



ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ

Σχολή Μηχανολόγων Μηχανικών

Τομέας Τεχνολογίας και Κατεργασιών

ΔΙΑΤΜΗΜΑΤΙΚΟ ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ "ΣΥΣΤΗΜΑΤΑ

ΑΥΤΟΜΑΤΙΣΜΟΥ" (Ακαδ. Έτος: 2021-22)

Προηγμένα Συστήματα Κατεργασιών - Ακαδ. Έτος 2021-22

3^η Υποχρεωτική εργασία

**Αναγνώριση μοτίβων στατιστικών διαγραμμάτων ελέγχου με τεχνητά
νευρωνικά δίκτυα**

Παπαγιαννούλης Ιωάννης 021-222-13

Ευστάθιος Ράπτης 021-222-17

Σαββάκης Πέτρος 021-222-18

Μελέτη SPC

Στην παρούσα εργασία μας ζητήθηκε η προσέγγιση μοτίβων SPC(Statistical Process Control) και η αυτόματη αναγνώριση τους απο ΤΝΔ. Η αρχική μας προσέγγιση ήταν η κατανόηση των απλών και των σύνθετων μοτίβων, καθώς και η εξοικείωση μας με τους κανόνες Western Electric(Οι κανόνες αυτοί εντοπίζουν εκτός ελέγχου ή μη τυχαίων συνθηκών μοτίβα σε διαγράμματα ελέγχου). Οι γλώσσες προγραμματισμού που επιλέξαμε να χρησιμοποιήσουμε είναι η Python για την παραγωγή και την παρουσίαση των δεδομένων, η γλώσσα προγραμματισμού R για χρήση και εξοικείωση με τα πρότυπα Western Electric και τέλος το πρόγραμμα MatLab για την δημιουργία και την εκπαίδευση του Τεχνητού Νευρωνικού Δικτύου.

Δημιουργία Μοτίβων

Η λογική που ακολουθήσαμε ήταν η δημιουργία μοτίβων 20 σημείων μεταξύ των τιμών ± 3 . Συγκεκριμένα δημιουργήθηκαν τα απλά μοτίβα Uptrend, Stratification, Point Outside of Control Limits και Instability. Επιπλέον δημιουργήθηκαν τα σύνθετα μοτίβα Cycle and Instability και Instability and Grouping. Ο τρόπος δημιουργίας τους ήταν ο εξής:

1. Δημιουργία ενός μοτίβου 20 σημείων για κάθε είδους μοτίβου που θα χρησιμοποιήσουμε (Uptrend, Stratification,....., Instability and Grouping) με τιμές που πληκτρολογούμε χειροκίνητα.
2. Αυτόματη παραγωγή 1000 μοτίβων για κάθε ένα από αυτά που τους προστίθεται τόσο White Noise όσο και τυχαίος θόρυβος απο Γκαουσιανή κατανομή(Εικόνες 1 έως 6 γραφική αναπαράσταση του χειροκίνητου μοτίβου σε σχέση με αυτό που παράγεται που έχει προστεθεί τυχαιότητα). Το τελικό Dataset που δημιουργείται είναι 6000 δειγμάτων μοτίβων.
3. Στο κάθενα από αυτά γίνεται υπολογισμός τεσσάρων παραμέτρων μέσος όρος των 20 σημείων του κάθε μοτίβου(Mean), ελάχιστη και μέγιστη τιμή(min, max) και η κύρτωση (Kurtosis) σύμφωνα με τον ορισμό Fisher. Είναι άξιο να σημειωθεί ότι μπορούν να προστεθούν και άλλοι υπολογισμοί σε μελλοντική υλοποίηση εφόσον η βιβλιοθήκη Python Pandas μας προσφέρει πληθώρα επιλογών. Επομένως η παρούσα κατάσταση του dataset μας είναι ένας πίνακας διαστάσεων 6000x24.
4. Για τα 6 μοτίβα που έχουμε θα πρέπει να δημιουργηθεί και ένα dataset με τα targets το οποίο θα είναι δυαδικό. Αυτό θα είναι της μορφής 6000x6. Δομής:

Stratification Pattern = 1 0 0 0 0 0

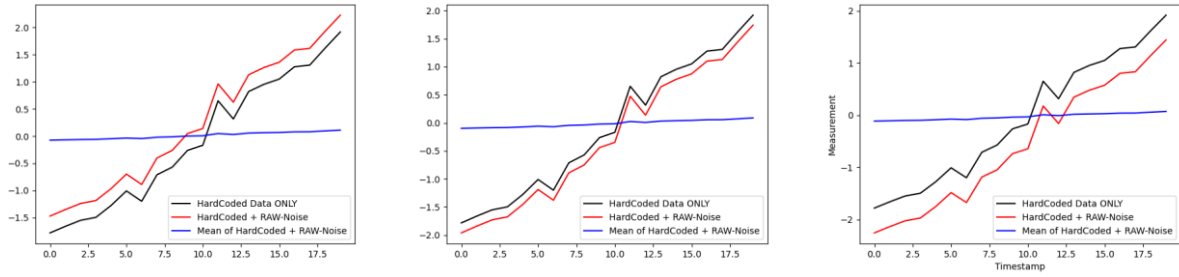
Uptrend = 0 1 0 0 0 0

Instability = 0 0 1 0 0 0

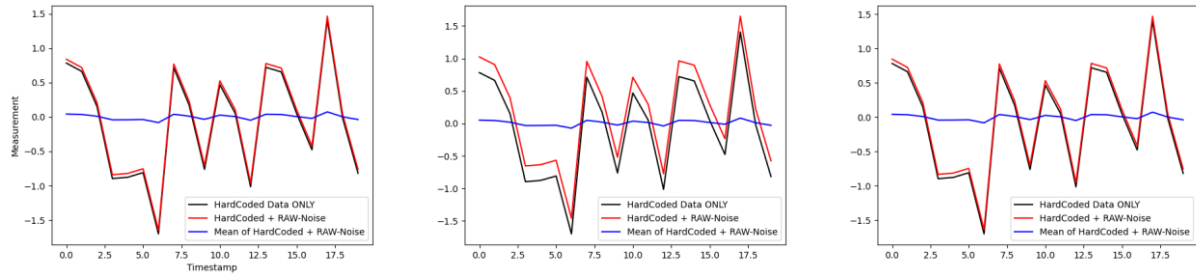
Point Outside of Control Limits = 0 0 0 1 0 0

Cycle and Instability = 0 0 0 0 0 1 0

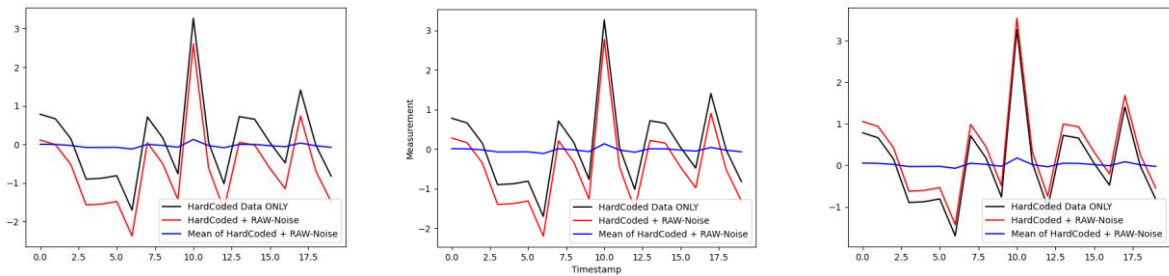
Instability and Grouping = 0 0 0 0 0 1



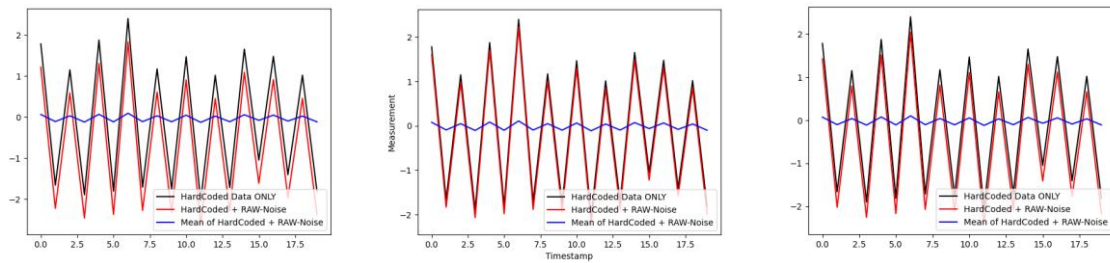
(Εικόνα 1 Παραδείγματα δημιουργίας δεδομένων για pattern: Uptrend)



