How should we price homes in Seattle?

In [1]:
```
# Ignore user warnings
import warnings
warnings.simplefilter("ignore", UserWarning)

# Load relevant packages
import pandas as pd
import numpy as np
from scipy import stats
import matplotlib.pyplot as plt
import seaborn as sns
sns.set_style('darkgrid')
import statsmodels.api as sm
import statsmodels.formula.api as smf
import pylab

%matplotlib inline
plt.style.use('ggplot')
```

Introduction (5 mts)

Business Context. You have been hired as a data scientist by a large real estate company in their Seattle office. Your job is to assist Seattle residents willing to sell their home with determining an optimal price to sell their property at in order to maximize their proceeds while still being able to find willing buyers. To do this, the firm would like you to build a pricing model for Seattle real estate, in order to maximize the probability of helping residents close sales (and thus maximizing commissions for the firm).

Business Problem. Your task is to **build a model that uses past sales data in Seattle to recommend an optimal sell price for any particular property**.

Analytical Context. The provided dataset was retrieved from Kaggle (https://www.kaggle.com/harlfoxem/housesalesprediction) and includes sales prices of houses in the state of Washington (King county, where Seattle is located) between May 2014 and May 2015. As we have learned, the primary tool to predict a response variable is the multiple regression model. However, sometimes the assumptions of a linear model are not met by our data. We will learn a set of strategies to mitigate some common issues that appear during regression analysis.

The case is structured as follows: you wil (1) conduct basic EDA of some of the variables to determine that standard linear regression is not sufficient; (2) learn about variable transformations and use these to improve the initial model; and finally (3) learn how to incorporate interaction effects (which are themselves a form of variable transformation involving two or more variables) into our model.

Data exploration (15 mts)

Let's start by reviewing the columns of the dataset and what they mean:

1. **id**: identification for a house
2. **date**: date house was sold
3. **price**: price house was sold at
4. **bedrooms**: number of bedrooms
5. **bathrooms**: number of bathrooms
6. **sqft_living**: square footage of the home
7. **sqft_lot**: square footage of the lot
8. **floors**: total floors (levels) in house
9. **waterfront**: whether or not the house has a view of a waterfront
10. **view**: whether or not the house has been viewed
11. **condition**: how good the condition of the house is
12. **grade**: overall grade given to the housing unit, based on King County grading system
13. **sqft_above**: square footage of the house apart from basement
14. **sqft_basement**: square footage of the basement
15. **yr_built**: year house was built

16. **yr_renovated**: year house was renovated
17. **zipcode**: zipcode of the house
18. **lat**: latitude coordinate of the house
19. **long**: longitude coordinate of the house

In [2]:
```
houses = pd.read_csv('kc_house_data.csv')
```

In [3]:
```
houses.head()
```

Out[3]:

	id	date	price	bedrooms	bathrooms	sqft_living	sqft_lot	floors	w
0	7129300520	20141013T000000	221900.0	3	1.00	1180	5650	1.0	
1	6414100192	20141209T000000	538000.0	3	2.25	2570	7242	2.0	
2	5631500400	20150225T000000	180000.0	2	1.00	770	10000	1.0	
3	2487200875	20141209T000000	604000.0	4	3.00	1960	5000	1.0	
4	1954400510	20150218T000000	510000.0	3	2.00	1680	8080	1.0	

5 rows × 21 columns

Exercise 1: (5 mts)

Analyze the distribution of house prices using `.describe()`, a QQ plot, and a histogram plot. Does it look Gaussian?

Answer. One possible solution is shown below:

In [4]:
```
houses['price'].describe()
```

Out[4]:
```
count    2.161300e+04
mean     5.400881e+05
std      3.671272e+05
```

```
min      7.500000e+04
25%      3.219500e+05
50%      4.500000e+05
75%      6.450000e+05
max      7.700000e+06
Name: price, dtype: float64
```

In [5]:
```
## QQ plot of price
stats.probplot(houses['price'], dist = "norm", plot = plt)
plt.title("QQ Plot for Prices")
plt.show()
```

In [6]:
```
## histogram plot of price
#sns.distplot(houses['price'],fit=stats.laplace, kde=False)
sns.distplot(houses['price'],fit=stats.norm, kde=False)
sns.set(color_codes=True)
plt.xticks(rotation=90)
plt.title("Histogram of prices")
```

Out[6]:
```
Text(0.5, 1.0, 'Histogram of prices')
```

The distribution does not look Gaussian. Looking at both the QQ plot and the histogram, we can see that the distribution of our data is heavily skewed.

Exercise 2: (5 mts)

Analyze the relationship between house prices and price per square foot of living space. What can you conclude? (Hint: use the `lmplot()` function in the `seaborn` library.)

Answer. We can create a regression plot to visualize this relatonship:

In [7]:
```
## linear relation between sqft_living and price
sns.lmplot(x='sqft_living',y='price',data=houses,
           line_kws = {'color': "red"} ,aspect= 2)
plt.title("Price vs. Sqft_living");
```

Given the way that house price vs. price per square foot seems to "fan out", we see that the relationship does not appear to be linear. In fact, it is not immediately obvious what sort of relationship is exhibited at all here.

Variable transformation (30 mts)

We have seen in Exercise 1 that the distribution of house prices is not Gaussian, and that this may be contributing to the "fanning out" effect we observed in Exercise 2. We want to find a way to remove the "fanning out" effect, as it implies that a linear fit becomes less and less suitable, with higher and higher variance from the line of best fit for large values of the predictor and response variables. A common method of addressing this issue is to transform the response variable and/or the predictor variable. Such a **variable transformation** involves applying a known function to one or more of these variables to achieve conditions that are suitable for the application of a linear model.

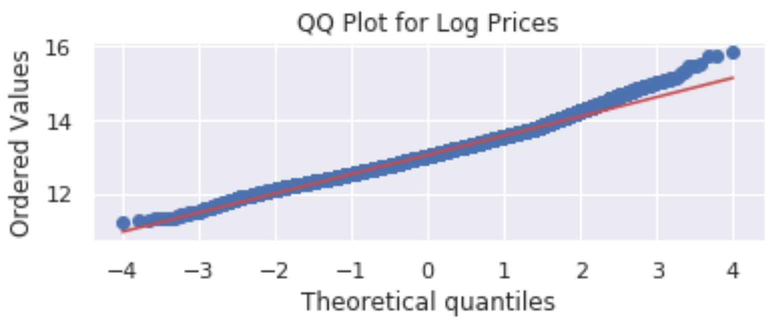
Typical mathematical functions used to transform variables include powers (quadratic, cubic, square root, etc.), logarithms, and trigonometric functions. Let's start with the logarithmic transformation to see if we can achieve some results.

Exercise 3: (5 mts)

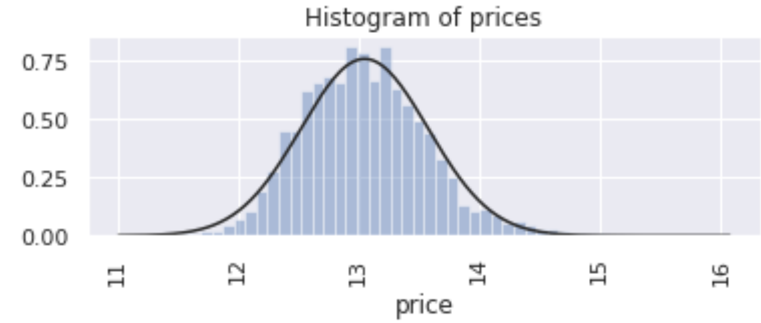
Take the logarithm of house prices and create plots to ascertain if this makes the distribution of the transformed variable roughly Gaussian.

Answer. One possible solution is shown below:

In [8]:
```
## QQ plot of price
plt.subplot(2,1,1)
stats.probplot(np.log(houses['price']), dist = "norm", plot = plt)
plt.title("QQ Plot for Log Prices")
plt.show()
plt.subplot(2,1,2)
sns.distplot(np.log(houses['price']),fit=stats.norm, kde=False)
sns.set(color_codes=True)
plt.xticks(rotation=90)
plt.title("Histogram of prices")
```



Out[8]:
```
Text(0.5, 1.0, 'Histogram of prices')
```



In [9]:
```
np.log(houses['price']).describe()
```

Out[9]:
```
count    21613.000000
mean        13.047817
std          0.526685
min         11.225243
25%         12.682152
50%         13.017003
75%         13.377006
max         15.856731
Name: price, dtype: float64
```

We can see from both the QQ plot and the histogram that the distribution is far closer to normal.

Building a linear model with transformed variables (15 mts)

Of course, we aren't just restricted to applying the logarithmic transformation to house prices; we can do it to any other variable in our dataset. Let's transform both house prices and price per square foot by this method and interpret the resulting linear model:

In [10]:
```
mod1 = smf.ols(formula='np.log(price) ~ np.log(sqft_living)', data=hous
es).fit()
print(mod1.summary())
```

```
                            OLS Regression Results
```

```
==============================================================================
=======
Dep. Variable:          np.log(price)   R-squared:
  0.456
Model:                            OLS   Adj. R-squared:
  0.455
Method:                 Least Squares   F-statistic:                     1.
808e+04
Date:                Fri, 29 May 2020   Prob (F-statistic):
   0.00
Time:                        16:02:55   Log-Likelihood:
-10240.
No. Observations:               21613   AIC:                             2.
048e+04
Df Residuals:                   21611   BIC:                             2.
050e+04
Df Model:                           1

Covariance Type:            nonrobust

==============================================================================
================
                          coef    std err          t      P>|t|
[0.025      0.975]
------------------------------------------------------------------------------
----------------
Intercept               6.7299      0.047    143.001      0.000
6.638       6.822
np.log(sqft_living)     0.8368      0.006    134.459      0.000
0.825       0.849
==============================================================================
=======
Omnibus:                      123.344   Durbin-Watson:
  1.978
Prob(Omnibus):                  0.000   Jarque-Bera (JB):
113.759
Skew:                           0.142   Prob(JB):
1.98e-25
Kurtosis:                       2.787   Cond. No.
```

```
   137.
==============================================================================
Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
```

We have to be mindful of how we interpret the coefficients. Although we could say that our results tell us that a 1 unit increase in the logarithm of living space will result in a 0.836 increase in the logarithm of the price, this is a very mechanical and not at all intuitive interpretation.

Mathematics can help us come up with a more intuitive interpretation. Note that the fit our above model has come up with is $\log price = 0.84 * \log sqft\_living + 6.73$. Exponentiating both sides, we get $price = e^{6.73} * sqft\_living^{0.84}$. This is a nonlinear relationship, so it's not as straightforward as "increasing `sqft_living` by 1 means that `price` goes up by X".

However, we can try reframing this in percentage terms; i.e. how does a 1 percent increase in `sqft_living` affect price? We can plug $sqft\_living_0 = 1.01 * sqft\_living$ into this equation to get $price_0 = e^{6.73} * 1.01^{0.84} * sqft\_living^{0.84} = 1.01^{0.84} * price \approx 1.0084 * price$; i.e. a 1 percent increase in living space results in a 0.84 percent increase in price. This percentage vs. percentage change comparison is known as **elasticity**.

Let's now build a linear model where the logarithmic transform is only applied to the house prices:

In [11]:
```python
mod2 = smf.ols(formula='np.log(price) ~ sqft_living', data=houses).fit()
print(mod2.summary())
```

```
                            OLS Regression Results
==============================================================================
```

```
Dep. Variable:          np.log(price)   R-squared:
  0.483
Model:                            OLS   Adj. R-squared:
  0.483
Method:                 Least Squares   F-statistic:                 2.
023e+04
Date:                Fri, 29 May 2020   Prob (F-statistic):
   0.00
Time:                        16:02:57   Log-Likelihood:
-9670.2
No. Observations:               21613   AIC:                         1.
934e+04
Df Residuals:                   21611   BIC:                         1.
936e+04
Df Model:                           1

Covariance Type:            nonrobust

======================================================================
========
                  coef    std err          t      P>|t|      [0.025
  0.975]
----------------------------------------------------------------------
--------
Intercept      12.2185      0.006   1916.883      0.000      12.206
  12.231
sqft_living     0.0004    2.8e-06    142.233      0.000       0.000
   0.000
======================================================================
=======
Omnibus:                        3.128   Durbin-Watson:
  1.979
Prob(Omnibus):                  0.209   Jarque-Bera (JB):
  3.149
Skew:                           0.027   Prob(JB):
  0.207
Kurtosis:                       2.974   Cond. No.
5.63e+03
======================================================================
=======
```

```
Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is
correctly specified.
[2] The condition number is large, 5.63e+03. This might indicate that t
here are
strong multicollinearity or other numerical problems.
```

The interpretation of the regression coefficient is once again different. We interpret the coefficient as a **semi-elasticity**, where an absolute increase in `sqft_living` (because it has not had the logarithm function applied to it) corresponds to a percentage increase `price`. Specifically, here we can say that an increase in living space by 1 square foot leads to a 0.04% percent increase in price.

Exercise 4: (10 mts)

Using the `sns.lmplot()` function, determine which of the above two models is "more linear".

Answer. One possible solution is given below:

In [12]:
```
houses['log_price'] = np.log(houses['price'])
houses['log_sqft_living'] = np.log(houses['sqft_living'])
```

In [13]:
```
## linear relation between sqft_living and log-price
sns.lmplot(x='sqft_living',y='log_price',data=houses,
           line_kws = {'color': "red"} ,aspect= 2, scatter_kws={"s": 5
})
plt.title("Log-Price vs. Sqft_living");
```

In [14]:
```
## linear relation between log-sqft_living and log-price
sns.lmplot(x='log_sqft_living',y='log_price',data=houses,
           line_kws = {'color': "red"} ,aspect= 2, scatter_kws={"s": 5
})
plt.title("Log-Price vs. Log_Sqft_living");
```

We can see from these plots that the data points of the log-log model cluster more uniformly around the line of best fit across different levels of the predictor variable as compared to the other model, suggesting that the log-log model is more linear.

