

## INDEX

### Introduction

Page **2**: Step 1. Data Visualization

Page **4**: Step 2. Dataset Verification and Repairing

Page **4**: Step 3. Data pre-processing

Page **5**: Step 4. Neural Network Creation

Page **6**: Step 5. Visualization of the trained Neural Network

Page **7**: Step 6. Neural Network Optimization

Page **7**: Step 7. Test Results

Page **9**: Step 8. Kohonen Network, elbow and silhouette method

Page **11**: Step 9. Conclusion

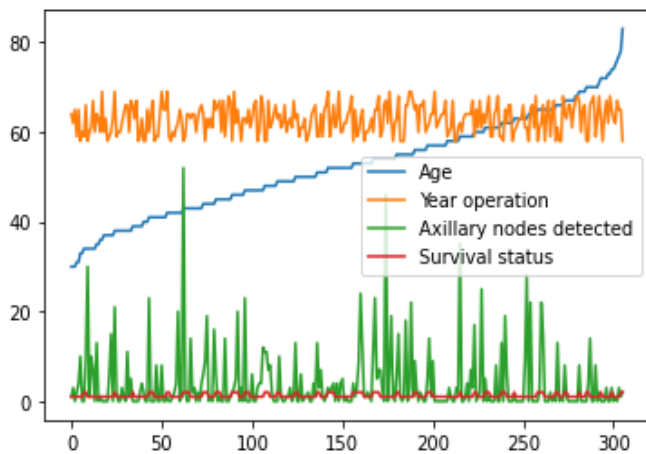
## INTRODUCTION

Για την υλοποίηση του τελικού προτζεκτ επέλεξα το Heberman's Survival Data set το οποίο περιεχει τις περιπτώσεις θνησιμότητας ασθενων οι οποίοι εκαναν χειρουργική επεμβαση για τον καρκίνο του μαστού . Το συγκεκριμενο dataset αποτελείται από 3 + 1 στηλες ,3 για τα δεδομενα του και 1 για το αποτελεσμα (target ) . Κάθε στηλη ονομαστικε αναλογα 'Age' , 'Year Operation' , 'Axillary Nodes Detected' , και για το target 'Survival status' .

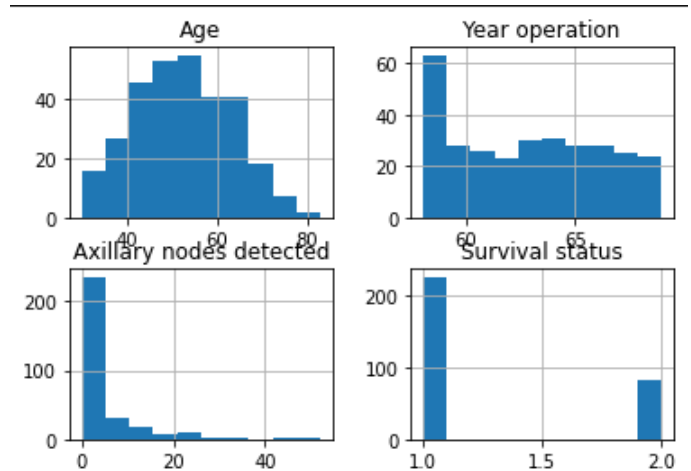
<https://archive.ics.uci.edu/ml/datasets/haberman's+survival>

### STEP 1 . DATA VISUALIZATION

Στο παρακατω γραφιμα προβαλουμε όλα τα δεδομενα σε ένα διαγραμμα plot [1.1.1], καθώς και μια αρχική ποσοστιαία απεικόνιση [1.2.1] των τιμών που περιεχονται .

