Literature Review

Running Head: LEARNING INTERNAL REPRESENTATIONS BY ERROR PROPAGATION.

Student Name: Youwen Zhang

Student Number: 12769687

**Learning Internal Representation by Error Propagation.**

**Content.**

This paper is a nitty gritty presentation of the perceptron learning algorithm for internalizing the perfect sets of combinations for the arbitrary networks. The arbitrary network work with rule of generalized delta (Grudin & Serge, 2015). Delta rule is a scheme that is used in implementation of gradient descent technique in figuring out weights that minimizes the sum squared error performance of the system. The key contribution of this paper is on the vital procedure called error propagation. Its relevance is on the fact that; gradient can be determined by specific units of the network relying on locally collected and available processed data. For that reason, empirical value addition of this paper is showing that the local minima problem does not have a significant in the application of gradient descent.

Statistically, error propagation or also referred to as propagation of uncertainty. It is the effect of parameter uncertainties on the variables that measures the dispersion of range of measured values. Due to measurement limitation, values of experimental measurement end up with uncertainties; which propagate as a result of variables combination in a given model network. Uncertainty can be represented in a number of ways; it can be relative error, absolute error and significantly, as a quantitative representative; standard deviation.

Error propagation concept was adopted in Internal Representation. It gave a clear understanding of a network that is simple two-layer. This network entails a set of input model that reaches input layer and mapped directly to a collection of output pattern at the output layer, and along with it has unrecognized units. Unrecognized unit emerge since most of the mapping focuses on input and output units and as a result lacking internal representation.

The networks developed by error propagation concepts have been useful in a number of application. The key attribute of this kind of network is that there is similar mapping of both input and output patterns. This characteristic is the reason behind the creation of reasonable generalization and performance that no other network has ever presented before. The resemblances in patterns is determined by their overlaps by the PDP system. Such overlaps are determined external part of learning system by unknown factor.

**Innovation.**

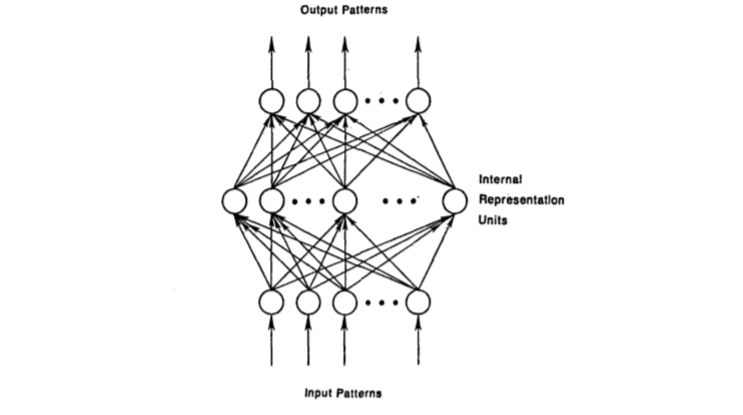
In the research process of this paper, it is catalyzed by the desire to reduce the inability of the system to point out certain mappings from input to output from similar input patterns leading to output patterns. The externally available representation similarities structural similarity of input is different to those patterns of output. Any network that doesn’t have hidden units, bares the risk of being unable to perform relevant mappings. For instance, this weakness is witnessed in exclusive-or; indicated in the table below.

|  |  |
| --- | --- |
| **INPUT PATTERN** | **OUTPUT PATTERN** |
| 00 | 0 |
| 01 | 1 |
| 10 | 1 |
| 11 | 0 |

It is evident that input patterns that overlap least end up producing identical output values. This is a kind of problems that can be solved by networks that doesn’t have hidden units to create specific internal representation of input patterns. However, when the input pattern takes in a third input of a value 1, when the first two values are 1; a two-layer system is a position to solve the problem. See the table below.

|  |  |
| --- | --- |
| **INPUT PATTERNS** | **OUTPUT PATTERNS.** |
| 000 | 0 |
| 010 | 1 |
| 100 | 1 |
| 111 | 0 |

There was a clear analysis of the conditions which were drafted by Minsky and Papert (1969, cited in Rumelhar & Hinton & Williams 1985) under which this kind of system can carry out mapping. But the concept gathered by this paper from studying the conditions, it is clear that if there are a number of layers made up of simple perceptron of hidden units together with original input pattern increased in number, there exist internal presentation of input pattern in the hidden units. This hidden units have similarity of patterns in the hidden units that can support any given mapping ranging from input to output units. See the diagram bellow.