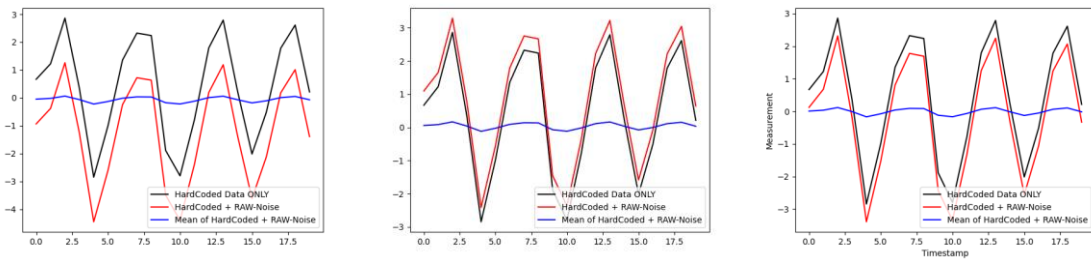
(Εικόνα 2 Παραδείγματα δημιουργίας δεδομένων για pattern: Stratification)



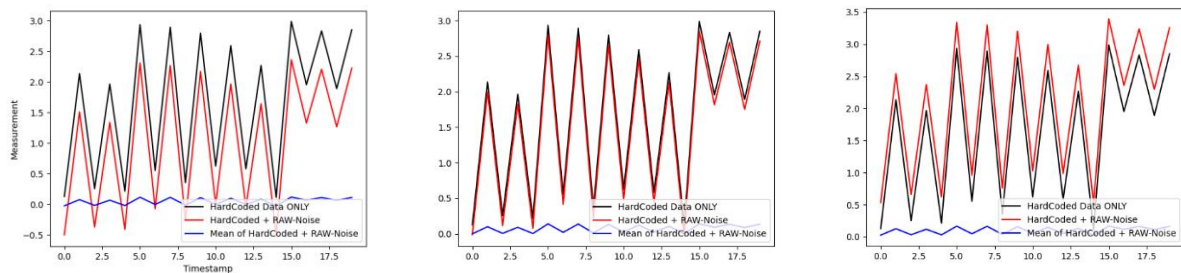
(Εικόνα 3 Παραδείγματα δημιουργίας δεδομένων για pattern: Point Outside Of Control)



(Εικόνα 4 Παραδείγματα δημιουργίας δεδομένων για pattern : Instability)



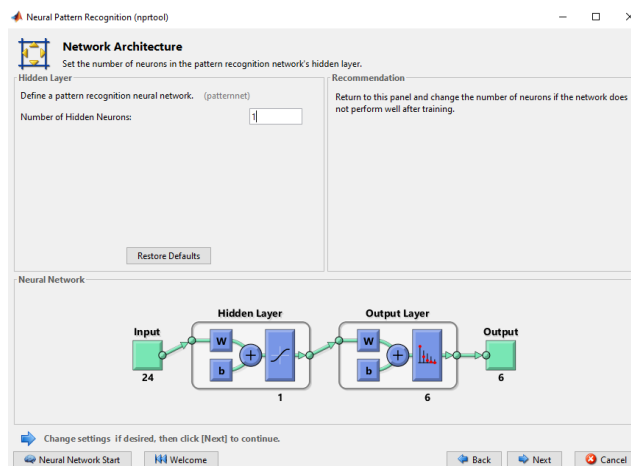
(Εικόνα 5 Παραδείγματα δημιουργίας δεδομένων για *pattern : Cycle and Instability*)

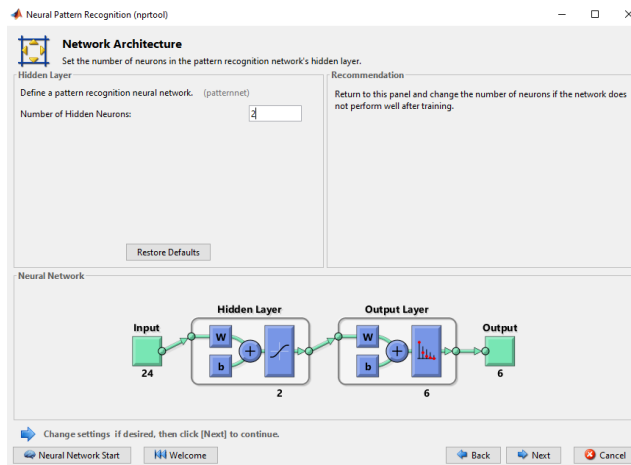


(Εικόνα 6 Παραδείγματα δημιουργίας δεδομένων για *pattern : Instability and Grouping*)

