Εθνικο Μετσοβιο Πολυτεχνειο
Σχολη Ηλεκτρολογων Μηχανικων και Μηχανικων Υπολογιστων
Τομεας Τεχνολογιας Πληροφορικης και Υπολογιστων

Μελέτη και αξιολόγηση τεχνικών για Μηχανική Χαρακτηριστικών (Feature Engineering)

Διπλωματικη Εργασια

του

ΚΩΝΣΤΑΝΤΙΝΟΥ ΑΛΑΤΖΑ

Επιβλέπων: Βασιλική Καντερέ
Επίκουρη Καθηγήτρια Ε.Μ.Π.

Εργαστηριο Συστηματων Βασεων Γνωσεων και Δεδομενων
Αθήνα, Ιούνιος 2021

Εθνικό Μετσόβιο Πολυτεχνείο
Σχολή Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών
Τομέας Τεχνολογίας Πληροφορικής και Υπολογιστών
Εργαστήριο Συστημάτων Βάσεων Γνώσεων και Δεδομένων

Μελέτη και αξιολόγηση τεχνικών για Μηχανική Χαρακτηριστικών (**Feature Engineering**)

Διπλωματικη Εργασια

του

ΚΩΝΣΤΑΝΤΙΝΟΥ ΑΛΑΤΖΑ

Επιβλέπων: Βασιλική Καντερέ
Επίκουρη Καθηγήτρια Ε.Μ.Π.

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την 22η Ιουνίου 2021.

(Υπογραφή)	*(Υπογραφή)*	*(Υπογραφή)*
.....	.....	.....
Βασιλική Καντερέ	Συμεών Παπαβασιλείου	Γεώργιος Στάμου
Επ. Καθηγήτρια Ε.Μ.Π.	Καθηγητής Ε.Μ.Π.	Καθηγητής Ε.Μ.Π.

Αθήνα, Ιούνιος 2021

(Υπογραφή)

.....

Κωνσταντινος Αλατζας

Διπλωματούχος Ηλεκτρολόγος Μηχανικός και Μηχανικός Υπολογιστών Ε.Μ.Π.

Εθνικό Μετσόβιο Πολυτεχνείο
Σχολή Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών
Τομέας Τεχνολογίας Πληροφορικής και Υπολογιστών
Εργαστήριο Συστημάτων Βάσεων Γνώσεων και Δεδομένων

Ευχαριστίες

Θα ήθελα να ευχαριστήσω την επιβλέπουσα καθηγήτρια κ. Βασιλική Καντερέ για την ανάθεση του θέματος της διπλωματικής εργασίας και την εμπιστοσύνη της.

Επίσης, θα ήθελα να ευχαριστήσω την οικογένειά μου, τους γονείς μου Παναγιώτη και Καλλιόπη και τον αδερφό μου Αλέξανδρο, για την απεριόριστη υποστήριξή τους.

Περίληψη

Αντικείμενο της διπλωματικής είναι η μελέτη και η αξιολόγηση αλγορίθμων για feature engineering σε δημοσίως διαθέσιμα σύνολα δεδομένων. Ειδικότερα, θα μελετηθεί και θα εκτελεστεί ο αλγόριθμος ReliefF, ένας από τους πιο σημαντικούς αλγορίθμους που έχει επιτυχώς εφαρμοστεί σε πολλές εφαρμογές επιλογής χαρακτηριστικών, και οι αλγόριθμοι TuRF και SURF που αποτελούν επεκτάσεις του αλγορίθμου ReliefF. Ο αλγόριθμος ReliefF είναι μια ευρέως εφαρμοσμένη μέθοδος στάθμισης χαρακτηριστικών που αξιολογεί την ποιότητα των χαρακτηριστικών ενός δοθέντος συνόλου δεδομένων, αντιστοιχίζοντας βάρη σε καθένα από αυτά. Μπορεί επίσης να χρησιμοποιηθεί σαν μια μέθοδος φιλτραρίσματος επιλογής χαρακτηριστικών, ορίζοντας επίπεδο σημαντικότητας και επιλέγοντας χαρακτηριστικά πάνω από αυτό. Ο αλγόριθμος TuRF προχωραέι τη διαδικασία επιλογής χαρακτηριστικών του ReliefF από διαδικασία ενός γύρου σε διαδικασία πολλών γύρων και ο αλγόριθμος SURF είναι επέκταση του ReliefF που καθορίζει αυτόματα τον ιδανικό αριθμό από γείτονες προς εξέταση κατά τη βαθμολόγηση των χαρακτηριστικών. Στο πρώτο μέρος της διπλωματικής γίνεται βιβλιογραφική αναφορά σε ερευνητικές εργασίες που αφορούν αλγορίθμους μηχανικής χαρακτηριστικών. Στο τελευταίο μέρος της διπλωματικής θα διεξαχθεί πειραματική αξιολόγηση των αλγορίθμων ReliefF, TuRF και SURF σε δημοσίως διαθέσιμα σύνολα δεδομένων. Πιο συγκεκριμένα, η αξιολόγηση της αποδοτικότητας των αλγορίθμων θα γίνει ως προς την πολυπλοκότητα, την ακρίβεια καθώς και άλλες μετρικές σε δημοσίως διαθέσιμα σύνολα δεδομένων.

Λέξεις Κλειδιά

Μηχανική χαρακτηριστικών, Επιλογή χαρακτηριστικών, Relief, ReliefF, TuRF, SURF, ReBATE.

Abstract

The subject of this diploma thesis is the study and evaluation of algorithms for feature engineering on publicly available data sets. Especially, the algorithm ReliefF, one of the most important algorithms that has been successfully applied in many feature selection applications, will be studied and executed, as well as the algorithms TuRF and SURF that are extentions of the RelieF algorithm. The ReliefF algorithm is a widely applied attribute weighting method that evaluates the quality of the attributes of a given data set, by attaching weights to each of them. It can be also used as a feature selection filtering method, by defining a relevance level and selecting features above that. The TuRF algorithm advances the feature selection process of ReliefF from a single-round process to a multi-round process and the SURF algorithm is an extention of ReliefF that automatically defines the ideal number of neighbors to consider during the attribute evaluation. In the first part of the thesis there is bibliographic reference to research works concerning feature engineering algorithms. In the final part of the thesis experimental evaluation of the algorithms ReliefF, TuRF and SURF will be conducted on publicly available data sets. More specifically, the evaluation of of the efficiency of the algorithms will be done according to complexity, precision and other metrics on publicly available data sets.

Keywords

Feature engineering, Feature selection, Relief Based Algorithms, ReliefF, TuRF, SURF, ReBATE

Περιεχόμενα

Κατάλογος Σχημάτων

Κεφάλαιο 1

Εισαγωγή

1.1 Επιλογή χαρακτηριστικών

Η επιλογή χαρακτηριστικών είναι συχνά ένα απαραίτητο έργο στην εξόρυξη δεδομένων και τη μοντελοποίηση (γενίκευση), ιδιαίτερα σε προβλήματα όπου τα δεδομένα είναι θορυβώδη, σύνθετα, και/ή περιέχουν ένα πολύ μεγάλο χώρο χαρακτηριστικών. Πολλές στρατηγικές επιλογής χαρακτηριστικών έχουν προταθεί με τα χρόνια, εμπίπτοντας γενικά σε μία από τρεις κατηγορίες: (1) μέθοδοι φιλτραρίσματος, (2) μέθοδοι περιτυλίγματος, ή (3) ενσωματωμένες μέθοδοι. Οι μέθοδοι επιλογής χαρακτηριστικών έχουν ακόμα χαρακτηριστεί με βάση το αν η επιλογή βασίζεται σε βάρη που ανατίθενται σε ξεχωριστά χαρακτηριστικά ή σε ένα υποψήφιο υποσύνολο χαρακτηριστικών.

Αποτελεί θεμελιώδη πρόκληση για σχεδόν κάθε έργο εξόρυξης δεδομένων ή μοντελοποίησης η αναγνώριση και ο χαρακτηρισμός σχέσεων μεταξύ ενός ή περισσότερων χαρακτηριστικών στα δεδομένα και κάποιας κλάσης. Στα περισσότερα σύνολα δεδομένων, μόνο ένα υποσύνολο από τα διαθέσιμα χαρακτηριστικά είναι σχετικά χαρακτηριστικά, δηλ. παρέχουν πληροφορία για τον προσδιορισμό της τιμής της κλάσης. Τα υπόλοιπα άσχετα χαρακτηριστικά, τα οποία σπάνια μπορούν να διακριθούν a priori σε αληθινά προβλήματα, δεν παρέχουν πληροφορία αλλά συνεισφέρουν στη συνολική διαστατικότητα του χώρου του προβλήματος. Αυτό αυξάνει τη δυσκολία και το υπολογιστικό φορτίο που τίθεται στις μεθόδους μοντελοποίησης. Η επιλογή χαρακτηριστικών θα μπορούσε γενικά να οριστεί ως η διαδικασία αναγνώρισης σχετικών χαρακτηριστικών και απόρριψης άσχετων χαρακτηριστικών.

