



# PORTUGAL REAL ESTATE DATA SCIENCE PROJECT

**ROY SHLOMO CHEN & YARIN AKIVA**



# WHAT IS REAL ESTATE

Real estate is a term that encompasses the buying, selling, and renting of land and buildings.

It involves the physical land and buildings, as well as the rights and privileges associated with them.

Real estate can be residential, commercial, or industrial, and can include the purchase and sale of land, buildings, and other improvements.

Real estate can also include the management of rental properties and other services related to the industry.



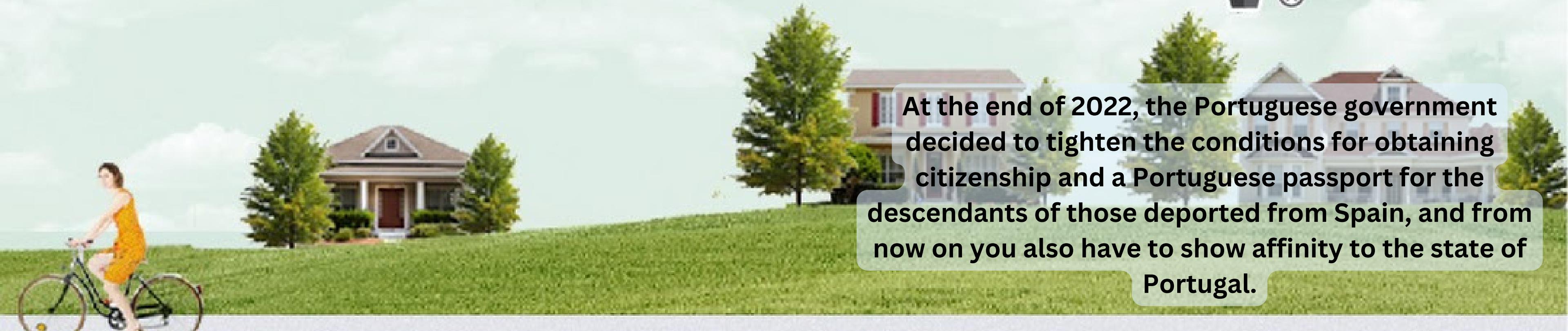


# WHY?

In recent years, Jews who could prove that they were deported from Spain, could submit an application to the Portuguese authorities and the Jewish community living in Porto or Lisbon, and ask for citizenship in Portugal only by proving that they are descendants of those deported from Spain.



At the end of 2022, the Portuguese government decided to tighten the conditions for obtaining citizenship and a Portuguese passport for the descendants of those deported from Spain, and from now on you also have to show affinity to the state of Portugal.





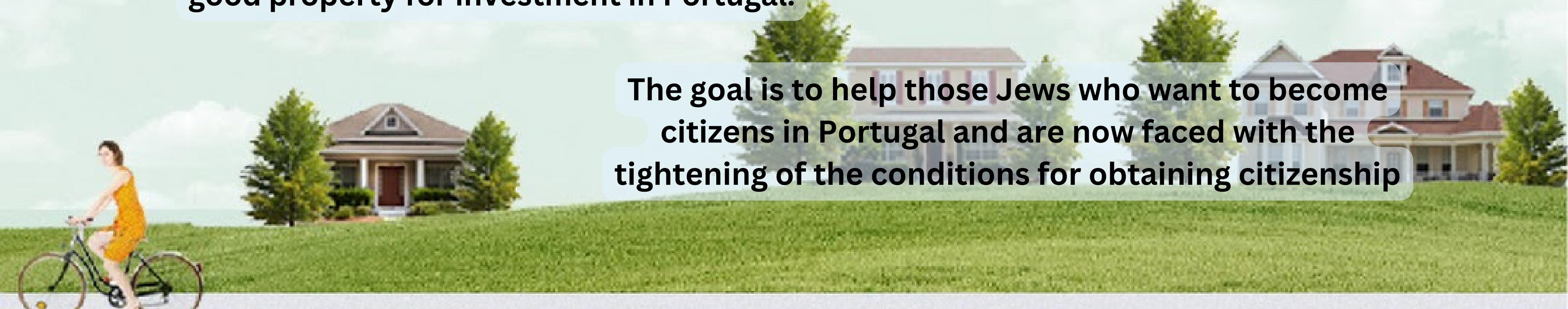
# WHY?

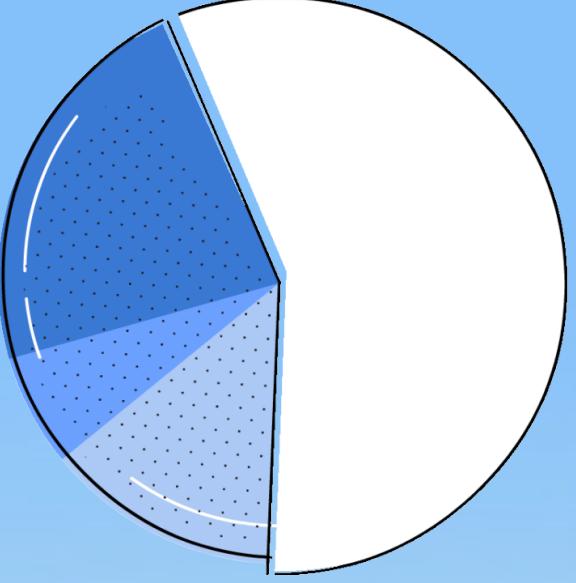
One of these ways is to invest or purchase real estate in Portugal.

We decided to check whether it is possible to analyze the information on properties in Portugal using the Remax Portugal website, and to determine what is a good property for investment in Portugal.



The goal is to help those Jews who want to become citizens in Portugal and are now faced with the tightening of the conditions for obtaining citizenship





# **RESEARCH QUESTION**



**How to identify a good  
investment property in Portugal?**

# MAIN PROCESS STEPS

- 
- 1
  - 2
  - 3
  - 4
  - 5

## Obtaining Data

web scraping for collection , and obtaining the data

## Data Handling

Cleaning,formattin and filtering the data, removing duplicates data

## Exploring Data

Visualizing and understandaing the data

## Machine Learning

Clustering the data in groups and modeling , and apply ML modeling on the data

## Interpretin Data

Presentation of data,understanding and delivering the results

# OBTAINING DATA

In this step, we will collect information of properties in Portugal in various big cities

we will crawl along web pages and scrape information about properties features

Main tools : BeautifulSoup, Selenium

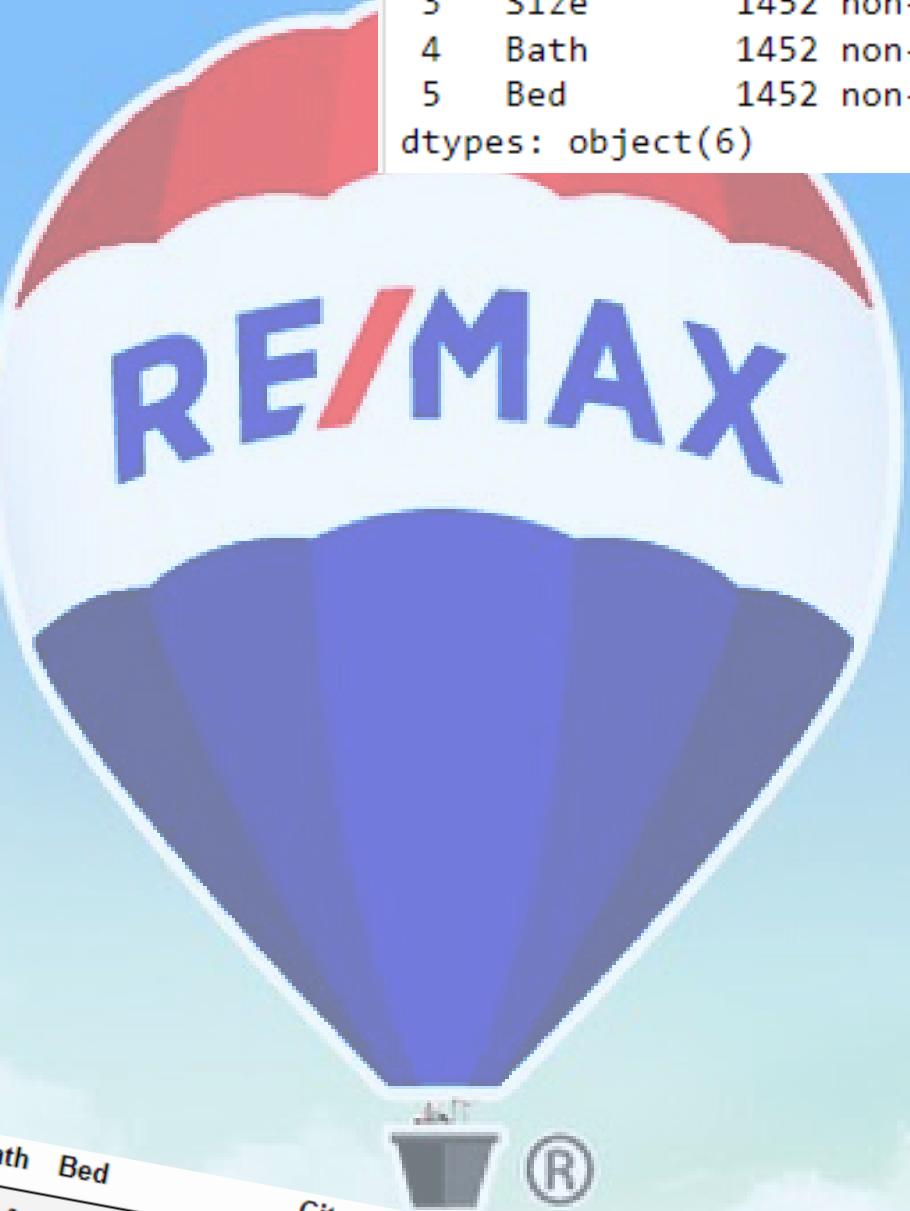
Source Data : <https://www.remax.pt/en/>

Price	Address	Prop_type	Size	Bath	Bed
255 000 €	Porto - Vila Nova de Gaia, São Felix da Mar...	Condo/Apartment	105 m2	2	2
198 500 €	Porto - Paços de Ferreira, Paços de Ferreira	Condo/Apartment	132 m2	2	3
138 000 €	Porto - Vila Nova de Gaia, Grijó e Sermonde	Condo/Apartment	97 m2	2	2
3 225 000 €	Porto - Porto, Paranhos	Condo/Apartment	89 m2	1	1
4 225 000 €	Porto - Porto, Paranhos	Condo/Apartment	53 m2	1	1

```

0    Price      1452 non-null   object
1    Address    1452 non-null   object
2    Prop_type  1452 non-null   object
3    Size       1452 non-null   object
4    Bath       1452 non-null   object
5    Bed        1452 non-null   object
dtypes: object(6)

```



To collect each property, we used Selenium because this is a dynamic scrapping,  
 After we passed the dynamic scrapping difficulty  
 We used BeautifulSoup to store the data in dataframes



	Price	Prop_type	Size	Bath	Bed	City	Division
0	295000.0	House	140.0	2.0	3.0	Porto	Lordelo do Ouro e Massarelos
1	22500.0	House	46.0	0.0	2.0	Baião	Ancede e Ribadouro
2	485000.0	House	152.0	3.0	4.0	Vila do Conde	Vila Chã
3	1250000.0	House	527.0	0.0	10.0	Porto	Aldoar
4	67000.0	House	75.0	1.0	2.0	Paços de Ferreira	Carvalhosa
...	...	...	...	...	...	...	...
11660	135000.0	Condo/Apartment	22.0	1.0	0.0	...	São Vicente
11661	130000.0	Condo/Apartment	18.0	1.0	0.0	Braga	São Vicente
11662	139900.0	Condo/Apartment	110.0	2.0	3.0	Barcelos	Barcelos
11663	799000.0	Condo/Apartment	720.0	12.0	7.0	Vieira do Minho	Guilhofrei
11664	170000.0	Condo/Apartment	113.0	2.0	2.0	Braga	(São José de São Lázaro e São João do Souto)
11665	rows x 7 columns						

```
Address      object  
Prop_type    object  
Size         object  
Bath         object  
Bed          object  
dtype: object
```

# DATA HANDLING

In this step, we cleaning, formattion and filtering the data.