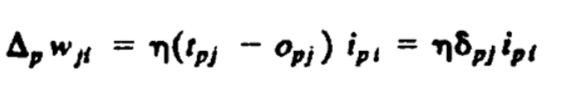
This is evidence to show that right connection from the units of input to a larger set of hidden units is critical condition to always have a representation (Rumelhar & Hinton & Williams 1985). It is also possible for a multilayer network; the input information is withdrawn into an internal representation which generates outputs and not the beginning pattern. If there are enough hidden units, input pattern will always be encoded. This is the reason for the generation of output pattern from any given input pattern, thus providing solution to XOR problem.

**Technical quality.**

This paper has an average technical quality based on the classification comparison done by a number of scholars about error propagation. The problem which was presented by Minsky and Papert (1969, cited in Rumelhar & Hinton & Williams 1985) has been tried to be solved by one major approaches. This approach is perceptron convergence procedure ((Widrow & Hoff 1960, cited in Rumelhar & Hinton & Williams 1985). This is a powerful model when dealing with hidden units which as three reactions. One reaction is that of competitive learning, where unsupervised rules are implemented to bring out the hidden unit. As vibrant it seems but it is not efficient due to lack of external force that makes sure that the developed hidden unit is appropriate for the needed mapping.

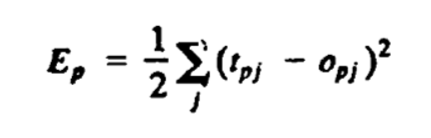
The weakness in the first response gave birth to the second response which is simply to answer an internal representation. This is an interactive activation model of perception model. And the final approach is driven from the attempt to build up a learning criteria capable of internalizing internal representation enough for handling task at hand (Grudin & Serge, 2015). From Boltzmann machines concept, it is evident the application of stochastic units, which requires two different phases in attaining equilibrium but limited to symmetric network. Another approach was developed by Barto (cited in Rumelhar & Hinton & Williams 1985) and a number of his colleagues by which they also employed the stochastic units (Barto & Anandan 1985, cited in Rumelhar & Hinton & Williams 1985). However, in this paper the key approach is that called general delta rule. To understand the logic behind general delta rule, we will start by deriving this rule.

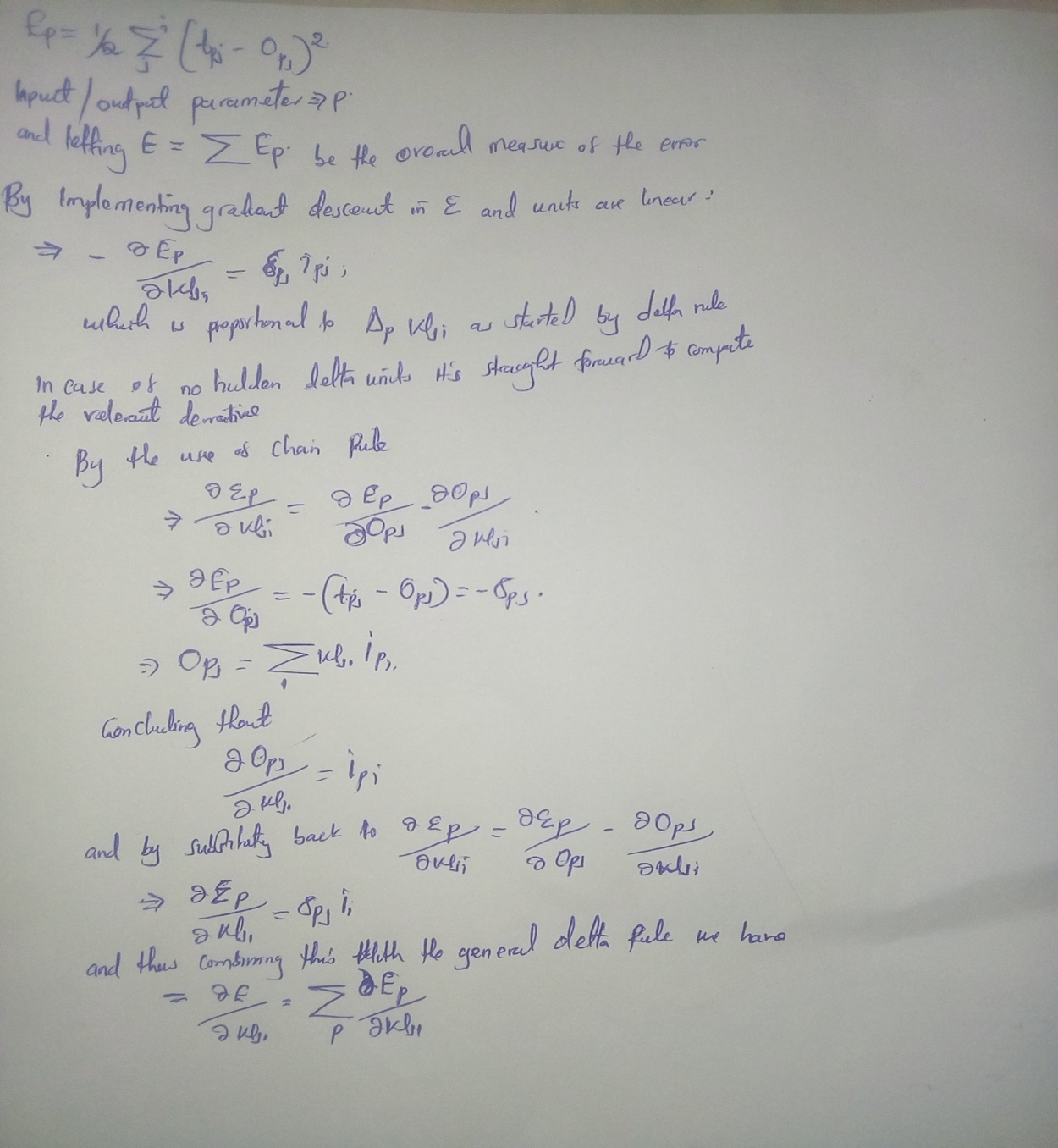
This paper proposed a learning a procedure that presents a set of input-output patterns. The theory behind it is that firstly the system used input vector to generate its output vector and compares it to the desired output. In the scenario of similarities, learning has to take place; contrary to which weights have to be shift with the purpose of minimizing the difference. Scenarios where there is no hidden units’ standard delta is generated. This model of changing weights follows the presentation of input/output pair **p** given by the formulae;