1.1.1 Μέθοδοι φίλτρου

Οι μέθοδοι φίλτρου χρησιμοποιούν ένα μέτρο που υπολογίζεται από τα γενικά χαρακτηριστικά των δεδομένων εκπαίδευσης για τη βαθμολόγηση των χαρακτηριστικών ή υποσυνόλων χαρακτηριστικών ως ένα βήμα επεξεργασίας πριν τη μοντελοποίηση. Τα φίλτρα είναι γενικά πολύ πιο γρήγορα και λειτουργούν ανεξάρτητα από τον αλγόριθμο γενίκευσης, που σημαίνει ότι τα επιλεγμένα χαρακτηριστικά μπορούν μετά να περαστούν σε οποιδήποτε αλγόριθμο μοντελοποίησης. Οι μέθοδοι φίλτρου μπορούν ακόμα να κατηγοριοποιηθούν χονδρικά με βάση τα μέτρα φιλτραρίσματος που χρησιμοποιούν, δηλ. πληροφορία, απόσταση, εξάρτηση, συνέπεια,

ομοιότητα, και στατιστικά μέτρα. Ο αλγόριθμοι επιλογής χαρακτηριστικών που θα αναλυθούν σε αυτή την εργασία αποτελούν μεθόδους φίλτρου.

1.1.2 Μέθοδοι ξεχωριστής αξιολόγησης

Κατά τη ξεχωριστή αξιολόγηση, η αντιστοίχηση βαρών ή κατάταξη των χαρακτηριστικών εκτιμά ξεχωριστά χαρακτηριστικά και τους αναθέτει βάρη/βαθμούς ανάλογα με τους βαθμούς σχετικότητάς τους. Μια μέθοδος φίλτρου μπορεί να είναι ή να να μην είναι μέθοδος ξεχωριστής αξιολόγησης. Οι αλγόριθμοι επιλογής χαρακτηριστικών που αφορύν αυτή την εργασία, εκτός από μεθόδους φίλτρου, είναι και μέθοδοι ξεχωριστής αξιολόγησης.

1.2 Relief-βασισμένοι αλγόριθμοι

Η παρούσα εργασία εστιάζει σε μέλη της οικογένειας των Relief-βασισμένων μεθόδων επιλογής χαρακτηριστικών που αναφέρονται ως RBAs (Relief-Based Algorithms) και μπορούν να χαρακτηριστούν ως μέθοδοι φιλτραρίσματος ξεχωριστής αξιολόγησης. Οι RBAs διατηρούν τα γενικά οφέλη των μεθόδων φιλτραρίσματος, δηλαδή είναι σχετικά γρήγοροι (με ασυμπτωτική πολυπλοκότητα της τάξης του $O(instances^2 \cdot features)$), και τα επιλεγμένα χαρακτηριστικά είναι ανεξάρτητα από τον αλγόριθμο γενίκευσης [9]. Κυρίως, οι RBAs είναι οι μόνες γνωστές μέθοδοι φιλτραρίσματος που έχουν τη δυνατότητα να συλλάβουν εξαρτήσεις χαρακτηριστικών στην πρόβλεψη αποτελέσματος, δηλαδή αλληλεπιδράσεις χαρακτηριστικών. Αυτή η μοναδική δυνατότητα έχει αποδοθεί στη χρήση κοντινότερων γειτόνων δειγμάτων του Relief στον υπολογισμό των βαρών των χαρακτηριστικών. Οι RBAs έχουν ακόμα το πλεονέκτημα ότι υπολογίζουν ξεχωριστά βάρη χαρακτηριστικών. Ένα άλλο αξιοσημείωτο χαρακτηριστικό των RBAs είναι ότι δεν απαλείφουν αλληλοσυσχετίσεις χαρακτηριστικών. Αυτό θα μπορούσε να θεωρηθεί είτε ως πλεονέκτημα είτε ως μειονέκτημα με βάση το πρόβλημα.

1.3 Βιοπληροφορική

Οι περισσότεροι RBAs έχουν αναπτυχθεί και εφαρμοστεί στα πλαίσια προβλημάτων γενετικής συσχέτισης. Τέτοια προβλήματα χαρακτηρίζονται συχνά ως (1) θορυβώδη, (2) ότι έχουν μικτούς τύπους δεδομένων (π.χ. διακριτά και συνεχή χαρακτηριστικά) και (3) ότι περίεχουν πολύ μεγάλους χώρους χαρακτηριστικών [10]. Ακόμα, η ανίχνευση σύνθετων μοτίβων της συσχέτισης μεταξύ χαρακτηριστικών και της κλάσης είναι ιδιαίτερου ενδιαφέροντος στη βιοπληροφορική. Συγκεκριμένα, αυτή η εργασία θεωρεί δύο σημαντικά φαινόμενα: την επίσταση, δηλαδή αλληλεπιδράσεις χαρακτηριστικών (ή αλληλεπιδράσεις γονιδίων στα πλαίσια της βιοπληροφορικής) και ετερογενείς συσχετίσεις με την κλάση, δηλαδή γενετική ετερογένεια ή φαινότυπος (στα πλαίσια προβλημάτων βιοπληροφορικής) [10]. Γενετική ετερογένεια προκύπτει όταν η ίδια ή παρόμοια φαινοτυποική κλάση μπορεί να είναι το αποτέλεσμα διαφορετικών ανεξάρτητων σχετικών χαρακτηριστικών (ή συνόλου σχετικών χαρακτηριστικών) σε διαφορετικά υποσύνολα του ίδιου πληθυσμού.

1.4 Αντικείμενο της διπλωματικής

1.4.1 Συνεισφορά

Η συνεισφορά της διπλωματικής συνοψίζεται ως εξής:

1. Αναφέρθηκαν βιβλιογραφικά ερευνητικές εργασίες σχετικά με αλγορίθμους επιλογής χαρακτηριστικών
2. Μελετήθηκε ο αλγόριθμος επιλογής χαρακτηριστικών ReliefF και οι αλγόριθμοι SURF και TuRF που βασίζονται σε αυτόν.
3. Αξιολογήθηκαν οι αλγόριθμοι και οι επεκτάσεις που είναι υλοποιημένες στη βιβλιοθήκη ReBATE της Python.
4. Αξιολογήθηκε η αποτελεσματικότητα των αλγορίθμων σε γενετικά σύνολα δεδομένων που αφορούν προβλήματα της Βιοπληροφορικής.

1.5 Οργάνωση του τόμου

Εργασίες σχετικές με το αντικείμενο της διπλωματικής παρουσιάζονται στο Κεφάλαιο 2. Το Κεφάλαιο 3 θέτει το θεωρητικό υπόβαθρο. Στο Κεφάλαιο 4 αναπτύσσεται η πειραματική αξιολόγηση των αλγορίθμων επιλογής χαρακτηριστικών και συζητούνται τα αποτελέσματα της μελέτης. Τέλος, στο Κεφάλαιο 5 συνοψίζονται τα συμπεράσματα της διπλωματικής και προτείνονται μελλοντικές επεκτάσεις.

Κεφάλαιο 2

Συγγενικές εργασίες

2.1 Αλγόριθμος Relief

Οι Kira και Rendell [1], [4] διατύπωσαν τον αρχικό αλγόριθμο Relief εμπνευσμένοι από μάθηση με βάση τα δείγματα. Ως μια μέθοδος ξεχωριστής αξιολόγισης φιλτραρίσματος χαρακτηριστικών, ο Relief υπολογίζει μια στατιστική για κάθε χαρακτηριστικό που μπορεί να χρησιμοποιηθεί για την εκτίμηση της ποιότητας ή της σχετικότητας ως προς την έννοια-στόχο (δηλ. την πρόβλεψη της τιμής της κλάσης). Αυτές οι στατιστικές χαρακτηριστικών αναφέρονται ως βάρη χαρακτηριστικών, ή ως βαθμοί χαρακτηριστικών που μπορούν να κυμαίνονται από -1 (χειρότερο) μέχρι +1 (καλύτερο). Αξίζει να σημειωθεί ότι ο αρχικός αλγόριθμος Relief περιοριζόταν σε προβλήματα δυαδικής ταξινόμησης, και δεν είχε μηχανισμό για να χειριστεί απουσιάζουσες τιμές.

2.1.1 Πλεονεκτήματα και περιορισμοί

Σχετικά με τα πλεονεκτήματα, ο Relief έχει παρουσιαστεί ως μη μυοπικός [12], δηλ. εκτιμάει την ποιότητα ενός δεδομένου χαρακτηριστικού σε σχέση με άλλα χαρακτηριστικά, και μη παραμετρικός [15], δηλ. ότι δεν κάνει υποθέσεις σχετικά με την κατανομή του πληθυσμού ή το μέγεθος του δείγματος. Η αποδοτικότητα του αλγορίθμου έχει αποδοθεί στο γεγονός ότι δεν εξερευνά υποσύνολα χαρακτηριστικών και δεν ασχολείται να προσπαθήσει να αναγνωρίσει ένα βέλτιστο ελάχιστο μέγεθος υποσυνόλου χαρακτηριστικών [4]. Αντίθετα, ο Relief αρχικά προοριζόταν να αναγνωρίζει ένα υποσύνολο χαρακτηριστικών που μπορεί να μην είναι το ελάχιστο και μπορεί ακόμα να περιέχει κάποια άσχετα και περιττά χαρακτηριστικά, αλλά είναι αρκετά μικρό ώστε να χρησιμοποιηθεί με άλλες προσεγγίσεις σε μια λεπτομερή ανάλυση [15]. Μια εξαντλητική αναζήτηση για αλληλεπιδράσεις μεταξύ όλων των ζευγών χαρακτηριστικών από μόνη της θα είχε μια χρονική πολυπλοκότητα της τάξης του $O(2^a)$, ενώ ο Relief έχει μια χρονική πολυπλοκότητα της τάξης του $O(a \cdot m \cdot n)$, ή $O(a \cdot n)$ αν $m < n$. Ακόμα, έχει προταθεί ότι ο Relief θα μπορούσε να θεωρηθεί ως ένας αλγόριθμος που μπορεί να σταματηθεί και να επιστρέψει αποτελέσματα οποιαδήποτε στιγμή, αλλά υποτίθεται ότι με περισσότερο χρόνο ή δεδομένα θα βελτιώσει τα αποτελέσματα [5].

