ML Project

Apartment Rent Prediction

Phase 1

Team Id: cs_22

احمد محمد عبد المجيد محمد 2021170045

عمرو علي سيد بغدادي 2021170376

حسن محمد حسن عبدالعظيم 2021170165

زياد عبد الحميد ابو الخير 2021170212

باسيلي اشرف باسيلي 2021170114

حسين اسامه عبدالعظيم 2021170169

Colab project link

Preprocessing and Feature engineering

Check nulls and redundant records for the entire data.

```
redundant records: 0
Null count for column 'id': 0
Null count for column 'category': 0
Null count for column 'title': 0
Null count for column 'body': 0
Null count for column 'amenities': 3185
Null count for column 'bathrooms': 30
Null count for column 'bedrooms': 7
Null count for column 'currency': 0
Null count for column 'fee': 0
Null count for column 'has_photo': 0
Null count for column 'pets_allowed': 3751
Null count for column 'price': 0
Null count for column 'price_display': 0
Null count for column 'price_type': 0
Null count for column 'square_feet': 0
Null count for column 'address': 2971
Null count for column 'cityname': 66
Null count for column 'state': 66
Null count for column 'latitude': 7
Null count for column 'longitude': 7
Null count for column 'source': 0
Null count for column 'time': 0
```

Title, Body: replace each value (string) with its length.

(Assumption: more length means more clarity)

Amenities:

Nulls:

- Filled with the mean value.

Values:

- First iterate over the values and cut the value which is a string to extract amenities. (“a, b, c” => “a” “b” “c”)
- Then save the frequency of each amenity.
- Then Iterate again and for each record we sum (1 / frequency [amenity])

- Finally Replace the record with the summation.

(Assumption: if the frequency of the amenity is small then its more valuable than amenity with much more frequency.)

Bathrooms:

Nulls:

- Since the null values in the column is very small then we drop nulls

Values:

- it's wrong that an apartment has a float number of bathrooms, so we replace each float with its integer value.

Bedrooms:

- While exploring the data, we noticed a records with 0 bedrooms and it turns out that 0 indicates that the unit is a studio, so we created a new column 'studio' its values is 1 if it's a studio (bedrooms == 0) else 0.

From the bedrooms and bathrooms, we generate new column rooms which is the addition of the two columns.

From the rooms and square feet, we generate new column room_ratio which is the division of the square feet on rooms column.

Currency, fee, has_photo, price_type, state, source:

Unique values for each column

```
['housing/rent/apartment' 'housing/rent/short_term' 'housing/rent/home']
['Monthly' 'Weekly' 'Monthly|Weekly']
['Thumbnail' 'Yes' 'No']
['No']
['USD']
['Cats,Dogs' 'NO_PETS' 'Dogs' 'Cats']
['RentDigs.com' 'RentLingo' 'RealRentals' 'ListedBuy' 'Listanza'
 'RENTCafé' 'RENTOCULAR' 'GoSection8' 'tenantcloud' 'Real Estate Agent'
 'rentbits']
['NC' 'WI' 'FL' 'NE' 'CA' 'LA' 'WA' 'OK' 'TX' 'MN' 'VA' 'IN' 'OR' 'OH'
 'NJ' 'PA' 'ND' 'KS' 'IL' 'AK' 'MA' 'AZ' 'SC' 'MD' 'IA' 'CO' 'GA' 'NY'
 'MO' 'TN' 'DC' 'MI' 'NH' 'UT' 'AR' 'CT' 'NV' 'RI' 'AL' 'SD' 'KY' 'VT'
 'NM' 'MS' 'MT' 'ID' 'HI' 'WV' 'DE' 'WY']
```

Since the number of unique values for each column is small, we do One Hot Encoding

Which is create a new column for each unique value.

City name:

- Nulls: Since the null values in the column is very small then we drop nulls
- Values: since each column of this columns have many unique (1500+) so we do Frequency Encoding instead of OHE

State:

- After doing OHE on the state column, we used it again to divide the states into time zones.

Pets allowed:

- Nulls: Filled the Nulls with "NO_PETS" (Assumption: null mean no pets allowed)
  To get this assumption we tried many ways to fill the nulls ex. Replace cats with 1, dogs with 1, cats/dogs with 2, and fill the nulls with the mean or with the value of the most corelated record this gives us our assumption.
- Values: do one hot encoding generates new 4 columns.