* Where tj= is the target input for jth component of the output pattern for pattern p,
* Opj= is the jth element of the actual output pattern produced by the presentation of input pattern p,
* Ipi= is the value of the ith element of the input pattern 8,j = tj - opj,
* Awj, is the change to be made to the weight from the i th to the j th unit following presentation of pattern p.

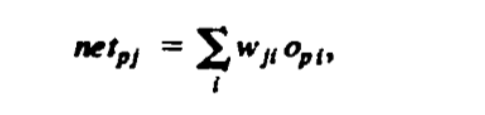
The general delta rule and gradient descent have a number of derivative method. Linear approach reduces the squares difference between actual and desired output added up over the output unit and all pairs of input-output vectors (Geoffroy & Norman 2012). These is achievable by first showing that the derivative of error measure with respect to each weight. Which at the end is proportional to the weight change as stated by delta rule?



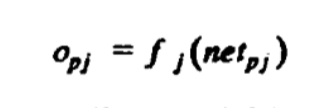


The calculation above concludes that the net change in wj, after one complete cycle of pattern presentations is proportional to this derivative and that the delta rule implements a gradient descent in E. In addition, this paper has shown how essential delta rule is in implementation to gradient descent. More significantly on sum-squared error for linear activation functions. The theoretical contribution of this paper is to show there are more efficient and effective computational derivatives. For the empirical contribution is to point out that the fatal problem of local minima is insignificant in a wide range of learning task. A generalized delta rule plays major role in arbitrary networks.

Since hidden units that has linear activation relevance does not have advantage, the analysis zeros down in setting semi-linear activation function. A semi linear activation function does generate output unit which is has a differentiable function at the total input.

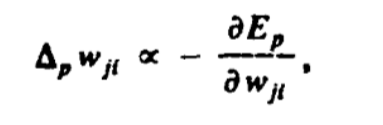


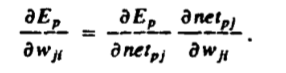
where oj = ij if unit i is an input unit. Which makes the equation bellow a semi linear activation function.

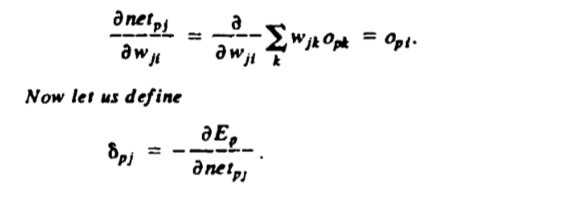


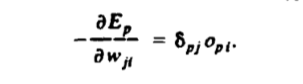
And that the function **f** is differentiable.

For that case, general delta rule will only work when the network is made up of units that have semi linear activation function. Thus, in order to acquire accurate generalized delta rule, the formulae have to be implemented;





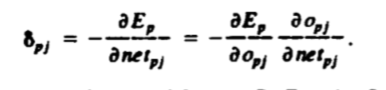




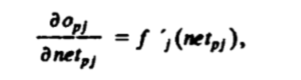
This implies that to implement gradient descent in E, the weight has to be shifted according to the formulae bellow.