Σχετικά με τους περιορισμούς, ή αρχική ανάλυση του Relief δηλώνει ότι ο αλγόριθμος

μπορεί να ξεγελαστεί από ανεπαρκείς κύκλους εκπαίδευσης (δηλ. ένα όχι αρκετά μεγάλο m). Η αρχική δημοσίευση δηλώνει επίσης ότι ο Relief είναι αρκετά ανεκτικός σε θόρυβο και ανεπηρέαστος από αλληλεπιδράσεις χαρακτηριστικών. Ωστόσο, πιο πρόσφατη δουλειά αναγνώρισε ότι ο Relief ήταν ευαίσθητος στην παρέμβαση του θορύβου με τη επιλογή των κοντινότερων γειτόνων [3]. Σημαντικότερα, έχει επανειλημμένα επιδειχθεί εμπειρικά και θεωρητικά ότι η επίδοση του Relief χειροτερεύει καθώς ο αριθμός των άσχετων χαρακτηριστικών γίνεται μεγάλος [5][15]. Αυτή η χειροτέρευση της επίδοσης στην αναγνώριση αλληλεπιδρόντων χαρακτηριστικών οφείλεται κυρίως στο γεγονός ότι ο υπολογισμός των γειτόνων και των βαρών του Relief γίνεται αυξανόμενα τυχαίος καθώς ο αριθμός των χαρακτηριστικών αυξάνεται. Αυτό είναι ένα παράδειγμα της κατάρας της διαστατικότητας. Ακόμα, δεν υπάρχει καθιερωμένος τρόπος να εκτιμηθεί πόσα από τα επιλεγμένα χαρακτηριστικά με υψηλούς βαθμούς είναι λάθος ανακαλύψεις. Είναι πιθανό αυτό το θέμα να αντιμετωπίζεται μέσω ελέγχου μεταθέσεων όπως προτείναι από τους McKinney κ.ά. [16].

2.2 Αλγόριθμος ReliefF

Ο αρχικός αλγόριθμος Relief σπάνια εφαρμόζεται στην πράξη πια και έχει αντικατασταθεί από τον ReliefF [3] ως ο καλύτερος γνωστός και πιο χρησιμοποιημένος RBA (Relief Based Algorithm) ως τώρα. Σημειώνεται ότι το F στον ReliefF αναφέρεται στην έκτη παραλλαγή του αλγορίθμου (από το A μέχρι το F) που προτάθηκε από τον Kononenko. Εδώ επισημαίνονται τρόποι που ο ReliefF διαφέρει από τον Relief. Πρώτο, ο ReliefF βασίζεται στον αριθμό των γειτόνων, μια παράμετρο k που ορίζεται από το χρήστη, που προσδιορίζει τη χρήση k κοντινότερων hits και k κοντινότερων misses στην ενημέρωση των βαθμών για κάθε δείγμα στόχο (αντί για ένα μόνο hit και miss). Αυτή η αλλαγή αύξησε την αξιοπιστία της εκτίμησης βάρους, ειδικά σε θορυβώδη προβλήματα. Ένα k με τιμή 10 προτάθηκε με βάση προκαταρκτικό εμπειρικό έλεγχο και έχει ευρέως υιοθετηθεί ως η προκαθορισμένη ρύθμιση. Αυτή η παραλλαγή του αλγορίθμου είχε αρχικά προταθεί υπό την ονομασία ReliefA.

Δεύτερο, τρεις διαφορετικές στρατηγικές προτάθηκαν για το χειρισμό απουσιάζουσων τιμών. Αυτές οι στρατηγικές προτάθηκαν υπό τις ονομασίες Relief(B-D). Όταν συναντάται μια απουσιάζουσα τιμή, η καλύτερη προσέγγιση (ReliefD), θέτει τη συνάρτηση $diff$ ίση με τη δεσμευμένη πιθανότητα κλάσης ότι δύο δείγματα έχουν διαφορετικές τιμές για το δεδομένο χαρακτηριστικό.

Τρίτο, δύο διαφορετικές στρατηγικές προτάθηκαν για το χειρισμό πολλών κλάσεων. Αυτές οι στρατηγικές προτάθηκαν υπό τις ονομασίες ReliefE και ReliefF. Ο ReliefF, ο οποίος κληρονόμησε τις αλλαγές που προτάθηκαν στους ReliefA και ReliefD, επιλέχθηκε ως η καλύτερη προσέγγιση. Κατά τη διάρκεια της βαθμολόγησης σε προβλήματα πολλών κλάσεων, ο ReliefF βρίσκει τα k κοντινότερα misses για κάθε άλλη κλάση, και σταθμίζει την ενημέρωση του βάρους με βάση τις πιθανότητες κάθε κλάσης. Εννοιλογικά, αυτό ενθαρρύνει τον αλγόριθμο να εκτιμήσει την ικανότητα των χαρακτηριστικών να διαχωρίσουν όλα τα ζεύγη κλάσεων ανεξάρτητα από το ποιές δύο κλάσεις είναι πιο κοντά μεταξύ τους. Τέλος, αφού είναι αναμενόμενο ότι καθώς η παράμετρος m πλησιάζει τον συνολικό αριθμό των δειγμάτων

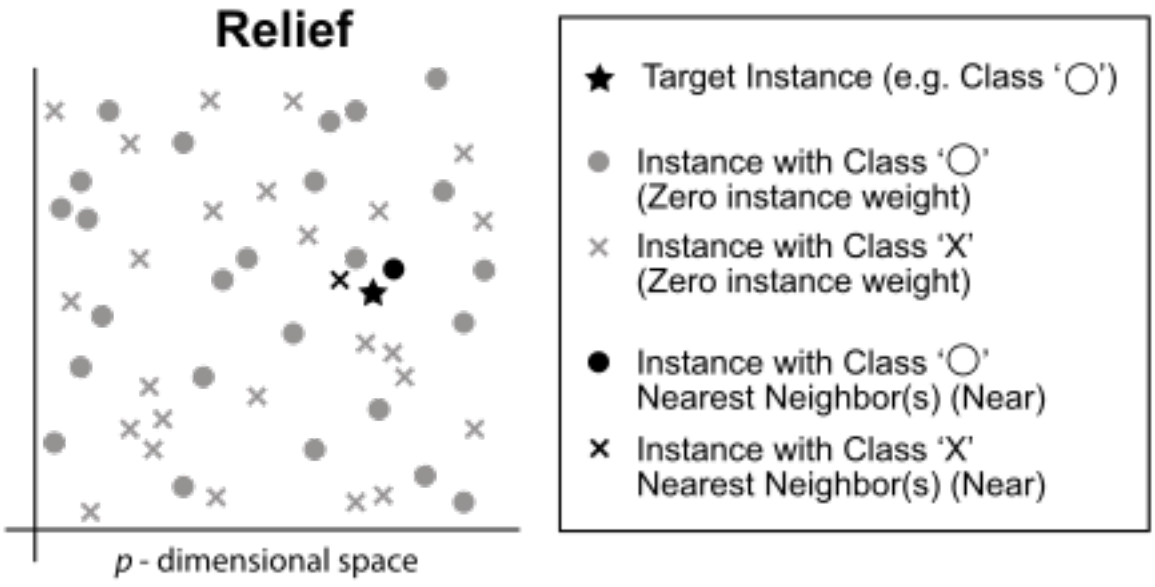
n, η ποιότητα των εκτιμήσεων των βαρών γίνεται πιο αξιόπιστη, ο Kononenko [3] πρότεινε την απλοποιητική υπόθεση ότι $m = n$. Με άλλα λόγια, κάθε δείγμα στο σύνολο δεδομένων γίνεται δείγμα στόχος μια φορά.

Ο Todorov [15] πρότεινε ότι υπάρχουν δύο κύριες κατευθύνσεις στην ανάπτυξη των RBAs: (1) στρατηγικές για την επιλογή και/ή την αντιστοίχιση βαρών στους γείτονες στη βαθμολόγηση (δηλ. αυτό που λέμε αναπτύξεις των κύριων αλγορίθμων), και (2) στρατηγικές για παραπάνω από ένα περάσματα από τα δεδομένα με επαναληπτικές υλοποιήσεις.

