Doron Shamai 312538325 Itay Merhav 205503352

E-Commerce

Project: Recommendation system

מבוא –

בעבודה זו ביצענו את כל החישובים על מאגר מידע של 100,000 ביקורות על סרטים. הביקורות במאגר הזה נכתבו על ידי 943 צופים והם ביקרו יחדיו 1682 סרטים. עבור כל סרט קיבלנו את הדירוג שהצופים העניקו לו. ואת המידע על הצופים שנתנו את הדירוג כמו מגדר, גיל וכו.

בעזרת כלל נתונים הללו יצרנו מודלים שונים למערכות המלצה, ופעלנו לפי השלבים השונים שלמדנו בקורס.
תחילה, עברנו על הנתונים וניתחנו את המידע על מנת להבין בצורה טובה יותר את הקשר בין הנתונים עצמם וההשפעה של כל נתון על הדירוג שניתן בסופו של דבר לכל סרט.
לאחר כן, ביצענו חלוקה של מאגר המידע לשני חלקים: train ו- test וזאת על מנת לבצע הערכה ולהסיק מידע נוסף על גבי הנתונים.

ביצענו הפרדה ושילוב של מאגר המידע עם מאגר מידע של אנשים אשר צפו בסרטים ודירגו את הסרטים ובכך יכולנו להציג פילוחים שונים ולקבל מידע נוסף שלא ניתן לקבל באמצעות הסתכלות על כל מאגר מידע בנפרד.

בנינו מודל חיזוי דירוג לכל הסרטים במאגר המסתמך על ממוצע הדירוג של הסרט. ביצענו זאת עבור האוכלוסייה הכללית ועבור אוכלוסיית הנשים ואוכלוסיית הגברים בנפרד.

בהמשך בנינו מודלים שונים במטרה לחזות את הדירוג של הסרטים. המודלים בהם השתמשנו התבססו בין היתר על Matrix factorization, דמיון בין הסרטים ואף התבססות על תוכן ששייך לסרט.

לאחר מכן יצרנו מודל מסוג Neural collaborating filtering. במסגרתו יצרנו רשתות שונות שביצעו את החיזוי.

בשלב האחרון השתמשנו במודל מסוג Deep FM. מודל זה משלב את ה- Matrix factorization יחד עם תחום רשתות הנוירונים למטרות החיזוי.

חלק א – ניתוח מידע

<u>שאלה 1:</u>

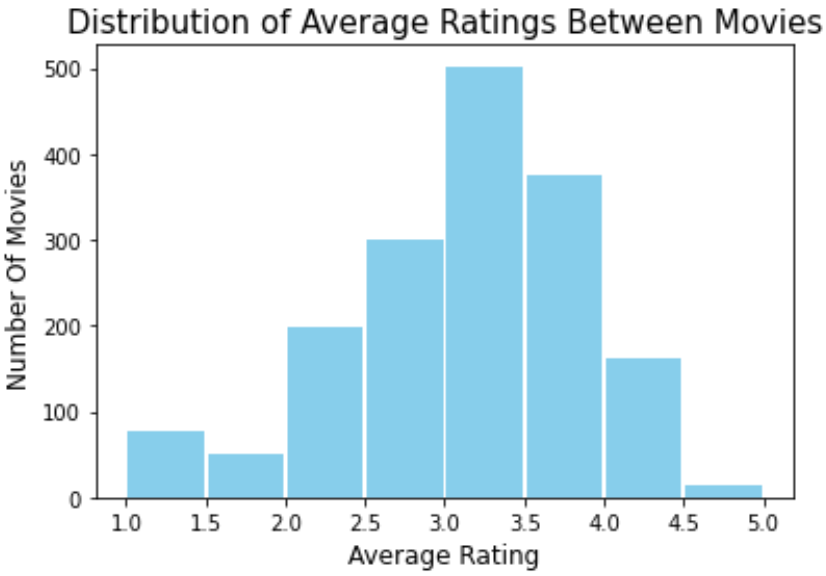
בשאלה זו התבקשנו לחשב את ממוצע הדירוג עבור כל סרט ולהציג את ההתפלגות של הסרטים על פי מאפיינים שונים.

נדגיש כי עבור כל סרט בדקנו כמה צופים דירגו אותו, ותמיד בחרנו את הסרטים עם ממוצע הדירוג הגבוה ביותר ושגם מספר הדירוגים שלהם הוא גבוה, כלומר אם סרט דורג פעם אחת בדירוג 5 וסרט אחר דורג עשר פעמים עם דירוג 5 ניקח את הסרט השני.

<u>סעיף א –</u>

בסעיף זה ביצענו חישוב של ממוצע הדירוג עבור כל סרט והצגנו את ההתפלגות של כמות הסרטים עבור כל ממוצע דירוג.
בנוסף ביצענו מיון של כל הסרטים לפי הדירוג ולפי ממוצע הדירוג והצגנו את שלושת הסרטים עם הממוצע הדירוג הגבוה ביותר.

יצרנו היסטוגרמה אשר מראה את ההתפלגות של ממוצע דירוג הסרטים ואת כמות הסרטים עבור כל ממוצע דירוג. ניתן להבחין מההיסטוגרמה כי מרבית הסרטים מדורגים בדירוג של בין 3 ל- 3.5 ולאחר מכן בין 3.5 ל- 4.



בנוסף לכך שלפנו את שלושת הסרטים בעלי ממוצע הדירוג הגבוה ביותר:

Top 3 Movies by Avera ge Rating:

	Movie title	Average rating
1189	Prefontaine (1997)	5.0
1293	Star Kid (1997)	5.0
1467	Saint of Fort Washington, The (1993)	5.0

הסתכלנו על המידע שיש לנו עבור שלושת הסרטים הללו וראינו כי שני סרטים הם סרטי דרמה. דבר זה הגיוני בהתחשב בכך שקטגוריית סרטי הדרמה היא הקטגוריה עם הכי הרבה סרטים. הסרט האמצעי הוא סרט ילדים אשר משלב פנטזיה, מדע בדיוני והרפתקאות.