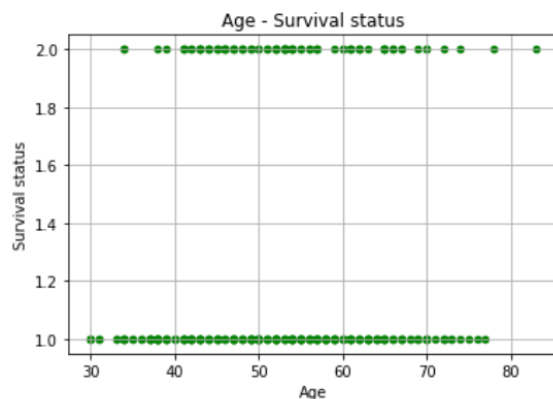


1.1 1



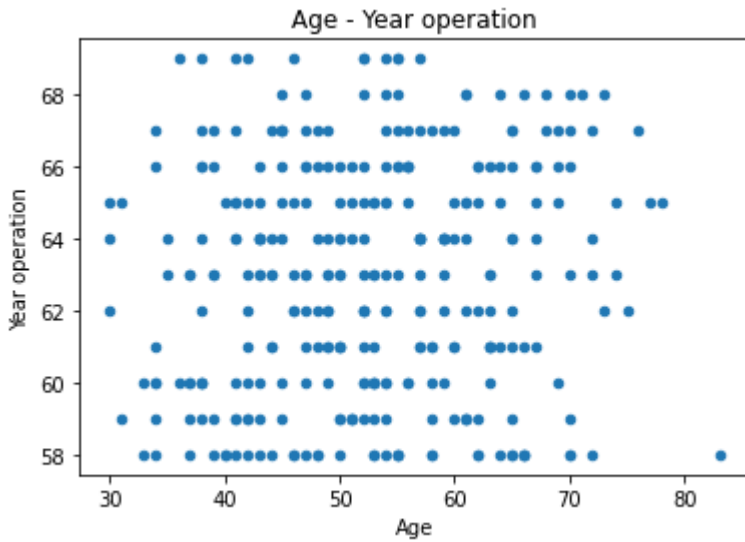
1.2 1

Όπως Παρατηρουμε στο διαγραμμα [1.3.1] τυπου scatter plot ο στοχος του δικτυου αποτελείται από 1 και 2 , για να το χρησιμοποιησουμε στο θα πρεπει να μετατραπει στο διαδικο συστημα . Ετσι τα 1 θα γινουν 001 και αναλογα το 2 θα μετατραπει σε 010 . Ο στοχος θα αποθηκευτει ως ενας πολυδιαστατος πινακας με διαστασεις 306 x 3

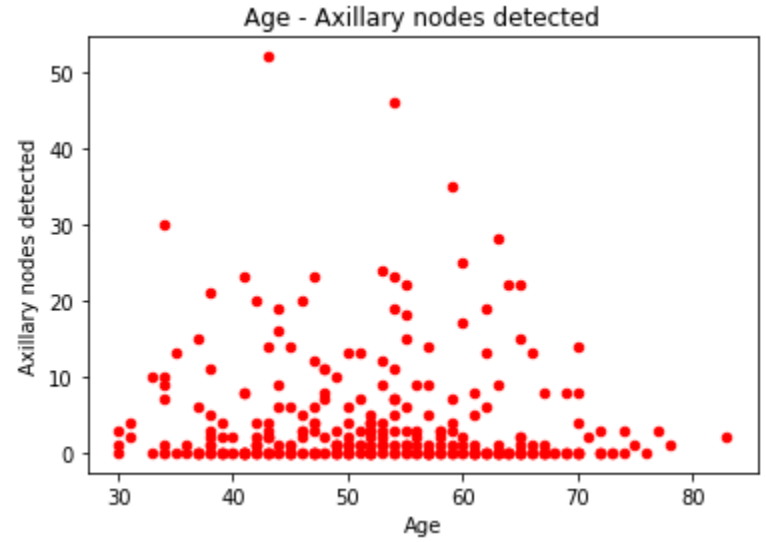


1.3 1

Στα διαγράμματα [1.4.1] και [1.5.1] μπορούμε να διακρίνουμε την σχέση Age – Axillary nodes , Παρατηρούμε ότι όσο μεγαλύτερος ο αριθμός Axillary nodes τσες περισσότερες οι πιθανότητες να επιβιώσει ο ασθενής . Ιδιαίτερα εντονα το βλέπουμε στις ηλικίες 40 και ανω

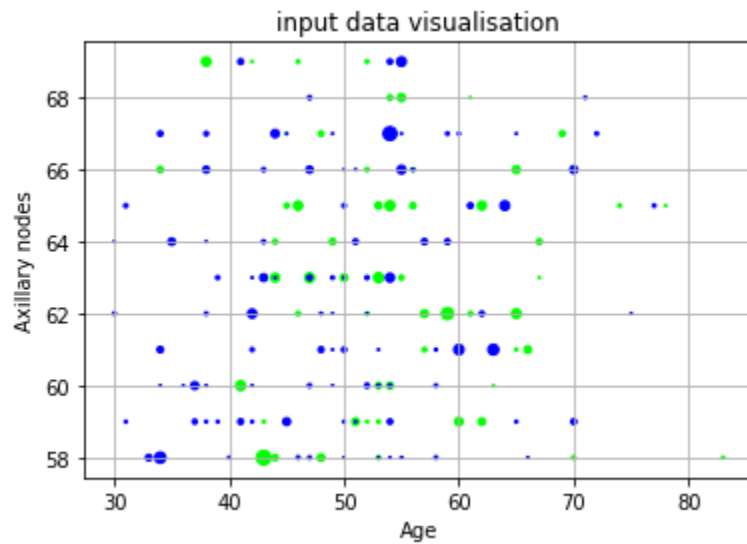


1.4 1



1.5 1

Στο τελευταio διαγραμμα [1.6.1] Scatter Plot απεικονιζονται όλα τα δεδομενα του Dataset



1.6 1

## STEP 2 (OPTIONAL). DATASET VERIFICATION AND REPAIRING

Ελέγχο το dataset για πιθανες null τιμες

	Age	Year operation	Axillary nodes detected	Survival status
0	False	False	False	False
1	False	False	False	False
2	False	False	False	False
3	False	False	False	False
4	False	False	False	False
...	...	...	...	...
301	False	False	False	False
302	False	False	False	False
303	False	False	False	False
304	False	False	False	False
305	False	False	False	False
306 rows × 4 columns				