2.3 Επιλογή γειτόνων και ανάθεση βαρών δειγμάτων

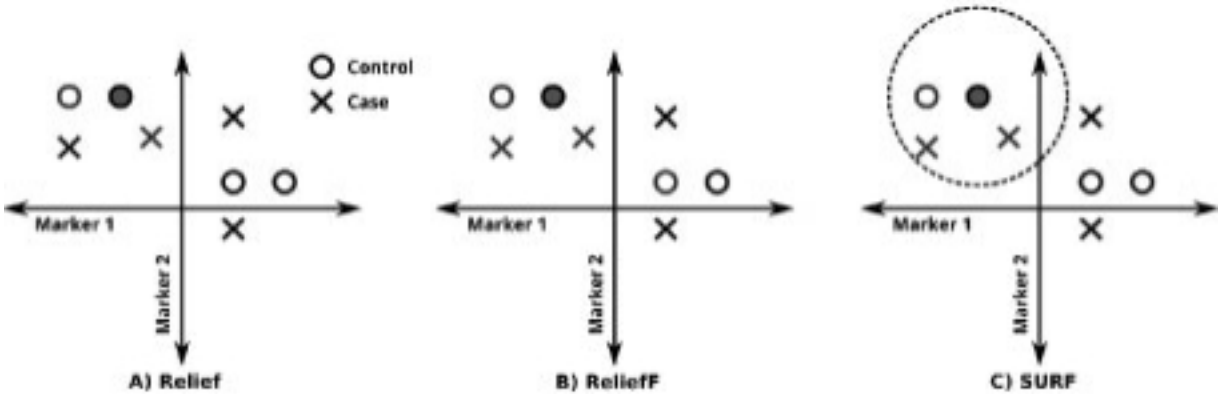
Ο αρχικός αλγόριθμος Relief χρησιμοποιούσε δύο κοντινότερους γείτονες (δηλ. ένα κοντινότερο hit και miss), το κάθε ένα με μια ίση απόδοση βάρους δείγματος [4]. Από τον ReliefA μέχρι τον ReliefF χρησιμοποιούνταν k κοντινότεροι γείτονες με ίση απόδοση βάρους δείγματος [3]. Ο επαναληπτικός Relief ήταν ο πρώτος που προσδιόρισε μια ακτίνα r γύρω από το δείγμα στόχο που θα όριζε το όριο για το οποίο δείγματα θα θεωρούνταν γείτονες [17]. Το αποτέλεσμα ήταν ότι οι κοντινότεροι γείτονες είχαν μια μεγαλύτερη επιρροή στην απόδοση βαρών από αυτούς προς την άκρη της ακτίνας. Όμοια με τον επαναληπτικό Relief, ο SURF χρησιμοποίησε ένα κατώφλι απόστασης T για να ορίσει δείγματα ως γείτονες (όπου το T ήταν ίσο με τη μέση απόσταση όλων των ζευγών δειγμάτων στα δεδομένα) [8]. Ωστόσο, σε αντίθεση με τον επαναληπτικό Relief, ο SURF χρησιμοποιεί ίσα βάρη δειγμάτων για όλα τα δείγματα που ορίζονται ως γείτονες.

Σχήμα 2.1: Αλγόριθμος Relief: Επιλογή γειτόνων



[13]

Σχήμα 2.2: Αλγόριθμοι Relief, ReliefF και SURF: Επιλογή γειτόνων



[14]

2.4 Αλγόριθμος SURF

Οι Greene-Penrod κ.ά. [8] εισήγαγαν τον Spatially Uniform ReliefF (SURF) και κατέληξαν στο αποτέλεσμα ότι η ικανότητα του SURF να ανιχνεύει αλληλεπιδράσεις σε γενετικά σύνολα δεδομένων είναι σημαντικά μεγαλύτερη από αυτή του ReliefF. Όμοια, ο SURF σε συνδυασμό με τη στρατηγική TuRF ξεπερνάει ως προς την επίδοση σημαντικά τον TuRF μόνο του. Είναι σημαντικό να σημειωθεί ότι αυτή η αύξηση του ρυθμού επιτυχίας δεν απαιτεί αύξηση στην αλγοριθμική πολυπλοκότητα και επιτρέπει αυξημένο ρυθμό επιτυχίας, ακόμα και με την αφαίρεση μιας ενοχλητικής παραμέτρου από τον αλγόριθμο.

2.5 Επαναληπτικές προσεγγίσεις

Όπως σημειώθηκε νωρίτερα, έχει γίνει κατανοητό ότι η επίδοση των κύριων RBAs υποβιβάζεται καθώς ο αριθμός των άσχετων χαρακτηριστικών γίνεται μεγάλος ιδιαίτερα σε σχέση με θορυβώδη προβλήματα. Όπως επισημαίνεται από τους Sun και Lee [18], αυτό συμβαίνει επειδή ένας βασικός RBA ορίζει κοντινότερους γείτονες στον αρχικό χώρο χαρακτηριστικών, οι οποίοι είναι αρκετά απίθανο να είναι ίδιοι στον βεβαρημένο χώρο (δηλ. στον χώρο όπου έχουμε αναθέσει χαμηλά βάρη σε χαρακτηριστικά που είναι το λιγότερο πιθανό να είναι σχετικά). Για να αντιμετωπιστεί αυτό το θέμα, έχουν προταθεί επαναληπτικές προσεγγίσεις που είναι ολοκληρωμένες με κύριους RBAs.

Ο TuRF παρουσιάζει μια αρκετά απλή επαναληπτική προσέγγιση που μπορεί εύκολα να ενθυλακώσει κάθε άλλο κύριο RBA παρά το γεγονός ότι αρχικά σχεδιάστηκε για να χρησιμοποιηθεί με τον ReliefF [7]. Ο TuRF είναι ουσιαστικά μια αναδρομική προσέγγιση απαλοιφής χαρακτηριστικών. Σε κάθε επανάληψη, τα χαρακτηριστικά με τους χαμηλότερους βαθμούς απαλοίφονται σχετικά με ότι αφορά υπολογισμούς αποστάσεων και ενημερώσεις βαρών χαρακτηριστικών. Ωστόσο η επιλογή του αριθμού των επαναλήψεων (p) δεν είναι τετριμμένη.

2.6 Αλγόριθμος TuRF

Το κύριο συμπέρασμα της μελέτης των Moore-White [7] ήταν ότι ο αλγόριθμος ReliefF είναι ικανός να αναγνωρίσει μη γραμμικές εξαρτήσεις χαρακτηριστικών σε γενετικά σύνολα δεδομένων. Ενώ αυτό ήταν ενθαρρυντικό, η ισχύς του ReliefF προσέγγιζε ένα λογικό επίπεδο (80%) όταν σχεδόν 500 χαρακτηριστικά επιλέγονταν από τα 1000. Είναι γνωστό ότι η ισχύς του ReliefF επηρεάζεται σημαντικά από τον αριθμό των θορυβωδών χαρακτηριστικών και τον αριθμό των δειγμάτων σε ένα σύνολο δεδομένων [5]. Αυτό οφείλεται στο ότι ο ReliefF εξετάζει ολόκληρο το διάνυσμα των χαρακτηριστικών κατά την εκτίμηση της ποιότητας του κάθε ξεχωριστού χαρακτηριστικού.

Ο στόχος τους ήταν να βελτιώσουν τον ReliefF με συστηματική αφαίρεση χαρακτηριστικών με τα χειρότερα βάρη ακολουθούμενη από επανεκτίμηση των βαρών για τα εναπομείναντα χαρακτηριστικά. Ως αποτέλεσμα, ο αλγόριθμος είναι πολύ ευαίσθητος στο πλαίσιο που βρίσκονται τα λειτουργικά χαρακτηριστικά. Η λογική είναι ότι το σήμα πρέπει να βελτιωθεί αφαιρώντας εκείνα τα χαρακτηριστικά που είναι πιο πιθανό να είναι θόρυβος. Τα αποτελέσματα της προσομοιωτικής τους μελέτης επέδειξαν ότι ο αλγόριθμος Tuned ReliefF (TuRF) είναι σημαντικά καλύτερος από τον ReliefF. Πράγματι, ο αλγόριθμος TuRF είχε μεγαλύτερη από 80% ισχύ να επιλέξει τις σωστές δύο αλληλεπιδράσεις γονιδίων σε ένα υποσύνολο από μόνο 50 από τα 1000 χαρακτηριστικά. Ο ReliefF είχε λιγότερη από 50% ισχύ.

Κεφάλαιο 3

Θεωρητικό υπόβαθρο

3.1 Αλγόριθμος Relief

Ο Relief είναι αλγόριθμος βασισμένος σε βάρη χαρακτηριστικών. Δεδομένου ενός συνόλου δεδομένων εκπαίδευσης S, αριθμού δειγμάτων m και ενός κατωφλίου σχετικότητας τ, ο Relief ανιχνεύει αυτά τα χαρακτηριστικά τα οποία είναι στατιστικά σχετικά. Υποθέτουμε ότι οι τιμές των χαρακτηριστικών είναι κατηγορικές (ενδεχομένως λογικές) ή αριθμητικές (ακέραιες ή πραγματικές). Διαφορές μεταξύ των τιμών των χαρακτηριστικών μεταξύ δύο δειγμάτων X και Y ορίζονται από την ακόλουθη συνάρτηση $diff$.

Αν τα x_k και y_k είναι κατηγορικά,

$$diff(x_k, y_k) = \begin{cases} 0, x_k = y_k \\ 1, x_k \neq y_k \end{cases}$$

Αν τα x_k και y_k είναι αριθμητικά,

$$diff(x_k, y_k) = (x_k - y_k)/nu_k,$$

όπου nu_k είναι μια σταθερά κανονικοποίησης που κανονικοποιεί τις τιμές της $diff$ στο διάστημα $[0, 1]$

Ο Relief επιλέγει ένα σύνολο αποτελούμενο από m τριάδες από δείγματα X και των $Near-hit$ και $Near-miss$ δειγμάτων τους. Ο Relief χρησιμοποιεί την d-διάστατη Ευκλείδεια απόσταση για την επιλογή των $Near - hit$ και $Near - miss$. Ο Relief καλεί μια συνάρτηση για να ενημερώσει το διάνυσμα βαρών χαρακτηρισιτκών W για κάθε σύνολο τριάδων και να καθορίσει το διάνυσμα μέσων βαρών χαρακτηριστικών $Relevance$. Τέλος, ο Relief επιλέγει εκείνα τα χαρακτηριστικά των οποίων το μέσο βάρος (relevance level) είναι πάνω από το δεδομένο κατώφλι τ.