נרצה לציין כי דבר זה לא היה לנו מובן מאליו אך לאחר שביצענו את סעיף ג' וראינו כי קטגורית הדרמה מובילה שם, החלטנו לחזור אחורה ולבדוק גם עבור הסרטים האלה.

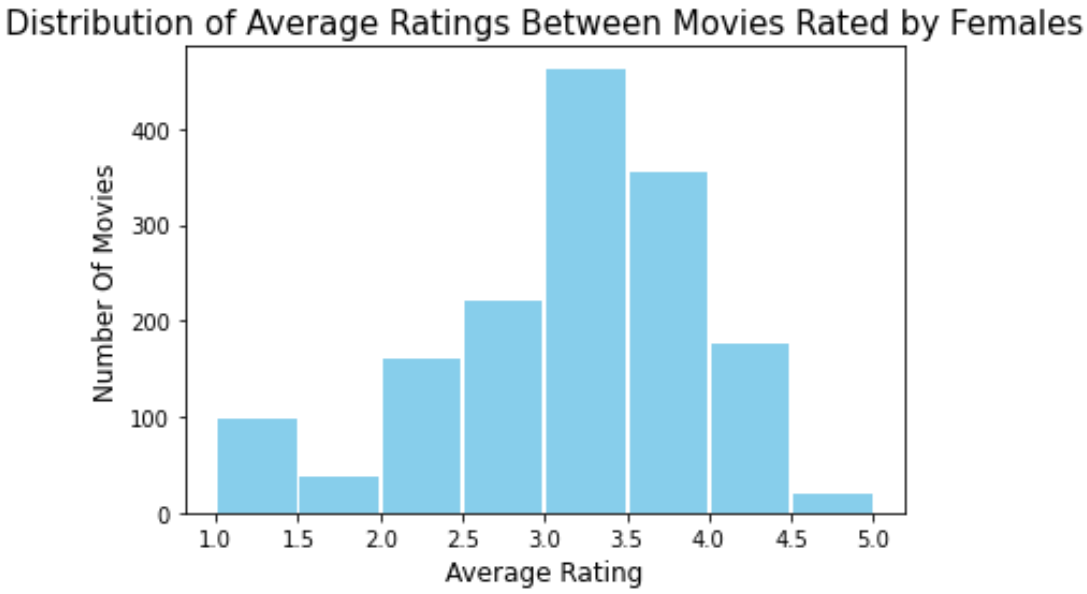
<u>סעיף ב –</u>

בסעיף זה ביצענו את אותם החישובים של סעיף א' אך ביצענו אותם עבור אוכלוסיית הנשים ועבור אוכלוסיית הגברים בנפרד.

נשים –

עבור אוכלוסיית הנשים ניתן לראות שההתפלגות של דירוג הסרטים נשמרת בדומה לכלל האוכלוסייה. רוב הסרטים מדורגים בין 3 ל- 3.5 ולאחר מכן בין 3.5 ל- 4.
אך כן נראה שיש גידול בכמות הסרטים שמדורגים בין 3.5 ל- 4 ובנוסף כמות הסרטים שמדורגים בין 4 ל- 4.5 כמעט שווה לכמות הסרטים שמדורגים בין 2.5 ל- 3 דבר זה שונה ממה שקורה באוכלוסייה הכללית ששם יש הרבה יותר סרטים שמדורגים בין 2.5 ל- 3 ולאחר מכן יש יותר סרטים בדירוג של בין 2 ל -2.5 ורק לאחר מכן הדירוג של הסרטים הוא בין 4 ל- 4.5.

נבין מכך כי הנשים מדרגות את אותם הסרטים בציונים גבוהים יותר מהאוכלוסייה הכללית.



שלושת הסרטים עם ממוצע הדירוג הגבוה ביותר אצל נשים הם:

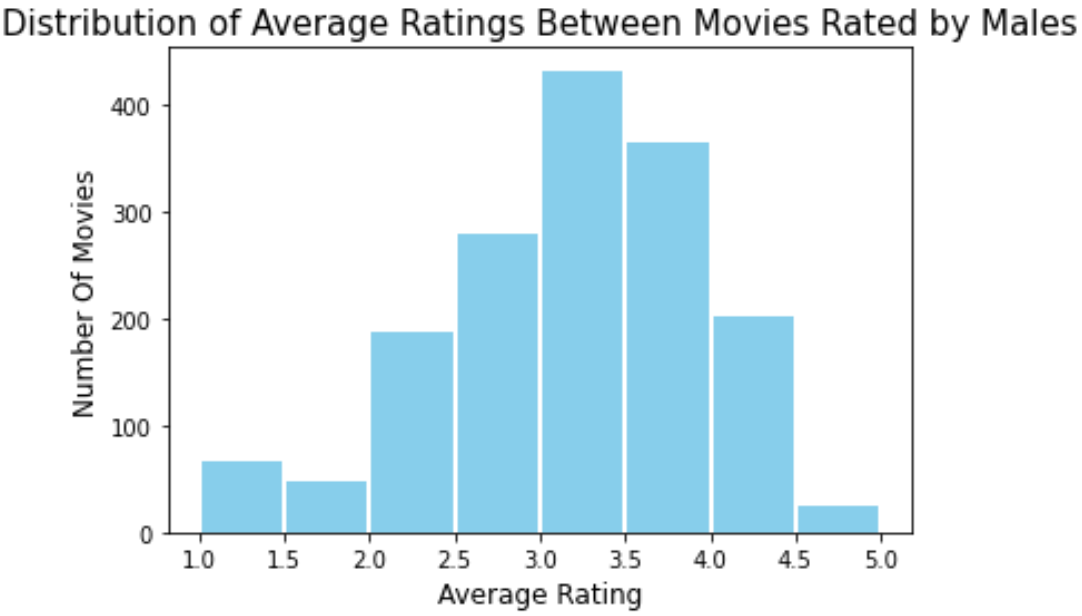
Top 3 Movies Average Rating For Females:

