.
- 5456 שורות והאינדקסים הם 0-5455
- אפשר לראות שיש ערכי non-null כי המספרים בעמודה זו לא זהים.
2. שינוי שמות הקטגוריות לצרכי נוחות
3. בהדפסת ()isnull().sum נקבל סיכום של ערכי הnon-null בכל עמודה.
כמובן שצריך לטפל בערכים אלו אבל יש יותר מדי כדי שפשוט נמחק אותם.
- מחיקת כל השורות עם ערכי non-null בעמודות המאפיינות את המשתמשים: Gender, Profile Age.
ישנן 3 אפשרויות לטיפול ב Gender:
  - מחיקת non והשארת ?
  - החלפת כל ה non ל ?
  - כל ? השאלה ל non

קיימים 279 ערכי non—null ו303 ערכי ? בעמודת ה gender.
אני חושבת שלמחוק כל כך הרבה נתונים ישפיע לרעה על התוצאות ולכן בחרתי למחוק רק את ערכי ה non-null ולהתייחס ל ? כקטגוריה שלישית במין.

טיפול בעמודת Marital Status: לאחר הטיפול בעמודות Gender ו Profile Age גיליתי שבלי מאמץ טיפלתי גם בעמודת Marital Status ומספר השורות שווה בשלוש העמודות (5153)

- המרת כל ערכי ה non-null להיות הממוצע של העמודה המתאימה.
  - המרת עמודת ה Local Services מ object ל float; נתקלתי בשגיאה וגיליתי שקיים ערך אחד ששווה ל 't2\2.' אני מבינה שזו טעות בודדת ולכן בוחרת לשנות אותו ל '2.2'. הפעלתי פה שיקול דעת; יכלתי למחוק אותו ואני מאמינה שלא היה משנה, 22.0 לא תואם את שאר הדירוגים שבין 0 ל 5 ולכן החלטתי ש 2.2 הכי מתאים להיות.

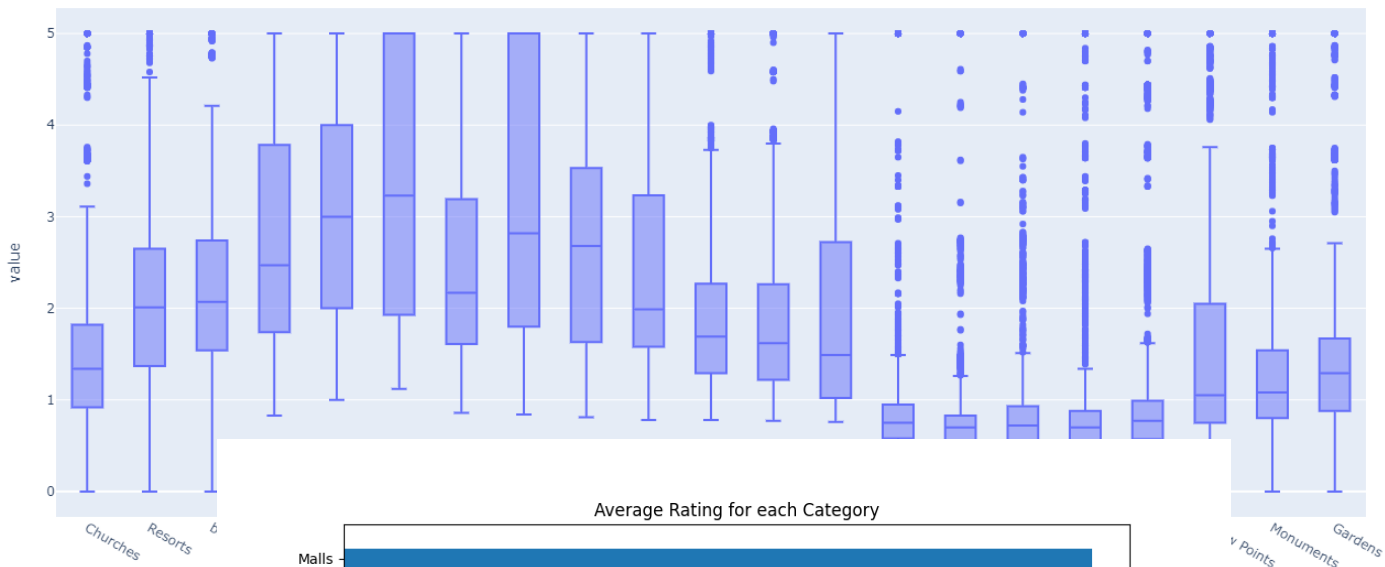
4. המרת כל הערכים הקטגוראלים למספריים.