Η πολυπλοκότητα του Relief είναι $\Theta(pmn)$. Αφού το m είναι μια τυχαία επιλεγμένη σταθερά, η πολυπλοκότητα είναι $\Theta(pn)$. Άρα ο αλγόριθμος μπορεί να επιλέγει στατιστικά σχετικά χαρακτηριστικά σε γραμμικό χρόνο ως προς τον αριθμό των χαρακτηριστικών και τον αριθμό των δειγμάτων εκπαίδευσης. [1]

Algorithm 1 Relief

Relief(S, m, τ)
Separate S into $S^+ = positiveInstances$ and $S^- = negativeInstances$
$W = (0, 0, ..., 0)$
for $i = 1$ to m **do**
    Pick at random an instance $X \in S$
    Pick at random one of the positive instances closest to X, $Z^+ \in S^+$
    Pick at random one of the negative instances closest to X, $Z^- \in S^-$
    if X is a positive instance **then**
        $Near - hit = Z^+$
        $Near - miss = Z^-$
    else
        $Near - hit = Z^-$
        $Near - miss = Z^+$
    end if
    $update - weight(W, X, Near - hit, Near - miss)$
    $Relevance = (1/m)W$
end for
for $i = 1$ to p **do**
    if $relevance_i \geq \tau$ **then**
        f_i is a relevant feature
    else
        f_i is an irrelevant feature
    end if
end for
update-weight$(W, X, Near - hit, Near - miss)$
for $i = 1$ to p **do**
    $W_i = W_i - diff(x_i, near - hit_i)^2 + diff(x_i, near - miss_i)^2$
end for

3.2 Επεκτάσεις του αλγορίθμου Relief - Αλγόριθμος ReliefF

3.2.1 Αξιόπιστη προσέγγιση βαρών

Η επιλογή των κοντινότερων γειτόνων είναι κρίσιμης σημασίας στον Relief. Ο σκοπός είναι να βρεθούν οι κοντινότεροι γείτονες σύμφωνα με τα σημαντικά χαρακτηριστικά. Περιττά και θορυβώδη χαρακτηριστικά μπορεί να επηρεάσουν σημαντικά την επιλογή των κοντινότερων γειτόνων και άρα η εκτίμηση των βαρών με θορυβώδη δεδομένα γίνεται αναξιόπιστη. Για να αυξήσει την αξιοπιστία της προσέγγισης των βαρών ο ReliefF αναζητά τα k κοντινότερα hits/misses αντί μόνο του ενός και παίρνει τον μέσο όρο της συνεισφοράς όλων των k κοντινότερων hits/misses. [2]

3.2.2 Ελειπή δεδομένα

Για να επιτρέψουμε στον Relief να αντιμετωπίσει ελειπή σύνολα δεδομένων, η συνάρτηση $diff(Attribute, Instance1, Instance2)$ στον ReliefF επεκτείνεται για απουσιάζουσες τιμές χαρακτηριστικών υπολογίζοντας την πιθανότητα δύο δεδομένα δείγματα να έχουν διαφορετικές τιμές για το δεδομένο χαρακτηρισιτκό:

- αν ένα δείγμα (π.χ. το $I1$) έχει άγνωστη τιμή:

$$diff(A, I_1, I_2) = 1 - P(value(A, I_2)|class(I_1))$$

- αν και τα δύο δείγματα έχουν άγνωστη τιμή:

$$diff(A, I_1, I_2) = 1 - \sum_{V}^{|values(A)|} P(V|class(I_1)) \times P(V|class(I_2))$$

Οι δεσμευμένες πιθανότητες προσεγγίζονται από σχετικές συχνότητες από το σύνολο δεδομένων εκπαίδευσης. [2]

3.2.3 Προβλήματα πολλών κλάσεων

Αντί να βρίσκει ένα κοντινό miss M από μια διαφορετική κλάση, ο ReliefF αναζητά k κοντινά misses $M_i, i = 1, 2, ..., k$ για κάθε διαφορετική κλάση C και παίρνει τον μέσο όρο της συνεισφοράς τους για την ενημέρωση της εκτίμησης $W(A)$.

$$W(A) = W(A) - \sum_{i=1}^{k} \frac{diff(A, R, H_i)}{n \times k} + \sum_{C \neq class(R)} \sum_{i=1}^{k} \frac{P(C)}{1 - P(class(R))} \times \frac{diff(A, R, M_i(C))}{n \times k}$$

Η ιδέα είναι ότι ο αλγόριθμος πρέπει να εκτιμάει την ικανότητα των χαρακτηριστικών να διαχωρίζουν κάθε ζεύγος κλάσεων ανεξάρτητα από το ποιές δύο κλάσεις είναι κοντινότερες μεταξύ τους.

Η χρονική πολυπλοκότητα του ReliefF είναι $O(N^2 \times |attributes|)$, όπου N είναι ο αριθμός των δειγμάτων εκπαίδευσης. [2]

3.2.4 Αλγόριθμος ReliefF

Η κεντρική ιδέα του ReliefF είναι η αποτίμηση της ποιότητας των χαρακτηριστικών με βάση την ικανότητά τους να ξεχωρίζουν δείγματα μιας κλάσης από κάποιας άλλης σε μια τοπική γειτονιά, δηλαδή, τα καλύτερα χαρακτηριστικά είναι αυτά που συνεισφέρουν περισσότερο στην αύξηση της απόστασης μεταξύ δειγμάτων διαφορετικών κλάσεων ενώ συνεισφέρουν λιγότερο στην αύξηση της απόστασης μεταξύ δειγμάτων της ίδιας κλάσης. [5, 19]

Algorithm 2 ReliefF

calculate prior probabilities $P(C)$ for all classes
set all weights $W[A] = 0.0$
for $i = 1$ to m **do**
  randomly select an instance R_i
  find k nearest hits H_j
  for all classes $C \neq cl(R_i)$ **do**
    from class C find k nearest misses $M_j(C)$
  end for
  for $A = 1$ to a **do**
    $\overline{H} = -\sum_{j=1}^{k} diff(A, R_i, H_j)/k$
    $\overline{M} = \sum_{C \neq cl(R_i)} \frac{P(C)}{1-P(cl(R_i))} \sum_{j=1}^{k} diff(A, R_i, M_j(C))/k$
    $W[A] = W[A] + (\overline{H} + \overline{M})/m$
  end for
end for
return W

Όπως μπορούμε να παρατηρήσουμε, ο αλγόριθμος ReliefF αποτελείται από ένα κύριο βρόχο που επαναλαμβάνεται m φορές, όπου m αντιστοιχεί στον αριθμό των δειγμάτων από δεδομένα προς εκτίμηση ποιότητας. Κάθε επιλεγμένο δείγμα R_i συνεισφέρει ισάξια στο διάνυσμα βαρών W μεγέθους a, όπου a είναι ο αριθμός των χαρακτηριστικών στο σύνολο δεδομένων. Η συνεισφορά για το A-οστό χαρακτηριστικό υπολογίζεται βρίσκοντας πρώτα k κοντινότερους γείτονες του δείγματος για κάθε κλάση στο σύνολο δεδομένων. Οι k γείτονες που ανήκουν στην ίδια κλάση με το δείγμα λέγονται hits (H), και οι άλλοι $k \cdot (c-1)$ γείτονες λέγονται misses (M), όπου c είναι ο συνολικός αριθμός των κλάσεων, και $cl(R_i)$, αντιστοιχεί στην κλάση του i-οστού δείγματος. Μόλις οι γείτονες βρεθούν, υπολογίζονται οι αντίστοιχες συνεισφορές τους στο A-οστό χαρακτηριστκό. Η συνεισφορά του συνόλου των hits $\overline{H}$ είναι ίση με το αντίθετο του μέσου όρου των διαφορών μεταξύ του δείγματος και κάθε hit. Αξίζει να σημειωθεί ότι αυτή είναι μια αρνητική συνεισφορά επειδή μόνο μη επιθυμητά χαρακτηριστικά πρέπει να συνεισφέρουν στη δημιουργία διαφορών μεταξύ γειτονικών δειγμάτων της ίδιας κλάσης. Ανάλογα, η συνεισφορά του συνόλου των misses $\overline{M}$ είναι ίση με τον βεβαρημένο μέσο όρο των διαφορών μεταξύ του δείγματος και κάθε miss. Αυτή είναι μια θετική συνεισφορά επειδή καλά χαρακτηριστικά πρέπει να βοηθάνε στη διαφοροποίηση μεταξύ δειγμάτων διαφορετικής κλάσης. Τα βάρη για αυτή την άθροιση ορίζονται σύμφωνα με την πιθανότητα κάθε κλάσης, υπολογισμένη από το σύνολο δεδομένων. Τέλος, αξίζει να αναφερθεί ότι η πρόσθεση των $\overline{H}$ και $\overline{M}$ και η διαίρεση με m, απλά υποδηλώνει άλλο ένα μέσο όρο μεταξύ των συνεισφορών

