Financial Prediction using ANFIS

CONTENT

- The goal of the project.
- What method is used to predict?
- Interpretation of the results.
- Conclusions and remarks.

MOTIVATION

- Our goal is to predict the GDP (gross domestic product) of USA using ANFIS, then compare the results with other existing method relying on statistical tools.

What is ANFIS?

- **Adaptive Network Based Inference System (ANFIS):**
  - Is a fuzzy inference system implemented in the framework of adaptive network.
  - It uses a hybrid learning strategy to update the parameters (weights).
  - It is used, as a basis for constructing a set of fuzzy if-then rules to generate desired output for a given input.
- **ANFIS architecture.**
  - We assume that the fuzzy inference system under consideration has two inputs x and y and one output z. The fuzzy rules are:

ANFIS Architecture

- **R1 if x is A1 and y is B1 then $f_1 = p_1*x + q_1*y + r_1$.**
- **R2 if x is A2 and y is B2 then $f_2 = p_2*x + q_2*y + r_2$.**

Layer 1: every node i in this layer is a square node function.

$O^1_i = \mu Ai(x)$, where x is the input to node i, and Ai is the linguistic label associated with this node function. In other words Oi^1 is the membership function of Ai .

Layer 2: every node in this layer is labeled Π that multiplies the incoming signals and sends the product out.

Layer 3: every node in this layer is a circle node labeled N. The i-th node calculates the ratio of the i-th rule's firing strength to the sum of all rules' firing strength:

$wi = wi/(w_1 + w_2)$, i= 1,2.

ANFIS Architecture (continue)

$wi = wi/(w_1 + w_2)$, $i = 1,2$.

Layer 4: every node i in this layer is a square node with a node function.

$$O4,i = \overline{wi}*fi = \overline{wi}(pi*x + qi*x + ri),$$

Where $\overline{wi}$ is the output of layer3, and{pi,qi,ri} is the parameter set.

Layer 5: the single node in this layer is a circle node labeled Σ that computes the overall output as the summation of all incoming signals.

- The form of membership functions is:

$\mu_{Ai}(x) = 1/(1 + [((x - c) / a)^2]^b.$

ANFIS

- The only weights in our application are the parameters of the membership functions in the input layer and the parameters in the fourth.
- The membership functions in the input layer have 3 parameters {a, b, c}.
- The number of parameters in the fourth layer is number of inputs + 1.
- The hybrid learning strategy uses both gradient descent for updating the weights of the input layer in the backward pass and the least squares estimate (LSE) for updating the fourth layer parameters in the forward pass.

The description of the input output and the rules

- **Description of the input**
  - The input vector is:

$(gdp(t-(D-1)\Delta),\ldots, gdp(t-\Delta), gdp(t))$,

$t = year + 1/4 * (quarter\_number)$.

quarter_number = 1,…, 4.

D = 2,..,10. (Values of D in the experiments)➔ It means the number of quarters to for the prediction.

D is the number of features for a given input.

$\Delta = 1/4$. Year = 1946,…, 1997.

$gdp(t + P)$ is the value to predict. $P = 1/4$➔One quarter ahead.

The description of input output (continue)

- Example:

  For D = 4.

  Vector = (gdp(97 quarter I), gdp(97 quarter II), gdp(97 quarter III), gdp(97 quarter IV),

  gdp(98 quarter I)).

  The GDPs from 87 first quarter to 87 fourth quarter are used to predict 1988 first quarter GDP.

- **Description of the output**

  The output is represented in the following form:

  The output = α*gdp(t - (D-1)Δ) + β*gdp(t - (D-2)Δ) + γ*gdp(t - Δ) + δ*gdp(t) + φ.

Description of Rules

- **Description of the rules**

The set of rules can be represented in the following form:

If gdp(t -(D-1)Δ) is $SMALL_1$ and gdp(t - (D-2)Δ) is $SMALL_2$ and gdp(t-(D-3)Δ) is $LARGE_3$...and gdp(t) is $SMALL_d$ then gdp(t +P) = $C_{1.}$X.