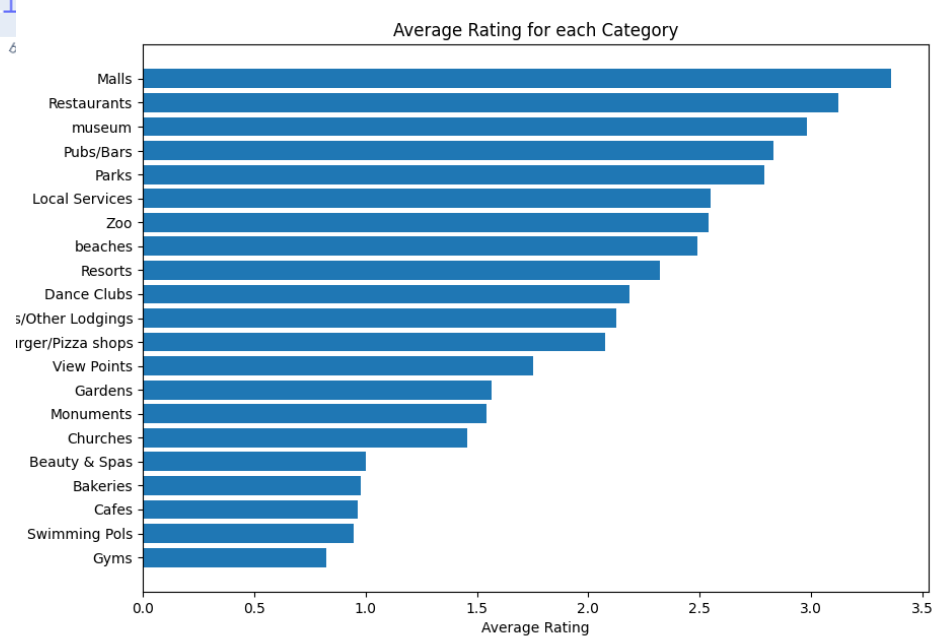
```
df['Profile Age'] = df["Profile Age"].replace({'<5': 0, '5-10': 1, '>10': 2})
df['Gender'] = df['Gender'].replace({'male': 0, 'female': 1, '?': 2})
df['Marital Status'] = df['Marital Status'].replace({'Single': 0, 'Married': 1})
```

5. טיפול בערכי קיצון:
ע"י התבוננות בתוצאות describe() ניתן לראות שבכל עמודות הדירוגים, הממוצע הוא בין 0 ל 5 ורק בעמודת Resorts יש ממוצע 6.16.
בדקתי כמה ערכים מעל 5 וגיליתי מספר קטן (9) ולכן בחרתי לשנות אותם לממוצע (2.32)

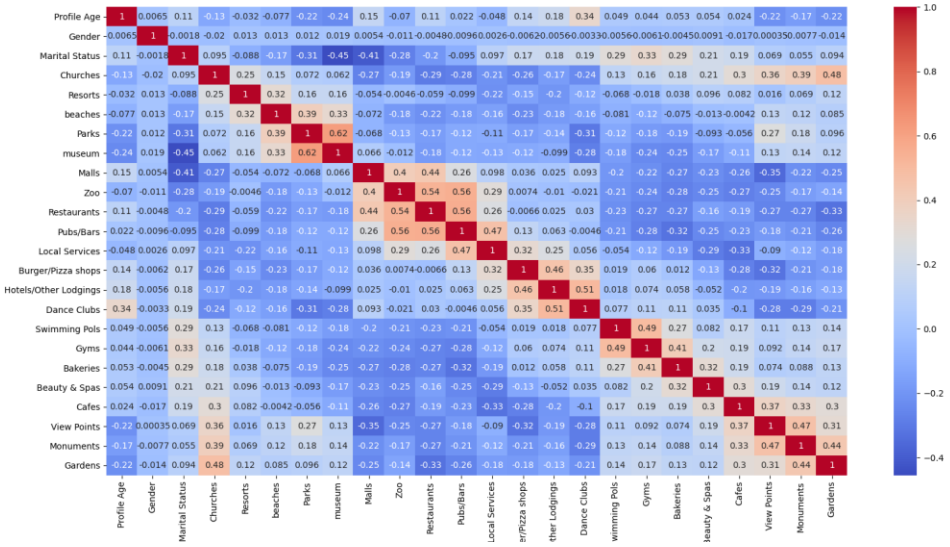
Exploratory data Analysis

1. שני גרפים שמראים את ממוצע הדירוגים.
אפשר לראות ש Gym קיבל את הדירוג הנמוך ביותר ו malls קיבל את הדירוג הגבוה ביותר.





2. בחרתי להראות את הקשר בין כל העמודות ע"י heatmap ולא גרף אחר בגלל שיש המון מידע ויותר קל להסתכל על כל כך הרבה מידע בטבלה מסודרת.
אפשר לראות שיש קשר חזק בין:
- Parks – museum
- Pubs/Bars – Zoo
- Pubs/Bars – Restaurants
- Zoo – Restaurants
- Hotels/Other Lodgings - Dance Clubs
- Gyms - Beauty & Spas

