Greater Seattle Area Housing: Sales Price Prediction

Francis Tan

2017-02-27

Summary

The goal of this project is to predict the sale price of a property by employing various predictive machine learning models in an ensemble given housing data such as the number of bedrooms/bathrooms, square footage, year built as well as other less intuitive variables as provided by the Zillow API.

Training Data

The most important element of any data science project is the data itself. This project heavily utilizes data from Zillow, a real estate destination for the internet generation. Fortunately, Zillow provides a public API which provides a convenience to an otherwise tedious task. Below are some basic information of the data.

```
>>> df.head()
       zpid             street      city state    zip FIPScounty  \
0  38447172     18314 48th Ave W  Lynnwood    WA  98037      53061
1  38448108     19011 Grannis Rd   Bothell    WA  98012      53061
2  38448131     2625 189th St SE   Bothell    WA  98012      53061
3  38449213  719 John Bailey Rd   Bothell    WA  98012      53061
4  38452743     5113 212th St SW  Lynnwood    WA  98036      53061

        useCode taxAssessmentYear taxAssessment yearBuilt    ...
bedrooms  \
0  SingleFamily              2015        222100      1967    ...
3
1  SingleFamily              2015        233400      1969    ...
3
2  SingleFamily              2015        486300      1999    ...
```

```
2
3  SingleFamily              2015        238800      1957    ...
3
4  SingleFamily              2015        294300      1960    ...
3

   lastSoldDate lastSoldPrice zestimate zestimateLastUpdated  \
0    11/07/2016        315000    326746           12/30/2016
1    10/06/2016        353000    368478           12/30/2016
2    02/01/2016        405000    673774           12/30/2016
3    07/22/2016        360000    369992           12/30/2016
4    05/12/2016        430000    460211           12/30/2016

   zestimateValueChange zestimateValueLow zestimateValueHigh  \
0                 -5752            310409             343083
1                 -2945            350054             386902
2                  -360            640085             707463
3                  1275            351492             388492
4                  1540            437200             483222

  zestimatePercentile    region
0                   0  Lynnwood
1                   0   Bothell
2                   0   Bothell
3                   0   Bothell
4                   0  Lynnwood

[5 rows x 23 columns]
```

Printing the *shape* attribute shows that we have 2826 observations and 23 columns.

```
>>> df.shape
(2826, 23)
```

Finally, printing the *columns* attribute produces a list of all column names.

```
>>> df.columns
Index([u'zpid', u'street', u'city', u'state', u'zip', u'FIPScounty',
       u'useCode', u'taxAssessmentYear', u'taxAssessment',
u'yearBuilt',
       u'lotSizeSqFt', u'finishedSqFt', u'bathrooms', u'bedrooms',
       u'lastSoldDate', u'lastSoldPrice', u'zestimate',
       u'zestimateLastUpdated', u'zestimateValueChange',
u'zestimateValueLow',
       u'zestimateValueHigh', u'zestimatePercentile', u'region'],
      dtype='object')
```

Since the goal of this project is to predict the sale price, it is obvious that the *lastSoldPrice* should be the response variable while the other columns can act as feature variables. Of course, some processing such as dummy variable conversion is required before training begins.

Data Collection Process

Although the availability of a public API has made the data collection process simple, there are some limitations that we had to be cognizant of. Our vision was to start with a "seed" property which in turn would collect "comps" or comparables. Comps are simply other properties that have similar features to our seed property. This will provide a buyer an idea of what the value of the property should be.

The first limitation is that the full set of information that we were looking for cannot be extracted from one API endpoint. Zillow does not provide an endpoint which returns property information of comps given a seed property. What it provides instead is one endpoint that returns a list of comp property IDs (Zillow Property ID or ZPID) given a seed property address and a separate endpoint that returns property information given a ZPID. Furthermore, the comp endpoint returns a maximum of 25 comps per seed property. Thus the collection process is divided into three steps:

1. Collect comp IDs given a property address using *GetDeepSearchResults*.
2. Loop through each ZPID, collect 25 more comps for each, and append results to list of the other ZPIDs.
3. Collect property information for each ZPID collected using *GetDeepComps*.

The second limitation is that Zillow has limited the number of calls allowed per day to 1000. This poses a problem if one's intent was to collect a significant amount of data. This limits our collection process further since we had to resort to making two calls. A simple solution was to include a sleep timer of 24 hours when a call encounters a rate limit warning. Although somewhat inconvenient, the solution achieved what we needed to accomplish.

Data Processing

The next step is to process or clean the data. We can immediately see that we need to convert many of these factor variables into dummy variables. This is easily achieved in Pandas using the *get_dummies()* function.