

ΠΑΝΕΠΙΣΤΗΜΙΟ ΙΩΑΝΝΙΝΩΝ  
ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ Η/Υ ΚΑΙ  
ΠΛΗΡΟΦΟΡΙΚΗΣ



Μεταπτυχιακό Μάθημα  
ΒΕΛΤΙΣΤΟΠΟΙΗΣΗ

Άσκηση 2

Ομάδα

Τάγκα Βιργινία, 418  
Τεριζή Χρυσούλα, 430

Δεκέμβριος 2020

# 1 Εισαγωγή

Στόχος της εργασίας είναι η εκπαίδευση ενός πιθανοτικού νευρωνικού δικτύου (Probabilistic Neural Network, PNN), το οποίο ενσωματώνει αλγορίθμους βελτιστοποίησης έτσι ώστε να μπορεί να προσαρμόζει από μόνο του (self-adaptive) την παράμετρο της διασποράς. Το PNN μπορεί να είναι είτε ομοσκεδαστικό (homoscedastic) είτε ετεροσκεδαστικός (heteroscedastic). Ομοσκεδαστικό στην Στατιστική σημαίνει ότι, μια ακολουθία τυχαίων μεταβλητών έχουν την ίδια διακύμανση ενώ ετεροσκεδαστικό σημαίνει ότι η διακύμανση μεταξύ των στοιχείων διαφέρει αντίστοιχα. Επομένως, σε ένα ομοσκεδαστικό PNN, η παράμετρος βελτιστοποίησης είναι μια μεταβλητή  $\sigma$  (spread parameter) ενώ σε ένα ετεροσκεδαστικό PNN η παράμετρος βελτιστοποίησης είναι ένας διαγώνιος πίνακας διασπορών  $\Sigma$  (matrix of spread parameters). Για την βελτιστοποίηση αυτών των μεταβλητών  $\sigma$  ή  $\Sigma$  χρησιμοποιούνται τέσσερις αλγόριθμοι βελτιστοποίησης και αυτοί είναι οι *Τυχαία Αναζήτηση* με χρήση *Gaussian κατανομής*, *Τυχαίος περίπατος* με *μεταβλητό βήμα*, *Nelder Mead* και *Simulated Annealing*. Επίσης, εφαρμόσαμε τις συγκεκριμένες τεχνικές στο σύνολο δεδομένων Breast Cancer Wisconsin χρησιμοποιώντας την τεχνική 10-CV (cross validation) για να διάσπαση του συνόλου δεδομένων σε σύνολο εκπαίδευσης και αξιολόγησης. Ως συνάρτηση βελτιστοποίησης θέτουμε την  $f(\sigma)$ , η οποία εξετάζει για το συγκεκριμένο  $\sigma$  το ποσοστό επιτυχημένης πρόβλεψης κατηγορίας (accuracy) στο σύνολο εκπαίδευσης. Έπειτα για το καλύτερο  $\sigma$  το οποίο έχει προκύψει από την εκπαίδευση με την τεχνική leave-one-out, δοκιμάζεται στο σύνολο αξιολόγησης. Επομένως, το πρόβλημα βελτιστοποίησης το οποίο καλούμαστε να επιλύσουμε είναι πρόβλημα μεγιστοποίησης.

Η αναφορά οργανώνεται ως εξής, Ενότητα 2 Μέθοδοι, μέσα στην οποία γίνεται μια σύντομη περιγραφή του πιθανοτικού νευρωνικού δικτύου PNN και των τεσσάρων μεθόδων βελτιστοποίησης οι οποίες έχουν προσαρμοστεί έτσι ώστε να ταιριάζουν με το πρόβλημα μεγιστοποίησης. Ακολουθεί η Ενότητα 3 Πειραματική αρχικοποίηση, μέσα στην οποία γίνεται αναφορά των δεδομένων που χρησιμοποιήθηκαν, της μεθόδου διαχωρισμού του συνόλου δεδομένων σε σύνολο εκπαίδευσης και αξιολόγησης καθώς επίσης και το πως έχουν αρχικοποιηθεί οι διάφοροι υπερ-παράμετροι από τις τέσσερις μεθόδους βελτιστοποίησης. Εν συνεχεία βρίσκεται η Ενότητα 4 στην οποία παρουσιάζονται και αναλύονται τα αποτελέσματα των πειραμάτων. Στο τέλος της αναφοράς παραθέτουμε τα εξής Παραρτήματα, Παράρτημα Α Στατιστικά, στο οποίο παραθέτουμε τα στατιστικά αποτελέσματα από κάθε μέθοδο βελτιστοποίησης σε πίνακες, Παράρτημα Β Box plots, στο οποίο απεικονίζονται τα box plots για κάθε μέθοδο βελτιστοποίησης, και τέλος, Παράρτημα C Εκτελέσιμα, στο οποίο περιγράφεται σύντομα η λειτουργία του κώδικα, τα εκτελέσιμα αρχεία καθώς επίσης και ο τρόπος εκτέλεσης του προγράμματος,

## 2 Μέθοδοι

Στην τρέχουσα Ενότητα γίνεται μια σύντομη αναφορά του τρόπου λειτουργίας των βασικών μεθόδων που χρησιμοποιούνται για την υλοποίηση ενός self-adaptive PNN με την χρήση αλγορίθμων βελτιστοποίησης.

### 2.1 PNN

Ένα πιθανοτικό νευρωνικό δίκτυο (PNN) είναι ένα νευρωνικό δίκτυο τεσσάρων επιπέδων το οποίο χρησιμοποιείται ευρέως σε προβλήματα ταξινόμησης και αναγνώρισης προτύπων. Στον αλγόριθμο PNN, η γονική συνάρτηση κατανομής πιθανότητας (PDF) κάθε κλάσης προσεγγίζεται από ένα παράθυρο Parzen και μια μη παραμετρική συνάρτηση. Στη συνέχεια, χρησιμοποιώντας PDF κάθε κλάσης, εκτιμάται η πιθανότητα κλάσης για νέα δεδομένα εισαγωγής και στη συνέχεια χρησιμοποιείται ο κανόνας Bayes για την κατανομή της κλάσης με την υψηλότερη οπίσθια πιθανότητα σε νέα δεδομένα εισαγωγής. Τα τέσσερα επίπεδα του PNN είναι τα εξής:

1. **Επίπεδο εισόδου.** Έστω  $x \in R^n$  το διάνυσμα εισόδου μεγέθους  $n$ . Περνάμε το διάνυσμα  $x$  σε κάθε έναν από τους νευρώνες του επόμενου επιπέδου (επίπεδο μοτίβου). Φτιάχνονται τόσοι τόσοι νευρώνες όσες είναι και οι κατηγορίες πρόβλεψης.
2. **Επίπεδο μοτίβου.** Ο κάθε νευρώνας περιέχει τα διανύσματα μοτίβων που ανήκουν στην συγκεκριμένη κατηγορία που δηλώνει ο συγκεκριμένος νευρώνας. Για το διάνυσμα εισόδου  $x$  υπολογίζεται η απόσταση του από κάθε έναν νευρώνα.

3. **Επίπεδο αθροίσματος.** Για τον κάθε νευρώνα υπολογίζεται μια προσέγγιση της συνάρτησης πιθανότητας (PDF).
4. **Επίπεδο εξόδου.** Σε αυτό το σημείο γίνεται η επιλογή της κατηγορίας για το διάνυσμα εισόδου  $x$ . Επιλέγεται ως κατηγορία εκείνη με την υψηλότερη σταθμισμένη τιμή PDF που έχει προκύψει από το προηγούμενο επίπεδο (επίπεδο αθροίσματος).

## 2.2 Μέθοδοι βελτιστοποίησης

### 2.2.1 Τυχαία Αναζήτηση με χρήση Gaussian κατανομής

Η τυχαία αναζήτηση περιλαμβάνει μεθόδους βελτιστοποίησης, γνωστές και ως μέθοδοι άμεσης αναζήτησης, χωρίς παραγώγους. Η μέθοδος βασίζεται στην παραγωγή τυχαίων αριθμών μέσα στο χώρο αναζήτησης, που ακολουθούν συγκεκριμένη συνεχή κατανομή πιθανότητας. Στην συγκεκριμένη εργασία έχει οριστεί ως κατανομή πιθανότητας η κανονική (gaussian) κατανομή. Η γενική μορφή της μεθόδου είναι:

$$x_{k+1} \sim \mathcal{N}(x_k, \sigma_k)$$

όπου  $\sigma_k$  είναι η τυπική απόκλιση της κανονική κατανομή και δεν θα πρέπει να υπάρχει σύγχυση με το  $\sigma$  που χρησιμοποιούμε ως παράμετρο βελτιστοποίησης. Θεωρούμε τυπική απόκλιση  $\sigma_k = 1$  (περισσότερες λεπτομέρειες για την επιλογή της συγκεκριμένης τιμής τυπικής απόκλισης περιγράφονται στην Ενότητα 3.3).

Έστω  $f(\mathbf{x})$  η αντικειμενική συνάρτηση που θέλουμε να μεγιστοποιήσουμε. Ο γενικός αλγόριθμος περιλαμβάνει τα ακόλουθα βήματα:

1. Αρχικοποίηση τυχαίου  $x \sim \mathcal{N}(0, 1)$  και υπολογισμός  $f(\mathbf{x})$ .
2. Έυρεση νέου τυχαίου  $x' \sim \mathcal{N}(0, 1)$  και υπολογισμός  $f(x')$ .
3. Εάν  $f(x') > f(\mathbf{x})$  τότε θέτουμε  $x = x'$  και επαναλαμβάνουμε τα βήματα 2 και 3 έως ότου φτάσουμε στον επιθυμητό αριθμό επαναλήψεων  $k_{max}$ .

### 2.2.2 Τυχαίος περίπατος με μεταβλητό βήμα

Ο τυχαίος περίπατος είναι μια στοχαστική διαδικασία που περιγράφει ένα μονοπάτι από τυχαία βήματα που πραγματοποιούνται μέσα σε ένα χώρο εξερεύνησης. Η γενική του μορφή είναι:

$$x_{k+1} = x_k + s p_k$$

όπου  $s \in R$  βήμα και  $p_k \in R^n$  τυχαία διεύθυνση μεγέθους  $n$ . Εδώ, το βήμα  $s$  αρχικά λαμβάνει μια προεπιλεγμένη τιμή έπειτα μεταβάλλεται (μεγαλώνει ή μικραίνει) αναλόγως με το αν έχουμε μεταβεί σε επιτυχημένα ή αποτυχημένα νέα σημεία αντίστοιχα. Επιτυχής θεωρείται το σημείο κατά το οποίο μεταβαίνουμε σε μια καλύτερη θέση, δηλαδή σε θέση που μεγιστοποιεί την αντικειμενική συνάρτηση.