	Movie title	Average rating
1303	Mina Tannenbaum (1994)	5.0
74	Faster Pussycat! Kill! Kill! (1965)	5.0
118	Maya Lin: A Strong Clear Vision (1994)	5.0

בנוסף ראינו שהסרט הראשון הוא סרט דרמה, הסרט השני הוא גם סרט דרמה והסרט השלישי הוא דוקומנטרי.

גברים –

עבור אוכלוסיית הגברים ניתן לראות שההתפלגות של דירוג הסרטים נשמרת בדומה לכלל האוכלוסייה. רוב הסרטים מדורגים בין 3 ל- 3.5 ולאחר מכן בין 3.5 ל- 4. בשונה מדירוג הנשים, אצל הגברים יש יותר סרטים שמדורגים בין 3.5 ל- 4.
נראה שגם אצל אוכלוסיית הגברים בדומה אצל הנשים יש כמעט שוויון בין כמות הסרטים שמדורגים בין 4 ל- 4.5 ובין כמות הסרטים שמדורגת בין 2 ל- 2.5. אך אצל אוכלוסיית הגברים כמות הסרטים שמדורגת בין 2.5 ל- 3 היא הרבה יותר גדולה מאשר אצל הנשים.



שלושת הסרטים עם ממוצע הדירוג הגבוה ביותר אצל גברים הם:

	Movie title	Average rating
1291	Star Kid (1997)	5.0
1173	Hugo Pool (1997)	5.0
1187	Prefontaine (1997)	5.0

בנוסף ראינו שהסרט הראשון הוא סרט ילדים פנטזיה והרפתקאות, הסרט השני הוא סרט רומנטי והסרט השלישי הוא דרמה.

נבדוק את השוני בין הגברים לנשים מבחינת הסרטים שדורגו

```
Top 3 Movies Max Difference Average Rating Between Females and Males:
```

	Movie title	Average rating
1327	Delta of Venus (1994)	4.000000
867	Two or Three Things I Know About Her (1966)	3.666667
1453	Sliding Doors (1998)	3.500000

נראה כי שלושת הסרטים הללו הם הסרטים עם ההפרש הגדול ביותר בדירוגים בין נשים לגברים.

נשים לב כי שלושת הסרטים האלה דורגו מאוד בממוצע גבוה אצל באוכלוסיית הגברים ובאוכלוסיית הנשים דורגו בממוצע מאוד נמוך

	Average rating_female	Average rating_male	Average rating
1327	1.0	5.000000	4.000000
867	1.0	4.666667	3.666667
1453	1.0	4.500000	3.500000

מכאן נוכל להבין מדוע הסרטים הללו הם עם הפער הגדול ביותר בין הגברים לנשים.

בנוסף, שלושת הסרטים הם מז'אנר הדרמה.

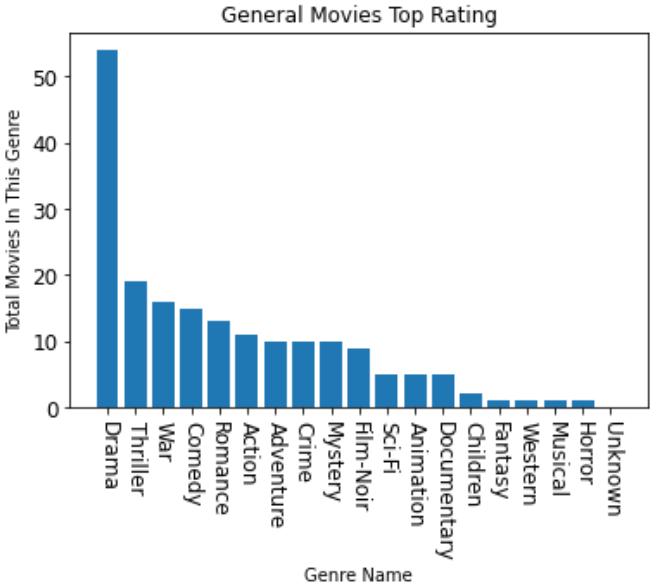
<u>סעיף ג –</u>

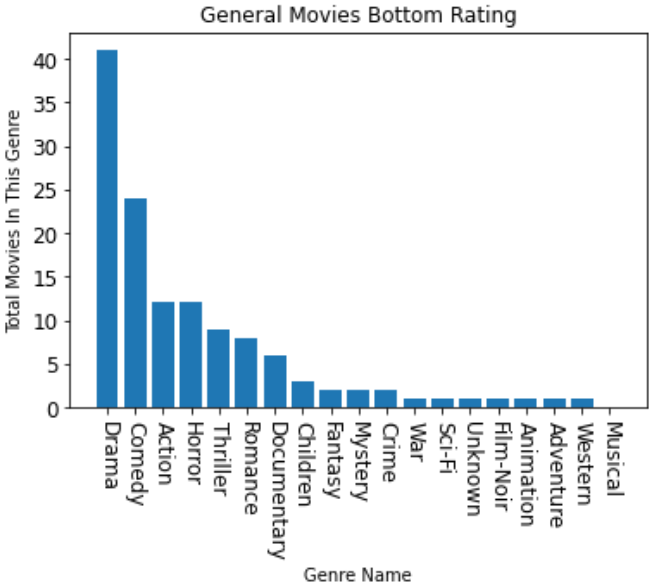
בסעיף זה בדקנו את התפלגות הקטגוריות של הסרטים, בדקנו עבור 100 הסרטים עם הדירוגים הגבוהים ו 100 הסרטים עם הדירוגים הנמוכים ביותר, ועבור חתך אוכלוסיות שונות.