όλων των m δειγμάτων. Αφού η συνάρτηση $diff$ επιστρέφει τιμές ανάμεσα στα 0 και 1, τα βάρη του ReliefF θα είναι στο διάστημα $[-1, 1]$, και πρέπει να ερμηνεύονται προς τη θετική κατεύθυνση: όσο μεγαλύτερο το βάρος, τόσο μεγαλύτερη η σχετικότητα του αντίστοιχου χαρακτηριστικού. [19]

Η συνάρτηση $diff$ χρησιμοποιείται σε δύο περιπτώσεις στον αλγόριθμο ReliefF. Ο προφανής είναι μεταξύ των γραμμών 10 και 11 για τον υπολογισμό του βάρους. Χρησιμοποιείται ακόμα και για την εύρεση αποστάσεων μεταξύ δειγμάτων, ορισμένων ως το άθροισμα των διαφορών για κάθε χαρακτηριστικό (απόσταση Manhattan). Η αρχική συνάρτηση $diff$ που χρησιμοποιείται για τον υπολογισμό της διαφοράς δύο δειγμάτων I_1 και I_2 για ένα συγκεκριμένο χαρακτηριστικό A ορίζεται στην (3.1) για κατηγορικά χαρακτηριστικά, και στη (3.2) για αριθμητικά χαρακτηριστικά. Ωστόσο, η τελευταία έχει αποδειχθεί ότι προκαλεί μια υποεκτίμηση των αριθμητικών χαρακτηριστικών σε σχέση με τα κατηγορικά σε σύνολα δεδομένων που περιέχουν και τους δύο τύπους χαρακτηριστικών. Εκ τούτου, προτάθηκε μια συνάρτηση ράμπας (3.2) για να αντιμετωπιστεί αυτό το πρόβλημα [6]. Η ιδέα πίσω από αυτό είναι η χαλάρωση της σύγκρισης ισότητας στην (3.2) με τη χρήση δύο κατωφλίων: t_{eq} είναι η μέγιστη απόσταση μεταξύ δύο χαρακτηριστικών ώστε να θεωρηθούν ίσα, και ανάλογα, t_{diff} είναι η ελάχιστη απόσταστη μεταξύ δύο χαρακτηριστικών ώστε να θεωρηθούν διαφορετικά. Οι προκαθορισμένες τους τιμές τίθενται στο 5 και στο 10% του διαστήματος τιμών του χαρακτηριστικού, αντίστοιχα. [19] Επιπρόσθετα, υπάρχουν άλλες εκδόσεις της συνάρτησης $diff$ για την αντιμετώπιση απουσιαζόντων δεδομένων.

$$diff(A, I_1, I_2) = \begin{cases} 0, value(A, I_1) = value(A, I_2) \\ 1, value(A, I_1) \neq value(A, I_2) \end{cases} \tag{3.1}$$

$$diff(A, I_1, I_2) = \frac{|value(A, I_1) - value(A, I_2)|}{max(A) - min(A)} \tag{3.2}$$

$$diff(A, I_1, I_2) = \begin{cases} 0, d \leq t_{eq} \\ 1, d > t_{diff} \\ \frac{d - t_{eq}}{t_{diff} - t_{eq}}, t_{eq} < d \leq t_{diff} \end{cases} \tag{3.3}$$

3.3 Αλγόριθμος TuRF

Ο ReliefF είναι ικανός να συλλάβει αλληλεπιδράσεις χαρακτηριστικών επειδή επιλέγει κοντινότερους γείτονες χρησιμοποιώντας ολόκληρο το διάνυσμα τιμών όλων τα χαρακτηριστικών. Ωστόσο, αυτό το πλεονέκτημα είναι και μειονέκτημα επειδή η ύπαρξη πολλών θορυβωδών χαρακτηριστικών μπορεί να μειώσει το σήμα που ο αλγόριμος προσπαθεί να συλλάβει. Ο Tuned ReliefF (TuRF) αφαιρεί συστηματικά χαρακτηριστικά που έχουν χαμηλές εκτιμήσεις ποιότητας έτσι ώστε ο ReliefF να αποτιμά αν τα εναπομείναντα χαρακτηριστικά μπορούν να επανεκτιμηθούν. [7]

Algorithm 3 TuRF

1: let a be the number of attributes
2: **for** $i = 1$ to n **do**
3: estimate ReliefF
4: sort attributes
5: remove worst n/a attributes
6: **end for**
7: **return** return last ReliefF estimate for each attribute

Το κίνητρο πίσω από αυτό τον αλγόριθμο είναι ότι οι εκτιμήσεις των αληθινών λειτουργικών χαρακτηριστικών του ReliefF θα βελτιωθούν καθώς τα θορυβώδη χαρακτηριστικά αφαιρούνται από το σύνολο δεδομένων [7].

3.4 Αλγόριθμος SURF

Όλοι οι Relief αλγόριθμοι αντιστοιχούν ένα βάρος σε κάθε χαρακτηριστικό. Όσο υψηλότερο το βάρος ενός χαρακτηριστικού, τόσο πιθανότερο είναι να προβλέπει τη κλάση. Δείγματα με όμοιες τιμές χαρακτηριστικών χρησιμοποιούνται για την προσαρμογή αυτών των βαρών. Ορίζουμε την απόσταση μεταξύ δύο δειγμάτων ως τον αριθμό των χαρακτηριστικών τους με διαφορετικές τιμές. Με αυτή τη μετρική απόστασης, κοντινότεροι γείτονες είναι πιο όμοιοι ως προς τις τιμές των χαρακτηριστικών.

Οι Relief αλγόριθμοι βασίζονται στην υπόθεση ότι αυτά τα χαρακτηριστικά κοντινών δειγμάτων προβλέπουν περισσότερο ή λιγότερο την κλάση. Οι αλγόριθμοι Relief προσαρμόζουν τα βάρη αυτών των χαρακτηριστικών προς τα πάνω αν τα δύο δείγματα έχουν διαφορετική κλάση, και προς τα κάτω κατά την ίδια ποσότητα αν έχουν την ίδια κλάση. Για την ακρίβεια, ο αρχικός αλγόριθμος Relief προσαρμόζει, για κάθε δείγμα I_i, τα βάρη των χαρακτηριστικών χρησιμοποιώντας το κοντινότερο hit του I_i (το δέιγμα το οποίο είναι κοντινότερο στο I_i και στην ίδια κλάση με το I_i) και το κοντινότερο miss του I_i (το δείγμα το οποίο είναι κοντινότερο στο I_i και σε άλλη κλάση από το I_i). Στην περίπτωση του SURF, για κάθε δείγμα I_i, αυτή η προσαρμογή γίνεται χρησιμοποιώντας κάθε hit και miss εντός ενός σταθερού κατωφλίου απόστασης T από το I_i. [8]

Κεφάλαιο 4

Πειραματική Αξιολόγηση

4.1 Παράμετροι αξιολόγησης

Η στρατηγική για την αξιολόγηση μιας προσέγγισης επιλογής χαρακτηριστικών μπορεί να εξαρτάται από το αν η έξοδος είναι μια λίστα κατάταξης χαρακτηριστικών ή ένα συγκεκριμένο υποσύνολο χαρακτηριστικών. Υποθέτοντας μια λίστα κατάταξης χαρακτηριστικών και ένα σύνολο δεδομένων όπου η βασική αλήθεια είναι γνωστή από πριν, το πιο σύνηθες είναι η εξέταση του που κατατάσσονται όλα τα σχετικά χαρακτηριστικά στη διατεταγμένη λίστα βαθμών χαρακτηριστικών [10]. Ιδανικά, όλα τα σχετικά χαρακτηριστικά θα έχουν υψηλότερους βαθμούς από τα άσχετα χαρακτηριστικά αλλά το πιο σημαντικό είναι τα σχετικά χαρακτηριστικά τουλάχιστον να είναι πάνω από το επίπεδο διαχωρισμού του επιλεγμένου υποσυνόλου χαρακτηριστικών.

Εναλλακτικά, αν η προσέγγιση επιλογής χαρακτηριστικών δίνει ως έξοδο ένα υποσύνολο χαρακτηριστικών μπορούμε να αξιολογήσουμε την επιτυχία (1) εξετάζοντας τον αριθμό των σχετικών και άσχετων χαρακτηριστικών που αποτελούν ένα επιλεγμένο υποσύνολο χαρακτηριστικών (υποθέτοντας ότι η βασική αλήθεια είναι γνωστή) ή (2) καθορίζοντας την ακρίβεια ελέγχου κάποιου μοντέλου αλγορίθμου γενίκευσης εκπαιδευμένου σε αυτό το υποσύνολο χαρακτηριστικών (αν η βασική αλήθεια δεν είναι γνωστή) [10]. Το μειονέκτημα της δεύτερης προσέγγισης είναι η δυσκολία του διαχωρισμού της επίδοσης της προσέγγισης επιλογής χαρακτηριστικών από τη μοντελοποίηση του αλγορίθμου γενίκευσης. Αν αντιμετωπίζουμε μια προσέγγιση που έχει χρησιμοποίησει ένα επίπεδο επιλογής για να ορίσει ένα υποσύνολο χαρακτηριστικών, τότε το μειονέκτημα της πρώτης προσέγγισης είναι ότι αξιολογούμε την αντιστοίχιση βαρών στα χαρακτηριστικά αλλά και το κριτήριο διαχωρισμού (το οποίο μπορεί επίσης να είναι δύσκολο να ξεχωριστεί).

