CourseProjectPart5

Cody

2024-06-21

Fit full model

You will look at a variety of ways to detect the presence of collinearity in your set of numerical predictors. Include "all" predictors from your data set in a full model

```
Call:
lm(formula = reviews_per_year ~ calculated_host_listings_count +
    price + minimum_nights + availability_365 + number_of_reviews +
    neighbourhood_group + room_type, data = NYC)

Residuals:
    Min      1Q  Median      3Q     Max
-51.899  -8.121  -5.567   3.509  71.666

Coefficients:
                                Estimate Std. Error t value Pr(>|t|)
(Intercept)                    15.975350   3.114249   5.130  3.5e-07 ***
calculated_host_listings_count  0.022891   0.019851   1.153 0.249135
price                           0.001731   0.005881   0.294 0.768508
minimum_nights                 -0.113514   0.029220  -3.885 0.000109 ***
availability_365                0.009305   0.004027   2.310 0.021072 *
number_of_reviews               0.222582   0.009384  23.719  < 2e-16 ***
neighbourhood_groupBrooklyn    -7.199720   2.996507  -2.403 0.016459 *
neighbourhood_groupManhattan   -7.723233   3.039214  -2.541 0.011200 *
neighbourhood_groupQueens      -1.969118   3.229814  -0.610 0.542222
room_typePrivate room          -0.921353   1.155578  -0.797 0.425464
room_typeShared room            1.766700   3.101460   0.570 0.569056
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 15.2 on 981 degrees of freedom
Multiple R-squared:  0.4036,    Adjusted R-squared:  0.3975
F-statistic: 66.38 on 10 and 981 DF,  p-value: < 2.2e-16
```

Assessing Collinearity

Assess this full model for problems with collinearity by calculating the VIF's for each variable and the condition number (remember to scale the variables); you can use the crude rules of thumb mentioned in your book and notes.

```
The condition indicies are:

1 1.147891 1.216153 1.317011 1.342762 1.407738 1.42515 1.468 1.701385 2.195533 1.721031e+14
```

The VIF values are:

```
                                  GVIF Df GVIF^(1/(2*Df))
calculated_host_listings_count 1.069648  1        1.034238
price                          1.508643  1        1.228268
minimum_nights                 1.015404  1        1.007673
availability_365               1.135523  1        1.065609
number_of_reviews              1.051299  1        1.025329
neighbourhood_group            1.138201  3        1.021809
room_type                      1.426709  2        1.092909
```

Collinearity Comments

In your summary, explain the reasoning you use in coming to a decision if there is a serious problem with collinearity in your set of predictors, referencing the condition number and the values of the VIF's.

- The VIF's did not point to any predictors being collinear with any other predictor, as the largest VIF did not get above 5, which is the rule of thumb for being moderately problematic in regards to collinearity. Most of the condition indices were fine too other than the maximum, the condition number is incredibly large. This indicates that there is collinearity with one of the principal components.

Predictor Selection

Use a few different variable selection procedures on your full data set.

Forward selection

```
Call:
lm(formula = reviews_per_year ~ number_of_reviews + neighbourhood_group +
    minimum_nights + availability_365, data = NYC)

Coefficients:
                (Intercept)             number_of_reviews
                   15.54790                       0.22083
 neighbourhood_groupBrooklyn  neighbourhood_groupManhattan
                   -6.99795                      -7.24796
   neighbourhood_groupQueens                minimum_nights
                   -1.88167                      -0.11136
            availability_365
                    0.01052
```

Stepwise selection

```
Call:
lm(formula = reviews_per_year ~ minimum_nights + availability_365 +
    number_of_reviews + neighbourhood_group, data = NYC)

Coefficients:
                (Intercept)                minimum_nights
                   15.54790                      -0.11136
            availability_365             number_of_reviews
                    0.01052                       0.22083
 neighbourhood_groupBrooklyn  neighbourhood_groupManhattan
                   -6.99795                      -7.24796
   neighbourhood_groupQueens
```

```
                    -1.88167
```

