International Journal of Academic Research in Business and Social Sciences
February 2012, Vol. 2, No. 2
ISSN: 2222-6990

Predicting Wheat Production in Iran Using an Artificial Neural Networks Approach

Reza Ghodsi, Ruzbeh Mirabdollah Yani, Rana Jalali, Mahsa Ruzbahman
Industrial Engineering Department, University of Tehran, Iran

ABSTRACT

An analysis of the effects of various factors such as climate factors on wheat yield was performed in Iran country in order to obtain models suitable for yield estimation and regional grain production prediction. Climate data from meteorological records and other data from central bank of Iran were employed. Annual data for 18 years were applied in this study. In order to predict wheat production, artificial neural networks (ANN) methodologies were tested for analyzing the data. To conduct this review, we have considered 8 factors as inputs and wheat production is used as output for the ANN algorithm. Our input variables are: rainfall, guaranteed purchasing price, area under cultivation, subsidy, insured area, inventory, import, population, value-added of agriculture group. The comparison of real wheat production with ANN output in the last five years of this study shows that the proposed ANN model is a suitable way of predicting wheat production.

KEY WORDS: Neural Networks, Forecasting, Wheat Production

1. Introduction

Iran is one of the most important countries in producing wheat in Asia and use of wheat in many part of this country is seen widely. In other word, high percentage of the area is devoted to agriculture, being wheat one of the main crops. The region is considered as suitable areas for grain crops production. The effects of soil properties and climate on wheat yield have been assessed in many areas of the Iran. For this reason, we gathered data's of rainfall from meteorological records in different parts of this country and used them as an input variable. Many other effectual factors such as guaranteed purchasing price, area under cultivation, subsidy, insured area, inventory, import, population and value-added of agriculture group are considered and gathered from central bank of Iran. These data's are in annual form and contain of 18 years from 1988 to 2006 [1].

In order to conduct this review, we have used ANN to predict the wheat production. ANNs are a promising alternative to econometric models. An ANN is an information-processing paradigm that is inspired by the way biological nervous systems, such as the brain, process information. ANNs, like people, learn by example. An ANN is configured for a specific application, such as pattern recognition, function approximator or data classification, through a learning process. Learning in biological systems involves adjustments to the synaptic connections that exist between the neurons. This is true for ANNs as well. They are made up simple processing units, which are linked by weighted connections to form structures that are able to learn relationships between sets of variables. This heuristic method can be useful for non-linear process that has an unknown functional form (Enders 2004). There has been a vast literature about ANNs, basically in the empirical field, showed that ANNs comparability or superiority to conventional methods for estimating functions [2, 3, 4, 5, 6, 7, 8, 9, 10]. So in this study, ANNs is selected for estimating system performance function and then performance evaluation. Commonly, neural network technique is used as a complementary tool for parametric and non-parametric methods such as DEA, to fit system performance functions and measure efficiency under non-linear contexts. In fact, applying ANNs can reduce the restrictive assumptions each of these methods [10].

2. Artificial Neural Networks

An ANN is an information processing paradigm that is inspired by the way biological nervous systems, such as the brain, process information. ANNs have been applied when there is no theoretical evidence about the functional forms. Therefore, ANNs are data-based, rather than model-based. The key element of this paradigm is the novel structure of the information processing system. It is composed of a large number of highly interconnected processing elements (neurons) working in unison to solve specific problems [10].

Among the different networks, the feed forward neural networks or multi layer perceptron (MLP) are the most commonly used in engineering science. MLP networks are normally arranged in three layers of neurons, the so-called multilayer structure:

- Input layer: its neurons (also called nodes or processing units) introduce the model inputs.
- Hidden layer(s) (one or more layers): its nodes combine the inputs with weights that are adapted during the learning process.
- Output layer: this layer provides the estimations of the network [10].

In these networks, the output is function of the linear combination of hidden units' activations; each one is a non-linear function of the weighted sum of inputs:

$$y = f(x, \theta) + \varepsilon \qquad (1)$$

Where x is the vector of explanatory variables, ε is the random error component. $f(x, \theta) = \hat{y}$ is the unknown function for estimation and prediction from the available data. Consider a MLP with three layers and one output. The network consists of the following form:

$$\hat{y} = F\left(\upsilon_0 + \sum_{j=1}^{m} H(\lambda_j + \sum_{i=1}^{n} x_i \theta_{ij}) \upsilon_j \right) \qquad (2)$$

Where:
$\hat{y}$: network output,
F : output unit activation function,
H: hidden unit activation function,
n : number of input units,
m: number of hidden units,
x_j :input vector for unit j (x_{ji} = i^{th} input to the j^{th} unit),
θ_{ij} : weight from input layer i to hidden unit j,
υ_o : output bias,

λ_j : hidden units biases (j = 1 . . . m),
υ_j : weights from hidden unit j to output (j = 1 . . . m).

