Greater Seattle Area Housing–Sales Price Prediction

Francis Tan

2017-02-27

Abstract

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.

	street	city	state	zip	lastSoldPrice
0	18314 48th Ave W	Lynnwood	WA	98037	315000
1	19011 Grannis Rd	Bothell	WA	98012	353000
2	2625 189th St SE	Bothell	WA	98012	405000
3	719 John Bailey Rd	Bothell	WA	98012	360000
4	5113 212th St SW	Lynnwood	WA	98036	430000

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

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

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

```
>>> training_raw.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.

```
>>> training = training_raw
>>> training['zip'] = training['zip'].astype('category')
>>> training['FIPScounty'] = training['FIPScounty'].astype('category')
>>> training['useCode'] = training['useCode'].astype('category')
>>> training['region'] = training['region'].astype('category')
>>> dummies = pd.get_dummies(training, drop_first=True)
>>> dummies.head()
       zpid  finishedSqFt  lastSoldPrice  zestimatePercentile  \
0  38447172          1223         315000                    0
1  38448108          1296         353000                    0
2  38448131          3226         405000                    0
3  38449213          1200         360000                    0
4  38452743          2300         430000                    0

   street_1000 Aberdeen Ave NE  street_10003 181st Ave NE  \
0                            0                          0
1                            0                          0
2                            0                          0
3                            0                          0
4                            0                          0

   street_10005 NE 204th St  street_10007 NE 12th St APT 106  \
0                         0                                0
1                         0                                0
2                         0                                0
3                         0                                0
4                         0                                0

   street_10010 NE 27th St  street_10015 NE 124th Pl # 304  \
0                        0                               0
1                        0                               0
2                        0                               0
3                        0                               0
4                        0                               0

          ...           region_West Queen Anne  region_West Wellington
\
0         ...                                0                       0
1         ...                                0                       0
2         ...                                0                       0
```

```
3         ...                            0                    0
4         ...                            0                    0

   region_West Woodland  region_Whittier Heights  region_Wilburton  \
0                     0                        0                 0
1                     0                        0                 0
2                     0                        0                 0
3                     0                        0                 0
4                     0                        0                 0

   region_Windermere  region_WoodRidge  region_Woodinville  \
0                  0                 0                   0
1                  0                 0                   0
2                  0                 0                   0
3                  0                 0                   0
4                  0                 0                   0

   region_Woodinville Heights  region_Yarrow Point
0                           0                    0
1                           0                    0
2                           0                    0
3                           0                    0
4                           0                    0

[5 rows x 17303 columns]
```