BITS F423T – THESIS
FINAL-SEMESTER PRESENTATION

GUIDED BY
PROF. P.K.THIRUVIKRAMAN
DR. C JAGADESH REDDY
(LV PRASAD EYE INSTITUTE)

SUBMITTED BY
ABISHEK KRISHNAN
2011B5A3511H

OUTLINE

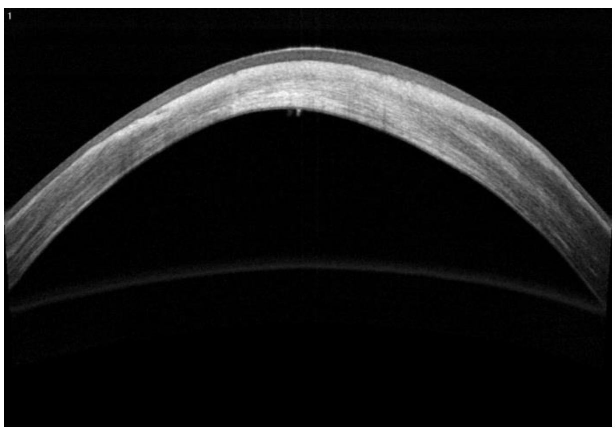
- 1. PROBLEM STATEMENT
- 2. BACKGROUND
- 3. RECAP OF WORK COMPLETED TILL MID-SEM
- 4. RESULTS OF THE THICKNESS OBTAINED

5. PREDICTION OF THE THICKNESS

- 5.A) INTRODUCTION TO NEURAL NETWORK
- 5.B) BACK-PROPAGATION ALGORITHM
- 5.C) LEVENBEVERG-MARQUARDT ALGORITHM
- 5.D) RESULTS OBTAINED
- 5.E) DETERMINATION OF NUMBER OF NEURONS IN THE HIDDEN LAYER – THEORETICAL STUDY