- Splitting columns
- Remove duplicates rows and cols
- Remove unwanted marks ( € , '/' )
- Edit Types of variabels



```
buy['Price'] = buy['Price'].str.replace(' ', '')  
buy['Price'] = buy['Price'].str.replace('€', '')  
buy['Address'] = buy['Address'].str.replace('Porto', '')  
buy['Address'] = buy['Address'].str.replace('Lisboa', '')  
buy['Address'] = buy['Address'].str.replace('Braga', '')  
buy['Address'] = buy['Address'].str.replace('-', '')  
buy['Address'] = buy['Address'].str.split(',', expand=True)[0]  
buy['City'] = buy['Address'].str.split(',', expand=True)[1]  
buy['Division'] = buy['Address'].str.strip() # removing Leading and tailing white spaces  
buy['City'] = buy['Size'].str.replace('m2', '')
```



For each column we change the type from category to numeric , help in the next step and make more easiest to work on the data

```
buy['Price'] = pd.to_numeric(buy['Price'], errors='coerce')
buy['Size'] = pd.to_numeric(buy['Size'], errors='coerce')
buy['Bath'] = pd.to_numeric(buy['Bath'], errors='coerce')
buy['Bed'] = pd.to_numeric(buy['Bed'], errors='coerce')
```

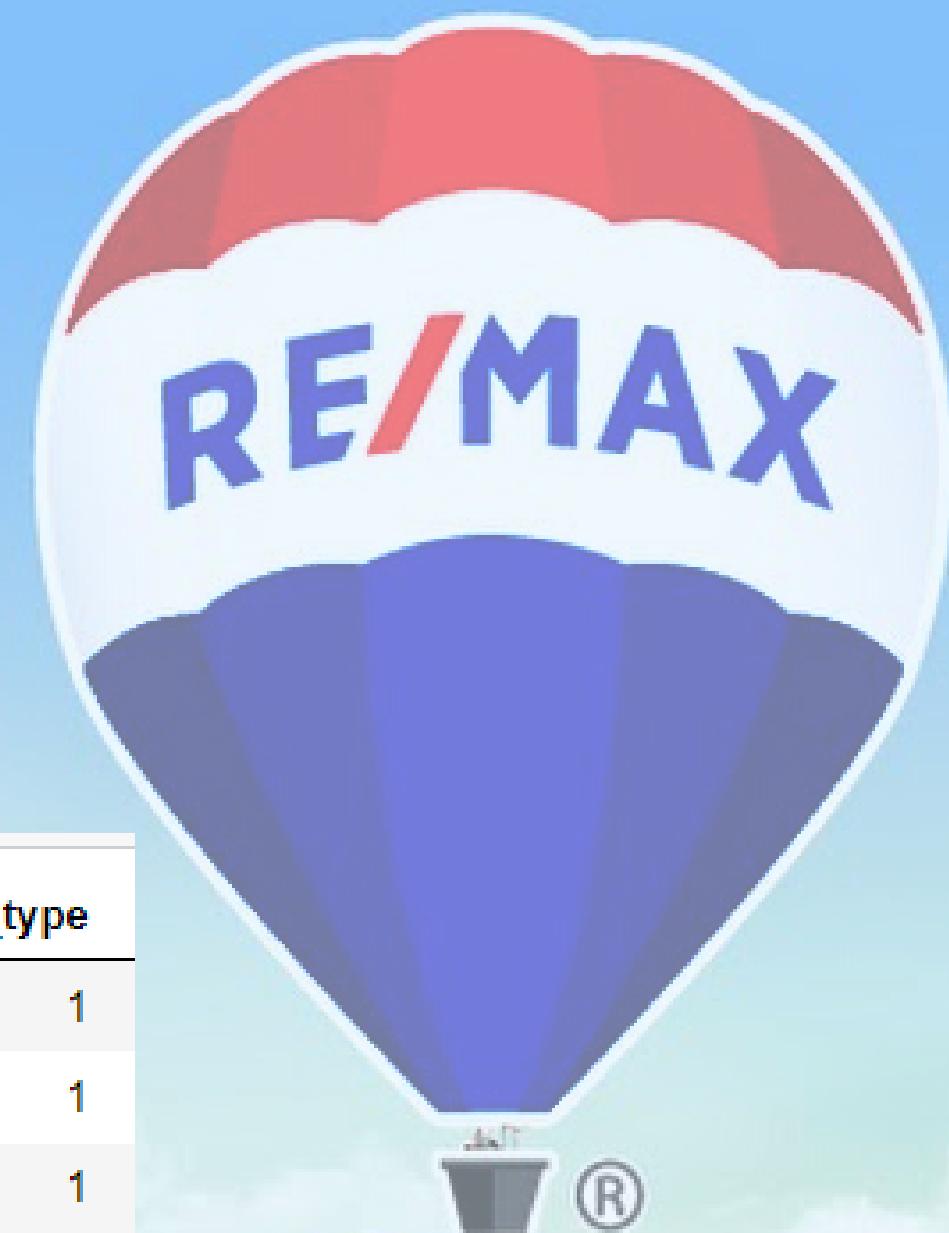
Using replace map  
we change all the property type to number  
1 to House and 2 to apartment

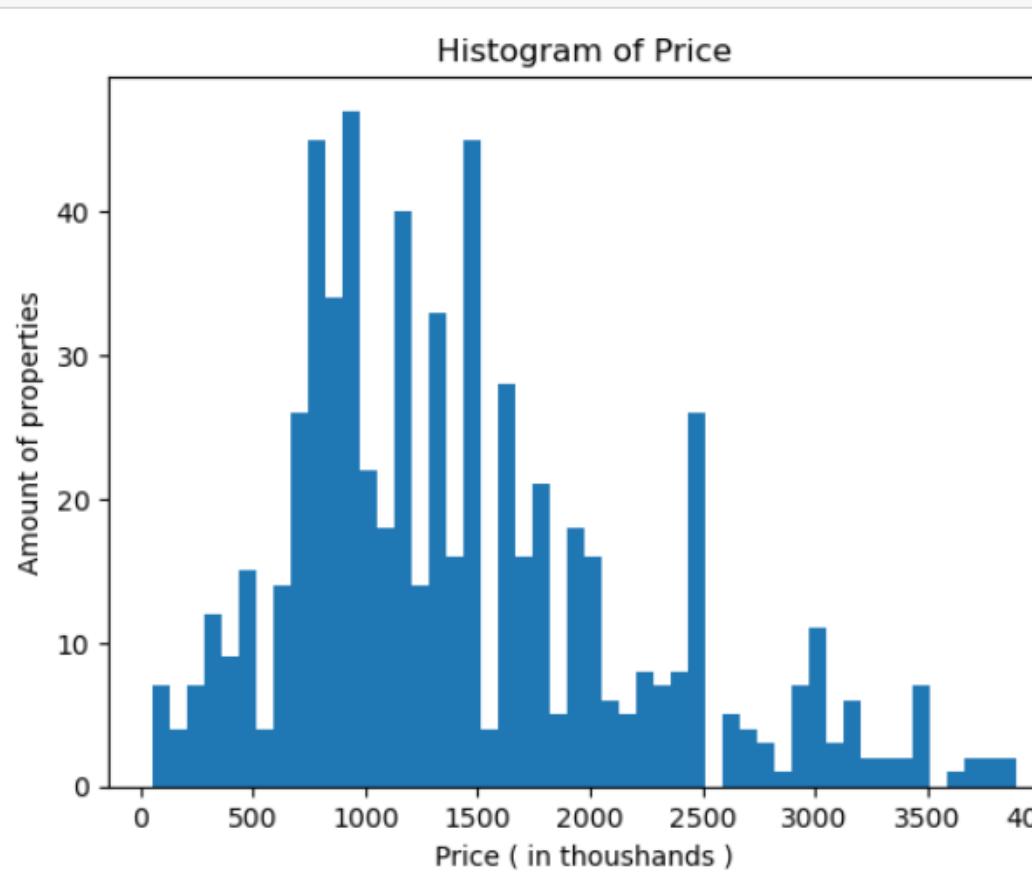
```
replace_map={'House':1,'Condo/Apartment':2}
buy2.replace(replace_map, inplace=True)
```