Έστω  $f(\mathbf{x})$  η αντικειμενική συνάρτηση που θέλουμε να μεγιστοποιούμε. Ο γενικός αλγόριθμος περιλαμβάνει τα ακόλουθα βήματα:

1. Αρχικοποίηση τυχαίας διεύθυνσης  $p$ , βήματος  $s$  και θέσης  $x$ .
2. Υπολογισμός νέας θέσης  $x_t = x + s p$
3. Αν  $f(x_t) > f(x)$  τότε μεγαλώνουμε το βήμα πολλαπλασιάζοντας το με μια παράμετρο αύξησης  $a$  και προκύπτει το νέο βήμα  $s_{large} = a * s$ . Η νέα θέση  $x_{large}$  υπολογίζεται ως εξής  $x_{large} = x + s_{large} p$ .
4. Αν  $f(x_{large}) > f(x_t)$  τότε αντικαθιστούμε τη θέση  $x$  και το βήμα  $s$  με τα βέλτιστα  $x_{large}$  και  $s_{large}$  αντίστοιχα.
5. Διαφορετικά, αντικαθιστούμε τη θέση  $x$  με την  $x_t$  και το βήμα  $s$  παραμένει ίδιο.

6. Διαφορετικά, μικραίνουμε το βήμα  $s$  πολλαπλασιάζοντας το με την παράμετρο συρρίκνωσης  $\beta$  και το νέο βήμα που προκύπτει είναι το  $s_{small} = \beta * s$  και υπολογίζουμε τη νέα θέση  $x_{small} = x + s_{small} p$ .
7. Αν  $f(x_{small}) > f(x)$  τότε αντικαθιστούμε τη θέση  $x$  και το βήμα  $s$  με τα βέλτιστα  $x_{small}$  και  $s_{small}$  αντίστοιχα.
8. Ανανεώνουμε την θέση  $x$  με την καλύτερη θέση που υπολογίσαμε και επαναλαμβάνουμε τα βήματα 2-8 έως ότου ολοκληρωθεί ο επιθυμητός αριθμός επαναλήψεων  $k_{max}$ .

### 2.2.3 Nelder Mead

Η μέθοδος Nelder-Mead είναι μια μέθοδος άμεσης αναζήτησης που χρησιμοποιεί ένα γεωμετρικό σχήμα που ονομάζεται simplex. Ο simplex είναι η  $n$ -διάστατη έκδοση ενός «τριγώνου». Για μία διάσταση, ο simplex είναι μια γραμμή. Για 2 διαστάσεις, είναι ένα τρίγωνο. Για 3 διαστάσεις, είναι ένα τετράεδρο κοκ. Το σχήμα δεν χρειάζεται να είναι συμμετρικό ή ισόπλευρο. Ένας  $n$ -διάστατος simplex έχει  $(n + 1)$  κορυφές.

Ο Nelder–Mead ξεκινά με έναν τυχαία παραγόμενο simplex. Το αρχικό σημείο,  $x_1$  θεωρείται τυχαίο. Έχουμε  $n$  διαστάσεις και χρειαζόμαστε  $(n + 1)$  σημεία από τα οποία το 1 έχει ήδη αποφασιστεί. Τα άλλα τελικά σημεία αρχικοποιούνται σε σχέση με το  $x_1$ , σε μικρή απόσταση κατά την κατεύθυνση κάθε διάστασης. Σε κάθε επανάληψη, προχωρά στην αναδι-αμόρφωση / μετακίνηση αυτού του simplex, επηρεάζοντας μόνο μία κορυφή κάθε φορά, προς μια βέλτιστη περιοχή στον χώρο αναζήτησης. Κατά τη διάρκεια κάθε βήματος, δοκιμάζει μία ή περισσότερες τροποποιήσεις στον τρέχοντα simplex και επιλέγει μια που να τον μετατοπίζει προς μια «καλύτερη» περιοχή. Σε μια ιδανική περίπτωση, οι τελευταίες επαναλήψεις αυτού του αλγορίθμου συνεπάγονται τη συρρίκνωση του simplex προς τα μέσα, προς το καλύτερο σημείο. Στο τέλος, επιστρέφεται η κορυφή του simplex που αποδίδει την βέλτιστη τιμή της αντικειμενικής συνάρτησης. Ας υποθέσουμε ότι έχουμε έναν  $n$ -διάστατο χώρο. Ο τρέχον simplex αποτελείται από τα ακόλουθα  $(n + 1)$  σημεία:  $x_1, x_2, \dots, x_{(n+1)}$ . Η ακρίβεια (accuracy) που προσπαθούμε να βελτιστοποιήσουμε (μεγιστοποίηση) είναι συνάρτηση του  $\sigma$ . Ο αλγόριθμος ελέγχει τα ακόλουθα τρία βήματα σε κάθε επανάληψη:

#### I. Διάταξη

Όλα τα σημεία ταξινομούνται, έτσι ώστε η τιμή του  $f(x)$  για το πρώτο σημείο  $f(x_1)$  να είναι η καλύτερη (υψηλότερο accuracy) και για το τελευταίο σημείο να είναι η χειρότερη (χαμηλότερο accuracy),

$$f(x_1) \geq f(x_2) \geq \dots \geq f(x_{n+1})$$

Ορίζουμε τους δείκτες των πρώτων (καλύτερων), δεύτερων (καλών) και τελευταίων (χειρότερων) σημείων να είναι  $b$  (best),  $g$  (good) και  $w$  (worst) αντίστοιχα.

$$f(x_w) \leq f(x_g) \leq \dots \leq f(x_b)$$

#### II. Υπολογισμός του Κέντρου Βάρους

Κέντρο Βάρους των καλύτερων σημείων (μέσος όρος) ως:

$$c = \frac{1}{n} \sum_{i \neq w} x_i$$

#### III. Μετασχηματισμός Σημείων

##### 1. Ανάκλαση (Reflection)

Υπολογίζεται το σημείο ανάκλασης ως:

$$x_{ref} = c + \alpha(c - x_w)$$

Το  $\alpha$  ονομάζεται παράμετρος ανάκλασης και συνήθως ισούται με 1. Εάν  $f(x_b) \geq f(x_{ref}) > f(x_g)$  δηλαδή, το  $x_{ref}$  να είναι καλύτερο από το καλό σημείο, αλλά όχι καλύτερο από το τρέχον καλύτερο, αντικαθιστούμε το  $x_w$  με το  $x_{ref}$  στο simplex και προχωράμε στην επόμενη επανάληψη (βήμα I Διάταξη).

## 2. Επέκταση (Expansion)

Εάν το σημείο ανάκλησης  $x_{ref}$  είναι το καλύτερο από το τρέχον καλύτερο ( $f(x_{ref}) > f(x_b)$ ), κινούμαστε προς την κατεύθυνση του  $x_{ref}$  από το  $c$ , προσπαθώντας να βρούμε μια ακόμη καλύτερη λύση. Το σημείο επέκτασης ορίζεται ως:

$$x_{exp} = c + \gamma(x_{ref} - c)$$

Το  $\gamma$  ονομάζεται παράμετρος επέκτασης και η τιμή του είναι 2. Αν το σημείο επέκτασης είναι καλύτερο από το σημείο ανάκλησης,  $f(x_{exp}) > f(x_{ref})$ , τότε αντικαθιστούμε το χειρότερο σημείο με το σημείο επέκτασης και αποκτάμε ένα νέο simplex και μεταβαίνουμε στην επόμενη επανάληψη (βήμα I Διάταξη). Διαφορετικά, αποκτάμε νέο simplex, αντικαθιστώντας το  $x_w$  με το σημείο ανάκλησης  $x_{ref}$  και εκτελούμε στην επόμενη επανάληψη (βήμα I Διάταξη). Η επέκταση οδηγεί πάντα σε ένα μεγαλύτερο simplex, που σημαίνει περισσότερη εξερεύνηση του χώρου αναζήτησης.

## 3. Συστολή (Contraction)

Στην περίπτωση αυτή ισχύει:  $f(x_{ref}) \geq f(x_b)$ . Ας υποθέσουμε ότι το σημείο ανάκλησης ήταν χειρότερο από το  $x_g$  (δηλαδή το καλό σημείο). Σε αυτήν την περίπτωση, η κατεύθυνση που ορίζεται από το  $x_{ref}$  μπορεί να μην είναι η ιδανική για κίνηση. Ως εκ τούτου, καταλήγουμε να συστέλλουμε το simplex. Το σημείο συστολής  $x_{con}$  ορίζεται ως:

$$x_{con} = c + \beta(x_w - c)$$

Το  $\beta$  ονομάζεται παράμετρος συστολής και συνήθως ισούται με 0.5 όταν αναφερόμαστε στην εξωτερική συστολή και ίση με  $-0.5$  όταν αναφερόμαστε στην εσωτερική συστολή. Εάν το σημείο συστολής είναι καλύτερο από το χειρότερο σημείο δηλ.  $f(x_{con}) > f(x_w)$  τότε αντικαθιστούμε το χειρότερο σημείο  $x_w$  με το σημείο συστολής  $x_{con}$  και αποκτάμε ένα νέο simplex. Πηγαίνουμε έπειτα στο βήμα I Διάταξη και συνεχίζουμε με την επόμενη επανάληψη.

## 4. Συρρίκνωση (Shrink)

Σε αυτήν την περίπτωση, επαναπροσδιορίζουμε ολόκληρο το simplex. Διατηρούμε μόνο το καλύτερο σημείο ( $x_b$ ) και ορίζουμε όλα τα άλλα σημεία σε σχέση με αυτό και τα προηγούμενα σημεία. Το νέο σημείο  $j^{th}$ , θα οριστεί τώρα ως:

$$x_j = x_b + \delta(x_j - x_b)$$

Το  $\delta$  ονομάζεται παράμετρος συρρίκνωσης και ισούται με 0,5. Στην περίπτωση της συρρίκνωσης, κάθε σημείο στο simplex μετακινείται προς το τρέχον καλύτερο σημείο, με την ελπίδα να συγκλίνει στην καλύτερη γειτονιά.

Ο τερματισμός επιτυγχάνεται όταν έχει ολοκληρωθεί ένας συγκεκριμένος αριθμός επαναλήψεων  $k_{max}$ .