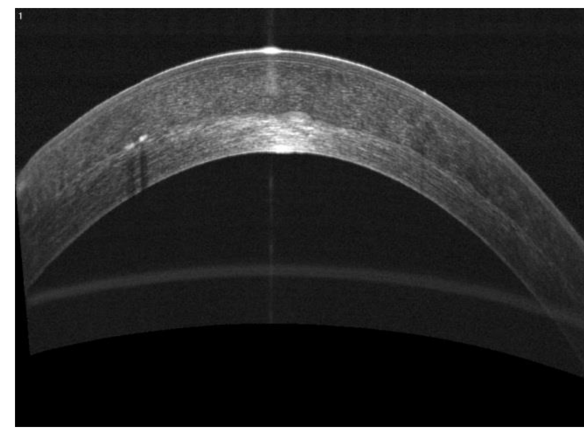
1.PROBLEM STATEMENT

- DETERMINING METRICS FOR RECONSTRUCTION OF CORNEA USING IMAGE PROCESSING AND NEURAL NETWORK MODELING



DAMAGED

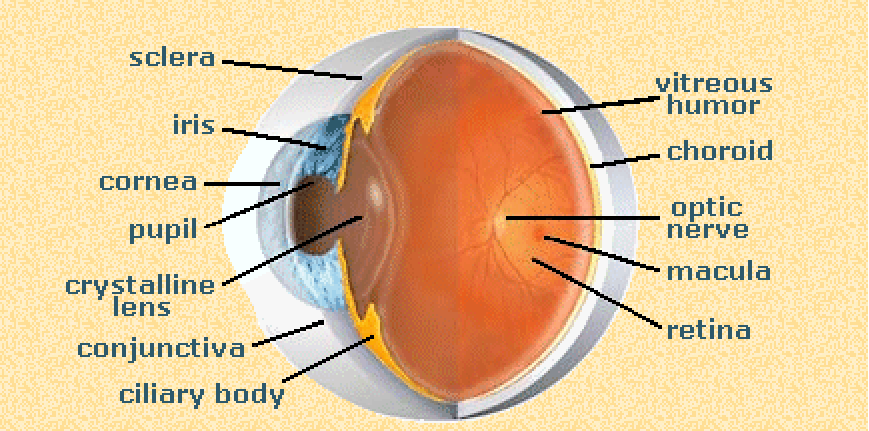
NORMAL

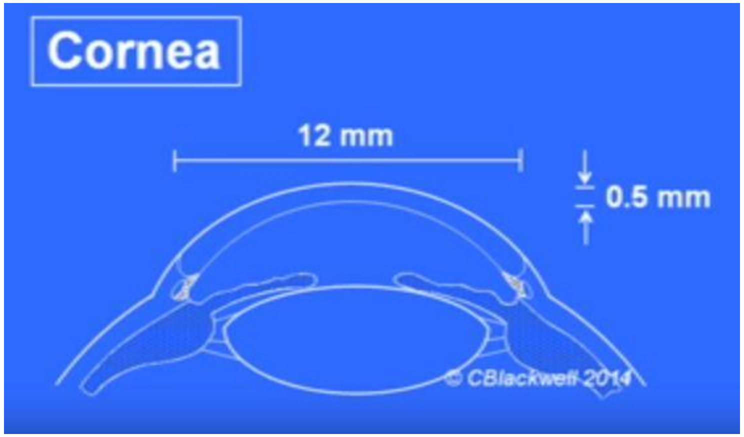


RECONSTRUCTED

BACKGROUND

- STRUCTURE OF EYE AND CORNEA

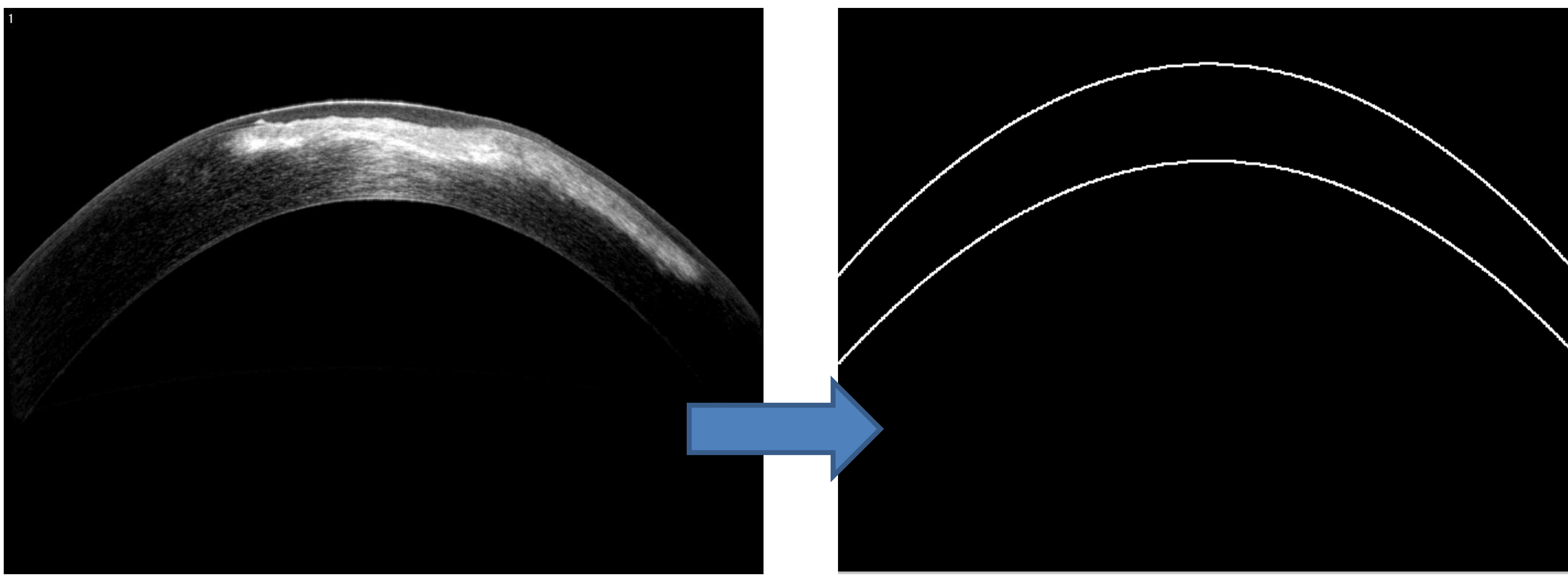




- DAMAGE TO CORNEA – SPHERICAL ABBERATION
- CORNEAL RECONSTRUCTION
- IMAGING TECHNIQUE – SCHEIMPFLUG IMAGING
- (A VARIATION OF MICHELSON INTERFEROMETER)