Box-Cox transformation (5 mts)

Logarithmic transformations are just one of the possible transformations that we discussed. Earlier, we mentioned powers (e.g. squares, cubes, square roots, etc.) as well as trigonometric functions. In some cases, choosing a transformation can be straightforward (e.g. the logarithm because it is easily interpretable); other times, it is much more difficult. A formal way to decide on which transformation to use is to estimate the coefficient λ of the Box-Cox transformation:

$$BC(\lambda) = \frac{Y^\lambda - 1}{\lambda}$$

If the estimate of λ is close to 2, we can use the quadratic transformation; if it is close to 0.5, the square root transformation; if it is close to zero or less than zero (negative), the logarithmic transformation; etc. In our case, we have:

In [15]:
```
price,fitted_lambda = stats.boxcox(houses['price'])
round(fitted_lambda,2)
```

Out[15]:
```
-0.23
```

This is less than zero, so it would seem that using the logarithmic transformation is sensible.

Multiple linear regression with transformed variables (25 mts)

Of course, as we have seen from the previous case, it doesn't make sense to restrict ourselves to modeling house prices based on only one predictor variable. Let's add in several more variables, some transformed and some not:

Exercise 5: (5 mts)

Fit a linear model of log `price` vs. log `sqft_living`, log `sqft_lot`, `bedrooms`, `floors`, `bathrooms`, `waterfront`, `condition`, `view`, `grade`, `yr_built`, `lat`, and `long`. Provide interpretations for the coefficients of log `sqft_living` and `waterfront`.

Answer. One possible solution is given below:

In [16]:
```
mod3 = smf.ols (formula = 'np.log(price) ~ np.log(sqft_living)+ np.log
(sqft_lot) +bedrooms + floors + bathrooms +waterfront + condition + wat
erfront + view + grade + yr_built + lat + long ', data = houses).fit()
print(mod3.summary())
```

```
                            OLS Regression Results
```

```
==============================================================================
=======
Dep. Variable:           np.log(price)   R-squared:                       
  0.762
Model:                             OLS   Adj. R-squared:                  
  0.762
Method:                  Least Squares   F-statistic:                     
  5772.
Date:                 Fri, 29 May 2020   Prob (F-statistic):              
   0.00
Time:                         16:03:06   Log-Likelihood:                 
-1284.6
No. Observations:                21613   AIC:                             
  2595.
Df Residuals:                    21600   BIC:                             
  2699.
Df Model:                           12                                    

Covariance Type:             nonrobust                                    

==============================================================================
================
                          coef    std err          t      P>|t|      
[0.025      0.975]
------------------------------------------------------------------------------
----------------
Intercept             -39.9251      2.036    -19.609      0.000     -4
3.916     -35.934
np.log(sqft_living)     0.4096      0.009     46.679      0.000       
0.392       0.427
np.log(sqft_lot)       -0.0076      0.002     -3.070      0.002      -
0.012      -0.003
bedrooms               -0.0256      0.002    -10.265      0.000      -
0.031      -0.021
floors                  0.0490      0.004     11.302      0.000       
0.040       0.057
bathrooms               0.0705      0.004     17.458      0.000       
0.063       0.078
```

```
waterfront                 0.3911      0.022     17.680      0.000
0.348       0.434
condition                  0.0552      0.003     18.832      0.000
0.049       0.061
view                       0.0715      0.003     27.024      0.000
0.066       0.077
grade                      0.1858      0.003     74.252      0.000
0.181       0.191
yr_built                  -0.0038   8.66e-05    -44.115      0.000      -
0.004      -0.004
lat                        1.3443      0.013    100.583      0.000
1.318       1.371
long                       0.0673      0.015      4.446      0.000
0.038       0.097
==============================================================================
=======
Omnibus:                      568.294   Durbin-Watson:
  1.981
Prob(Omnibus):                  0.000   Jarque-Bera (JB):             1
119.493
Skew:                           0.182   Prob(JB):                     8.
04e-244
Kurtosis:                       4.054   Cond. No.
2.30e+06
==============================================================================
=======

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is
correctly specified.
[2] The condition number is large, 2.3e+06. This might indicate that th
ere are
strong multicollinearity or other numerical problems.
```

All the variables are statistically significant (all p - values less than 0.01). Overall this linear model explains over 76 percent of the total variability of the response variable.

An increase of one percent in living space leads to an increase of 0.4096 percent in price. A property with a water view has an increase in price of 39.11 percent.

What other factors may impact the price that we have left out? Some that may play a role include proximity to services (hospitals, schools, commercial areas, movie theaters, metro stops...), crime rates, etc. Our dataset does not have a comprehensive list of possible factors; however, we do have some variables that would be interesting to investigate further.

In general, house prices change depending on the location. Two houses with comparable features can be priced very differently depending on the neighborhood and geographic position. In this dataset, we have zipcode and geographic coordinates. Let us start by taking a look at the relationship between latitude and prices.

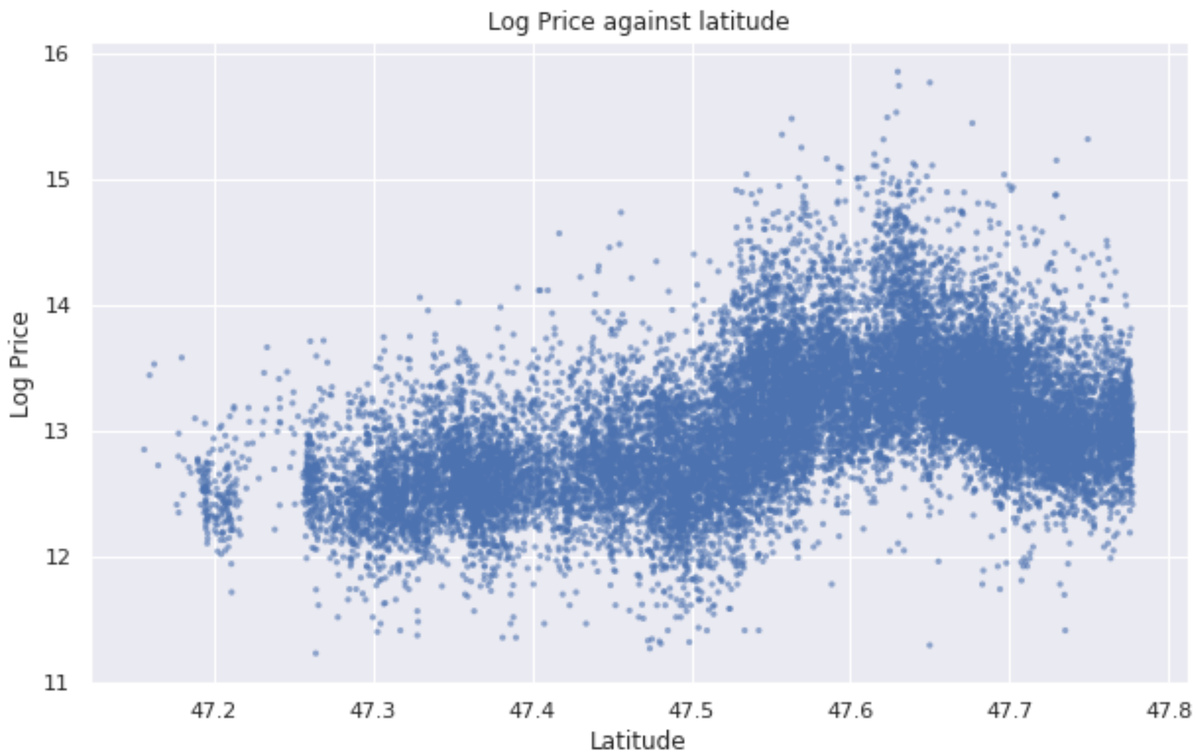
Exercise 6: (5 mts)

Examine the relationship between latitude and the logarithm of house prices. What do you observe?

Answer. One possible solution is shown below:

In [17]:

```
fig = plt.figure(figsize = (10,6))
ax = fig.add_subplot(111)
ax.scatter(houses['lat'],np.log(houses['price']),c ='b',alpha=0.6,edgec
olors='none',s=10)
plt.xlabel("Latitude")
plt.ylabel("Log Price")
ax.set_title("Log Price against latitude")
plt.show();
```



We can see that there is a nonlinear relationship between latitude and price. Based on the concave curvature in the right half of the plot above, it seems that adding a quadratic term would be able to help us explain this.

Exercise 7: (10 mts)

Add the square of the latitude as an additional predictor to the model in Exercise 5. Is the term significant? Use the AIC score (described below) to evaluate whether or not the fit has improved.

Background on AIC Score: One of the drawbacks of R^2 is that it can never decrease when the set of predictors is increased. In other words, there is no penalty for continuing to add variables that have little explanatory power. Consequently, selecting predictor variables trying to maximize

R^2 can lead to choosing unnecessarily complex and redundant models. How do we choose a model that offers a good quality of fit while minimizing the number of features?

There are several model selection criteria that quantify the quality of a model by managing the tradeoff between goodness-of-fit and simplicity. The most common one is the AIC (Akaike Information Criterion). The AIC penalizes the addition of more terms to a model, so in order for an updated model to have a better AIC, its R^2 needs to improve at least as much as the additional imposed penalty. The smaller a model's AIC, the higher its quality.

For now, do not worry about the technical details behind AIC (although you are free to look them up yourself). In future cases on **regularization**, you will learn more about the rationale behind why these sorts of estimators matter and how to construct and use them in model-building.

Answer. One possible solution is shown below:

In [18]:
```
## lat square effect
mod4 = smf.ols ( formula = 'np.log(price) ~ np.log(sqft_living)+ np.log
(sqft_lot) +bedrooms + floors + bathrooms +waterfront + condition + wat
erfront + view + grade + yr_built + lat + I(lat**2) + long ', data = ho
uses  ).fit()
print(mod4.summary())
```

```
                            OLS Regression Results

==============================================================================
=======
Dep. Variable:          np.log(price)   R-squared:
  0.778
Model:                            OLS   Adj. R-squared:
  0.778
Method:                 Least Squares   F-statistic:
  5816.
Date:                Fri, 29 May 2020   Prob (F-statistic):
   0.00
Time:                        16:03:11   Log-Likelihood:
-554.41
No. Observations:               21613   AIC:
  1137
```

```
  1137.
Df Residuals:                   21599   BIC:
  1249.
Df Model:                          13

Covariance Type:            nonrobust

========================================================================
================
                          coef    std err          t      P>|t|
[0.025      0.975]
------------------------------------------------------------------------
----------------
Intercept           -7773.2083    199.024    -39.057      0.000    -816
3.311   -7383.106
np.log(sqft_living)     0.4084      0.008     48.137      0.000
0.392       0.425
np.log(sqft_lot)        0.0122      0.002      4.968      0.000
0.007       0.017
bedrooms               -0.0253      0.002    -10.498      0.000       -
0.030      -0.021
floors                  0.0488      0.004     11.643      0.000
0.041       0.057
bathrooms               0.0660      0.004     16.913      0.000
0.058       0.074
waterfront              0.3755      0.021     17.555      0.000
0.334       0.417
condition               0.0598      0.003     21.083      0.000
0.054       0.065
view                    0.0687      0.003     26.853      0.000
0.064       0.074
grade                   0.1763      0.002     72.508      0.000
0.172       0.181
yr_built               -0.0032   8.53e-05    -37.186      0.000       -
0.003      -0.003
lat                   326.2171      8.361     39.019      0.000      30
9.830     342.604
I(lat ** 2)            -3.4174      0.088    -38.858      0.000       -
3.590      -3.245
long                   -0.0232      0.015     -1.568      0.117       -
```

```
0.052       0.006
======================================================================
=======
Omnibus:                      649.112   Durbin-Watson:
  1.984
Prob(Omnibus):                  0.000   Jarque-Bera (JB):             1
420.202
Skew:                           0.175   Prob(JB):                    4.
05e-309
Kurtosis:                       4.206   Cond. No.
3.54e+08
======================================================================
=======

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is
correctly specified.
[2] The condition number is large, 3.54e+08. This might indicate that t
here are
strong multicollinearity or other numerical problems.
```

In [19]:
```
print(mod3.aic)
```

```
2595.2534342787694
```

In [20]:
```
print(mod4.aic)
```

```
1136.8143169405448
```

We can see that the coefficient is highly significant. The R-squared has increased by about 1%. Comparing model 3 and model 4 we can see that the AIC has improved from 2597.253 to 1138.814.