בחרנו לבדוק את האוכלוסיות הבאות: אוכלוסיית הגברים ואוכלוסיית הנשים. בנוסף, בדקנו עבור גילאים שונים. אנשים בני 30 ומתחת ואנשים מעל לגיל 30.

נציג את ההיסטוגרמות של התפלגות הקטגוריות של הסרטים עבור כל סוג אוכלוסייה שבדקנו עם הסרטים שדורגו הכי גבוה והסרטים שדורגו הכי נמוך.

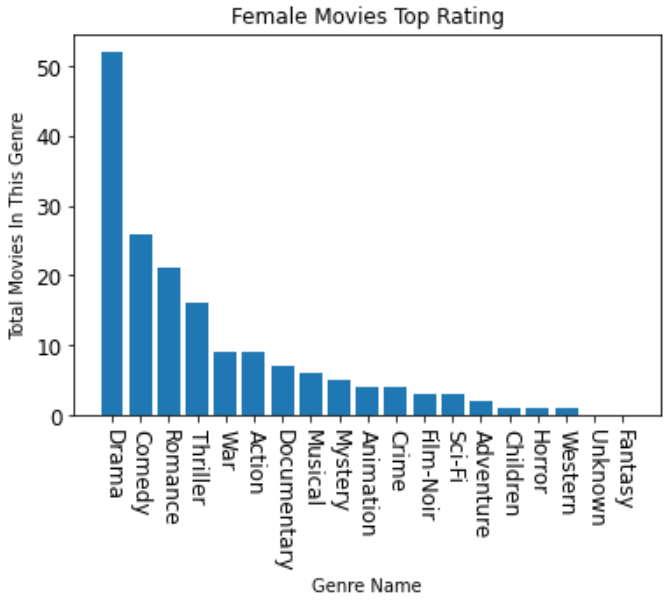
<u>עבור האוכלוסייה הכללית:</u>

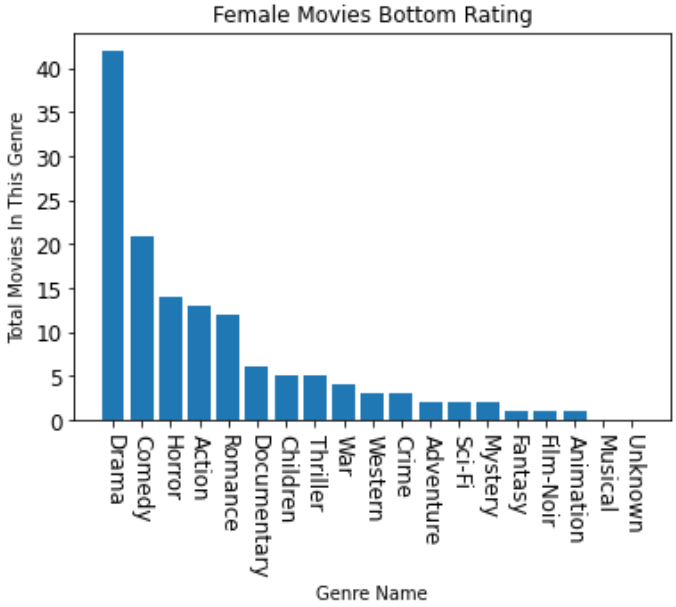




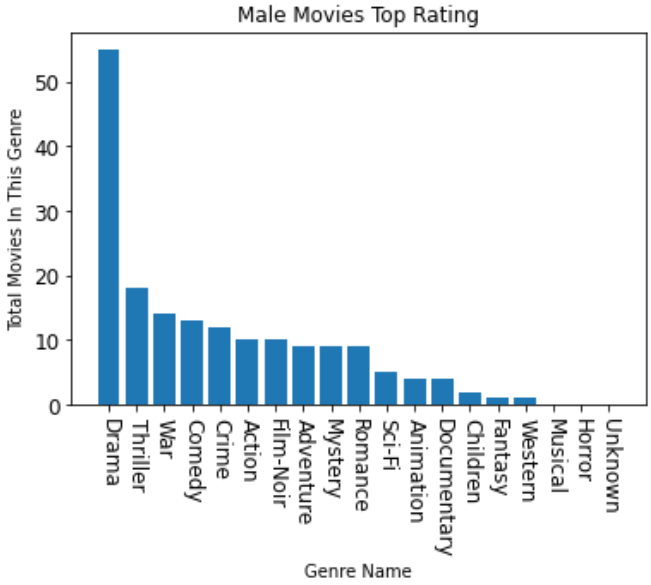
ניתן לראות שהסרטים עם הדירוג הגבוהה ביותר שייכים לז'אנר הדרמה אך גם הסרטים עם הדירוג הנמוך ביותר שייכים לז'אנר הזה. דבר זה הגיוני מכיוון שהרבה סרטים משתייכים לז'אנר של סרטי הדרמה.

