Final Project Artificial Neural Networks

Nikos Periklis Chourdsas -Tp4774

[nikos.hourdas@gmail.com](mailto:nikos.hourdas@gmail.com)

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 silhuette method

Page **11**: Step 9. Conclusion

Introduction

Για την υλοποιηση του τελικου προτζεκτ επελεξα το Heberman’s Survival Data set το οποιο περιεχει τις περιπτωσεις θνησιμοτητας ασθενων οι οποιοι εκαναν χειρουργικη επεμβαση για τον καρκινο του μαστού . Το συγκεκριμενο dataset αποτελειται 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 προβαλουμαι όλα τα δεδομενα σε ένα διαγραμμα plot [1.1.1], καθως και μια αρχικη ποσοστιαια απεικονηση [1.2.1] των τιμων που περιεχονται .

Chart, histogram

Description automatically generatedChart, histogram

Description automatically generated

1.2 1

1.1 1

Όπως Παρατηρουμε στο διαγραμμα [1.3.1] τυπου scatter plot ο στοχος του δικτυου αποτελειται από 1 και 2 , για να το χρησιμποποιησουμε 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

A picture containing line chart

Description automatically generated

1.3 1

Στα διαγραμματα [1.4.1] και [1.5.1] μπορουμε να διακρινουμε την σχεση Age – Axillary nodes , Παρατηρουμε ότι οσο μεγαλυτερος ο αριθμος Axillary nodes τοσες περισσοτερες οι πιθανοτητες να επιβιωσει ο ασθενης . Ιδιατερα εντονα το βλεπουμε στις ηλικιες 40 και ανω

Chart, scatter chart

Description automatically generatedChart, scatter chart

Description automatically generated

1.5 1

1.4 1

Στο τελευταιο διαγραμμα [1.6.1] Scatter Plοt απεικονιζονται όλα τα δεδομενα του Dataset

Chart, scatter chart

Description automatically generated

1.6 1

Step 2 (optional). Dataset Verification and Repairing

Ελεγχο το dataset gia πιθανες null τιμες

A picture containing text, monitor, screen, black

Description automatically generated

2.1 1

Από τον παραπανω πινακα [2.1.2] μπορουμε να διευκρινησουμε αν υπαρχει καποια «κενη» σειρα στον πινακα του dataset . Αφου μας εμφανιζει σε ολες τις γραμμες False σημαινει ότι ο πινακας μας δεν εχει καποιο κενο κελή και μπορουμε να συνεχισουμε την διαδικασια χωρις καποια περεταιρω τροποιηση

Step 3. Data pre-processing

Χωριζουμε το dataset σε 2 κομματια . Το ένα κομματι θα είναι για να εκπαιδευσουμε το δικτυο ενώ το δευτερο κομματι θα χρησιμοποιειται για να δοκιμασουμε το performance του δικτυου . Το αρχικο split που εφαρμοζουμε είναι 70 – 30 % .

Επειτα χρησιμοποιουμε την συναρτηση MinMaxScaler() από την βιβλιοθηκη scikit-learn η οποια αφου την εφαρμοσουμε αλλαζει το ευρως τιμων σε μηδεν και ένα [ 0 , 1 ]

Step 4. Neural Network Creation

Για την δημιουργια του Νευρωνικου δικτυου χρησιμοποιησα “feed forward” νευρονα η οποια υλοποιειται μεσω της συναρτησης newff της βιλβιοθηκης neurolab . Στην συνεχεια το activation function του output layer και των hidden layer αλλαζει και χρησιμοποιει την λογαριθμικη συναρτηση Log Sigmoid η οποια περιοριζει την επιστροφη δεδομενων της στις τιμες [0,1]

Η εκπαιδευση του δικτυου γινεται με resilient propagation χρησιμοποιωνταςτην συναρτηση train\_rprop , η οποια είναι ενας ευρετικος αλγοριθμος για supervised learning . Οι βασικοι παραμετροι για το συγκεκριμενο δικτυο ηταν οι παρακατω ,

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

• Epochs between displays: 100

ενώ ο πινακας [4.1.2.1] και το γραφικα [4.1.1] ηταν το output του συγκεκριμενου training

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

4.1.2 1

**Chart, line chart

Description automatically generated**

4.1 1

Στο Επομενο μερος της ενοτητας αυτης αναγραφονται τα δεδομενα που συλεχθηκαν δοκιμαζοντας διαφορετικα Activation και training functions

Στον πινακα που ακολουθει υπαρχει καταγεγραμενο το σφαλμα και οι εποχες που χρειαστηκαν για διαφορετικους αλγοριθμους 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 πινακα που ακολουθει υπαρχει καταγεγραμενο το σφαλμα και οι εποχες που χρειαστηκαν για διαφορετικους αλγοριθμους training

|  |  |  |  |
| --- | --- | --- | --- |
|  | error | Epoch | Accuracy |
| train\_gd | 59.99991233251792 | 1000 | 77.17 |
| train\_rprop | 4.93682590500  4924e-06 | 16 | 100.0 |
| train\_gdx | 9.6053136654124  81e-06 | 272 | 100.0 |

Μπορουμε να παρατηρησουμε ότι δεν ταιριαζουν τα ιδια ειδη training and activation function για το ιδιο dataset.