Id and Time:

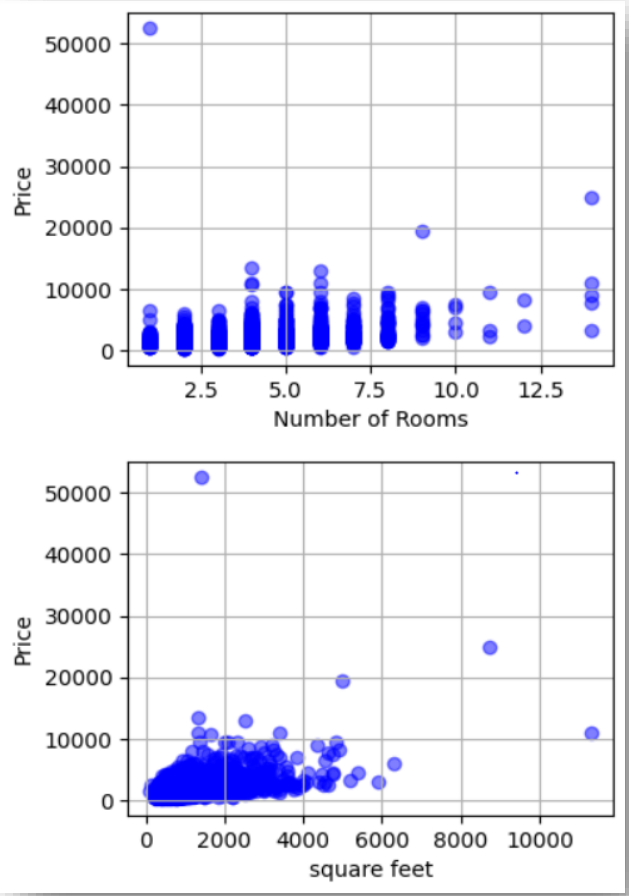
- Use min-max scaling technique to scale the values since it is a very large number and can affect the model efficiency.

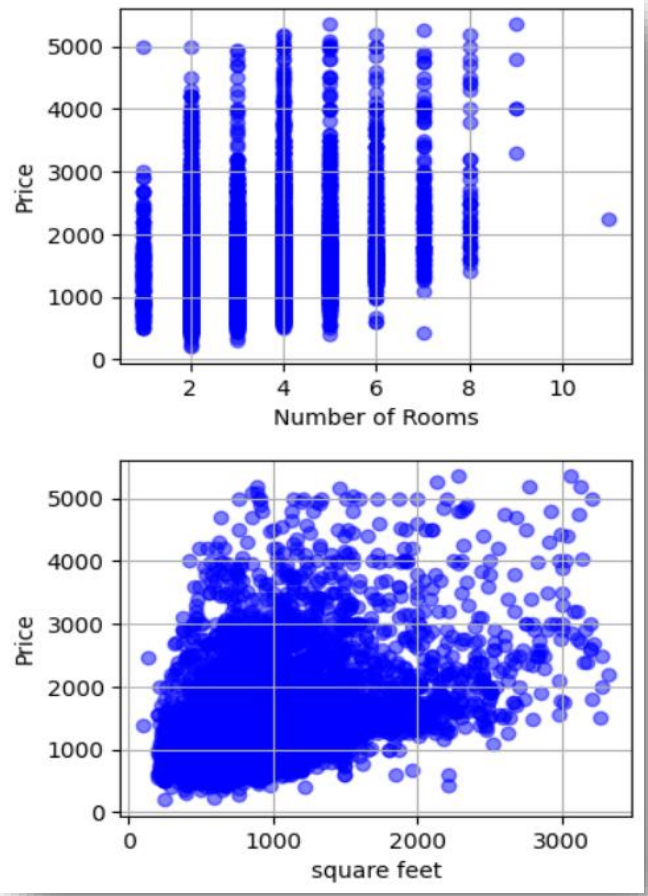
Address:

- **Fill nulls with 0 else 1.**

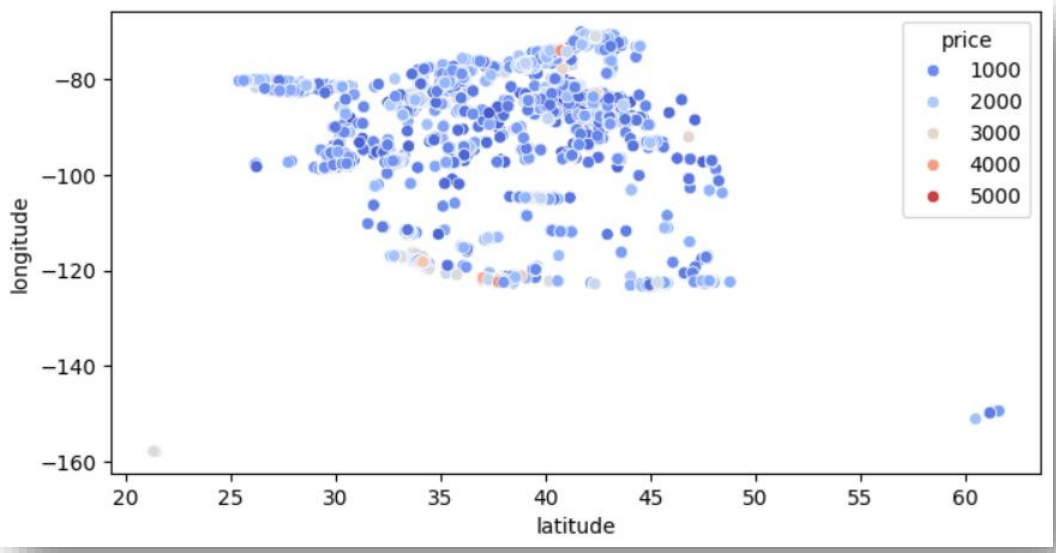
Analysis

Show the relationship between square feet and number of rooms with price before and after removing outliers.