### 2.2.4 Simulated Annealing

Η προσομοιωμένη ανόπτηση είναι άλλη μία τεχνική βελτιστοποίησης. Ο αλγόριθμος προσομοιωμένης ανόπτησης αρχικά βασίστηκε στη διαδικασία ανόπτησης στη μεταλλουργία. Η ανόπτηση περιλαμβάνει θέρμανση και ψύξη ενός υλικού με σκοπό την αλλαγή των φυσικών του ιδιοτήτων. Καθώς το μέταλλο ψύχεται, η νέα του δομή στερεώνεται, προκαλώντας κατά συνέπεια το μέταλλο να διατηρήσει τις νέες του ιδιότητες. Στην προσομοίωση ανόπτησης διατηρούμε μια μεταβλητή θερμοκρασίας για την προσομοίωση αυτής της διαδικασίας. Αρχικά θέτουμε υψηλή θερμοκρασία και στη συνέχεια την αφήνουμε να μειωθεί αργά καθώς εκτελείται ο αλγόριθμος. Αυτή η έννοια της αργής ψύξης που εφαρμόζεται στον αλγόριθμο προσομοιωμένης ανόπτησης ερμηνεύεται ως μια αργή μείωση της πιθανότητας αποδοχής χειρότερων λύσεων καθώς διερευνάται ο χώρος λύσης. Η αποδοχή χειρότερων λύσεων επιτρέπει μια πιο εκτεταμένη αναζήτηση για την ολική βέλτιστη λύση. Γενικά, οι προσομοιωμένοι αλγόριθμοι ανόπτησης λειτουργούν ως εξής. Η θερμοκρασία μειώνεται προοδευτικά από μια αρχική θετική τιμή στο μηδέν. Σε κάθε βήμα, ο αλγόριθμος επιλέγει τυχαία μια λύση κοντά στην τρέχουσα, μετρά την ποιότητά της και

μετακινείται σε αυτήν σύμφωνα με τις πιθανότητες επιλογής καλύτερων ή χειρότερων λύσεων. Οι πιθανότητες αυτές εξαρτώνται από τη θερμοκρασία και κατά τη διάρκεια της αναζήτησης παραμένουν αντίστοιχα στο 1 (ή σε θετικό αριθμό) και μειώνονται προς το μηδέν.

Έστω  $f(\mathbf{x})$  η αντικειμενική συνάρτηση που θέλουμε να ελαχιστοποιήσουμε. Ο γενικός αλγόριθμος περιλαμβάνει τα ακόλουθα βήματα:

1. Προσδιορισμός σημείου  $y_{k+1} = x_k + p * s_k$ , όπου  $x_k$  η προηγούμενη θέση,  $p > 0$  σταθερό βήμα και  $s_k$  τυχαία διεύθυνση με μέτρο  $\|s_k\| = 1$
2. Υπολογισμός συνάρτησης αποδοχής Metropolis:

$$Facc(x, y, t) = \min\{1, e^{\frac{-(f(y)-f(x))}{t}}\} \quad , t \text{ θερμοκρασία}$$

3. Εύρεση τυχαίου αριθμού  $\text{random} \sim \mathcal{U}(0, 1)$  και σύγκριση με συνάρτηση  $Facc(x_k, y_{k+1}, t_k)$
4. Εάν  $\text{random} \leq Facc$  τότε αντικαθιστούμε τη νέα θέση  $x_{k+1} = y_{k+1}$
5. Διαφορετικά,  $x_{k+1} = x_k$
6. Ανανεώνουμε το σύνολο σημείων  $Z_{k+1} = Z_k \cup \{y_{k+1}\}$
7. Επιλέγουμε διαδικασία ψύξης  $U(Z_{k+1}) = \beta(f(x_{k+1}) - f^*)^\alpha$  και μειώνουμε τη θερμοκρασία  $t_{k+1}$ . Τα  $\alpha, \beta$  είναι σταθερές θετικές παράμετροι ελέγχου και η  $f^*$  η ελάχιστη τιμή της  $f$  που ανανεώνεται αν βρεθεί κάποια μικρότερη.
8. Ανανεώνουμε τη θέση  $x$  με την καλύτερη θέση που υπολογίσαμε και επαναλαμβάνουμε τα βήματα 1-8 έως ότου ολοκληρωθεί ο επιθυμητός αριθμός επαναλήψεων  $k_{max}$ .

### 3 Πειραματική αρχικοποίηση

Στην τρέχουσα ενότητα αρχικά γίνεται μια περιγραφή του συνόλου δεδομένων με το οποίο εργαστήκαμε και το τι είδους επεξεργασία του έχουμε υποβάλλει. Στην συνέχεια, γίνεται αναφορά για τον τρόπο διάσπασης του συνόλου δεδομένων σε σύνολο εκπαίδευσης και αξιολόγησης, το σύνολο των πειραμάτων που έχουμε πραγματοποιήσει για τις διαφορετικές αρχικοποιήσεις των υπερπαραμέτρων για την κάθε μέθοδο βελτιστοποίησης που έχουμε εξετάσει καθώς επίσης και οι τιμές που επιλέχθηκαν για το συνολικό πλήθος συναρτησιακών υπολογισμών.

#### 3.1 Σύνολο Δεδομένων

Το dataset που χρησιμοποιήθηκε προέρχεται από το UCI Machine Learning Repository, Breast Cancer Wisconsin (Original) Data Set<sup>1</sup> για τον καρκίνο του μαστού. Περιλαμβάνει 699 δείγματα με 10 χαρακτηριστικά, τα οποία περιγράφονται στον Πίνακα 1 από τα οποία χρησιμοποιήθηκαν τα 9 χαρακτηριστικά (Feature 1 - Feature 9). Το πρώτο χαρακτηριστικό αναφέρεται στον κωδικό (ID) της εγγραφής και για αυτό δεν λήφθηκε υπόψιν. Το κάθε δείγμα του συνόλου μπορεί να κατηγοριοποιηθεί σε δύο είδη (κατηγορία), είτε σε καλοήγη μορφή είτε σε κακοήγη μορφή. Στο σύνολο δεδομένων εμφανίζονται 16 εγγραφές στις οποίες δεν υπάρχει η πληροφορία του Feature 6. Για αυτές τις εγγραφές θέτουμε μια τυχαία τιμή από το εύρος τιμών (1 - 10) του συγκεκριμένου χαρακτηριστικού.

Για να μπορέσουμε να κατανοήσουμε σε ποια από τις δύο κατηγορίες **ομοσκεδαστικό** ή **ετεροσκεδαστικό** ανήκει το πρόβλημα μελετήσαμε αρχικά τα δεδομένα για κάθε ένα από τα 9 χαρακτηριστικά του συνόλου δεδομένων που διαθέτουμε. Συγκεκριμένα, στους Πίνακες 2 και 3 βλέπουμε για κάθε χαρακτηριστικό την ελάχιστη (Min), μέγιστη (Max) και μέση (Mean) τιμή, την διακύμανση (Variance) και τυπική απόκλιση (Standard deviation) των δεδομένων χωρίς κάποιου είδους κανονικοποίηση (γνήσια δεδομένα) και με κανονικοποίηση αντίστοιχα. Παρατηρούμε ότι, η διακύμανση για κάθε χαρακτηριστικό στα γνήσια δεδομένα διαφέρει επομένως, το πρόβλημα μας είναι ετεροσκεδαστικό διότι για κάθε σημείο του διαγωνίου πίνακα  $\Sigma$  θα πρέπει να αρχικοποιηθεί και ένα διαφορετικό  $\sigma_i, i \in [1, 9]$ . Αντίστοιχα, στον Πίνακα

<sup>1</sup><https://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+%28Original%29>

Table 1: Παρουσιάζονται τα χαρακτηριστικά πεδία κάθε εγγραφής από το σύνολο δεδομένων, το εύρος τιμών κάθε χαρακτηριστικού και η τελική αντιστοίχιση τους.

| Χαρακτηριστικά                      | Εύρος τιμών              | Αντιστοίχιση σε          |
|-------------------------------------|--------------------------|--------------------------|
| Κωδικός δείγματος (ID)              | 1 - 10                   | Αγνοήθηκε                |
| Πάχος συστάδας                      | 1 - 10                   | Feature 1                |
| Ομοιομορφία μεγέθους κυψέλης        | 1 - 10                   | Feature 2                |
| Ομοιομορφία του κυτταρικού σχήματος | 1 - 10                   | Feature 3                |
| Οριακή πρόσφυση                     | 1 - 10                   | Feature 4                |
| Ενιαίο μέγεθος επιθηλιακού κελιού   | 1 - 10                   | Feature 5                |
| Bare πυρήνες                        | 1 - 10                   | Feature 6                |
| Χρωματοτίνη Bland                   | 1 - 10                   | Feature 7                |
| Κανονικός πυρήνας                   | 1 - 10                   | Feature 8                |
| Μιτώσεις                            | 1 - 10                   | Feature 9                |
| Κατηγορία                           | 2 καλοήθης<br>4 κακοήθης | 1 καλοήθης<br>2 κακοήθης |

3 όπου τα δεδομένα δέχονται κανονικοποίηση, η διακύμανση κάθε χαρακτηριστικού είναι ίδια και ίση με την μονάδα. Επομένως, το πρόβλημα μας μπορεί να μετατραπεί σε ομοσκεδαστικό, όπου κάθε στοιχείο  $\sigma_i$  του διαγώνιου πίνακα  $\Sigma$  θα παίρνει ίδια τιμή δηλαδή  $\Sigma = \sigma^2 I$  όπου  $\sigma_i = \sigma, i \in [1, 9]$ .

Επομένως, το τελικό σύνολο δεδομένων με το οποίο εργαστήκαμε αποτελείται από 699 εγγραφές με 9 χαρακτηριστικά η κάθε εγγραφή και των οποίων οι τιμές είναι **κανονικοποιημένες**.

Table 2: Ελάχιστη, μέγιστη, μέση τιμή, διακύμανση και τυπική απόκλιση κάθε χαρακτηριστικού στα δεδομένα χωρίς κανονικοποίηση.

|           | Min | Max | Mean | Variance | Standard deviation |
|-----------|-----|-----|------|----------|--------------------|
| Feature 1 | 1   | 10  | 4.41 | 7.91     | 2.81               |
| Feature 2 | 1   | 10  | 3.13 | 9.29     | 3.04               |
| Feature 3 | 1   | 10  | 3.20 | 8.18     | 2.96               |
| Feature 4 | 1   | 10  | 2.80 | 8.14     | 2.85               |
| Feature 5 | 1   | 10  | 3.21 | 4.89     | 2.21               |
| Feature 6 | 1   | 10  | 3.61 | 13.27    | 3.65               |
| Feature 7 | 1   | 10  | 3.43 | 5.93     | 2.43               |
| Feature 8 | 1   | 10  | 2.86 | 9.31     | 3.05               |
| Feature 9 | 1   | 10  | 1.58 | 2.93     | 1.71               |

Table 3: Ελάχιστη, μέγιστη, μέση τιμή, διακύμανση και τυπική απόκλιση κάθε χαρακτηριστικού στα δεδομένα χωρίς κανονικοποίηση.

|           | Min   | Max  | Mean | Variance | Standard deviation |
|-----------|-------|------|------|----------|--------------------|
| Feature 1 | -1.21 | 1.98 | 0.0  | 1.0      | 1.0                |
| Feature 2 | -0.69 | 2.25 | 0.0  | 1.0      | 1.0                |
| Feature 3 | -0.74 | 2.28 | 0.0  | 1.0      | 1.0                |
| Feature 4 | -0.63 | 2.52 | 0.0  | 1.0      | 1.0                |
| Feature 5 | -1.0  | 3.06 | 0.0  | 1.0      | 1.0                |
| Feature 6 | -0.70 | 1.77 | 0.0  | 1.0      | 1.0                |
| Feature 7 | -1.0  | 2.69 | 0.0  | 1.0      | 1.0                |
| Feature 8 | -0.61 | 2.33 | 0.0  | 1.0      | 1.0                |
| Feature 9 | -0.34 | 4.90 | 0.0  | 1.0      | 1.0                |

3.2 Διάσπαση σε σύνολο εκπαίδευσης - αξιολόγησης

Η τεχνική που επιλέχθηκε για να γίνει η διάσπαση του συνόλου δεδομένων σε σύνολο εκπαίδευσης και αξιολόγησης είναι η k-fold cross validation. Συγκεκριμένα, επιλέξαμε 10-CV και για κάθε ένα από αυτά τα 10-fold πραγματοποιήσαμε 10 πειράματα.