Prop_type
House
...
Condo/Apartment



Prop_type
1
1
1
1
1
...
2
2
2
2
2



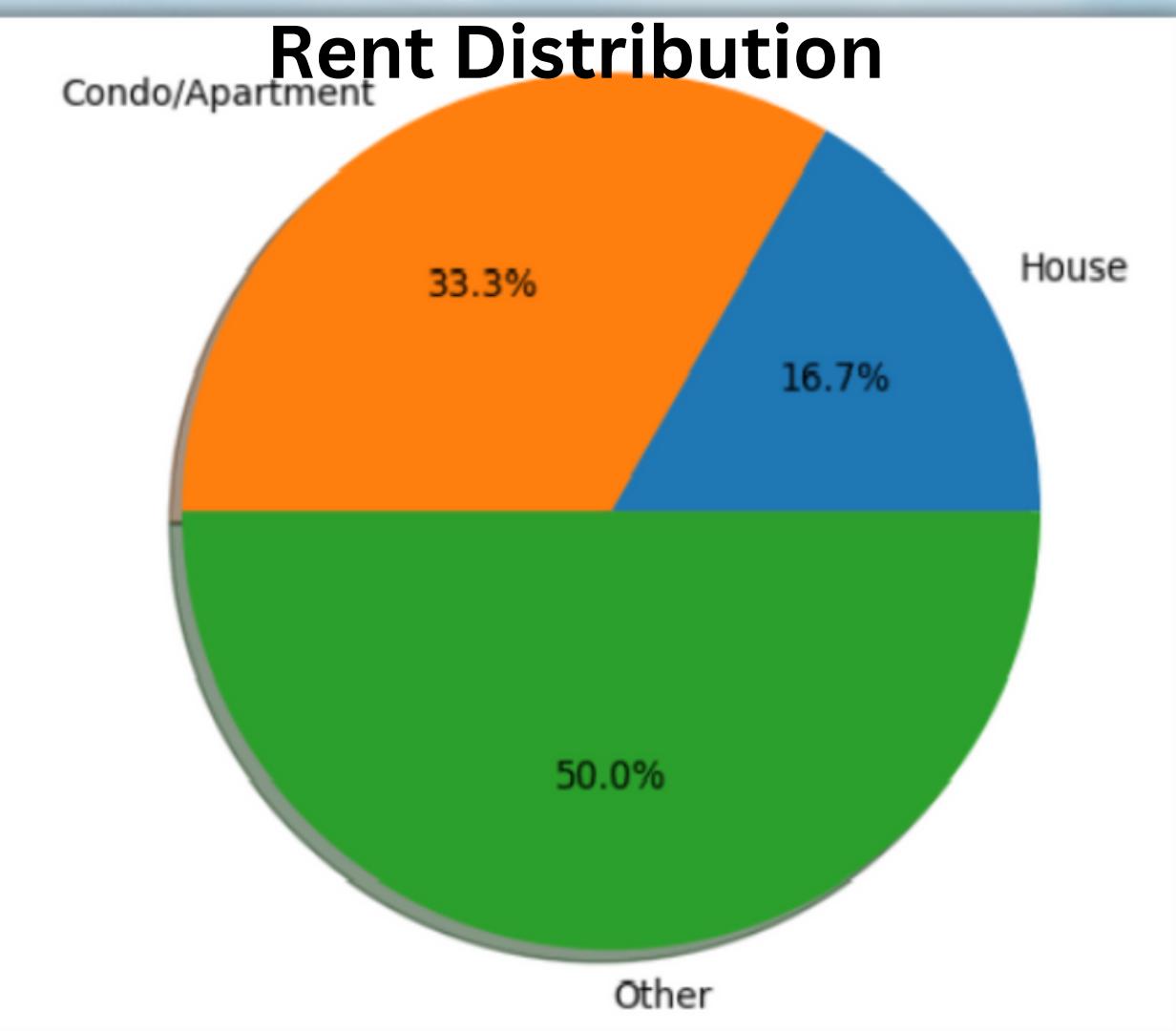
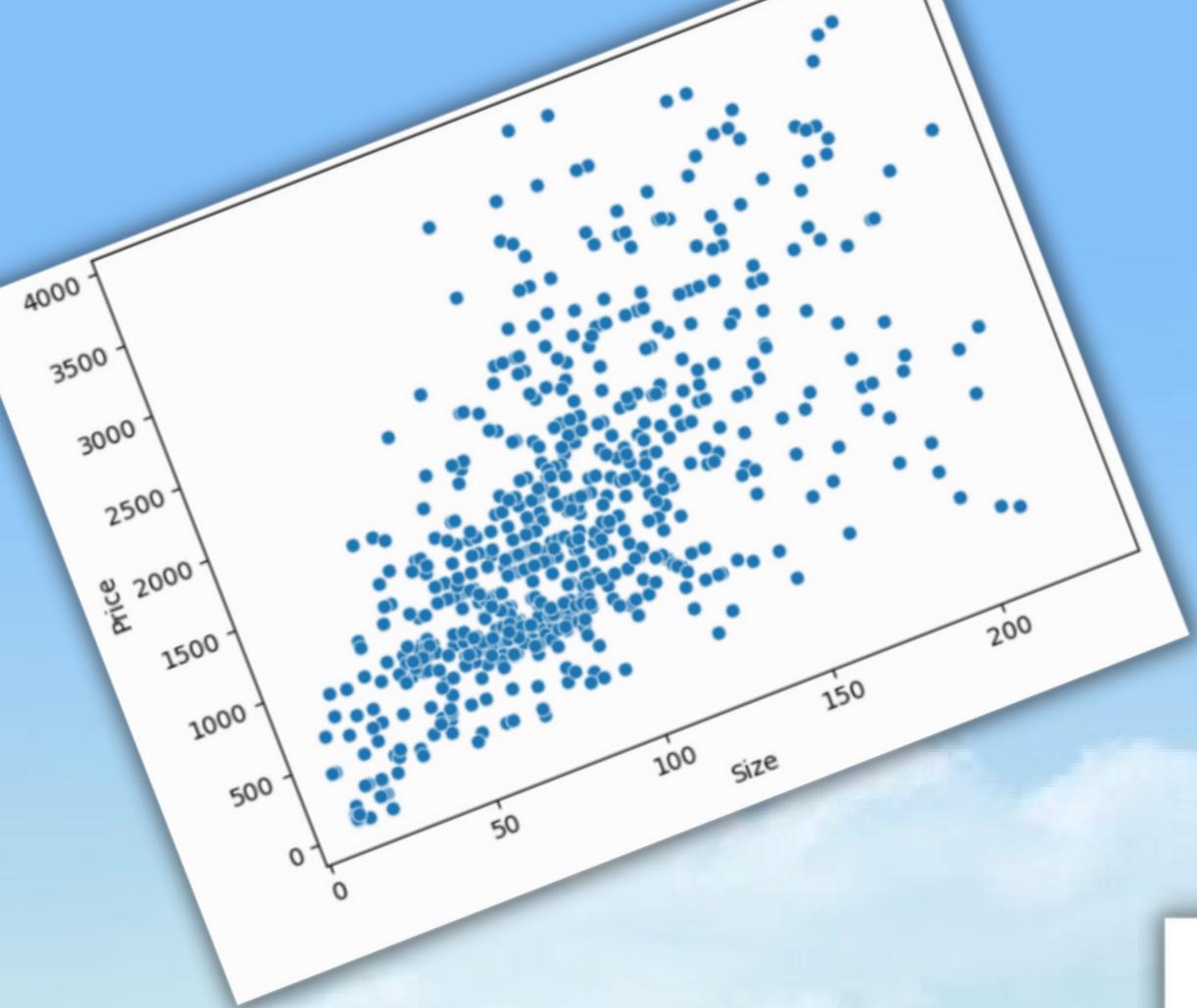


# EXPLORING DATA



We did some visualization with our data  
after all the cleaning  
with this EDA we did outliers treatment  
and remove unnecessary





	coefficients
Size	488.658274
Bath	177.944244
Bed	54.404818
City_Lisboa	-17.258632
City_Oeiras	-221.661878
City_Other	-798.581239
City_Porto	-512.934645
City_Sintra	-612.063164

# MACHINE LEARNING

## Supervised Learning :

Supervised learning is a type of machine learning in which the computer is provided with labeled data and it is expected to generate a predictive model. This model is then used to make predictions about new data. Supervised learning can be used to classify data, predict outcomes, and identify patterns.



```
# strat train the model , splitting to 20/80
```

```
X = rent2.drop(columns = ['Price'])  
y = rent2['Price']
```

```
# 20% of the data will be used for testing and 80% will be used for training.
```