3. RECAP OF WORK COMPLETED TILL MID-SEM

- EDGE DETECTION AND RECONSTRUCTION OF CORNEAL EDGES
- (Computing Environment Used - MATLAB)



3. RECAP OF WORK COMPLETED TILL MID-SEM (CONTD)

- STEPS INVOLVED IN DEVELOPING THE ALGORITHM
- 1. REMOVAL OF NOISES (SALT & PEPPER , DE-BLUR)
- 2.COMPLEMENT AND DETECT EDGES
- 3. FILL THE GAPS TO FINE TUNE EDGE DETECTION

3. RECAP OF WORK COMPLETED TILL MID-SEM (CONTD)

- 4. DILATE THE IMAGE FOR EXTRACTION OF LINES
- 5. FURTHER EDGE DETECTION
- 6. EXTRACT AND DILATE
- 7. POLYNOMIAL FITTING AND DILATION

- 8. MEASUREMENT OF DISTANCES

4. RESULTS OF THE THICKNESS OBTAINED

Sr. No	Image Used	Thickness of the cornea (µm)
1	Patient1-Other eye	492.1063
2	Patient1-postop6months	461.2228
3	Patient1-preop	570.3445
4	Patient2-other eye	469.4584
5	Patient2-postop	487.9885
6	Patient2preop	504.4597
7	Patient3postop	494.1652
8	Patient3preop	500.3419
9	Patient4postop	547.6966
10	Patient4preop	494.1652

4. RESULTS OF THE THICKNESS OBTAINED (CONTD)

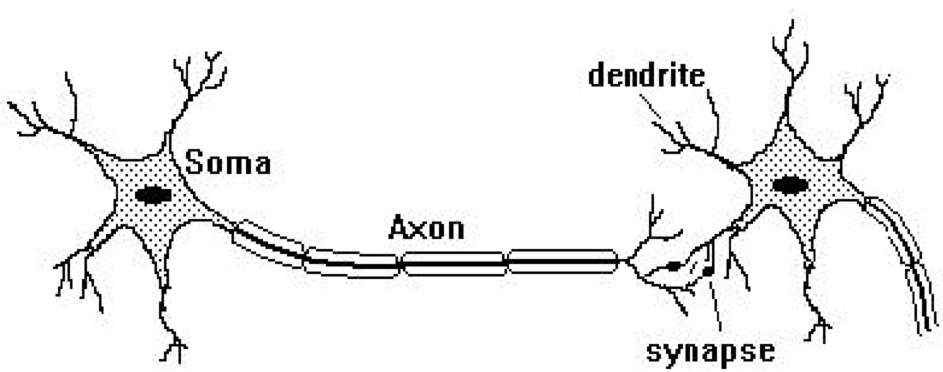
Comparison between the thickness of cornea of Normal and Post-Surgery eye.

Patient No.	Thickness of Normal (Other Eye) (μm)	Thickness of Post-Surgery (μm)	Percentage Change (%)
1	492.1063	461.2228	-6.27
2	469.4584	487.9885	3.95
3	499.3	494.1652	-1.03
4	499.3	547.6966	9.69

Overall Average Change (%) = **+1.57** (Increase from the Normal Thickness)

