

**The Frankfurt University of Applied Sciences**

**A Project Report**

**On**

**Artificial Neural Networks**

**Implementation of Delta Rule Learning in**

**System Identification.**

***Submitted for the partial fulfillment of the degree***

***in Masters of Information Technology***

***by***

**Bala Vinay Kumar Pappu**

**under the guidance of**

**Mr. Damir Dobric**

**Frankfurt University of Applied Sciences**

Faculty of Computer Science and Engineering  
Nibelungenplatz 1  
60318 Frankfurt am Main  
**Germany**

Table of Contents

[1. Overview 3](#_Toc487417549)

[1.1. Introduction: 3](#_Toc487417550)

[1.2. Adaptive Filters using Delta Rule or LMS (least-mean-square): 4](#_Toc487417551)

[1.3. Filters with Finite Impulse Response (FIR Filters): 4](#_Toc487417552)

[2. Scope of the project 5](#_Toc487417553)

[a. To develop the C# code to implement the system identification using supervised learning Delta rule algorithm. 5](#_Toc487417554)

[b. To develop this code C# in .NET Core and implement it in Learning API with Unit Test. 5](#_Toc487417555)

[3. Project Description: 5](#_Toc487417556)

[3.1 Delta Rule Algorithm: 6](#_Toc487417557)

[3.2 System Identification: 6](#_Toc487417558)

[3.3 Implementation: 7](#_Toc487417559)

[3.3.1 Training: 7](#_Toc487417560)

[3.3.2 Predicting: 7](#_Toc487417561)

[3.3.3 Results: 8](#_Toc487417562)

[3.3.4 Unit Testing: 8](#_Toc487417563)

[3.3.5 Test Results: 9](#_Toc487417564)

[4. References: 9](#_Toc487417565)

[5. Conclusion: 9](#_Toc487417566)

1. **Overview**
   1. **Introduction:**

**Machine learning** is a method of data analysis that automates analytical model

building. Using algorithms that iteratively learn from data, machinelearningallows computers to find hidden insights without being explicitly programmed where to look.

There are two types of Learning.

* + 1. **Supervised Learning**

It is called supervised learning because the process of an algorithm learning from the training dataset can be thought of as a teacher supervising the learning process. We know the correct answers; the algorithm iteratively makes predictions on the training data and is corrected by the teacher. Learning stops when the algorithm achieves an acceptable level of performance.

* + 1. **Unsupervised Learning**

Unsupervised learning is where you only have input data (X) and no corresponding output variables. The goal for unsupervised learning is to model the underlying structure or distribution in the data in order to learn more about the data.

Unsupervised learning problems can be further grouped into clustering and association problems.

1. **Clustering**: A clustering problem is where you want to discover the inherent groupings in the data, such as grouping customers by purchasing behaviour, for example K-Means
2. **Association**:  An association rule learning problem is where you want to discover rules that describe large portions of your data, for example Apriori algorithm

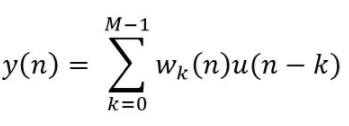
Among neural network models, the Delta rule learning is Supervised learning algorithm. developed by Widrow and Hoff, the delta rule, also called the Least Mean Square (LMS) method, is one of the most commonly used learning rules. The Delta rule is also called *LMS* (*least-mean-square*) *rule* or *Widrow-Hoff rule*.

* 1. **Adaptive Filters using Delta Rule or *LMS* (*least-mean-square*):**

Adaptive linear filters are linear dynamical system with variable or adaptive structure and parameters. They have the property to modify the values of their parameters, i.e. their transfer function, duri.ng the processing of the input signal, in order to generate signal at the output which is without undesired components, degradation, noise and interference signals. The goal of the adaptation is to adjust the characteristics of the filter through an interaction with the environment in order to reach the desired values. The operation of adaptive filters is based on the estimation of the statistical properties of the signal in its environment, while modifying the value of its parameters in order to minimize a certain criterion function. The criterion function may be determined in a number of ways, depending on the particular purpose of the adaptive filter, but usually it is a function of some reference signal. The reference signal may be defined as the desired response of the adaptive filter, and in that case the role of the adaptive algorithm is to adjust the parameters of the adaptive filter in such a way to minimize the error signal, which represents the difference between the signal at the output of the adaptive filter and the reference signal.

* 1. **Filters with Finite Impulse Response (FIR Filters):**

One of the ways to overcome the lack of potential instability in an IIR digital filter is to design a filter with zeroes only, which in comparison to a recursive IIR 34 2 Adaptive Filtering structure has only the direct branch or a non-recursive structure. The memory of such filters is limited, i.e. their impulse response is equal to zero outside some limited time interval, and because of that they are denoted as the filters with finite impulse response (FIR). In other words, a transient process in such a system, which is initiated immediately after bringing excitation and which lasts until the output signal assumes a stationary value, i.e. the system enters equilibrium or steady state, has a finite duration. The number of the delay elements, M, denotes the order of the filter and the duration of its impulse response. The output signal of the filter, y(p), is determined by the values of the parameters {Wk} and it represents a linear combination of the actual and the previous samples of the input signal, x(t). These parameters are the object of estimation in an adaptive process, i.e. they vary with time index k. In this manner, the filter output signal is defined by the linear difference equation.