### 2.1.1

Από τον παραπάνω πίνακα [2.1.2] μπορούμε να διευκρινίσουμε αν υπάρχει κάποια «κενή» σειρά στον πίνακα του dataset . Αφού μας εμφανίζει σε όλες τις γραμμές False σημαίνει ότι ο πίνακας μας δεν έχει κάποιο κενό κελί και μπορούμε να συνεχίσουμε την διαδικασία χωρίς κάποια περεταιρω τροποποίηση

## STEP 3. DATA PRE-PROCESSING

Χωρίζουμε το dataset σε 2 κομμάτια . Το ένα κομμάτι θα είναι για να εκπαιδευσουμε το δίκτυο ενώ το δεύτερο κομμάτι θα χρησιμοποιείται για να δοκιμάσουμε το performance του δικτύου . Το αρχικό split που εφαρμόζουμε είναι 70 – 30 % .

Επειτα χρησιμοποιούμε την συνάρτηση MinMaxScaler() από την βιβλιοθήκη scikit-learn η οποία αφού την εφαρμόσουμε αλλάζει το εύρος τιμών σε μηδέν και ένα [ 0 , 1 ]

#### STEP 4. NEURAL NETWORK CREATION

Για την δημιουργία του Νευρωνικού δικτύου χρησιμοποιήσα "feed forward" νευρονα η οποία υλοποιείται μέσω της συναρτησης newff της βιβλιοθήκης neurolab . Στην συνέχεια το activation function του output layer και των hidden layer αλλάζει και χρησιμοποιεί την λογαριθμική συναρτηση Log Sigmoid η οποία περιορίζει την επιστροφή δεδομένων της στις τιμές [0,1]

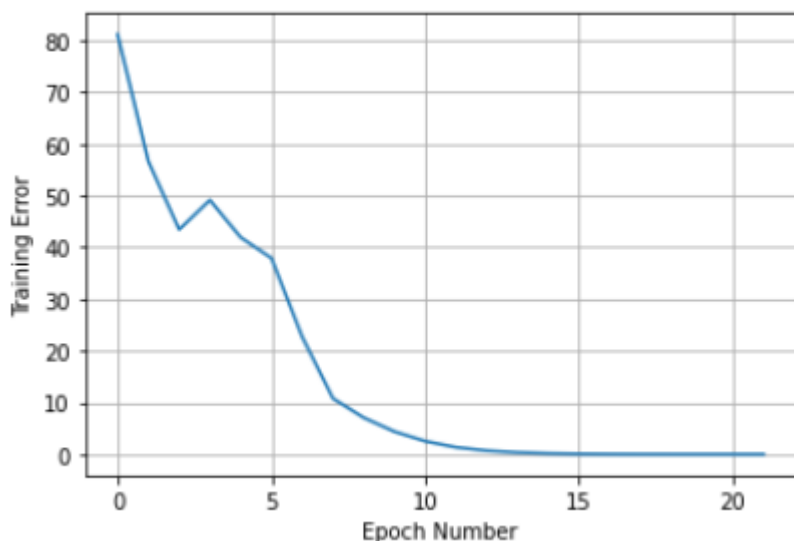
Η εκπαίδευση του δικτύου γίνεται με resilient propagation χρησιμοποιώντας την συναρτηση train\_rprop , η οποία είναι ένας ευρετικός αλγόριθμος για supervised learning . Οι βασικοί παραμετροί για το συγκεκριμένο δίκτυο ήταν οι παρακάτω ,

- Learning rate: 0.3 • Maximum number of epochs to train: 3000 • Performance goal: 1e-5
- Epochs between displays: 100

ενώ ο πίνακας [4.1.2.1] και το γραφικά [4.1.1] ήταν το output του συγκεκριμένου training

Epoch	Error
5	28.21383731704254
10	0.8527816224711587
16	4.936825905004924e-06
The goal of learning is reached	

4.1.2.1



4.1.1

Στο Επομενο μέρος της ενότητας αυτής αναγράφονται τα δεδομένα που συλλεχθηκαν δοκιμαζοντας διαφορετικά Activation και training functions

Στον πίνακα που ακολουθεί υπάρχει καταγεγραμμένο το σφάλμα και οι εποχές που χρειαστηκαν για διαφορετικούς αλγόριθμους training

	Error	Epoch	Accuracy
trans.LogSig	4.936825905004924e-06	16	100.0
trans.SoftMax	epoch 1 – 25: 59.99 epoch 26-1000: nan	1000	0.0
trans.TanSig	214.0	1000	0.0

