University of New Haven
Tagliatela college of Engineering
Data Science, Computer Engineering and Computer Science

HOUSE RENTAL PRICE PREDICTION

MACHINE LEARNING

&

DATA ANALYTICS - I

CSCI – 6671

Date: 12/7/2020

By,

Namitha Nagaraju

Supervisor,

Travis Millburn

1. Introduction and motivation

Machine learning is an application of artificial intelligence (AI) that provides computers the ability to learn and improve from experience without being explicitly programmed. There are many types of machine learning algorithms:

- **Supervised machine learning algorithms:** Uses past examples to predict future. This can be of two types- Classifiers and Regressors.
  Classifiers predict data and categorize them. Ex: K-Nearest-Neighbours Classifier, Support Vector Classifier.
  Regressors predict a continuous numeric value. Ex: Linear regressor, Decision Tree Regressor etc.,
- **Unsupervised machine learning algorithms:** Used to explore the data and can draw inferences from datasets. This can provide new information from raw data. Ex: Principle Component Analysis, Clustering algorithms etc.,
- **Semi-supervised machine learning algorithms:** For both labelled and unlabelled data, we can use this type of Machine Learning algorithm that falls somewhere in between supervised and unsupervised learning. Ex: Document Classifier
- **Reinforcement machine learning algorithms:** This is a type of machine learning technique that enables an agent to learn in an interactive environment by trial and error using feedback from its own actions and experiences.

International students who come to the USA to pursue their education, find it hard to estimate their cost of living during their studies. The rental amount is one of the biggest concerns for students. Specifications such as number of beds and baths needed, parking and laundry options, the size of the property, the type of property and the pet space requirements vary from one student to another. This project aims at providing an approximated house rental price given the above features to a machine learning model.

Many supervised machine learning algorithms are applied to a USA House rental price dataset. This dataset was obtained from https://www.kaggle.com/rkb0023/houserentpredictiondataset. A subset of the dataset has been taken to analyse the rental prices in Connecticut to particularly help the University of New Haven international/ national students.

2. About the USA Housing Dataset

This is a public data dataset was originally collected by Austin Reese on 2020-01-07 from Craiglist.org. The dataset comprises of 265190 house records with 22 columns/features.

id	type	dogs_allowed	laundry_options
url	sqfeet	smoking_allowed	parking_options
region, state	beds	wheelchair_access	image_url
region_url	baths	electric_vehicle_charge	description
price	cats_allowed	comes_furnished	latitude, longitude

3. Pre-processing

3.1Feature selection

From the above list f features, only the following most needed features for students have been selected. They are

Independent variables:

- Type
- Square feet
- beds
- Baths
- Cats_allowed
- Dogs_allowed
- Wheelchair_access
- Comes_furnished
- Electric_vehicle_ charge
- Smoking allowed
- Laundry-options
- Parking-options

Dependent variable:

- Price

3.2Removing outliers

Outliers were observed in two features: Square feet and price. These were removed. Only prices up to $2500 was selected assuming that a student would not be willing to pay more than that by general observation. Prices below $500 seems unlikely. Similarly for the square feet, the data points with size between 500-2000 square feet was selected. The beds and baths did not have any large numbers and hence data was not filtered based on this. The below are the graphs obtained while removing outliers.