Let's now add the zipcode to our model:

In [21]:
```
## location effect with dummy variables
```

```
mod5 = smf.ols ( formula = 'np.log(price) ~ np.log(sqft_living)+ np.log
(sqft_lot) +bedrooms + floors + bathrooms +waterfront + condition + wat
erfront + view + grade + yr_built + lat + I(lat**2) + long + C(zipcod
e)', data = houses  ).fit()
print(mod5.summary())
```

```
                            OLS Regression Results

=======================================================================
=======
Dep. Variable:          np.log(price)   R-squared:
  0.879
Model:                            OLS   Adj. R-squared:
  0.879
Method:                 Least Squares   F-statistic:
  1910.
Date:                Fri, 29 May 2020   Prob (F-statistic):
   0.00
Time:                        16:03:15   Log-Likelihood:
6026.8
No. Observations:               21613   AIC:                         -1.
189e+04
Df Residuals:                   21530   BIC:                         -1.
123e+04
Df Model:                          82

Covariance Type:            nonrobust

=======================================================================
================
                          coef    std err          t      P>|t|
[0.025      0.975]
-----------------------------------------------------------------------
----------------
Intercept           -5882.6548    526.169    -11.180      0.000    -691
3.984   -4851.325
C(zipcode)[T.98002]     0.0200      0.017      1.208      0.227       -
0.012       0.052
C(zipcode)[T.98003]    -0.0119      0.015     -0.804      0.421       -
0.041       0.017
```

```
C(zipcode)[T.98004]     0.9101      0.028     32.355      0.000
0.855       0.965
C(zipcode)[T.98005]     0.5146      0.030     17.169      0.000
0.456       0.573
C(zipcode)[T.98006]     0.4547      0.026     17.704      0.000
0.404       0.505
C(zipcode)[T.98007]     0.4419      0.031     14.293      0.000
0.381       0.503
C(zipcode)[T.98008]     0.4523      0.030     15.326      0.000
0.394       0.510
C(zipcode)[T.98010]     0.3120      0.025     12.308      0.000
0.262       0.362
C(zipcode)[T.98011]     0.2602      0.037      7.108      0.000
0.188       0.332
C(zipcode)[T.98014]     0.1927      0.041      4.747      0.000
0.113       0.272
C(zipcode)[T.98019]     0.2136      0.040      5.393      0.000
0.136       0.291
C(zipcode)[T.98022]     0.3399      0.024     13.880      0.000
0.292       0.388
C(zipcode)[T.98023]    -0.0631      0.014     -4.648      0.000      -
0.090      -0.036
C(zipcode)[T.98024]     0.3140      0.037      8.503      0.000
0.242       0.386
C(zipcode)[T.98027]     0.3753      0.026     14.191      0.000
0.323       0.427
C(zipcode)[T.98028]     0.2015      0.036      5.667      0.000
0.132       0.271
C(zipcode)[T.98029]     0.4724      0.030     15.928      0.000
0.414       0.530
C(zipcode)[T.98030]     0.0020      0.017      0.118      0.906      -
0.032       0.036
C(zipcode)[T.98031]    -0.0240      0.019     -1.292      0.196      -
0.060       0.012
C(zipcode)[T.98032]    -0.1287      0.020     -6.313      0.000      -
0.169      -0.089
C(zipcode)[T.98033]     0.5712      0.031     18.518      0.000
0.511       0.632
C(zipcode)[T.98034]     0.3229      0.033      9.881      0.000
```

```
0.259       0.387
C(zipcode)[T.98038]      0.1920      0.019      9.995      0.000
0.154       0.230
C(zipcode)[T.98039]      1.0861      0.037     29.279      0.000
1.013       1.159
C(zipcode)[T.98040]      0.6648      0.026     25.771      0.000
0.614       0.715
C(zipcode)[T.98042]      0.0526      0.016      3.199      0.001
0.020       0.085
C(zipcode)[T.98045]      0.3142      0.036      8.834      0.000
0.245       0.384
C(zipcode)[T.98052]      0.4500      0.031     14.297      0.000
0.388       0.512
C(zipcode)[T.98053]      0.4481      0.034     13.292      0.000
0.382       0.514
C(zipcode)[T.98055]     -0.0070      0.021     -0.328      0.743      -
0.049       0.035
C(zipcode)[T.98056]      0.1444      0.023      6.239      0.000
0.099       0.190
C(zipcode)[T.98058]      0.0387      0.020      1.912      0.056      -
0.001       0.078
C(zipcode)[T.98059]      0.2042      0.023      8.974      0.000
0.160       0.249
C(zipcode)[T.98065]      0.3722      0.033     11.197      0.000
0.307       0.437
C(zipcode)[T.98070]      0.0356      0.025      1.443      0.149      -
0.013       0.084
C(zipcode)[T.98072]      0.3015      0.036      8.287      0.000
0.230       0.373
C(zipcode)[T.98074]      0.3943      0.031     12.847      0.000
0.334       0.454
C(zipcode)[T.98075]      0.4312      0.030     14.342      0.000
0.372       0.490
C(zipcode)[T.98077]      0.2904      0.038      7.676      0.000
0.216       0.365
C(zipcode)[T.98092]      0.0821      0.015      5.565      0.000
0.053       0.111
C(zipcode)[T.98102]      0.6994      0.032     21.700      0.000
0.636       0.763
```

```
C(zipcode)[T.98103]      0.5426      0.030     18.208      0.000
0.484        0.601
C(zipcode)[T.98105]      0.6933      0.031     22.540      0.000
0.633        0.754
C(zipcode)[T.98106]      0.0775      0.024      3.215      0.001
0.030        0.125
C(zipcode)[T.98107]      0.5516      0.031     17.928      0.000
0.491        0.612
C(zipcode)[T.98108]      0.1021      0.026      3.900      0.000
0.051        0.153
C(zipcode)[T.98109]      0.7096      0.032     22.107      0.000
0.647        0.772
C(zipcode)[T.98112]      0.7930      0.029     27.537      0.000
0.737        0.849
C(zipcode)[T.98115]      0.5516      0.030     18.263      0.000
0.492        0.611
C(zipcode)[T.98116]      0.4565      0.026     17.471      0.000
0.405        0.508
C(zipcode)[T.98117]      0.5155      0.031     16.889      0.000
0.456        0.575
C(zipcode)[T.98118]      0.2292      0.024      9.680      0.000
0.183        0.276
C(zipcode)[T.98119]      0.6845      0.030     22.566      0.000
0.625        0.744
C(zipcode)[T.98122]      0.5481      0.028     19.758      0.000
0.494        0.602
C(zipcode)[T.98125]      0.3041      0.032      9.399      0.000
0.241        0.367
C(zipcode)[T.98126]      0.2688      0.025     10.968      0.000
0.221        0.317
C(zipcode)[T.98133]      0.1944      0.033      5.827      0.000
0.129        0.260
C(zipcode)[T.98136]      0.3992      0.025     15.937      0.000
0.350        0.448
C(zipcode)[T.98144]      0.4202      0.026     15.981      0.000
0.369        0.472
C(zipcode)[T.98146]      0.0226      0.023      0.985      0.325      -
0.022        0.067
C(zipcode)[T.98148]     -0.0319      0.029     -1.090      0.276      -
```

```
0.089      0.025
C(zipcode)[T.98155]     0.1852      0.035      5.322      0.000
0.117      0.253
C(zipcode)[T.98166]     0.0839      0.021      4.013      0.000
0.043      0.125
C(zipcode)[T.98168]    -0.1574      0.022     -7.066      0.000      -
0.201     -0.114
C(zipcode)[T.98177]     0.3235      0.035      9.275      0.000
0.255      0.392
C(zipcode)[T.98178]    -0.0625      0.023     -2.716      0.007      -
0.108     -0.017
C(zipcode)[T.98188]    -0.0918      0.023     -3.999      0.000      -
0.137     -0.047
C(zipcode)[T.98198]    -0.0715      0.017     -4.166      0.000      -
0.105     -0.038
C(zipcode)[T.98199]     0.5506      0.029     18.691      0.000
0.493      0.608
np.log(sqft_living)     0.4229      0.006     66.616      0.000
0.410      0.435
np.log(sqft_lot)        0.0703      0.002     33.767      0.000
0.066      0.074
bedrooms               -0.0148      0.002     -8.202      0.000      -
0.018     -0.011
floors                  0.0194      0.003      5.966      0.000
0.013      0.026
bathrooms               0.0383      0.003     13.145      0.000
0.033      0.044
waterfront              0.4698      0.016     29.193      0.000
0.438      0.501
condition               0.0418      0.002     19.413      0.000
0.038      0.046
view                    0.0640      0.002     32.841      0.000
0.060      0.068
grade                   0.1117      0.002     58.285      0.000
0.108      0.115
yr_built               -0.0006   7.15e-05     -7.745      0.000      -
0.001     -0.000
lat                   245.3030     22.157     11.071      0.000     20
1.873    288.733
```

```
I(lat ** 2)             -2.5750      0.233    -11.045      0.000      -
3.032      -2.118
long                    -0.4050      0.052     -7.749      0.000      -
0.507      -0.303
======================================================================
=======
Omnibus:                     1422.131   Durbin-Watson:
  2.001
Prob(Omnibus):                  0.000   Jarque-Bera (JB):             5
908.533
Skew:                          -0.193   Prob(JB):
   0.00
Kurtosis:                       5.532   Cond. No.
1.27e+09
======================================================================
=======

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is
correctly specified.
[2] The condition number is large, 1.27e+09. This might indicate that t
here are
strong multicollinearity or other numerical problems.
```

From the output, we can see how many of the different ZIP codes exert a significant effect. The R-squared of the model has increased dramatically to 87.92%.

Modeling interaction effects (50 mts)

As we have seen during the EDA cases, interaction effects can complicate the perceived effect of the predictor variables on the outcome of interest. Let's dig into potential interactions by looking at three of the predictors in tandem: `waterfront`, geographic position (`lat` and `long`), and `sqft_living`. Specifically, is the effect of geographic position and `sqft_living` different among the houses that have a waterfront view vs. those that do not?

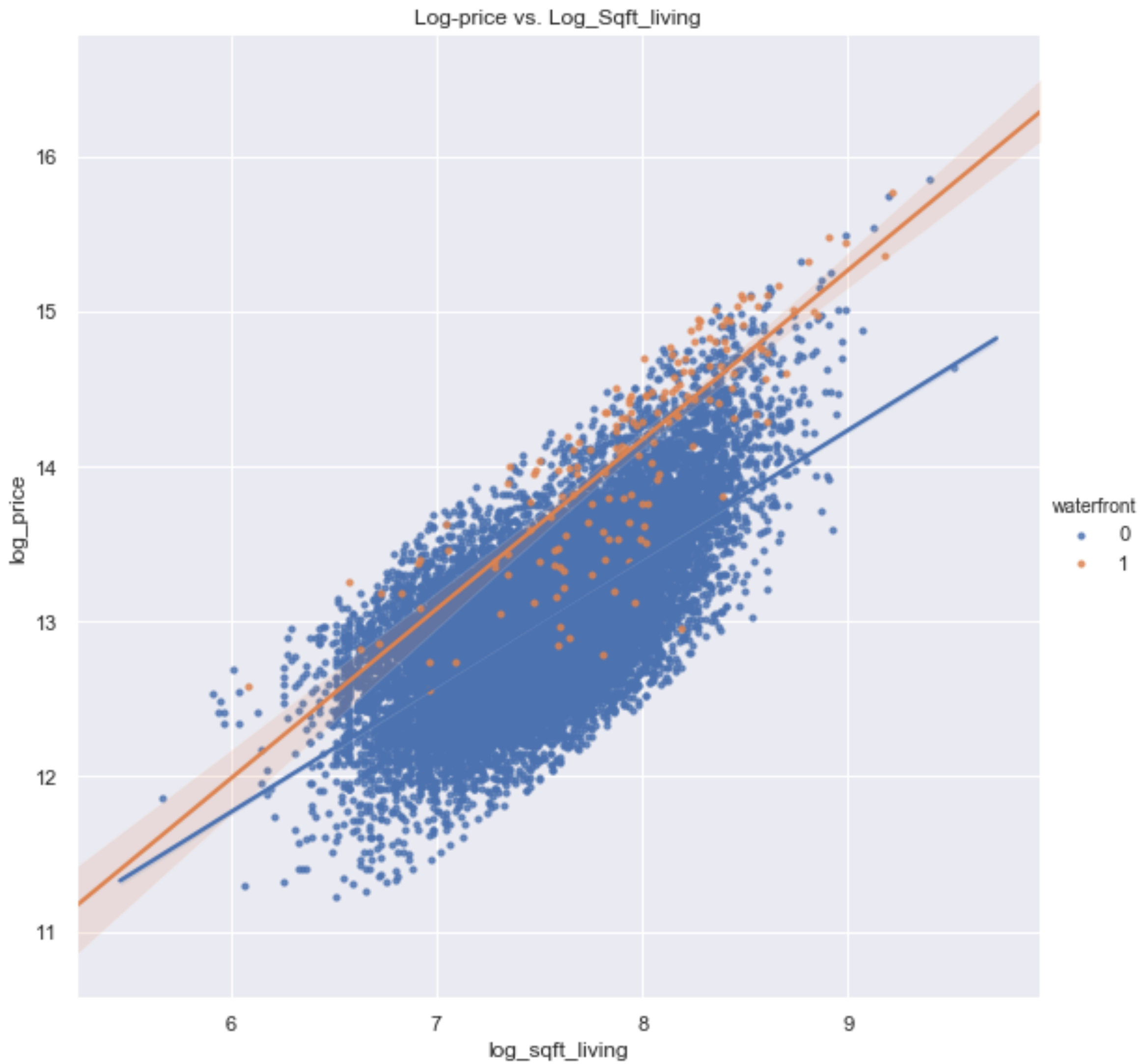
To study this interaction effect, let's fit two separate regression lines for each subgroup (here, the two subgroups are having a waterfront or not):

In [61]:
```
sns.lmplot(x = 'lat', y= 'log_price',data = houses,
           hue="waterfront", height=8, scatter_kws={"s": 10})
plt.title("Log-price vs. Latitude")
```

Out[61]:
```
Text(0.5, 1, 'Log-price vs. Latitude')
```

In [62]:
```
sns.lmplot(x='log_sqft_living', y='log_price',data=houses,
              hue="waterfront", height=8, scatter_kws={"s": 10})
plt.title("Log-price vs. Log_Sqft_living")
```