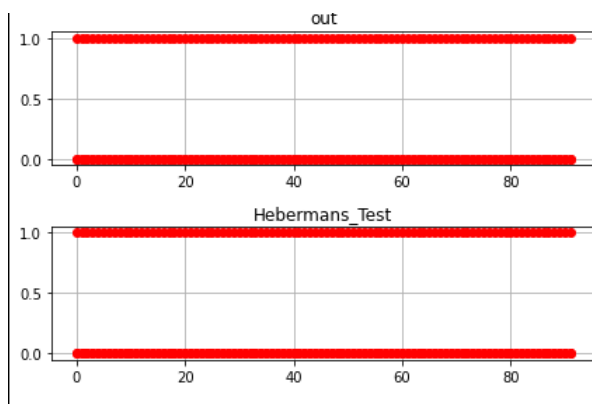
Στον πίνακα που ακολουθεί υπάρχει καταγεγραμμένο το σφάλμα και οι εποχές που χρειάστηκαν για διαφορετικούς αλγορίθμους training

	error	Epoch	Accuracy
train_gd	59.99991233251792	1000	77.17
train_rprop	4.936825905004924e-06	16	100.0
train_gdx	9.605313665412481e-06	272	100.0

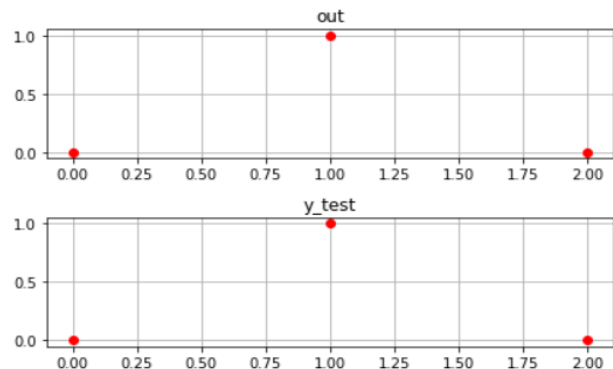
Μπορούμε να παρατηρήσουμε ότι δεν ταιριαζουν τα ίδια είδη training and activation function για το ίδιο dataset.

#### STEP 5. VISUALIZATION OF THE TRAINED NEURAL NETWORK

Χρησιμοποιώντας το βελτιστο δίκτυο από τις δοκιμές της προηγούμενης ενότητας, μπορούμε να υπολογίσουμε το Accuracy του δικτύου καθώς και τη ομοιότητα του με το αρχικό class



5.1 1



5.2 1

Συγκρίνοντας τις δυο παραστάσεις στα παραπάνω γραφήματα [5.1.1 & 5.2.1] παρατηρούμε ότι το output του δικτύου είναι ιδιαίτερα όμοιο με το αρχικό fragment του dataset, τα αποτελέσματα είναι σχεδόν τα ίδια με αυτά της αρχικής κλάσης

## STEP 6. NEURAL NETWORK OPTIMIZATION

Αφου βρεθηκε το βελτιστο ειδος νευρωνικου δικτυου και activation function που θα χρησιμοποιησουμε ειμαστε σε θεση να πειραματιστουμε με τον αριθμο νευρωνων καθως και το πλυθος στο hidden layer.

Όλα τα τεστ τρεχτηκαν 5 φορες και βρεθηκε ο μεσος ορος στα αναλογα αποτελεσματα

Number of neurons	Hidden layer	Average epochs	Best result
4	2	19.2	15
4	1	17.4	13
4	0	12	9
5	2	15	13
5	1	19.6	15
5	0	13.4	11
6	2	15.8	14
6	1	13.8	11
6	0	14.2	10

Ο στοχος επιτιγχανεται σε ολες τις περιπτωσεις σε λιγοτερο από 50 εποχες για αυτό και παραπανω δοκιμες με αριθμο εποχων μεγαλυτερο των 1000 δεν κριθηκε απαιριτητο. Θα συνεχισουμε τις δοκιμες με την αρχικη μορφη του δικτυου αφου μας εδωσε ποιο σταθερα αποτελεσματα κατά την διαρκεια των δοκιμων χωρις μεγαλες διακυμανσεις στον αριθμο των εποχων που χρειαστηκε για να ολοκληρωθει