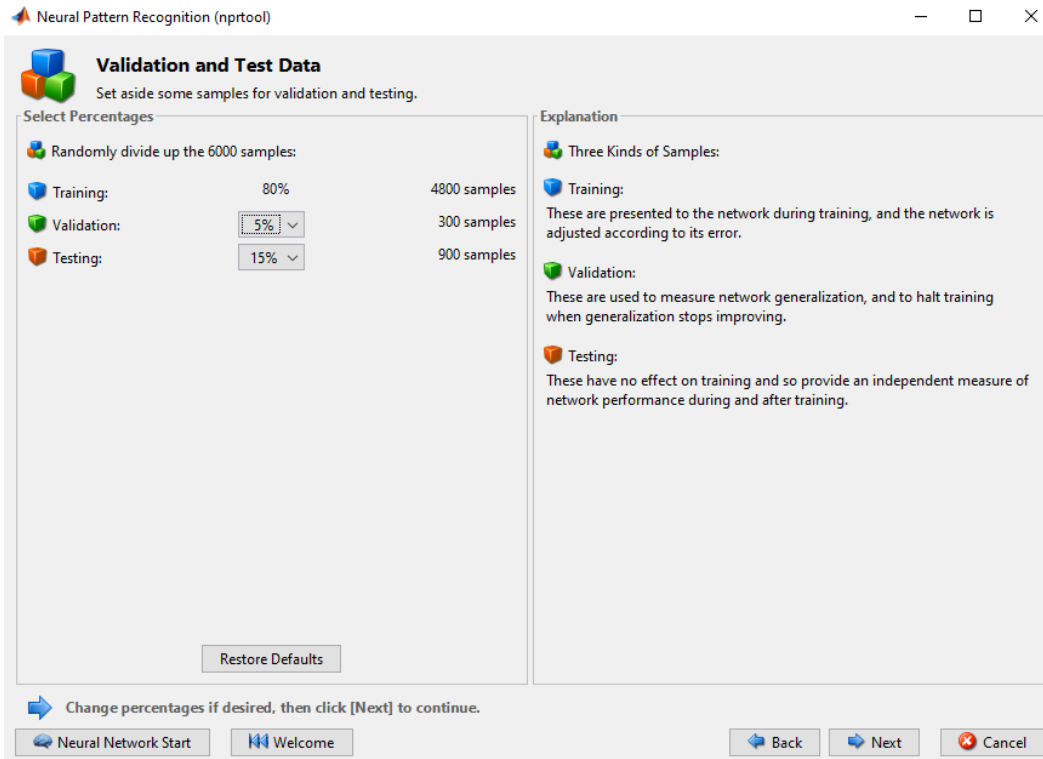
Δημιουργία του Τεχνητού Νευρωνικού Δικτύου

Η δημιουργία του ΤΝΔ έγινε με την χρήση της βιβλιοθήκης Neural Pattern Recognition του MatLab. Επιλέξαμε αυτό το εργαλείο για την εύκολη χρήση που προσφέρει καθώς και την πληθώρα επιλογών. Παρακάτω βλέπουμε τα βήματα που ακολουθήσαμε για την δημιουργία 2 ΤΝΔ. Η μοναδική διαφορά των δύο ΤΝΔ είναι ότι το ένα απαρτίζεται από ένα Hidden Layer και το άλλο από δύο (Εικόνα 7). Η εκπαίδευση των ΤΝΔ έγινε με ποσοστό 80-5-15 δηλαδή 4800 training data, 300 validation data και 900 testing data (Εικόνα 8).





