Introducing an Inequalities Index from the Simple Perceptron Pattern in ANN: an unemployment inequalities application in Regional Economics

D. Tsiotas* and S. Polyzos

Dep. of Planning and Regional Development, University of Thessaly,Pedion Areos, Volos, 38 334, Greece.

_____

Abstract

This article introduces a new inequalities index, which was inspired from the simple perceptron's pattern in the artificial neural networks (ANN) scientific field and nominated by the authors *Modulus Perceptron Inequalities Index* (MPII). The structural similarities of this ANN pattern with a binary logic gate, in the Digital Electronics Theory (DET), lead to deal with the existence of operational similarities between these two models. The introduced index aroused while examining the potentials of the simple perceptron to solve the non-linear separable "*exclusive disjunction*" (XOR) problem, by using techniques of the Theory of Numbers in Mathematics. Since the XOR architecture is considered by the DET an inequality detector, due to its ability to result to the same outcome in the same input values, the solution of the XOR problem for a simple perceptron authorizes the ANN and DET XOR models to be considered identical, in the extend that inequalities is regarded. The above XOR models utility appears to operate satisfactory in Regional Science's inequalities research. The examination of MPII's behavior in comparison with the Theil index, concluded to be almost identical in the extend that slope conservation is regarded and more capable on cases that the Theil's index cannot operate.

Keywords: neural networks, simple perceptron, XOR problem, inequalities measurement, Theil index.

_____

1. Introduction

The ANN simple perceptron pattern was proposed in 1962 by Rosenblatt and was the first and simplest one neuron ANN model. In 1969 Minsky and Papert presented the limitations of that model, regarding its inability to solve non-linear separable problems [1], such as the XOR [2]. Thenceforth, the research about two level neuron patterns was abandoned [1]. The next period, multi-layer ANN neuron patterns were introduced to overcome simple perceptron's limitations [3]. The utility of the model was rearranged in modern research and some solutions of the XOR problem were proposed, by using computational techniques from Complex Number Analysis in Mathematics [4] and from Quantum Computation [5]. The XOR model's architecture is considered from DET an inequality detector, due to its ability to result the same numerical outcome when the input data consists of the same values [6-7]. Solving the XOR problem for a simple perceptron render this ANN model to operate as an electronic XOR gate and to perform similarly, regarding the inequalities detection. These identical XOR patterns (simple perceptron and DET's logical gate) may also provide utility in Regional Economics inequalities research, a scientific sector that enumerates a considerable number of measures, such as the Theil index [8-9]. At the following a solution of the XOR problem for

the ANN simple perceptron is given, by using calculation techniques from the Theory of Numbers in Mathematics [10]. Secondly, the proposed simple perceptron pattern is being formulated to an inequalities index (MPII) and it is compared to the Theil index, regarding their inequalities performance.

2. The simple perceptron and the XOR problem

Simple perceptron (Figure 1) constitutes a solo neuron ANN pattern, consisting of n incoming synapses with weights wi that receive n-input signals si (si = 0 or 1, i=1, 2, 3,…, n). Inside the neuron a weighted summation takes place, due to relation (1) [1].

$$S = \sum_{i}^{n} s_i \cdot w_i \tag{1}$$

When the sum of products is higher than a threshold value θ (S ≥ θ) the neuron activates the value 1, otherwise the neuron stays inactive [1]. The XOR operation [2],[4], concerns a logical function, which recommends a combination of the basic AND, OR and NOT logical operations (functions) as described at relation (2).

XOR: A(XOR)B = {A (AND) NOTB}
    (OR) {NOTA (AND) B} $\tag{2}$

The XOR gate operator (Figure 1) receives two binary inputs (into A and B receptors) and results a binary output, as shown in Table 1. When the input signals (A, B) have the same value the outcome turns to zero, otherwise it turns to monad. The XOR truth table (Table 1) elects the XOR's attribute to indicate inequality (activates 1 when the inputs are unequal).