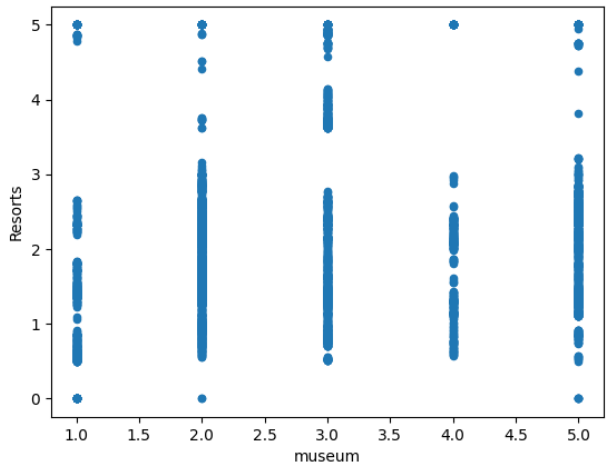


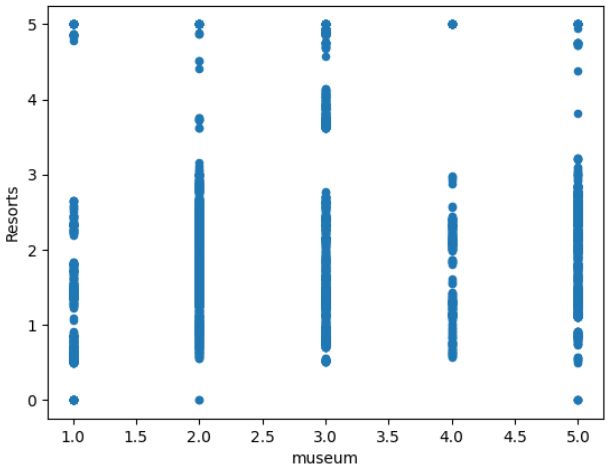
בחרתי בכל זאת לראות את הקשר בין עמודות ספציפיות:

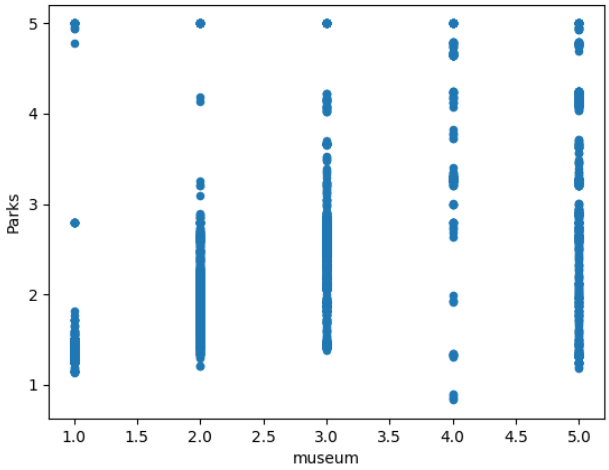
3. Boxplots בין מוזיאונים (target) למאפיינים המשתמשים
Scatterplots – בין מוזיאונים לכל הקטגוריות;

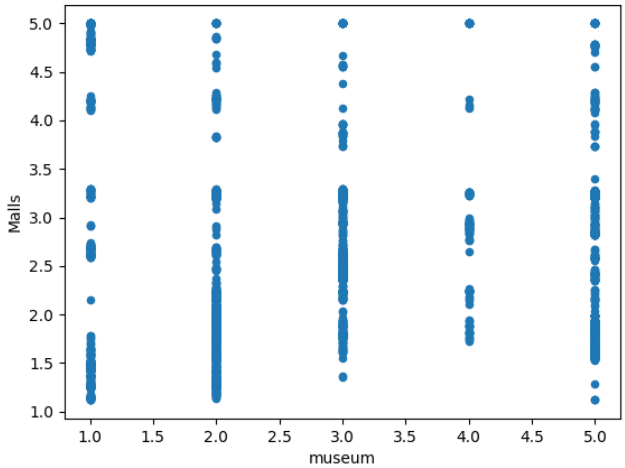
לדוגמה אפשר לראות:

- אנשים שלא דירגו את resorts מעל 3, לא דירגו את museum מעל 3
- אנשים שדירגו את museum 4 לא נתנו מעל 3 ל beaches או מתחת 0.75
- רוב האנשים שדירגו מעל 2 parks לא יופיעו בדירוג 1 של museum.
- אנשים שדירגו malls מעל 3 לרוב לא ידרגו את museum בציון 4.