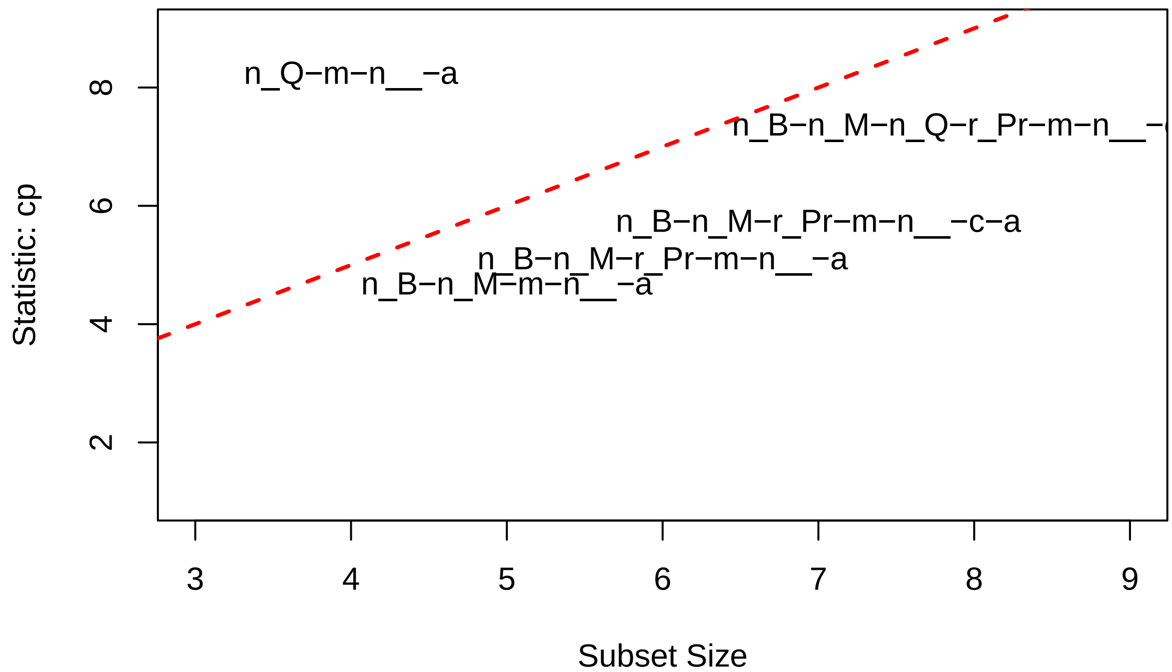
Backward selection

```
Call:
lm(formula = reviews_per_year ~ minimum_nights + availability_365 +
    number_of_reviews + neighbourhood_group, data = NYC)

Coefficients:
                 (Intercept)                minimum_nights
                    15.54790                      -0.11136
            availability_365             number_of_reviews
                     0.01052                       0.22083
 neighbourhood_groupBrooklyn  neighbourhood_groupManhattan
                    -6.99795                      -7.24796
   neighbourhood_groupQueens
                    -1.88167
```

- All selection procedures landed on including availability_365, minimum_nights, Number_of_reviews, and neighbourhood_group.

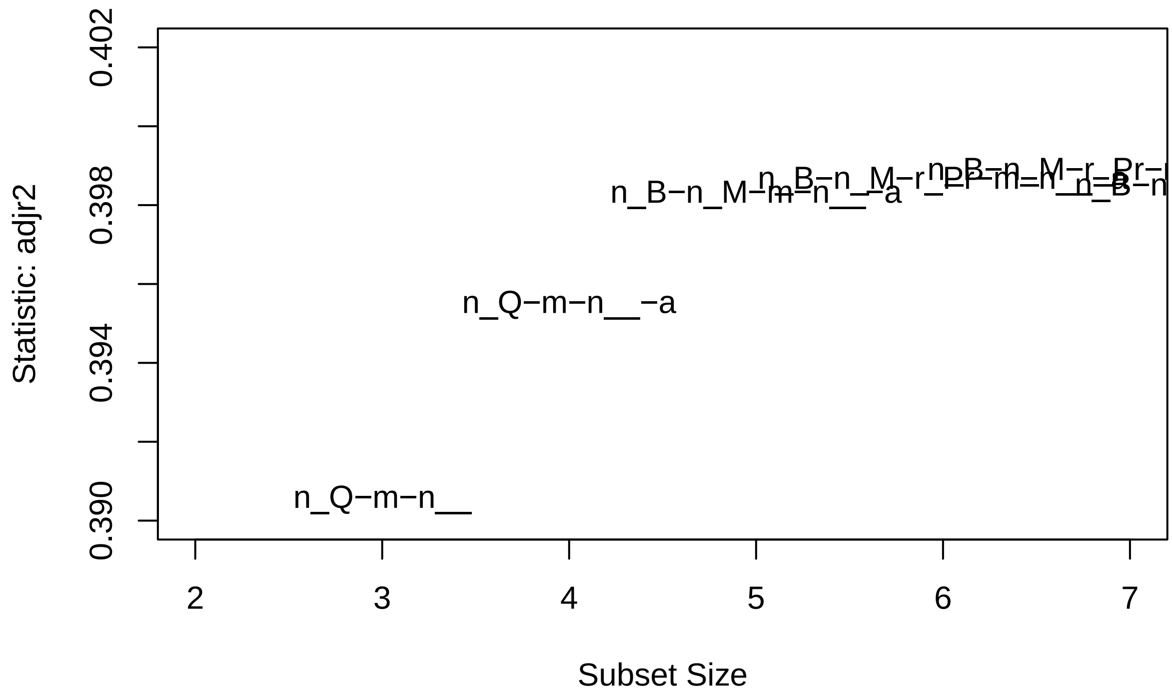
Mallow's Cp

```
                               Abbreviation
neighbourhood_groupBrooklyn             n_B
neighbourhood_groupManhattan            n_M
neighbourhood_groupQueens               n_Q
room_typePrivate room                  r_Pr
room_typeShared room                   r_Sr
price                                     p
minimum_nights                            m
number_of_reviews                       n__
calculated_host_listings_count            c
availability_365                          a
```