Table 1. The XOR operation truth table

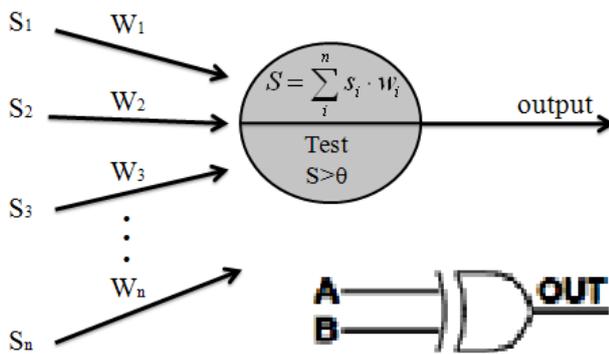| input -A-	input -B-	output -O-
0	0	0
0	1	1
1	0	1
1	1	0

Fig. 1. The simple perceptron and XOR gate circuit structures

It can be considered that the zero result of the XOR gate operates as an equality index and, oppositional, the monad as an inequality index. In order to conform with the DET terms (where the zero/not conducting status of the XOR gate refers to an inactive state that cannot be decoded as a current flow in DET's applications) the equality and inequality logical detectors are considered the NOT XOR (XNOR) and the XOR logical gates in correspondence [6-7]. The XOR problem in the simple perceptron refers to the case that n=2. This problem [2] consists of the fact that there is no combination of the (real) weight values (w1, w2) for the input signal function of relation (3) that verifies the XOR Table 1. Moreover, relation (3) denotes that, for every possible pair of weights (w1, w2), the (0,0), (1,1) and (0,1), (1,0) vector pairs will never exist at opposite half plans (Figure 2), whose boundary is figured by line (3) [1].

$$f(\overline{s}) = f(s_1, s_2) = s_1 \cdot w_1 + s_2 \cdot w_2 = \theta \qquad (3)$$

3. Solving the XOR problem, by using techniques of the Theory of Numbers

The XOR problem can be treated with success, in accordance with the complex-number [4] and quantum bit [5] solutions, by applying a threshold function with a modulus operant, whereas n is the dividend, m is the divisor and r is the residual. The n, k, m, r are integers (n, k, m, r

$\in \mathbb{Z}$) and the factor m is not zero, m≠0. The Euclidean Algorithm with the modulus operant is expressed [10]:

$$n = (k \cdot m + r) \bmod m \equiv r(\bmod m) \Leftrightarrow n \equiv r(\bmod m) \qquad (4)$$

In order to apply the modulus operant in the ANN simple perceptron model, the domain manifolds of n, k, and r parameters are expanded to real number ($\square$) manifolds. Parameter m is excluded from this expansion, since it is correlated with the discrete number of a neuron's inputs (synapses). Hence, it is defined that n, k, r parameters are real numbers and m is a non zero integer, so as relation (5) to be defined, where the divisor enumerates the number of the signal inputs (m=2). In this paper the symbolism "f(S)=(S)mod2" denotes that the modulus operand is being applied after the S sum calculation and the "f(S)≡S(mod2)" that the parameter S stands for the calculation result after the modulus operant application. Function (5) can transform relation (3)'s half plans into two saw-shaped plan areas (P1´ and P2´), as shown at Figure 2. The modulus operant transforms the linear attitude of relation (1) into a periodical linear saw-shaped attitude, so to be possible for non-linear separable cases to be solved by simple perceptron models.

$$f(S) = \left(\sum_{i=1}^{2} s_i w_i\right) \bmod m \xrightarrow{m=2} (S(s_1, s_2)) \bmod 2 \qquad (5)$$