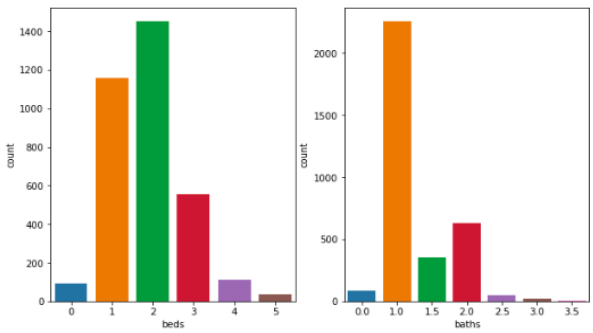

Figure 1: No outliers for beds and baths

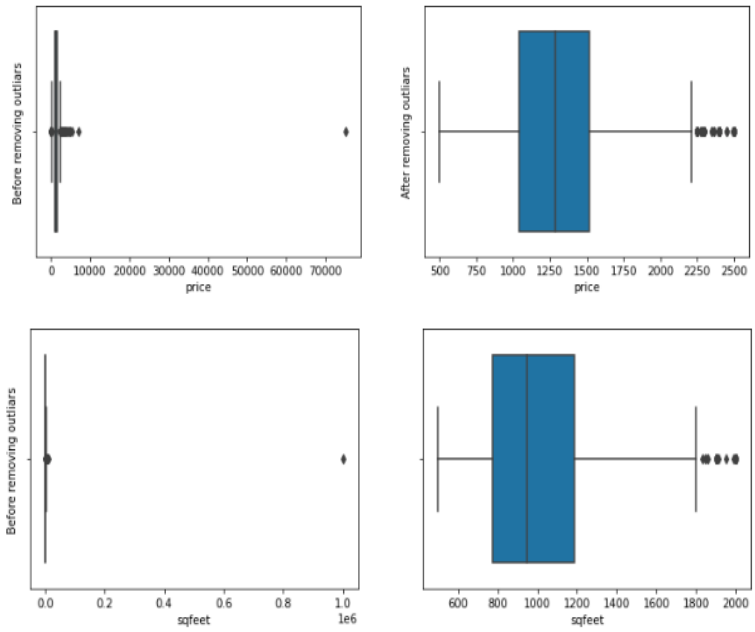

Figure 2: Removing outliers for price and square feet

3.3Managing missing values

The features laundry options and parking options had many missing values. These are categorical variables. The parking options can hold the values- attached garage, carport, detached garage, no parking, off-street parking, street parking or valet parking. The laundry options can hold the values- in building, on site, no on site, with hookups, with in unit.

We can manage the missing values of a categorical variable by one of the following options.

- Ignore the rows with missing data
- Replace by the mode
- Treat it as another category
- Predict using another Classifier

This project has used the last method by predicting the laundry and parking options values by using a basic **K-Nearest-Neighbours** algorithm.

4. Exploratory Data Analysis

To understand what supervised machine learning algorithm must be applied to our data, the data was analysed. The relationship between the different variables can be visualised by plotting a pair-wise plot or heat map.

It can be observed from Figure 3 that, none of the variables are linearly related to the price. These is a slight linear relationship between price and square feet, but

not a very clear one. Other factors also influence the prediction of the house rental price. Figure 4. Shows the correlation between variables. None of them are highly corelated with the price variable.

