Recuperating the expected market price when most prices are massively overstated

Heinrich Matzinger, *

November 7, 2018

Abstract

We consider a model where prices in a Market are given only online, but a large percentage of the prices do not correspond to market prices but are strongly overstated. A small percentage of the prices available online correspond to the correct market prices. The task is having only access to the online prices being able to find the market prices. (Unsupervised learning problem). For this one is not given the information about which prices are market value and which are not. Typically we look at a regression model for the market prices with a relatively large feature space and only few instances. We do not assume any knowledge about the non-market values which are posted except that they are overstated and not understated. We show two algorithms capable of finding with high precision the generalized linear model for the market price when given the data containing only a small percentage of market prices and otherwise corrupted with overstated prices. We think of non-US housing markets where the effective sales prices are not available online. Often, the only prices available to the public are prices contained in online advertisements. A small percentage of these prices may correspond to the approximate market prices and many prices may be overstated. There will typically not be understated prices except for a few outliers, since such object would be sold immediately.

1 The problem

If it wouldn't be for the regression, the problem would be easy. So, if online you have the same object offered at very different prices, then you could chose the smallest price. That price should be the market price. (Provided that this price is really available.) If the objects to be sold with their price advertised online are houses, things maybe very different. Again, we assume that we are not in the US, where true transaction prices are available online. So, if somebody were to create a online cite for price evaluation of a real estate object (Like zillow in the us) for example, the difficulties could be several:

*School of Mathematics, Georgia Institute of Technology, Atlanta, GA 30332.
Email:

1. No (or almost no) real transaction prices are known for sure. The only available information are prices which are displayed in online adds.

2. Among those online prices the larger percentage are massively overstated. But in many case, there can not be large contingents of under-marked-priced houses since they would soon be sold. Also, if transactions take place, (market is not frozen) a certain percentage of the prices visible online must correspond to the true market prices

3. Unlike the US, in many European countries people tend to stay in the same house for 50 years or more. Hence, in a neighborhood there will be only a few houses which are offered.

4. Village to village, region to region may be very different. So, you need to make an estimate on a very few available online prices in the given region of the house, since prices of houses further away are not very relevant

5. The market price may more or less follow a somewhat nice linear (or quadratic model). But the overstated house prices do not follow a "smooth" model...you can not assume a model for the overstated prices other than them being overstated, that is their expected value given the features above market price.

6. Main problem is that you do not know which of the prices are overstated, and it is not simple to see due to the sparsity and large feature space. This is an unsupervised learning problem.

7. In reality, the percentage of house-prices in your collection, which are true market prices is not known to you.

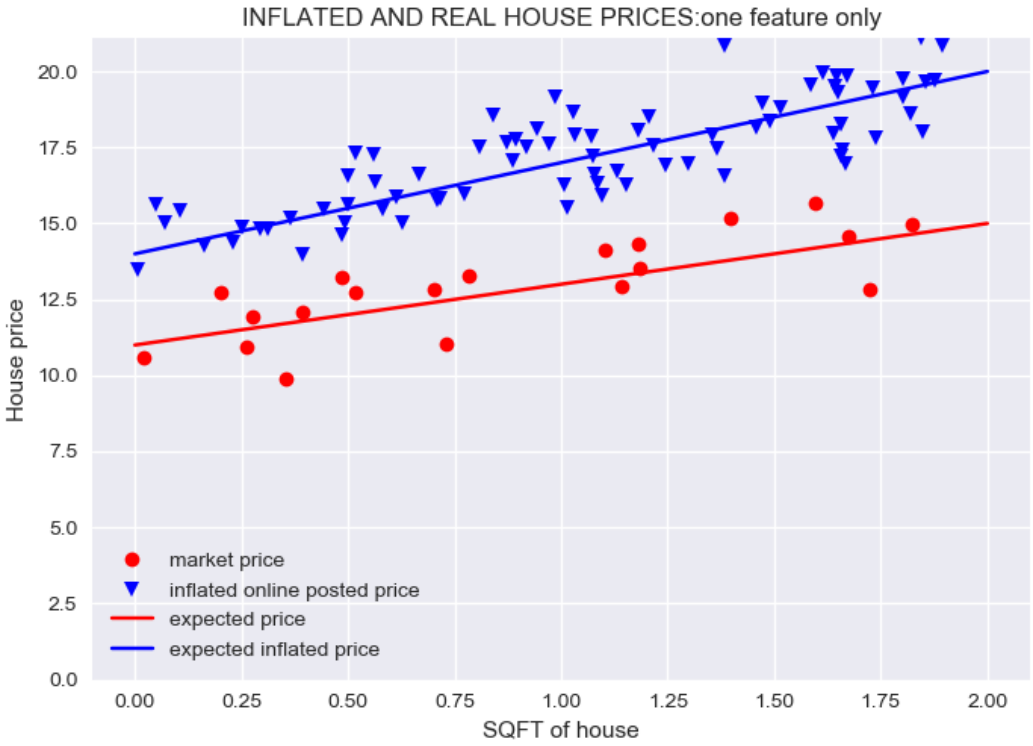
8. The exact model (linear, quadratic...) for the marked price is now known.