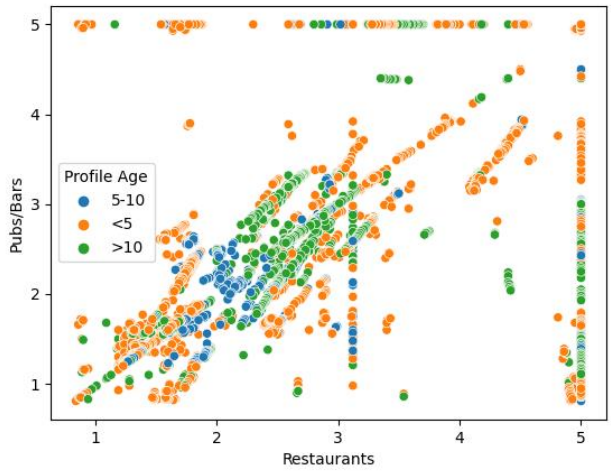


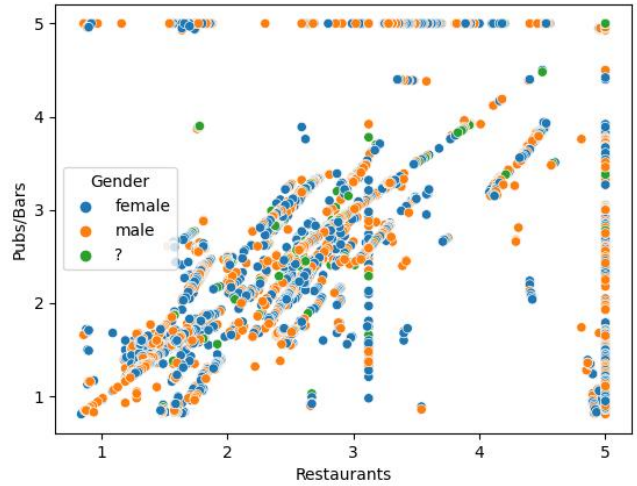


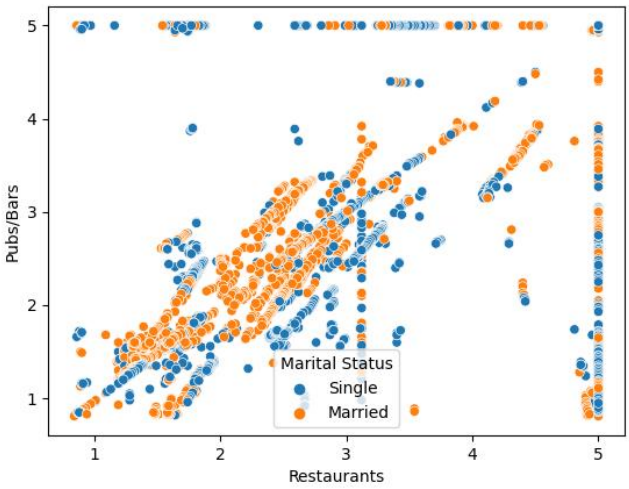
4. הראיתי את הקשר בין קטגוריות שראינו שיש בניהן קשר חזק ועם המאפיינים השונים של המשתמשים.

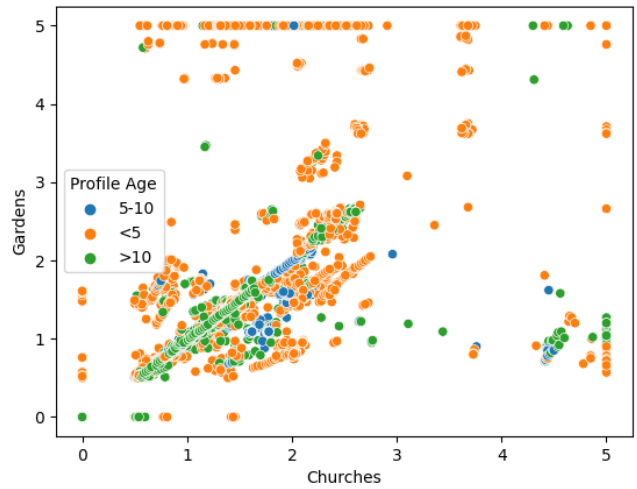
לדוגמה:

- אפשר לראות שיש קשר בין שכבת הגיל 5-10 לבין דירוג נמוך בין 1-3 של פאבים ומסעדות.
- אפשר לראות קשר לינארי בין גברים שדירגו פאבים, מסעדות.
- אפשר לראות שרוב המדרגים באים ומסעדות מתחת לציון 3, נשואים.
- ישנו קשר לינארי בין דירוג גנים וכנסיות ואפשר לראות שרוב הדירוגים של שכבת גיל 5-10 הם דיי נמוכים בסביבת ה 1-2.
שכבת הגיל מעל 10 – קשר לינארי ברור; דירוג הגנים והכנסיות הוא בהתאמה.