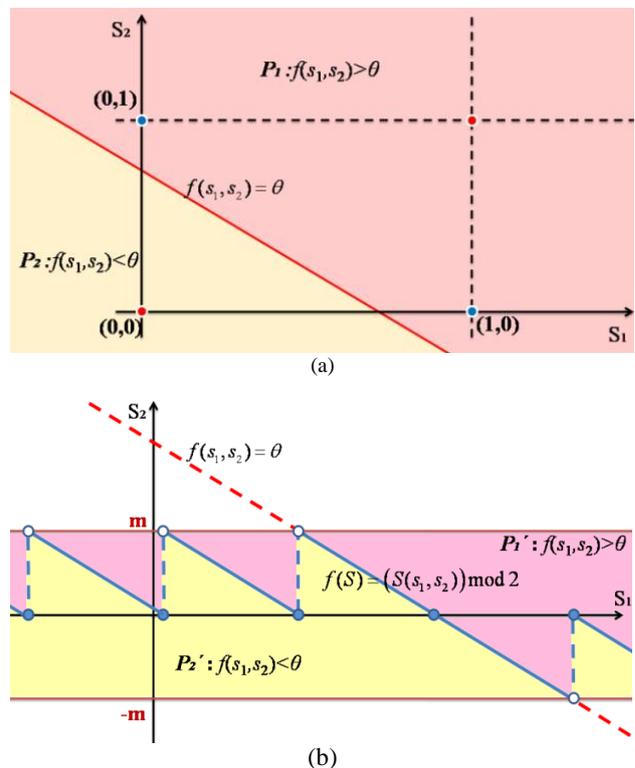**Fig. 2.** (a) The line $s_1w_1+s_2w_2 = \theta$ separates the plan into two half plans. (b) The saw-shaped transformation that the modulus operant causes to relation (3).

By setting $(s_1, s_2) = (0,0), (0,1), (1,0)$ and $(1,1)$ and considering $(w_1, w_2) = (1,1)$ a solution to the XOR problem is achieved. Although the XOR problem is not defined for more than 2 inputs, the modulus neuron operant can be generalized to an *n*-input perceptron, by dividing *S* with the inputs number of the neuron. This generalization is hence

utilized for inequalities measurement and is displayed at relation (6), where *S* is defined in relation (1), the *modulus operant* is defined in relation (4), *v* denotes the neuron feature as a graph vertex and $deg_+(v)$ [11] declares the number of the input synapses of the neuron. The above generalization opines definitely at the equality case and in analogy the XNOR consideration [6-7] at inequality.

$$f(S) = S \bmod(deg_+(v)), \ f(S) \in [0,n] \tag{6}$$

4. The simple perceptron as an inequalities index in Regional Science

Regional Science belongs to modern sciences [12] and it utilizes scientific material and conclusions from other Sciences. The "*Gravity Econometric Models*" [13-15] and the "*Entropy Measures*" [16-17] suggest two characteristic research fields on Regional Science, where the proffer of Physics is determinative. Regarding the research of inequalities in Regional and Urban Economics the Theil index constitutes a simple and effective quantitative tool, which introduced by Henri (Hans) Theil in 1967 [9]. The index appertains in the entropy measures family [17], a family class which has been developed by the utilization of Thermodynamics in Physics, and is described in relation (7). In relation (7) x_i is the arithmetic value of the observation *i* for the variation *X* (i=1,…,n), *lnx* is the napierian logarithmic function and *μ* the mean of the *n* observations x_i (i=1,…,n). Theil index ranges into the interval $0 \leq T \leq \ln n$ [18],[9].

$$T = \frac{1}{n}\sum_{i=1}^{n} \frac{x_i}{\mu} \cdot \ln\frac{x_i}{\mu} \tag{7}$$

The simple perceptron pattern with the modulus threshold function can also operate as an inequalities measure. Let's consider the general case, described by relation (6). In order to conform to the binary structure of the XOR table (Table 1), the input signal values have to be normalized into the interval [-1,1] and weights vector *W* has to be $W=[w_1,w_2,...,w_n]=[1,1,...,1]$. This normalization is held by dividing the input signals with a quantity s_m. Three cases are descried:

First, the input signal vector consists of positive values. Then the vertex $S = [s_1,s_2,...,s_n]$ is divided by the maximum input value $s_m=\max\{s_i:i=1,2,...,n\}$ and turns to $S' = [\frac{s_1}{s_m}$, $\frac{s_2}{s_m},..., \frac{s_n}{s_m}]$, whereas $\frac{s_i}{s_m} \in [0,1] \ \forall \ i=1,2,...,n$. Second, the input signal vector consists of negative values. Then the s_m value is defined as the minimum signal value $s_m=\min\{s_i:i=1,2,...,n\}$ and the vector S' components belongs to interval [-1,0]. Finally, in the case the input signal vector consists of mixed (both negative and positive) values the s_m value is the function (8):

$$s_m = \begin{cases} \max\{s_i, i=1,...,n\}, if\ \mu > 0 \\ \max\{|\max\{s_i\}, \min\{s_i\}|, i=1,...,n\}, if\ \mu = 0 \\ \min\{s_i, i=1,...,n\}, if\ \mu < 0 \end{cases} \tag{8}$$

where $\mu = \sum_{i=1}^{n} \frac{s_i}{n}$ and $|x|$ is the absolute operator of a real value *x*. The output interval for relation (8) is [-1,1]. After defining the above limitations, the simple modulus perceptron architecture can form the index of relation (9), nominated by the authors "*modulus perceptron inequalities index*" (MPII):

$$MPII = \frac{(\sum_{i=1}^{n} \frac{s_i}{s_m}) \bmod n}{n} \tag{9}$$

In the equality case, where every $s_i=a \in \square$, $\forall$ i=1,2,…,k-1,k+1,…,n , both the MPII and Theil index turns zero. In the complete inequality case, where the total of signals is collected into one value (let be s_k) and all the others equal to zero ($s_i=0 \ \forall \ i=1,2,…,k-1,k+1,…,n$) the MPII equals to 1/n, where Theil index equals to *lnn*. In the case that the signal values of the sample are almost equal the MPII calculation gives monad where the Theil index equals to zero. It can be written that: MPII $\in$ {0}U[1/n,1). The summarization comparison of the two indices is shown at Table 2.

Table 2. Theil index's and MPII's range of values

	Equality case	Inequality case	Slight ineq. inequalities
Theil index	0	ln*n*	$\to 0$
MPII	0	1/n	$\to 1$

5. MPII's comparative simulation and application on Greek unemployment inequalities

The present section studies the MPII's behavior in comparison with the Theil index, for the same set of input signal vectors (patterns). Both Theil index and MPII were simulated by using the software package Microsoft Excel 2007. In the created application, 20 simulation tables were constructed, each for a separate random input signal vector (pattern). Each vector pattern consists of 52 input signal values (chosen to be equal in amount with the number of Greek Prefectures). The input signal values were generated randomly by the software. In order to broaden (at will) the software's random values interval [0,1], a coefficient was set that multiplies each random value. The two indices were calculated over the same random input values, so for every chosen coefficient two output vectors, one for Theil index ($t_1, t_2, …, t_{20}$) and one for MPII ($r_1, r_2, …, r_{20}$), to be resulted. The results were given as percentages from the maximum value, by dividing each index with the maximum possible value ($\frac{MPII}{\max MPII} = \frac{MPII}{1/n}$, $\frac{Theil}{\max Theil} = \frac{Theil}{\ln n}$).

The result vectors ($t_1, t_2,…, t_{20}$) and ($r_1, r_2,…, r_{20}$) were represented graphically in a comparative graph. Initially, the Theil index and the MPII were compared, but the graphical results appeared to lie under an almost reversed relation. That observation agreed with the conclusion made previously that the generalized XOR modulus perceptron structure opines definitely as an equality index and the XNOR gate structure as an equality, so the Reversed MPII

(RMPII=MPII^{-1}) was considered, in order to "reverse" MPII's measurement utility and to obtain more obvious results. The comparative procedure conducted by setting values sequentially for the interval range coefficient, such as 1, 5, 100, 1000, -1, -5, -100 and -1000. The results are shown at Appendix *A*. As it can be distinguished, the percentages RMPII curve behavior is almost identical to the Theil index's, regarding the way that the value alternations are conserved. RMII percentages present a transposition to higher values, so it can be concluded that the non-normalized values of the index are smoother than Theil's.

