פרויקט גמר מדעי הנתונים

אסף יחזקאל

בני בטש

בן פרץ

שאלת מחקר 1

בהינתן כמה קריטריונים, האם ניתן לחזות מחיר דירה?

שלב ראשון:
Crawling

נעזרנו בסלניום על מנת להוציא מידע מאתר מדלן ויצרנו מספר דאטא-פריימים של שכונות בעיר תל אביב:

	Unnamed: 0	adress	date	square	price for meter	number of rooms	floor	year of build	total price
0	0	VIEW- ויו תל אביב יפו	26.10.2021	143	22,867 ₪	6.0	2	2019	3.27 מ' ₪
1	1	ארבר 13 תל אביב יפו	25.10.2021	63	38,095 ₪	3.0	8	2021	2.4 מ' ₪
2	2	VIEW- ויו תל אביב יפו	18.10.2021	104	25,836 ₪	4.0	13	2019	2.69 מ' ₪
3	3	ארבר 13 תל אביב יפו	9.9.2021	82	31,341 ₪	3.0	7	2021	2.57 מ' ₪
4	4	ארבר 13 תל אביב יפו	25.8.2021	81	32,098 ₪	3.0	9	2021	2.6 מ' ₪
...	...	...	...	...	...	...	...	...	...
987	987	גרינבוים 31	15.1.2008	63	7,038 ₪	3.0	4	1990	443.43 א' ₪
988	988	שיק 10	14.1.2008	65	6,705 ₪	3.0	2	1972	435.83 א' ₪
989	989	שקד 21	14.1.2008	90	6,111 ₪	3.0	1	1974	550 א' ₪
990	990	שז"ר זלמן 29	13.1.2008	100	9,000 ₪	4.0	5	1992	900 א' ₪
991	991	ארבר מנחם 31	1.1.2008	75	6,000 ₪	3.0	2	1970	450 א' ₪

992 rows × 9 columns

מידע על השכונות לפני הטיפול בנתונים:

	Unnamed: 0	Unnamed: 0.1	Average_rent	School_grade	Socio_economic	Cleanliness_Maintenance	Kindergarten	Parking	Public_Transport	Feeling_confident
0	0	נווה עופר	4450	4.5/10	3/10	3.9	4.3	4.6	4.5	4.0
1	1	הגוש הגדול	10500	9/10	9/10	4.4	4.4	3.7	3.5	4.7
2	2	קרית שלום	6200	7.7/10	3/10	4.1	4.2	4.3	4.2	4.7
3	3	פלורנטין	6050	3.5/10	5/10	2.0	2.3	2.9	4.1	3.5
4	4	נווה שאנן	5000	5/10	4/10	2.6	2.8	3.6	4.6	2.8
5	5	כוכב הצפון	10500	10/10	9/10	4.5	4.1	4.5	3.8	4.6

Gardens	Recreation_and_leisure	Shopping	Pedestrian_comfort
4.6	2.2	3.1	4.7
4.2	2.9	3.1	3.7
4.8	2.9	3.3	4.7
2.1	4.5	4.2	3.8
2.7	2.7	2.8	4.0
4.5	3.1	3.0	4.3

שלב שני:
Data Handling

בשלב הזה הורדנו את כל הכפילויות שהיו לנו.

שינינו את כל המשתנים הקטגוריאליים לנומריים, כגון קומה, כתובת וכו'.