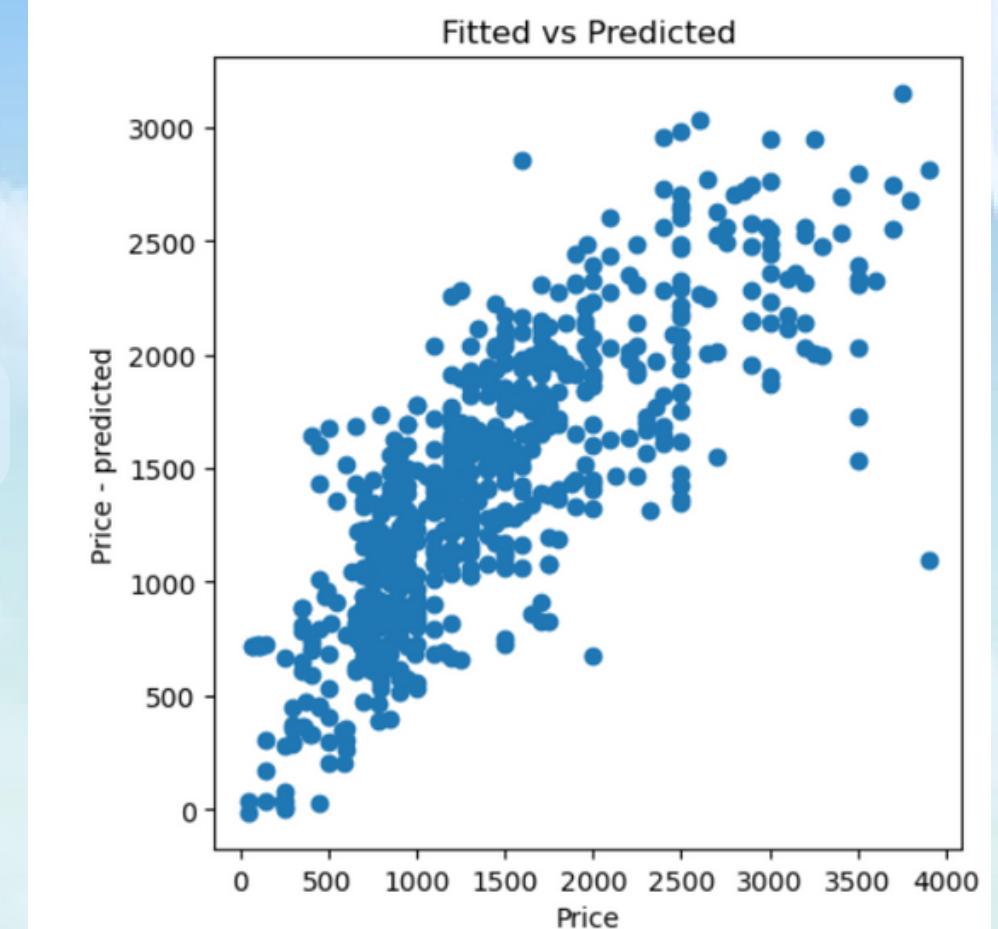
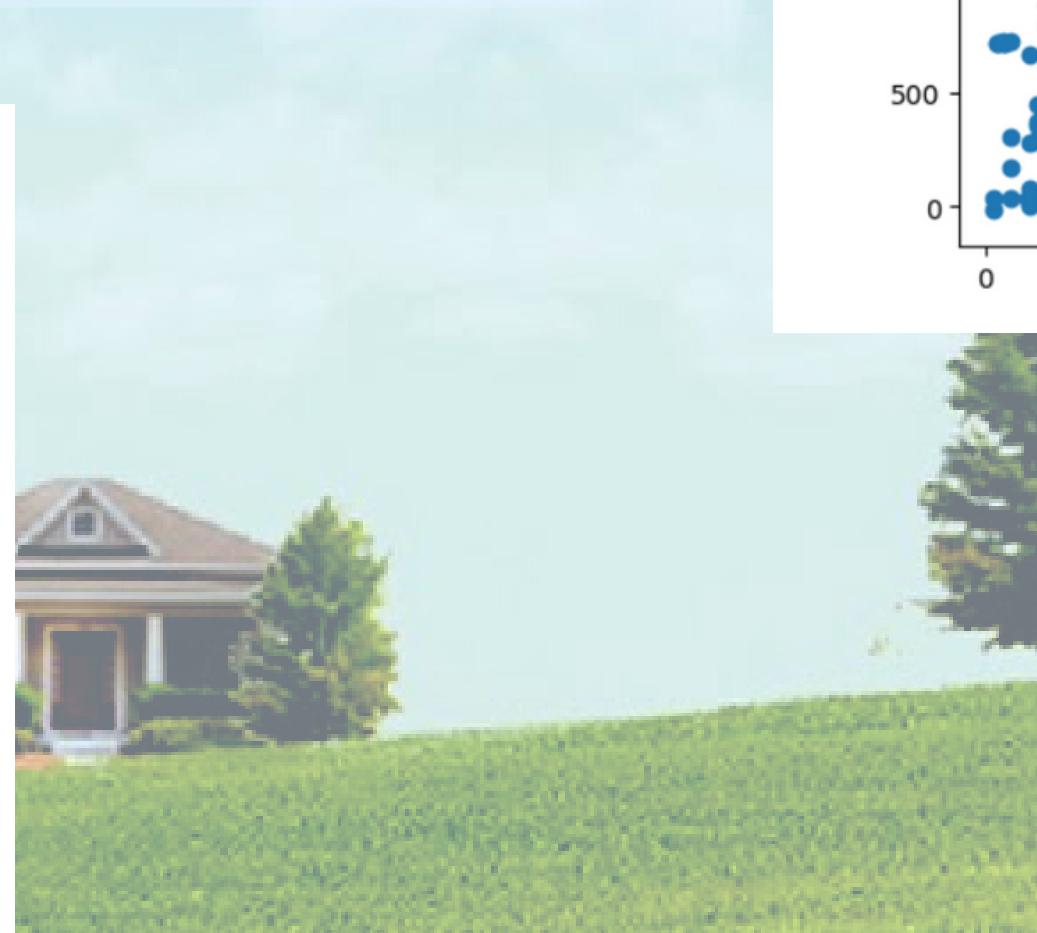
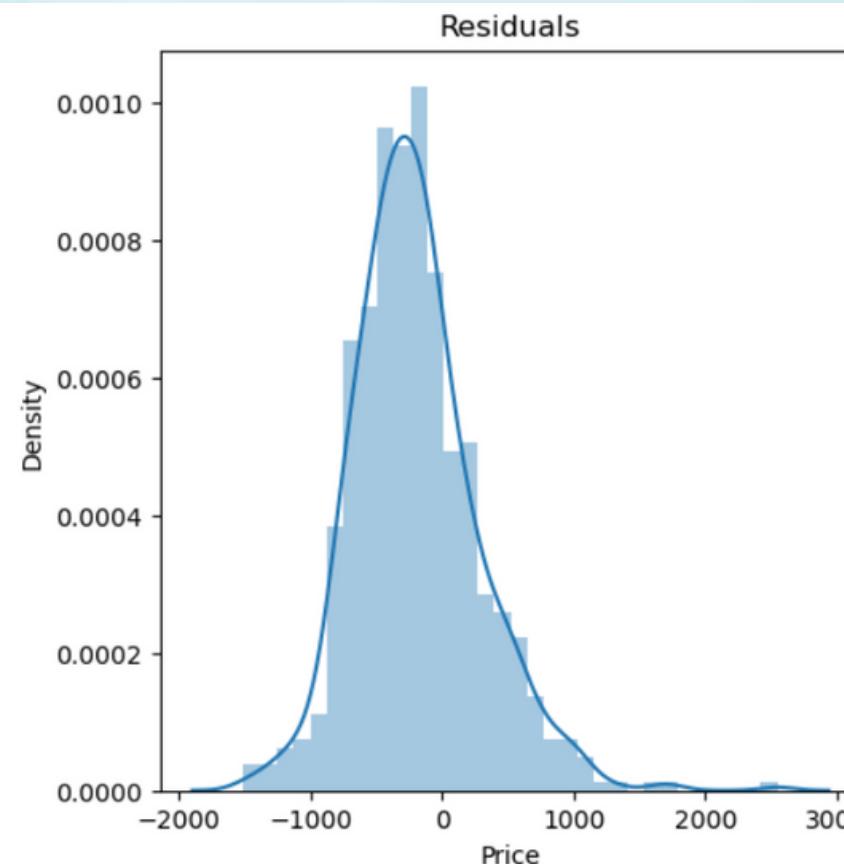
```
X_train, X_test, y_train, y_test = train_test_split(X,y, test_size=0.2)
```

```
col = ['Size', 'Bath', 'Bed'] # used to scale data in the same range
```

```
scaler = RobustScaler().fit(X_train[col]) # RobustScaler() to scale the columns 'Size', 'Bath', and 'Bed' in the training set
```

```
X_train[col] = scaler.transform(X_train[col])
```

**Supervised Learning :**  
we use 20% of the data to test  
and 80% to train  
our X column was the Bed,Size and Bath  
and our Y column wash the Price



	Size	Bath	Bed	City_Lisboa	City_Oeiras	City_Other	City_Porto	City_Sintra
560	0.136364	0.0	0.5	1	0	0	0	0
601	0.827273	0.0	1.0	1	0	0	0	0
320	-0.263636	-1.0	-0.5	1	0	0	0	0
374	2.027273	-2.0	-1.0	1	0	0	0	0
254	1.281818	1.0	0.5	0	0	1	0	0



```
# checking R2 for train/test
# The R2 score is a measure of how well the Linear regression model fits the data

print(lm_scaled.score(X_train, y_train)) # This line prints the R2 score for the training dataset
print(lm_scaled.score(X_test, y_test)) # This line prints the R2 score for the testing dataset
```

```
0.6270767613029067
0.5546334818921739
```

## Supervised Learning :

we test and train first the rent data  
to understand how can we now  
to calculate the predicted rent

while we give to the program the price property  
and his details such as bed, bath, size.



## Supervised Learning :

we start to train and test when we scaled the data  
and after we did the same without scaling ( without  
modeling on the Z-score )

```
print('Mean Absolute Error:', metrics.mean_absolute_error(y_test, y_pred))
print('Mean Squared Error:', metrics.mean_squared_error(y_test, y_pred))
print('Root Mean Squared Error:', np.sqrt(metrics.mean_squared_error(y_test, y_pred)))
```

```
Mean Absolute Error: 453.2954578178126
Mean Squared Error: 354860.86853835644
Root Mean Squared Error: 595.7019964196497
```

## Supervised Learning :

While we get our linear regression equation

we load the buy data

and fit the data to look like the same data

as the rent .

After we train the Price data

with the linear regression predict we got previously

and then we get the predicted rent column about the

buy data

	Price	Size	Bath	Bed	City_Cascais	City_Lisboa	City_Other	City_Porto	City_Sintra	predicted_rent
0	295000	140	2	3	0	0	0	1	0	1761.992029
1	22500	46	0	2	0	0	1	0	0	387.169896
2	485000	152	3	4	0	0	1	0	0	1831.693994
3	1250000	527	0	10	0	0	0	1	0	5211.002405
4	67000	75	1	2	0	0	1	0	0	766.781801
5	147000	475	4	4	0	0	1	0	0	4649.000758
6	280000	120	4	3	0	0	1	0	0	1631.450533
7	459000	149	8	3	0	0	1	0	0	2428.539470
8	150000	160	1	3	0	0	1	0	0	1545.632665
9	162000	463	0	5	0	0	1	0	0	4066.942106



$x = \text{buydf.drop(columns=}$   
 $\text{buydf[} \text{'predicted_rent'} \text{]} = [\text{'Price'}])$   
 $\text{lm.predict(x)}$



The last step  
is to load the buy data after adding the predicted rent  
column.

we calculate with the Portugal formula to  
calculate the monthly mortgage and other  
monthly expense , such as insurance or property tax  
and make one column that shows , how much  
profit the property owner stay with at the end of the  
month.

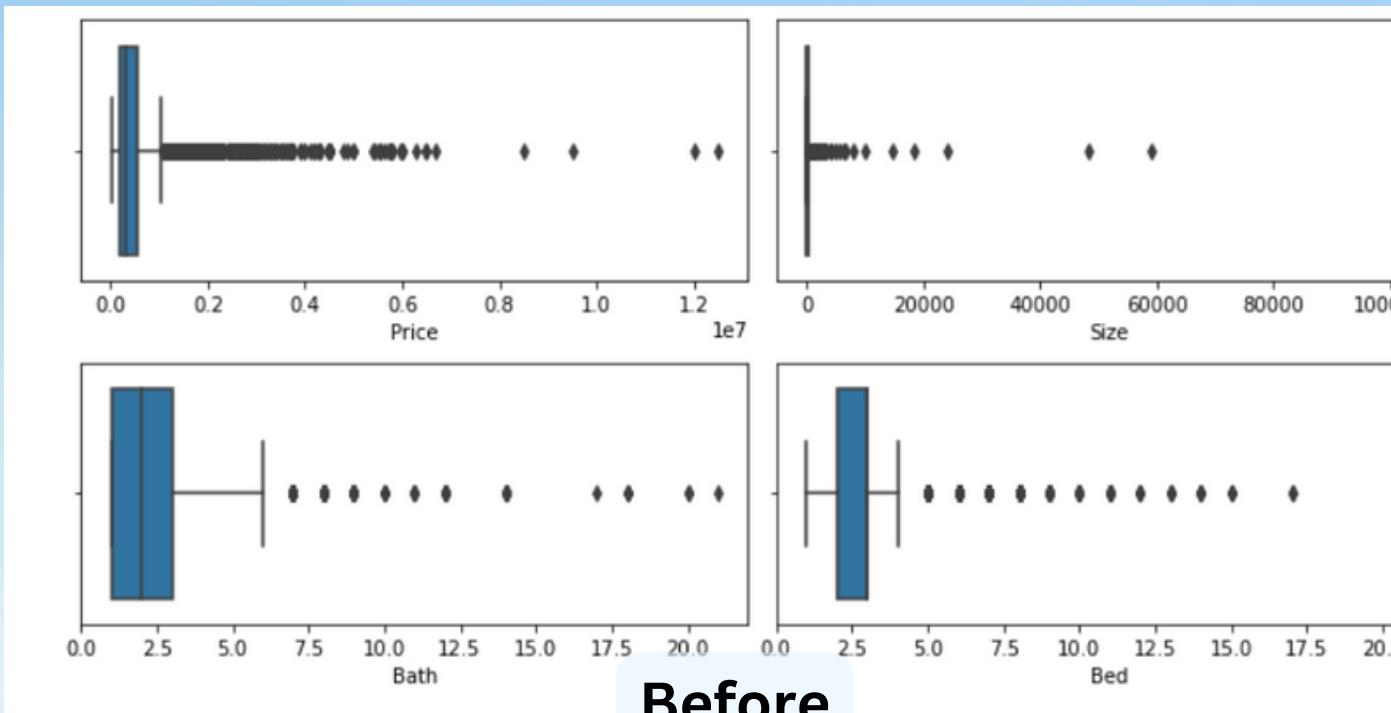


	Price	Size	Bath	Bed	City_Cascais	City_Lisboa	City_Other	City_Porto	City_Sintra	Monthly_Profit
0	295000	140	2	3	0	0	0	1	0	-286.157971
1	22500	46	0	2	0	0	1	0	0	231.344896
2	485000	152	3	4	0	0	1	0	0	-1535.756006
3	1250000	527	0	10	0	0	0	1	0	-3468.497595
4	67000	75	1	2	0	0	1	0	0	302.391801