4.2 ReBATE

Για να διευκολυνθεί η πρόσβαση στους διάφορους RBAs και να προωθηθεί η τρέχουσα ανάπτυξη και εφαρμογή, οι Urbanowicz κ.ά. [10] υλοποίησαν το Relief-Based Algorithm Training Environment (ReBATE). Με το ReBATE επιδιώκεται να ισορροπηθεί η ευελιξία

τύπων δεδομένων, η αποδοτικότητα του χρόνου εκτέλεσης και η άνεση της ανάπτυξης σε ένα Python package framework. Το ReBATE έχει υλοποιηθεί με τους κύριους RBAs στους οποίους περιλαμβάνονται οι ReliefF, SURF και ο επαναληπτικός αλγόριθμος TuRF που εξετάζονται σε αυτή την εργασία. Αυτοί οι RBAs επιλέχθηκαν για αυτή την εργασία επειδή είχαν αναπυχθεί αποκλειστικά και προηγουμένως αξιολογηθεί για θορυβώδη προβλήματα και προβλήματα αλληλεπίδρασης χαρακτηριστικών και υπήρχαν διαθέσιμες υλοποιήσεις σε Python για κάθε ένα.

Το ReBATE περιλαμβάνει ένα στάδιο προεπεξεργασίας των δεδομένων που ανιχνεύει αυτόματα απαραίτητα χαρακτηριστικά τύπων δεδομένων. Συγκεκριμένα αυτό περιλαμβάνει (1) τη διάκριση διακριτών και αριθμητικών χαρακτηριστικών, (2) τη διάκριση διακριτής και αριθμητικής κλάσης, (3) αναγνώριση του μεγίστου-ελαχίστου διαστήματος τιμών για αριθμητικό χαρακτηριστικό ή κλάση, (4) για διακριτές κλάσεις, το καθορισμό του αριθμού των διαφορετικών κλάσεων (δηλ. binary ή multi-class) καθώς και τον αριθμό των δειγμάτων που έχουν την κάθε ετικέτα κλάσης και (5) την αναγνώριση της παρουσίας ελλειπών δεδομένων, με ένα καθιερωμένο αναγνωριστικό, π.χ. N/A. Αυτή η προεπεξεργασία αυτοματοποιεί τη προσαρμογή του κάθε RBA στους σχετικούς τύπους δεδομένων.

4.2.1 Μικτοί τύποι δεδομένων

Για αποδοτικότητα, η ReBATE προ-κανονικοποιεί κάθε συνεχή μεταβλητή (δηλ. χαρακτηριστικό ή κλάση) έτσι ώστε να ανήκει σέ ένα διάστημα τιμών από 0 μέχρι 1 [10]. Ενώ η συνάρτηση $diff$ αποδίδει καλά όταν τα χαρακτηριστικά είναι ομοιόμορφα διακριτά ή συνεχή, έχει σημειωθεί ότι δεδομένου ενός συνόλου δεδομένων με διακριτά και συνεχή χαρακτηριστικά, η συνάρτηση $diff$ μπορεί να υποτιμήσει την ποιότητα των συνεχών χαρακτηριστικών [12]. Μια προτεινόμενη λύση σε αυτό το πρόβλημα είναι μια συνάρτηση ράμπας που αναθέτει αφελώς στη $diff$ 0 ή 1 αν οι τιμές συνεχών χαρακτηριστικών απέχουν μεταξύ τους μια ελάχιστη ή μέγιστη απόσταση ορισμένη από το χρήστη. Ωστόσο αυτή η αφελής προσέγγιση προσθέτει δύο επιπλέον παραμέτρους ορισμένες από το χρήστη που απαιτούν βελτιστοποίηση για κάθε πρόβλημα.

4.2.2 Παλινδρόμηση

Η θεμελιώδης πρόκληση της προσαρμογής των Relief αλγορίθμων σε συνεχείς κλάσεις είναι ότι χάνεται ένας καθαρός ορισμός του hit ή miss, δηλαδή η έννοια της ίδιας ή διαφορετικής κλάσης. Ο αλγόριθμος Regressional ReliefF (RReliefF) [11] πρότεινε ένα είδος πιθανότητας δύο δείγματα να ανήκουν σε δύο διαφορετικές κλάσεις. Αυτή η πιθανότητα μοντελοποιείται με την απόσταση μεταξύ τιμών χαρακτηριστικού και κλάσης δύο δειγμάτων εκπαίδευσης όπως περιγράφεται από τους Robnik-Sikonja και Kononenko. Αυτό περιλαμβάνει μια εκθετική αντιστοίχιση βαρών στις συνεισφορές δειγμάτων στο $W[A]$ με βάση την απόσταση μεταξύ δειγμάτων. Οι τρέχουσες μέθοδοι της ReBATE δεν εφαρμόζουν βάρη δειγμάτων με βάση την απόσταση, αλλά αφού ο RReliefF απαιτεί ένα επιπλέον βήμα υπολογίζοντας τις πιθανότητες, οι μέθοδοι της ReBATE προτείνουν ένα απλούστερο σχήμα παλινδρόμησης [10].

Συγκεκριμένα, η ReBATE υπολογίζει την τυπική απόκλιση της συνεχούς κλάσης (σ_E) και την εφαρμόζει ως ένα απλό κατώφλι για τον προσδιορισμό του αν δύο δείγματα θα θεωρηθούν ένα hit ή ένα miss. Αυτό εξυπηρετεί στη διάκριση της συνεχούς κλάσης σε ίδια κλάση ή διαφορετική κλάση από τη προοπτική του δείγματος στόχου. Αυτή η προτεινόμενη προσαρμογή των RBAs σε προβλήματα παλινδρόμησης απαιτεί μόνο προ-υπολογισμό του σ_E, και αλλαγή του ορισμού του hit από $C_i = C_j$ σε $|C_i - C_j| < \sigma_E$, και του ορισμού του miss από $C_i \neq C_j$ σε $|C_i - C_j| \geq \sigma_E$.

4.2.3 Απουσιάζουσες τιμές

Οι απουσιάζουσες τιμές χαρακτηριστικών πρέπει να αντιμετωπιστούν από τους RBAs σε δύο σημεία στον αλγόριθμο: (1) Στον υπολογισμό των αποστάσεων μεταξύ ζευγών δειγμάτων και (2) στην ενημέρωση των βαρών των χαρακτηριστικών. Μια στρατηγική που προτάθηκε στον ReliefF (ή ακριβέστερα στον ReliefD) είχε θεωρηθεί ως η καλύτερη με ελάχιστη εμπειρική έρευνα [3]. Επίσης ήταν σχεδιασμένη αποκλειστικά για προβλήματα με διακριτές κλάσεις. Συγκεκριμένα, ανάλογα με το αν ένα ή και τα δύο δείγματα έχουν μια απουσιάζουσα τιμή για το δεδομένο χαρακτηριστικό, η συνάρτηση $diff$ επιστρέφει την πιθανότητα οι καταστάσεις των χαρακτηριστικών να είναι διαφορετικές δεδομένης της κλάσης του κάθε δείγματος. Αυτή η προσέγγιση είναι σιωπηρά μια μορφή παρεμβολής, κάνοντας μια εκπαιδευμένη εικασία ως προς το τί θα μπορούσε να είναι η απουσιάζουσα τιμή. Υπό τις σωστές συνθήκες, αυτό μπορεί όντως να βελτίωσει την επίδοση, αλλά αν η εικασία είναι λάθος, θα μπορούσε το ίδιο εύκολα να βλάψει την επίδοση. Ακόμα, αυτή η προσέγγιση παρουσιάζει μεγαλύτερη υπολογιστική και εννοιολογική πρόκληση για να επεκταθεί σε δεδομένα συνεχών κλάσεων.

Στη ReBATE προτείνεται αυτή που λέγεται αγνωστική προσέγγιση ως προς απουσιάζουσες τιμές που μοιάζει περισσότερο με μία που εξετάζεται στον ReliefC [3]. Η ιδέα πίσω από μια αγνωστική προσέγγιση είναι ότι άγνωστες, απουσιάζουσες τιμές πρέπει να αγνοούνται (δηλ. να αντιμετωπίζονται ουδέτερα) χρησιμοποιώντας κανονικοποίηση για να παρακαμφθεί η συμπερίληψή τους αντί της απόπειρας να γίνει μια εικασία για τις αντίστοιχες τιμές. Αντίθετα, η ReliefC μέθοδος είναι μόνο μερικά αγνωστική, χρησιμοποιεί τη ReliefB μέθοδο [3] για να συνεισφέρει αφελώς $diff$ με τιμή $1 - \frac{1}{\#Unique\_Feature\_Values}$ όταν μια απουσιάζουσα τιμή συναντάται στον υπολογισμό της απόστασης μεταξύ ενός ζεύγους δειγμάτων [3]. Για παράδειγμα, αυτή η συνεισφορά θα ήταν 0.5 αν το χαρακτηριστικό είχε δύο πιθανές καταστάσεις, ή 0.25 αν είχε τέσσερις. Ωστόσο, όταν ο ReliefC ενημερώνει τα βάρη των χαρακτηριστικών, χαρακτηριστικά με απουσιάζουσες τιμές δε συνεισφέρουν τίποτα και η απόσταση βαθμός κανονικοποιείται ώστε να αντικατοπρίζει ότι υπολογίστηκε χρησιμοποιώντας $(a - \#Missing\_Features)$, όπου $\#Missing\_Features$ είναι ο αριθμός των χαρακτηριστικών όπου μια απουσιάζουσα τιμή παρατηρήθηκε για τουλάχιστον ένα από τα δύο δείγματα. Εναλλακτικά, οι ReBATE μέθοδοι εφαρμόζουν αυτή την αγνωστική μεταχείριση των απουσιάζουσων τιμών τόσο στον υπολογισμό των αποστάσεων ζευγών δειγμάτων όσο και για τις ενημερώσεις των βαρών των χαρακτηριστικών. Αυτή η προσέγγιση εύκολα ολοκληρώνεται με όλους τους RBAs, και όλες τις άλλες επεκτάσεις τύπων δεδομένων [10].