5. PREDICTION OF THICKNESS

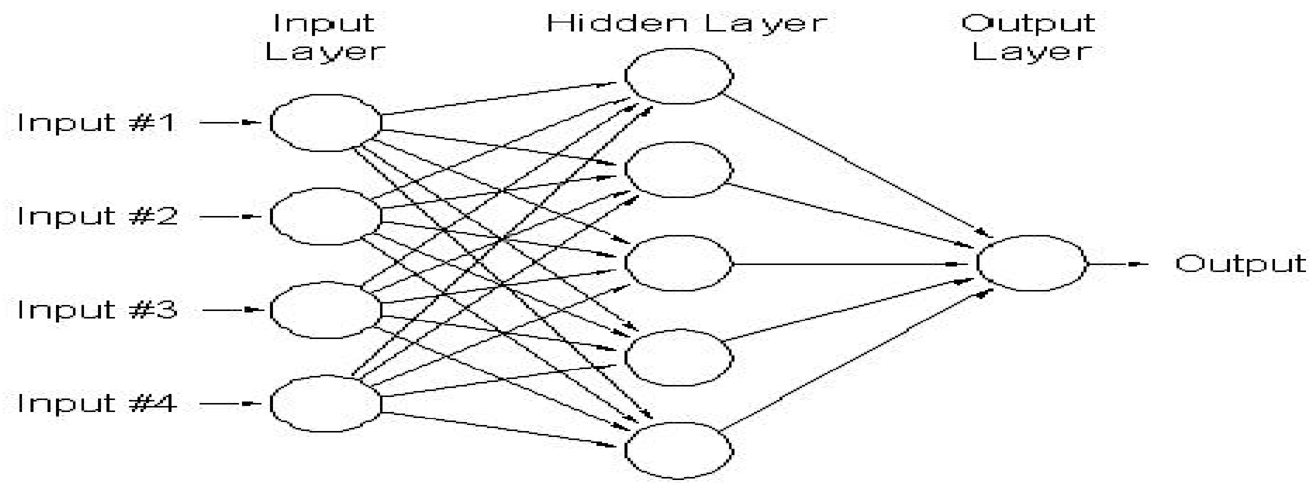
- 5.A) INTRODUCTION TO NEURAL NETWORK
  - BIOLOGICAL OVERVIEW



- STEPS INVOLVED
- 1. SIGNAL CONSISTS OF SYNAPTIC WEIGHTS
- 2. CALCULATION OF CUMULATIVE STIMULUS
- 3. ALL OR NONE (SPIKE TRAVELS THROUGH AXON)

5.A) INTRODUCTION TO NEURAL NETWORK (CONTD)

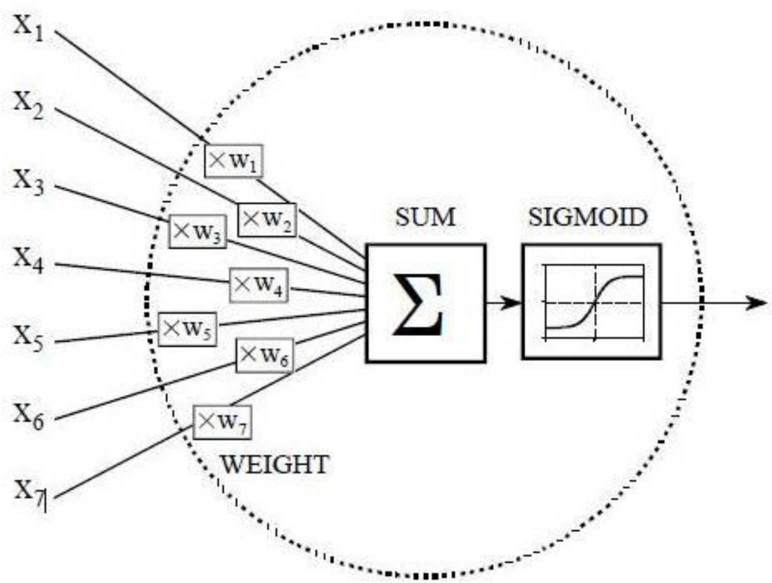
- A SIMPLE MODEL



- 2 TYPES OF NN BASED LEARNINGS
- A) BACK PROPAGATION (OUTPUT VALUES ARE KNOWN BEFOREHAND)
- B) CLUSTERING (UNSUPERVISED – OUTPUT VALUES ARE UNKNOWN)

5.A) INTRODUCTION TO NEURAL NETWORK (CONTD)