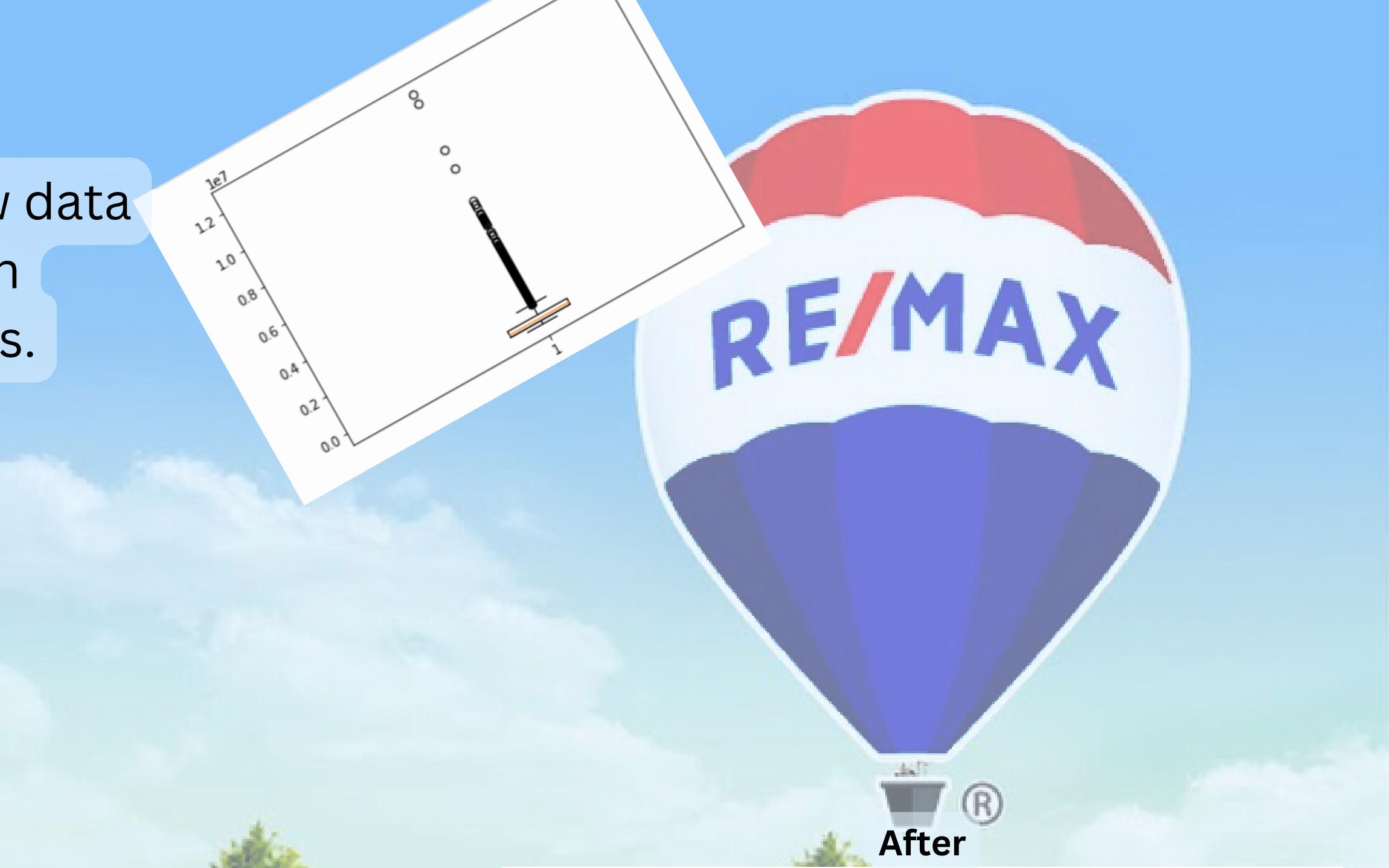
## Eda and Outliers treatment:

We did again

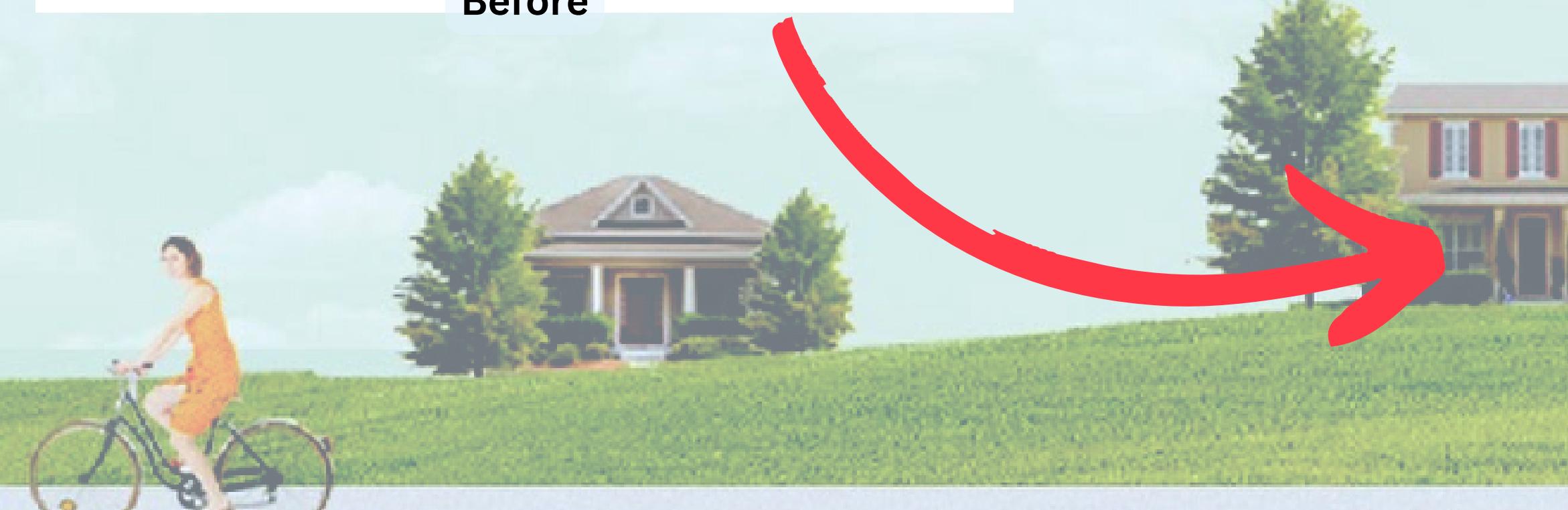
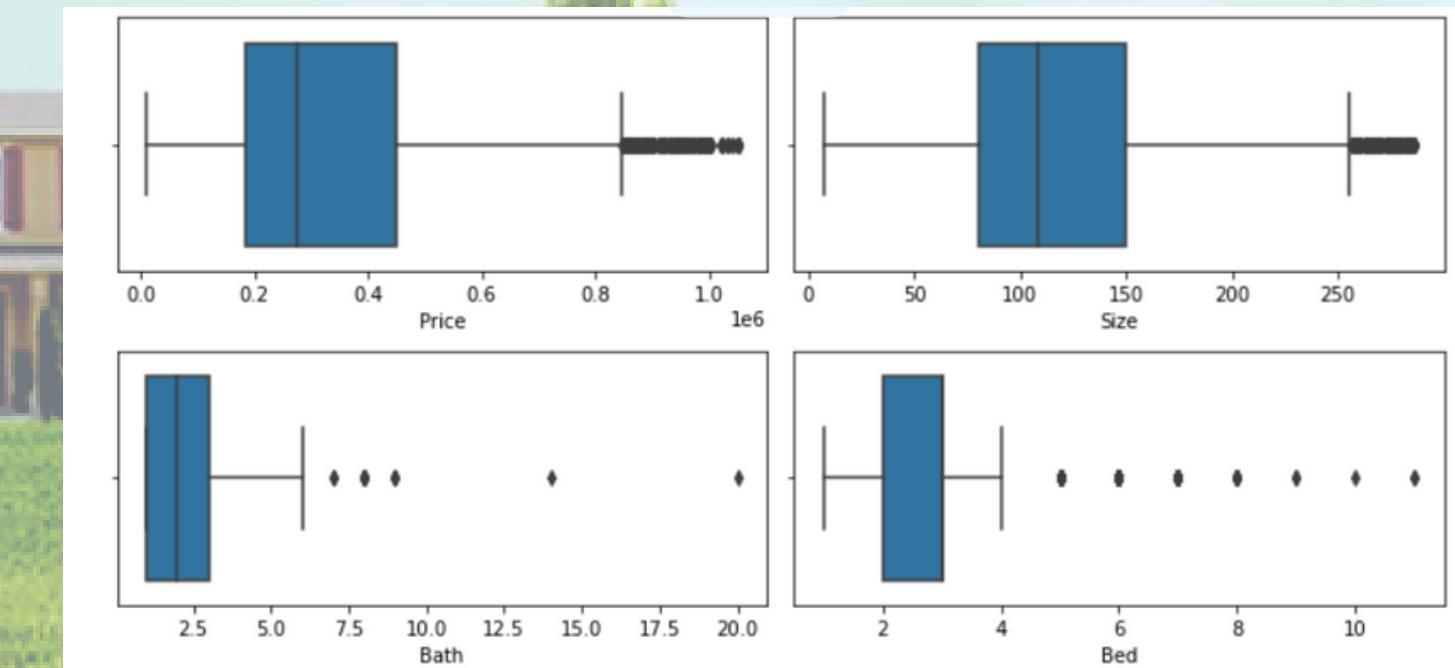
Eda and outliers treatment on the new data  
to make a cleaner data to work on  
with the machine learning modules.