From Equation 2, it can be observed that MLPs are mathematical models often equivalent to conventional models in econometrics (linear regression, auto regressive moving average (ARMA) models for time series analysis), but with specific estimation methods [11]. In Figure 1 a MLP with three layers and one output is shown [10].

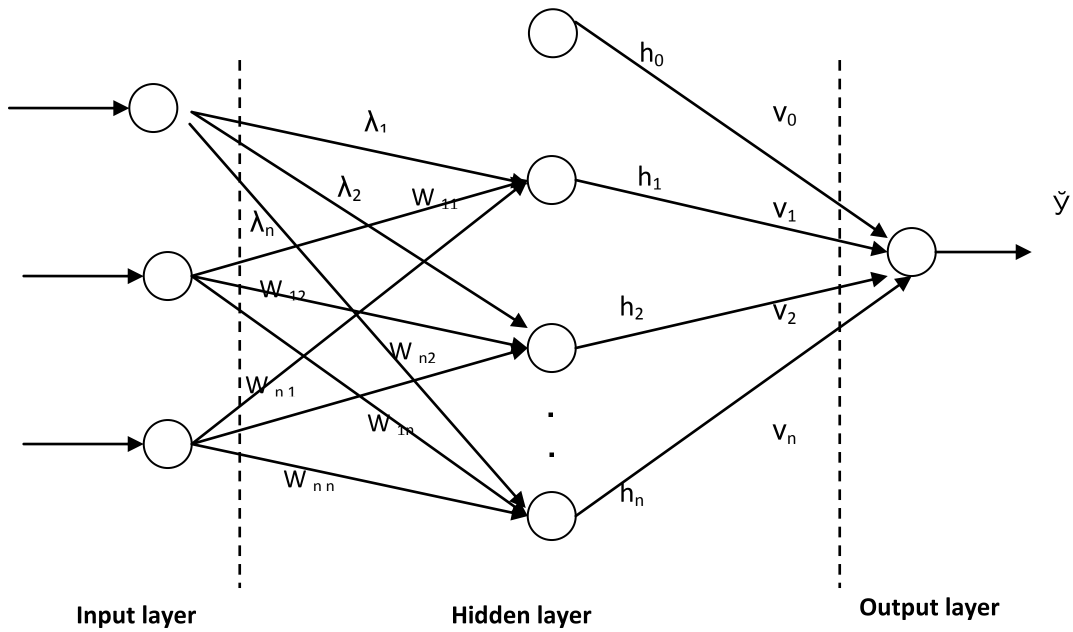

Figure 1: Single output, three layer feed forward MLP neural network

The activation function for output layer is generally linear. The non-linear feature is introduced at the hidden transfer function. From the previous universal approximation studies, these transfer functions must have mild regularity conditions: continuous, bounded, differentiable and monotonic increasing [12]. The most popular transfer function is sigmoid or logistic, nearly linear in the central part. Architecture selection is one major issue with implications on the empirical results and consists of:

1. Input and output variables number.
2. Hidden layers' number.
3. Hidden and output activation function.
4. Learning algorithm

All of the above issues are open questions today and there are several answers to each one. The hidden units' number is determined by a trial-error process considering m = 1, 2, 3, 4 . . . . Finally, it is common to eliminate 'irrelevant' inputs or hidden units [10, 13].

Another critical issue in ANNs is the neural learning or model estimation based upon searching the weights that minimize some cost function such as square error:

$$\text{Min } [E(y - f(x, \theta))^2] \ \theta \in \Theta \qquad (3)$$

The most popular learning algorithm is the Back Proportion (BP). BP learning is a kind of supervised learning introduced by Werbos (1974) [14] and later developed by Rumelhart and McClelland (1986) [15]. Desirable output for input set is made by this algorithm. Error in each neuron is the difference between ANN output and real output. The interconnections weight and threshold value in each neuron is adjusted to minimize the error. Let E denote error function. In this algorithm for reducing error, the weights vector (ϑ) is adjusted. For this $\frac{\partial E}{\partial \theta}$ is calculated. Let:

o_j = output of unit j,
θ_{ij} : weight from input layer i to hidden unit j,
z_j: $\theta_j . x_j$ the weighted sum of inputs for unit j,