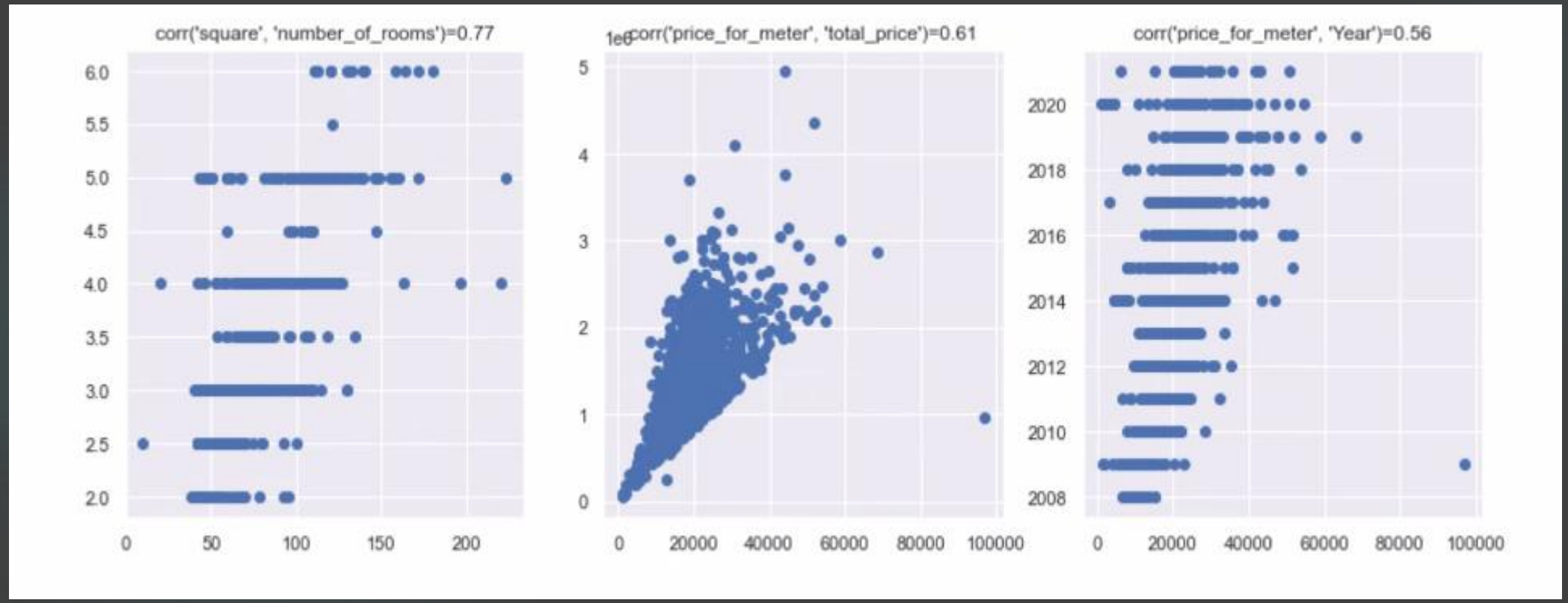
```python
def Data_h(df):
    df.drop('Unnamed: 0',axis=1,inplace=True)
    df['floor'] = df['floor'].replace('0' ,'מ')
    df['floor'] = df['floor'].str.replace(r'\D', '')
    df['floor']=pd.to_numeric(df['floor'])
   #
    df.drop(columns=['total price'],inplace =True)
    df.insert(7, 'total price', 0)
    #
    df['price for meter'] = df['price for meter'].str.replace(r'\D', '')
    df['price for meter']=pd.to_numeric(df['price for meter'])
    #
    #
    df['square'] = df['square'].str.replace(r'\D', '')
    df['square']=pd.to_numeric(df['square'])
    #
    df['total price']=df['square']*df['price for meter']
    #
    #
    df['Day'] = [d.split('.')[0] for d in df.date]
    df['Month'] = [d.split('.')[1] for d in df.date]
    df['Year'] = [d.split('.')[2] for d in df.date]
    df.drop(columns=['date'],inplace =True)
    #
    df['Day']=pd.to_numeric(df['Day'])
    df['Month']=pd.to_numeric(df['Month'])
    df['Year']=pd.to_numeric(df['Year'])
```