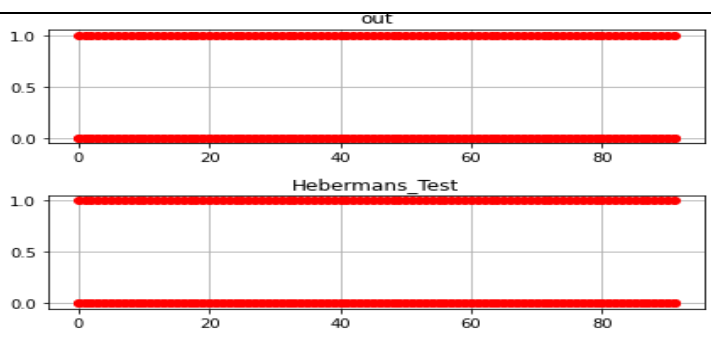
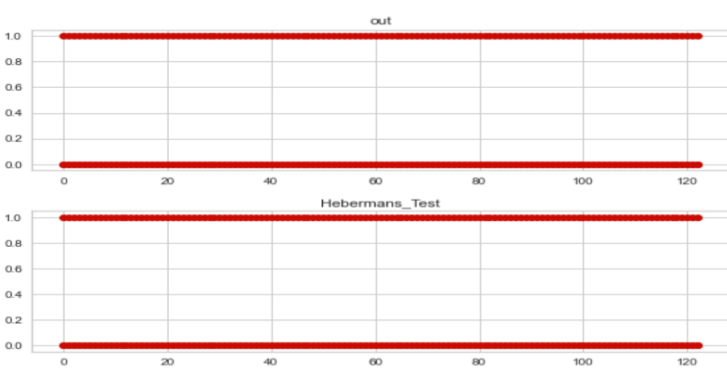
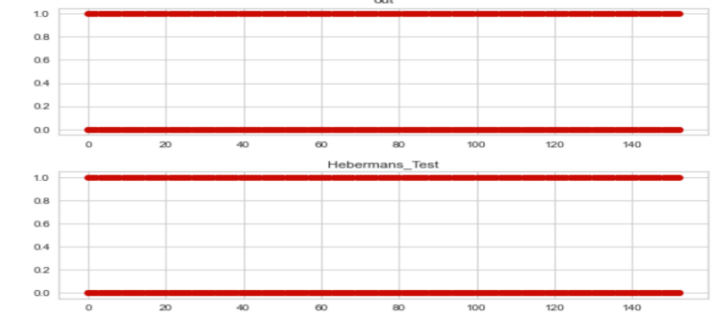
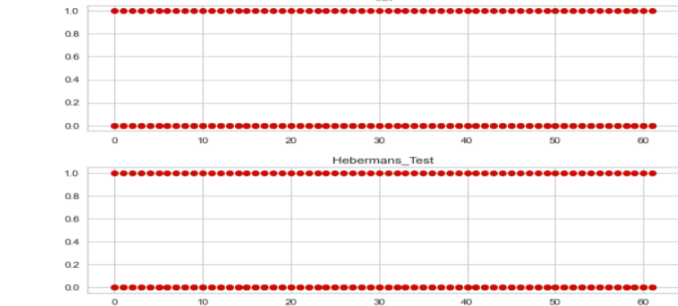
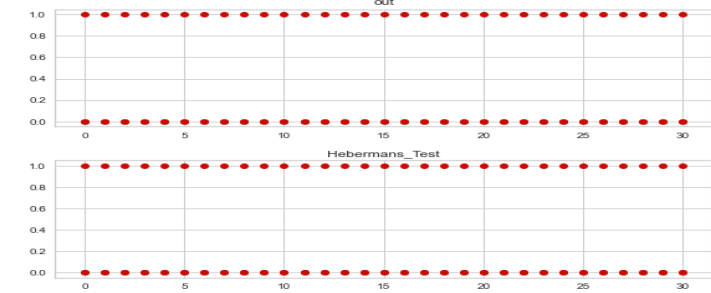
## STEP 7. TEST RESULTS

Όταν αλλαξουμε το ποσοστο με το οποιο διαμοιραζουμε το Training set και το Test set δεν υπαρχουν μεγαλες αλλαγες στον αριθμο που χρειαζεται ώστε να επιτυχουμε το goal του δικτυου . Ο πινακας [7.1.1] καταγραφει τις εποχες που χρειαστηκαν ώστε να φτασουμε το goal αλλα και το accuracy του δικτυου , το οποιο σταθερα παραμενει πολύ ψηλα

Split Rate (%)	70 -30	60-40	50 – 50	80 -20	90 -10
Epochs	15	15	18	21	26
Accuracy	100.0%	100.0%	100.0%	100.0%	100.0%
Error	1.1309042186301525e-05	3.143361368741777e-05	9.864391643613825e-06	4.113408152647851e-05	1.0194909011916465e-05

7.1 1

Η Μεγαλη αλλαγη που μπορουμε να σημειωσουμε είναι η μειωση διγματοληψιας που προερχεται από το Testing set [7.2.1]