- The Mallow's Cp for most of the recommended models fall under the p+1 rule of thumb for a properly fit model. These models do not include the full categorical variable, however, it is picking apart singular variables in the category which is not good practice. This information will be considered but not weighted heavily in deciding the final model.

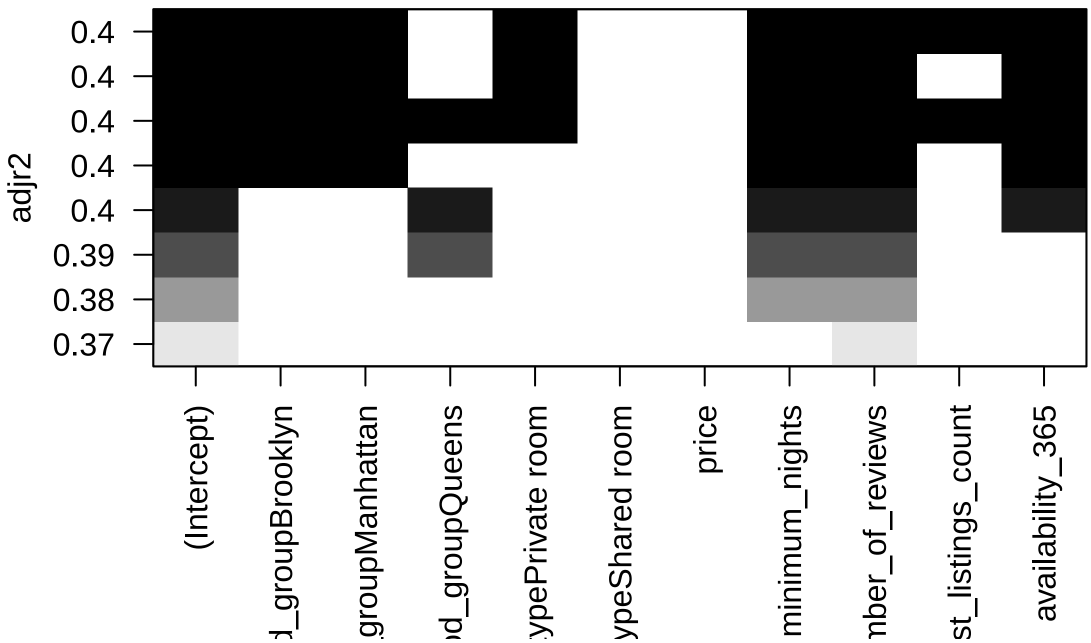
Adjusted R squared value



	Abbreviation
neighbourhood_groupBrooklyn	n_B
neighbourhood_groupManhattan	n_M
neighbourhood_groupQueens	n_Q
room_typePrivate room	r_Pr
room_typeShared room	r_Sr
price	p
minimum_nights	m
number_of_reviews	n__
calculated_host_listings_count	c
availability_365	a

- Adjusted R squared values are very similar between recommended models indicating that all models will have similar predictive power.