טיפלנו במשתנים חריגים, כפי שנלמד בהרצאות.

```
38      sns.boxplot(x = df['number of rooms'])
39
40      Q1 = np.percentile(df['number of rooms'] , 25)
41      Q3 = np.percentile(df['number of rooms'] , 75)
42      IQR = Q3 - Q1
43
44
45      df['number of rooms'][(df['number of rooms'] < Q1 - IQR ) | (df['number of rooms'] > Q3 + 2.5*IQR)] = np.nan
46      df = df[df['number of rooms'].notna()]
47
48
49
50      sns.boxplot(x = df['total price'])
51
52      Q1 = np.percentile(df['total price'] , 25)
53      Q3 = np.percentile(df['total price'] , 75)
54      IQR = Q3 - Q1
55
56
57      df['total price'][(df['total price'] < Q1 - IQR ) | (df['total price'] > Q3 + 2.5*IQR)] = np.nan
58      df = df[df['total price'].notna()]
59
60
61
62
63
64
65
66      df=df.dropna()
67      df.duplicated()
68
69
70      return df
```

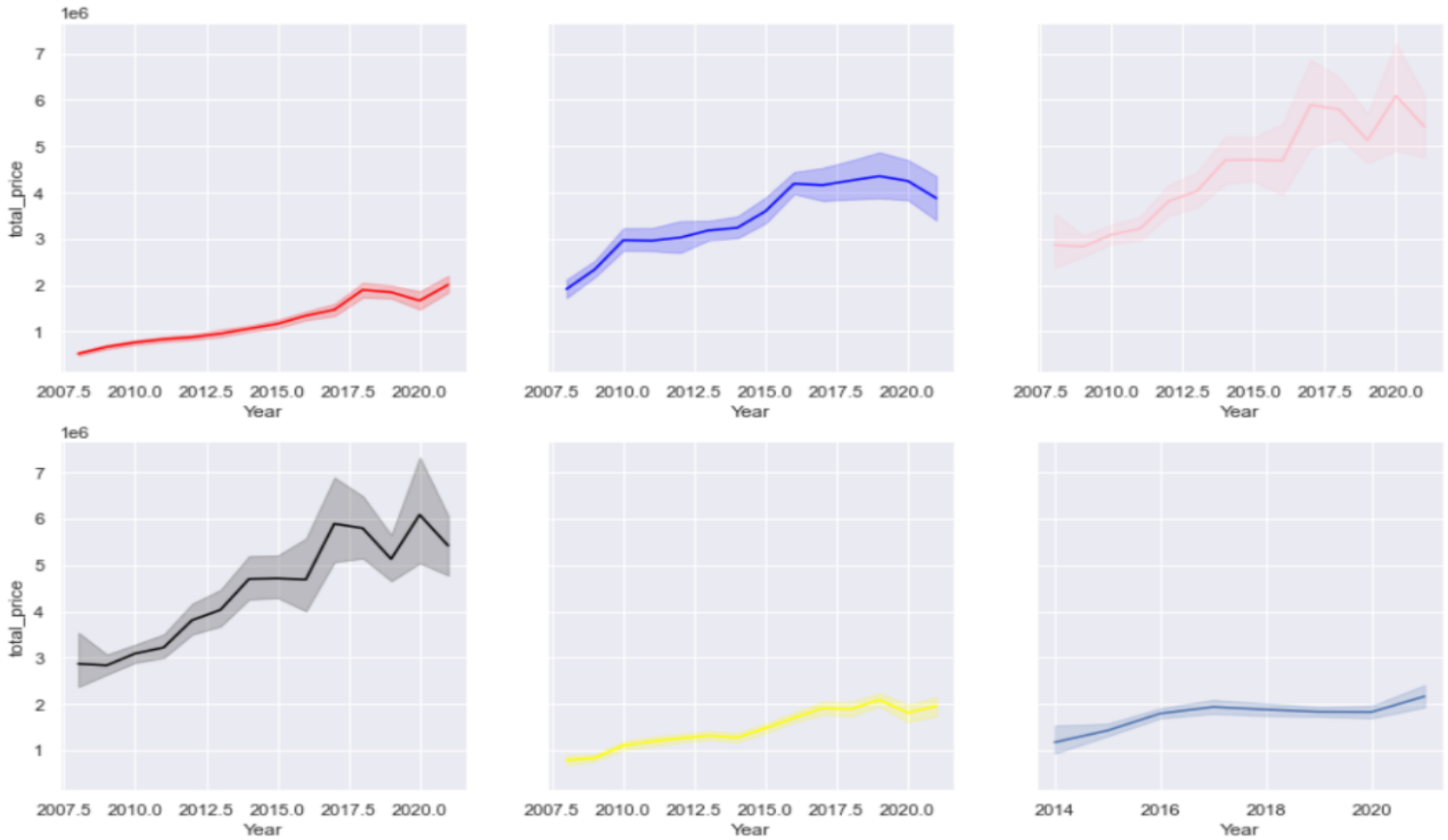
שלב שלישי:
EDA

בשלב זה השתמשנו בגרפים על מנת להציג את כל המידע שהוצאנו על הדירות בצורה ויזואלית.