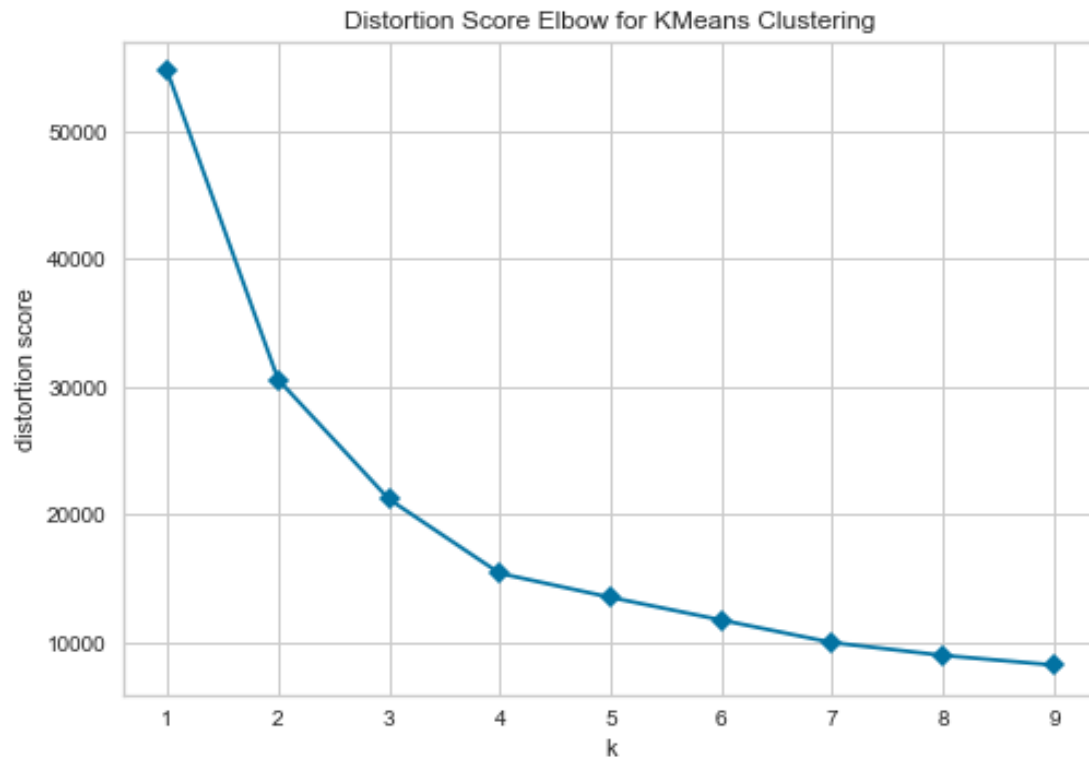
Test split	Graph
70 – 30	 <p>The graphs for the 70-30 test split show two horizontal red lines at y=1.0. The top graph, labeled 'out', has an x-axis from 0 to 80. The bottom graph, labeled 'Hebermans_Test', also has an x-axis from 0 to 80.</p>
60 – 40	 <p>The graphs for the 60-40 test split show two horizontal red lines at y=1.0. The top graph, labeled 'out', has an x-axis from 0 to 120. The bottom graph, labeled 'Hebermans_Test', also has an x-axis from 0 to 120.</p>
50 – 50	 <p>The graphs for the 50-50 test split show two horizontal red lines at y=1.0. The top graph, labeled 'out', has an x-axis from 0 to 140. The bottom graph, labeled 'Hebermans_Test', also has an x-axis from 0 to 140.</p>
80 – 20	 <p>The graphs for the 80-20 test split show two horizontal red lines at y=1.0. The top graph, labeled 'out', has an x-axis from 0 to 60. The bottom graph, labeled 'Hebermans_Test', also has an x-axis from 0 to 60.</p>
90 – 10	 <p>The graphs for the 90-10 test split show two horizontal red lines at y=1.0. The top graph, labeled 'out', has an x-axis from 0 to 30. The bottom graph, labeled 'Hebermans_Test', also has an x-axis from 0 to 30.</p>



## STEP 8. KOHONEN NETWORK, ELBOW AND SILHUETTE METHOD

Μέχρι τώρα το δίκτυο και οι περιπτώσεις που έχουμε εξετάσει είναι του τύπου Supervised learning . Ένα Kohonen network είναι αυτό διοργανωμένο δίκτυο . Δημιουργεί cluster του dataset όταν ακόμα δεν είμαστε σίγουροι τι είναι αυτές οι ομάδες στην αρχή .

Για να δημιουργηθεί ένα kohonen network πρέπει να υλοποιηθεί πρώτα η μέθοδος elbow ώστε να υπολογιστεί ο αριθμός των cluster .