Secondly, in order to cover the case of the mixed-value signal vectors (with both positive and negative signal values) another simulation attempt occurred. That simulation lied on the same philosophy with the ordinary simulation, with the only difference that the random values were chosen alternately positive and negative and the real indices values (not percentage) are presented. The results (coefficient=10 chosen) on that case (Figure 3) show that in the mixed-value case *the Theil index is unable to opine* for the inequalities status, in opposition with the RMPII that is capable. The Theil index's inability is ought to the presence of the logarithmic function of relation (7), where the expression *lnx* is not defined for negative values (*x*<0). In the case that the input signal vector consists of completely negative values (described by the ordinary simulation by choosing a negative coefficient) the Theil index can conclude to a numerical result, because the negative values will be converted to positive by their division by the also negative mean value *μ* at the Theil index's relation (7).
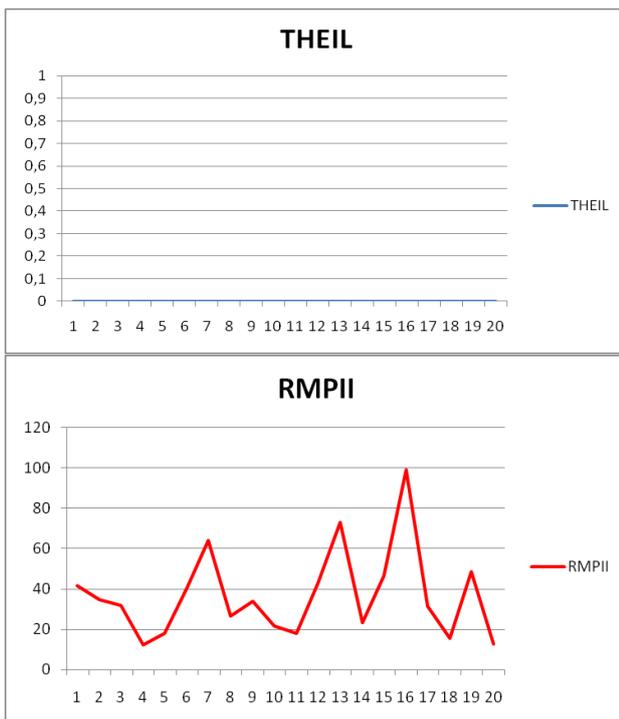

Fig. 3. Graphical behavior of the two indexes on mixed-value cases (coefficient=10).

Finally, an application of Theil index and RMPII took place on real data of unemployment in Greece. The input data regard the annual amount of unemployed people of each Greek Prefecture during a 26 years' period (1965-2006) [19]. Each Prefecture consists of an input signal pattern with 26 signal values (of the 26 available years of data). The

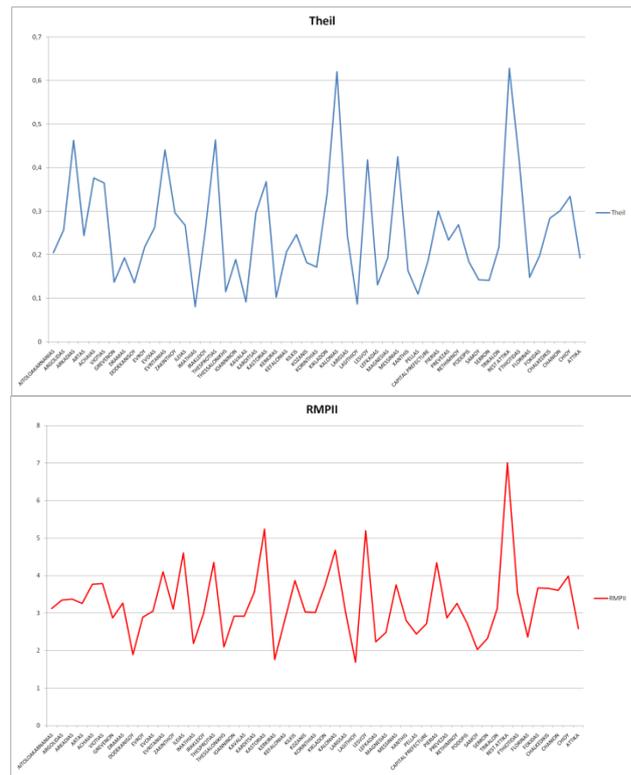Theil index and MPII which were calculated for each Greek Prefecture and the 52 results are presented at Figure 4.