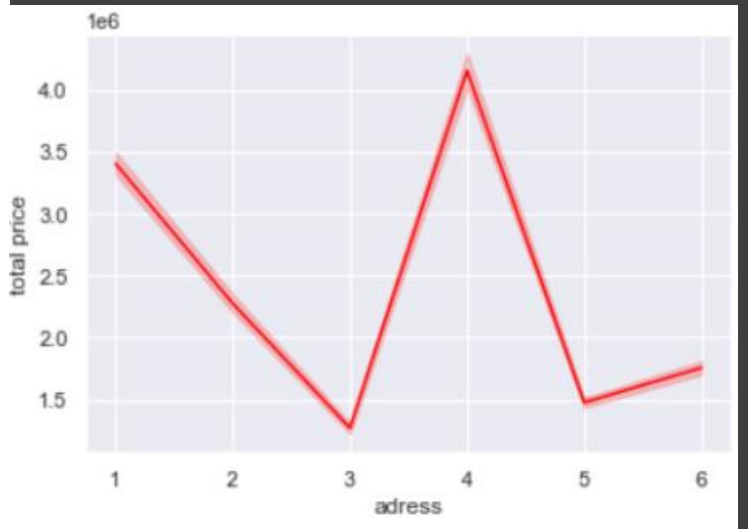


In [49]:

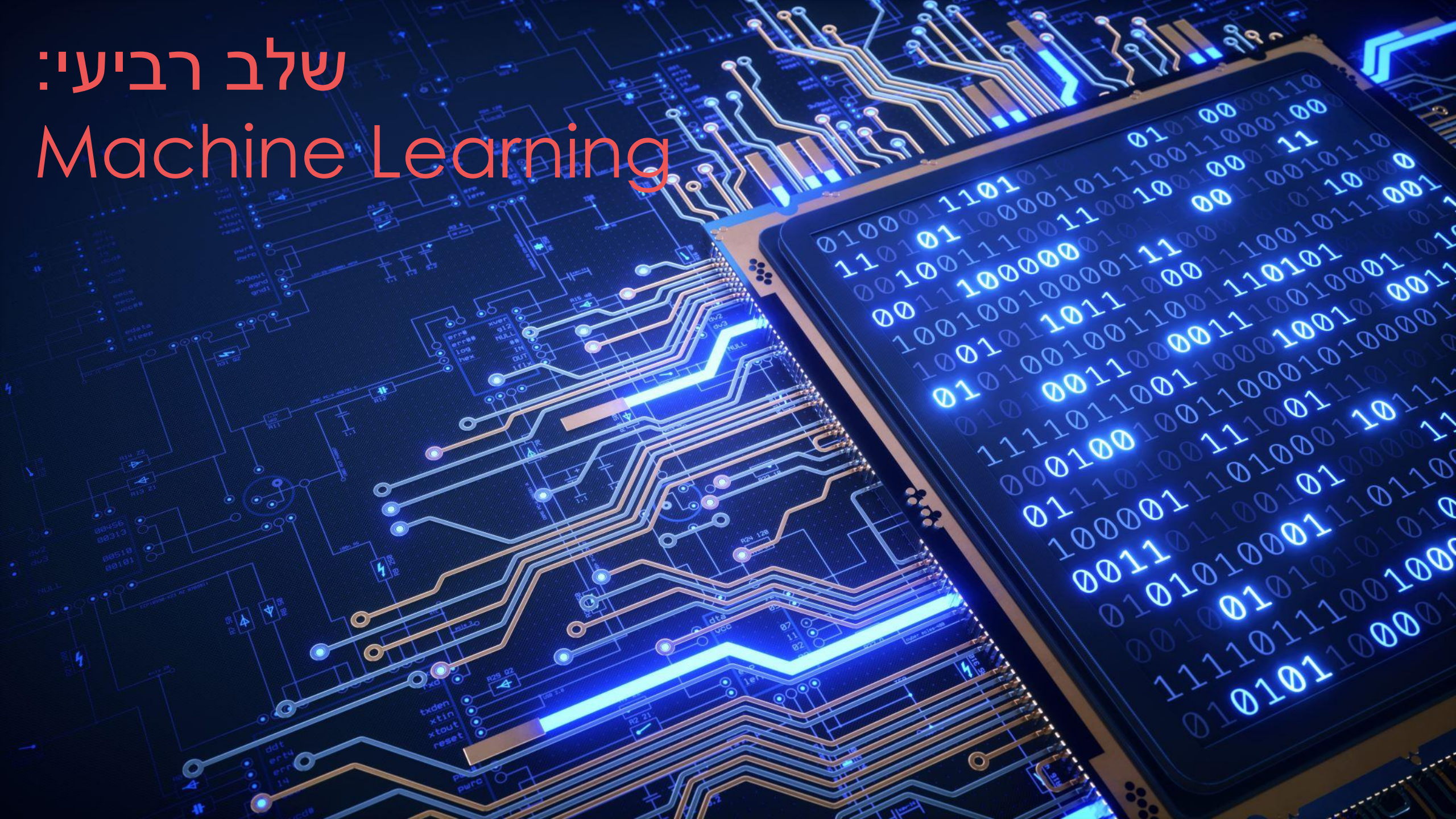
```
fig, axes = plt.subplots(2,3, figsize=(15, 10), sharey=True)


sns.lineplot(ax=axes[0][0],x='Year',y='total_price',data=df_neve_ofer,color='red')
sns.lineplot(ax=axes[0][1],x='Year',y='total_price',data=df_hagush_hagadol,color='blue')
sns.lineplot(ax=axes[0][2],x='Year',y='total_price',data=df_north_star,color='pink')
sns.lineplot(ax=axes[1][0],x='Year',y='total_price',data=df_north_star,color='black')
sns.lineplot(ax=axes[1][1],x='Year',y='total_price',data=df_kiryat_shalom,color='yellow')
sns.lineplot(ax=axes[1][2],x='Year',y='total_price',data=df_neve_shaanan)
```



בעזרת גרפים אלו ניתן לראות את התפלגות מחירי הדירות.



שלב רביעי:
Machine Learning



רגרסיה לינארית:

בשלב זה נעזרנו במודל של רגרסיה לינארית ובהסתמך על הקשרים חזינו את מחירי הדירות העתידיים.

דיוק המודל:

In [84]:

```
X = dfN.drop(columns = ['total_price','Day','Month','adress','price_for_meter'])
y = dfN['total_price']


linreg = LinearRegression()
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.2)

linreg.fit(X_train,y_train)
y_pred = linreg.predict(X_test)
evaluate_value = r2_score(y_test,y_pred)
evaluate_value
```

Out[84]: 0.749467810278869

חיזוי מחיר עתידי:

In [80]:

```
print('Enter square meters:')
square = input()
print('Enter number of rooms:')
number_of_rooms = input()
print('Enter floor:')
floor = input()
print('Enter year build:')
year_of_build = input()
print('Enter year :')
Year = input()
df = pd.DataFrame({"square":square,"number_of_rooms":number_of_rooms,"floor":floor,"year_of_build":year_of_build,"Year":Year
y_pred = linreg.predict(df)
print(y_pred)
```

```
Enter square meters:
100
Enter number of rooms:
5
Enter floor:
2
Enter year build:
2020
Enter year :
2024
[2892519.04207063]
```

שאלת מחקר 2

האם שכונה עדיפה למגורי משפחות?

