NEW YORK CITY AIRBNB LISTING PRICE PREDICTION

Vladimir Andrianov
Äriinfotehnoloogia, IABM34_Virumaa
Tallinn University of Technology

25.11.2021

NEW YORK CITY AIRBNB LISTING PRICE PREDICTION

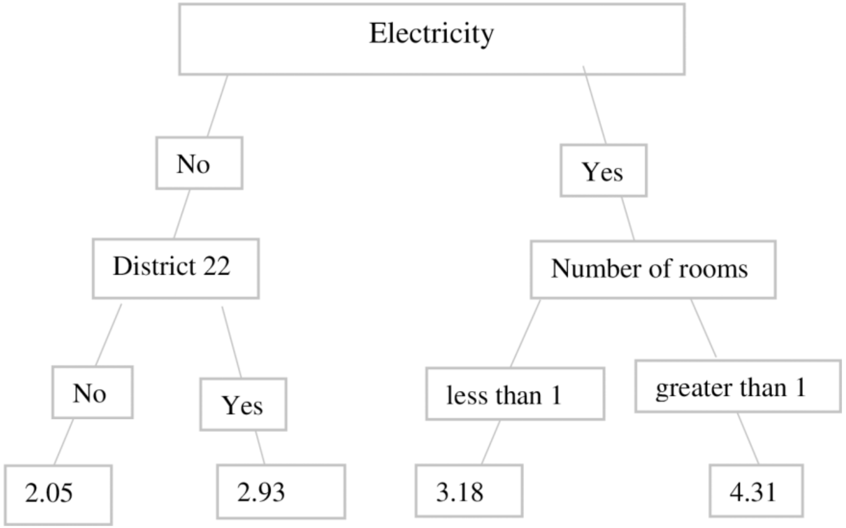
- Introduction
  - Airbnb, Prices
  - Work environment
- Related Works
  - Predicting the rental value of houses
- Dataset
  - Kaggle
- Analysis
  - Charts, data sorting, correlations, maps
- Classification
  - Methods & Results
- Regression
  - Methods & Results

INTRODUCTION

- Airbnb
  - Metrics
  - Rental prices
  - Correlations
  - Price prediction
- Work environment
  - Python
  - Anaconda
  - Jupyter Notebook

RELATED WORKS

- “Predicting the rental value of houses in household surveys in Tanzania, Uganda and Malawi: Evaluations of hedonic pricing and machine learning approaches“
- Rental value prediction



Variable+	Relative information	Variable	Relative information
	2010		2012
Electricity	22.68	Electricity	28.51
External wall	12.20	Number of rooms	13.34
Floor	10.98	District 20	13.09
Number of rooms	10.09	Flush toilet	7.49
Private water tap	7.09	External wall	5.56
District 20	6.67	Floor	5.32
Dwelling	6.12	Dwelling	4.83
District 64	2.86	District 64	3.55
Public tap water	2.60	Unprotected well	3.53
Flush toilet	2.31	Private water tap	3.44
Borehole water	2.14	Borehole water	2.93
District 27	2.07	Public water tap	2.71
District 33	1.79	Shared toilet	2.65
Unprotected well	1.75	Private toilet	2.01
Private toilet	1.66	Protected well	2.00
Protected well	1.66	District 41	1.90
District 32	1.54	District 37	1.74
Shared toilet	1.54	VIP toilet	1.64
District 37	1.52	District 51	1.36
Roof	1.22	District 4	1.04

+ District fixed effect variables are used in the estimation of the models and the specific results for these variables are not reported in the Tables in the interest of space. There are 66 districts, 9 districts, and 32 districts included in the data from Uganda, Tanzania, and Malawi, respectively.