So, let us describe what we simulated. We took q to be the percentage of houses which have the online visible price to correspond to the true market price. the total of houses is m. There are $m \times q$ houses where the online price and the market price correspond. The house number i if it is such a house where the market price equals the online visible price satisfies:

$$y_i = b_0 + X_{i1}b_1 + \ldots + X_{ir}b_r + \epsilon_i \tag{1.1}$$

where ϵ_i a random "error" with expectation 0. (The house-specific uncertainty). Also, y_i is the true price of house number i, which is known and the coefficients b_j are the"true model coefficients for the market price". Furthermore X_{i1},...,X_{ir} are the feature of the house i which are also known to us. So, equation 1.1 holds for the houses which have their online price equal to the true market price. In our simulations, these are the first $q \cdot m$ houses. For the other houses, (in our simulations when $i > m \cdot q$) we assume that

$$y_i = b_0 + X_{i1}b_1 + \ldots + X_{ir}b_r + h(X_{i1}, \ldots, X_{ir}) + \epsilon_i \tag{1.2}$$



where we know nothing about the function $h(.,.,...,)$ except that it is always non-negative. In our simulations, we took h to be a linear function given by a vector c with non-negative coefficients so that:

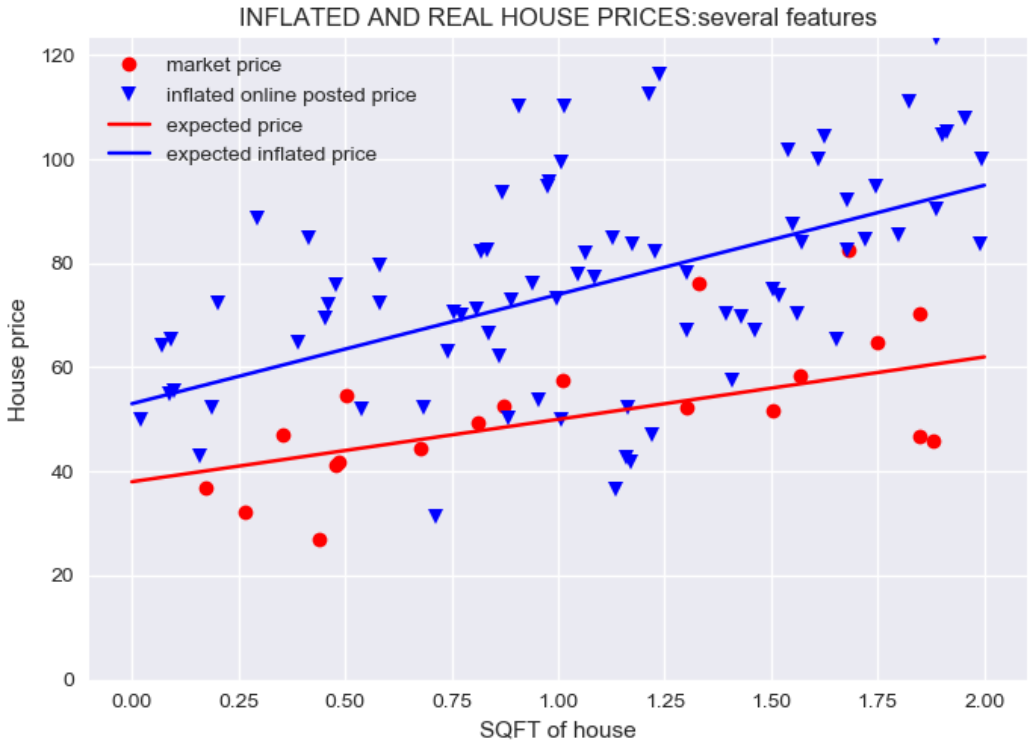
$$y_i = b_0 + c_0 + X_{i1}(b_1 + c_1) + \ldots + X_{ir}(b_r + c_r) + h(X_{i1}, \ldots, X_{ir}) + \epsilon_i \tag{1.3}$$

We took the ϵ_i to be i.i.d standard normal for both the market-price-houses and the inflated-price houses.