Show the relationship between longitude and latitude with price.

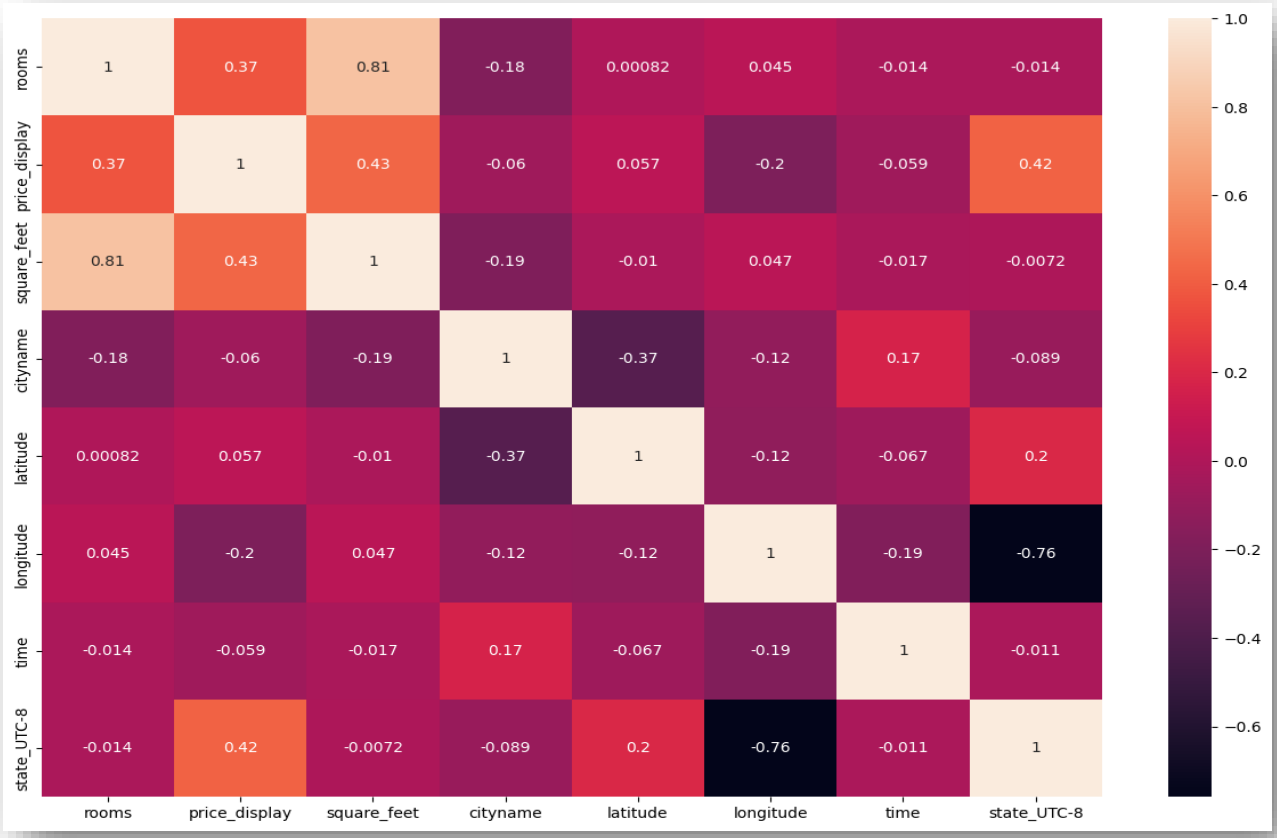


* This plot gives us the idea of dividing the states into time zones which increases the model accuracy.

Heat Map

shows the correlation between columns used in the model (correlation > 0.05) and price.

Polynomial model



Lasso model.

shows the correlation between columns used in the model (correlation > 0.03) and price.

Testing and validation:

Polynomial regression models non-linear relationships by fitting polynomial curves, while Lasso regression, a type of linear regression, performs variable selection and regularization by penalizing the absolute values of coefficients.

We split the data into 0.3 test and 0.7 train, used Cross validation with 10 folds.

Polynomial model

- Evaluation for polynomial regression with degree 3
- Discard id, title doesn't add value.

```
Polynomial Model cross validation mean square error : 189032.8257663265
Polynomial Model cross validation mean score : 0.6215260952915593
Polynomial test data Mean Squared Error: 189861.01044405156
Polynomial test data R2 score 0.6417010797614504
```

Lasso model.

- Evaluation for Lasso regression with alpha 0.01

```
Lasso Mean Squared Error: 312526.24316594785
Lasso Model  R2 score 0.41021163212671696
```

Conclusion

This phase highlighted the importance of pre-processing, as the data needed a lot of pre-processing, that made us gain a lot of hands-on experience in preprocessing different techniques.

THANK YOU