	Uganda		Tanzania		Malawi		Overall performance score
	2010	2012	2014	2016	2014	2016	
OLS+	1.00	1.00	1.00	1.00	1.00	1.00	-
Ridge	0.94	0.96	0.99	0.97	0.95	1.01	83%
LASSO+ +	0.89	0.92	1.00	0.88	0.95	1.01	83%
Tree	0.84	0.83	1.11	1.60	1.11	1.05	33%
Bagging	0.78	0.80	0.98	1.28	0.88	0.91	83%
Forest	0.82	0.88	1.00	0.81	0.84	0.88	83%
Boosting	0.88	0.85	0.96	0.96	0.87	0.91	100%

+OLS = Ordinary Least Squares

+ +LASSO = Least Absolute Shrinkage and Selection Operator.

https://doi.org/10.1371/journal.pone.0244953.t011

DATASET

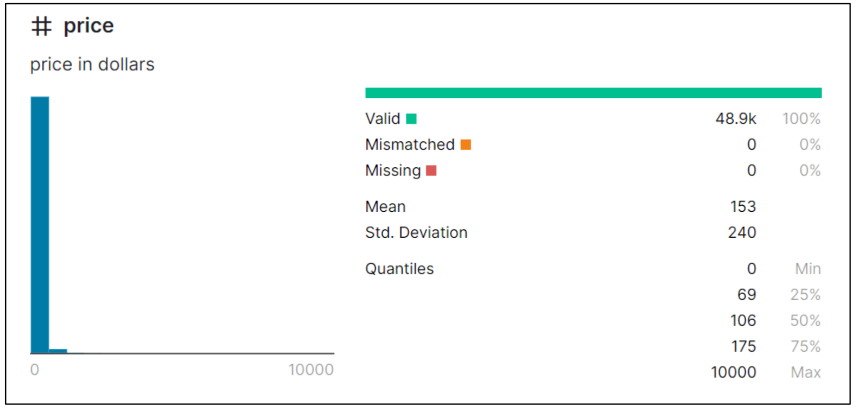
- Kaggle
- Airbnb public data
- 2019 year
- NYC
- Usability
  - 10/10
  - Data dictionary
  - 48895 rows
  - 16 columns

ANALYSIS

- Charts
  - Data distribution
  - Detecting outliers in the data
- Data sorting
  - Dropping outliers, attributes
  - Converting categorical data to factors
- Correlations
  - Correlation matrixes
  - Numbers and factors
- Maps
  - Heat map
  - Price map & longitude and latitude

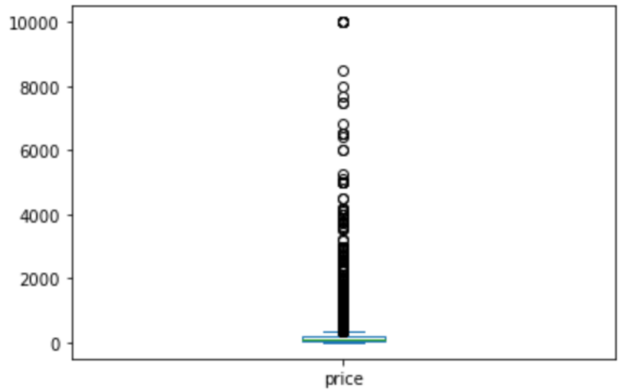
CHARTS

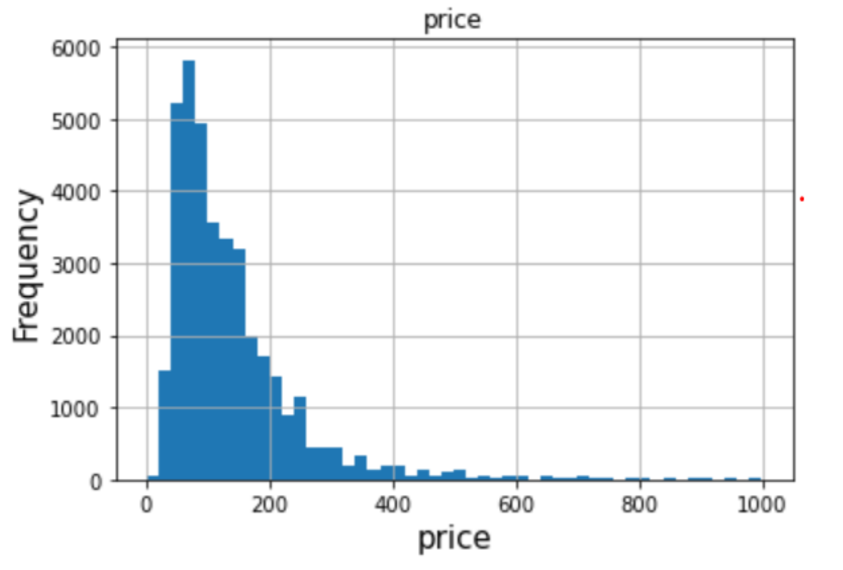
- Charts
  - Data distribution
  - Detecting outliers in the data




```python
df[['price']].plot.box()
outliers = df[df["price"]>1000]
print("number of outliers", outliers.shape[0])
```

```
number of outliers 239
```