By applying chain rule, we have:

$$\frac{\partial E}{\partial \theta_{ji}} = \frac{\partial E}{\partial z_j} . \frac{\partial z_j}{\partial \theta_{ji}}$$

It is proof that:

$-\frac{\partial E}{\partial \theta_{ji}} = \delta_i . o_j$, That: $\frac{\partial E}{\partial z_j} = \delta_i$

Thus:

$$\Delta \theta_{ji} = \eta \delta_i o_j$$

$$\theta_{ji}(k+1) = \theta_{ji}(k) + \eta \delta_i o_j$$

This algorithm process can be as follows:

$$\theta(k+1) = \theta(k) - \eta \frac{\partial E}{\partial \theta}(k) \quad (4)$$

Or

$$\theta(k+1) = \theta(k) + \eta \nabla f(x,\theta)\left[y - f(x,\theta)\right] \quad (5)$$

BP is an iterative process (k indicators iteration). Parameters are revised from the error function (E) gradient by the learning rate η, constant or variable. The error propagates backwards to correct the weights until some stoppage criterion – epoch, error goals – is reached [10].
Adding a term called momentum can improve this algorithm:

$$\theta(k+1) = \theta(k) + \eta \frac{\partial E}{\partial \theta}(k) + \mu \Delta \theta(k-1)$$

After neural training (training set), new observations (validation and/or test sets) are presented to the network to verify the so-called generalization capability [16]. ANNs have advantages, but logically they also have several drawbacks. Therefore, ANNs can learn from experience and can generalize, estimate, predict, with few assumptions about data and relationships between variables. These attributes have made the ANN approach fairly efficient for problem solving. Hence, ANNs have an important role when these relationships are unknown (non-parametric method) or non-linear (non-linear method), provided there are enough observations with flexible form and universal approximation property. However, the flexibility could cause learning of the noise. Finally, algorithm convergence and trial and error process are also some relevant drawbacks [10].

3. Methodology

An integrated ANN algorithm is proposed to predict wheat production. In this article, we have followed these steps:

1. First of all, we determined input(s) and output (P) variables of the model.
2. We considered data set S in all available previous periods which describes the input-output relationship. Assume that there are n years to be evaluated. Note that the current period data (Sc) does not belong to S.
3. We divided S in to two subsets: training (S1) and test (S2) data.
4. We have used ANN method to estimate relation between input(s) and output(s). For this reason, we have followed these steps (Murat, 2005):
   - Select architecture and training parameters.
   - Train the model using the training data (S1).
   - Evaluate the model using the test data (S2).
   - Repeat these steps using different architectures and training parameters.
   - Select the best network architecture (ANN) from the testing data error (MAPE).

$$mape = \frac{1}{N}\sum_{i=1}^{N}\left|\frac{\mathrm{Actualvalue_i} - \mathrm{Setpointvalue_i}}{\mathrm{Setpointvalue_i}}\right|$$

(N: the number of rows)

5. Run ANN for Sc [10].
6. Analyzing performances of proposed ANN algorithm by using ANOVA.

4. The Experiment: The Integrated ANN Algorithm

4.1. Data Collection and Analysis

For this study, we have used 8 main factors as inputs. One of them is rainfall and we have got data’s of rainfall from meteorological records. Other seven inputs are collected from central bank of Iran. These data’s are in annual form and contain of 18 years from 1988 to 2006

4.2. Running the Proposed Algorithm

Step 1: As shown before, the first step is determination of input (S) and output (P) variables of the model. We have chosen 8 main factors (rainfall, guaranteed purchasing price, area under cultivation, subsidy, insured area, inventory, population, value-added of agriculture group) as input variables. The output factor is Wheat production, as it is shown in table 1. Then, for each factor, we normalized the data to avoid undesirable effect of unbalanced data in ANN algorithm. The raw and normalized data are shown in Appendix 1 and 2.

Table 1: effective factors

	factor	type
1	rainfall	Input
2	guaranteed purchasing price	Input
3	area under cultivation	Input
4	subsidy	Input
5	insured area	Input
6	inventory	Input
7	population	Input
8	value-added of agriculture group	Input
9	Wheat production	Output

Step 2: 18 rows of data are collected (1988 to 2006). This is done to enhance our integrated ANN algorithm. Furthermore, ANN requires extensive data set to be trained (usually more than 100 data).

