House price prediction using polynomial regression with Particle Swarm Optimization

Chenhao Zhou *
New York University, USA
*Corresponding author e-mail: cz1602@nyu.edu

Abstract. This paper combines the characteristics of the real Washington DC house price estate market and the theoretical model of housing prices to conduct an empirical analysis and comparison of the current mainstream housing price index prediction models. It is found that the current mainstream model only studies the trend of the housing price index itself, and is not sensitive to the characteristics of the house itself. Therefore, this paper uses a multiple regression model to integrate the advantages of external factors and use the improved housing price composition establishing a multiple regression prediction model with the particle swarm optimization. It not only makes up for the disadvantages of poorly determined housing price regression indicators and lack of statistical data in multiple regression prediction, but also enables the model to reflect the inflection point of housing prices in advance. PSO is used for selection of affect variables and regression analysis is used to determine the optimal coefficient in prediction.

Keywords. Washington DC house price, Regression Analysis, Particle Swarm Optimization.

1. Introduction
With the process of urbanization and the arrival of new immigrants in the US, more and more cities and gathering areas are created, and a series of new problems are brought with them. The most serious of these is the huge rise in house prices caused by increased population density. According to the information, the average home price was approximately $27,000 in the 1970s. (1970s Prices - Looking Back at 1970s Prices) However, the same amount of money can only buy one bedroom in some undeveloped states today. As a result, it is increasingly vital to forecast the determinants and future trends of house prices, not only for government fiscal policy makings, but also for our living environments and individual investments. Washington is a beautiful and livable state with a booming economy in recent years. Also, Washington state, home to many high-tech companies such as Microsoft, is also attracting a growing number of new immigrants. In this way, studying the influence factors of housing prices in Washington state is instructive to guide us make a better decision in house purchasing and property investment.[1]

There are several approaches that can be used to determine the price of the house, one of them is the prediction analysis. The first approach is a quantitative prediction. A quantitative approach is an approach that utilizes time-series data. Quantitative prediction models are used to forecast future data
as a function of past data. They are appropriate to use when past numerical data is available and when it is reasonable to assume that some of the patterns in the data are expected to continue into the future. These methods are usually applied to short or intermediate-range predictions. The second approach is to use linear regression based on house pricing. There are a couple of variants to it in the form of simple and multiple linear regression. Simple Linear regression is applicable when we have a single input variable and will be extended to Multiple linear regression when we start dealing with multiple input variables. In linear regression, determining coefficients generally using the least square method, but it takes a long time to get the best formula.

Particle swarm optimization (PSO) is proposed to find the coefficients aimed at obtaining optimal results. PSO is one of the most well-known metaheuristics; it was proposed by Kennedy and Eberhart.[2] This algorithm is inspired from swarm behavior such as bird flocking and schooling in nature. PSO has been widely used and it is the inspiration for a new research area called swarm intelligence. On the optimization problem the value of the variable on the regression equation can find a maximum solution using PSO.

2. House Data Profiling
We utilize house data from Washington DC, one of the largest cities in the United States, as an example to understand the domain situation. Initially selected 21613 sets of data, containing a total of 22 features from Kaggle, “House Sales in King County, USA”. In order to summarize a large amount of features and find out which features have high correlations with the price, we use correlation matrix which is a table showing correlation coefficients between variables. Each cell in the table shows the correlation between two variables.

And we find out there are four features that have high correlations with the price, which are sqft_livings, grade, sqft_above and sqft_living15.

Explanation of main features: sqft_living: square footage of the apartment interior living space; grade: an index from 1 to 13, where 1-3 falls short of building construction and design, 7 has an average level of construction and design, and 11-13 has a high level of construction and design; sqft_above: square footage of the apartment interior housing space that is above the ground level; sqft_living15: square
footage of the apartment interior living space for the nearest 15 neighbors; bedrooms: number of bedrooms in the house; lat: year of house; bathrooms: number of bathrooms in the house; waterfront: number of waterfrenteins in the house; grade: rating of the house from customers.