(Εικόνα 7 Αρχιτεκτονική των δύο ΤΝΔ)

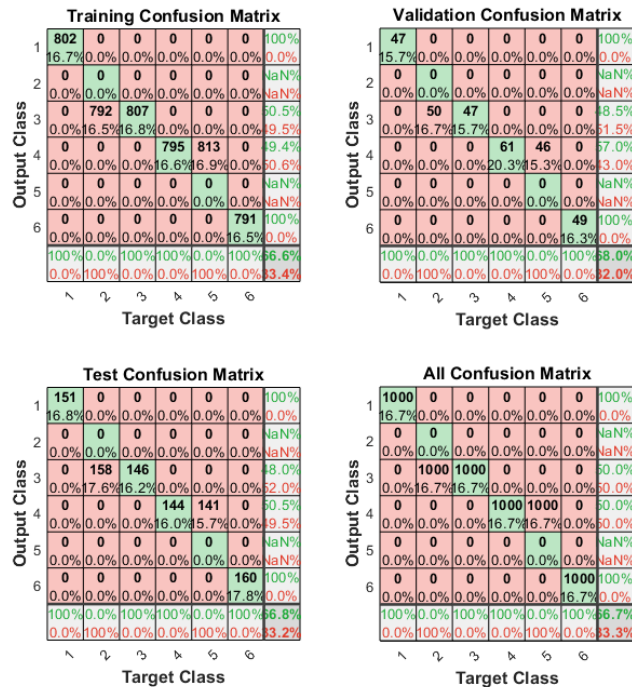


(Εικόνα 8 δομή δεδομένων εκπαίδευσης)

Αποτελέσματα Εκπαίδευσης

Έπειτα απο την εκπαίδευση και των δύο ΤΝΔ καταλήξαμε στο εξής εύρος αποτελεσμάτων(Εικόνες 9-13 περιέχουν μία προσομοίωση για το κάθε νευρωνικό) :

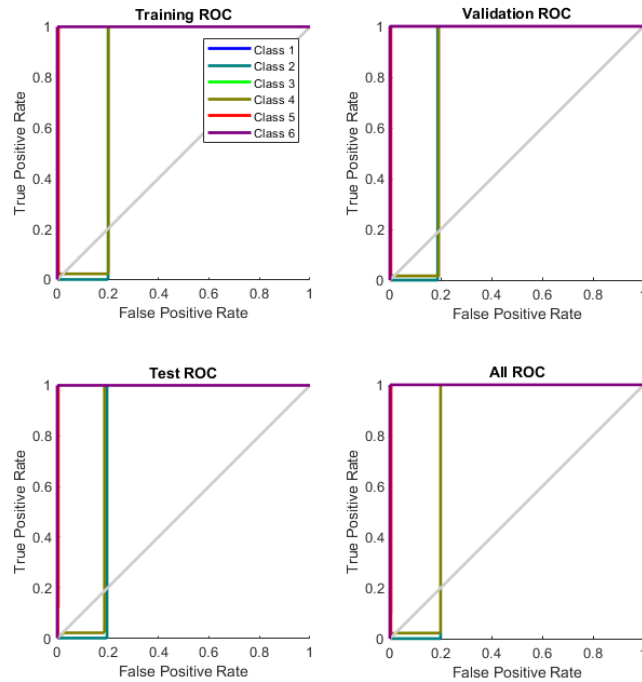
1 Hidden Layer Neural Net	Cross-Entropy Error Measure	Fraction of samples misclassified
Training:	2.5-6.5	28-39%
Validation:	9.6-13.5	29-39%
Testing:	9.9-13.7	32-40%
2 Hidden Layer Neural Net		
Training:	5.9-6.3	0%
Validation:	16.8-17.6	0%
Testing:	16.9-17.7	0%