If gdp(t - (D-1)Δ) is $LARGE_1$ and gdp(t - (D-2)Δ) is $LARGE_2$ and gdp(t - (D-3)Δ) is $LARGE_3$…and gdp(t) is $LARGE_d$ then gdp(t +P) = C_n.X.

Where n is the number of rules and C_i is a row vector of the matrix C((n+1)*(n+1)). This matrix represents the result of the learning.

Prediction Performance

- **Error measurement**
  - We have different error measure:
    - RMSE: root mean squared error.
    - NDEI: non-dimensional error index.
    - APE: average percentage.
  - Let n be the number of data.
  - T is the target output and O is the method output.
  - $e_i = T(i) - O(i)$.
  - $RMSE = \sqrt{(\Sigma_i\ e_i^2)}/\sqrt{n}$.
  - $NDEI = RMSE / \sqrt{\Sigma_i\ (t(i) - 1/n\ \Sigma_i\ t(i))^2}$.
  - $APE = 1/n * \Sigma_i\ |e_i| / |t(i)| * 100\%$.

ANFIS Performance

First experiment

The testing set is composed of 10 vectors.

The testing set is composed of the next ten 10 vectors

use the ten previous vectors to predict the ten next GDPs.

Second experiment

The testing set is composed of 9/10 of the number of data.

The testing set is composed of 1/10 of the number of data.

- Concerning decrease and increase rate of the step size, we can consider them as the following 0.9 (➔decrease by 10%) and 1.1 (➔increase by 10%).
- The method adjusts the step size following these two heuristic rules to increase the convergence time

ANFIS Performance (continue)

Summary of the results:

T.RMSE	C.RMSE	T.NDEI	C.NDEI	T.APE	C.APE
0.3424	15.2887	0.0191	1.0166	0.0006	0.0252
1.3980	30.6977	0.0295	0.5706	0.0014	0.0285
0.6896	48.0816	0.0049	0.2487	0.0003	0.0172
3.6909	43.0415	0.0094	0.0052	0.0002	0.0052

Second experiment

Case1: using two previous quarters to predict the third one.

N° of input	N° MF	N° Rules	N° epoch	T. RMSE
2	4	16	120	18.0781

C.RMSE	T.NDEI	C. NDEI	T.APE	C.APE
24.5054	0.0098	0.0546	0.0082	0.0025

Case 2: using the three previous quarters to predict the third one.

N° of input	**N° MF**	**N° Rules**	**N° epoch**	**T. RMSE**
3	2	8	105	18.0011

C.RMSE	**T.NDEI**	**C. NDEI**	**T.APE**	**C.APE**
25.4228	0.0096	0.0537	0.0080	0.0025

Case3: using the four previous quarters to predict the third one.

N° of input	**N° MF**	**N° Rules**	**N° epoch**	**T. RMSE**
4	2	16	170	16.6546

C.RMSE	**T.NDEI**	**C. NDEI**	**T.APE**	**C.APE**
24.0554	0.0071	0.03278	0.0083	0.0020

Conclusion and remarks

- Error measures diminish when we increase the number of data in the training set.
- Error measures diminish also when we increase the number of epoch.
- Error measures for the training set are less than the error measures for the testing set.
- When increasing the length of the vector for predicting the output the errors diminish.
- ANFIS can achieve a highly nonlinear mapping➔ better prediction.
- ANFIS uses less parameter than other neural networks➔the only layers that have parameters are the input layer and the fourth layer.

Conclusion and remarks (continue)

- ANFIS performance decreases in the case of long time prediction.
- ANFIS uses a hybrid learning strategy: ➔Not only can this hybrid decrease the dimension of the search space in the gradient method, but, in general, it will also cut down substantially the convergence time.