Fig. 4. Graphical behavior of the two indexes on real unemployment data in Greece

6. Conclusions

The XOR gate architecture in the Digital Electronics Theory has structural and functional similarities to the simple perceptron model in ANN. Respectively with the XOR and XNOR electronic gate utility, which refers to the equal and unequal electronic signals detection, the simple perceptron can also be used as an inequalities detector, under the condition that it can solve the XOR problem, without the attachment of hidden layers into its structure. By considering a threshold function with a modulus operator, as in relation (6), the simple ANN perceptron pattern can solve the non-linear separable XOR problem. The proposed ANN model (nominated by the authors as *modulus perceptron*) is simpler than complex or quantum calculations, which suggest some solutions of the XOR problem in modern ANN research. The XOR architecture in the modulus ANN simple perceptron pattern can be formed to an index that can operate as a measure of inequalities in a similar way with the Theil index in Regional Econometrics. In the present paper the XOR problem solution achieved for an ANN simple perceptron, with reference to Number Theory in Mathematics, and a new inequalities index was introduced, MPII. The MPII and the Reversed MPII were examined in their behavior in comparison with the Theil index, by a simulation that took place with the use of Microsoft Excel 2007 accounting software package. The Theil index was compared with the proposed MPII (and in particular with the reversed MPII due to theoretical assumptions) and the results indicated that the new proposed MPII index operates sufficiently as an inequalities index, by the time it conserves the Theil's index slope and attitude alternations. Moreover,