Out[62]:
```
Text(0.5, 1, 'Log-price vs. Log_Sqft_living')
```



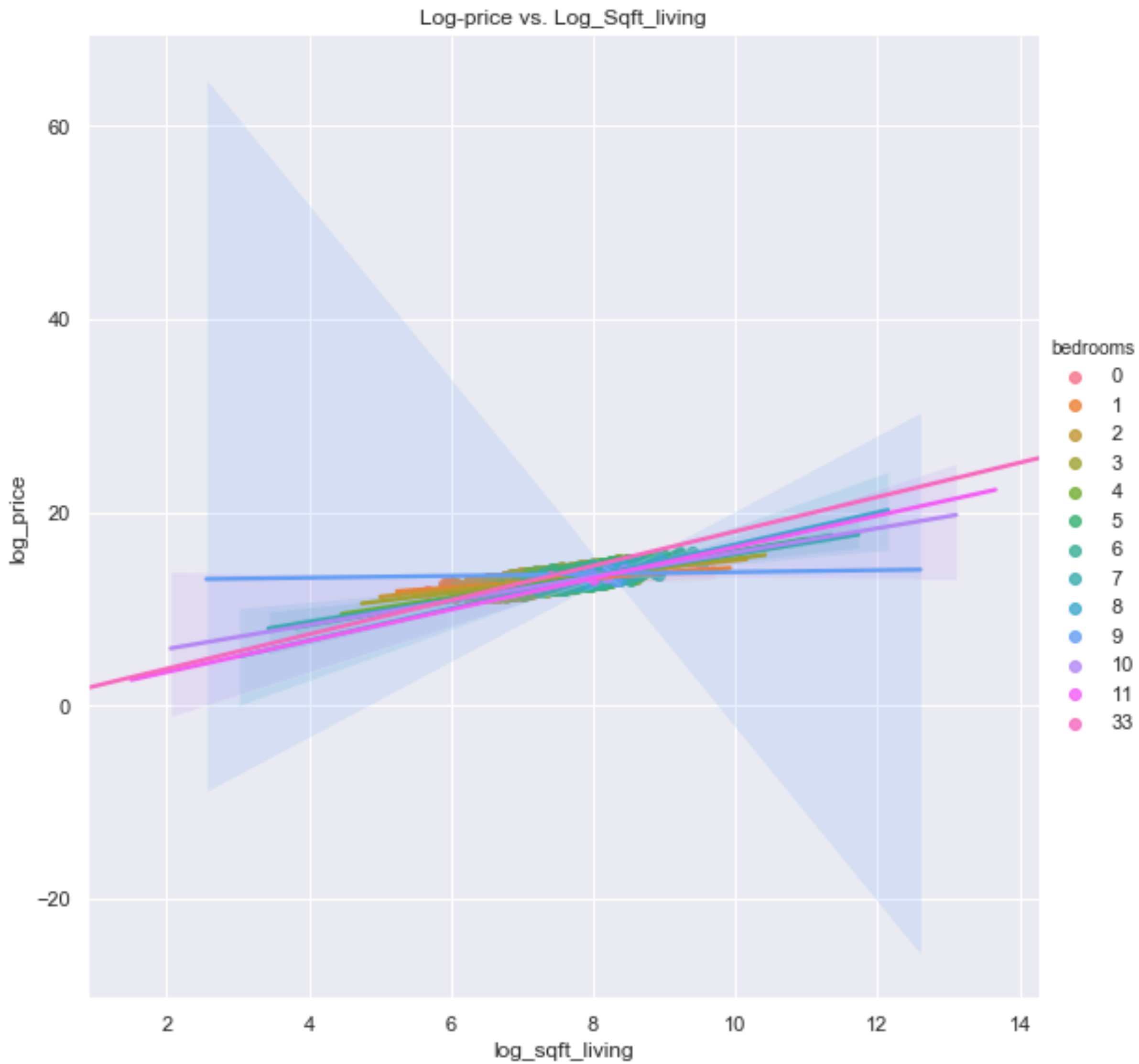
We see that the effects of `latitude` and the logarithm of `sqft_living` are more pronounced for the subgroup of houses with a waterfront view.

Exercise 8: (5 mts)

Using the `lmplot()` function in `seaborn`, create a regression plot of the logarithm of `price` vs. the logarithm of `sqft_living`, interacting on the number of bedrooms. What patterns do you observe?

Answer. One possible solution is shown below:

In [63]:
```
sns.lmplot(x = 'log_sqft_living', y= 'log_price',data = houses, hue="be
drooms", height=8)
plt.title("Log-price vs. Log_Sqft_living");
```



There is not a clear monotonic pattern in regression line slope as the number of bedroooms increases, so we will say that the number of bedrooms does *not* interact with the relationship between price and square footage.

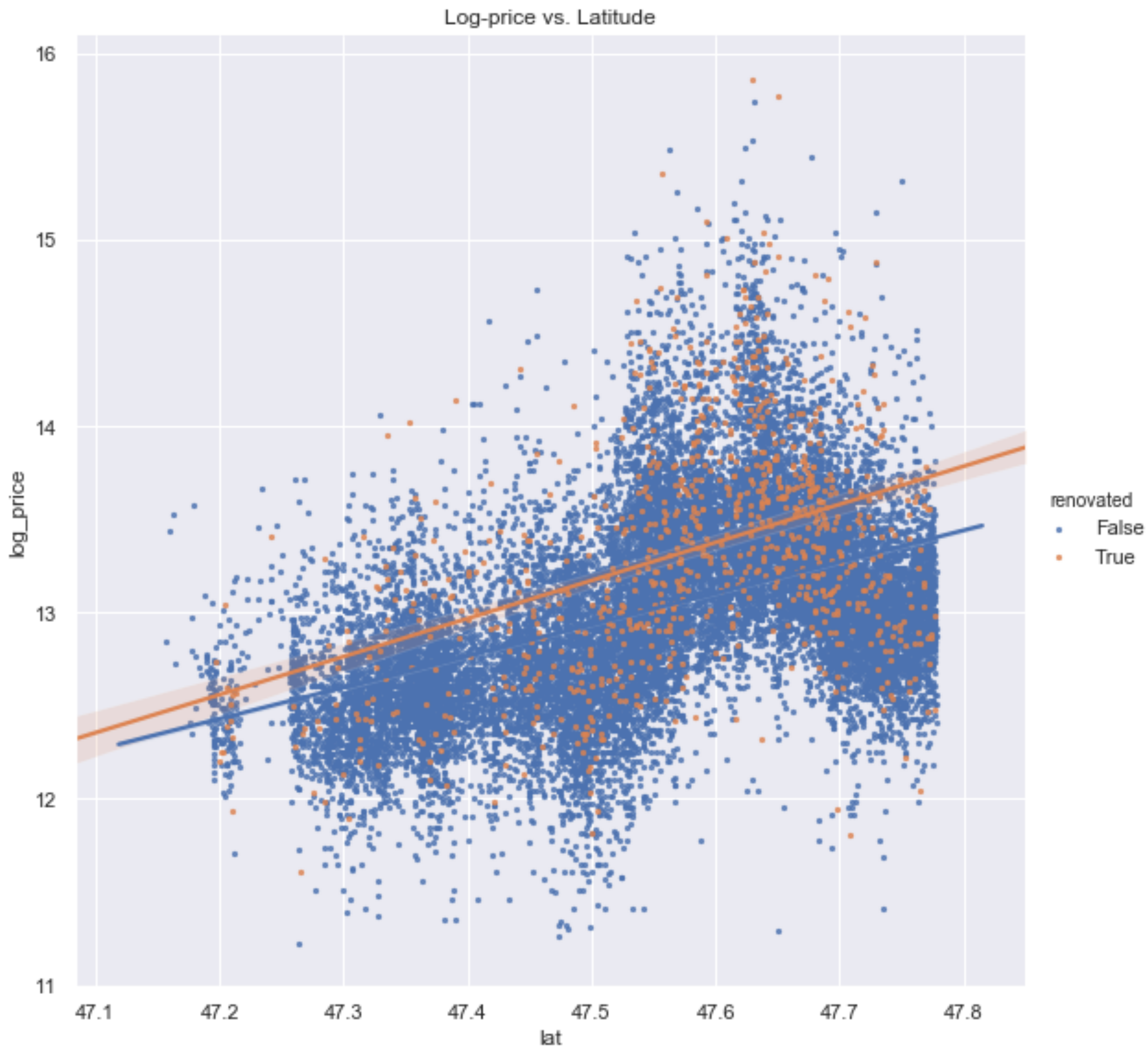
Exercise 9: (10 mts)

Create a regression plot to see if a house's renovation status interacts with the relationship between the logarithm of `price` and the logarithm of `sqft_living`. What can you conclude?

Answer. One possible solution is shown below:

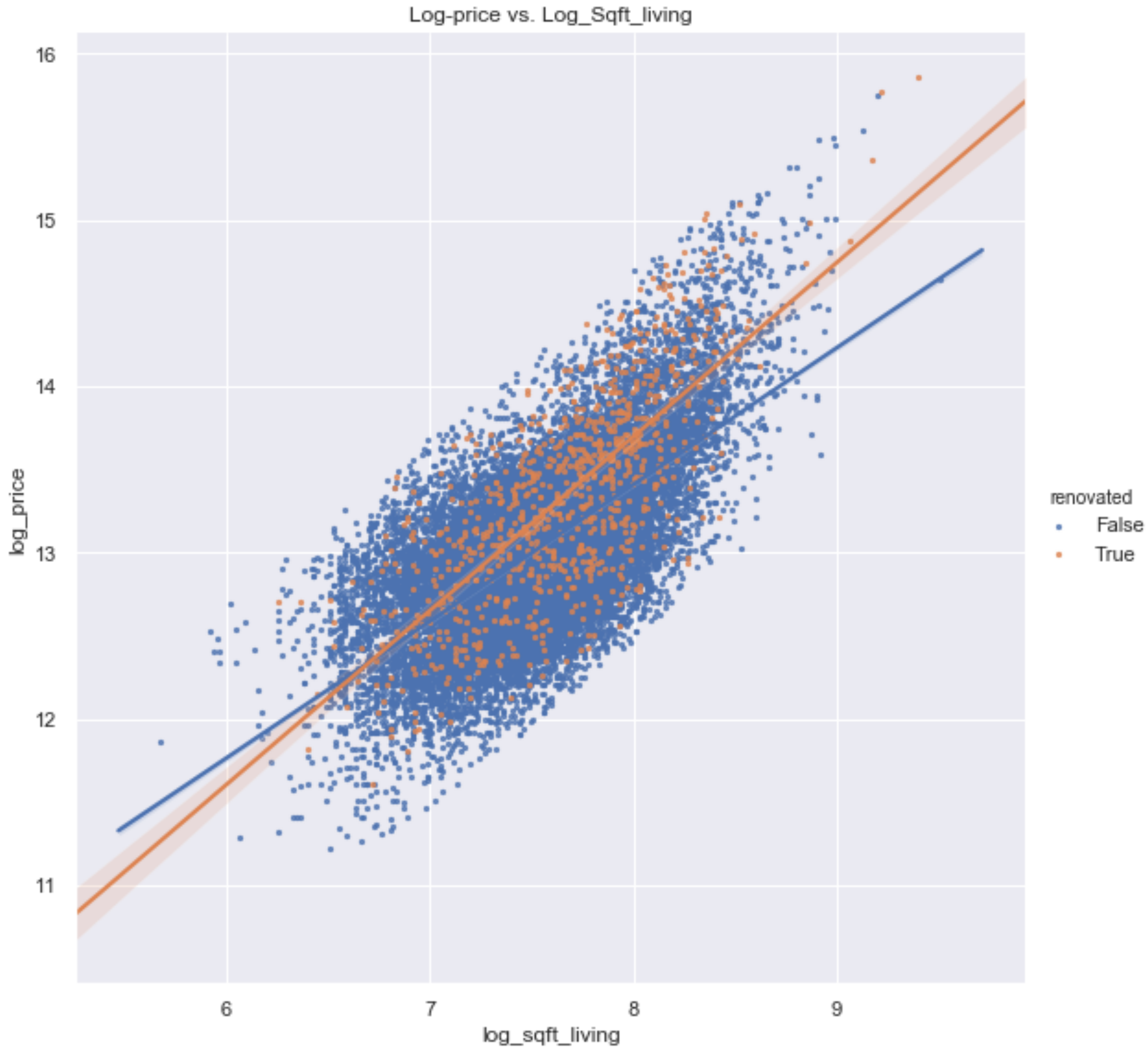
In [64]:
```
houses['renovated'] = houses['yr_renovated'] >0
```

In [65]:
```
sns.lmplot(x = 'lat', y= 'log_price',data = houses, hue="renovated", height=8, scatter_kws={"s": 5});
plt.title("Log-price vs. Latitude");
```



In [66]:
```
sns.lmplot(x = 'log_sqft_living', y= 'log_price',data = houses,
              hue="renovated", height=8, scatter_kws={"s": 5})
plt.title("Log-price vs. Log_Sqft_living")
```

Out[66]: Text(0.5, 1, 'Log-price vs. Log_Sqft_living')



We can see that the relationship between the predictor and outcome variables is more pronounced for houses that have been renovated.

We verify this with the following statistical models:

Esto es nuevo

In [67]:
```
formula = ('np.log(price) ~ lat*C(waterfront)')
mod5_1 = smf.ols(formula=formula, data=houses).fit()
print(mod5_1.summary())
```

```
                            OLS Regression Results

=======================================================================
=======
Dep. Variable:          np.log(price)   R-squared:
  0.236
Model:                            OLS   Adj. R-squared:
  0.236
Method:                 Least Squares   F-statistic:
  2228.
Date:                Thu, 14 Nov 2019   Prob (F-statistic):
   0.00
Time:                        01:50:02   Log-Likelihood:
-13898.
No. Observations:               21613   AIC:                        2.
780e+04
Df Residuals:                   21609   BIC:                        2.
784e+04
Df Model:                           3

Covariance Type:            nonrobust

=======================================================================
===================
                           coef    std err          t      P>|t|
[0.025      0.975]
-----------------------------------------------------------------------
-------------------
Intercept              -68.0880      1.078    -63.180      0.000
-70.200     -65.976
C(waterfront)[T.1]    -102.3040     14.909     -6.862      0.000     -
131.526     -73.082
lat                      1.7058      0.023     75.280      0.000
  1.661       1.750
lat:C(waterfront)[T.1]   2.1753      0.314      6.936      0.000
  1.561       2.790
```

```
==============================================================================
=======
Omnibus:                     1283.881   Durbin-Watson:
  1.957
Prob(Omnibus):                  0.000   Jarque-Bera (JB):               1
927.815
Skew:                           0.510   Prob(JB):
   0.00
Kurtosis:                       4.049   Cond. No.
2.27e+05
==============================================================================
=======

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is
correctly specified.
[2] The condition number is large, 2.27e+05. This might indicate that t
here are
strong multicollinearity or other numerical problems.
```

In [68]:

```python
formula = ('np.log(price) ~ np.log(sqft_living)*C(waterfront)')
mod5_2 = smf.ols(formula=formula, data=houses).fit()
print(mod5_1.summary())
```

```
                            OLS Regression Results

==============================================================================
=======
Dep. Variable:          np.log(price)   R-squared:
  0.236
Model:                            OLS   Adj. R-squared:
  0.236
Method:                 Least Squares   F-statistic:
  2228.
Date:                Thu, 14 Nov 2019   Prob (F-statistic):
   0.00
Time:                        01:50:02   Log-Likelihood:
-13898.
No. Observations:               21613   AIC:                            2.
```

```
780e+04
Df Residuals:                   21609   BIC:                             2.
784e+04
Df Model:                           3

Covariance Type:            nonrobust

=========================================================================
===================
                             coef    std err          t      P>|t|
[0.025      0.975]
-------------------------------------------------------------------------
-------------------
Intercept                -68.0880      1.078    -63.180      0.000
-70.200     -65.976
C(waterfront)[T.1]      -102.3040     14.909     -6.862      0.000     -
131.526     -73.082
lat                        1.7058      0.023     75.280      0.000
  1.661       1.750
lat:C(waterfront)[T.1]     2.1753      0.314      6.936      0.000
  1.561       2.790
=========================================================================
=======
Omnibus:                     1283.881   Durbin-Watson:
  1.957
Prob(Omnibus):                  0.000   Jarque-Bera (JB):                1
927.815
Skew:                           0.510   Prob(JB):
   0.00
Kurtosis:                       4.049   Cond. No.
2.27e+05
=========================================================================
=======

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is
correctly specified.
[2] The condition number is large, 2.27e+05. This might indicate that t
```

```
here are
strong multicollinearity or other numerical problems.
```

For the latitude model (mod5_1) the difference between the two intercepts is -102.303 and the difference between the two slopes is 2.175. They are both statistically significant, as we can conclude from the very small p-values. For the log of square feet model (mod5_2) the difference between the two intercepts is -1.415 and the difference between the two slopes is 0.272. Also in this case they are both statistically significant, as we can conclude from the very small p-values. We can conclude that there is a significant interaction effect between the effect of geographic position and house size when compared for properties with and without a waterfront view. In particular the price of houses with a waterfront increase faster with size and latitude.