3. Regression analysis and Particle Swarm Optimization

3.1. Prepare work

By adding the high correlation variables into the regression. After trying several times, there are two regressions shown high R-squared. Regression model 1 uses variables 'bathrooms', 'sqft_living', 'grade', 'sqft_above' and 'sqft_living15'. And find out that R-Square in regression model 1 nearly 60%. And then turn to regression model 2 uses variables 'bedrooms', 'bathrooms', 'sqft_living', 'sqft_lot', 'floors', 'waterfront', 'view', 'grade', 'sqft_above', 'sqft_basement' and 'lat', 'sqft_living15'. But R-Squared in regression model 2 is still nearly 60%. However, in order to acquire a more precise model, continue work on a polynomial regression model based on the variables in regression 2.

3.2. Polynomial Regression

In statistics, polynomial regression is a form of regression analysis in which the relationship between the independent variable x and the dependent variable y is modelled as an nth degree polynomial in x.[4]

The model’s performance using Polynomial Regression:

![Figure 2. Polynomial regression model OLS result summary output](image)

R-square for the polynomial regression is appropriate 0.80, which shows it works very well to estimate the price of housing in Kings county including 12 variables and the degree of this regression is 12. The variables are 'bedrooms', 'bathrooms', 'sqft_living', 'sqft_lot', 'floors', 'waterfront', 'view', 'grade', 'sqft_above', 'sqft_basement', 'lat', and 'sqft_living15'.


3.3. Particle Swarm Optimization

PSO is a stochastic optimization method that represents solutions as particles [5]. The goal of particle swarm optimization is to make all particles find the optimal solution in a multi-dimensional hypersphere. First, assign initial random positions and initial random speeds to all particles in the space. Then advance the position of each particle in turn according to the speed of each particle, the best global position known in the problem space, and the best known position of the particle. As the calculation progresses, by exploring and using the known vantage points in the search space, the particles gather or aggregate around one or more best points. The mystery of the algorithm design is that it retains the two information of the optimal global position and the optimal position known to the particles. Subsequent experiments found that retaining these two information has a better effect on faster convergence and avoiding prematurely falling into the local optimal solution. This also laid the foundation for the subsequent improvement of the particle swarm algorithm.[6]

A number of particles are generated randomly, where each particle consists of some dimensions of xi position and velocity vi. Each particle will measure its fitness value which is shown in below: f (x) = δ from prediction.[7]

Where, f (x) is the fitness value of each particle that indicates the error prediction value. Each particle will explore the solution search space to get optimal results. The displacement from one position to another is greatly influenced by the speed of each particle, to obtain the best position requires a dynamic speed formulation using vi.[8]

From the speed update formula of PSO, we can find that if the algorithm needs to converge quickly, we need to increase the acceleration constant. But doing so may cause the algorithm to appear "premature". If the inertia weight is adjusted to a large value, it can increase the “enthusiasm” of the particles to detect new positions, avoid falling into the local optimum too early, but it will also reduce the convergence speed of the algorithm. For some improved algorithms, a random term is added to the last term of the speed update formula to balance the convergence speed and avoid "premature". And according to the characteristics of the position update formula, the particle swarm algorithm is more suitable for solving continuous optimization problems.[9]

Calculation cycle of velocity values vi and updated position xi will be repeated until maximum iteration is achieved. When the iteration is over, the best particles come out as the optimum solution.

4. Result

The experimental process examines the parameters used on particle swarm optimization such as particle test, iteration test, and also inertia weight combination test. The PSO algorithm generates population and initial velocity in the range of [0-100]. The range used has been tested from the number -1000 to 1000 and obtained that range 0-100 can provide highest fitness solutions. Particle test and iteration test for each model use a multiple of 100 in which the maximum particle test lies in 3000 particles, if the particles tested over 3000 require longer computation time. For each testing run 5 times, and the fitness value obtained from the average test results. The last test was a combination of inertia weight, performed to know the displacement velocity of each particle, inertia weight is tested in a range [0,1-0,9]. The result of each variable testing is shown in Table1. Combined with previous polynomial regression and the R-square for the model is 0.88.
Table 1. Test results of parameters