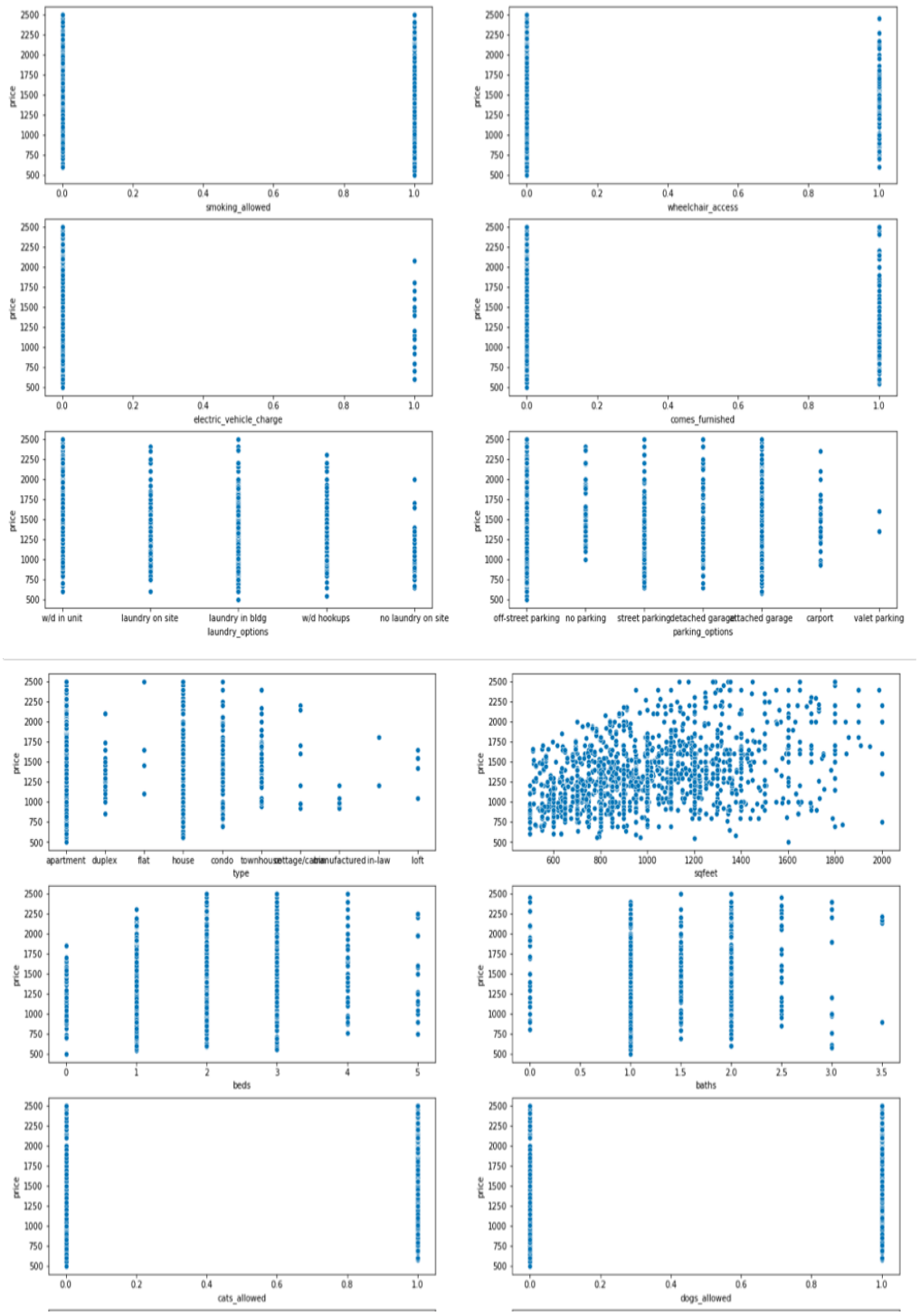

Figure 3: Relationship between the dependent and independent variables

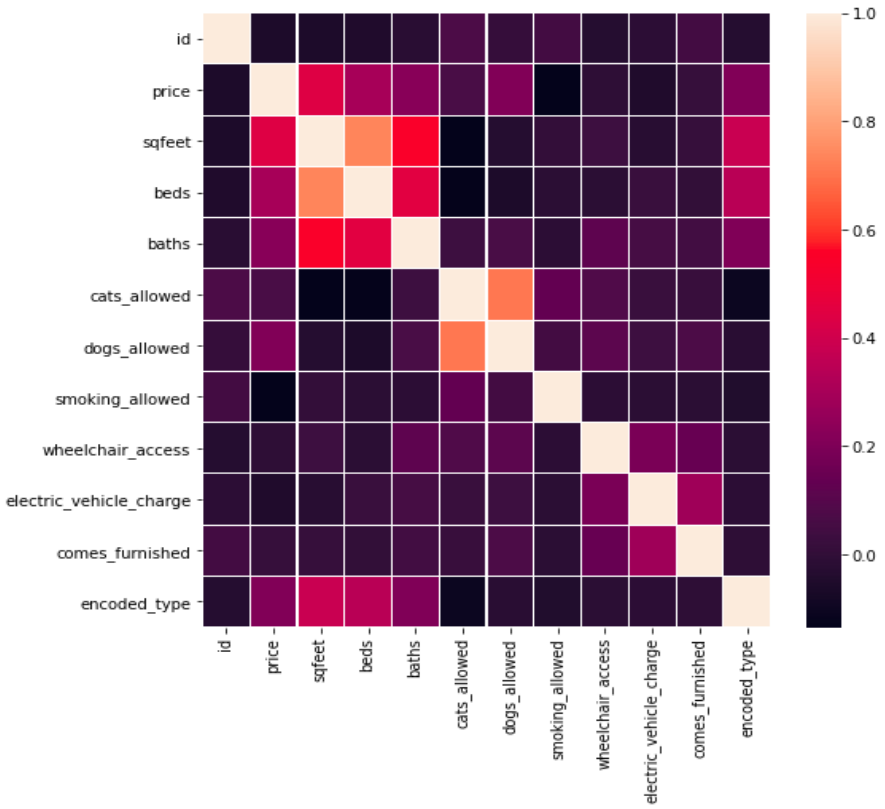

Figure 4: Correlation between the variables

5. Model Selection

To use multiple regression or support vector machines there needs to be a linear relationship between the dependent variable and each of your independent variables or the dependent variable and the independent variables collectively.