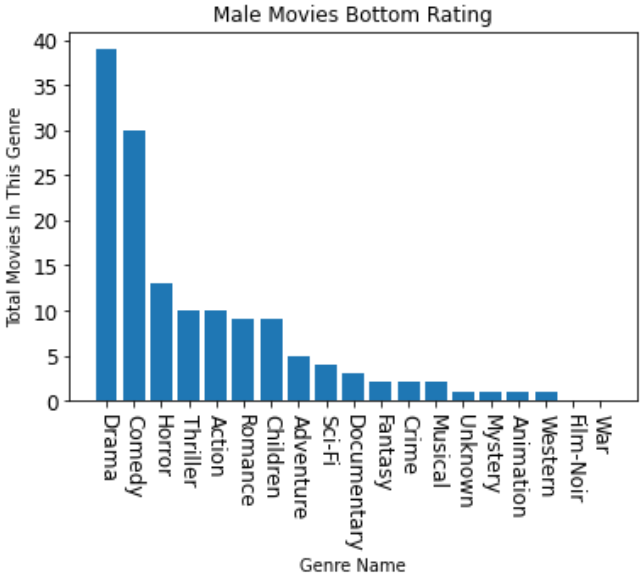
עבור אוכלוסיית הנשים:



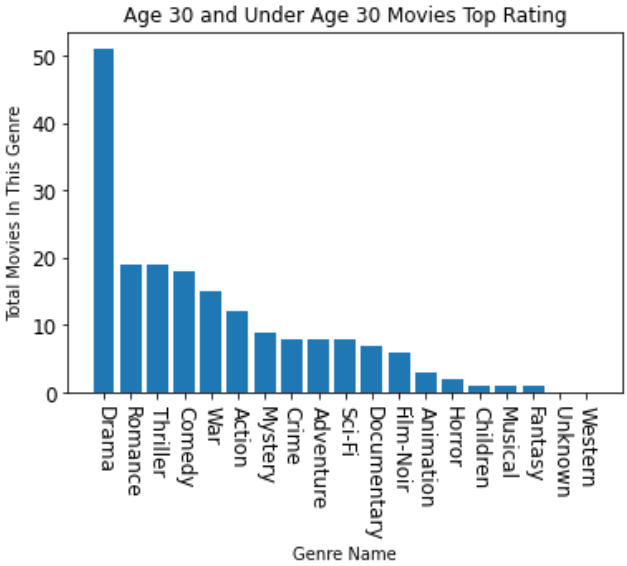


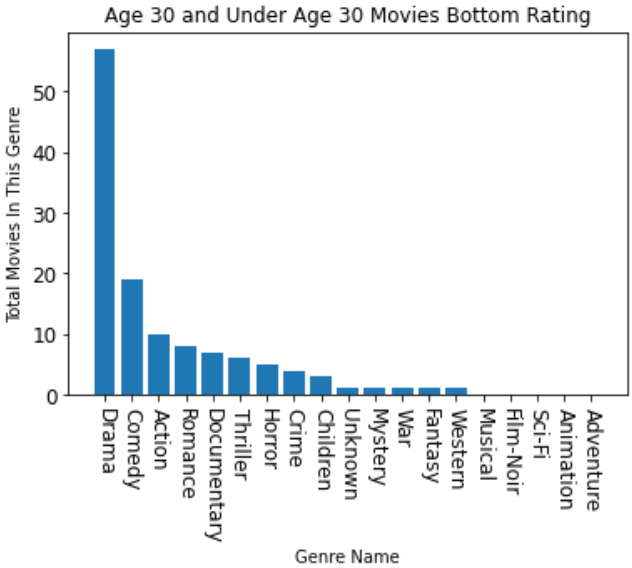
עבור אוכלוסיית הגברים:



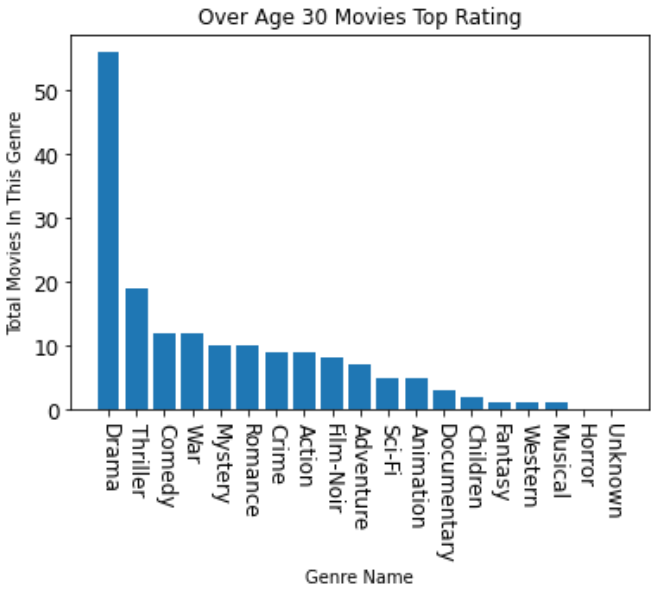


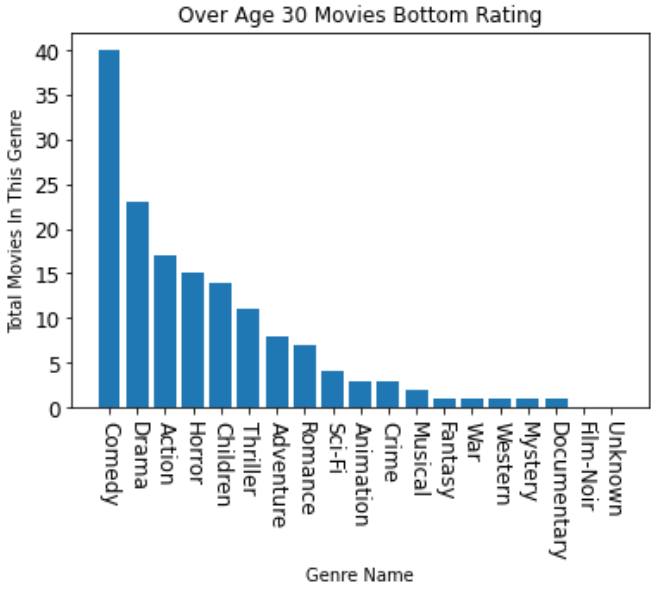
עבור אוכלוסייה בגיל 30 ומטה:





עבור אוכלוסייה מעל גיל 30:





נראה שעבור האוכלוסייה מעל גיל 30, הסרטים עם הדירוגים הנמוכים ביותר הם סרטי הקומדיה.
ניתן לחשוב שדבר זה נובע אולי מכך שסרטי קומדיה יכולים להיות טיפשיים או שטותיים ואנשים בגיל מבוגר יותר פחות מתחברים לזה. זו השערה בלבד ולא נתיימר לדעת מדוע באמת אלו התוצאות.

נסכם את המעבר על התוצאות, שלושת הקטגוריות של הסרטים שמדורגים הכי גבוה אצל נשים הם: דרמה, קומדיה ורומנטיקה בהתאמה כך שדרמה יש הכי הרבה סרטים, לאחר מכן קומדיה ובמקום השלישי סרטים רומנטיים.

אצל גברים הסרטים שמדורגים הכי גבוה הם: דרמה, מתח ומלחמה.

לעומת זאת שלושת הקטגוריות של הסרטים שמדורגים הכי נמוך אצל הנשים ואצל הגברים הם באותן הקטגוריות שהן: דרמה, קומדיה ואימה.

סעיף ד –

בסעיף זה התבקשנו לומר מי הם הסרטים הפופולאריים ביותר וכיצד נמדוד פופולאריות.

לדעתנו, פופולאריות נמדדת על ידי כמות האנשים שצפו בסרט ודירגו אותו ועל ידי ממוצע הדירוג. למשל אם אני ארצה ללכת לראות סרט, אני כנראה אסתכל כמה אנשים דירגו כל סרט ומה היה ממוצע הדירוג. כי במידה ויש סרט שיש לו ממוצע דירוג מאוד גבוה אבל מעט מאוד אנשים דירגו אותו זה לא נותן לי קנה מידה אמיתי האם הסרט הוא פופולארי ומהצד השני, במידה וסרט דורג המון פעמים, אך הדירוג שלו נמוך גם אז זה לא סרט שהייתי אומרת שהוא פופולארי.

ולכן ניסינו לשמור על טרייד-אוף בין כמות גבוה של אנשים שצפו בסרט ודירגו אותו לבין ממוצע דירוג גבוה. ביצענו זאת על ידי מיונים שונים, או מיון על פי כמות הדירוגים או מיון על פי ממוצע הדירגו כל פעם של קבוצה קטנה יותר של סרטים.

עשרת הסרטים הפופולאריים ביותר לפי החישוב שביצענו:

Movie title	Average rating	Item count
Star Wars (1977)	4.358491	583
Silence of the Lambs, The (1991)	4.289744	390
Godfather, The (1972)	4.283293	413
Raiders of the Lost Ark (1981)	4.252381	420
Titanic (1997)	4.245714	350
Empire Strikes Back, The (1980)	4.204360	367
Princess Bride, The (1987)	4.172840	324
Fargo (1996)	4.155512	508
Monty Python and the Holy Grail (1974)	4.066456	316
Pulp Fiction (1994)	4.060914	394

סעיף ה –

בסעיף זה נחשב את ה sparsity, כלומר נבדוק כמה המטריצה דלילה, המשמעות היא האם יש הרבה אנשים שדירגו הרבה סרטים. ולכן נחשב את

נחשב את כמות הסרטים ואת כמות האנשים שדירגו את הסרטים. נכפיל את שני המספרים הללו ונחלק את המספר שקיבלנו בסך כל הביקורות שיש.

נקבל שה- sparsity הוא 0.06304669364224531

בנוסף, נחשב בממוצע כמה פעמים אדם דירג סרטים. נבצע זאת על ידי חלוקה של סך כל הביקורות בכמות האנשים שדירגו.

נקבל שכל אדם בממוצע דירג 106 סרטים.

חלק ב – המלצות לא אישיות

<u>שאלה 2:</u>

<u>סעיף א –</u>

בסעיף זה נתבקשנו לבנות מודל חיזוי מבוסס לכל סרט על סמך ממוצע הדירוג שחושב בסעיף א' של שאלה 1. מודל החיזוי שלנו מחזיר את הממוצע כפי שהוא מחושב על ה- train set. לאחר ביצוע המודל חישבנו את ערך ה- MAE של המודל: 0.694.

<u>סעיף ב –</u>

בסעיף זה נתבקשנו לחזור על לחזור על התהליך מהסעיף הקודם אך כעת פיצלנו את הנתונים לפי מגדר, כלומר חלוקה לגברים ולנשים. על מנת לבצע זאת איחדנו את נתוני הדירוג יחד עם נתונים על הצופים.

התוצאות מסוכמות בטבלה הבאה:

מגדר	MAE
נשים	0.925
גברים	0.921

התוצאות שקיבלנו בסעיף א' יותר טובות מהתוצאות שקיבלנו עבור בסעיף ב' כיוון שהשגיאה קטנה יותר. התוצאות שקיבלנו הגיוניות כיוון שככל שיש פחות אז הערך הממוצע עשוי להתרחק מהערך האמיתי. כאשר פיצלנו את ה- data set לפי מגדר אז השפענו על ערך זה.