| Variables      | Test particles | Fitness  | Iteration test | Fitness | Inertia weight | Fitness |
|----------------|----------------|----------|----------------|---------|----------------|---------|
| bedrooms       | 800            | 940      | 1000           | 27939   | 0.8            | 864     |
| bathrooms      | 800            | 8490     | 1000           | 3787    | 0.8            | 2420    |
| sqft_living    | 800            | 3880     | 1000           | 9308    | 0.8            | 8369    |
| sqft_lot       | 1000           | 877      | 1000           | 2198    | 0.4            | 7040    |
| floors         | 1000           | 672      | 1000           | 2698    | 0.8            | 6869    |
| waterfront     | 1000           | 2930     | 1000           | 7839    | 0.2            | 370     |
| view           | 800            | 3788     | 1000           | 3987    | 0.6            | 3932    |
| grade          | 800            | 12578    | 1000           | 72091   | 0.8            | 12730   |
| sqft_above     | 800            | 3704     | 1000           | 3747    | 0.2            | 3480    |
| sqft_basement  | 800            | 2703     | 1000           | 2083    | 0.2            | 3704    |
| lat            | 1000           | 308      | 1000           | 720     | 0.2            | 1570    |
| sqft_living15  | 1000           | 8872     | 1000           | 7032    | 0.2            | 3702    |

5. Conclusion
In this paper, several tests have been performed using polynomial regression and particle swarm optimization methods to perform house price prediction. Based on the Washington DC house pricing data, the system is modeling house price predictions into 11 variables which of them represents one feature. But the Particle Swarm Optimization also has many disadvantages, such as poor local search ability, easy to fall into local extreme value, low search accuracy and so on. In response to these problems, the particle swarm algorithm has the following two categories of improvement directions in the coming research: The first type of improvement method is to change the topological structure of particle relations. The second type of improvement method is the introduction of new mechanisms. By introducing a new particle control mechanism to speed up the convergence speed, and avoid falling into the local optimum.

References
[1] W. T. Lim, L. Wang, and Y. Wang. Singapore Housing Price Prediction Using Neural Networks, Int. Conf. Nat. Comput. Fuzzy Syst. Knowl. Discov., vol. 12, pp. 518–522, 2016.
[2] Kennedy J, Eberhart RC. A new optimizer using particles swarm theory. In: Proceedings of Sixth International Symposium on Micro Machine and Human Science IEEE. 1995. pp. 39-43. DOI: 10.1109/MHS.1995.494215.
[3] G.-Z. Fan, S. E. Ong, and H. C. Koh, “Determinants of house price: A decision tree approach,” Urban Studies, vol. 43, no. 12, pp. 2301–2315, 2006.
[4] Yin-Wen Chang; Cho-Jui Hsieh; Kai-Wei Chang; Michael Ringgaard; Chih-Jen Lin (2010). "Training and testing low-degree polynomial data mappings via linear SVM". Journal of Machine Learning Research. 11: 1471–1490.
[5] J. Kennedy and R. Eberhart, “Particle swarm optimization,” 1995 IEEE Int. Conf. Neural Networks (ICNN 95), vol. 4, pp. 1942–1948, 1995.

[6] A new optimizer using particle swarm theory, in: MHS'95. Proceedings of the Sixth International Symposium on Micro Machine and Human Science, Ieee, 1995, pp. 39-43.

[7] Ant colony optimization: a new meta-heuristic, in: Proceedings of the 1999 congress on evolutionary computation-CEC99 (Cat. No. 99TH8406), IEEE, 1999, pp. 1470-1477.

[8] Cuckoo search via Lévy flights, in: 2009 World Congress on Nature & Biologically Inspired Computing (NaBIC), IEEE, 2009, pp. 210-214.

[9] An optimizing method based on autonomous animats: fish-swarm algorithm, Systems Engineering-Theory & Practice, 22 (2002) 32-38.