### 3.3 Αρχικοποίηση υπερπαραμέτρων

Κάθε μέθοδος βελτιστοποίησης αποτελείται από ένα σύνολο υπερπαραμέτρων, οι οποίες αρχικοποιούνται από τον χρήστη. Δοκιμάσαμε διάφορες αρχικές τιμές για τις υπερπαραμέτρους αυτές οι οποίες παρουσιάζονται παρακάτω.

**Τυχαία Αναζήτηση με χρήση Gaussian κατανομής.** Η συγκεκριμένη μέθοδος δεν δέχεται κάποια συγκεκριμένη υπερπαραμέτρο. Η τιμή η οποία χρειάζεται να προσδιοριστεί είναι η τυπική απόκλιση με την οποία η Gaussian κατανομή παράγει νέες τιμές για την μεταβλητή  $\sigma$ . Θέτουμε την τυπική απόκλιση ίση με 1 βάση και των αποτελεσμάτων που προέκυψαν μετά την κανονικοποίηση των δεδομένων. Επομένως, το μοναδικό κριτήριο αξιολόγησης της μεθόδου είναι ανάλογα με το διαφορετικό πλήθος των συνολικών συναρτησιακών υπολογισμών που θέτουμε.

**Τυχαίος περίπατος με μεταβλητό βήμα.** Η μέθοδος δέχεται τρεις υπερπαραμέτρους. Αυτές είναι οι  $\alpha$ ,  $\beta$  και  $s$  όπως αναφέρεται στην Ενότητα 2.2.2, οι οποίες εκφράζουν παράμετρο αύξησης, παράμετρο συρρίκνωσης και βήμα αντίστοιχα. Συγκεκριμένα, εξετάσαμε συνολικά 8 διαφορετικές εκδοχές της μεθόδου οι οποίες προκύπτουν τις ακόλουθες αρχικοποιήσεις,

$$\alpha = [1.5, 2.0], \beta = [0.7, 0.5], s = [0.1, 0.5]$$

**Nelder Mead.** Η μέθοδος δέχεται τέσσερις υπερπαραμέτρους όπως περιγράφεται και στην Ενότητα 2.2.3, αυτές είναι οι  $\alpha$ ,  $\beta_{in}$ ,  $\beta_{out}$ ,  $\gamma$  και  $\delta = 0.5$  και κάθε μια αντιστοιχίζεται σε κάθε είδος βήματος, ανάκλασης, εσωτερικής συστολής, εξωτερικής συστολής, επέκτασης και συρρίκνωσης. Για τις τέσσερις αυτές υπερπαραμέτρους ισχύει το εξής,

$$-1 < \beta_{out} < 0 < \beta_{in} < \alpha < \gamma$$

Η κάθε υπερπαραμέτρος παίρνει τιμές προκαθορισμένες. Επομένως εξετάσαμε μια μόνο εκδοχή για την συγκεκριμένη μέθοδο βελτιστοποίησης και αυτή είναι η εξής,

$$\beta_{out} = -0.5, \beta_{in} = 0.5, \alpha = 1 \text{ και } \gamma = 2$$

**Simulated Annealing.** Η μέθοδος δέχεται δυο υπερπαραμέτρους όπως περιγράφεται και στην Ενότητα 2.2.4, αυτές είναι οι  $\alpha$  και  $\beta$  οι οποίες χρειάζονται για την ανανέωση της τιμής της θερμοκρασίας ή διαφορετικά την πιθανότητα αποδοχής σημείων τα οποία οδηγούν σε αύξηση (ή μείωση) της συναρτησιακής τιμής (όταν δηλώνει ποσοστό επιτυχημένης πρόβλεψης). Συγκεκριμένα, εξετάσαμε συνολικά 6 διαφορετικές εκδοχές της μεθόδου οι οποίες προκύπτουν τις ακόλουθες αρχικοποιήσεις,

$$\alpha = [1.0, 2.0], \beta = [1.0, 5.0, 10.0]$$

### 3.4 Πλήθος συναρτησιακών υπολογισμών

Οι μέθοδοι βελτιστοποίησης οι οποίες εξετάζονται στην τρέχουσα εργασία ανήκουν στην κατηγορία των στοχαστικών μεθόδων. Σε τέτοιου είδους μεθόδους υπάρχουν δύο είδη πειραμάτων, τα οποία είναι *Εύρεση Καλύτερης Λύσης σε Δοθέντα Χρόνο* και *Απαιτούμενος Χρόνος για Εύρεση Λύσης Συγκεκριμένης Ποιότητας*. Στην τρέχουσα εργασία έχουμε επιλέξει το πρώτο είδος πειραμάτων και θέτουμε ως προκαθορισμένο χρόνο, το συνολικό πλήθος συναρτησιακών υπολογισμών,  $k_{max} = 100$  όπου εξετάζουμε ανά 10 συναρτησιακούς υπολογισμούς την εξέλιξη των μεθόδων. Δηλαδή, εξετάζουμε  $k_{max} = [10, 20, \dots, 100]$ .

## 4 Αποτελέσματα

Στην τρέχουσα ενότητα παρουσιάζουμε τα αποτελέσματα που προέκυψαν έπειτα από τα εκτέλεση ενός πλήθους πειραμάτων. Συγκεκριμένα, η ενότητα οργανώνεται ως εξής, αρχικά εξετάζουμε την καλύτερη εκδοχή της κάθε μεθόδου βελτιστοποίησης βάση των διαφορετικών αρχικοποιήσεων των υπερπαραμέτρων και των διαφορετικών τιμών συνολικών συναρτησιακών υπολογισμών που εξετάζουμε. Έπειτα, έχοντας καταλήξει στην καλύτερη εκδοχή κάθε μεθόδου βελτιστοποίησης γίνεται σύγκριση των καλύτερων μεθόδων μεταξύ τους καθώς επίσης και σύγκριση και με δημοσιευμένα αποτελέσματα.



## 4.1 Εξαγωγή καλύτερης εκδοχής κάθε μεθόδου

Αρχικά, για κάθε ένα από τα 10 πειράματα και για κάθε ένα από τα 10-CV και για κάθε διαφορετική τιμή συνολικών συναρτησιακών υπολογισμών (10, 20, ..., 100) εξετάζουμε το καλύτερο (υψηλότερο) ποσοστό επιτυχημένης πρόβλεψης κατηγορίας (accuracy). Επομένως, για κάθε διαφορετική τιμή  $k_{max}$  διατηρούμε την καλύτερη πληροφορία (accuracy) για αυτό και για αυτές τις τιμές εξετάζουμε τις καλύτερες εκδοχές των τεσσάρων μεθόδων βελτιστοποίησης.

### 4.1.1 Τυχαία Αναζήτηση με χρήση Gaussian κατανομής

Αρχικά, στον Πίνακα 7 παραθέτουμε κάποιες στατιστικές τιμές, όπως μέση τιμή (Mean), ενδιάμεση τιμή (Median), τυπική απόκλιση (St.Dev), ελάχιστη (Min), μέγιστη (Max) και 95% διάστημα εμπιστοσύνης (CI low και CI up), που προκύπτουν για την μέθοδο *Τυχαίας Αναζήτησης* με χρήση *Gaussian* κατανομής για διαφορετικές τιμές συνολικού πλήθους συναρτησιακών υπολογισμών  $k_{max}$  και διατηρώντας κάθε φορά για το κάθε  $k_{max}$  τις καλύτερες τιμές accuracy που προκύπτουν από όλα τα πειράματα. Παρατηρούμε ότι ακόμα και από τους πρώτους 10 συναρτησιακούς υπολογισμούς πετυχαίνεται ελάχιστη τιμή επιτυχίας σωστής πρόβλεψης κατηγορίας (accuracy) ίση με 0.913 καθώς επίσης και μέγιστη τιμή ίση με 1.0 για το σύνολο αξιολόγησης. Επίσης, παρόλο που αυξάνουμε τους συναρτησιακούς υπολογισμούς, από 10 σε 20 και ούτω καθεξής μέχρι τους συνολικά 100 συναρτησιακούς υπολογισμούς παρατηρούμε ότι δεν έχει υπάρξει κάποια βελτίωση του ποσοστού επιτυχίας σωστής πρόβλεψης (accuracy).

Στην Εικόνα 1 απεικονίζονται τα αντίστοιχα box plots που προκύπτουν από τον Πίνακα 7. Με μια πρώτη ματιά φαίνεται ότι δεν υπάρχει κάποια διαφορά είτε χρησιμοποιήσουμε 10 είτε 100 συνολικά συναρτησιακούς υπολογισμούς. Εφαρμόζοντας και έλεγχο στατιστικής σημαντικότητας 95% παρατηρούμε ότι δεν εμφανίζεται διαφορά ανάμεσα από 60 μέχρι 100 συνολικά συναρτησιακών υπολογισμών.