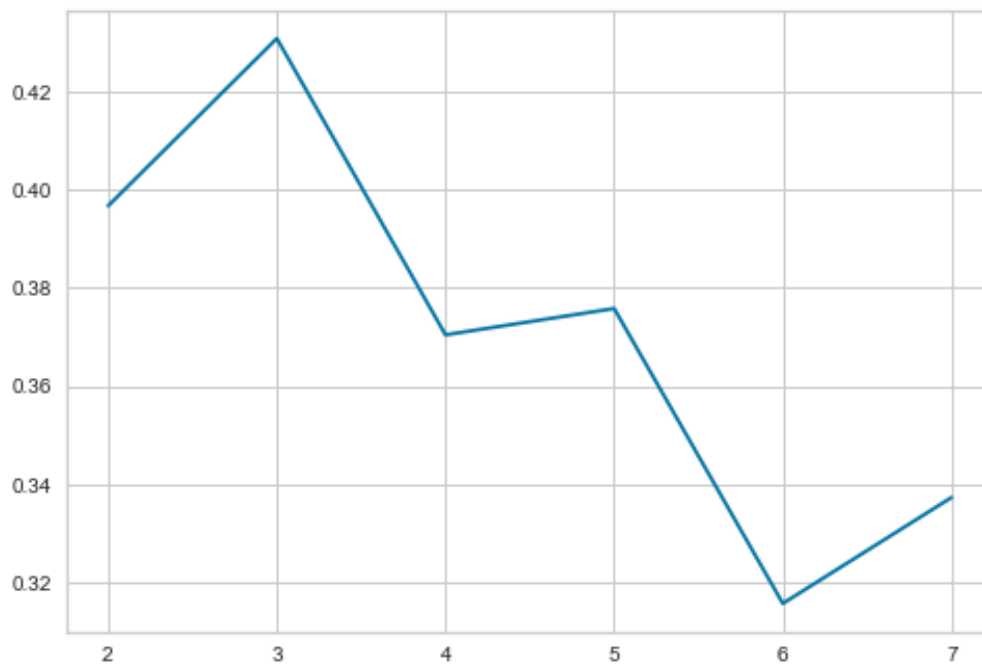


### 8.1.1

Όπως παρατηρούμε στο διαγράμμα [8.1.1] όσο αυξάνεται ο αριθμός των cluster μειώνεται το distortion . Σε αυτό το παραδειγμα συμπεραίνουμε ότι ο βέλτιστος αριθμός cluster είναι 3 , αφού μετά από αυτόν τον αριθμό clusters οι αλλαγές στο distortion αλλάζει με ολο και πιο μικρο ρυθμο .

Στην συνέχεια θα χρησιμοποιήσουμε την silhouette method/analysis η οποία συγκρίνει τα αποτελέσματα ενός data point σε ένα cluster συγκρινοντας το με άλλα cluster . Η διαδικασία παίρνει τιμές που κυμαίνονται ανάμεσα σε  $[-1, 1]$  . Όταν το αποτέλεσμα πλησιάζει το 1 τότε το cluster έχει μεγάλη συμπύκνωση σημείων , αντιθετα στο -1 σημαίνει ότι είναι μακριά από άλλα cluster . Σε περίπτωση που πλησιάζουμε το 0 σαν αποτέλεσμα οι τιμές σε αυτή την περιοχή κάνουν overlap με γειτονικά cluster

Στην γραφική παρασταση [8.2.1] παρατηρούμε ότι το silhouette score μεγιστοποιείται στο  $k = 3$ , αρα θα χρησιμοποιήσουμε 3 cluster