Profile Age
5-10
<5
>10
Pubs/Bars
Restaurants


Gender
female
male
?
Pubs/Bars
Restaurants

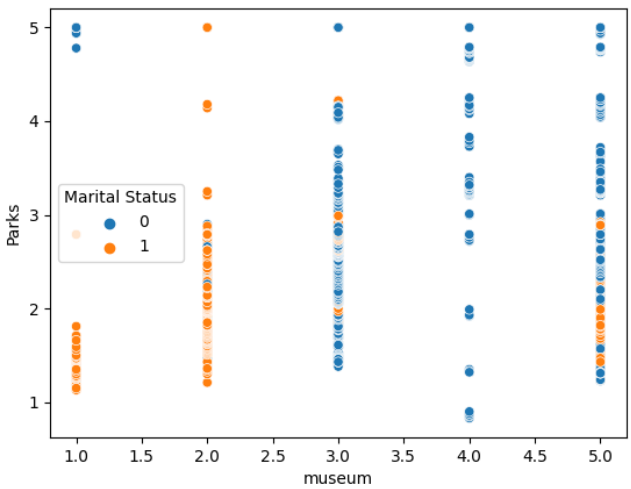

Marital Status
Single
Married
Pubs/Bars
Restaurants

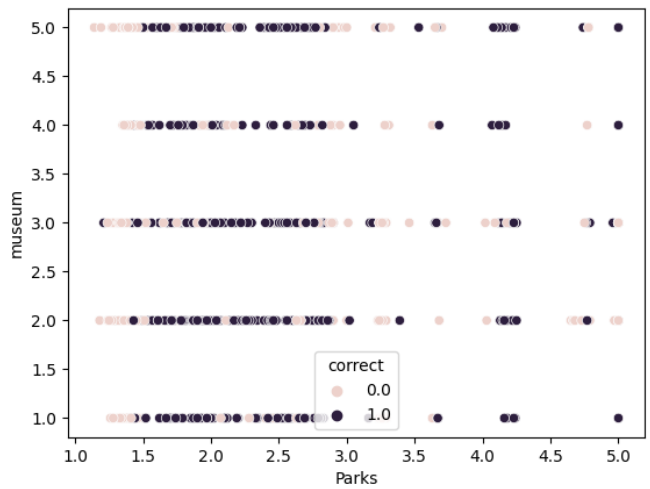

Profile Age
5-10
<5
>10
Gardens
Churches

Classification Model

1. הרצת Naiv Base על 4 זוגות פיצ'רים שבחרתי לפי גרפים שמראים שיש חלוקה טובה לקבוצות.
והשוואת ה accuracy של המודל.
בנוסף הצגתי גרף שמראה את הקיטלוג הנכון של המודל והצגתי טבלה עם הדירוג המקורי,
החיזוי והאם המודל צדק או לא והדפסתי את מספר הקיטלוגים הנכונים לפי כל ערך.
לפי התוצאות בחרתי בשני פיצ'רים: parks – Marital status.
אחוז הדיוק היה הגבוה ביותר: 58.26%.
ראה מסמכים בתקיית choose2features

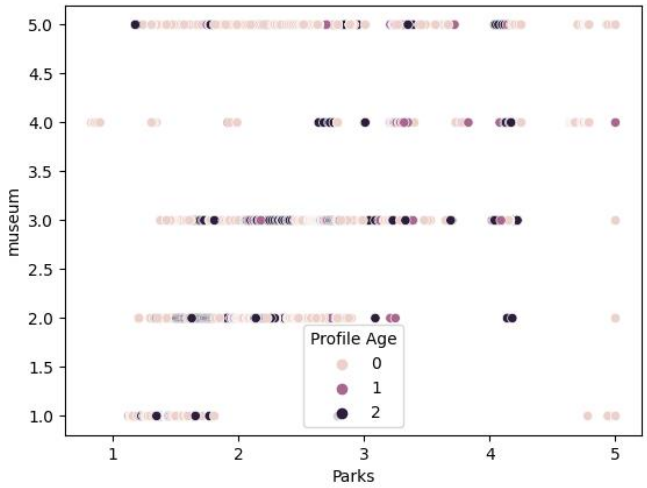
Parks, Marital status:
Accuracy: 58.26%

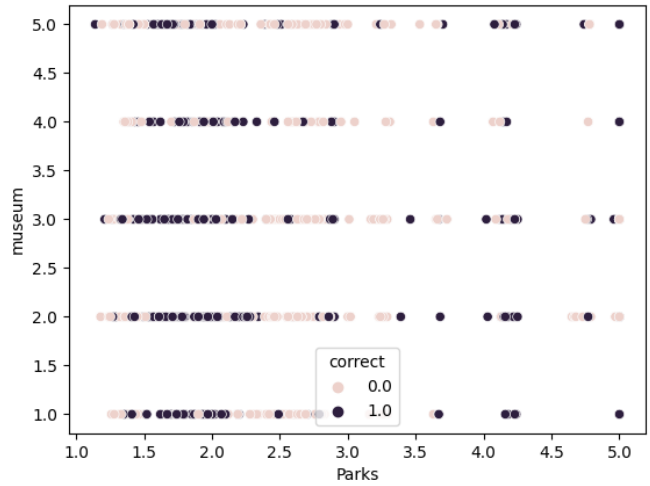





```
Parks   Marital Status  accuracy:  58.26 %
      index  Parks  Marital Status  index  museum  prediction  correct
0      2765   2.61               1   2765       3           2        0
1      3889   5.00               1   3889       3           5        0
2      2710   1.14               1   2710       1           2        0
3      4908   2.61               0   4908       3           3        1
4      4293   1.43               1   4293       1           2        0
...     ...    ...             ...    ...     ...         ...      ...
1284   4189   2.77               0   4189       3           3        1
1285   2075   5.00               1   2075       5           5        1
1286   3333   5.00               1   3333       5           5        1
1287   2349   5.00               1   2349       5           5        1
1288   4642   1.34               1   4642       1           2        0
```