### 4.1.2 Τυχαίος περίπατος με μεταβλητό βήμα

Αρχικά, έχουμε εξετάσει συνολικά 8 διαφορετικές εκδοχές της μεθόδου για διαφορετικές τιμές των υπερπαραμέτρων της μεθόδου όπως έχουμε ήδη αναφέρει στην Ενότητα 3.3. Για κάθε μια εκδοχή εξετάζουμε ποια τιμή  $k_{max}$  αποδίδει καλύτερα. Αρχικά, στην Εικόνα 3 απεικονίζονται τα box plots για κάθε εκδοχή του αλγορίθμου για τις διαφορετικές τιμές  $k_{max}$  που εξετάζουμε. Με μια πρώτη ματιά παρατηρούμε ότι ίσως να μην υπάρχει κάποια διαφορά ανάμεσα στο διαφορετικό πλήθος συνολικών συναρτησιακών πράξεων ( $k_{max}$ ) πέρα από την περίπτωση όπου η παράμετρος επέκτασης είναι ίση με  $\alpha = 1.5$ ,  $\beta = 0.7$  και  $s = 0.1$  (Εικόνα 3 (c)) όπου το εύρος του accuracy ξεκινάει από 0.88 ενώ στις υπόλοιπες περιπτώσεις από 0.91. Εφαρμόζοντας και έλεγχο στατιστικής σημαντικότητας 95% σε κάθε περίπτωση παρατηρούμε ότι δεν εμφανίζεται κάποια σημαντική διαφορά όταν το  $k_{max}$  αλλάζει. Συγκεκριμένα για κάθε μια εκδοχή ( $\alpha$ ,  $\beta$ ,  $s$ ) έχουμε τα ακόλουθα,

- (1.5, 0.5, 0.1) → Για κάθε  $k_{max}$  η ελάχιστη τιμή accuracy είναι 0.913 ενώ η μέγιστη τιμή είναι 1.0. Το CI low και CI up κυμαίνονται από 0.95 έως 0.97. Δηλαδή, δεν εντοπίστηκε κάποια βελτίωση στο accuracy καθώς αυξήσαμε το πλήθος των επαναλήψεων.
- (1.5, 0.5, 0.5) → Για κάθε  $k_{max}$  η ελάχιστη τιμή accuracy είναι 0.913 ενώ η μέγιστη τιμή είναι 1.0. Το CI low και CI up κυμαίνονται από 0.95 έως 0.96. Δηλαδή, δεν εντοπίστηκε κάποια βελτίωση στο accuracy καθώς αυξήσαμε το πλήθος των επαναλήψεων.
- (1.5, 0.7, 0.1) → Για κάθε  $k_{max}$  η ελάχιστη τιμή accuracy είναι 0.88 ενώ η μέγιστη τιμή είναι 1.0. Το CI low και CI up κυμαίνονται από 0.95 έως 0.96. Δηλαδή, δεν εντοπίστηκε κάποια βελτίωση στο accuracy καθώς αυξήσαμε το πλήθος των επαναλήψεων.
- (1.5, 0.7, 0.5) → Για κάθε  $k_{max}$  η ελάχιστη τιμή accuracy είναι 0.913 ενώ η μέγιστη τιμή είναι 1.0. Το CI low και CI up κυμαίνονται από 0.95 έως 0.96. Δηλαδή, δεν εντοπίστηκε κάποια βελτίωση στο accuracy καθώς αυξήσαμε το πλήθος των επαναλήψεων.
- (2, 0.5, 0.1) → Για κάθε  $k_{max}$  η ελάχιστη τιμή accuracy είναι 0.913 ενώ η μέγιστη τιμή είναι 1.0. Το CI low και CI up κυμαίνονται από 0.95 έως 0.97. Δηλαδή, δεν εντοπίστηκε κάποια βελτίωση στο accuracy καθώς αυξήσαμε το πλήθος των επαναλήψεων.

- (2, 0.5, 0.5) → Για κάθε  $k_{max}$  η ελάχιστη τιμή accuracy είναι 0.913 ενώ η μέγιστη τιμή είναι 1.0. Το CI low και CI up κυμαίνονται από 0.95 έως 0.96. Δηλαδή, δεν εντοπίστηκε κάποια βελτίωση στο accuracy καθώς αυξήσαμε το πλήθος των επαναλήψεων.
- (2, 0.7, 0.1) → Για κάθε  $k_{max}$  η ελάχιστη τιμή accuracy είναι 0.913 ενώ η μέγιστη τιμή είναι 1.0. Το CI low και CI up κυμαίνονται από 0.95 έως 0.97. Δηλαδή, δεν εντοπίστηκε κάποια βελτίωση στο accuracy καθώς αυξήσαμε το πλήθος των επαναλήψεων. (Πίνακας 9)
- (2, 0.7, 0.5) → Για κάθε  $k_{max}$  η ελάχιστη τιμή accuracy είναι 0.913 ενώ η μέγιστη τιμή είναι 1.0. Το CI low και CI up κυμαίνονται από 0.95 έως 0.96. Δηλαδή, δεν εντοπίστηκε κάποια βελτίωση στο accuracy καθώς αυξήσαμε το πλήθος των επαναλήψεων.

Στην συνέχεια, συγκρίνοντας τις διαφορετικές εκδοχές της μεθόδου μεταξύ τους με Kruskal-Wallis τεστ, παρατηρούμε ότι η καλύτερη εκδοχή είναι εκείνη με αρχικοποίηση υπερπαραμέτρων  $\alpha = 2$ ,  $\beta = 0.7$  και  $s = 0.1$ . Στον Πίνακα 9 παρουσιάζονται τα στατιστικά της καλύτερης εκδοχής.

#### 4.1.3 Nelder Mead

Αρχικά, στον Πίνακα 8 παραθέτουμε κάποιες στατιστικές τιμές που προκύπτουν για την μέθοδο *Nelder-Mead* για διαφορετικές τιμές συνολικού πλήθους συναρτησιακών υπολογισμών  $k_{max}$  και διατηρώντας κάθε φορά για το κάθε  $k_{max}$  τις καλύτερες τιμές accuracy που προκύπτουν από όλα τα πειράματα. Παρατηρούμε ότι ακόμα και από τους πρώτους 10 συναρτησιακούς υπολογισμούς πετυχαίνεται ελάχιστη τιμή επιτυχίας σωστής πρόβλεψης κατηγορίας (accuracy) ίση με 0.884 καθώς επίσης και μέγιστη τιμή ίση με 1.0 για το σύνολο αξιολόγησης. Επίσης, παρόλο που αυξάνουμε τους συναρτησιακούς υπολογισμούς, από 10 σε 20 και ούτω καθεξής μέχρι τους συνολικά 100 συναρτησιακούς υπολογισμούς παρατηρούμε ότι δεν έχει υπάρξει κάποια βελτίωση του ποσοστού επιτυχίας σωστής πρόβλεψης (accuracy).

Στην Εικόνα 2 απεικονίζονται τα αντίστοιχα box plots που προκύπτουν από τον Πίνακα 8. Με μια πρώτη ματιά φαίνεται ότι δεν υπάρχει κάποια διαφορά είτε χρησιμοποιήσουμε 10 είτε 100 συνολικά συναρτησιακούς υπολογισμούς. Εφαρμόζοντας και έλεγχο στατιστικής σημαντικότητας 95% παρατηρούμε ότι δεν εμφανίζεται διαφορά ανάμεσα από 10 μέχρι 100 συνολικά συναρτησιακών υπολογισμών.

#### 4.1.4 Simulated Annealing

Αρχικά, έχουμε εξετάσει συνολικά 6 διαφορετικές εκδοχές της μεθόδου για διαφορετικές τιμές των υπερπαραμέτρων της μεθόδου όπως έχουμε ήδη αναφέρει στην Ενότητα 3.3. Για κάθε μια εκδοχή εξετάζουμε ποια τιμή  $k_{max}$  αποδίδει καλύτερα και έπειτα κάνουμε την σύγκριση ανάμεσα στις καλύτερες εκδοχές του αλγορίθμου. Στην Εικόνα 4 απεικονίζονται τα box plots για κάθε εκδοχή του αλγορίθμου για τις διαφορετικές τιμές  $k_{max}$  που εξετάζουμε. Παρατηρούμε ότι, σε δυο περιπτώσεις, όταν  $\alpha = 2.0$ ,  $\beta = 1.0$  και όταν  $\alpha = 1.0$ ,  $\beta = 5.0$  (Εικόνες 4 (b) και (c)), εμφανίζονται τιμές εκτός των box plots για κάθε  $k_{max}$  που εξετάζουμε. Επίσης, όταν η παράμετρος  $\beta$  ισούται με  $\beta = 10.0$ , το ύψος των box plots αυξάνεται, δηλαδή η μέθοδος πετυχαίνει accuracies τα οποία είναι αρκετά χαμηλά γύρω στο 0.60 (Εικόνες 4 (d), (e) και (f)). Για κάθε εκδοχή έχουμε εφαρμόσει έλεγχο στατιστικής σημαντικότητας 95% και παρατηρήσαμε ότι δεν εμφανίζεται κάποια σημαντική διαφορά για τις διάφορες τιμές  $k_{max}$  που ελέγχουμε για κάθε εκδοχή της μεθόδου.

Στην συνέχεια, συγκρίνοντας τις διαφορετικές εκδοχές της μεθόδου μεταξύ τους με Kruskal-Wallis τεστ, παρατηρούμε ότι η καλύτερη εκδοχή είναι εκείνη με αρχικοποίηση υπερπαραμέτρων  $\alpha = 1$  και  $\beta = 1$ . Στον Πίνακα 10 παρουσιάζονται τα στατιστικά αποτελέσματα της καλύτερης εκδοχής.

### 4.2 Σύγκριση καλύτερων μεθόδων

Αρχικά, για τις μεθόδους *Τυχαίου Περιπάτου* με *μεταβλητό βήμα* και *Simulated Annealing*, οι υπολογισμοί έγιναν για τις παραμέτρους που απέδωσαν τις καλύτερες μέσες ακρίβειες. Έτσι, για την μέθοδο *Τυχαίου Περιπάτου* με *μεταβλητό βήμα* έχουμε  $\alpha = 2$ ,  $\beta = 0.7$ ,  $s = 0.1$  και για

την Simulated Annealing  $\alpha = 1, \beta = 1$ . Για κάθε μέθοδο βελτιστοποίησης (Τυχαία Αναζήτηση με χρήση Gaussian κατανομής, Nelder Mead, Τυχαίος Πείπατος με μεταβλητό βήμα, Simulated Annealing) και για κάθε αριθμό επαναλήψεων  $k_{max}$ , ο έλεγχος κανονικότητας Shapiro-Wilk έδειξε ότι τα δείγματα δεν προέρχονται από την κανονική κατανομή. Όλες οι τιμές των  $p$  είναι μικρότερες από 0.05 (επίπεδο σημαντικότητας  $\alpha = 95\%$ ) και επομένως η μηδενική υπόθεση  $H_0$  απορρίπτεται. Στον Πίνακα 4 παρουσιάζονται το πόσο διαφέρουν οι μέθοδοι μεταξύ τους για κάθε διαφορετικό  $k_{max}$ .