the proposed MPII seems to be more capable that Theil index, since it can operate in a wider range of cases due to the logarithmic deficiency of the second. In the case that the set of input signals consists of mixed (both positive and negative) v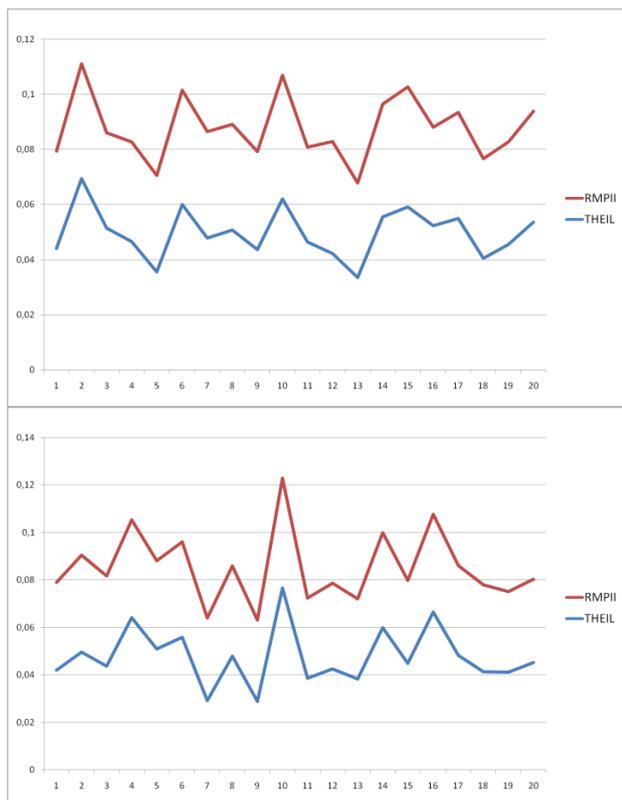alues the MPII can provide estimation for the inequalities status in opposition with the Theil index. The proposed inequalities index is considered to be very useful measure on a total aspect.
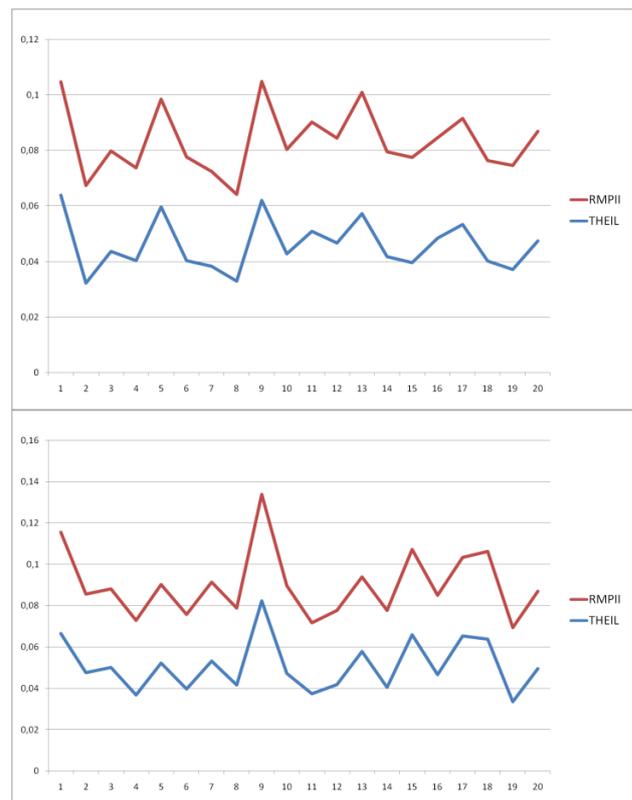
_____