Step 3: S_1 is 13 rows of data for training (1988 to 2001) and S_2 is 5 rows of data for testing (2002 to 2006).

Step 4: In order to get the best ANN for this study, 11 distinct ANN models are tested to find the optimum model for the output which is wheat production. In the other words, we have found best ANN model for the output. Max number of neurons in the first hidden layer is set to 100. Each model is run 100 to take care of possible bias or noise. The architecture of the ANN-MLP models and their MAPE values are shown in Table 2. It seems the 6th ANN-MLP model for wheat production has the lowest MAPE and consequently is chosen as the preferred or optimum model. Figures 1 shows the MAPE of 11 ANN models in table 2. It can be seen that the fluctuations are within a narrow range. Therefore, it verifies previous finding that the 6th ANN-MLP is an ideal candidate for the purpose of our study.

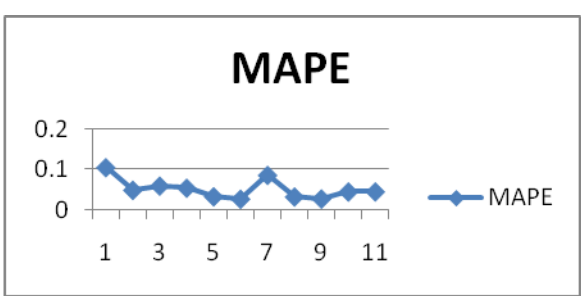

Figure1: Relative error estimate for the 11 ANN models

Step 5: Therefore, the 6th ANN-MLP is selected for estimating the wheat production. In order to identify an ANN output for each year, we tested our preferred model for the last five years including 2002 to 2006. Figure 2 present the distance between the results of the preferred ANN and original data for each output. This figure validates that the preferred model is relatively close to actual data. Appendixes 1 and 2 show raw data and accumulated data used in the integrated ANN algorithm.

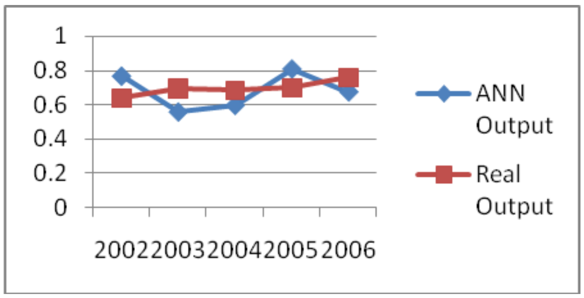

Figure 2: Result of the selected ANN and original data for wheat production

Table 2: Architecture of the 25 ANN-MLP models and their associated relative error (MAPE)

ANN-MLP Model number	Learning method	Number of neurons in first hidden layer	First transfer function	Second transfer function	MAPE
1	LM	52	tansig	purelin	0.105544
2	BR	99	tansig	purelin	0.049151
3	GDX	2	tansig	purelin	0.059857
4	SCG	5	tansig	purelin	0.054477
5	GDA	89	tansig	purelin	0.033023
6	***CGP***	***80***	***tansig***	***purelin***	***0.027049***
7	CGF	40	tansig	purelin	0.086307
8	BFG	62	tansig	purelin	0.03251
9	GD	60	tansig	purelin	0.027534
10	GDM	4	tansig	purelin	0.045054
11	OSS	35	tansig	purelin	0.045249

BFG : BFGS quasi-Newton backpropagation
BR : Bayesian regularization
CGF : Fletcher-Powell conjugate gradient backpropagation
CGP : Polak-Ribiére conjugate gradient backpropagation
GD : Gradient descent backpropagation
GDA : Gradient descent with adaptive learning rule backpropagation

GDM : Gradient descent with momentum backpropagation
GDX : Gradient descent with momentum and adaptive learning rule backpropagation
LM : Levenberg-Marquardt backpropagation
OSS : One step secant backpropagation
SCG : Scaled conjugate gradient backpropagation

Table 3: Comparison of ANN Output and Real Output for the last 5 years

Year	2002	2003	2004	2005	2006
ANN Output	0.765386	0.556301	0.595047	0.806812	0.673632
Real Output	0.641907	0.695782	0.683364	0.700367	0.758778

Step 6: By using ANOVA the equal average assumption was not rejected between different years. From the above experiment we concluded that the purposed ANN algorithm was effective for predicting wheat production. Thus, the proposed ANN algorithm could be applied for predicting production of wheat in future years.