Table 4: Αποτελέσματα Shapiro - Wilk ελέγχου κανονικότητας

|           | Gaussian RS |         | Nelder Mead |         | Random Walk |         | Sim. Annealing |         |
|-----------|-------------|---------|-------------|---------|-------------|---------|----------------|---------|
| $k_{max}$ | Statistic   | p value | Statistic   | p value | Statistic   | p value | Statistic      | p value |
| 10        | 0.915       | 0.000   | 0.910       | 0.000   | 0.901       | 0.000   | 0.910          | 0.000   |
| 20        | 0.912       | 0.000   | 0.910       | 0.000   | 0.898       | 0.000   | 0.909          | 0.000   |
| 30        | 0.900       | 0.000   | 0.910       | 0.000   | 0.898       | 0.000   | 0.910          | 0.000   |
| 40        | 0.901       | 0.000   | 0.910       | 0.000   | 0.897       | 0.000   | 0.910          | 0.000   |
| 50        | 0.898       | 0.000   | 0.910       | 0.000   | 0.900       | 0.000   | 0.910          | 0.000   |
| 60        | 0.896       | 0.000   | 0.910       | 0.000   | 0.901       | 0.000   | 0.910          | 0.000   |
| 70        | 0.896       | 0.000   | 0.910       | 0.000   | 0.899       | 0.000   | 0.910          | 0.000   |
| 80        | 0.896       | 0.000   | 0.910       | 0.000   | 0.900       | 0.000   | 0.905          | 0.000   |
| 90        | 0.896       | 0.000   | 0.910       | 0.000   | 0.900       | 0.000   | 0.902          | 0.000   |
| 100       | 0.896       | 0.000   | 0.910       | 0.000   | 0.899       | 0.000   | 0.902          | 0.000   |

Άρα, για να ελέγξουμε εάν τα δείγματα των 4 μεθόδων έχουν διαφορετική κατανομή θα χρησιμοποιήσουμε τον μη παραμετρικό έλεγχο Kruskal-Wallis. Τα δείγματα μας εδώ θεωρούνται ανεξάρτητα, διότι οι τελικοί υπολογισμοί δεν έχουν γίνει για κάθε kfold, όπου οι αλγόριθμοι εφαρμόζονται στο ίδιο training και test set, αλλά στα συγχωνευμένα αποτελέσματα των 10 διαφορετικών kfold. Η μηδενική υπόθεση εδώ υποστηρίζει ότι όλα τα δείγματα δεδομένων προέρχονται από την ίδια κατανομή, δηλαδή οι πληθυσμιακοί διάμεσοι όλων των ομάδων (μεθόδων) είναι ίσες. Για κάθε τιμή  $k_{max}$ , ο έλεγχος έδειξε ότι τα 4 δείγματα προέρχονται από την ίδια κατανομή και επομένως η μηδενική υπόθεση δεν μπορεί να απορριφθεί. Όλες οι τιμές των  $p$  είναι μεγαλύτερες από 0.05. Στον Πίνακα 5 παρουσιάζονται τα αποτελέσματα του μη παραμετρικό έλεγχο Kruskal-Wallis.

Ως συμπέρασμα, δεν υπάρχει στατιστικά σημαντική διαφορά στην πληθυσμιακή διάμεσο της ακρίβειας (accuracy) ως προς τις 4 μεθόδους βελτιστοποίησης. Τα αποτελέσματα μπορούν να γενικευτούν στις μέσες τιμές, καθώς οι πληθυσμοί από όπου προέρχονται τα δεδομένα μας σε κάθε μέθοδο είναι συμμετρικοί ( $\text{Mean} \approx \text{Median}$ ).

Table 5: Αποτελέσματα Kruskal - Wallis H Test

| $k_{max}$ | Statistic | p value |
|-----------|-----------|---------|
| 10        | 0.343     | 0.952   |
| 20        | 0.470     | 0.925   |
| 30        | 0.530     | 0.912   |
| 40        | 0.468     | 0.926   |
| 50        | 0.516     | 0.915   |
| 60        | 0.505     | 0.918   |
| 70        | 0.474     | 0.924   |
| 80        | 0.545     | 0.909   |
| 90        | 0.568     | 0.904   |
| 100       | 0.504     | 0.918   |

### 4.3 Σύγκριση με τα δημοσιευμένα αποτελέσματα

Έχοντας βρει τις καλύτερες εκδοχές της κάθε μεθόδου βελτιστοποίησης, σε αυτό το σημείο θα γίνει η σύγκριση τους με τον αλγόριθμο που προτείνεται στην δημοσίευση<sup>2</sup>. Στον Πίνακα 6

<sup>2</sup>Georgiou, Vasileios & Pavlidis, Nicos & Parsopoulos, Konstantinos & Alevizos, D. & Vrahatis, Michael. (2004). Optimizing the Performance of Probabilistic Neural Networks in a Bionformatics Task

Table 6: Σύγκριση Στατιστικών αποτελεσμάτων για 10-KFold δειγματοληψία και  $k_{max} = 10$

| Methods        | Mean  | Median | St.Dev. | Min   | Max   | CI low | CI up |
|----------------|-------|--------|---------|-------|-------|--------|-------|
| PSO            | 0.958 | 0.958  | 0.0032  | 0.951 | 0.967 | 0.956  | 0.959 |
| Gaussian RS    | 0.965 | 0.971  | 0.024   | 0.913 | 1.0   | 0.960  | 0.969 |
| Nelder Mead    | 0.963 | 0.971  | 0.025   | 0.884 | 1.0   | 0.958  | 0.968 |
| Random Walk    | 0.965 | 0.971  | 0.025   | 0.913 | 1.0   | 0.959  | 0.969 |
| Sim. Annealing | 0.965 | 0.971  | 0.025   | 0.913 | 1.0   | 0.959  | 0.969 |

παρουσιάζονται κάποιες στατιστικές τιμές των 5 συνολικών πλέον μεθόδων. Διαπιστώνουμε ότι οι μέσες τιμές, οι διάμεσοι, οι τυπικές αποκλίσεις, τα διαστήματα εμπιστοσύνης και οι μέγιστες ακρίβειες είναι μικρότερες από αυτές που υπολογίσαμε στην δική μας εργασία. Η ελάχιστη ακρίβεια της μεθόδου PNN + PSO που προτείνεται ισούται με 0.951 και είναι μεγαλύτερη από αυτή που υπολογίσαμε στην μέθοδο Nelder Mead δηλαδή ίση με 0.884. Η μέγιστη ακρίβεια της μεθόδου PNN + PSO που προτείνεται ισούται με 0.967 και είναι μικρότερη από αυτή που υπολογίσαμε εμείς σε όλες τις μεθόδους βελτιστοποίησης δηλαδή ίση με  $\max = 1$ .

## A Στατιστικά

Table 7: Στατιστικά μεθόδου Gaussian Τυχαίας Αναζήτησης.

| $k_{max}$ | Mean  | Median | St.Dev. | Min   | Max | CI low | CI up |
|-----------|-------|--------|---------|-------|-----|--------|-------|
| 10        | 0.965 | 0.971  | 0.024   | 0.913 | 1.0 | 0.960  | 0.969 |
| 20        | 0.964 | 0.971  | 0.024   | 0.913 | 1.0 | 0.959  | 0.969 |
| 30        | 0.964 | 0.971  | 0.025   | 0.913 | 1.0 | 0.959  | 0.969 |
| 40        | 0.964 | 0.971  | 0.025   | 0.913 | 1.0 | 0.959  | 0.969 |
| 50        | 0.965 | 0.971  | 0.025   | 0.913 | 1.0 | 0.960  | 0.970 |
| 60        | 0.965 | 0.971  | 0.025   | 0.913 | 1.0 | 0.960  | 0.970 |
| 70        | 0.965 | 0.971  | 0.025   | 0.913 | 1.0 | 0.960  | 0.970 |
| 80        | 0.965 | 0.971  | 0.025   | 0.913 | 1.0 | 0.960  | 0.970 |
| 90        | 0.965 | 0.971  | 0.025   | 0.913 | 1.0 | 0.960  | 0.970 |
| 100       | 0.965 | 0.971  | 0.025   | 0.913 | 1.0 | 0.960  | 0.970 |

Table 8: Στατιστικά μεθόδου Nelder—Mead για διάφορες τιμές  $k_{max}$ .

| $k_{max}$ | Mean  | Median | St.Dev. | Min   | Max | CI low | CI up |
|-----------|-------|--------|---------|-------|-----|--------|-------|
| 10        | 0.963 | 0.971  | 0.025   | 0.884 | 1.0 | 0.958  | 0.968 |
| 20        | 0.963 | 0.971  | 0.025   | 0.884 | 1.0 | 0.958  | 0.968 |
| 30        | 0.963 | 0.971  | 0.025   | 0.884 | 1.0 | 0.958  | 0.968 |
| 40        | 0.963 | 0.971  | 0.025   | 0.884 | 1.0 | 0.958  | 0.968 |
| 50        | 0.963 | 0.971  | 0.025   | 0.884 | 1.0 | 0.958  | 0.968 |
| 60        | 0.963 | 0.971  | 0.025   | 0.884 | 1.0 | 0.958  | 0.968 |
| 70        | 0.963 | 0.971  | 0.025   | 0.884 | 1.0 | 0.958  | 0.968 |
| 80        | 0.963 | 0.971  | 0.025   | 0.884 | 1.0 | 0.958  | 0.968 |
| 90        | 0.963 | 0.971  | 0.025   | 0.884 | 1.0 | 0.958  | 0.968 |
| 100       | 0.963 | 0.971  | 0.025   | 0.884 | 1.0 | 0.958  | 0.968 |

Table 9: Στατιστικά μεθόδου Τυχαίος Περίπατος με μεταβλητό βήμα για τις καλύτερες παραμέτρους  $\alpha = 2$ ,  $\beta = 0.7$  και βήμα  $s = 0.1$ .

| $k_{max}$ | Mean  | Median | St.Dev. | Min   | Max | CI low | CI up |
|-----------|-------|--------|---------|-------|-----|--------|-------|
| 10        | 0.965 | 0.971  | 0.025   | 0.913 | 1.0 | 0.959  | 0.969 |
| 20        | 0.965 | 0.971  | 0.025   | 0.913 | 1.0 | 0.960  | 0.970 |
| 30        | 0.965 | 0.971  | 0.025   | 0.913 | 1.0 | 0.960  | 0.970 |
| 40        | 0.965 | 0.971  | 0.025   | 0.913 | 1.0 | 0.959  | 0.970 |
| 50        | 0.965 | 0.971  | 0.025   | 0.913 | 1.0 | 0.960  | 0.970 |
| 60        | 0.965 | 0.971  | 0.025   | 0.913 | 1.0 | 0.960  | 0.970 |
| 70        | 0.965 | 0.971  | 0.025   | 0.913 | 1.0 | 0.959  | 0.969 |
| 80        | 0.965 | 0.971  | 0.025   | 0.913 | 1.0 | 0.959  | 0.970 |
| 90        | 0.965 | 0.971  | 0.025   | 0.913 | 1.0 | 0.960  | 0.970 |
| 100       | 0.965 | 0.971  | 0.025   | 0.913 | 1.0 | 0.959  | 0.969 |