- THE HIDDEN & OUTPUT NODES ARE



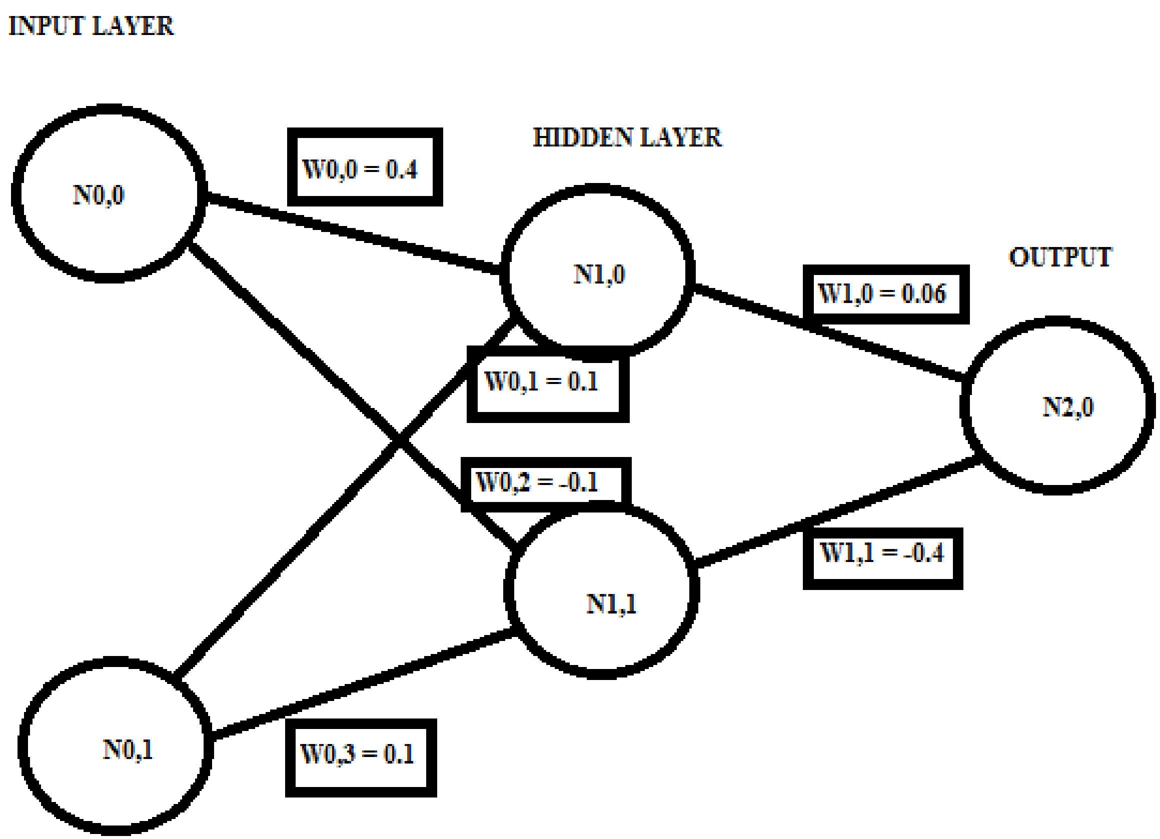
- SIGMOID FUNCTION : $\boldsymbol{f(x) = \frac{1}{1+e^{-x}}}$

5.B) BACK PROPAGATION ALGORITHM

- ALGORITHM CAN BE DECOMPOSED INTO 4 STEPS
- 1) FEED-FORWARD COMPUTATION
- 2) BACK PROPAGATION TO THE 0/P LAYER
- 3) BACK PROPAGATION TO THE HIDDEN LAYER
- 4) WEIGHT UPDATES

5.B) BACK PROPAGATION ALGORITHM(CONTD)

- WORKED OUT EXAMPLE



5.B) BACK PROPAGATION ALGORITHM(CONTD)

- Data Set for the Neural Network

PATTERN DATA FOR "AND"			
n 0,0	n 0,1	Output n 2,0	
0	0	0	
0	1	0	
1	0	0	
1	1	1	

- β = Learning Rate = 0.45
- α = Momentum Term = 0.9
- $\boldsymbol{f(x) = \frac{1}{1+e^{-x}}}$

5.B) BACK PROPAGATION ALGORITHM(CONTD)

- CALCULATION OF HIDDEN LAYER NEURONS
- N1,0 = $f(x1) = f(w0{,}0 * n0{,}0\ + w0{,}1 * n0{,}1\ ) = f(0.4 + 0.1) = f(0.5) = 0.622459$
- N1,1 = $f(x2) = f(w0{,}2 * n0{,}0\ + w0{,}3 * n0{,}1\ ) = f(-0.1 - 0.1) = f(-0.2) = 0.450166$