Optimized for adjusted R-squared value



- The minimum_nights, availability_365, and number_of_reviews variables are all included in the majority of models that are optimized for the adjusted R-squared value.

Model chosen from variable selection tools

```
Call:
lm(formula = reviews_per_year ~ minimum_nights + availability_365 +
    number_of_reviews + neighbourhood_group, data = NYC)

Residuals:
    Min      1Q  Median      3Q     Max
-52.090  -8.050  -5.504   3.479  72.115

Coefficients:
                              Estimate Std. Error t value Pr(>|t|)
(Intercept)                  15.547899   2.964978   5.244 1.92e-07 ***
minimum_nights               -0.111358   0.029105  -3.826 0.000138 ***
availability_365              0.010523   0.003912   2.690 0.007263 **
number_of_reviews             0.220832   0.009313  23.713  < 2e-16 ***
neighbourhood_groupBrooklyn  -6.997951   2.979934  -2.348 0.019053 *
neighbourhood_groupManhattan -7.247964   2.993194  -2.421 0.015637 *
neighbourhood_groupQueens    -1.881665   3.223636  -0.584 0.559550
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 15.19 on 985 degrees of freedom
```

```
Multiple R-squared:  0.4015,    Adjusted R-squared:  0.3979
F-statistic: 110.2 on 6 and 985 DF,  p-value: < 2.2e-16
```

- Since all of the step AIC selection procedures landed on Number_of_reviews, minimum_nights, availability_365, and neighborhood group as variables to include in the model, I decided these predictors would be good to include.

Final complete model

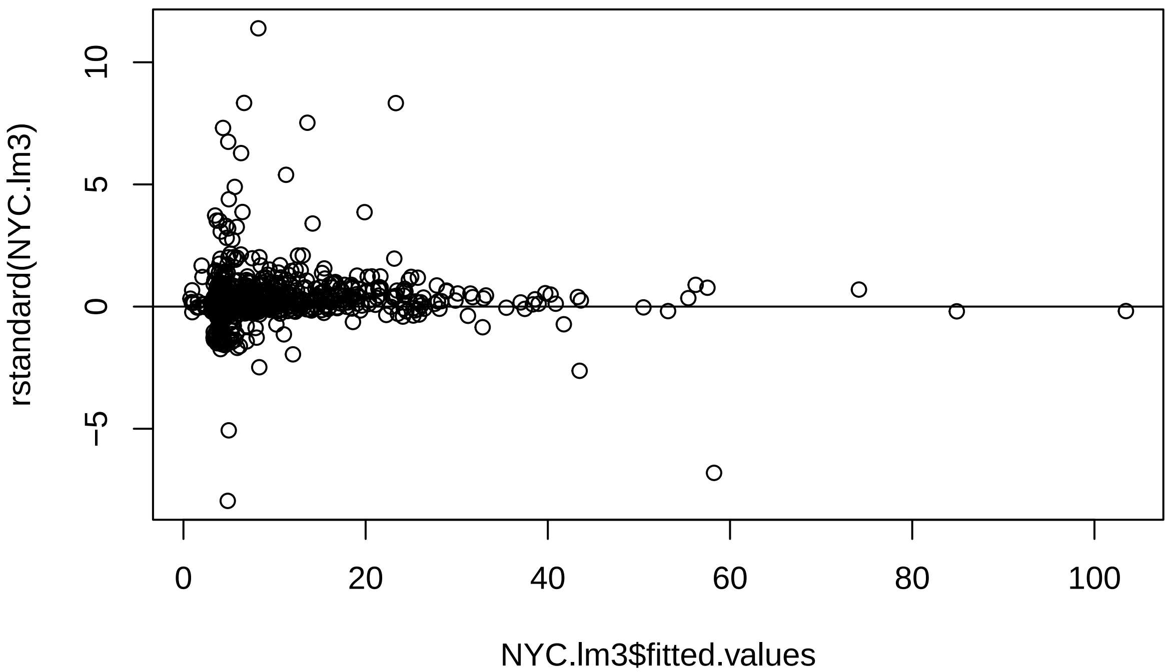
Use your accumulated knowledge with your data set to ideally settle on a model.