Incorporating interaction effects into a linear model (15 mts)

Of course, the above methodology is very inefficient for two reasons:

1. It can only incorporate one interaction effect at a time.
2. It requires fitting multiple linear regression models, depending on the value(s) of the interacting term.

Let's start with our base model which includes all of the other variables we have discussed before, along with separate fixed effects for `waterfront` and `renovated` (but no interaction):

In [69]:

```
formula = ('np.log(price) ~ np.log(sqft_living)+ np.log(sqft_lot) + bed
rooms + floors + bathrooms '
           '+ waterfront + condition + C(waterfront) + view + grade + y
r_built + lat + I(lat**2) '
           '+ long + C(zipcode)+ C(renovated)')
mod6 = smf.ols(formula=formula, data=houses).fit()
print(mod6.summary())
```

```
                            OLS Regression Results
```

```
==============================================================================
=======
Dep. Variable:           np.log(price)   R-squared:
  0.880
Model:                             OLS   Adj. R-squared:
  0.879
Method:                  Least Squares   F-statistic:
  1894.
Date:                 Thu, 14 Nov 2019   Prob (F-statistic):
   0.00
Time:                         01:50:04   Log-Likelihood:
6064.5
No. Observations:                21613   AIC:                          -1.
196e+04
Df Residuals:                    21529   BIC:                          -1.
129e+04
Df Model:                           83

Covariance Type:             nonrobust

==============================================================================
=================
                          coef    std err          t      P>|t|
[0.025      0.975]
------------------------------------------------------------------------------
-----------------
Intercept           -6000.5172    525.440    -11.420      0.000   -703
0.419   -4970.615
C(waterfront)[T.1]      0.2312      0.008     28.748      0.000
0.215       0.247
C(zipcode)[T.98002]     0.0206      0.017      1.244      0.213      -
0.012       0.053
C(zipcode)[T.98003]    -0.0109      0.015     -0.738      0.460      -
0.040       0.018
C(zipcode)[T.98004]     0.9089      0.028     32.366      0.000
0.854       0.964
C(zipcode)[T.98005]     0.5169      0.030     17.277      0.000
0.458       0.576
C(zipcode)[T.98006]     0.4552      0.026     17.754      0.000
```

```
0.405       0.505
C(zipcode)[T.98007]      0.4437      0.031     14.375      0.000
0.383       0.504
C(zipcode)[T.98008]      0.4550      0.029     15.443      0.000
0.397       0.513
C(zipcode)[T.98010]      0.3080      0.025     12.169      0.000
0.258       0.358
C(zipcode)[T.98011]      0.2647      0.037      7.243      0.000
0.193       0.336
C(zipcode)[T.98014]      0.1971      0.041      4.863      0.000
0.118       0.277
C(zipcode)[T.98019]      0.2179      0.040      5.512      0.000
0.140       0.295
C(zipcode)[T.98022]      0.3415      0.024     13.971      0.000
0.294       0.389
C(zipcode)[T.98023]     -0.0630      0.014     -4.649      0.000      -
0.090      -0.036
C(zipcode)[T.98024]      0.3142      0.037      8.522      0.000
0.242       0.386
C(zipcode)[T.98027]      0.3769      0.026     14.277      0.000
0.325       0.429
C(zipcode)[T.98028]      0.2064      0.036      5.812      0.000
0.137       0.276
C(zipcode)[T.98029]      0.4756      0.030     16.065      0.000
0.418       0.534
C(zipcode)[T.98030]      0.0029      0.017      0.169      0.866      -
0.031       0.036
C(zipcode)[T.98031]     -0.0236      0.019     -1.276      0.202      -
0.060       0.013
C(zipcode)[T.98032]     -0.1286      0.020     -6.320      0.000      -
0.169      -0.089
C(zipcode)[T.98033]      0.5725      0.031     18.593      0.000
0.512       0.633
C(zipcode)[T.98034]      0.3271      0.033     10.024      0.000
0.263       0.391
C(zipcode)[T.98038]      0.1926      0.019     10.044      0.000
0.155       0.230
C(zipcode)[T.98039]      1.0816      0.037     29.207      0.000
1.009       1.154
```

```
C(zipcode)[T.98040]      0.6621      0.026     25.708      0.000
0.612       0.713
C(zipcode)[T.98042]      0.0514      0.016      3.132      0.002
0.019       0.084
C(zipcode)[T.98045]      0.3168      0.036      8.921      0.000
0.247       0.386
C(zipcode)[T.98052]      0.4534      0.031     14.432      0.000
0.392       0.515
C(zipcode)[T.98053]      0.4513      0.034     13.410      0.000
0.385       0.517
C(zipcode)[T.98055]     -0.0055      0.021     -0.259      0.796        -
0.047       0.036
C(zipcode)[T.98056]      0.1439      0.023      6.231      0.000
0.099       0.189
C(zipcode)[T.98058]      0.0378      0.020      1.871      0.061        -
0.002       0.077
C(zipcode)[T.98059]      0.2037      0.023      8.968      0.000
0.159       0.248
C(zipcode)[T.98065]      0.3751      0.033     11.303      0.000
0.310       0.440
C(zipcode)[T.98070]      0.0316      0.025      1.281      0.200        -
0.017       0.080
C(zipcode)[T.98072]      0.3056      0.036      8.412      0.000
0.234       0.377
C(zipcode)[T.98074]      0.3975      0.031     12.976      0.000
0.337       0.458
C(zipcode)[T.98075]      0.4342      0.030     14.467      0.000
0.375       0.493
C(zipcode)[T.98077]      0.2957      0.038      7.828      0.000
0.222       0.370
C(zipcode)[T.98092]      0.0827      0.015      5.615      0.000
0.054       0.112
C(zipcode)[T.98102]      0.7057      0.032     21.929      0.000
0.643       0.769
C(zipcode)[T.98103]      0.5476      0.030     18.404      0.000
0.489       0.606
C(zipcode)[T.98105]      0.6996      0.031     22.780      0.000
0.639       0.760
C(zipcode)[T.98106]      0.0773      0.024      3.215      0.001
```

```
0.030        0.125
C(zipcode)[T.98107]      0.5557      0.031     18.090      0.000
0.495        0.616
C(zipcode)[T.98108]      0.1054      0.026      4.032      0.000
0.054        0.157
C(zipcode)[T.98109]      0.7152      0.032     22.316      0.000
0.652        0.778
C(zipcode)[T.98112]      0.7971      0.029     27.722      0.000
0.741        0.853
C(zipcode)[T.98115]      0.5557      0.030     18.429      0.000
0.497        0.615
C(zipcode)[T.98116]      0.4564      0.026     17.498      0.000
0.405        0.508
C(zipcode)[T.98117]      0.5206      0.030     17.083      0.000
0.461        0.580
C(zipcode)[T.98118]      0.2321      0.024      9.816      0.000
0.186        0.278
C(zipcode)[T.98119]      0.6880      0.030     22.718      0.000
0.629        0.747
C(zipcode)[T.98122]      0.5506      0.028     19.883      0.000
0.496        0.605
C(zipcode)[T.98125]      0.3074      0.032      9.518      0.000
0.244        0.371
C(zipcode)[T.98126]      0.2699      0.024     11.032      0.000
0.222        0.318
C(zipcode)[T.98133]      0.1978      0.033      5.936      0.000
0.132        0.263
C(zipcode)[T.98136]      0.3992      0.025     15.964      0.000
0.350        0.448
C(zipcode)[T.98144]      0.4225      0.026     16.096      0.000
0.371        0.474
C(zipcode)[T.98146]      0.0206      0.023      0.903      0.367        -
0.024        0.065
C(zipcode)[T.98148]     -0.0299      0.029     -1.026      0.305        -
0.087        0.027
C(zipcode)[T.98155]      0.1894      0.035      5.452      0.000
0.121        0.257
C(zipcode)[T.98166]      0.0814      0.021      3.896      0.000
0.040        0.122
```

```
C(zipcode)[T.98168]      -0.1559      0.022     -7.012      0.000      -
0.200      -0.112
C(zipcode)[T.98177]       0.3255      0.035      9.351      0.000
0.257       0.394
C(zipcode)[T.98178]      -0.0605      0.023     -2.631      0.009      -
0.105      -0.015
C(zipcode)[T.98188]      -0.0910      0.023     -3.972      0.000      -
0.136      -0.046
C(zipcode)[T.98198]      -0.0722      0.017     -4.211      0.000      -
0.106      -0.039
C(zipcode)[T.98199]       0.5527      0.029     18.794      0.000
0.495       0.610
C(renovated)[T.True]      0.0579      0.007      8.671      0.000
0.045       0.071
np.log(sqft_living)       0.4225      0.006     66.665      0.000
0.410       0.435
np.log(sqft_lot)          0.0709      0.002     34.107      0.000
0.067       0.075
bedrooms                 -0.0141      0.002     -7.827      0.000      -
0.018      -0.011
floors                    0.0175      0.003      5.388      0.000
0.011       0.024
bathrooms                 0.0349      0.003     11.905      0.000
0.029       0.041
waterfront                0.2312      0.008     28.748      0.000
0.215       0.247
condition                 0.0451      0.002     20.677      0.000
0.041       0.049
view                      0.0637      0.002     32.732      0.000
0.060       0.067
grade                     0.1115      0.002     58.283      0.000
0.108       0.115
yr_built                 -0.0004   7.51e-05     -4.691      0.000      -
0.000      -0.000
lat                     250.2116     22.126     11.308      0.000     20
6.842     293.581
I(lat ** 2)              -2.6267      0.233    -11.283      0.000      -
3.083      -2.170
long                     -0.4128      0.052     -7.911      0.000      -
```

```
0.515      -0.311
==============================================================================
=======
Omnibus:                     1424.478   Durbin-Watson:
  2.001
Prob(Omnibus):                  0.000   Jarque-Bera (JB):             5
984.714
Skew:                          -0.188   Prob(JB):
   0.00
Kurtosis:                       5.550   Cond. No.
4.47e+19
==============================================================================
=======

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is
correctly specified.
[2] The smallest eigenvalue is 9.74e-29. This might indicate that there
are
strong multicollinearity problems or that the design matrix is singula
r.
```

In [70]:

```
## effect of a waterfront view different for houses that were recently
 renovated
formula = ('np.log(price) ~ np.log(sqft_living)*C(renovated) + np.log(s
qft_lot) + bedrooms + floors + bathrooms '
           ' + condition + view + grade + yr_built + lat*C(waterfront)
 + I(lat**2) '
           '+ long + C(zipcode)')
mod7 = smf.ols (formula=formula, data=houses).fit()
print(mod7.summary())
```