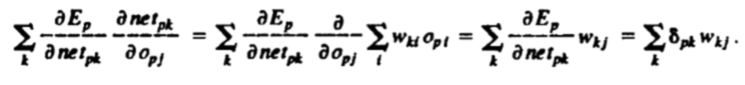
Just as in standard delta, implementation of chain rule is significant in this part by changing the output as a function of changes in the input. Thus, we have



And computing for the second factor;



for any output unit Uj, If Uj is not an output unit we end up using the chain rule to write the final output pattern.



For that reason, it is very clear that this paper is of good quality. The result on this paper can be replicated by other promising scholars and produce similar results or even produce a better research results (Geoffroy & Norman 2012). Comparing this work to earlier done and published work, it is evident that content was collected from enough different sources of earlier done experiments and accurate results.

**Application and X-factor.**

The application domain of general delta rule to learning internal presentation by error propagation is more than appropriate. Its application is delivered effectively by duo-phase. During the first phase; there is presentation and propagation of the input through the network in order to generate output value of each unit. The generated output is then compared to the desired unit, producing signals of an error for each output unit. The second face is built up from backward pass through network (GGeoffroy & Norman 2012). Along with its signal is passed to each unit within the network and relevant weight changes are made. Second phase creates a room recursive computations by the formulae derived above. This phase propagates errors back one layer and the same process is conducted for each and every layer. Just like that a method for finding weights in a feedforward is generated by gradient descent method.

This paper also outlined some key observation of this technique. Weights doesn’t have to be varied, once the weight is in the network it can be automatically fixed and along with its errors get propagated. Output units too has no restrictions whatsoever, since they don’t have to really receive inputs from the previous layers, and for that reason they end up with two kinds of errors (that error from the direct comparison with the target and that passed through the other output units whose activation it affects). The concept behind error propagation by general delta rule finally through this paper gave birth to a learning procedure which in principle have revolutionized weight sets to arbitrary mapping from input to output (Geoffroy & Norman 2012). However, just like any other innovation, challenges and limitation always exist. For this case, the problem of local maxima and minima and bigger one is that of knowing how long it will take the system to learn. This calls for more research and advancement to be made on this paper.

This paper has the potential to spark a class discursion; one discursion question that will bring the dawn of class discursion is; what hidden units the system actually develops in the solution of particular problems? This question inquires to what type of internal presentation the system creates. Since this paper hasn’t looked for that answer, it will bring a discursion spirit in class basing on the number of simulations that has been carried out by this paper containing optimistic expectations (Rumelhar & Hinton & Williams 1985). The most interesting of this article is that fact that, it is amazing how weights doesn’t have to be varied, once the weight is in the network it can be automatically fixed and along with its errors get propagated. Output units too has no restrictions whatsoever, since they don’t have to really receive inputs from the previous layers. This is a concept which will revolutionize industries that will implement this concept.

More importantly, the error propagation concept reinforced by general delta rule provided the solution to exclusive-OR problem. This was possible by running the XOR problem as many times as possible and surprisingly, the system solved the problem. The solution was reached after a sweep amounting to 558 through four stimulus patterns with a learning rate 0.5. Another amazing discovery is that of finding solution to simple addition problem. It is interesting since we figured out the condition under which local minima can be found and also how to avoid it

**Presentation.**

This paper had an impressive quality in presentation. The concept is incorporated in both simple and complex mode (Rumelhar & Hinton & Williams 1985). For the simplification bit of it is done by tables with elaborated titles and variables. The complex part is simplified by simple and straight forward formulae to make sure that the reader of the report has humble and fun time reading it. Clear compilation of formulae, theories behind those formulae and tables gave it a smooth flow of content from one paragraph to the next, page after page. The two concepts; that of error propagation and general delta rule are well married to each other clear.

**Conclusion.**

In a nutshell, given the fact that the theoretical part of this report does not guarantee solution for all solvable problems. However, the analysis results have shown that, practically error propagation criteria provide solution in almost every scenario presented. And by this paper, have provided answers to Minsky and Papert's (1969, cited in Rumelhar & Hinton & Williams 1985) challenge by providing a sufficient learning results. These results are powerful enough criticize their pessimistic believe about learning multilayer machine.

**Reference.**

Amy Geoffroy and Donald A. Norman 2012, ‘Ease of Tapping the Fingers in a Sequence Depends on the Mental Encoding’, March 2012.

David E. Rumelhart, Geoffrey E. Hinton and Ronald J. Williams 1985, ‘Learning Internal I Representations by Error Propagation’, *D. E. Rumelhart & J. L. McClelland (Eds.), Parallel Distributed Processing: Explorations in the Microstructure of Cognition*, September 1985, Vol.1.

Jonathan T. Grudin and Serge Larochelle 2015, ‘Digraph Frequency Effects in Skilled Typing’, February 2015.