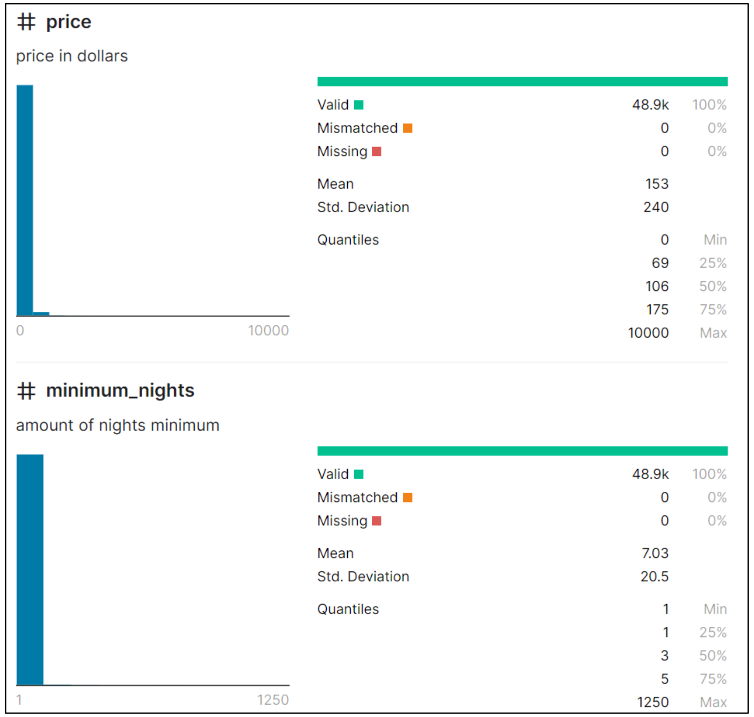


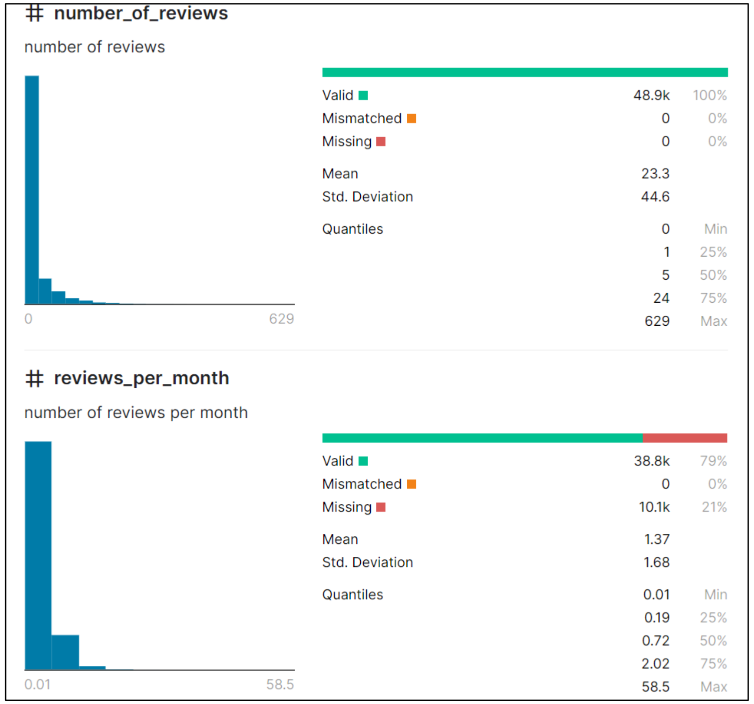
DATA SORTING

- Data sorting
  - Dropping outliers, attributes
  - Converting categorical data to factors

```
df=df[df['price'] < 1000]
df=df[df['minimum_nights']<=365]
df=df[df['number_of_reviews']<=200]
df=df[df['reviews_per_month']<=10]  # Drops missing too.
df.drop(['id','name','host_name','last_review', 'host_id'],
axis=1, inplace=True)
print(df.shape)
```