חלק ג – המלצות אישיות

<u>שאלה 3:</u>

<u>סעיף א –</u>

בסעיף זה שילבנו מידע מכמה קבצים על מנת ליצור Data Frame אחד שיהיה נוח לעבודה. שילבנו את הקבצים של הדירוג יחד עם הקבצים על הסרטים והמשתמשים. לאחר האיחוד יצרנו שלושה מודלים לחיזוי:

- Matrix Factorization
- Item Similarity
- Item Content

<u>סעיף ב –</u>

בסעיף זה חישבנו את ה- Mean absolute error(MAE) עבור כל אחד מהמודלים שיצרנו בסעיף הקודם. בטבלה הבאה נציג את החישובים:

שם המודל	MAE
Matrix Factorization	0.442
Item Similarity	3.47
Item Content	3.52

<u>סעיף ג –</u>

השוואה בים המודלים:

ניתן לראות כי המודל Matrix Factorization היה בעל השגיאה הנמוכה ביותר בפער די גדול לעומת יתר המודלים שיצרנו.

נציג כעת את הזמנים שלקח לכל אחד מהמודלים לרוץ:

שם המודל	זמן ריצה
Matrix Factorization	1.98 שניות
Item Similarity	0.21 שניות
Item Content	13.58 דקות

ניתן לראות כי המודל הכי מהיר היה המודל של Item Similarity אשר לקחת פחות משניה אחת לסיום ההרצה. במקום השני עם קצת יותר מ-2 שניות היה המודל Matrix Factorization שהיה בעל השגיאה הנמוכה ביותר. המודל של Item Content לקח סדר גודל של 14 דקות אשר בהשוואה למודלים האחרים מדובר על זמן ריצה גדול מאוד. לסיכום היינו ממליצים על המודל Matrix Factorization כיוון שהניב תוצאות טובות בזמן מאוד נמוך.

שאלה 4:

סעיף א –

בסעיף זה מימשנו מודל Neural Collaborating Filtering למטרת חיזוי של ה-rating אשר מכיל רק שכבת hidden אחת.

ניתן לראות את המודל שיצרנו בתמונה הבאה:

```
ncf model
Model: "model"
__________________________________________________________________________________________________
Layer (type)                    Output Shape         Param #     Connected to
==================================================================================================
user_input (InputLayer)         [(None, 1)]          0
__________________________________________________________________________________________________
item_input (InputLayer)         [(None, 1)]          0
__________________________________________________________________________________________________
user_embedding (Embedding)      (None, 1, 20)        18860       user_input[0][0]
__________________________________________________________________________________________________
item_embedding (Embedding)      (None, 1, 20)        33640       item_input[0][0]
__________________________________________________________________________________________________
flatten (Flatten)               (None, 20)           0           user_embedding[0][0]
__________________________________________________________________________________________________
flatten_1 (Flatten)             (None, 20)           0           item_embedding[0][0]
__________________________________________________________________________________________________
concatenate (Concatenate)       (None, 40)           0           flatten[0][0]
                                                                 flatten_1[0][0]
__________________________________________________________________________________________________
dense (Dense)                   (None, 1)            41          concatenate[0][0]
__________________________________________________________________________________________________
dropout_1 (Dropout)             (None, 1)            0           dense[0][0]
__________________________________________________________________________________________________
prediction (Dense)              (None, 1)            2           dropout_1[0][0]
==================================================================================================
Total params: 52,543
Trainable params: 52,543
Non-trainable params: 0
```

בחרנו להשתמש ב- optimizer מסוג Adamax וה- loss function היא mse.

סעיף ב –

בסעיף זה חישבנו את ה- MAE של המודל שיצרנו בסעיף הקודם וגם יצרנו מודלים נוספים וגם עבורם חישבנו את ה- MAE.

תוצאות ה- MAE עבור המודל שיצרנו בסעיף הקודם: 0.745

כעת נציג את המודלים הנוספים שבנינו בסעיף זה:

מודל 2: העלנו את פרמטר ה- hidden_dim ל- 10 (כאשר במודל הראשון הוא היה 1).

```
ncf model
Model: "model_2"
__________________________________________________________________________________________________
Layer (type)                    Output Shape         Param #     Connected to
==================================================================================================
user_input (InputLayer)         [(None, 1)]          0
__________________________________________________________________________________________________
item_input (InputLayer)         [(None, 1)]          0
__________________________________________________________________________________________________
user_embedding (Embedding)      (None, 1, 20)        18860       user_input[0][0]
__________________________________________________________________________________________________
item_embedding (Embedding)      (None, 1, 20)        33640       item_input[0][0]
__________________________________________________________________________________________________
flatten_4 (Flatten)             (None, 20)           0           user_embedding[0][0]
__________________________________________________________________________________________________
flatten_5 (Flatten)             (None, 20)           0           item_embedding[0][0]
__________________________________________________________________________________________________
concatenate_2 (Concatenate)     (None, 40)           0           flatten_4[0][0]
                                                                 flatten_5[0][0]
__________________________________________________________________________________________________
dense_2 (Dense)                 (None, 10)           410         concatenate_2[0][0]
__________________________________________________________________________________________________
dropout_5 (Dropout)             (None, 10)           0           dense_2[0][0]
__________________________________________________________________________________________________
prediction (Dense)              (None, 1)            11          dropout_5[0][0]
==================================================================================================
Total params: 52,921
Trainable params: 52,921
Non-trainable params: 0
__________________________________________________________________________________________________
```