- CALCULATION OF OUTPUT NEURON
- N2,0 = $f(x3) = f(w1{,}0 * n1{,}0\ + w1{,}1 * n1{,}1\ ) = f(0.037 - 0.18) = 0.464381$

- ERROR CALCULATION OF OUTPUT NEURON
- $$N2{,}0_{Error}\ = n2{,}0 * (1 - n2{,}0) * (N2{,}0_{Desired}\ - N2{,}0)$$
$$= 0.464381 * (1 - 0.464381) * (1 - 0.464381) = 0.133225$$

5.B) BACK PROPAGATION ALGORITHM(CONTD)

- CALCULATION OF NEW WEIGHTS (Rate of Change)

- $\Delta W1{,}0 = \beta * N2{,}0_{Error} * n1{,}0 = 0.45 * 0.133225 * 0.622459 = 0.037317$
- $W1{,}0_{New} = w1{,}0_{old} + \Delta W1{,}0 + \left(\alpha * \Delta(t-1)\right) = 0.06 + 0.037317 + 0 = 0.097137$

- $\Delta W1{,}1 = \beta * N2{,}0_{Error} * n1{,}1 = 0.45 * 0.133225 * 0.450166 = 0.026988$
- $W1{,}1_{New} = w1{,}1_{old} + \Delta W1{,}1 + \left(\alpha * \Delta(t-1)\right) = -0.4 + 0.026988 = -0.373012$

5.B) BACK PROPAGATION ALGORITHM(CONTD)

- **CALCULATION OF RATE OF CHANGES FOR INPUT LAYER**

$$N1{,}0_{Error} = N2{,}0_{Error} * W1{,}0_{New} = 0.133225 * 0.097317 = 0.012955$$

$$N1{,}1_{Error} = N2{,}0_{Error} * W1{,}1_{New} = 0.133225 * (-0.373012) = -0.049706$$

$$\Delta W0{,}0 = \beta * N1{,}0_{Error} * n0{,}0 = 0.45 * 0.012965 = 0.005834$$
$$\Delta W0{,}1 = \beta * N1{,}0_{Error} * n0{,}1 = 0.45 * 0.012965 = 0.005834$$
$$\Delta W0{,}2 = \beta * N1{,}0_{Error} * n0{,}0 = 0.45 * -0.049706 = -0.022368$$
$$\Delta W0{,}3 = \beta * N1{,}0_{Error} * n0{,}1 = 0.45 * -0.049706 = -0.022368$$

5.B) BACK PROPAGATION ALGORITHM(CONTD)

- NEW WEIGHTS BETWEEN INPUT AND HIDDEN LAYER

$$W0{,}0_{New} = W0{,}0_{Old} + \Delta W0{,}0 + \left(\alpha * \Delta(t-1)\right) = 0.4 + 0.005834 = 0.405834$$

$$W0{,}1_{New} = W0{,}1_{Old} + \Delta W0{,}1 + \left(\alpha * \Delta(t-1)\right) = 0.1 + 0.005834 = 0.105834$$

$$W0{,}2_{New} = W0{,}2_{Old} + \Delta W0{,}2 + \left(\alpha * \Delta(t-1)\right) = -0.1 - 0.022368 = -0.122368$$

$$W0{,}3_{New} = W0{,}3_{Old} + \Delta W0{,}3 + \left(\alpha * \Delta(t-1)\right) = -0.1 - 0.022368 = -0.122368$$

- PERFORMING THE ABOVE STEPS WITH NEW WEIGHTS
- The calculated error was 0.133225 and new calculated error is 0.131102
- Number of iterations in typical NN would be any number from ten to ten thousand.

