Can we predict house prices using known features of each house and a supervised learning approach?

Florence Galliers

18/10/2020

Contents

1 Background:

1.1 Objectives:

- Understand which attributes of houses given in the data set can be used to effectively construct a prediction model for house price (dependent variable).
- Minimize the differences between predicted and actual house prices by using model selection to choose the most accurate model.

2 Data:

This data is from houses sold in 2014 in Washington, USA. It contains the sale price along with attributes of each house. There are 4600 entries and 17 variables in the original data set.

Although the data is 6 years old, it is an interesting data set that contains a large amount of information and number of variables. I was not able to find a similar data set from a more recent year, or from the UK, and so I am going to go ahead with this data set as I feel it will produce some interesting results.

2.1 Data Preparation:

After some exploratory data analysis it became clear that some of the variables in the data set had missing values or were not going to be relevant to this analysis. After removing them we are left with 11 variables.

There was originally a variable called sqft_basement which gave the size of the basement if present, however a lot of houses did not have basement so I felt it would be more useful to turn this variable from numeric into binary (if_basement). I also changed the variable year_renovated into a binary variable (if_renovated), as again, not all the houses had been renovated.

I carried out a check to see if there were any zero values for the price variable as these are not acceptable in a housing price situation, a house cannot cost nothing. These were removed from the data set.

After carrying out a preliminary linear regression model and plotting the diagnostic plots, using Cook's distanced I identified three data entries classed as outliers. These were removed from the data set.

The year_built variable from the original data set has been changed into a house_age variable by subtracting the year built from 2014.

The price variable was changed into thousands of dollars to make it a more manageable variable.

```
## standardGeneric for "t" defined from package "base"
##
## function (x)
## standardGeneric("t")
## <environment: 0x7faba0d60a38>
## Methods may be defined for arguments: x
## Use  showMethods("t")  for currently available ones.
```

The remaining variables and their descriptions are shown in Table 1:

Table 1: Data Dictionary

Variable	Description
price	House sale price in thousands of US dollars
bedrooms	Number of bedrooms
bathrooms	Number of bathrooms
sqft_living	Area of house in square feet
sqft_lot	Area of whole housing lot in square feet
floors	Number of floors in the house
condition	Condition of house, 1 to 5
if_basement	1 = if house has a basement, 0 = no basement
house_age	Year that the house was built subtracted from 2014
if_renovated	1 = if house has been renovated, 0 = if no renovation
city	Location of house to the nearest city in Washington, USA

3 Methods:

As house price is a continuous variable I have taken a supervised learning approach and will be using regression to look at the relationship between house price and features of each house.

Review approaches tried or considered…

Summary of final approach and justification of why this approach was chosen:

```r
set.seed(2)
n = nrow(data2) #number of rows
train_index = sample(1:n, size = round(0.8*n), replace=FALSE)
train = data2[train_index ,] #takes 80% of the data for training set
test = data2[-train_index ,] #remaining 20% for the test set
```

```r
lm_model <- lm(price ~ .-condition-if_renovated,
               data = train)
# make predictions on test set
```