We can now formulate our problem:

- Given the data $y = (y_1, \ldots, y_m)$ and the matrix of features X where X_{ij} is the j-th feature of house number i, estimate $b = (b_0, b_1, \ldots, b_r)$. You are not given any other information except that non-market houses have an expected online price above market value. You do not know among others which houses are given at market price.

Now assume that c is "quite a bit away from 0". Then if we only have one feature in a plot we will see "by eye" that there are two linear models present and hence it will be easy to separate the two, that is the overpriced and the real market prices. This is what can be seen in the first of our figures. There we took $b = (11, 2)$ and $c = (3, 1)$. Each feature was simulated like an independent uniform between 0 and 2.



The next simulation was done assuming 3 features. Again the features are each uniform between 0 and 2. But this time, we took $b = (11, 12, 13, 14))$ and $c = (0, 9, 8, 7)$. So, really c has entries which are quite "far away from 0. Hence, the points should be mostly well separated in the full feature space. But, if we represent only one feature vs price, we can not distinguish easily by eye separate the market values from inflated prices. The other features add too much noise. So, if we had an enormous amount of data points, things would be easy: among house having all about the same features take the lowest prices. These will belong to the market prices. But in a higher feature space and sparsity of data points (also due to the fact that in reality there are many different classes of houses, which follow almost separated markets), it is not possible to find many houses having about the same features. So, we see in our second figure the problem. we can no longest easily distinguish between the two type of prices. When plotted against only one dimension the two groups mix up. This is due to the other feature acting as noise. This means that an algorithm is needed to peel off the inflated prices, in the higher dimensional feature space, where "by eye" this is not feasible. We have developed two such algorithms which on our simulated data work almost perfectly.

2 Our algorithms

We have two algorithms:

- The first algorithm consists of "quantile regression" optimized with the help of gradient descent. It works as follows. Let us say you have a rough estimate for the vector of coefficients $\vec{b} = (b_1, b_2, \ldots, b_r)$ Let that current estimate of $\vec{b}$ be denoted by $\vec{b}_{est}$. To start you can take a linear regression for the full data. Then, we are going to try by gradient descent to minimize the following function

$$\texttt{targe function} = \sum_i f(y_i - \vec{b}_{est} \cdot \vec{X}_i)$$

  where $\vec{X}_i$ is the feature vector of house i. We take the function $f(.)$ to be

$$f(x) = x$$

  when $x > 0$ and

$$f(x) < -10x$$

  when $x < 0$. With that function the minimum will be reached when we reach the lower 10% quantile of our points with respect to the linear model for the market values. (note that cost function is convex....so we are sure to find the unique solution with gradient descent). It is easy to prove that with enough points the optimal solution is going to be close to the linear model for market values.

- The second algorithms peals of points from top, but using the general least square regression as reference.

We strongly believe that this is feasible in with the real data at hand because in our simulations our algorithms worked very well. **the result with the quantify-regression-gradient-descent method we found:**
we simulated for "true" values

$$b = (11, 12, 13, 14), c = (1, 9, 8, 7)$$

and $m = 200$ and $q = 0.2$. Hence only 200 houses total, but even less marked-prices. Only about 20 market prices with three dimensional feature space. our estimate with gradient descent is

$$\hat{b} = (12.0, 11.58, 12.70, 13.72)$$

and the estimation error

$$b_{error} = (1.05, -0.4, -0.2, -0.27)$$

not that in terms of relative errors these terms are all less than 10%. If we just take the linear regression of the full data set (containing both market points and inflated points),

the least square model is a convex combination, which takes on the following value in our case:

$$0.2b + 0.8(b + c) = (11.8, 19.2, 19.4, 19, 6)$$

so no comparison to our algorithm. A main task will be to improve on the first coefficient. if c_0 is taken 0, the result for b_0 even gets worse... also we will not have 200 data points but even much less and specially distributed across different classes of houses.... Finally, to select our target function we used the fact that we knew q. In reality,we do not know q. so instead we have to try our algorithm with different values of q and then try to see what makes sense. In reality this allows to bring back the problem to a one dimensional unkown q from a multi-dimensional problem. With the help of only a few "real-market houses" one could then estimate q.