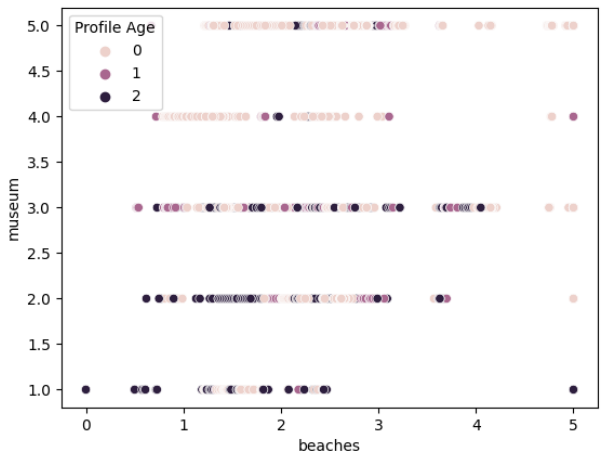
Parks - Profile age
Accuracy: 44.38%

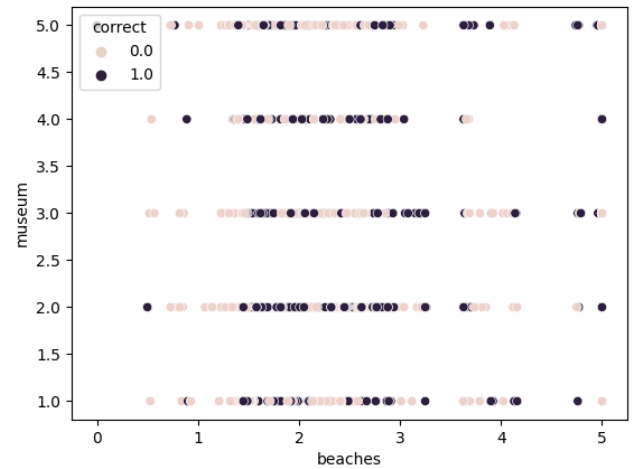





```
Parks   Profile Age  accuracy:  44.38 %
      index  Parks  Profile Age  index  museum  prediction  correct
0      2765   2.61            0   2765       3           2        0
1      3889   5.00            2   3889       3           3        1
2      2710   1.14            0   2710       1           1        1
3      4908   2.61            2   4908       3           2        0
4      4293   1.43            0   4293       1           2        0
...     ...    ...          ...    ...     ...         ...      ...
1284   4189   2.77            1   4189       3           2        0
1285   2075   5.00            2   2075       5           3        0
1286   3333   5.00            0   3333       5           5        1
1287   2349   5.00            0   2349       5           5        1
1288   4642   1.34            2   4642       1           1        1
```

Beaches - Profile age:
Accuracy: 40.42%

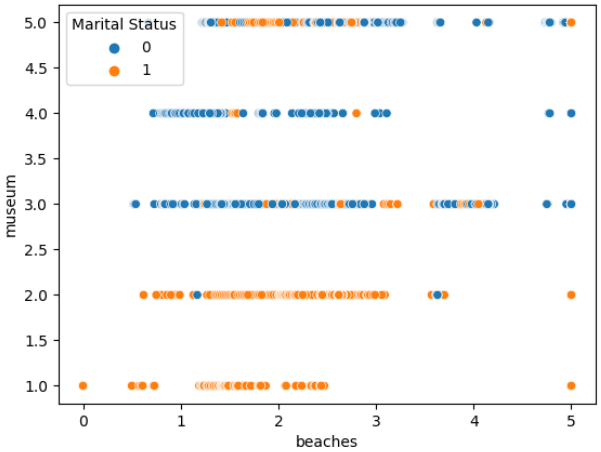


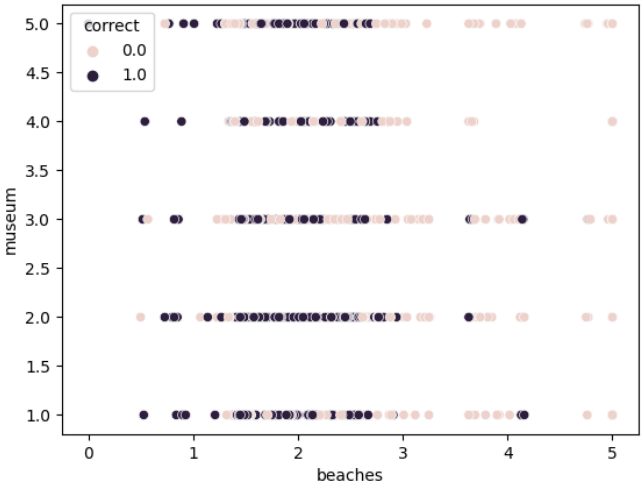



```
Name: prediction, dtype: int64
      index  beaches  Profile Age  index  museum  prediction  correct
0      2765     2.63            0   2765       3           5        0
1      3889     2.28            2   3889       3           2        0
2      2710     1.59            0   2710       1           2        0
3      4908     2.61            2   4908       3           2        0
4      4293     1.45            0   4293       1           2        0
...     ...      ...          ...    ...     ...         ...      ...
1284   4189     2.06            1   4189       3           2        0
1285   2075     2.23            2   2075       5           2        0
1286   3333     2.90            0   3333       5           5        1
1287   2349     2.83            0   2349       5           5        1
1288   4642     5.00            2   4642       1           3        0

[1289 rows x 7 columns]
beaches   Profile Age  accuracy:  40.42 %
```