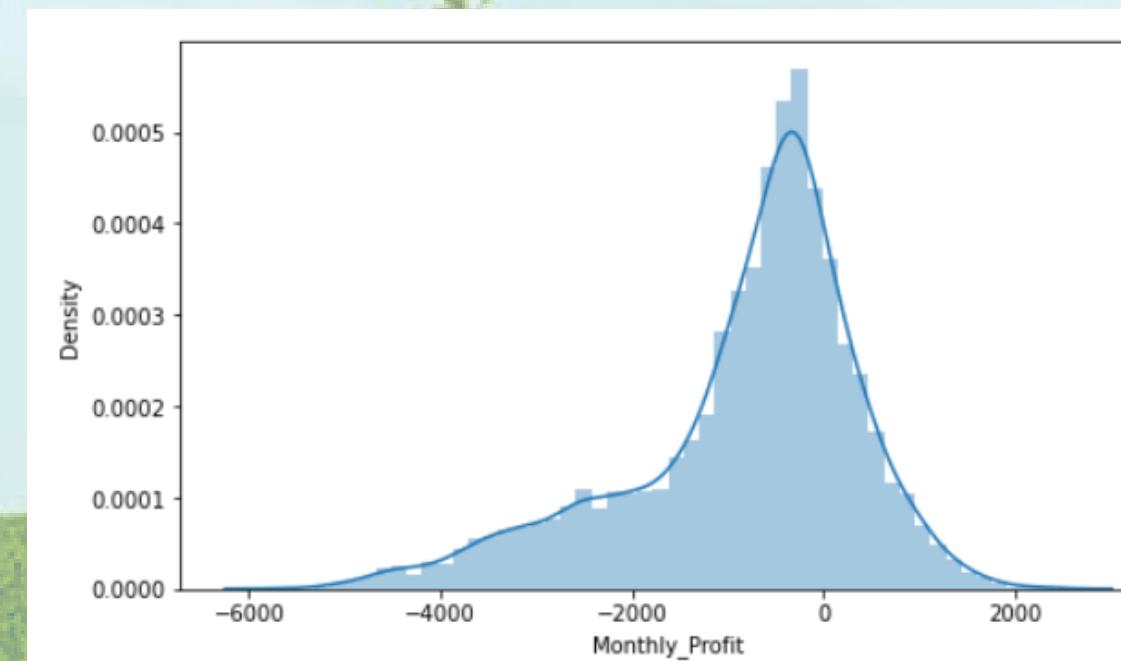
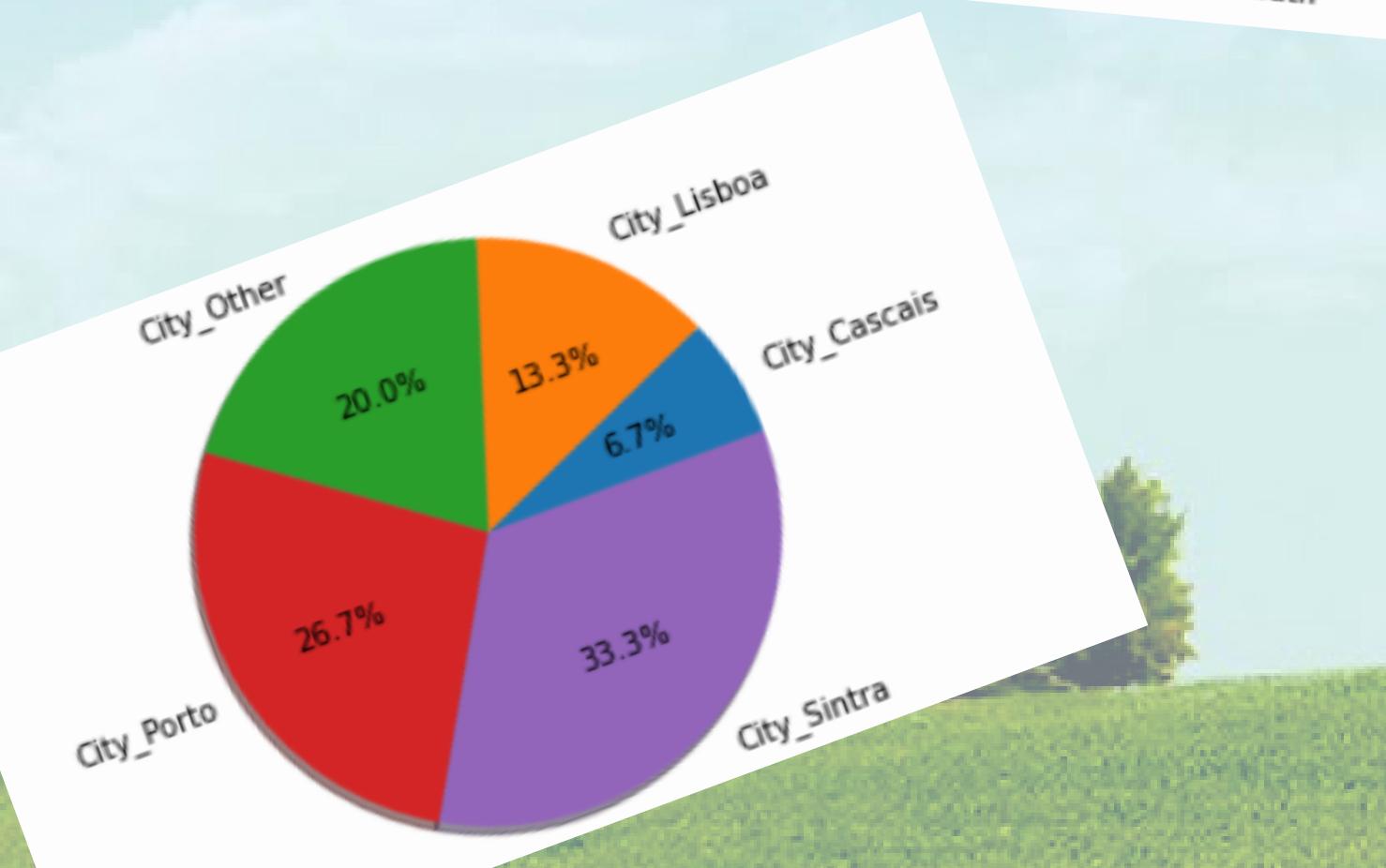
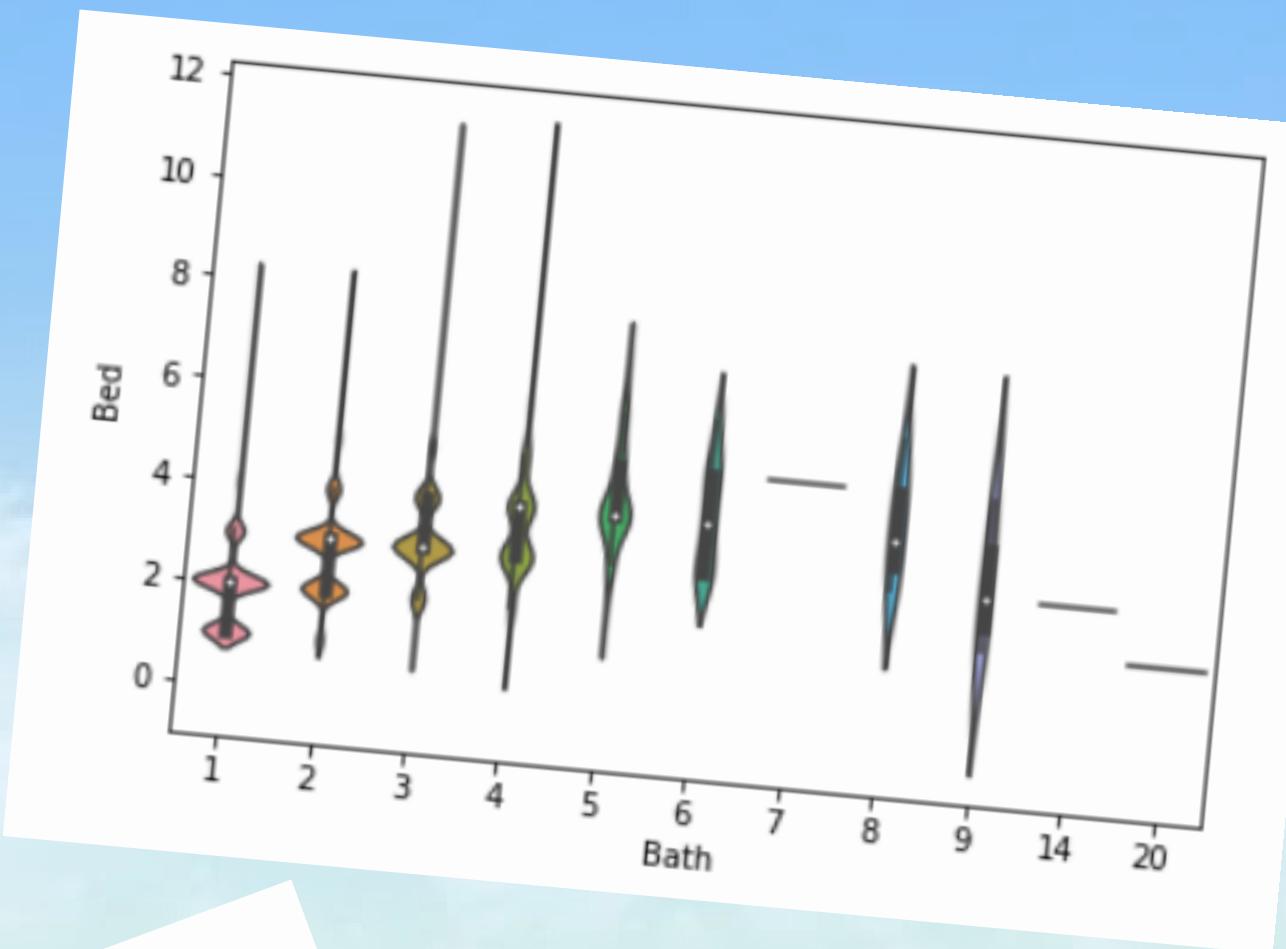
Before