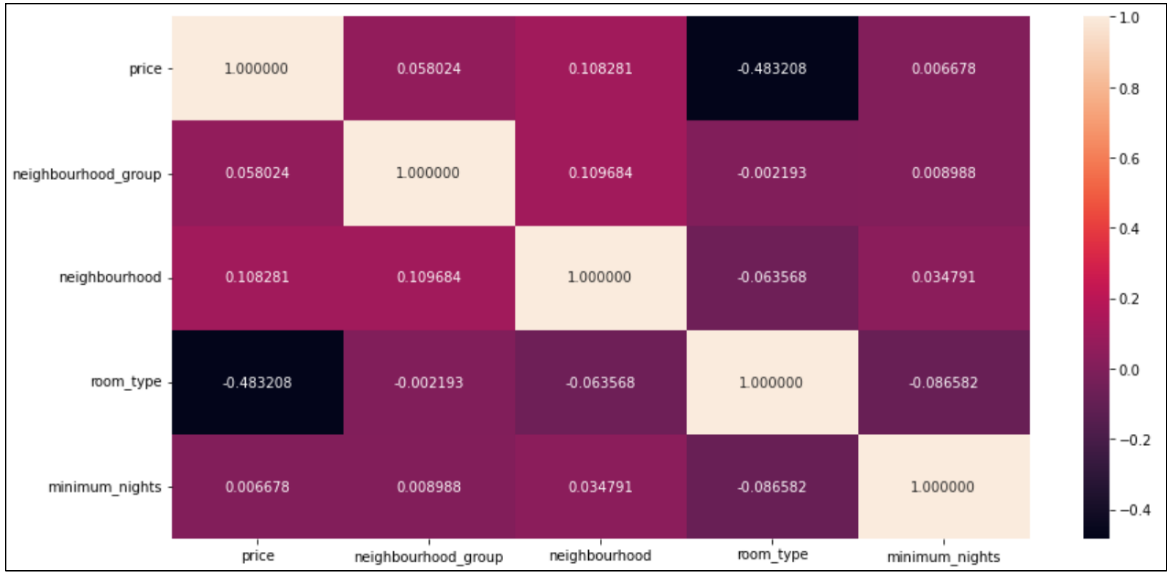
```
(38015, 11)
```

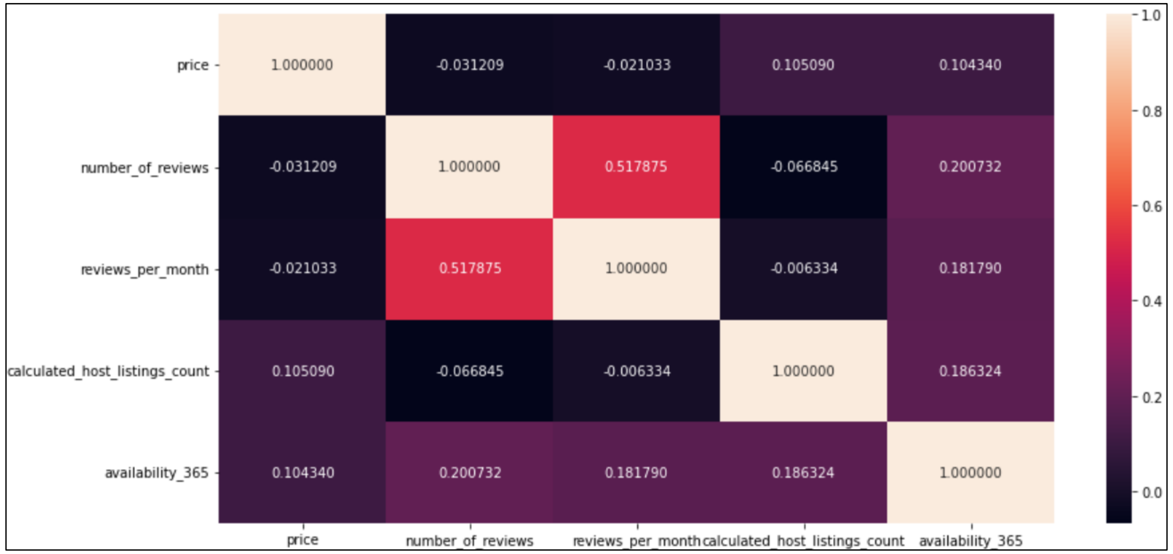