Beaches - Marital status

Accuracy: 49.03%





	index	beaches	Marital Status	index	museum	prediction	correct
0	2765	2.63	1	2765	3	2	0
1	3889	2.28	1	3889	3	2	0
2	2710	1.59	1	2710	1	2	0
3	4908	2.61	0	4908	3	3	1
4	4293	1.45	1	4293	1	2	0
...	...	...	...	...	...	...	...
1284	4189	2.06	0	4189	3	3	1
1285	2075	2.23	1	2075	5	2	0
1286	3333	2.90	1	3333	5	2	0
1287	2349	2.83	1	2349	5	2	0
1288	4642	5.00	1	4642	1	5	0

[1289 rows x 7 columns]
beaches Marital Status accuracy: 49.03 %

.2a

הערה: את חלק זה הרצתי בג'ופיטר

1. קריאת דאטה-סט מהמסמך תשובות שיצרתי בסעיף הראשון.
2. הרצת עץ החלטה עם קריטריון gini ובמקביל הרצה עם קריטריון entropy;
   - כמעט ואין שינוי בתוצאות עם עדיפות לgini.
   - העץ שנבנה שונה; **ראה מסמכים בתיקיית 2a**

```
metrics_classific(y_test,predicted_gini,X_test)
```

```
[[166   8   0   0   8]
 [ 14 416  23   1  27]
 [  3  23 371   5  39]
 [  1   1   9  54  28]
 [ 12  18  29  16 274]]
              precision    recall  f1-score   support

           1       0.85      0.91      0.88       182
           2       0.89      0.86      0.88       481
           3       0.86      0.84      0.85       441
           4       0.71      0.58      0.64        93
           5       0.73      0.79      0.76       349

    accuracy                           0.83      1546
   macro avg       0.81      0.80      0.80      1546
weighted avg       0.83      0.83      0.83      1546

Accuracy: 82.86%
```

```
metrics_classific(y_test,predicted_entropy,X_test)
```

```
[[166   9   1   1   5]
 [ 15 410  28   3  25]
 [  1  21 373   3  43]
 [  2   2   4  56  29]
 [  9  16  29  26 269]]
              precision    recall  f1-score   support

           1       0.86      0.91      0.89       182
           2       0.90      0.85      0.87       481
           3       0.86      0.85      0.85       441
           4       0.63      0.60      0.62        93
           5       0.73      0.77      0.75       349

    accuracy                           0.82      1546
   macro avg       0.79      0.80      0.79      1546
weighted avg       0.83      0.82      0.82      1546

Accuracy: 82.41%
```

.2b

1. בדיקת חשיבות הקטגוריות השונות למודל מסעיף קודם לצורך בחירת 5 פיצ'רים המשמעותיים ביותר.

	coef
Parks	0.285038
Gyms	0.058333
Local Services	0.038672
Zoo	0.031621
Restaurants	0.027285
Marital Status	0.027159
Beauty & Spas	0.025458
Dance Clubs	0.025130
Burger/Pizza shops	0.024417
beaches	0.024193
Malls	0.023223
Resorts	0.022847
Monuments	0.022194
Pubs/Bars	0.021294
Cafes	0.019670
Swimming Pols	0.016110
Churches	0.015699
Hotels/Other Lodgings	0.014095
View Points	0.012716
Gardens	0.012491
Bakeries	0.008821
Gender	0.002144
Profile Age	0.000693

2. הרצת העץ עם חמישה פיצ'רים נבחרים עומק 5; אפשר לראות שיש שינוי קטן לרעה אך הפחתנו משמעותית את מספר הפיצ'רים.
ראה מסמכים בתיקיית .2b

```
X_train, X_test, y_train, y_test = train_test_split(X_new, y, test_size=0.3, random_state=1)
clf_gini = DecisionTreeClassifier(criterion='gini', max_depth=5, random_state=1)
clf_gini= clf_gini.fit(X_train, y_train)
predicted_gini = clf_gini.predict(X_test)
```

```
metrics_classific(y_test, predicted_gini,X_test)
```

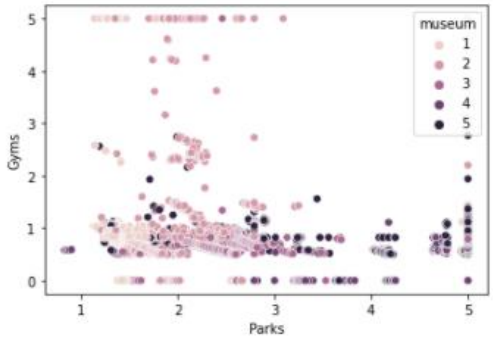
```
[[160  14   0   0   8]
 [ 15 403  29   3  31]
 [  0  46 334   6  55]
 [  0   1   1  21  70]
 [  9  22  34   8 276]]
              precision    recall  f1-score   support

           1       0.87      0.88      0.87       182
           2       0.83      0.84      0.83       481
           3       0.84      0.76      0.80       441
           4       0.55      0.23      0.32        93
           5       0.63      0.79      0.70       349

    accuracy                           0.77      1546
   macro avg       0.74      0.70      0.70      1546
weighted avg       0.77      0.77      0.77      1546

Accuracy: 77.23%
```