Step 5. Visualization of the trained Neural Network

Χρησιμοποιοντας το βελτιστο δικτυο από τις δοκιμες της προηγουμενης ενοτητας ,μπορουμε να υπολογισουμε το Accuracy του δικτυου καθως και τη ομοιωτητα του με το αρχικο class

Chart, scatter chart

Description automatically generated

Calendar

Description automatically generated

5.2 1

5.1 1

Συγκρινοντας τις δυο παραστασεις στα παραπανω γραφηματα [5.1.1 & 5.2.1] παρατηρουμε ότι το output του δικτυου είναι ιδιαιτερα ομοιο με το αρχικο fragment του dataset , τα αποτελεσματα είναι σχεδον τα ιδια με αυτά της αρχικης κλασης

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 απαραιτητο. 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** | 1.13090421863  01525e-05 | 3.1433613687417  77e-05 | 9.8643916436138  25e-06 | 4.1134081526478  51e-05 | 1.01949090119  16465e-05 |

7.1 1

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

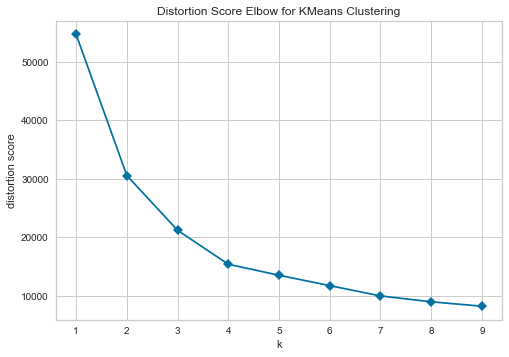
|  |  |
| --- | --- |
| Test split | **Graph** |
| 70 – 30 | Calendar  Description automatically generated |
| 60 – 40 |  |
| 50 – 50 |  |
| 80 – 20 |  |
| 90 – 10 |  |

7.2.1

### Step 8. Kohonen Network, elbow and silhuette method

Μεχρι τωρα το δικτυο και οι περιπτωσεις που εχουμε εξετασει είναι του τυπου Supervised learning . Ένα Kohonen netowrk είναι αυτό διοργανωμενο δικτυο . Δημιουργει cluster του dataset όταν ακομα δεν ειμαστε σιγουροι τι είναι αυτές οι ομαδες στην αρχη .

Για να δημιουργηθει ένα kohonen network πρεπει να υλοποιηθει πρωρτα η μεθοδος elbow ωστες να υπολογιστη ο αριθμος των cluster .

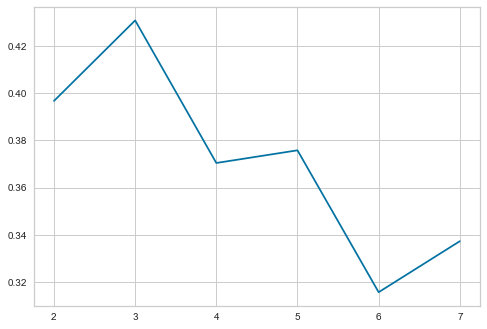


8.1 1

Όπως παρατηρουμε στο διαγραμμα [8.1.1] οσο αυξανεται ο αριθμος των cluster μειωνεται το distortion . Σε αυτό το παραδειγμα συμπεραινουμε ότι ο βελτιστος αριθμςο cluster είναι 3 , αφου μετα από αυτόν τον αιρθμο clusters οι αλλαγες στο distortion αλλαζει με ολο και πιο μικρο ρυθμο .

Στην συνεχεια θα χρησιμοιποιησουμε την silhuette method/analysis η οποια συγκρινει τα αποτελεσματα ενός data point σε ένα cluster συκγρινοντας το με αλλα cluster . Η διαδικασια παιρνει τιμες που κυμενονται αναμεσσα σε [-1 1] .Οταν το αποτελεσμα πλησιαζει το 1 τοτε το cluster εχει μεγαλη συμπικνωση σημειων , αντιθετα στο -1 σημαινει ότι είναι μακρια από αλλα cluster . Σε περιπτωση που πλησιαζουμε το 0 σαν αποτελεσμα οι τιμες σε αυτή την περιοχη κανουν overlap με γειτονικα cluster

Στην γραφικη παρασταση [8.2.1] παρατηρουμε ότι το silhuette score μεγιστοιποιειται στο k = 3, αρα θα χρησιμοποιεισουμε 3 cluster

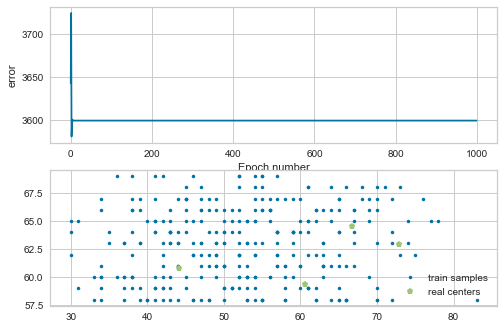


8.2 1

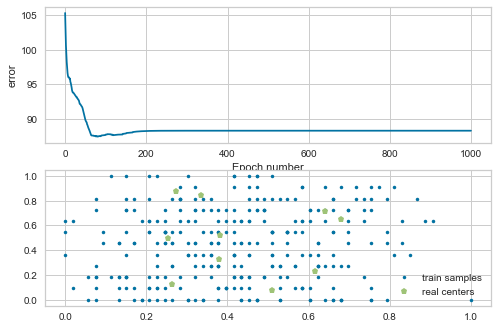
Χρησιμοποιουμαι συνεργατικα τις μεθοδους elbow και silhuette ώστε να κανουμε μια πιο ακριβη και σωστη αποφαση επιλογης cluster

Δημιουργουμε ένα απλο δικτυο με 4 output layers , ενώ χρησιμοποιουμε δυο μεθοδους εκπαιδευεσης:

1. Consience Takes All



1. Winner takes All



**Step 9. Conclusion**