It can be seen from the above Exploratory Data Analysis that, there is no evident linear relationship between the independent variables. Hence, Decision Trees or Ensemble methods of Decision Trees are used to predict the house rental price.

Using the Python's sci-kit-learn library, the following regression algorithms were used to predict the rental price.

Label encoding:

DecisionTreeRegressor(), RandomForestRegressor(), AdaBoostRegressor(), BaggingRegressor(), GradientBoostingRegressor() and XGBRegressor()
The figure below shows the training and testing error obtained by using the above models. The matric used to calculate the error is: Mean absolute percentage error

$$M = \frac{1}{n}\sum_{t=1}^{n}\left|\frac{A_t - F_t}{A_t}\right|$$

M = mean absolute percentage error
n = number of times the summation iteration happens
A_t = actual value
F_t = forecast value

Figure 5: Means absolute percentage error

	Algorithm	Train Error	Test Error
0	Decision Tree	1.765674	10.159285
1	Random Forest	4.622644	9.117373
2	Boostrap Aggregation	4.899082	9.546345
3	Ada Boosting	20.791031	20.213649
4	Gradient Boosting	13.153642	13.393528
5	Extreme gradiant boosting	4.559587	9.339084

Figure 6: Variance and bias of different algorithms

Decision trees are highly overfitting with just 1% error with training data and 10% with testing data. Ada boosting and gradient boosting have high training and testing errors. Hence, in the following sections, hyper-parameter tuning is done with RandomForestRegressor(), BaggingRegressor() and XGBRegressor()

6. Hyper-parameter tuning for RandomForestRegressor(), BaggingRegressor() and XGBRegressor()

The below parameters given in Table 1 are tuned using sk-learn's randomSearchCV() .A 5 fold cross-validation was done 100 times. The following results were obtained

N_estimators	183
Min_samples_split	3
Min_samples_leaf	1
Max_features	6
Max_depth	18

Table 1: Hyper parameter tuning for Random Forest

	R^2-TEST	R^2-TRAIN
BEFORE TUNING	**0.725**	**0.939**
AFTER TUNING	**0.743**	**0.926**

Table 2: R-squared scores before and after tuning for Random Forest

It can be observed that, the R-square value (goodness of fit measure of the models) for random forests after parameter tuning has increased for test data and decreased for train data. This implies that it works better for new data and has lesser complexity than before. It has reduced overfitting.

N_estimators	113
Max_samples	1635
Bootstrap_features	False
Max_features	5
Bootstrap	False
Base_estimator	None

Table 3: Hyper parameter tuning for Bagging regressor

	R^2-TEST	R^2-TRAIN
BEFORE TUNING	**0.703**	**0.927**
AFTER TUNING	**0.646**	**0.735**

Table 4: R-squared scores before and after tuning for bagging regressor

In case of Bagging regressor, the model does not perform better than the previous one because the R-squared value has decreased by 6%. However, this has reduced the overfitting nature of the previous model by a 19%.

N_estimators	78
Subsample	1
Min_child_weight	5
Max_depth	81
Gamma	0.5
Colsample_bytree	0.530

Table 5: Hyper parameter tuning for Extreme gradient boosting

	R^2-TEST	R^2-TRAIN
BEFORE TUNING	**0.719**	**0.944**
AFTER TUNING	**0.726**	**0.962**

Table 6: R-squared scores before and after tuning for Extreme gradient boosting

Extreme gradient boost with parameter tuning has increased the R-squared value of the test data but is overfitting 2% more than the model without parameter tuning. By analysing the three algorithms' results after parameter tuning, we can conclude that, the tuned Random Forests performs best. Hence Random Forest Regressor with the hyper parameters mentioned in Table 1. is selected for future rental price prediction.

7. Predicting

```
My inputs: 4  1000  2  1  0  1  0  1  0  0  1  1
PREDICTED RENT AMOUNT IS:  [1047.62737445]
```

The above values were inputted to predict the price of a "Flat" with 1000 square feet, 2 beds, 1 bathroom, Smoking allowed, with wheel chair access, one site laundry and carport type of parking. The model predicted the value- $1047.62

8. Future implications

- The mean absolute percentage error using this model is 4% which is high considering that this application of this machine learning model will be for students who will be residing at Connecticut.
- A larger dataset will be utilized.
- A web-application will be built using features such as longitude and latitudes