One-way ANOVA: ANN output; real output

Analysis of Variance

Source	DF	SS	MS	F	P
Factor	1	0.00069	0.00069	0.10	0.755
Error	8	0.05298	0.00662		
Total	9	0.05366			

5. Conclusion

Iran is one of the most important countries in producing wheat in Asia and use of wheat in many part of this country is seen widely. For this reason, we have decided to predict production of wheat in Iran. In order to conduct this review, we have used ANN as a prediction tool and chose 8 significant factors in wheat producing. We gathered data's of rainfall from meteorological records in different parts of this country and used them as an input variable. Many other effectual factors such as guaranteed purchasing price, area under cultivation, subsidy, insured area, inventory, import, population and value-added of agriculture group are considered and gathered from central bank of Iran. These data's are annual and contain of 18 years from 1988 to 2006.In other words; we have considered 8 factors as input factors. In order to get the best ANN for this study, 11 distinct ANN models are tested to find the optimum model for wheat production. In this way, we have found best ANN model for the output. Max number of neurons in the first hidden layer is set to 100. Each model is run 100 times to take care of possible bias or noise. As shown in table 1 the 6th ANN-MLP model for wheat production has the lowest MAPE and consequently is chosen as the preferred or optimum model. By using the best ANN model for our review, we predict production of wheat for 5 years including 2002 to 2006. The results show that the proposed ANN model is a suitable way of predicting wheat production.

7. References:

[1] Alvarez, R., 2009. Predicting average regional yield and production of wheat in the Argentine Pampas by an artificial neural network approach. *European. Journal of Agronomy 30*, 2009, 70–77.

[2] Tang, Z., Almeida, C., Fishwick, P. A Time series forecasting using neural networks vs. Box–Jenkins methodology. *Journal of Simulation 57 (5)*, 1991, 303–310.

[3] Tang, Z., Fishwick, P.A. Back-propagation neural nets as models for time series forecasting. ORSA *Journal on Computing 5 (4)*, 1993, 374–385.

[4] Jhee, W.C., Lee, J.K. Performance of neural networks in managerial forecasting. *Intelligent Systems in Accounting Finance and Management 2*, 1993, 55–71.

[5] Hill, T., O'Connor, M., Remus, W. Neural network models for time series forecasts. *Management Science 42 (7)*, 1996, 1082–1092.

[6] Kohzadi, N., Boyd, M.S., Kermanshahi, B., Kaastra, I. A comparison of artificial neural network and time series models for forecasting commodity prices. *Neurocomputing 10 (3)*, 1996, 169–181.

[7] Chiang, W.C., Urban, T.L., Baldridge, G.W. A neural network approach to mutual fund net asset value forecasting. Omega, *The International Journal of Management Science 24,* 1996, 205–215.

[8] Indro, D.C., Jiang, C.X., Patuwo, B.E., Zhang, G.P. Predicting mutual fund performance using artificial neural networks. Omega, *The International Journal of Management Science 27*, 1999, 373–380.

[9] Brian Hwarng, H. Insights into neural-network forecasting of time series corresponding to ARMA(p;q) structures. *Omega 29*, 2001, 273–289.

[10] Azadeh, A., Ghaderi, S.F., Anvari, M., Saberi, M., 2006f. Measuring performance electric power generations using artificial neural networks and fuzzy clustering. *Proceedings of the 32nd Annual Conference of the IEEE Industrial Electronics Society—IECON'06, Conservatoire National des Arts & Metiers, Paris, France*.

[11] Cheng, B., Titterington, D.M. Neural networks: a review from a statistical perspective. *Statistical Science 9 (1)*, 1994, 2–54.

[12] Hornik, K., Stinchcombe, M., White, H. Multilayer feed forward networks are universal approximators. *Neural Networks 2*, 1989, 359–366.

[13] White, H. Some asymptotic results for learning in single hiddenlayer feedforward network models. *Journal of the American Statistical Association 84 (408),* 1989 1003–1013.

[14] Werbos, P.I., 1974. Beyond regression: new tools for prediction and analysis in the behavior sciences. Ph.D. Thesis, Harvard University, Cambridge, MA.

[15] Rumelhart, D.E., McClelland, J.L., 1986. Parallel Distributed Processing: Explorations in the Microstructure of Cognition. Foundations, vol. 1. MIT Press, Cambridge, MA.