After

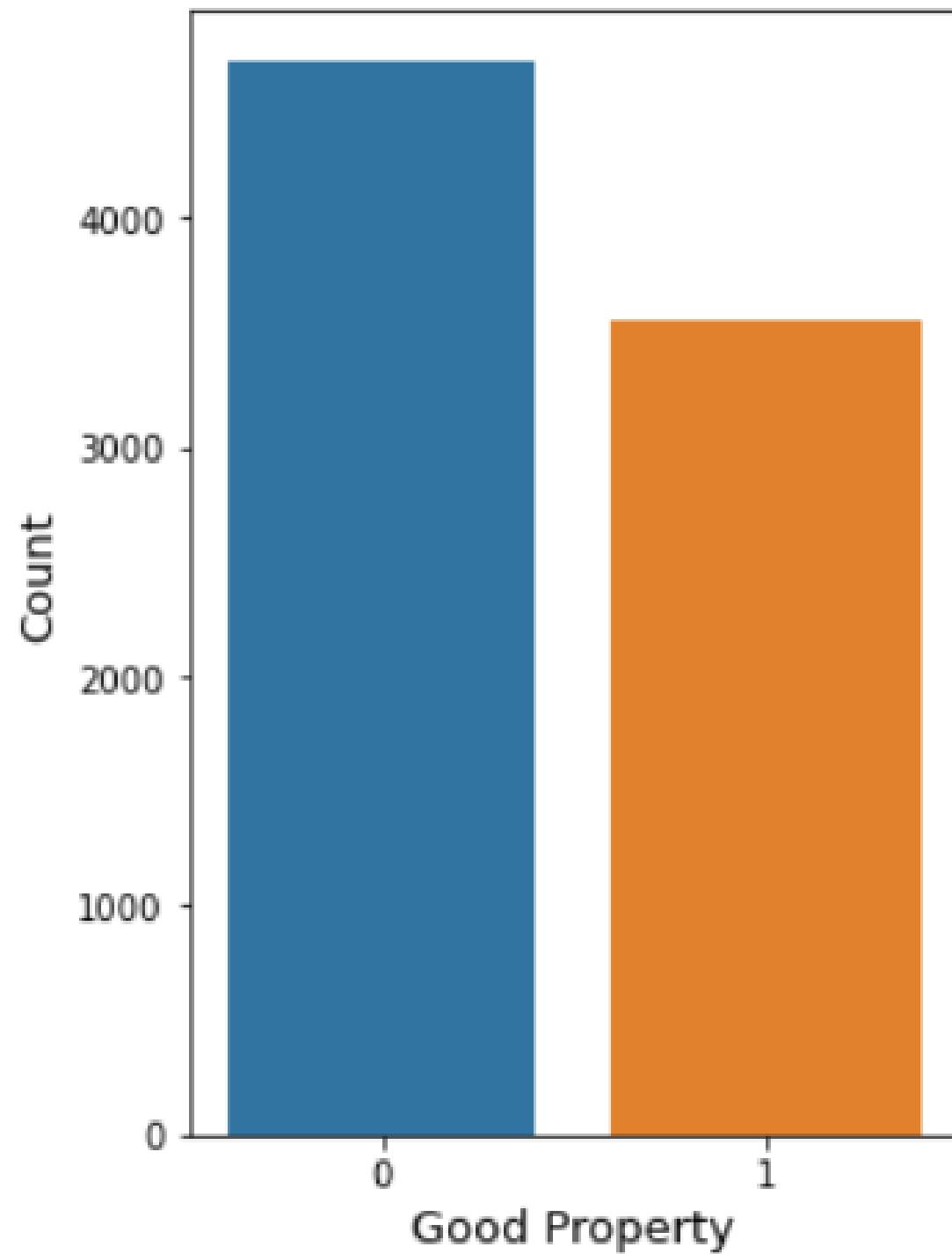


# Some of the visualization we get after cleaning the data



We decide that good property  
is a property that his owner  
doesn't lost more than 400 euro per month

not good: 56.9999999999999%  
good: 43.0%



## Supervised Learning :

Again , we did a machine learning, But now on the buy data frame , and the target column to predict and learnin the machine on is the 'good\_prop' column.

### Classification model

```
In [39]: # feature = X  
# target = Y  
# X created by dropping the 'good_prop' from the original data set  
# The target vector y is then assigned to the good_prop column from the original dataset  
X = ML.drop(columns = ['good_prop'])  
predictors = X.columns  
y = ML['good_prop']
```

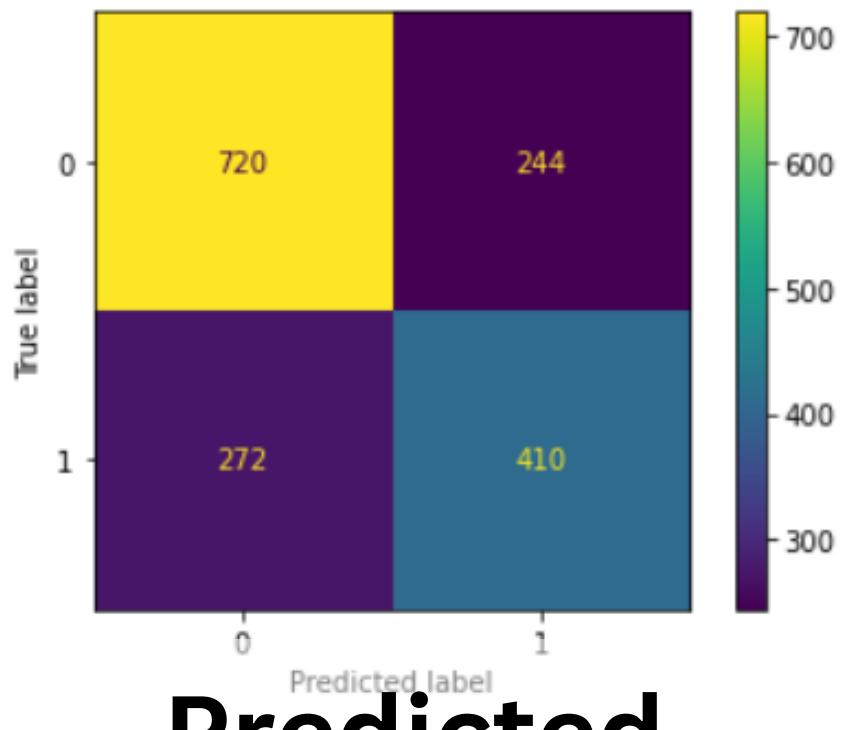
### Logistic Regression

```
In [40]: X_train, X_test, y_train, y_test = train_test_split(x,y, test_size=0.2)  
  
z = StandardScaler()  
z.fit(X_train[['Size']]) # size is the only continuous var  
  
X_train['Size'] = z.transform(X_train[['Size']]) # transform - to subtract the mean and divide by the standard deviation in  
X_test['Size'] = z.transform(X_test[['Size']]) # order to get the standardized values  
  
להחסיר את הממוצע ולהחלק בסטיית התקן על מנת לקבל את הערכים המותקנים #
```



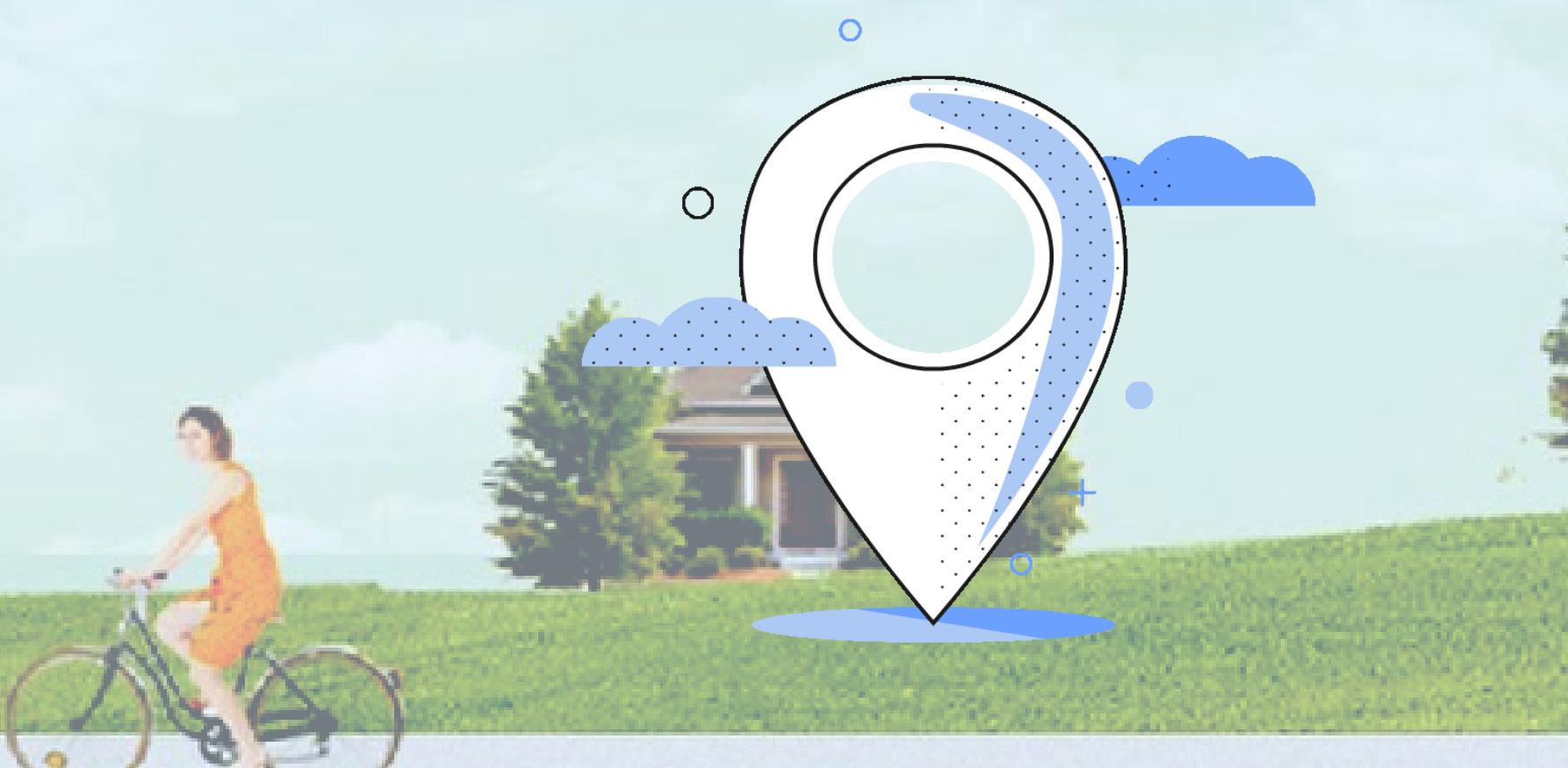
$$(10.1) \text{ Accuracy} = \frac{T_p + T_n}{T_p + T_n + F_p + F_n}$$
$$(10.2) \text{ Precision} = \frac{T_p}{T_p + F_p}$$
$$(10.3) \text{ Recall} = \frac{T_p}{T_p + F_n}$$
$$(10.4) F_1 = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}$$