CORRELATIONS

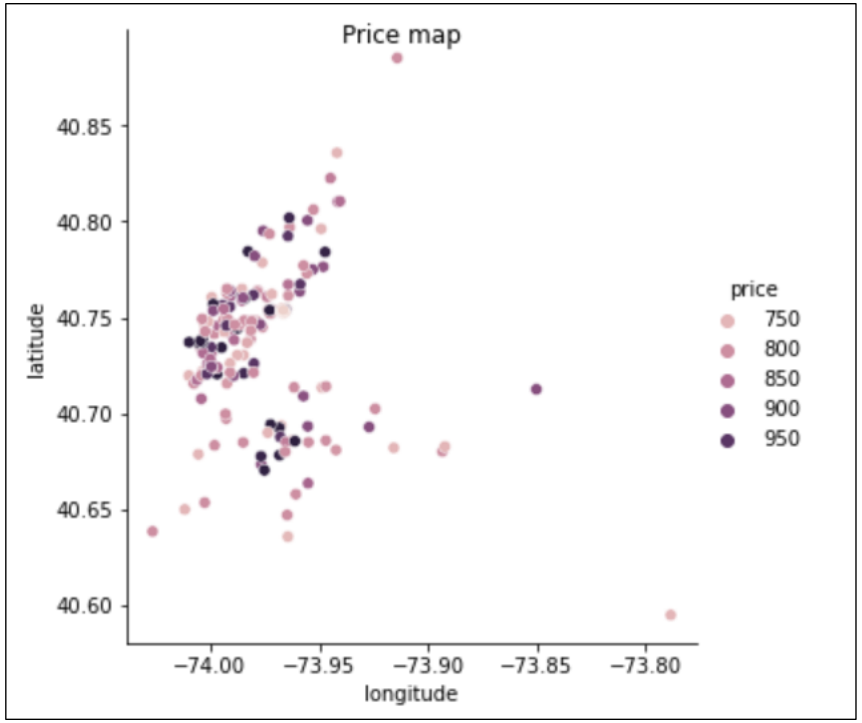
- Correlations
  - Correlation matrixes
  - Numbers and factors