5.C)LEVENBERG MARQUARDT ALGORITHM

- Also known as damped least-square (DLS) method.
- Given a set of pairs of variables (x_i, y_i)
- To optimize the parameter β of the model curve $f(x, \beta)$
- Sum of squares of deviations becomes minimum
  $S(\beta) = \sum_{i=1}^{m}[y_i - f(x_i, \beta)]^2$

5.C)LEVENBERG MARQUARDT ALGORITHM (CONTD)

- SOLUTION

- As it is an iterative procedure , firstly we provide an initial guess for β.
- In iteration step, the parameter β is replaced by a new estimate β+δ.
- To determine δ, the function $f(x_i, \beta + \delta)$ is approximated by $f(x_i, \beta + \delta) \approx f(x_i, \beta) + J_i\,\delta$

where $J_i = \frac{\partial\, f(x_i,\beta)}{\partial \beta}$

5.C)LEVENBERG MARQUARDT ALGORITHM (CONTD)

- At the minimum of the sum of squares $S(\beta)$
- The gradient of S w.r.t β will be zero
- The first-order approximation for the next iteration would be $S(\beta + \delta) \approx \sum_{i=1}^{m}(y_i - f(x_i, \beta) - J_i\delta)^2$
- Or in Vector notation $S(\beta + \delta) \approx \| y - f(\beta) - J\delta \|^2$
- Derivate w.r.t δ and setting it to zero

- $(J^T J)\delta = J^T[y - f(\beta)]$

5.C)LEVENBERG MARQUARDT ALGORITHM (CONTD)

- Levenberg's contribution is to replace the previous equation by a "damped version"
- $(J^T J + \lambda I)\delta = J^T[y - f(\beta)]$
- Marquardt argued that to reach the directions where the gradient is smaller , we need to replace I with diagonal elements of $J^T J$.
- Levenberg-Marquardt Algorithm

$$(J^T J + \lambda \boldsymbol{diag}(J^T J))\delta = J^T[y - f(\beta)]$$

5.D) RESULTS OBTAINED

- To predict the thickness of the post-surgical cornea a MATLAB based solution
- Implemented using Levenberg-Marquardt Back Propagation Algorithm

Patient No.	Actual thickness Post Surgery (µm)	Predicted Thickness Post Surgery (µm)
1	461.228	487.643
2	487.98	581.15
3	494.165	529.17
4	547.696	546.16

- Prediction Error (%) = **+7.687** (Increase from the Actual Thickness)

5.D) RESULTS OBTAINED (Contd)

- In general the number of neurons in the Hidden layer is obtained by Trial & Error
- Results for 1-10 Neurons

No. of Hidden Neurons	Regression Value
1	0.28312
2	0.69818
3	0.44278
4	0.74866
5	0.36021
6	0.5522
7	0.67276
8	0.79043
9	0.39937
10	0.1144

5.E) DETERMINATION OF NUMBER OF NEURONS IN THE HIDDEN LAYER – THEORETICAL STUDY

- Based on "*A Two Phase Method for Determining the Number of Neurons in the Hidden Layer of a 3-Layer neural Network*" – Kazuhiro Shin-ike,MNCT,Kyoto
- **Phase-I** : Candidates for the number of neurons is determined by using the back-propagation method.
- **Phase-II** : Optimal number of neurons is determined by considering the generalization capacity.

5.E (Contd)

- Experiment Implemented
- To measure the predictive accuracy of the output values of Ex-OR circuit.
- 352 data are obtained and grouped into 4 groups of 88.
- Using Trial & Error Method
- Step-1: 3 groups for Training and 1 group for validation

5.E (Contd)

- Step-2: Number of neurons in Hidden Layer
- 5,10,15,20,25 and 30.
- Step-3: Training until the weights – 20,000 times or sum-squared error <= 0.3
- Step-4: Repeat Step-1 to 3 for 10 times and take the best outcome

No. of Neurons	Training Iteration
5	3059
10	1402
15	5434
20	2414
25	2221
30	2522

5.E (Contd)

- Using Two-phase method
- Step-1: Apply hit & trial for 2 groups of data for training in the first phase.
- Selection criteria for next level is >90%

No. of Neurons	No. of correct answers	Correctness Rate (%)
5	146	83
10	159	90.3
15	159	90.3
20	164	93.2
25	160	91
30	159	90.3

5.E (Contd)

- Step-2: For the 2nd phase , use new data which were not used to train the neural network and perform the simulations

No. of Neurons	No. of correct answers
10	68
15	65
20	65
25	74
30	59

- Step-3: Select which yields the maximum no. of correct answers.

THANK YOU