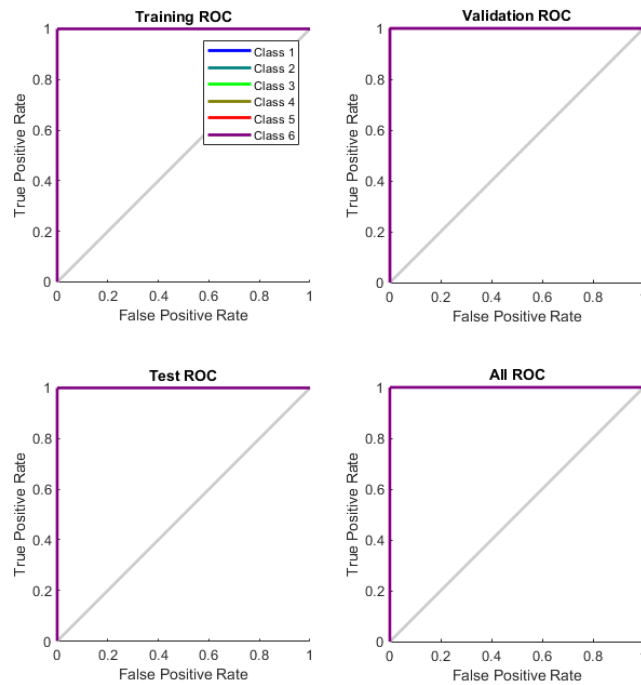
(Εικόνα 9 Plot Confusion για νευρωνικό δίκτυο με ένα Hidden Layer)



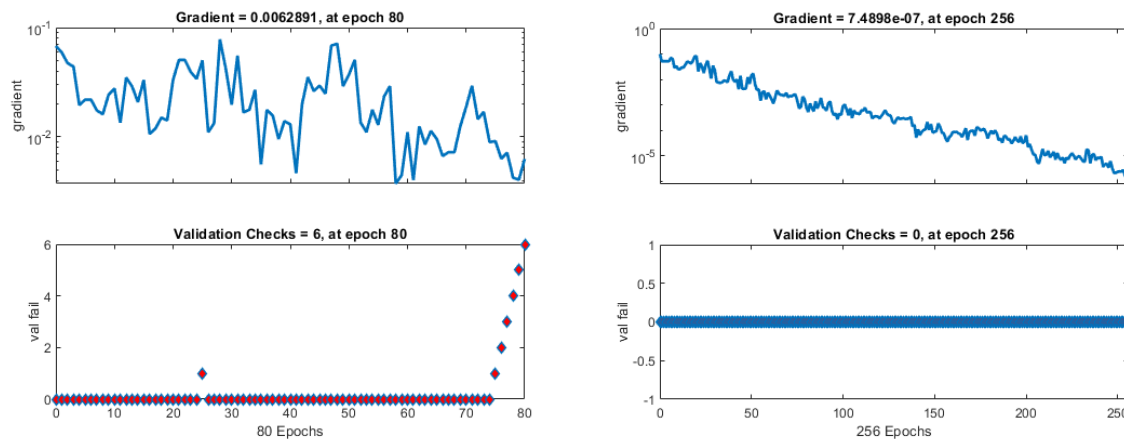
(Εικόνα 10 Plot Confusion για νευρωνικό δίκτυο με δύο Hidden Layers)



(Εικόνα 11 Plot ROC για νευρωνικό δίκτυο με ένα Hidden Layer)



(Εικόνα 12 Plot ROC για νευρωνικό δίκτυο με δύο Hidden Layers)



(Εικόνα 13 Training state of Neural Nets Gradients in correlation with epochs for 1 Layer and 2 Layers)

Συμπεράσματα

Τα δύο ΤΝΔ μας έδωσαν αρκετά αξιόπιστα αποτελέσματα και δεν εντοπίσαμε σημάδια overfitting, ωστόσο η παροχή περισσότερων patterns τόσο απλών όσο και σύνθετων σίγουρα θα καθιστούσε την δουλειά του ΤΝΔ πιο απαιτητική. Επιπρόσθετα παρατηρούμε ότι η το ΤΝΔ με ένα Hidden Layer, έχει κυρίως δυσκολία στον διαχωρισμό των Class 3-4-6 (Instability, Point Outside of Control Limits και

Instability and Grouping) μοτίβων τα οποία έχουν τις περισσότερες ομοιότητες μεταξύ τους. Μια από τις μελλοντικές αναβαθμίσεις που θα μπορούσαν να κάνουν πιο αξιόπιστο το σύστημα μας θα ήταν η άντληση πρωταρχικών μοτίβων από πραγματικά μηχανήματα κατεργασιών. Τέλος είναι αρκετά εύκολο με την δομή που έχει το πρόγραμμα μας στην Python να προστεθούν πληθώρα άλλων μεθόδων δεδομένων για τα μοτίβα όπως UCL, LCL και πρόσθεση χαρακτηριστικών (median, percentage change between the current and a prior element κτλ).