Table 10: Στατιστικά μεθόδου Simulated Annealing για τις καλύτερες παραμέτρους  $\alpha = 1$ ,  $\beta = 1$ .

| $k_{max}$  | Mean  | Median | St.Dev. | Min   | Max | CI low | CI up |
|------------|-------|--------|---------|-------|-----|--------|-------|
| <i>10</i>  | 0.965 | 0.971  | 0.025   | 0.913 | 1.0 | 0.959  | 0.969 |
| <i>20</i>  | 0.965 | 0.971  | 0.025   | 0.913 | 1.0 | 0.959  | 0.969 |
| <i>30</i>  | 0.965 | 0.971  | 0.024   | 0.913 | 1.0 | 0.959  | 0.969 |
| <i>40</i>  | 0.965 | 0.971  | 0.024   | 0.913 | 1.0 | 0.959  | 0.969 |
| <i>50</i>  | 0.965 | 0.971  | 0.024   | 0.913 | 1.0 | 0.959  | 0.969 |
| <i>60</i>  | 0.965 | 0.971  | 0.024   | 0.913 | 1.0 | 0.959  | 0.969 |
| <i>70</i>  | 0.965 | 0.971  | 0.024   | 0.913 | 1.0 | 0.959  | 0.969 |
| <i>80</i>  | 0.964 | 0.971  | 0.025   | 0.913 | 1.0 | 0.959  | 0.970 |
| <i>90</i>  | 0.964 | 0.971  | 0.025   | 0.913 | 1.0 | 0.959  | 0.969 |
| <i>100</i> | 0.964 | 0.971  | 0.025   | 0.913 | 1.0 | 0.959  | 0.969 |

## B Box plots

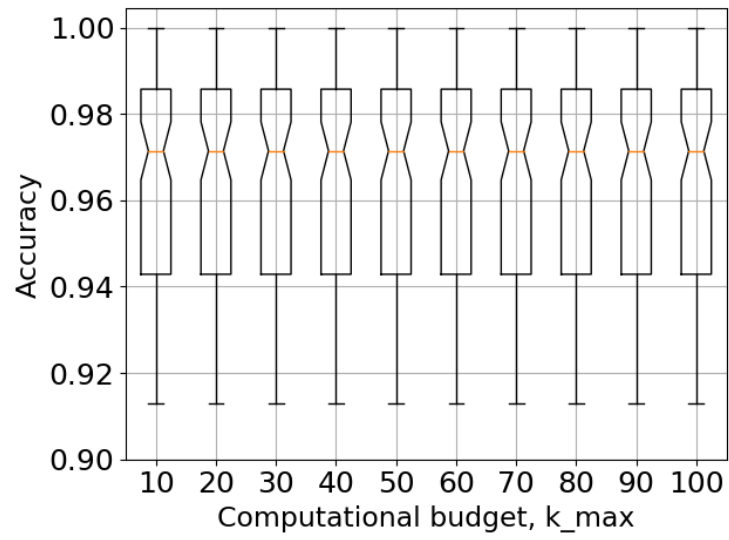


Figure 1: Box Plots μεθόδου Gaussian Τυχαίας Αναζήτησης.

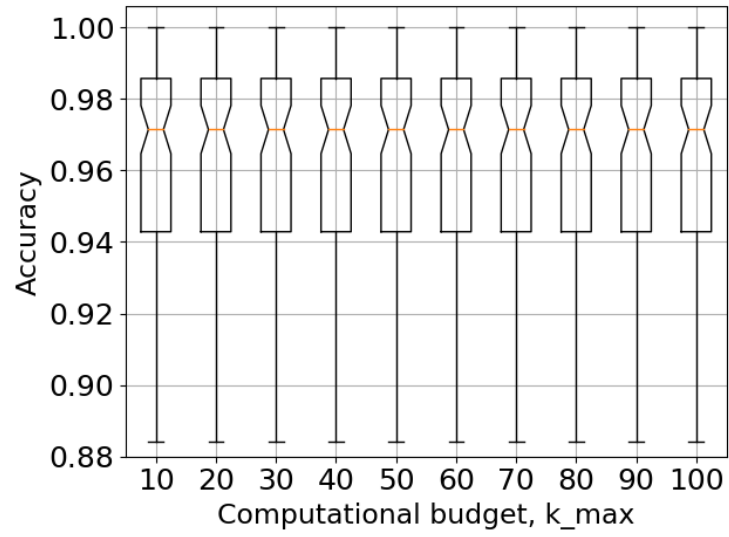
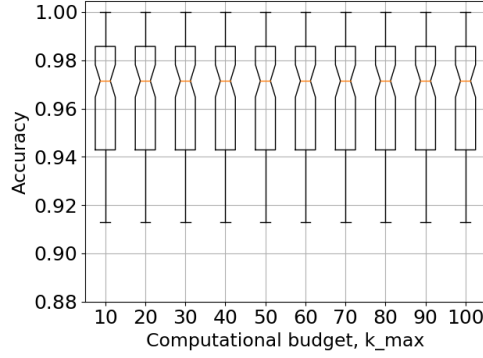
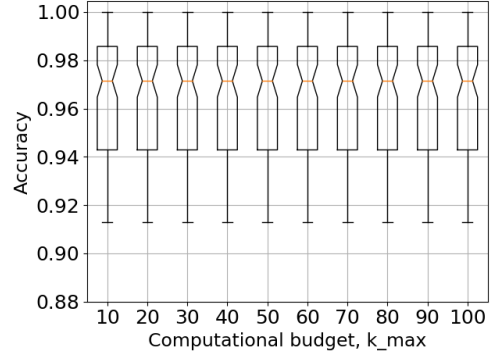


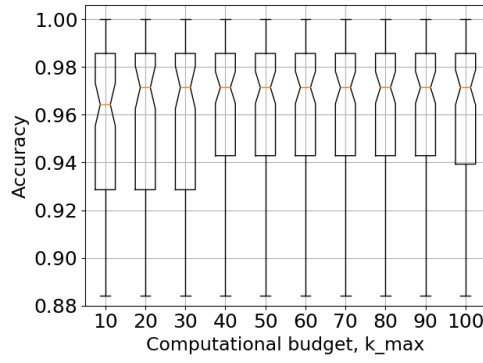
Figure 2: Box Plots μεθόδου Nelder–Mead.



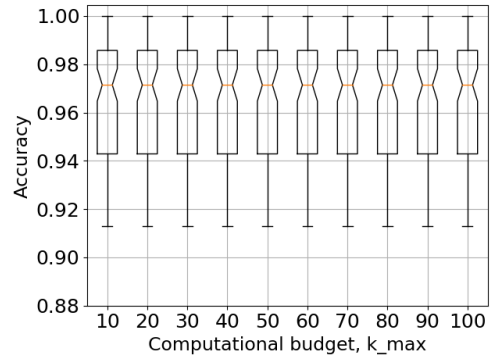
(a)  $\alpha=1.5, \beta=0.5, s=0.1$



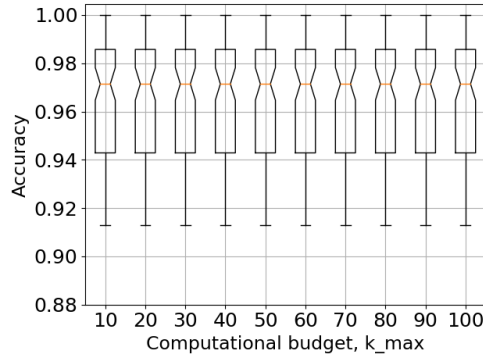
(b)  $\alpha=1.5, \beta=0.5, s=0.5$



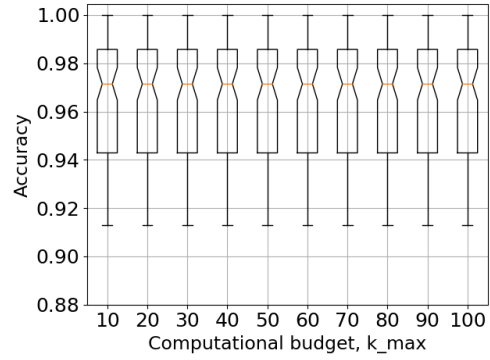
(c)  $\alpha=1.5, \beta=0.7, s=0.1$



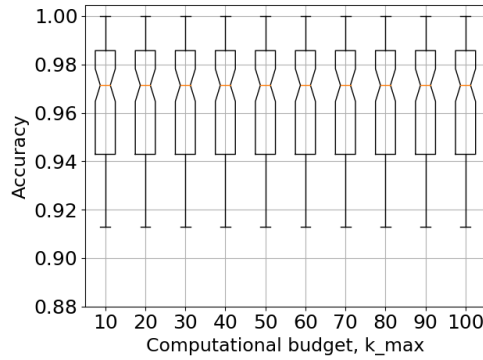
(d)  $\alpha=1.5, \beta=0.7, s=0.5$



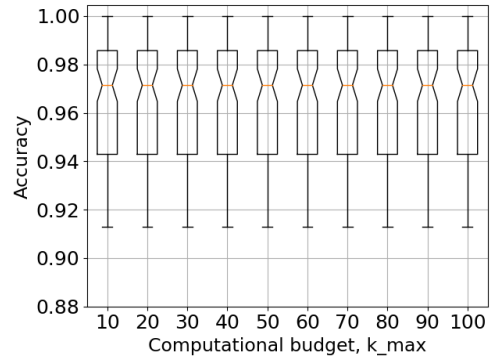
(e)  $\alpha=2, \beta=0.5, s=0.1$



(f)  $\alpha=2, \beta=0.5, s=0.5$



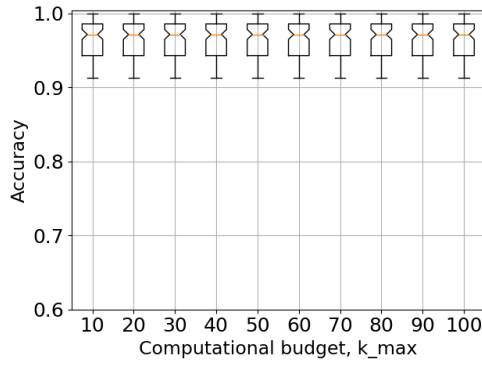
(g)  $\alpha=2, \beta=0.7, s=0.1$



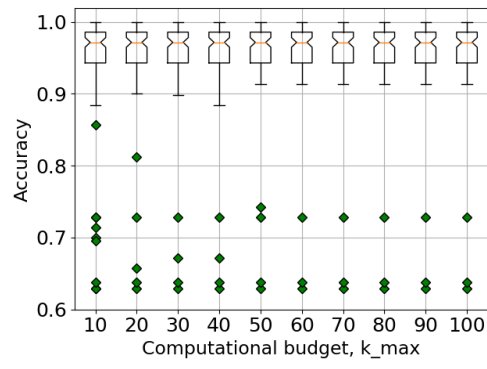
(h)  $\alpha=2, \beta=0.7, s=0.5$

Figure 3: Box plots μεθόδου Τυχαίος Περίπατος με μεταβλητό βήμα για τις διάφορες εκδοχές όπου  $\alpha$  είναι η μεταβλητή επέκτασης,  $\beta$  η μεταβλητή συρρίκνωσης και  $s$  το βήμα.