Final Model

```
Call:
lm(formula = reviews_per_year ~ minimum_nights + availability_365 +
    number_of_reviews + new_neighbourhood_group, data = NYC,
    weights = minimum_nights^2)

Weighted Residuals:
    Min      1Q  Median      3Q     Max
-511.73   -7.06    2.07   26.16  775.76

Coefficients:
                                         Estimate Std. Error t value Pr(>|t|)
(Intercept)                              1.314274   0.766117   1.715  0.08657
minimum_nights                           0.007726   0.001166   6.624 5.76e-11
availability_365                        -0.002067   0.001605  -1.288  0.19794
number_of_reviews                        0.169161   0.001216 139.088  < 2e-16
new_neighbourhood_groupBrooklyn.Manhattan  2.354999   0.748653   3.146  0.00171
new_neighbourhood_groupQueens            3.461809   1.110356   3.118  0.00188

(Intercept)                              .
minimum_nights                           ***
availability_365
number_of_reviews                        ***
new_neighbourhood_groupBrooklyn.Manhattan **
new_neighbourhood_groupQueens            **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 69.14 on 986 degrees of freedom
Multiple R-squared:  0.9534,    Adjusted R-squared:  0.9531
F-statistic:  4032 on 5 and 986 DF,  p-value: < 2.2e-16
```

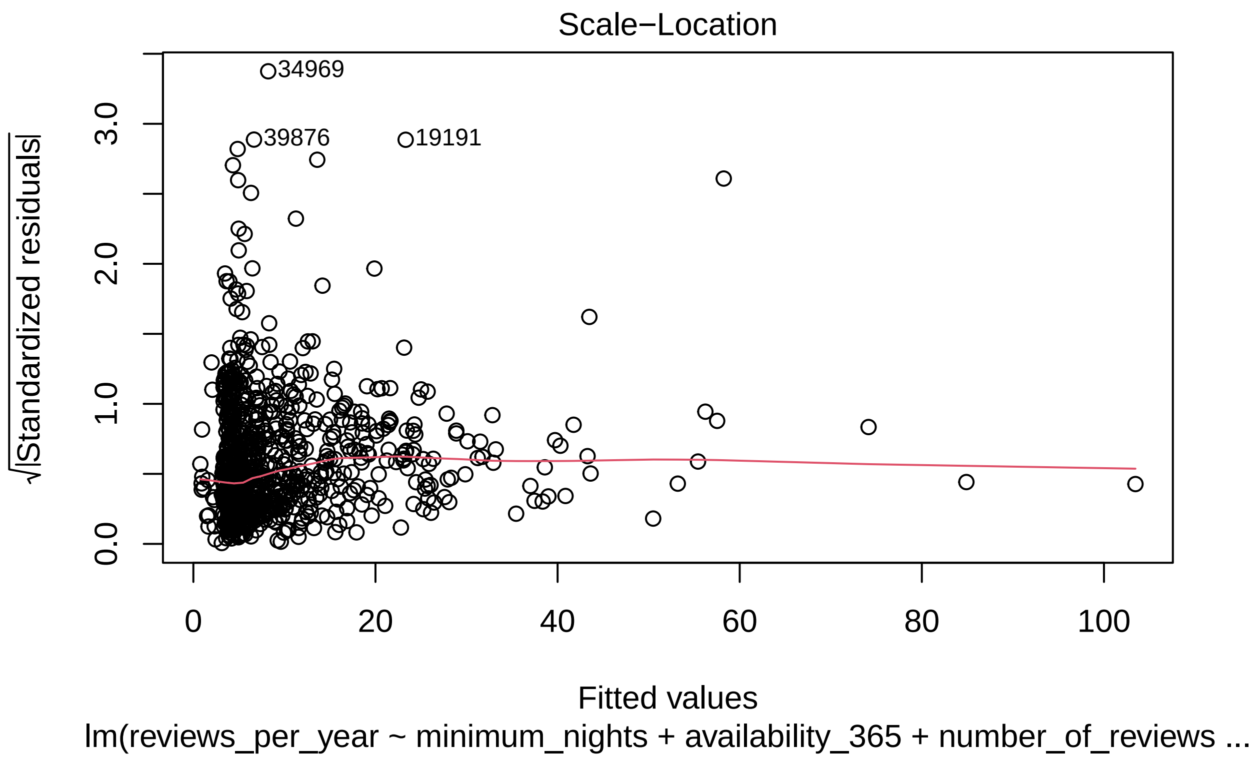
- In the Final model, I also included a weighting of the residuals based on the minimum nights variable, as this raises the correlation coefficient of the model significantly. The residuals became more spread out as the minimum required number of nights to stay increased, which is why I included this in the model fit as a residual weight.