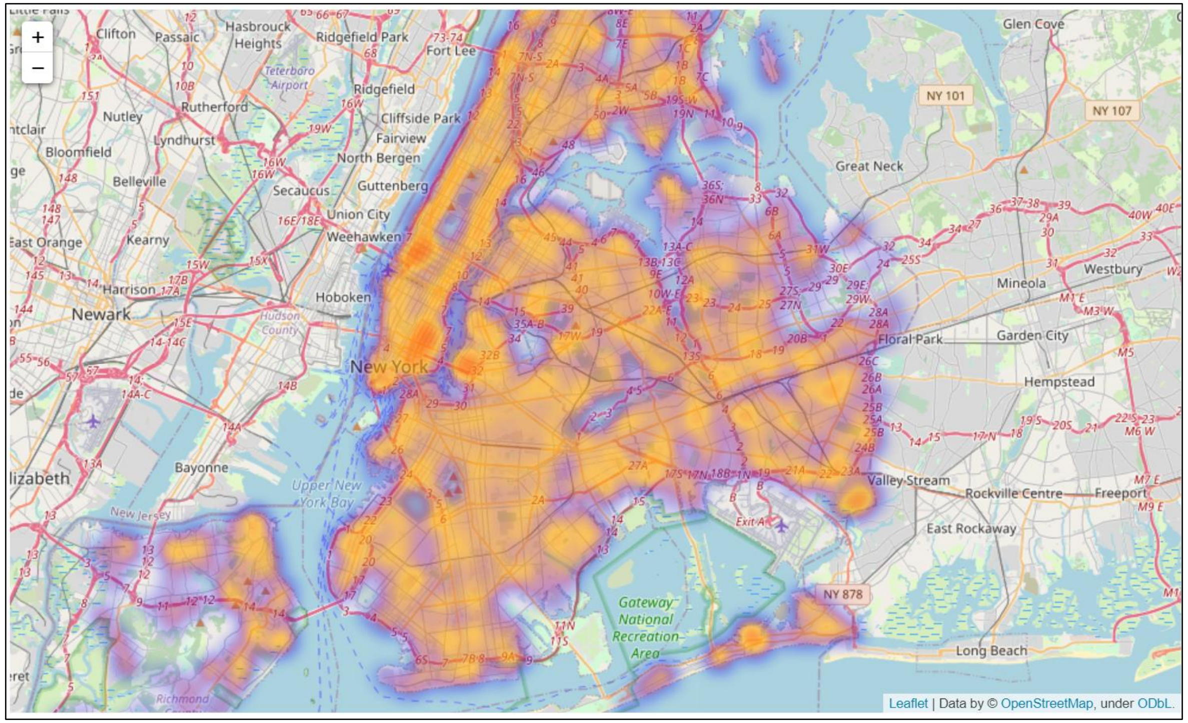




MAPS

- Heat map
- Price map & longitude and latitude (0.2:blue, 0.4: purple, 0.6: orange, 1.0: red)





CLASSIFICATION

- Theory "bigger apartment is and the closer it to the center, the more expensive it is, and cheapest rooms are shared"
- Room type classification based on "neighbourhood_group" and "price"
- Methods
  - Naive Bayes
  - Support Vector Machine (SVM)
  - Decision Tree

NAIVE BAYES, SUPPORT VECTOR MACHINE (SVM)

- Naive bayes is faster than SVM, SVM “sigmoid” ran out of time (>2 hours)
- Well-picked SVM is more accurate

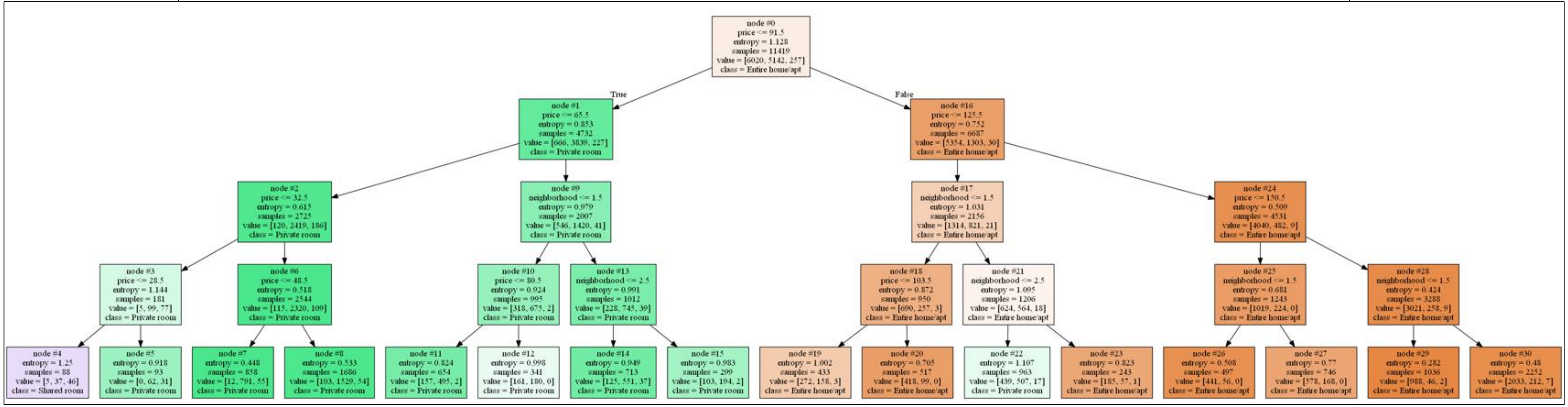
```
Accuracy GaussianNB bayes 0.7231849720264355
Time for 10 fold CV on bayes is: 0.05402469635009765 6
Accuracy MultinomialNB bayes 0.7205197927758903
Time for 10 fold CV on bayes is: 0.0419957637786865 2
```

```
Accuracy sigmoid SVM 0.46
Time for 5 fold CV on sigmoid SVM is: 85.94s
Accuracy rbf SVM 0.8
Time for 5 fold CV on rbf SVM is: 74.77s
Accuracy linear SVM 0.8
Time for 5 fold CV on linear SVM is: 1312s
Accuracy sigmoid SVM no data
Time for 5 fold CV on sigmoid SVM is: no data
```