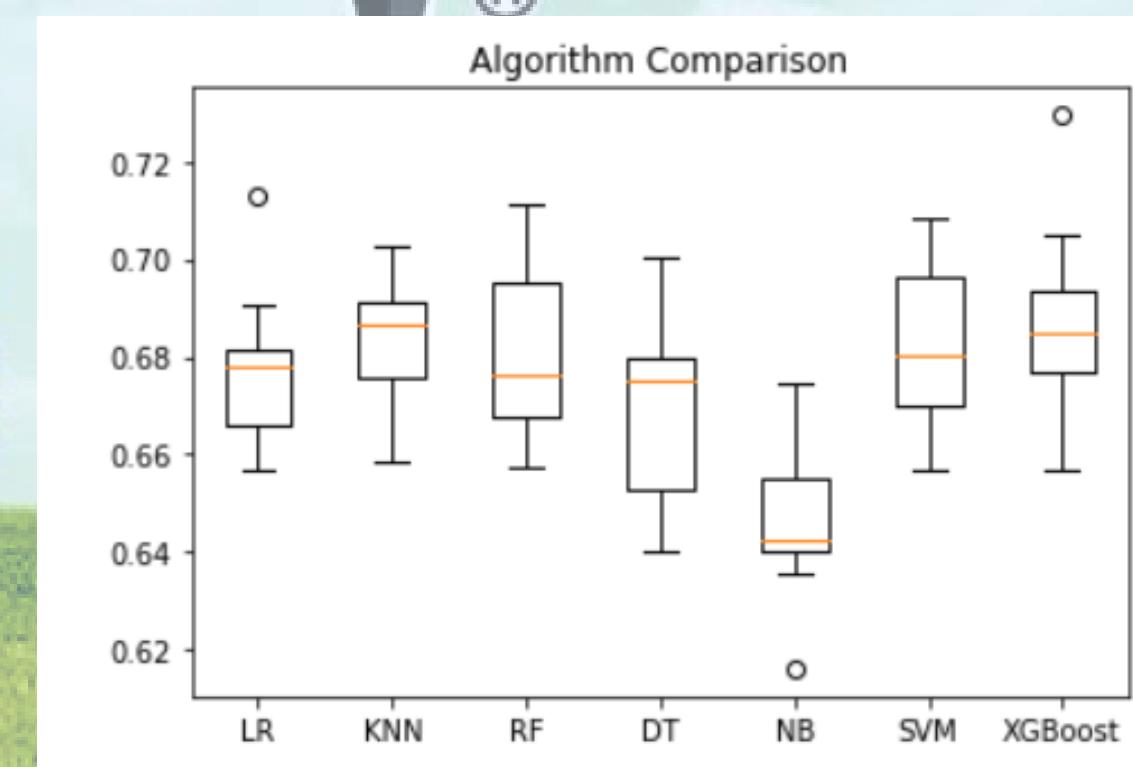
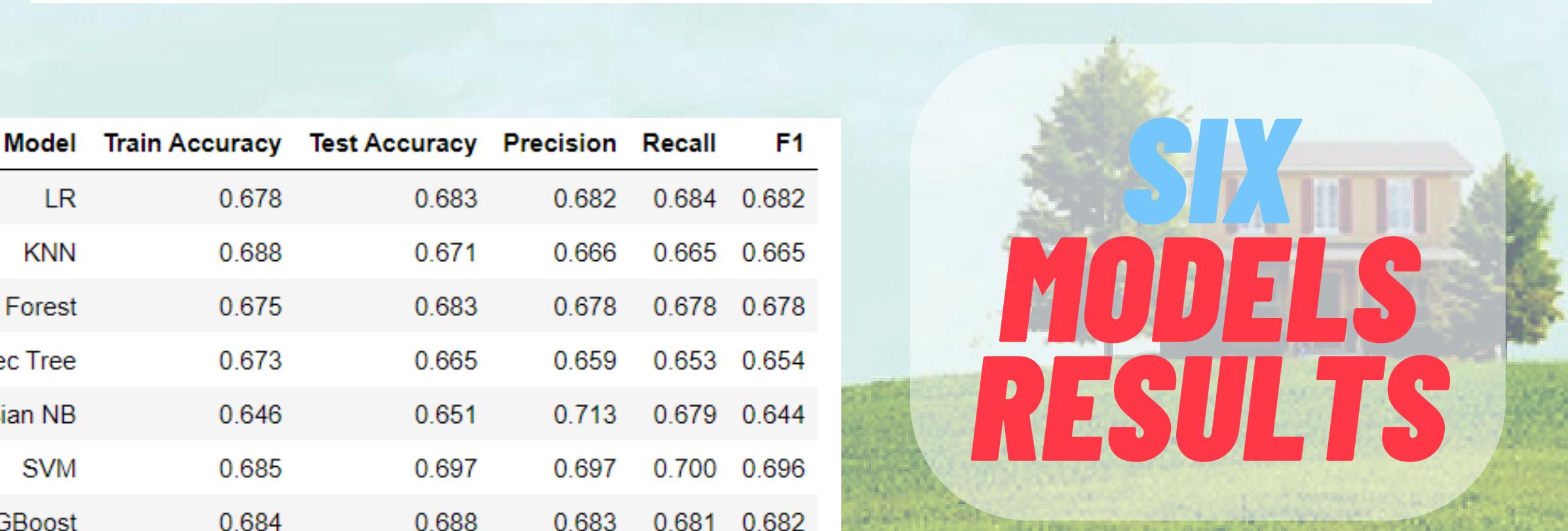
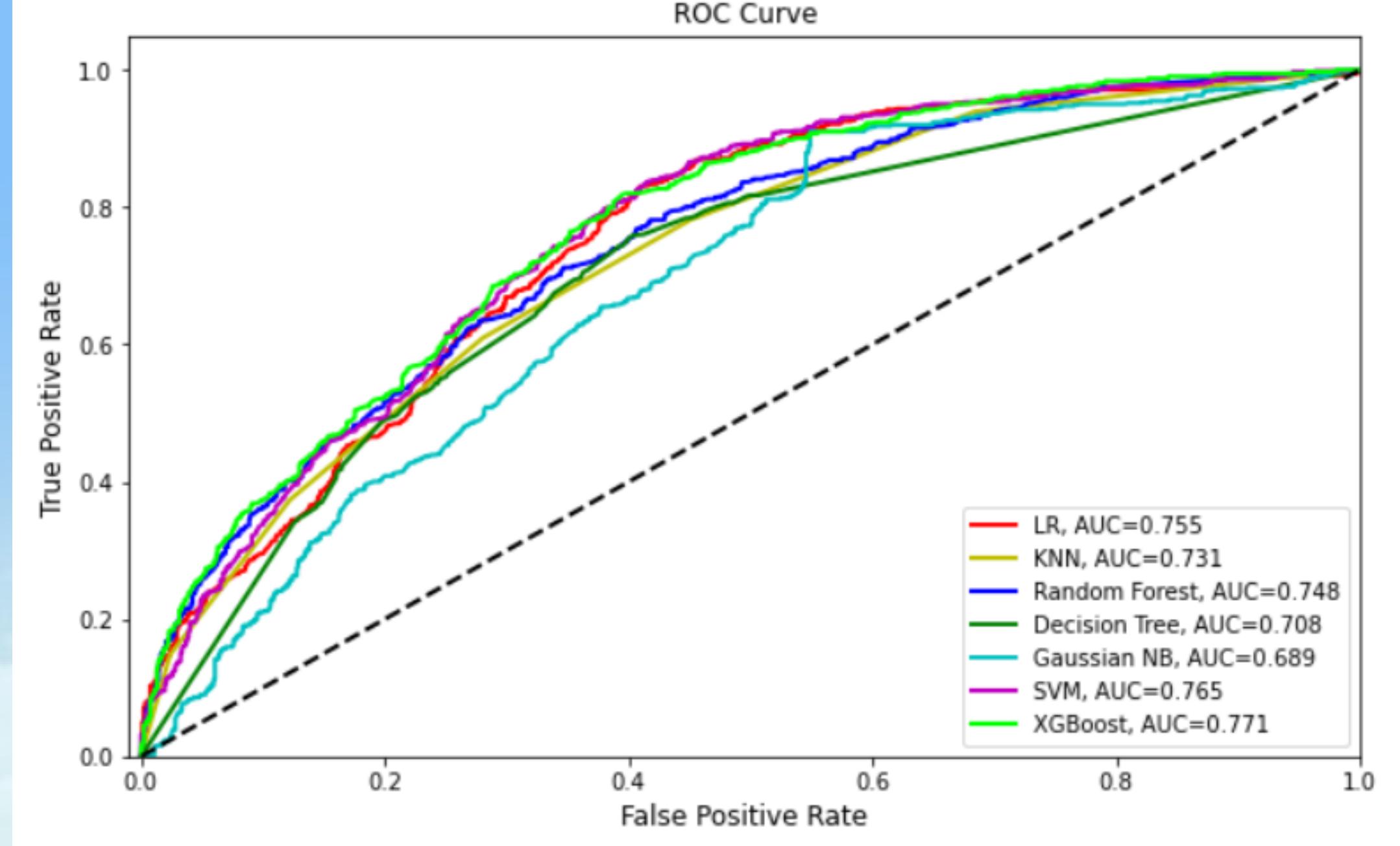




**Predicted  
accuracy**

# MACHINE LEARNING RESULTS



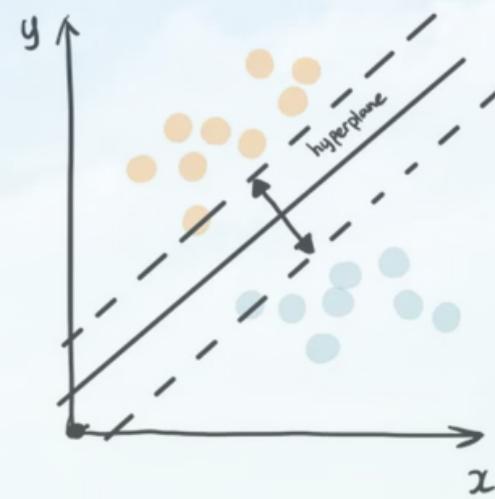


As we can see  
the best score we get is from the  
SVM model.

## FINAL CONCLUSION

5	SVM	0.685	0.697	0.697	0.700	0.696
---	-----	-------	-------	-------	-------	-------

4. Support Vector Machine (SVM)



Our conclusion is that it is indeed possible to identify a good investment property in Portugal.

We have indeed seen that machine learning models achieve quite good results.

It was also said that to improve the accuracy more information is needed which cannot be obtained

Apartment sale and rental site such as the construction price, the quality of the neighborhood, average electricity and water prices for the property

The specific one being tested, and more