Residuals vs Fitted values



- The residuals are scattered evenly around the centered horizontal line, which indicates the linearity assumption is a good one.

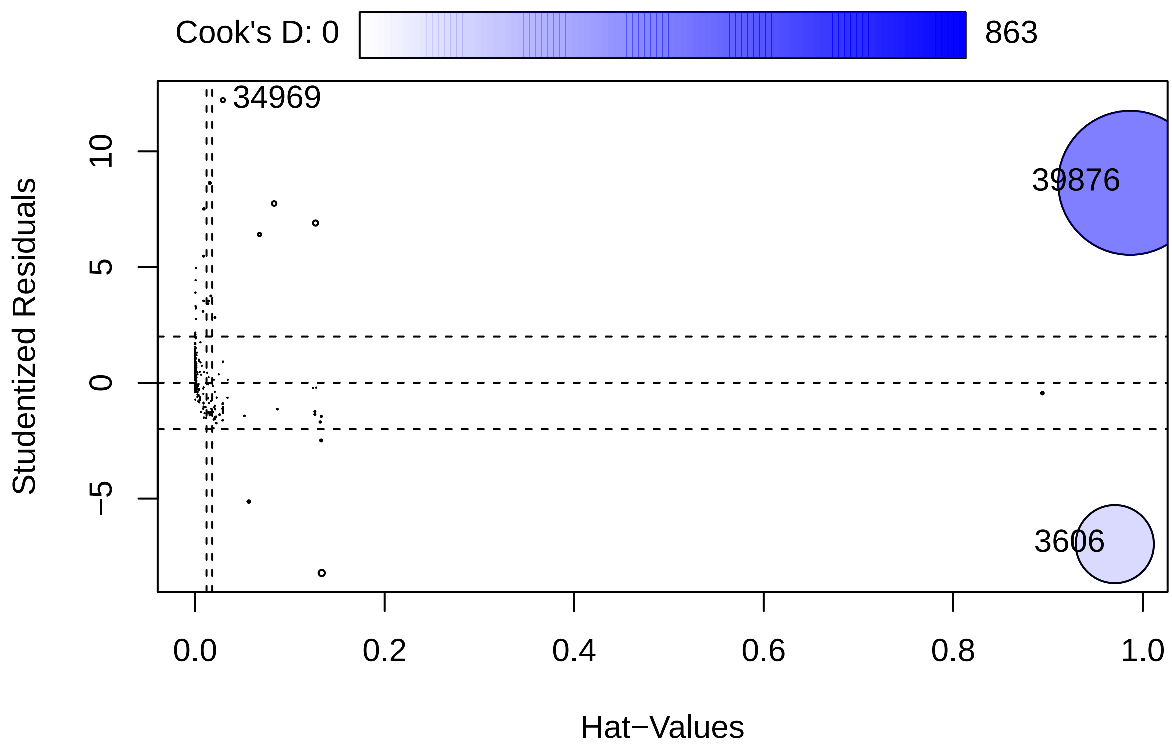
Homoscedasticity




```
Non-constant Variance Score Test
Variance formula: ~ fitted.values
Chisquare = 1.970961, Df = 1, p = 0.16035
```

- The scale-location plot now shows a horizontal line indicating consistent variability of the data. The NCV test results show further confirmation of the results seen in the plot.

Influential Observations




```
       StudRes        Hat       CookD
3606  -6.967999 0.97049129 253.8929144
34969 12.214957 0.02937722   0.6543019
39876  8.641090 0.98676108 863.0821246
```

- There are outliers in this dataset that are skewing results (points 39876 and 3606 on the influence plot above). These were left in as I can not make the judgement of whether or not these points are legitimate. With such a large cook's distance however it is clear there are very influential points in this dataset.

Collinearity detection

```
The condition number is:
```

```
1.40291e+15
```

```
                                   GVIF Df GVIF^(1/(2*Df))
calculated_host_listings_count 1.069648  1        1.034238
price                          1.508643  1        1.228268
minimum_nights                 1.015404  1        1.007673
availability_365               1.135523  1        1.065609
number_of_reviews              1.051299  1        1.025329
neighbourhood_group            1.138201  3        1.021809
room_type                      1.426709  2        1.092909
```

- The VIF indicates that collinearity is not an issue with the predictors directly, so I am not concerned about it with this dataset, as this means the predictor variables are largely orthogonal to each other and add unique predictive power to the data. The condition number is very large, however, which means that there is overlap in principal components.