I effect

```
                            OLS Regression Results

==============================================================================
=======
Dep. Variable:          np.log(price)   R-squared:
  0.880
Model:                            OLS   Adj. R-squared:
  0.879
```

```
Method:                 Least Squares   F-statistic:

  1855.
Date:                Thu, 14 Nov 2019   Prob (F-statistic):
   0.00
Time:                        01:50:05   Log-Likelihood:
6088.9
No. Observations:               21613   AIC:                            -1.
201e+04
Df Residuals:                   21527   BIC:                            -1.
132e+04
Df Model:                          85

Covariance Type:            nonrobust

========================================================================
=====================================
                                                   coef    std err
t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------
-------------------------------------
Intercept                                    -5894.6846    525.102    -11.2
26      0.000   -6923.924   -4865.445
C(renovated)[T.True]                            -0.1618      0.108     -1.4
92      0.136      -0.374       0.051
C(waterfront)[T.1]                             -40.1983      6.092     -6.5
99      0.000     -52.139     -28.258
C(zipcode)[T.98002]                              0.0202      0.017      1.2
22      0.222      -0.012       0.053
C(zipcode)[T.98003]                             -0.0110      0.015     -0.7
44      0.457      -0.040       0.018
C(zipcode)[T.98004]                              0.9138      0.028     32.5
55      0.000       0.859       0.969
C(zipcode)[T.98005]                              0.5232      0.030     17.4
97      0.000       0.465       0.582
C(zipcode)[T.98006]                              0.4614      0.026     18.0
03      0.000       0.411       0.512
C(zipcode)[T.98007]                              0.4496      0.031     14.5
75      0.000       0.389       0.510
C(zipcode)[T.98008]                              0.4595      0.029     15.6
```

```
07       0.000       0.402       0.517

C(zipcode)[T.98010]                          0.3098       0.025      12.2
52       0.000       0.260       0.359
C(zipcode)[T.98011]                          0.2707       0.037       7.4
14       0.000       0.199       0.342
C(zipcode)[T.98014]                          0.2041       0.041       5.0
40       0.000       0.125       0.283
C(zipcode)[T.98019]                          0.2244       0.040       5.6
80       0.000       0.147       0.302
C(zipcode)[T.98022]                          0.3386       0.024      13.8
65       0.000       0.291       0.387
C(zipcode)[T.98023]                         -0.0618       0.014      -4.5
67       0.000      -0.088      -0.035
C(zipcode)[T.98024]                          0.3204       0.037       8.6
97       0.000       0.248       0.393
C(zipcode)[T.98027]                          0.3825       0.026      14.4
96       0.000       0.331       0.434
C(zipcode)[T.98028]                          0.2113       0.035       5.9
57       0.000       0.142       0.281
C(zipcode)[T.98029]                          0.4816       0.030      16.2
76       0.000       0.424       0.540
C(zipcode)[T.98030]                          0.0047       0.017       0.2
73       0.785      -0.029       0.038
C(zipcode)[T.98031]                         -0.0210       0.019      -1.1
37       0.256      -0.057       0.015
C(zipcode)[T.98032]                         -0.1271       0.020      -6.2
52       0.000      -0.167      -0.087
C(zipcode)[T.98033]                          0.5777       0.031      18.7
78       0.000       0.517       0.638
C(zipcode)[T.98034]                          0.3312       0.033      10.1
59       0.000       0.267       0.395
C(zipcode)[T.98038]                          0.1948       0.019      10.1
68       0.000       0.157       0.232
C(zipcode)[T.98039]                          1.0842       0.037      29.2
89       0.000       1.012       1.157
C(zipcode)[T.98040]                          0.6664       0.026      25.8
87       0.000       0.616       0.717
C(zipcode)[T.98042]                          0.0535       0.016       3.2
62       0.001       0.021       0.086
```

```
C(zipcode)[T.98045]                                 0.3219      0.035      9.0

72      0.000       0.252       0.391
C(zipcode)[T.98052]                                 0.4594      0.031     14.6
31      0.000       0.398       0.521
C(zipcode)[T.98053]                                 0.4579      0.034     13.6
15      0.000       0.392       0.524
C(zipcode)[T.98055]                                -0.0016      0.021     -0.0
77      0.939      -0.043       0.040
C(zipcode)[T.98056]                                 0.1488      0.023      6.4
42      0.000       0.103       0.194
C(zipcode)[T.98058]                                 0.0417      0.020      2.0
66      0.039       0.002       0.081
C(zipcode)[T.98059]                                 0.2086      0.023      9.1
88      0.000       0.164       0.253
C(zipcode)[T.98065]                                 0.3813      0.033     11.4
97      0.000       0.316       0.446
C(zipcode)[T.98070]                                 0.0588      0.025      2.3
58      0.018       0.010       0.108
C(zipcode)[T.98072]                                 0.3115      0.036      8.5
81      0.000       0.240       0.383
C(zipcode)[T.98074]                                 0.4032      0.031     13.1
69      0.000       0.343       0.463
C(zipcode)[T.98075]                                 0.4402      0.030     14.6
76      0.000       0.381       0.499
C(zipcode)[T.98077]                                 0.3025      0.038      8.0
14      0.000       0.229       0.377
C(zipcode)[T.98092]                                 0.0827      0.015      5.6
23      0.000       0.054       0.112
C(zipcode)[T.98102]                                 0.7108      0.032     22.1
04      0.000       0.648       0.774
C(zipcode)[T.98103]                                 0.5530      0.030     18.6
00      0.000       0.495       0.611
C(zipcode)[T.98105]                                 0.7038      0.031     22.9
34      0.000       0.644       0.764
C(zipcode)[T.98106]                                 0.0819      0.024      3.4
06      0.001       0.035       0.129
C(zipcode)[T.98107]                                 0.5610      0.031     18.2
77      0.000       0.501       0.621
C(zipcode)[T.98108]                                 0.1102      0.026      4.2
```

```
20         0.000         0.059         0.161

C(zipcode)[T.98109]                                   0.7209        0.032       22.5
09         0.000         0.658         0.784
C(zipcode)[T.98112]                                   0.8029        0.029       27.9
44         0.000         0.747         0.859
C(zipcode)[T.98115]                                   0.5610        0.030       18.6
19         0.000         0.502         0.620
C(zipcode)[T.98116]                                   0.4617        0.026       17.7
11         0.000         0.411         0.513
C(zipcode)[T.98117]                                   0.5258        0.030       17.2
67         0.000         0.466         0.586
C(zipcode)[T.98118]                                   0.2367        0.024       10.0
20         0.000         0.190         0.283
C(zipcode)[T.98119]                                   0.6936        0.030       22.9
19         0.000         0.634         0.753
C(zipcode)[T.98122]                                   0.5560        0.028       20.0
91         0.000         0.502         0.610
C(zipcode)[T.98125]                                   0.3099        0.032        9.6
05         0.000         0.247         0.373
C(zipcode)[T.98126]                                   0.2744        0.024       11.2
25         0.000         0.227         0.322
C(zipcode)[T.98133]                                   0.2028        0.033        6.0
93         0.000         0.138         0.268
C(zipcode)[T.98136]                                   0.4039        0.025       16.1
63         0.000         0.355         0.453
C(zipcode)[T.98144]                                   0.4270        0.026       16.2
78         0.000         0.376         0.478
C(zipcode)[T.98146]                                   0.0258        0.023        1.1
31         0.258        -0.019         0.071
C(zipcode)[T.98148]                                  -0.0268        0.029       -0.9
21         0.357        -0.084         0.030
C(zipcode)[T.98155]                                   0.1921        0.035        5.5
38         0.000         0.124         0.260
C(zipcode)[T.98166]                                   0.0895        0.021        4.2
85         0.000         0.049         0.131
C(zipcode)[T.98168]                                  -0.1520        0.022       -6.8
37         0.000        -0.196        -0.108
C(zipcode)[T.98177]                                   0.3296        0.035        9.4
77         0.000         0.261         0.398
```

```
C(zipcode)[T.98178]                             -0.0550      0.023     -2.3

97      0.017      -0.100      -0.010
C(zipcode)[T.98188]                             -0.0875      0.023     -3.8
21      0.000      -0.132      -0.043
C(zipcode)[T.98198]                             -0.0646      0.017     -3.7
63      0.000      -0.098      -0.031
C(zipcode)[T.98199]                              0.5579      0.029     18.9
85      0.000       0.500       0.616
np.log(sqft_living)                              0.4213      0.006     66.2
85      0.000       0.409       0.434
np.log(sqft_living):C(renovated)[T.True]         0.0289      0.014      2.0
34      0.042       0.001       0.057
np.log(sqft_lot)                                 0.0708      0.002     34.0
49      0.000       0.067       0.075
bedrooms                                        -0.0142      0.002     -7.8
55      0.000      -0.018      -0.011
floors                                           0.0174      0.003      5.3
45      0.000       0.011       0.024
bathrooms                                        0.0345      0.003     11.7
63      0.000       0.029       0.040
condition                                        0.0452      0.002     20.7
31      0.000       0.041       0.049
view                                             0.0634      0.002     32.6
25      0.000       0.060       0.067
grade                                            0.1114      0.002     58.2
85      0.000       0.108       0.115
yr_built                                        -0.0003   7.51e-05     -4.6
43      0.000      -0.000      -0.000
lat                                            245.7639     22.112     11.1
14      0.000     202.423     289.105
lat:C(waterfront)[T.1]                           0.8552      0.128      6.6
74      0.000       0.604       1.106
I(lat ** 2)                                     -2.5800      0.233    -11.0
89      0.000      -3.036      -2.124
long                                            -0.4142      0.052     -7.9
44      0.000      -0.516      -0.312
========================================================================
=======
Omnibus:                     1409.500   Durbin-Watson:
```

```
  2.001

Prob(Omnibus):                  0.000   Jarque-Bera (JB):            5
902.036
Skew:                          -0.183   Prob(JB):
   0.00
Kurtosis:                       5.534   Cond. No.
1.27e+09
=======================================================================
=======

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is
correctly specified.
[2] The condition number is large, 1.27e+09. This might indicate that t
here are
strong multicollinearity or other numerical problems.
```

In [71]:
```
print(mod7.aic)
```

```
-12005.868261991935
```

We can see that both the effect and renovations have a positive impact on price. The effect of a waterfront view is 46.25 percent on prices of comparable homes, while the effect of renovations is 5.794 percent. So far we have looked at global effects of predictors, irrespective of the levels of the other variables. However we might ask, is the effect of a waterfront view different for houses that were recently renovated? To answer this question we need to add an interaction term.

In [72]:
```
formula = ('np.log(price) ~ np.log(sqft_living)*waterfront + np.log(sqf
t_living)*renovated + np.log(sqft_lot)'
           '+ bedrooms + floors + bathrooms '
           '+ waterfront + condition + view + grade + yr_built + lat +
 I(lat**2) + long + C(zipcode)')
mod8= smf.ols(formula=formula, data=houses).fit()
print(mod8.summary())
```

```
                            OLS Regression Results

==============================================================================
=======
Dep. Variable:          np.log(price)   R-squared:
  0.880
Model:                            OLS   Adj. R-squared:
  0.879
Method:                 Least Squares   F-statistic:
  1851.
Date:                Thu, 14 Nov 2019   Prob (F-statistic):
   0.00
Time:                        01:50:06   Log-Likelihood:
6068.8
No. Observations:               21613   AIC:                              -1.
197e+04
Df Residuals:                   21527   BIC:                              -1.
128e+04
Df Model:                          85

Covariance Type:            nonrobust

==============================================================================
==================================
                                           coef    std err          t
     P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
----------------------------------
Intercept                            -5996.9104    525.362    -11.415
     0.000   -7026.658   -4967.163
renovated[T.True]                       -0.1488      0.109     -1.368
     0.171      -0.362       0.064
C(zipcode)[T.98002]                      0.0200      0.017      1.214
     0.225      -0.012       0.052
C(zipcode)[T.98003]                     -0.0110      0.015     -0.748
     0.454      -0.040       0.018
C(zipcode)[T.98004]                      0.9079      0.028     32.329
     0.000       0.853       0.963
C(zipcode)[T.98005]                      0.5170      0.030     17.283
```

```
          0.000        0.458        0.576
     C(zipcode)[T.98006]                          0.4555        0.026       17.769
          0.000        0.405        0.506
     C(zipcode)[T.98007]                          0.4433        0.031       14.364
          0.000        0.383        0.504
     C(zipcode)[T.98008]                          0.4537        0.029       15.400
          0.000        0.396        0.511
     C(zipcode)[T.98010]                          0.3081        0.025       12.176
          0.000        0.259        0.358
     C(zipcode)[T.98011]                          0.2647        0.037        7.244
          0.000        0.193        0.336
     C(zipcode)[T.98014]                          0.1967        0.041        4.853
          0.000        0.117        0.276
     C(zipcode)[T.98019]                          0.2174        0.040        5.501
          0.000        0.140        0.295
     C(zipcode)[T.98022]                          0.3414        0.024       13.965
          0.000        0.293        0.389
     C(zipcode)[T.98023]                         -0.0629        0.014       -4.645
          0.000       -0.089       -0.036
     C(zipcode)[T.98024]                          0.3138        0.037        8.513
          0.000        0.242        0.386
     C(zipcode)[T.98027]                          0.3767        0.026       14.270
          0.000        0.325        0.428
     C(zipcode)[T.98028]                          0.2060        0.036        5.803
          0.000        0.136        0.276
     C(zipcode)[T.98029]                          0.4753        0.030       16.055
          0.000        0.417        0.533
     C(zipcode)[T.98030]                          0.0026        0.017        0.151
          0.880       -0.031        0.036
     C(zipcode)[T.98031]                         -0.0240        0.019       -1.298
          0.194       -0.060        0.012
     C(zipcode)[T.98032]                         -0.1290        0.020       -6.341
          0.000       -0.169       -0.089
     C(zipcode)[T.98033]                          0.5721        0.031       18.583
          0.000        0.512        0.632
     C(zipcode)[T.98034]                          0.3264        0.033       10.003
          0.000        0.262        0.390
     C(zipcode)[T.98038]                          0.1922        0.019       10.024
          0.000        0.155        0.230
```

```
C(zipcode)[T.98039]                            1.0794      0.037     29.137
     0.000       1.007       1.152
C(zipcode)[T.98040]                            0.6606      0.026     25.646
     0.000       0.610       0.711
C(zipcode)[T.98042]                            0.0513      0.016      3.123
     0.002       0.019       0.083
C(zipcode)[T.98045]                            0.3160      0.036      8.901
     0.000       0.246       0.386
C(zipcode)[T.98052]                            0.4533      0.031     14.428
     0.000       0.392       0.515
C(zipcode)[T.98053]                            0.4511      0.034     13.405
     0.000       0.385       0.517
C(zipcode)[T.98055]                           -0.0058      0.021     -0.271
     0.786      -0.047       0.036
C(zipcode)[T.98056]                            0.1434      0.023      6.209
     0.000       0.098       0.189
C(zipcode)[T.98058]                            0.0376      0.020      1.863
     0.062      -0.002       0.077
C(zipcode)[T.98059]                            0.2036      0.023      8.964
     0.000       0.159       0.248
C(zipcode)[T.98065]                            0.3748      0.033     11.296
     0.000       0.310       0.440
C(zipcode)[T.98070]                            0.0400      0.025      1.606
     0.108      -0.009       0.089
C(zipcode)[T.98072]                            0.3052      0.036      8.403
     0.000       0.234       0.376
C(zipcode)[T.98074]                            0.3974      0.031     12.973
     0.000       0.337       0.457
C(zipcode)[T.98075]                            0.4345      0.030     14.477
     0.000       0.376       0.493
C(zipcode)[T.98077]                            0.2959      0.038      7.835
     0.000       0.222       0.370
C(zipcode)[T.98092]                            0.0826      0.015      5.610
     0.000       0.054       0.111
C(zipcode)[T.98102]                            0.7050      0.032     21.911
     0.000       0.642       0.768
C(zipcode)[T.98103]                            0.5472      0.030     18.393
     0.000       0.489       0.605
C(zipcode)[T.98105]                            0.6987      0.031     22.753
```

```
      0.000        0.639        0.759
C(zipcode)[T.98106]                          0.0770       0.024       3.200
      0.001        0.030        0.124
C(zipcode)[T.98107]                          0.5552       0.031      18.077
      0.000        0.495        0.615
C(zipcode)[T.98108]                          0.1050       0.026       4.019
      0.000        0.054        0.156
C(zipcode)[T.98109]                          0.7151       0.032      22.317
      0.000        0.652        0.778
C(zipcode)[T.98112]                          0.7971       0.029      27.728
      0.000        0.741        0.853
C(zipcode)[T.98115]                          0.5553       0.030      18.418
      0.000        0.496        0.614
C(zipcode)[T.98116]                          0.4565       0.026      17.503
      0.000        0.405        0.508
C(zipcode)[T.98117]                          0.5201       0.030      17.069
      0.000        0.460        0.580
C(zipcode)[T.98118]                          0.2313       0.024       9.787
      0.000        0.185        0.278
C(zipcode)[T.98119]                          0.6878       0.030      22.717
      0.000        0.628        0.747
C(zipcode)[T.98122]                          0.5502       0.028      19.873
      0.000        0.496        0.605
C(zipcode)[T.98125]                          0.3067       0.032       9.499
      0.000        0.243        0.370
C(zipcode)[T.98126]                          0.2695       0.024      11.016
      0.000        0.222        0.317
C(zipcode)[T.98133]                          0.1971       0.033       5.919
      0.000        0.132        0.262
C(zipcode)[T.98136]                          0.3993       0.025      15.971
      0.000        0.350        0.448
C(zipcode)[T.98144]                          0.4215       0.026      16.062
      0.000        0.370        0.473
C(zipcode)[T.98146]                          0.0208       0.023       0.909
      0.363       -0.024        0.066
C(zipcode)[T.98148]                         -0.0301       0.029      -1.033
      0.302       -0.087        0.027
C(zipcode)[T.98155]                          0.1885       0.035       5.429
      0.000        0.120        0.257
```

```
C(zipcode)[T.98166]                             0.0814      0.021      3.896
     0.000       0.040       0.122
C(zipcode)[T.98168]                            -0.1563      0.022     -7.029
     0.000      -0.200      -0.113
C(zipcode)[T.98177]                             0.3248      0.035      9.332
     0.000       0.257       0.393
C(zipcode)[T.98178]                            -0.0602      0.023     -2.618
     0.009      -0.105      -0.015
C(zipcode)[T.98188]                            -0.0912      0.023     -3.979
     0.000      -0.136      -0.046
C(zipcode)[T.98198]                            -0.0718      0.017     -4.189
     0.000      -0.105      -0.038
C(zipcode)[T.98199]                             0.5526      0.029     18.793
     0.000       0.495       0.610
np.log(sqft_living)                             0.4210      0.006     66.165
     0.000       0.409       0.434
np.log(sqft_living):renovated[T.True]           0.0271      0.014      1.904
     0.057      -0.001       0.055
waterfront                                     -0.0103      0.227     -0.045
     0.964      -0.455       0.434
np.log(sqft_living):waterfront                  0.0593      0.028      2.084
     0.037       0.004       0.115
np.log(sqft_lot)                                0.0708      0.002     34.042
     0.000       0.067       0.075
bedrooms                                       -0.0141      0.002     -7.781
     0.000      -0.018      -0.011
floors                                          0.0175      0.003      5.388
     0.000       0.011       0.024
bathrooms                                       0.0345      0.003     11.743
     0.000       0.029       0.040
condition                                       0.0452      0.002     20.717
     0.000       0.041       0.049
view                                            0.0636      0.002     32.694
     0.000       0.060       0.067
grade                                           0.1114      0.002     58.199
     0.000       0.108       0.115
yr_built                                       -0.0003   7.51e-05     -4.608
     0.000      -0.000      -0.000
lat                                           250.0677     22.123     11.304
```

```
     0.000      206.705      293.430
I(lat ** 2)                                  -2.6252       0.233     -11.278
     0.000       -3.081       -2.169
long                                         -0.4112       0.052      -7.879
     0.000       -0.513       -0.309
========================================================================
=======
Omnibus:                       1424.704   Durbin-Watson:
  2.001
Prob(Omnibus):                    0.000   Jarque-Bera (JB):             5
990.854
Skew:                            -0.187   Prob(JB):
   0.00
Kurtosis:                         5.552   Cond. No.
1.27e+09
========================================================================
=======

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is
correctly specified.
[2] The condition number is large, 1.27e+09. This might indicate that t
here are
strong multicollinearity or other numerical problems.
```