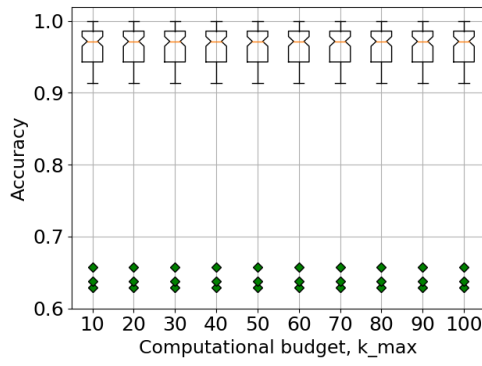




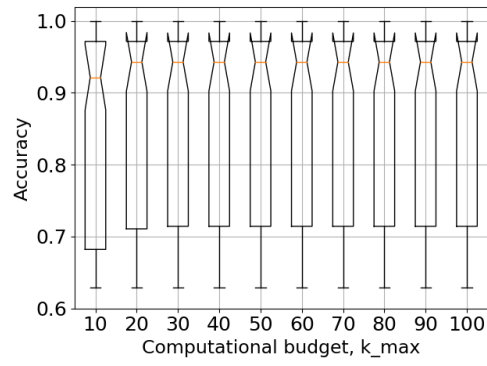
(a)  $\alpha=1.0, \beta=1.0$



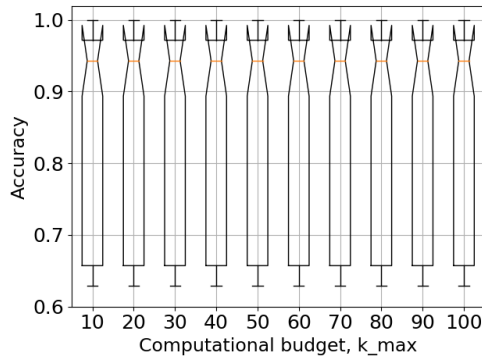
(b)  $\alpha=2.0, \beta=1.0$



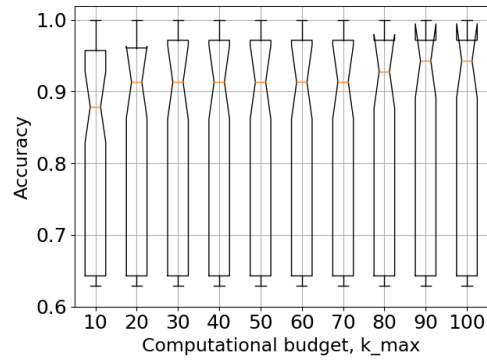
(c)  $\alpha=1.0, \beta=5.0$



(d)  $\alpha=2.0, \beta=5.0$



(e)  $\alpha=1.0, \beta=10.0$



(f)  $\alpha=2.0, \beta=10.0$

Figure 4: Box plots μεθόδου Simulated Annealing για τις διάφορες εκδοχές όπου  $\alpha$  και  $\beta$  οι μεταβλητές για την ανανέωση της τιμής της θερμοκρασίας.

## C Εκτελέσιμα

Στο τρέχων παράρτημα περιγράφεται η λειτουργία του κώδικα που αναπτύχθηκε για την τρέχουσα εργασία και ο τρόπος εκτέλεσης του εκτελέσιμου αρχείου. Ακολουθεί η περιγραφή των 7 επισυναπτόμενων αρχείων:

### 1. **main.py**

Χωρίζει τα δεδομένα σε σύνολο εκπαίδευσης και αξιολόγησης ( $10 - CV$ ) και εκτελεί 10 πειράματα για κάθε ένα  $k$ -fold για μια συγκεκριμένη μέθοδο βελτιστοποίησης που δέχεται σαν είσοδο από τον χρήστη.

### 2. **loadData.py**

Διαβάζει τα δεδομένα από το αρχείο και επιστρέφει τα κανονικοποιημένα δεδομένα.

### 3. **PNN.py**

Γίνεται η υλοποίηση του νευρωνικού δικτύου PNN.

### 4. **random\_search.py**

Αρχικά, αρχικοποιεί τις τιμές  $\sigma$  που παράγονται από την Gaussian κατανομή και τις αποθηκεύει σε αντίστοιχα αρχεία μέσα στον φάκελο *seeds/random\_search*. Έπειτα, ξεκινάει η εκτέλεση της μεθόδου για τις διάφορες τιμές  $k_{max}$  που ελέγχουμε. Για κάθε  $k_{max}$  διατηρούμε το καλύτερο  $\sigma$  που προέκυψε από την εκπαίδευση του PNN και γίνεται η αξιολόγηση για αυτό το  $\sigma$  στο σύνολο αξιολόγησης. Τέλος, αποθηκεύονται σε αρχεία μέσα στο φάκελο *results/random\_search* τόσο για το σύνολο εκπαίδευσης (πάνω μέρος) όσο και για το σύνολο αξιολόγησης (κάτω μέρος) τα καλύτερα  $\sigma$  και αντίστοιχα τα accuracy.

### 5. **random\_search\_step.py**

Αρχικά, αρχικοποιεί τις τυχαίες διευθύνσεις που χρειάζονται για την μέθοδο και τις αποθηκεύει σε αρχεία μέσα στους φακέλους *seeds/random\_search\_step*. Έπειτα, ξεκινάει η εκτέλεση της μεθόδου για τις διάφορες τιμές  $k_{max}$  που ελέγχουμε και για όλες τις διαφορετικές εκδοχές του αλγορίθμου. Για κάθε  $k_{max}$  και εκδοχή αλγορίθμου διατηρούμε το καλύτερο  $\sigma$  που προέκυψε από την εκπαίδευση του PNN και γίνεται η αξιολόγηση για αυτό το  $\sigma$  στο σύνολο αξιολόγησης. Τέλος, αποθηκεύονται σε αρχεία μέσα στο φάκελο *results/random\_search\_step* τόσο για το σύνολο εκπαίδευσης (πάνω μέρος) όσο και για το σύνολο αξιολόγησης (κάτω μέρος) τα καλύτερα  $\sigma$  και αντίστοιχα τα accuracy.

### 6. **nelder\_mead.py**

Αρχικά, αρχικοποιεί τις τρεις τυχαίες αρχικές θέσεις  $\sigma$  που χρειάζονται για την μέθοδο και τις αποθηκεύει σε αρχεία μέσα στους φακέλους *seeds/nelder\_mead*. Έπειτα, ξεκινάει η εκτέλεση της μεθόδου για τις διάφορες τιμές  $k_{max}$  που ελέγχουμε και για όλες τις διαφορετικές εκδοχές του αλγορίθμου. Για κάθε  $k_{max}$  και εκδοχή αλγορίθμου διατηρούμε το καλύτερο  $\sigma$  που προέκυψε από την εκπαίδευση του PNN και γίνεται η αξιολόγηση για αυτό το  $\sigma$  στο σύνολο αξιολόγησης. Τέλος, αποθηκεύονται σε αρχεία μέσα στο φάκελο *results/nelder\_mead* τόσο για το σύνολο εκπαίδευσης (πάνω μέρος) όσο και για το σύνολο αξιολόγησης (κάτω μέρος) τα καλύτερα  $\sigma$  και αντίστοιχα τα accuracy.

### 7. **simulated\_annealing.py**

Αρχικά, αρχικοποιεί την αρχική τιμή  $\sigma$ , την θερμοκρασία και τις τιμές της διεύθυνσης και τις αποθηκεύει σε αντίστοιχα αρχεία μέσα στον φάκελο *seeds/simulated\_annealing*. Έπειτα, ξεκινάει η εκτέλεση της μεθόδου για τις διάφορες τιμές  $k_{max}$  που ελέγχουμε. Για κάθε  $k_{max}$  διατηρούμε το καλύτερο  $\sigma$  που προέκυψε από την εκπαίδευση του PNN και γίνεται η αξιολόγηση για αυτό το  $\sigma$  στο σύνολο αξιολόγησης. Τέλος, αποθηκεύονται σε αρχεία μέσα στο φάκελο *results/simulated\_annealing* τόσο για το σύνολο εκπαίδευσης (πάνω μέρος) όσο και για το σύνολο αξιολόγησης (κάτω μέρος) τα καλύτερα  $\sigma$  και αντίστοιχα τα accuracy.

## Οδηγίες ορθής εκτέλεσης κώδικα

Η υλοποίηση των μεθόδων έγινε σε γλώσσα **Python 3**. Οι βιβλιοθήκες που έχουν χρησιμοποιηθεί είναι οι Numpy και scikit-learn. Επίσης, τα modules που χρησιμοποιήθηκαν είναι τα `acoolson`, `math`, `sys` και `os`. Η γενική μορφή της εντολής εκτέλεσης του προγράμματος είναι η εξής:

**python3 main.py method**

όπου `method`  $\in \{\text{random\_search}, \text{random\_search\_step}, \text{nelder\_mead}, \text{simulated\_annealing}\}$ . Παρακάτω παρατίθεται ένα ενδεικτικό παράδειγμα εκτέλεσης για την μέθοδο:

**python3 main.py random\_search**

Τέλος, σε περίπτωση που θέλετε να δοκιμάσετε να τρέξετε το πρόγραμμα για μικρότερο πλήθος συναρτησιακών υπολογισμών  $k_{max}$  (επαναλήψεων), θα πρέπει να προσδιορίσετε μέσα στο κάθε αρχείο μεθόδων την μεταβλητή **kmax** με το οποίο δηλώνεται το συνολικό πλήθος συναρτησιακών υπολογισμών και την μεταβλητή **k\_max** με την οποία δηλώνεται ανά πόσες επαναλήψεις θα διαιρέσετε το συνολικό αυτό πλήθος συναρτησιακών υπολογισμών έτσι ώστε να δείτε την εξέλιξη του accuracy για διάφορες τιμές συνολικών συναρτησιακών υπολογισμών.