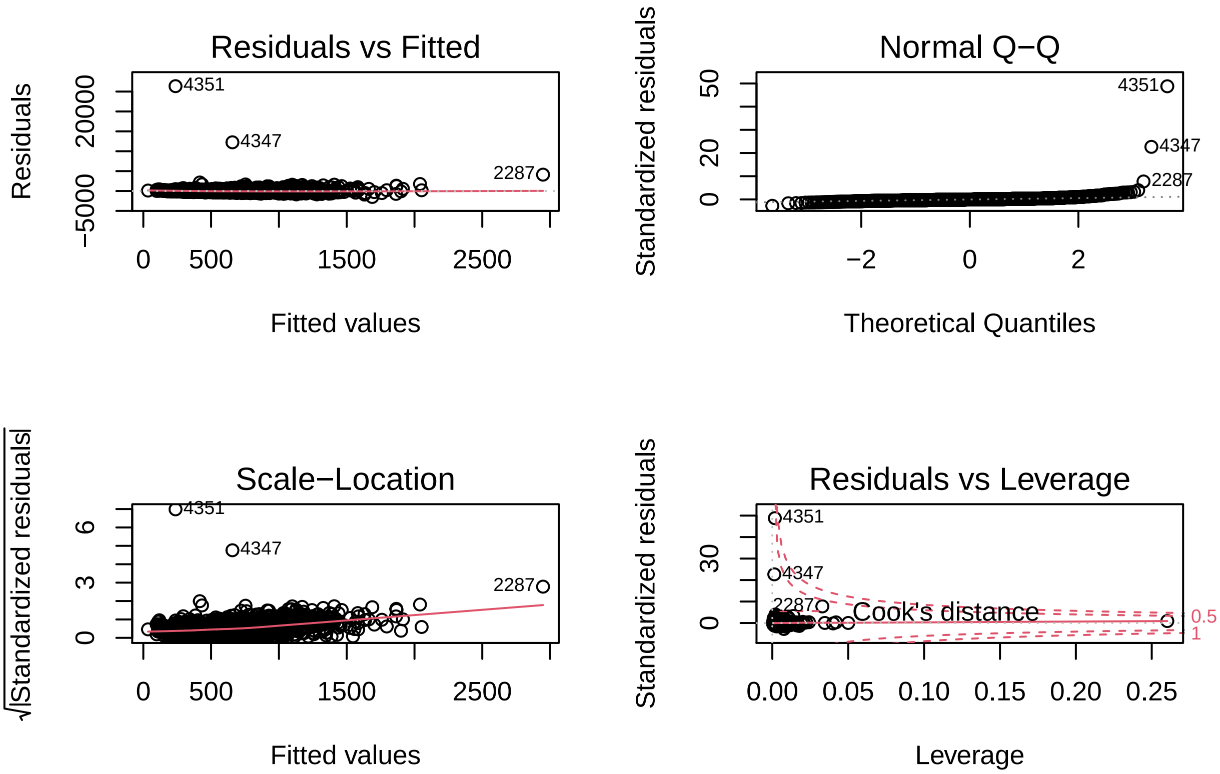
```r
lm_pred <- predict(lm_model, test)
# calculate MSE
mean((test[, "price"] - lm_pred)^2)
```

```
## [1] 68505.72
```

```r
summary(lm_model)
```

```
##
## Call:
## lm(formula = price ~ . - condition - if_renovated, data = train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1493.5  -138.2   -24.9    84.3 26352.5
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)   -1.462e+02  5.134e+01  -2.848  0.00442 **
## bedrooms      -6.792e+01  1.274e+01  -5.329 1.04e-07 ***
## bathrooms      8.220e+01  2.110e+01   3.895 9.99e-05 ***
## sqft_living    2.805e-01  1.609e-02  17.430  < 2e-16 ***
## sqft_lot      -7.401e-04  2.590e-04  -2.858  0.00429 **
## floors         3.055e+01  2.163e+01   1.412  0.15795
## if_basement1 -1.227e+01  2.096e+01  -0.585  0.55837
## house_age      3.058e+00  3.664e-01   8.347  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 540.6 on 3630 degrees of freedom
## Multiple R-squared:  0.1876, Adjusted R-squared:  0.186
## F-statistic: 119.7 on 7 and 3630 DF,  p-value: < 2.2e-16
```

```r
# Diagnostic plots of the linear regression model
par(mfrow=c(2,2))
plot(lm_model)
```



4 Results:

Summary of major results, graphs, diagnostic outputs

Strictly relevant to the objectives

Must include a link to the Github repository containing a fully reproducible and documented analysis Reported in scientific style.

4.1 Conclusions:

~1 paragraph

5 References

5.1 3-5 peer reviewed references

-
-
-
-