In [73]:
```python
print(mod8.aic)
```

```
-11965.54786190483
```

Exercise 10: (15 mts)

Our reference model (mod7) contained the following predictors: `bedrooms`, `floors`, `bathrooms`, `condition`, `view`, `grade`, `yr_built`, `long`, `log(sqft_living) * C(renovated)`, `lat * C(waterfront)`, `I(lat**2)`, `log(sqft_lot)` and `C(zipcode)`. The AIC for this model was -12005.86 and the R2 is 0.8798.

Expand this model above by doing the following:

2. Add a term which accounts for the square of the year the house was built
3. Add an interaction term for the presence of a basement in affecting the relationship between the longitude coordinate and the price of a house

Compare the model fit and AIC with the previous model.

Answer. One possible solution is shown below:

In [74]:
```
formula = ('np.log(price) ~ np.log(sqft_living)*C(renovated) + np.log(s
qft_lot) + bedrooms + floors + bathrooms '
           '+ condition + view + grade + yr_built + lat*C(waterfront) +
 I(lat**2) + long + C(zipcode)'
           '+ I(yr_built**2)')
mod9 = smf.ols(formula=formula, data=houses).fit()
print(mod9.summary())
```

```
                            OLS Regression Results

=======================================================================
=======
Dep. Variable:          np.log(price)   R-squared:
  0.884
Model:                            OLS   Adj. R-squared:
  0.883
Method:                 Least Squares   F-statistic:
  1899.
Date:                Thu, 14 Nov 2019   Prob (F-statistic):
   0.00
Time:                        01:50:08   Log-Likelihood:
6428.5
No. Observations:               21613   AIC:                         -1.
268e+04
Df Residuals:                   21526   BIC:                         -1.
199e+04
Df Model:                          86

Covariance Type:            nonrobust
```

```
==========================================================================
======================================
                                                  coef    std err
t      P>|t|      [0.025      0.975]
--------------------------------------------------------------------------
--------------------------------------
Intercept                                   -6012.6468    516.949    -11.6
31      0.000   -7025.905   -4999.389
C(renovated)[T.True]                           -0.3822      0.107     -3.5
69      0.000      -0.592      -0.172
C(waterfront)[T.1]                            -40.7918      5.997     -6.8
02      0.000     -52.547     -29.037
C(zipcode)[T.98002]                             0.0233      0.016      1.4
36      0.151      -0.009       0.055
C(zipcode)[T.98003]                             0.0072      0.015      0.4
98      0.619      -0.021       0.036
C(zipcode)[T.98004]                             0.9283      0.028     33.5
85      0.000       0.874       0.982
C(zipcode)[T.98005]                             0.5507      0.029     18.6
96      0.000       0.493       0.608
C(zipcode)[T.98006]                             0.4802      0.025     19.0
26      0.000       0.431       0.530
C(zipcode)[T.98007]                             0.4816      0.030     15.8
44      0.000       0.422       0.541
C(zipcode)[T.98008]                             0.4917      0.029     16.9
50      0.000       0.435       0.549
C(zipcode)[T.98010]                             0.3011      0.025     12.0
96      0.000       0.252       0.350
C(zipcode)[T.98011]                             0.2840      0.036      7.8
98      0.000       0.213       0.354
C(zipcode)[T.98014]                             0.2023      0.040      5.0
73      0.000       0.124       0.280
C(zipcode)[T.98019]                             0.2224      0.039      5.7
19      0.000       0.146       0.299
C(zipcode)[T.98022]                             0.3422      0.024     14.2
30      0.000       0.295       0.389
C(zipcode)[T.98023]                            -0.0471      0.013     -3.5
31      0.000      -0.073      -0.021
C(zipcode)[T.98024]                             0.3110      0.036      8.5
```

```
74       0.000       0.240       0.382
C(zipcode)[T.98027]                            0.3871      0.026      14.9
03       0.000       0.336       0.438
C(zipcode)[T.98028]                            0.2208      0.035       6.3
22       0.000       0.152       0.289
C(zipcode)[T.98029]                            0.4922      0.029      16.8
97       0.000       0.435       0.549
C(zipcode)[T.98030]                            0.0048      0.017       0.2
88       0.774      -0.028       0.038
C(zipcode)[T.98031]                           -0.0093      0.018      -0.5
13       0.608      -0.045       0.026
C(zipcode)[T.98032]                           -0.1095      0.020      -5.4
66       0.000      -0.149      -0.070
C(zipcode)[T.98033]                            0.5869      0.030      19.3
74       0.000       0.527       0.646
C(zipcode)[T.98034]                            0.3515      0.032      10.9
50       0.000       0.289       0.414
C(zipcode)[T.98038]                            0.1913      0.019      10.1
41       0.000       0.154       0.228
C(zipcode)[T.98039]                            1.0983      0.036      30.1
36       0.000       1.027       1.170
C(zipcode)[T.98040]                            0.6911      0.025      27.2
51       0.000       0.641       0.741
C(zipcode)[T.98042]                            0.0579      0.016       3.5
81       0.000       0.026       0.090
C(zipcode)[T.98045]                            0.3316      0.035       9.4
91       0.000       0.263       0.400
C(zipcode)[T.98052]                            0.4753      0.031      15.3
73       0.000       0.415       0.536
C(zipcode)[T.98053]                            0.4433      0.033      13.3
88       0.000       0.378       0.508
C(zipcode)[T.98055]                            0.0032      0.021       0.1
55       0.877      -0.038       0.044
C(zipcode)[T.98056]                            0.1495      0.023       6.5
77       0.000       0.105       0.194
C(zipcode)[T.98058]                            0.0579      0.020       2.9
10       0.004       0.019       0.097
C(zipcode)[T.98059]                            0.2036      0.022       9.1
07       0.000       0.160       0.247
```

```
C(zipcode)[T.98065]                                0.3734      0.033     11.4
35       0.000       0.309       0.437
C(zipcode)[T.98070]                                0.0264      0.025      1.0
74       0.283      -0.022       0.075
C(zipcode)[T.98072]                                0.3251      0.036      9.0
97       0.000       0.255       0.395
C(zipcode)[T.98074]                                0.4183      0.030     13.8
75       0.000       0.359       0.477
C(zipcode)[T.98075]                                0.4458      0.030     15.0
98       0.000       0.388       0.504
C(zipcode)[T.98077]                                0.3151      0.037      8.4
79       0.000       0.242       0.388
C(zipcode)[T.98092]                                0.0852      0.014      5.8
79       0.000       0.057       0.114
C(zipcode)[T.98102]                                0.6984      0.032     22.0
59       0.000       0.636       0.760
C(zipcode)[T.98103]                                0.5360      0.029     18.3
08       0.000       0.479       0.593
C(zipcode)[T.98105]                                0.7033      0.030     23.2
81       0.000       0.644       0.763
C(zipcode)[T.98106]                                0.0714      0.024      3.0
16       0.003       0.025       0.118
C(zipcode)[T.98107]                                0.5413      0.030     17.9
09       0.000       0.482       0.601
C(zipcode)[T.98108]                                0.0959      0.026      3.7
29       0.000       0.046       0.146
C(zipcode)[T.98109]                                0.7021      0.032     22.2
65       0.000       0.640       0.764
C(zipcode)[T.98112]                                0.7957      0.028     28.1
28       0.000       0.740       0.851
C(zipcode)[T.98115]                                0.5674      0.030     19.1
29       0.000       0.509       0.626
C(zipcode)[T.98116]                                0.4499      0.026     17.5
29       0.000       0.400       0.500
C(zipcode)[T.98117]                                0.5188      0.030     17.3
05       0.000       0.460       0.578
C(zipcode)[T.98118]                                0.2249      0.023      9.6
67       0.000       0.179       0.270
C(zipcode)[T.98119]                                0.6695      0.030     22.4
```

```
64       0.000       0.611       0.728
C(zipcode)[T.98122]                           0.5280       0.027      19.3
68       0.000       0.475       0.581
C(zipcode)[T.98125]                           0.3282       0.032      10.3
30       0.000       0.266       0.390
C(zipcode)[T.98126]                           0.2645       0.024      10.9
87       0.000       0.217       0.312
C(zipcode)[T.98133]                           0.2129       0.033       6.4
97       0.000       0.149       0.277
C(zipcode)[T.98136]                           0.3930       0.025      15.9
73       0.000       0.345       0.441
C(zipcode)[T.98144]                           0.4047       0.026      15.6
64       0.000       0.354       0.455
C(zipcode)[T.98146]                           0.0290       0.022       1.2
88       0.198      -0.015       0.073
C(zipcode)[T.98148]                          -0.0134       0.029      -0.4
67       0.640      -0.070       0.043
C(zipcode)[T.98155]                           0.2101       0.034       6.1
50       0.000       0.143       0.277
C(zipcode)[T.98166]                           0.0979       0.021       4.7
56       0.000       0.058       0.138
C(zipcode)[T.98168]                          -0.1458       0.022      -6.6
63       0.000      -0.189      -0.103
C(zipcode)[T.98177]                           0.3481       0.034      10.1
62       0.000       0.281       0.415
C(zipcode)[T.98178]                          -0.0416       0.023      -1.8
39       0.066      -0.086       0.003
C(zipcode)[T.98188]                          -0.0750       0.023      -3.3
25       0.001      -0.119      -0.031
C(zipcode)[T.98198]                          -0.0510       0.017      -3.0
20       0.003      -0.084      -0.018
C(zipcode)[T.98199]                           0.5676       0.029      19.6
19       0.000       0.511       0.624
np.log(sqft_living)                           0.4126       0.006      65.8
52       0.000       0.400       0.425
np.log(sqft_living):C(renovated)[T.True]      0.0592       0.014       4.2
29       0.000       0.032       0.087
np.log(sqft_lot)                              0.0852       0.002      40.2
04       0.000       0.081       0.089
```

```
bedrooms                                                -0.0105      0.002     -5.9
16       0.000      -0.014      -0.007
floors                                                  -0.0125      0.003     -3.6
74       0.000      -0.019      -0.006
bathrooms                                                0.0251      0.003      8.6
23       0.000       0.019       0.031
condition                                                0.0540      0.002     24.8
61       0.000       0.050       0.058
view                                                     0.0667      0.002     34.7
70       0.000       0.063       0.070
grade                                                    0.1081      0.002     57.3
06       0.000       0.104       0.112
yr_built                                                -0.1896      0.007    -26.2
59       0.000      -0.204      -0.175
lat                                                    258.2540     21.773     11.8
61       0.000     215.577     300.931
lat:C(waterfront)[T.1]                                   0.8676      0.126      6.8
78       0.000       0.620       1.115
I(lat ** 2)                                             -2.7113      0.229    -11.8
35       0.000      -3.160      -2.262
long                                                    -0.4630      0.051     -9.0
15       0.000      -0.564      -0.362
I(yr_built ** 2)                                      4.833e-05   1.84e-06     26.2
12       0.000    4.47e-05    5.19e-05
========================================================================
=======
Omnibus:                     1552.582   Durbin-Watson:
  2.005
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              6
876.465
Skew:                          -0.214   Prob(JB):
   0.00
Kurtosis:                       5.730   Cond. No.
1.64e+12
========================================================================
=======

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is
```

```
correctly specified.
[2] The condition number is large, 1.64e+12. This might indicate that t
here are
strong multicollinearity or other numerical problems.
```