הוספנו לדאטה-פריים עמודות מידע רלוונטיות לשאלה, ובעזרת מידע זה באפשרותנו לחזות האם שכונה מיועדת למגורי משפחות.

	adress	square	price for meter	number of rooms	floor	year of build	total price	Day	Month	Year	School_grade	Kindergarten	Feeling_confident	Gardens	fam
0	1	147.0	40816.0	5.0	7.0	2019	5999952.0	5	10	2021	4.5	4.4	4.7	4.2	1
1	1	112.0	39732.0	4.0	2.0	2015	4449984.0	18	8	2021	4.5	4.4	4.7	4.2	1
2	1	88.0	46590.0	3.0	1.0	2016	4099920.0	10	8	2021	4.5	4.4	4.7	4.2	1
3	1	88.0	46590.0	3.0	1.0	1980	4099920.0	9	8	2021	4.5	4.4	4.7	4.2	1
4	1	98.0	40816.0	4.0	5.0	2009	3999968.0	9	8	2021	4.5	4.4	4.7	4.2	1
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
979	6	50.0	22948.0	2.0	4.0	2016	1147400.0	16	12	2014	2.5	2.8	2.8	2.7	0
980	6	50.0	22439.0	2.0	2.0	2016	1121950.0	16	12	2014	2.5	2.8	2.8	2.7	0
981	6	76.0	14736.0	3.0	3.0	1958	1119936.0	16	12	2014	2.5	2.8	2.8	2.7	0
982	6	60.0	7333.0	3.0	0.0	1960	439980.0	16	12	2014	2.5	2.8	2.8	2.7	0
983	6	56.0	17142.0	4.0	1.0	1961	959952.0	15	12	2014	2.5	2.8	2.8	2.7	0

5712 rows × 15 columns

In [10]:
```
dfQ1=pd.concat([df_hagush_hagadol, df_florentin, df_neve_ofer, df_north_star, df_kiryat_shalom, df_neve_shaanan], axis=0)

#dfQ1
```

In [14]:
```
dfQ1['fam']=0
for i in range (5712):
    if(dfQ1.iloc[i, 13] <3 or dfQ1.iloc[i, 12]<3 or dfQ1.iloc[i, 11]<2.5 or dfQ1.iloc[i, 10]<2.45):
        dfQ1.iloc[i, 14] = 0
    else:
        dfQ1.iloc[i, 14] = 1
#dfQ1
```

In [15]:
```
X = dfQ1.drop(['fam'], axis=1)
y = dfQ1['fam']
```

In [16]:
```
xtrain, xtest, ytrain, ytest = train_test_split(X, y, test_size=0.2, random_state=0)

```

In [17]:
```
clf = LogisticRegression(solver='lbfgs', max_iter=20000) # max_iter is the number of itrations it takes for all centroids to
clf.fit(xtrain, ytrain)

y_pred = clf.predict(xtest)
```

In [18]:
```
print("Accuracy is:",metrics.accuracy_score(ytest, y_pred))

```

```
Accuracy is: 0.7891513560804899
```

שאלת מחקר 3

"היצע וביקוש"

מצא את הדירה המתאימה עבורך

ביצענו תנאים שנותנים אינדיקציה מתוך הדאטה פריים אשר עונים על השאלה "מהן הדירות המתאימות עבורך?"

In [6]:

```
print('Enter square meters:')
square = input()
print('Enter number of rooms:')
number = input()
print('Enter your price:')
price = input()

for i in range(5712):
    if df_ALL.iloc[i,1 ] < int(square)+20 and df_ALL.iloc[i,1 ] > int(square)-10 and df_ALL.iloc[i,3 ] == int(number)
    and df_ALL.iloc[i,6 ]<=int(price) and df_ALL.iloc[i,5 ]>=2020 :
        print(df_ALL.iloc[i])
```

```
Enter square meters:
100
Enter number of rooms:
4
Enter your price:
3500000
adress                  1.0
square                100.0
price for meter      9652.0
number of rooms         4.0
floor                   4.0
year of build        2020.0
total price        965200.0
Day                    21.0
Month                   3.0
Year                 2019.0
Name: 107, dtype: float64
adress                   2.0
square                 102.0
```

להלן הדירות המתאימות עבורך