[16] Schiffmann, W., Joost, M., Werner, R. Optimization of the backpropagation algorithm for training multilayer perceptrons. Technical Report 16/92, Institute of Physics, University of Koblenz. 1992

8. Appendix :

Appendix 1: raw data used in the integrated ANN algorithm

	Input 1	Input 2	Input 3	Input 4	Input 5	Input 6	Input 7	Input 8	Output
Year	rainfall	guaranteed purchasing price	area under cultivation	subsidy	insured area	inventory	value-added of agriculture group	population	Wheat production
1989	9066.3	100	6257	82	821	498	5893	53187	6010
1990	8420.6	100	6278	155	853	939	6591	54496	8012
1991	7614.9	130	6193	250.2	887	786	8977	55837	8793
1992	9657.6	150	6640	512.5	902	442	12033	56656	10179
1993	10286.4	225	6807	1154.4	914	493	15331	57488	10732
1994	10387.8	260	6782	2095.4	983.3	1171	20482	58331	10870
1995	10520	330	6567	2881	1256	1197	34575	59187	11228
1996	7603.2	410	6328	3609	1360	427	38868	60055	10015
1997	9678.4	480	6299	3390	1217.1	1425	43162	61070	10045
1998	9304.5	600	6180	4447.5	1234.4	663	56751	62103	11955
1999	8475.2	672	4739	5200	1278.1	1032	65421	63152	8673
2000	8082.1	875	5101	5835	1026.6	1680	79121	64219	8088
2001	8609.4	1050	5553	6818.7	1084.4	2476	85238	65301	9459
2002	7157.8	1300	6241	10060.5	2252.3	2831	111276	66300	12450
2003	9013.4	1500	6409	11788	3933.9	2607	130226	67315	13440
200	9460.	1700	6605	14048	4079.	3300	156697	68345	14568

4	5			.8	6				
2005	10412.5	1870	6951	24577.8	4004.3	2426	171811	69390	14308
2006	9082.3	2050	6879	25870	3868	4099.1	207037	70496	14664

Appendix 2: Accumulated data used in the integrated ANN algorithm

	Input 1	Input 2	Input 3	Input 4	Input 5	Input 6	Input 7	Input 8	output
Year	rainfall	guaranteed purchasing price	area under cultivation	subsidy	insured area	inventory	value-added of agriculture group	population	Wheat production
1989	0.526614	0.063084	0.521282	0.005571	0.229806	0.154292	0.038842	0.44904	0.287043
1990	0.489109	0.063084	0.523032	0.010531	0.238763	0.290924	0.043443	0.460092	0.382661
1991	0.44231	0.082009	0.51595	0.016998	0.24828	0.243521	0.059169	0.471413	0.419962
1992	0.56096	0.094625	0.55319	0.034819	0.252479	0.136942	0.079312	0.478328	0.486159
1993	0.597483	0.141938	0.567103	0.078429	0.255838	0.152743	0.10105	0.485352	0.512571
1994	0.603373	0.164017	0.565021	0.142359	0.275235	0.362803	0.135001	0.492469	0.519162
1995	0.611052	0.208176	0.547109	0.195732	0.351567	0.370858	0.227892	0.499696	0.53626
1996	0.44163	0.258642	0.527197	0.245192	0.380677	0.132294	0.256188	0.507024	0.478326
1997	0.562168	0.302801	0.524781	0.230313	0.340678	0.441498	0.28449	0.515594	0.479759
1998	0.54045	0.378501	0.514867	0.302159	0.345521	0.205413	0.374059	0.524315	0.570982
1999	0.49228	0.423921	0.394815	0.353283	0.357753	0.319737	0.431204	0.533171	0.414231
2000	0.469447	0.551981	0.424974	0.396424	0.287355	0.520503	0.521504	0.54218	0.386291
2001	0.500075	0.662377	0.46263	0.463256	0.303534	0.767122	0.561823	0.551315	0.451771
2002	0.415759	0.820086	0.519949	0.683501	0.630441	0.877109	0.733445	0.559749	0.594624

2003	0.523541	0.946253	0.533945	0.800865	1.101137	0.807708	0.858349	0.568318	0.641907
2004	0.549511	1.07242	0.550275	0.954462	1.14192	1.022416	1.032825	0.577014	0.695782
2005	0.604808	1.179662	0.5791	1.669792	1.120843	0.75163	1.132445	0.585837	0.683364
2006	0.527544	1.293212	0.573102	1.757583	1.082691	1.269995	1.364627	0.595174	0.700367