In [75]:
```
print(mod9.aic)
```

```
-12682.956456731124
```

In [76]:
```
# built dummy variable to separate houses with a basement and houses wi
th no basement
houses['has_basement'] = (houses['sqft_basement'] > 0) * 1.0
```

In [77]:
```
# estimate a model with an interaction between the longitude coordinate

# and the presence of a basement.
formula = ('np.log(price) ~ np.log(sqft_living)*C(renovated) + np.log(s
qft_lot) + bedrooms + floors + bathrooms '
           '+ condition + view + grade + yr_built + lat * C(waterfront)
 + I(lat**2) + long + C(zipcode) '
           '+ has_basement * long')
mod10 = smf.ols(formula=formula, data=houses).fit()
print(mod10.summary())
```

```
                            OLS Regression Results

==============================================================================
=======
Dep. Variable:          np.log(price)   R-squared:
  0.881
Model:                            OLS   Adj. R-squared:
  0.881
Method:                 Least Squares   F-statistic:
  1833.
Date:                Thu, 14 Nov 2019   Prob (F-statistic):
   0.00
Time:                        01:50:10   Log-Likelihood:
6199.6
No. Observations:               21613   AIC:                             -1.
```

```
222e+04
Df Residuals:                   21525   BIC:                          -1.
152e+04
Df Model:                          87

Covariance Type:            nonrobust

========================================================================================================
                                               coef    std err
t      P>|t|      [0.025      0.975]
--------------------------------------------------------------------------------------------------------
Intercept                                -6308.3439    523.203    -12.0
57       0.000   -7333.860   -5282.827
C(renovated)[T.True]                        -0.2039      0.108     -1.8
88       0.059      -0.416       0.008
C(waterfront)[T.1]                         -40.0337      6.064     -6.6
02       0.000     -51.920     -28.147
C(zipcode)[T.98002]                          0.0163      0.016      0.9
93       0.321      -0.016       0.048
C(zipcode)[T.98003]                         -0.0093      0.015     -0.6
34       0.526      -0.038       0.019
C(zipcode)[T.98004]                          0.9153      0.028     32.7
43       0.000       0.860       0.970
C(zipcode)[T.98005]                          0.5272      0.030     17.6
99       0.000       0.469       0.586
C(zipcode)[T.98006]                          0.4651      0.026     18.2
12       0.000       0.415       0.515
C(zipcode)[T.98007]                          0.4502      0.031     14.6
55       0.000       0.390       0.510
C(zipcode)[T.98008]                          0.4626      0.029     15.7
66       0.000       0.405       0.520
C(zipcode)[T.98010]                          0.3097      0.025     12.3
12       0.000       0.260       0.359
C(zipcode)[T.98011]                          0.2827      0.036      7.7
72       0.000       0.211       0.354
C(zipcode)[T.98014]                          0.2061      0.040      5.1
15       0.000       0.127       0.285
```

```
C(zipcode)[T.98019]                               0.2316      0.039       5.8
87       0.000       0.154       0.309
C(zipcode)[T.98022]                               0.3434      0.024      14.1
29       0.000       0.296       0.391
C(zipcode)[T.98023]                              -0.0590      0.013      -4.3
72       0.000      -0.085      -0.033
C(zipcode)[T.98024]                               0.3188      0.037       8.6
96       0.000       0.247       0.391
C(zipcode)[T.98027]                               0.3938      0.026      14.9
55       0.000       0.342       0.445
C(zipcode)[T.98028]                               0.2237      0.035       6.3
33       0.000       0.155       0.293
C(zipcode)[T.98029]                               0.4780      0.029      16.2
30       0.000       0.420       0.536
C(zipcode)[T.98030]                               0.0011      0.017       0.0
65       0.948      -0.032       0.034
C(zipcode)[T.98031]                              -0.0231      0.018      -1.2
55       0.210      -0.059       0.013
C(zipcode)[T.98032]                              -0.1237      0.020      -6.1
12       0.000      -0.163      -0.084
C(zipcode)[T.98033]                               0.5824      0.031      19.0
07       0.000       0.522       0.642
C(zipcode)[T.98034]                               0.3417      0.032      10.5
22       0.000       0.278       0.405
C(zipcode)[T.98038]                               0.1871      0.019       9.8
09       0.000       0.150       0.224
C(zipcode)[T.98039]                               1.0789      0.037      29.2
78       0.000       1.007       1.151
C(zipcode)[T.98040]                               0.6677      0.026      26.0
53       0.000       0.617       0.718
C(zipcode)[T.98042]                               0.0484      0.016       2.9
61       0.003       0.016       0.080
C(zipcode)[T.98045]                               0.3175      0.035       8.9
91       0.000       0.248       0.387
C(zipcode)[T.98052]                               0.4658      0.031      14.8
90       0.000       0.405       0.527
C(zipcode)[T.98053]                               0.4493      0.033      13.4
18       0.000       0.384       0.515
C(zipcode)[T.98055]                              -0.0038      0.021      -0.1
```

```
78        0.859      -0.045       0.038
C(zipcode)[T.98056]                           0.1430       0.023        6.2
22        0.000       0.098       0.188
C(zipcode)[T.98058]                           0.0396       0.020        1.9
71        0.049       0.000       0.079
C(zipcode)[T.98059]                           0.1981       0.023        8.7
60        0.000       0.154       0.242
C(zipcode)[T.98065]                           0.3699       0.033       11.2
05        0.000       0.305       0.435
C(zipcode)[T.98070]                           0.0581       0.025        2.3
42        0.019       0.009       0.107
C(zipcode)[T.98072]                           0.3258       0.036        9.0
09        0.000       0.255       0.397
C(zipcode)[T.98074]                           0.4026       0.030       13.2
06        0.000       0.343       0.462
C(zipcode)[T.98075]                           0.4321       0.030       14.4
70        0.000       0.374       0.491
C(zipcode)[T.98077]                           0.3103       0.038        8.2
56        0.000       0.237       0.384
C(zipcode)[T.98092]                           0.0815       0.015        5.5
68        0.000       0.053       0.110
C(zipcode)[T.98102]                           0.7378       0.032       23.0
24        0.000       0.675       0.801
C(zipcode)[T.98103]                           0.5717       0.030       19.3
07        0.000       0.514       0.630
C(zipcode)[T.98105]                           0.7255       0.031       23.7
33        0.000       0.666       0.785
C(zipcode)[T.98106]                           0.0914       0.024        3.8
20        0.000       0.045       0.138
C(zipcode)[T.98107]                           0.5814       0.031       19.0
20        0.000       0.522       0.641
C(zipcode)[T.98108]                           0.1200       0.026        4.6
16        0.000       0.069       0.171
C(zipcode)[T.98109]                           0.7400       0.032       23.2
05        0.000       0.678       0.803
C(zipcode)[T.98112]                           0.8242       0.029       28.7
94        0.000       0.768       0.880
C(zipcode)[T.98115]                           0.5812       0.030       19.3
67        0.000       0.522       0.640
```

```
C(zipcode)[T.98116]                         0.4745      0.026     18.2
69      0.000      0.424      0.525
C(zipcode)[T.98117]                         0.5435      0.030     17.9
23      0.000      0.484      0.603
C(zipcode)[T.98118]                         0.2461      0.024     10.4
64      0.000      0.200      0.292
C(zipcode)[T.98119]                         0.7157      0.030     23.7
34      0.000      0.657      0.775
C(zipcode)[T.98122]                         0.5749      0.028     20.8
55      0.000      0.521      0.629
C(zipcode)[T.98125]                         0.3247      0.032     10.1
07      0.000      0.262      0.388
C(zipcode)[T.98126]                         0.2831      0.024     11.6
31      0.000      0.235      0.331
C(zipcode)[T.98133]                         0.2180      0.033      6.5
78      0.000      0.153      0.283
C(zipcode)[T.98136]                         0.4158      0.025     16.6
99      0.000      0.367      0.465
C(zipcode)[T.98144]                         0.4443      0.026     17.0
07      0.000      0.393      0.495
C(zipcode)[T.98146]                         0.0253      0.023      1.1
12      0.266     -0.019      0.070
C(zipcode)[T.98148]                        -0.0326      0.029     -1.1
24      0.261     -0.089      0.024
C(zipcode)[T.98155]                         0.2059      0.035      5.9
60      0.000      0.138      0.274
C(zipcode)[T.98166]                         0.0887      0.021      4.2
66      0.000      0.048      0.129
C(zipcode)[T.98168]                        -0.1487      0.022     -6.7
27      0.000     -0.192     -0.105
C(zipcode)[T.98177]                         0.3422      0.035      9.8
84      0.000      0.274      0.410
C(zipcode)[T.98178]                        -0.0515      0.023     -2.2
51      0.024     -0.096     -0.007
C(zipcode)[T.98188]                        -0.0885      0.023     -3.8
84      0.000     -0.133     -0.044
C(zipcode)[T.98198]                        -0.0661      0.017     -3.8
73      0.000     -0.100     -0.033
C(zipcode)[T.98199]                         0.5753      0.029     19.6
```

```
39        0.000        0.518        0.633
np.log(sqft_living)                                 0.4518        0.007       67.2
61        0.000        0.439        0.465
np.log(sqft_living):C(renovated)[T.True]            0.0345        0.014        2.4
40        0.015        0.007        0.062
np.log(sqft_lot)                                    0.0669        0.002       31.8
95        0.000        0.063        0.071
bedrooms                                           -0.0156        0.002       -8.6
48        0.000       -0.019       -0.012
floors                                             -0.0062        0.004       -1.7
09        0.087       -0.013        0.001
bathrooms                                           0.0403        0.003       13.6
99        0.000        0.035        0.046
condition                                           0.0462        0.002       21.2
86        0.000        0.042        0.050
view                                                0.0657        0.002       33.8
54        0.000        0.062        0.069
grade                                               0.1069        0.002       55.4
79        0.000        0.103        0.111
yr_built                                           -0.0003     7.49e-05       -3.4
39        0.001       -0.000       -0.000
lat                                               263.2078       22.032       11.9
46        0.000      220.023      306.393
lat:C(waterfront)[T.1]                              0.8518        0.128        6.6
78        0.000        0.602        1.102
I(lat ** 2)                                        -2.7637        0.232      -11.9
22        0.000       -3.218       -2.309
long                                               -0.4075        0.052       -7.8
38        0.000       -0.509       -0.306
has_basement                                       -7.2390        2.610       -2.7
74        0.006      -12.354       -2.124
has_basement:long                                  -0.0588        0.021       -2.7
55        0.006       -0.101       -0.017
========================================================================
=======
Omnibus:                       1458.357   Durbin-Watson:
  2.000
Prob(Omnibus):                    0.000   Jarque-Bera (JB):              6
307.671
```

```
Skew:                           -0.186   Prob(JB):
   0.00
Kurtosis:                        5.620   Cond. No.
1.27e+09
==============================================================================
=======

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is
correctly specified.
[2] The condition number is large, 1.27e+09. This might indicate that t
here are
strong multicollinearity or other numerical problems.
```

The effects are significant in both cases. Both models improve the fit of our reference models, but the addition of a square term in the building year has a stronger impact compared to the interaction between basement and longitude.

In [78]:

```
# the r-squared results
r7 = mod7.rsquared
r9 = mod9.rsquared
r10 = mod10.rsquared

# the aic results
aic7 = mod7.aic
aic9 = mod9.aic
aic10 = mod10.aic

print("-------------- R Squared results --------------")
print("Model 7 -", r7)
print("Model 9 -", r9)
print("Model 10 -", r10)
print("\n----------------- AIC results -----------------")
print("AIC 7 -", aic7)
print("AIC 9 -", aic9)
print("AIC 10 -", aic10)
```

```
-------------- R Squared results --------------
Model 7 - 0.8798452507027756
Model 9 - 0.8835618615444407
Model 10 - 0.8810696008728642

----------------- AIC results -----------------
AIC 7 - -12005.868261991935
AIC 9 - -12682.956456731124
AIC 10 - -12223.229661644618
```

Conclusions (5 mts)

In this case, we applied various types of transformations to the predictor and response variables to improve the quality of our linear modeling. In particular, we found that fitting the logarithm of house prices allowed us to get better results. Using our understanding of transformations, we were able to effectively model nonlinear relationships, such as the quadratic relationship between latitude and the log of price. Finally, we tied in our understanding of interaction effects from previous EDA cases in order to directly model and quantify the interaction of renovation and waterfront status on square footage.

Takeaways (5 mts)

Variable transformations are a powerful technique to improve the quality of our linear models. In particular:

1. Transforming the dependent variable can improve linearity and resolve the problem of uneven variance around the line of best fit.
2. Transforming the independent variables can be useful to improve the quality of the fit, capture nonlinear relationships between the independent and response variables, and test a wider range of hypotheses.
3. Interaction terms are a specific type of variable transformation, involving the product of two other independent variables. They can capture dependencies in the relationship between a predictor variable and the response variable on the value of a third variable.