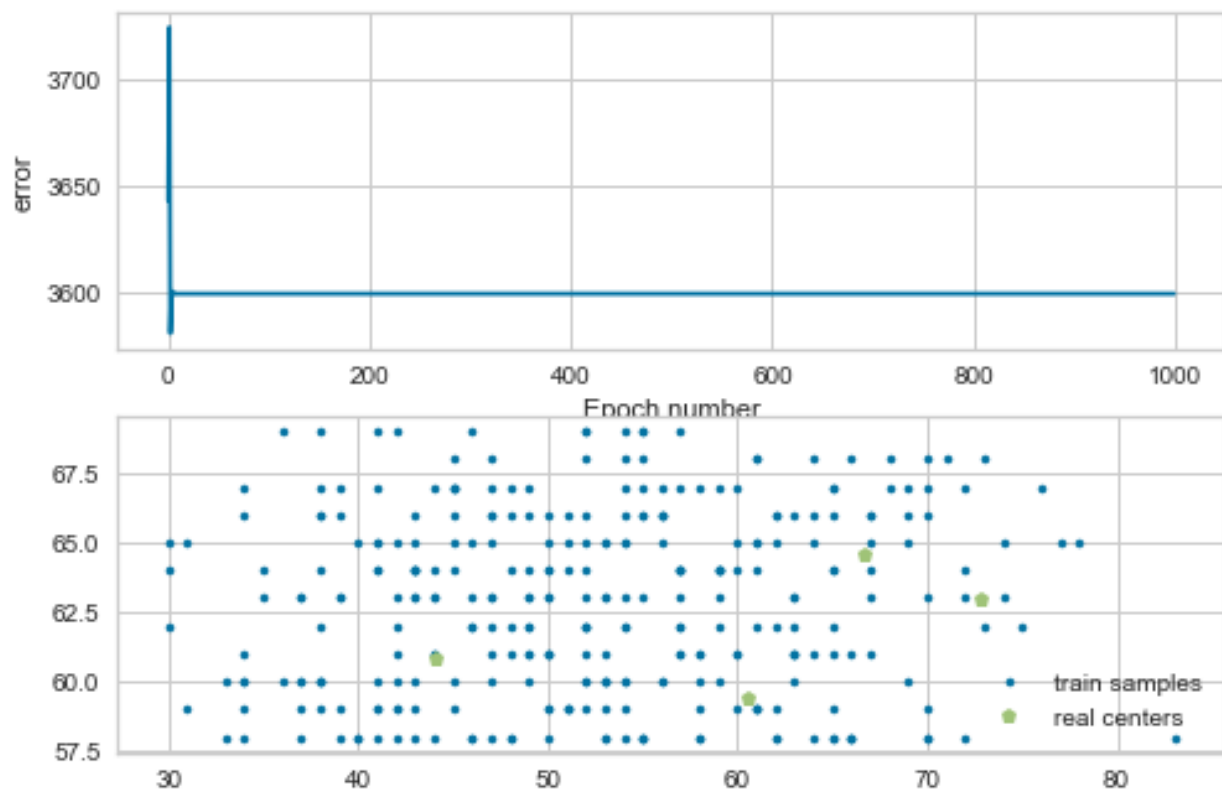


### 8.2.1

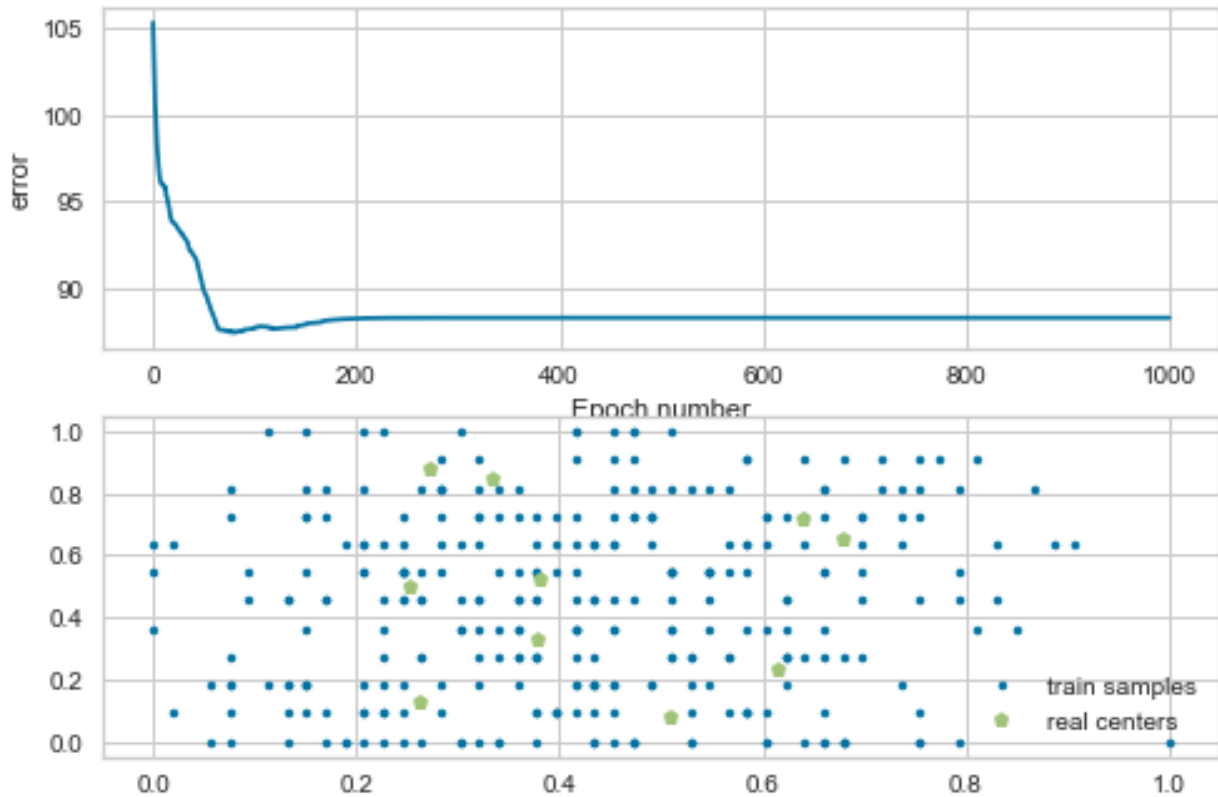
Χρησιμοποιούμε συνεργατικά τις μεθόδους elbow και silhouette ώστε να κάνουμε μια πιο ακριβή και σωστή απόφαση επιλογής cluster

Δημιουργούμε ένα απλό δίκτυο με 4 output layers , ενώ χρησιμοποιούμε δυο μεθόδους εκπαίδευσης:

1. Consience Takes All



## 2. Winner takes All



### Step 9. Conclusion

Κατά την γνώμη μου ένα αποτελεσματικό νευρωνικό δίκτυο δεν εξαρτάται από μια πτυχή του περισσότερο από τις άλλες. Δεν μπορούμε να βασιστούμε μόνο στην σωστή διαχείριση των δεδομένων ώστε να δημιουργήσουμε ένα δίκτυο που επεξεργάζεται γρήγορα τα δεδομένα αυτά, είναι εξίσου σημαντικό να χρησιμοποιούμε την σωστή αναλογία νευρώνων ή το σωστό είδος δικτύου, το activation function που μας δίνει το αποτέλεσμα σε λιγότερες εποχές. Ένα νευρωνικό δίκτυο όπως και ο εγκέφαλος του ανθρώπου, που είναι βασισμένο, δεν έχει κάποιο κομμάτι το οποίο θα λύσει το πρόβλημα πιο αποτελεσματικά. Εν κατακλείδι πιστεύω ότι ένα σωστό νευρωνικό δίκτυο δεν εξαρτάται από μια παραμετρο αλλά από τον συνδυασμό τεχνικών που χρησιμοποιούμε ώστε να λύσουμε το πρόβλημα που έχουμε μπροστά μας.