DECISION TREE

- Min. requirements: 31 nodes ($2n-1=31$, $n=5$)
  - Depth = 4
  - Min_samples_leaf = 2

```
Accuracy decision trees 0.8083388548022695
Time for 5 fold CV on decision trees is: 0.07102847099304199
```



RESULTS

- Decision tree
  - Most accurate
  - Mid-performance
- SVM
  - Accurate as decision tree
  - Slowest
- Naive Bayes
  - Inaccurate but fastest

value	decision trees	Naive Bayes	SVM
acc	0.808450	0.710125	0.803872
time	0.072001	0.040026	160.056998

REGRESSION

- Regression
  - Linear regression
    - Reducing range
  - Ridge regression
  - Lasso regression
  - Elastic-net regression
  - Polynomial regression

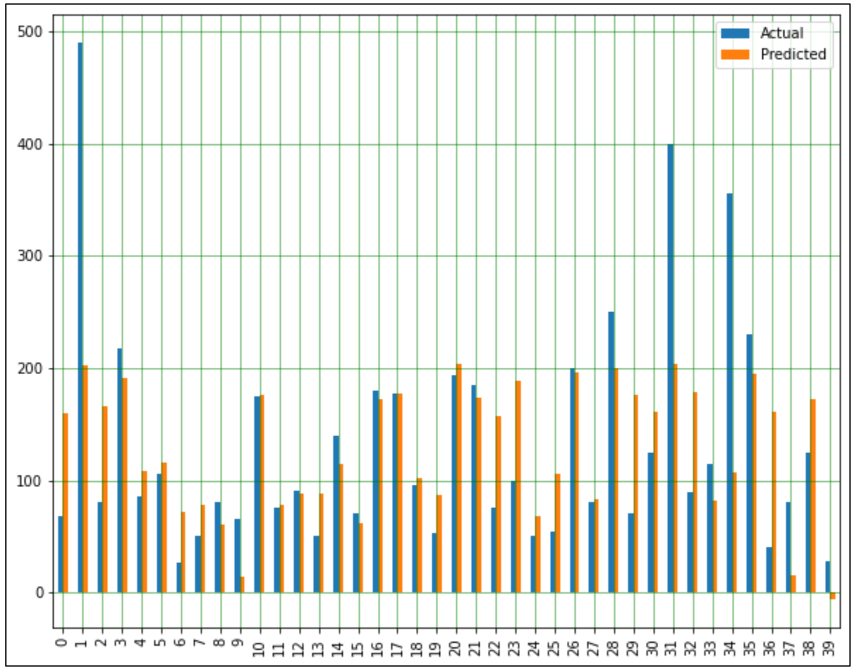
	value	Linear regr	Ridge	Lasso	Elastic_net	Polynomial
0	R^2	0.402685	0.402613	0.352653	0.400905	0.476309
1	time	0.003001	0.001999	0.003000	0.003000	0.030000