1. **Scope of the project**
2. To develop the C# code to implement the system identification using supervised learning Delta rule algorithm.

b. To develop this code C# in .NET Core and implement it in Learning API with Unit Test.

1. **Project Description:**

The Delta Rule or least-mean-square (LMS) algorithm is a linear adaptive filtering algorithm that consists of two basic processes:

1. A filtering process, which involves (a) computing the output of a transversal filter produced by a set of tap inputs, and (b) generating an estimation error by comparing this output to a desired response.
2. An adaptive process which involves the automatic adjustment of the tap weights of the filter in accordance with the estimation error.

Adaptive structures have been used for different applications in adaptive filtering such as:

* Noise cancellation
* System identification
* Adaptive predictor
* Equalization
* Inverse modeling
* Interference cancellation.

In this project, we are going to implement our example in system identification.

* 1. **Delta Rule Algorithm:**

For a given input vector, the output vector is compared to the correct answer. If the difference is zero, no learning takes place; otherwise, the weights are adjusted to reduce this difference. The change in weight from ui to uj is given by: dwij = r\* ai \* ej, where r is the learning rate, ai represents the activation of ui and ej is the difference between the expected output and the actual output of uj. If the set of input patterns form a linearly independent set then arbitrary associations can be learned using the delta rule.

*The gradient descent rule updates the weights after calculating the whole error accumulated from all examples, the incremental version approximates the gradient descent error decrease by updating the weights after each training example.*

Incremental gradient descent is implemented according to the *Delta rule*:



* 1. **System Identification:**

The purpose of system identification is to build mathematical models for dynamical systems from experimental data.

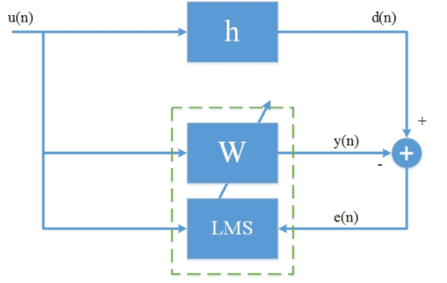


Figure 3.2:Block Diagram of System Identification

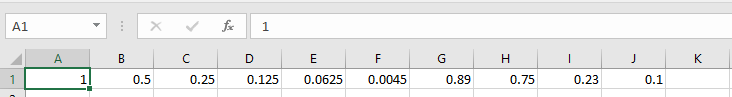
Essential Process involving in System Identification.

Figure 1 shows an adaptive filter structure that can be used for system identification or modeling. The same input u is to an unknown system in parallel with an adaptive filter. The error signal e is the difference between the response of the unknown system d and the response of adaptive filter y. The error signal fed back to the adaptive filter and is used to update the adaptive filter’s coefficients until the overall out y = d. When this happens, the adaptation process is finished, and e approaches zero.

* 1. **Implementation:**

So now it is evident that we have an idea of the entire process of System Identification and we can implement it in C# .NET core.

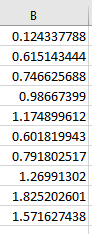
For the implementation, we have

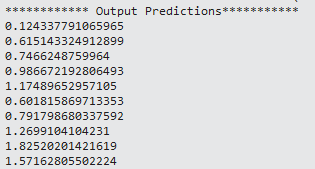
* + 1. **Training**:
  1. Generate some random input data for Delta Rule / LMS filter input.
  2. Assume a system that we are going to estimate it like this: H[M] = 
  3. Build desired signal by convolving the generated random data and assumed H.
  4. Calculate the adaptive filter output y.
  5. Calculate the error signal
  6. Update each LMS filter weights
  7. Repeat the entire adaptive process for the next output sample point.
     1. **Predicting:**

After the Training, the adaptive filter weights are updated with the coefficients of the system model. Now for Predicting with the updated filter model, we are giving some known input samples to the trained filter model and comparing its output with the known filter output.

* + 1. **Results:**

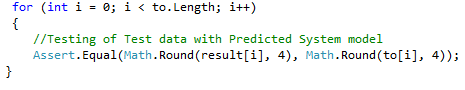
Below are the input and output values in a file. The Input in the file is convolved with the updated filter coefficients.





* + 1. **Unit Testing:**

The Predicted output values and the output of the file is tested with Assert.Equal function and rounded to 4 digits.



Where

to.Length = File output values.

Result – Predicted output values

to – output values from the file.

* + 1. **Test Results:**



1. **References:**

* <https://en.wikipedia.org/wiki/Delta_rule>
* The Adaptive Filters, Simon Haykin
* <http://uhavax.hartford.edu/compsci/neural-networks-delta-rule.html>
* Adaptive Filters , Bernard Widrow ( <http://www-isl.stanford.edu/~widrow/papers/b1971adaptivefilters.pdf>)
* <https://en.wikipedia.org/wiki/System_identification>
* <https://en.wikipedia.org/wiki/Black_box>

1. **Conclusion**:

Successfully developed the C# code and implemented the system identification using supervised learning Delta rule algorithm and also developed this code C# in .NET Core and implemented it in Learning API with Unit Test.