מודל 3: השתמשנו כעת ב- optimizer אחר. במקום Adamax בחרנו ב- SGD.

```
ncf model
Model: "model_3"
__________________________________________________________________________________________________
Layer (type)                    Output Shape         Param #     Connected to
==================================================================================================
user_input (InputLayer)         [(None, 1)]          0
__________________________________________________________________________________________________
item_input (InputLayer)         [(None, 1)]          0
__________________________________________________________________________________________________
user_embedding (Embedding)      (None, 1, 20)        18860       user_input[0][0]
__________________________________________________________________________________________________
item_embedding (Embedding)      (None, 1, 20)        33640       item_input[0][0]
__________________________________________________________________________________________________
flatten_6 (Flatten)             (None, 20)           0           user_embedding[0][0]
__________________________________________________________________________________________________
flatten_7 (Flatten)             (None, 20)           0           item_embedding[0][0]
__________________________________________________________________________________________________
concatenate_3 (Concatenate)     (None, 40)           0           flatten_6[0][0]
                                                                 flatten_7[0][0]
__________________________________________________________________________________________________
dense_3 (Dense)                 (None, 20)           820         concatenate_3[0][0]
__________________________________________________________________________________________________
dropout_7 (Dropout)             (None, 20)           0           dense_3[0][0]
__________________________________________________________________________________________________
prediction (Dense)              (None, 1)            21          dropout_7[0][0]
==================================================================================================
Total params: 53,341
Trainable params: 53,341
Non-trainable params: 0
__________________________________________________________________________________________________
```

מודל 4: הגדלנו את גודל השכבה כעת מכילה 30 נוירונים (במקום 20 במודל הראשון).

```
ncf model
Model: "model_4"
__________________________________________________________________________________________________
Layer (type)                    Output Shape         Param #     Connected to
==================================================================================================
user_input (InputLayer)         [(None, 1)]          0
__________________________________________________________________________________________________
item_input (InputLayer)         [(None, 1)]          0
__________________________________________________________________________________________________
user_embedding (Embedding)      (None, 1, 30)        28290       user_input[0][0]
__________________________________________________________________________________________________
item_embedding (Embedding)      (None, 1, 30)        50460       item_input[0][0]
__________________________________________________________________________________________________
flatten_8 (Flatten)             (None, 30)           0           user_embedding[0][0]
__________________________________________________________________________________________________
flatten_9 (Flatten)             (None, 30)           0           item_embedding[0][0]
__________________________________________________________________________________________________
concatenate_4 (Concatenate)     (None, 60)           0           flatten_8[0][0]
                                                                 flatten_9[0][0]
__________________________________________________________________________________________________
dense_4 (Dense)                 (None, 20)           1220        concatenate_4[0][0]
__________________________________________________________________________________________________
dropout_9 (Dropout)             (None, 20)           0           dense_4[0][0]
__________________________________________________________________________________________________
prediction (Dense)              (None, 1)            21          dropout_9[0][0]
==================================================================================================
Total params: 79,991
Trainable params: 79,991
Non-trainable params: 0
__________________________________________________________________________________________________
```

נסכם בטבלה את תוצאות ה- MAE של כל אחד מהמודלים:

שם המודל	פרמטרים	MAE	זמן אימון
Model_1	Hidden Layer = 1 Optimizer = Adamax K_LATENT = 20	0.881	18 שניות
Model_2	Hidden Layer = 10 Optimizer = Adamax K_LATENT = 20	0.753	75 שניות
Model_3	Hidden Layer = 20 Optimizer = SGD K_LATENT = 20	0.757	31 שניות
Model_4	Hidden Layer = 20 Optimizer = Adamax K_LATENT = 30	0.751	63 שניות

<u>סעיף ג –</u>

ניתן לראות כי לא קיים הבדל גדול בין המודלים שבנינו. נציין כי המודל הרביעי בו ה- K_LATENT גדול יותר הביא לשגיאה הכי נמוכה מבין המודלים אך לא בפער מאוד גדול. מבחינת זמן האימון, למודל הראשון היה זמן האימון הקצר ביותר. היינו ממליצים על המודל הראשון שהניב תוצאות טובות בזמן אימון הכי קצר.

שאלה 5:

סעיף א –

בסעיף זה הצענו 2 מודלים אשר מבוססים על ספריית DeepCtr.

1. מודל 1- DeepFM עם הפרמטרים הבאים:
   - Optimizer: Adam
   - Loss Function: MSE
2. מודל 2- DeepFM עם הפרמטרים הבאים:
   - Optimizer: SGD
   - Loss Function: MSE

סעיף ב –

בסעיף זה בחרנו 2 מאפיינים של צופה אשר ישולבו כל פעם בשני המודלים שהצענו בסעיף הקודם.

המאפיינים הם:

- Timestamp + Gender
- Age + Occupation

בנוסף בסעיף זה ביצענו את האימון של 2 המודלים. שני המודלים אומנו עם 30 חזרות לכל אימון.

סעיף ג –

נסכם את התוצאות בטבלה הבאה:

שם המודל	פרמטרים	MAE	זמן אימון
Model_1	Optimizer = Adam Loss function = MSE מאפיינים: timestamp+ gender	1.06	30 שניות
Model_2	Optimizer = SGD Loss function = MSE מאפיינים: timestamp+ gender	0.85	30 שניות
Model_1	Optimizer = Adam Loss function = MSE מאפיינים: Age+ Occupation	0.86	30 שניות
Model_2	Optimizer = SGD Loss function = MSE מאפיינים: Age+ Occupation	0.84	30 שניות

סעיף ד + ה –

ניתן לראות כי המודל השני הביא לתוצאות טובות יותר מבחינת השגיאה אשר הייתה נמוכה יותר עבור 2 זוגות המאפיינים שבחרנו לעומת השגיאה במודל הראשון. מבחינת זמני ריצה לא היה כלל הבדל בין ההרצות. ההרצה הטובה ביותר הייתה במודל השני עם המאפיינים של Age + Occupation. לדעתו מאפיינים אלו מתאימים יותר לצורך החיזוי. לסיכום המודל המומלץ מבחינתנו לחיזוי rating הוא המודל השני.