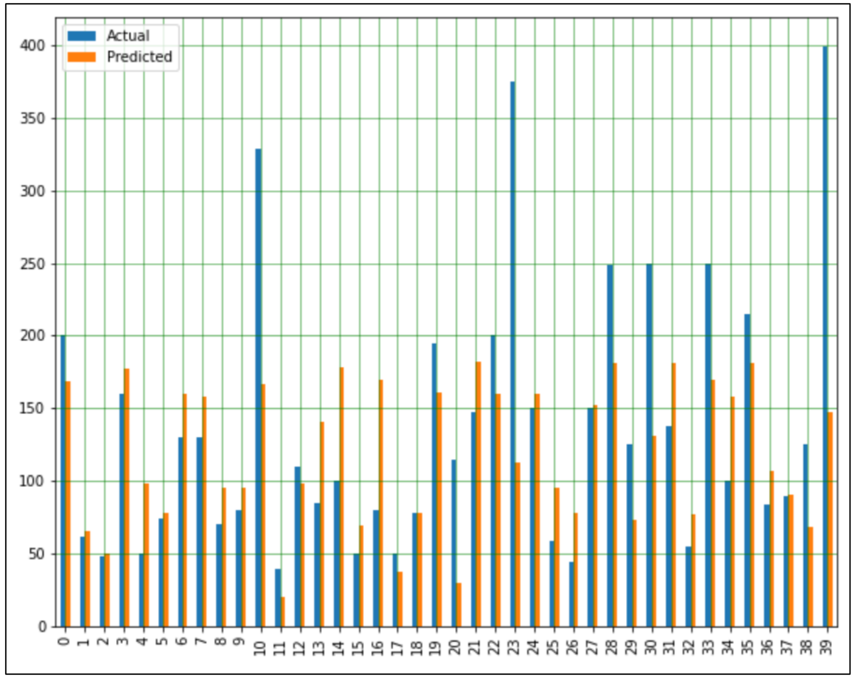
LINEAR REGRESSION

- Full range (mean 153, >1000 dropped)
- Drop outliers "<400"

```
Mean Absolute Error: 54.05353334605293
Mean Squared Error: 8228.12922431871
Root Mean Squared Error: 90.7090360676306
relative error 0.3987589140787031
R^2 0.2794420828008861
time:  0.0020003318786621094
```

```
Mean Absolute Error: 41.0533891337769
Mean Squared Error: 3232.0780056666217
Root Mean Squared Error: 56.85136766751194
relative error 0.302855407535180l4
R^2 0.4026853234464336
time:  0.0030007362365722656
```





RIDGE REGRESSION

- Helps in case of multicollinearity
  - Not the case with current data
  - Similar to linear regression results

```
Mean Absolute Error: 41.05208913242377
Mean Squared Error: 3232.4714553482204
Root Mean Squared Error: 56.854827898325574
relative error 0.30285540753518014
R^2 0.4026126106979544
time:  0.0030007362365722656
```

LASSO REGRESSION

- Less accurate than linear and ridge regressions

```
Mean Absolute Error: 43.5168772546846
Mean Squared Error: 3502.803319775741
Root Mean Squared Error: 59.18448546516004 4
relative error 0.3028554075351801 4
R^2 0.3526530522095674 4
time:  0.0030007362365722656
```

ELASTIC-NET REGRESSION

- Alpha:
  - 1
  - Changed from 1 to 0.001

```
Mean Absolute Error: 49.87572487356842
Mean Squared Error: 4267.152975811097
Root Mean Squared Error: 65.32344889709282
relative error 0.30285540753518014
R^2 0.2113949307256488б
time:  0.0030007362365722656
```

```
Mean Absolute Error: 41.115367994940414
Mean Squared Error: 3241.711054804337
Root Mean Squared Error: 56.93602598359265
relative error 0.30285540753518014
R^2 0.40090505650187525
time:  0.0030007362365722656
```

POLYNOMIAL REGRESSION

- Helps in case of non-linear relationship
  - Helps with current data
  - Best result (default 1 degree and 5)

```
Mean Absolute Error: 41.05338913376738
Mean Squared Error: 3232.0780056666636
Root Mean Squared Error: 56.85136766751231
relative error 0.30285540753518014
R^2 0.4026853234446356
time:  0.0030007362365722656
```

```
Mean Absolute Error: 37.249304025055096
Mean Squared Error: 2833.7018813676873
Root Mean Squared Error: 53.232526535640666
relative error 0.30285540753518014
R^2 0.47630851738235147
time:  0.0030007362365722656
```

RESULTS

- Linear: average accuracy, average speed
- Ridge: average accuracy, **best speed**
- Lasso: worst accuracy, average speed
- Elastic-net: average accuracy, average speed
- Polynomial: **best accuracy**, average speed

	value	Linear regr	Ridge	Lasso	Elastic_net	Polynomial
0	R^2	0.402685	0.402613	0.352653	0.400905	0.476309
1	time	0.003001	0.001999	0.003000	0.003000	0.030000

THANK YOU !