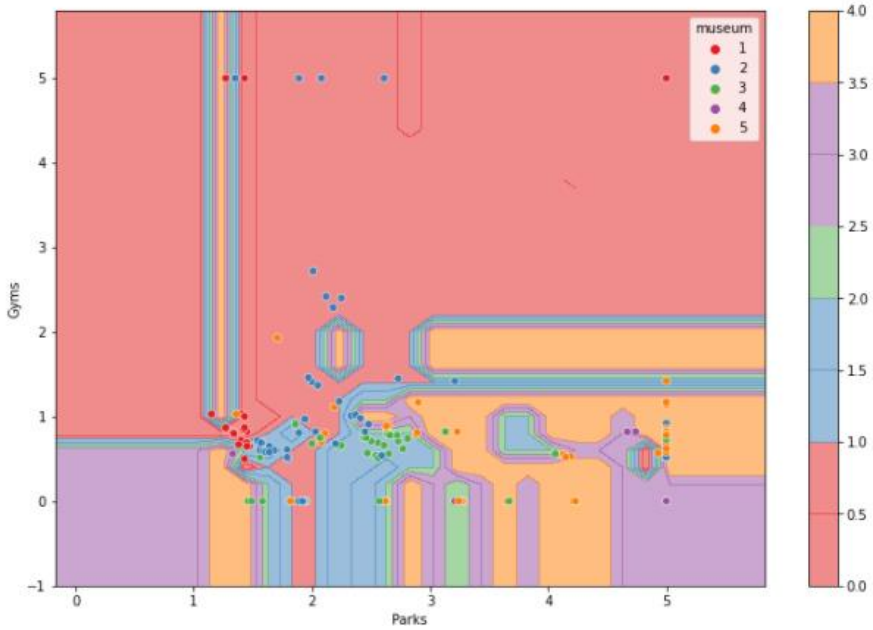
3. הרצת העץ עם חמישה פיצ'רים נבחרים עומק 4; אחוז הדיוק יורד משמעותית.

```
X_train, X_test, y_train, y_test = train_test_split(X_new, y, test_size=0.3, random_state=1)
clf_gini = DecisionTreeClassifier(criterion='gini', max_depth=4, random_state=1)
clf_gini= clf_gini.fit(X_train, y_train)
predicted_gini = clf_gini.predict(X_test)
metrics_classific(y_test, predicted_gini,X_test)
fig = plt.figure(figsize=(25,20))
_ = tree.plot_tree(clf_gini,
                   feature_names=X_new.columns,
                   class_names=['5','4','3','2','1'],
                   filled=True)
plt.savefig('fig_gini_4.png')
```

```
[[162  14   0   0   6]
 [ 15 388  35   0  43]
 [  0  44 314   0  83]
 [  1   1   1   0  90]
 [ 21  25  24   0 279]]
              precision    recall  f1-score   support

           1       0.81      0.89      0.85       182
           2       0.82      0.81      0.81       481
           3       0.84      0.71      0.77       441
           4       0.00      0.00      0.00        93
           5       0.56      0.80      0.66       349

    accuracy                           0.74      1546
   macro avg       0.61      0.64      0.62      1546
weighted avg       0.72      0.74      0.72      1546

Accuracy: 73.93%
```

4. הרצתי עץ החלטה עם קריטריון gini וללא הגבלת עומק.
אפשר לראות גרף שמראה את הקשר בין parksו gyms והצבע שלהן לפי דירוג המוזיאון.
אפשר לראות שכמה ש parks יותר גבוה כך הדירוג של המוזאונים גבוה יותר.
בנוסף הוספתי גרף שרואים בו את הסיווגים הנכונים והסיווגים הלא נכונים לפי הצבעים.





Summary

קיבלתי דאטה-סט שהיה צריך לטפל בו מבחינת ערכים חסרים, ערכי קיצון ושגיאות.
הרצתי גרפים שונים וגיליתי הרבה מידע וקשרים בין הפיצ'רים השונים.
ראיתי שעצי החלטה מביאים תשובה טובה יותר מ naïve base.
לעצי ההחלטה נתתי קריטריונים שונים; gini, entropy, מספר פיצ'רים שונה ועומקים שונים: 4,5.
ראיתי שהשינוי בין gini ו entropy לא גדול במיוחד, גם השינוי בין כל הפיצ'רים ל 5 הטובים ביותר לא היה גדול כל כך והשתלם לקצץ בכמות של הפיצ'רים וגיליתי שהשוני בין עומק 5 לעומק 4 כבר היה גדול יחסית ונתן תחזיות פחות טובות משמעותית.