References

1. P. Argyrakis, Neural Networks and Applications, Patra, Greece, Hellenic Open University Publications, (2001).
2. A. Rogers, J. Keating, R. Shorten, D. Heffernan, Chaotic maps and pattern recognition – the XOR problem, Chaos, Solitons and Fractals, 14, 57–70 (2002).
3. E. Rumelhart, E. Hinton, J. Williams, Parallel distributed processing (vol. 1), Cambridge, MIT Press Publications (1986).
4. T. Nitta, Solving the XOR problem and the detection of symmetry using a single complex-valued neuron, Neural Networks, 16, 1101 (2003).
5. M. Maeda, Qubit neuron according to quantum circuit for XOR problem, Applied Mathematics and Computation, 185, 1015 (2007).
6. B. Paton, Fundamentals of Digital Electronics, Austin, National Instruments Corporation (1998).
7. J. Crowe, B. Gill, Introduction to Digital Electronics, Oxford, Newnes Publications (2001).
8. K. Clements, Henri Theil, 1924–2000, International Journal of Forecasting, 17, 141 (2001).
9. S. Polyzos, Regional Development, Athens, Greece, Kritiki Publications (2011).
10. T. Apostol, Introduction to Analytic Number Theory, New York, Springer-Verlag Publications (1976).
11. R. Diestel, Graph Theory, Third Edition, Heidelberg, Springer-Verlag Publications (2005).
12. R. Capello, Regional Economics, NY, Editions Routledge (2007).
13. R. Feenstra, J. Markusen, A. Rose, Using the gravity model to differentiate among alternative theories of trade, Canadian Journal of Economics, 34 (2), 430 (2001).
14. J. Lewer, H. Van den Berg, A gravity model of immigration, Economics Letters, 99, 164 (2008).
15. M. Yu, Trade, democracy, and the gravity equation”, Journal of Development Economics, 91, 289 (2010).
16. A. Golan, Information and entropy econometrics—editor's view, Journal of Econometrics (Guest editorial), 107, 1 (2002).
17. M. Stephane, F. Seyte, M. Terraza, Decomposition of Gini and the generalized entropy inequality measures, Economics Bulletin, 4 (7), 1 (2003).
18. J. Foster, An Axiomatic Characterization of the Theil Measure of Income Inequality, Journal of Economic Theory, 31, 105 (1983).
19. Epilogi, The Greek Prefectures, Athens, Greece (2006).
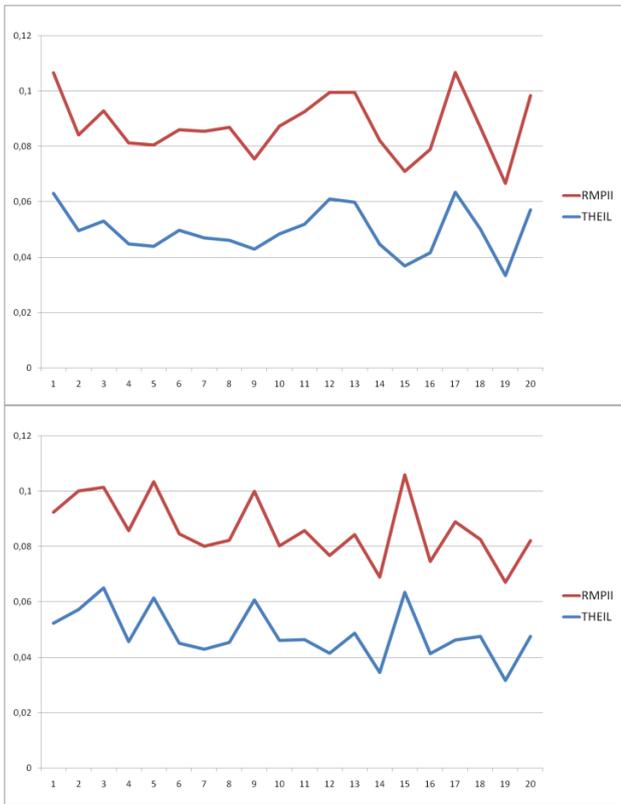
Appendix A
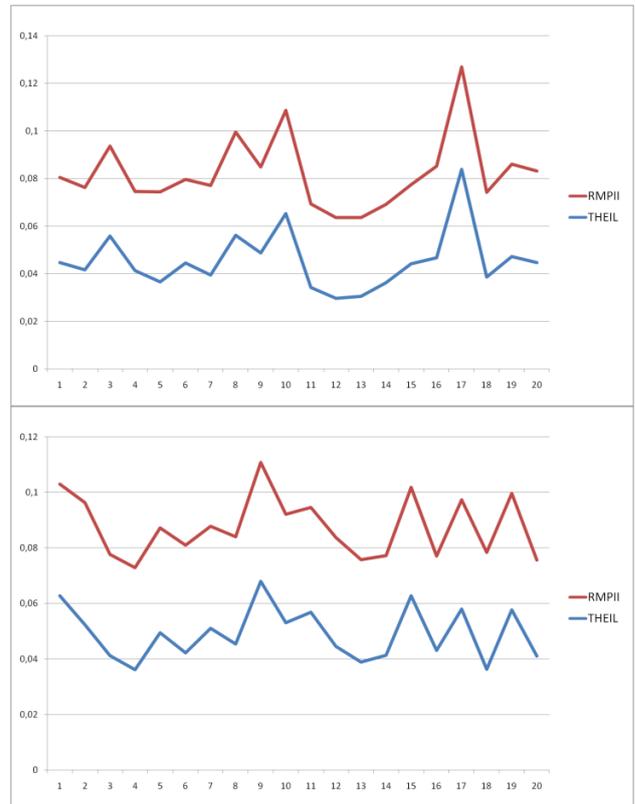
Comparative graphs of Theil index and RMPII

Coefficients: 1 and 5

Coefficients: 100 and 1000

Coefficients: -1 and -5

Coefficients: -100 and -1000