4.3 Σύνολα δεδομένων

Κληρονομικότητα είναι ένας γενετικός όρος που υποδεικνύει πόσο μια διαφοροποίηση κλάσης οφείλεται στα γενετικά χαρακτηριστικά. Στο παρόν πλαίσιο, η κληρονομικότητα μπορεί να θεωρηθεί ως το πλάτος του σήματος, όπου μια κληρονομικότητα τιμής 1 είναι ένα καθαρό σύνολο δεδομένων (δηλ. με το σωστό μοντέλο, οι τιμές των κλάσεων θα προβλέπονται πάντα σωστά με βάση τιμές χαρακτηριστικών), και μια κληρονομικότητα τιμής 0 θα ήταν ένα τελείως θορυβώδες σύνολο δεδομένων χωρίς συσχετίσεις κλάσεων που να έχουν νόημα. Όλα τα χαρακτηριστικά προσομοιώνονται ως μονοί πολυμορφισμοί νουκλεοτιδίων (single nucleotide polymorphisms - SNP) που θα μπορούσαν να έχουν μια διακριτή τιμή (0, 1 ή 2) αναπαριστώντας πιθανούς γονότυπους.

Ακολουθεί περιγραφή των βασικών συνόλων δεδομένων:

- **GAMETES_Epistasis_2-Way_20atts_0.4H_EDM-1_1**:
  Αποτελείται από 20 διακριτά χαρακτηριστικά και διακριτή κλάση, 1600 δείγματα, και η κληρονομικότητά του είναι 0.4

- **GAMETES_Epistasis_2-Way_continuous_endpoint_a_20s_1600her_0.4__maf_0.2_EDM-2_01**:
  Αποτελείται από 20 διακριτά χαρακτηριστικά και συνεχή κλάση (παλινδρόμηση), 1600 δείγματα, και η κληρονομικότητά του είναι 0.4

- **GAMETES_Epistasis_2-Way_missing_values_0.1_a_20s_1600her_0.4__maf_0.2_EDM-2_01**: Αποτελείται από 20 διακριτά χαρακτηριστικά με απουσιάζουσες τιμές με συχνότητα 0.1 και διακριτή κλάση, 1600 δείγματα, και η κληρονομικότητά του είναι 0.4

- **GAMETES_Epistasis_2-Way_mixed_attribute_a_20s_1600her_0.4__maf_0.2_EDM-2_01**:
  Αποτελείται από 20 μικτά (διακριτά και συνεχή) χαρακτηριστικά και διακριτή κλάση, 1600 δείγματα, και η κληρονομικότητά του είναι 0.4

4.4 Εκτελέσεις του αλγορίθμου ReliefF

Οι διαφορετικές εκτελέσεις του ReliefF θα λέγονται στα αποτελέσματα: ReliefF 10 NN (δηλ. αυθεντικός ReliefF), ReliefF 100 NN, ReliefF 10% NN και ReliefF 50% NN. Οι πρώτοι δύο είναι ReliefF με k 10 και 100, αντίστοιχα. Οι επόμενοι δύο εξετάζουν την ανάθεση του k με τρόπο εξαρτώμενο από το σύνολο δεδομένων, θέτοντας το k με βάση ένα ορισμένο από το χρήστη ποσοστό των δειγμάτων στα δεδομένα. Για παράδειγμα, αν το n ήταν 1000 δείγματα, ο ReliefF 10% NN θα χρησιμοποιούσε 100 συνολικά δείγματα, άρα $k = 50$, δηλ. 50 hits και 50 misses.

4.5 Αποτελέσματα της μελέτης - Συμπερασμάτα αξιολόγησης

4.5.1 Επίσταση

Εστιάζοντας στην επίδοση του ReliefF για διαφορετικές τιμές του k προκύπτει μια παρατήρηση. Συγκεκριμένα ο ReliefF 100 NN αποτυγχάνει αν $n = 200$. Για αυτή τη τιμή του k σε ένα ισορροπημένο σύνολο δεδομένων, ο ReliefF χρησιμοποιεί όλα τα δείγματα ως γείτονες. Αυτό αφαιρεί δραστικά την προϋπόθεση ότι τα δείγματα που χρησιμοποιούνται στη βαθμολόγηση είναι κοντινά και μετατρέπει τον ReliefF (χρησιμοποιώντας όλους τους γείτονες) σε ένα μυοπικό αλγόριθμο, ανίκανο να χειριστεί αλληλεπιδράσεις χαρακτηριστικών. Αυτό επιβεβαιώνεται εμπειρικά από τα αποτελέσματα. Αντίθετα, εξέταση των αποτελεσμάτων του ReliefF 10 NN ($n = 1600$) σε σύγκριση με άλλους RBAs υποδηλώνει ότι αυξανόμενος θόρυβος αντιμετωπίζεται καλύτερα από ένα κάπως μεγαλύτερο k. Πρέπει να σημειωθεί ότι η έννοια ενός χαμηλού ή υψηλού k θα πρέπει πάντα να θεωρείται σε σχέση με το n. Για παράδειγμα, ενώ το $k = 100$ ήταν κακή επιλογή για ένα μέγεθος δείγματος 200, αυτή η ρύθμιση απέδοσε σχετικά καλά όταν το n ήταν 1600.

4.5.2 Αριθμός χαρακτηριστικών

Τα αποτελέσματα εκτελέστηκαν σε σύνολα δεδομένων με σχετικά μικρό αριθμό χαρακτηριστικών για να σωθεί υπολογιστικός χρόνος ενώ εξετάζεται η λειτουργία των αλγορίθμων. Αυξάνοντας τον αριθμό των χαρακτηριστικών, ο ReliefF με $k = 10$ αρχίζει να αποτυγχάνει υποδηλώνοντας ότι ένας μικρός αριθμός από γείτονες αποδίδει λιγότερο καλά σε θορυβώδη προβλήματα. Γενικά, σε συνδυασμό με μια επαναληπτική προσέγγιση, για παράδειγμα TuRF, τα χαμηλότερα κατεταγμένα χαρακτηριστικά θα αφαιρούνταν σε κάθε επανάληψη. Άρα, θα ήταν χρήσιμο εδώ να εξεταστεί ποιό ποσοστό των χαρακτηριστικών θα μπορούσε να αφαιρεθεί στη πρώτη (και τις επόμενες) επαναλήψεις χωρίς να χαθούν σχετικά χαρακτηριστικά.

4.5.3 Τύποι δεδομένων

Περιλαμβάνεται ανάλυση για διαφορετικούς τύπους δεδομένων. Συνολικά, η επιτυχία των RBA υλοποιήσεων της ReBATE υποδηλώνει η απλούστερη και υπολογιστικά λιγότερο ακριβή RBA παλινδρόμηση προσφέρει μια λειτουργική εναλλακτική αυτής που προτείνεται στον RReliefF. Ακόμα, προβλήματα πολλών κλάσεων λύθηκαν από όλες τις μεθόδους επιλογής χαρακτηριστικών, επιδεικνύοντας τη βασική αποτελεσματικότητα της επέκτασης πολλών κλάσεων για τον Relief που υποστηρίζει η ReBATE.

4.5.4 Απουσιάζουσες τιμές

Ιδιαίτερα για την ανάλυση απουσιάζουσων τιμών, οι μέθοδοι της ReBATE κατέταξαν τα χαρακτηριστικά ιδανικά, ενώ η εκτέλεση μεθόδων της scikit-learn δε θα μπορούσε να ολοκλη-

ρωθεί αφού η scikit-learn δεν είναι ρυθμισμένη να χειρίζεται απουσιάζουσες τιμές. Θα απαιτούταν προεπεξεργασία όπως αφαίρεση των δειγμάτων με απουσιάζουσες τιμές ή συμπλήρωση.

4.5.5 Ανισορροπία κλάσης

Για μια ανισορροπία κλάσης του 0.9 (δηλ. 90% κλάση 0, 10% κλάση 1), παρατηρείται ότι ο ReliefF με ένα μεγάλο αριθμό γειτόνων (δηλ. 50%) αποτυγχάνει να αποδώσει, ο ReliefF με 100 NN και ο SURF παρουσιάζουν ελαφριές απώλειες, αλλά οι υπόλοιποι RBAs αποδίδουν. Για δεδομένα με απουσιάζουσες τιμές με συχνότητα 0.5, παρατηρείται ότι όλες οι μέθοδοι ReliefF αποδίδουν με εξαίρεση τον ReliefF 10 NN (οι λιγότεροι γείτονες στη βαθμολόγηση). Αυτό υποδεικνυέι ότι οι περισσότεροι γείτονες στη βαθμολόγηση κάνουν αυτές τις μεθόδους πιο ανθεκτικές σε απουσιάζουσες τιμές, και επιδεικνύει ότι η υλοποιημένη στρατηγική απουσιάζουσων τιμών στη ReBATE είναι επιτυχής. Για μεικτά (διακριτά και συνεχή) χαρακτηριστικά, κανένας από τους RBAs δε χειρίζεται το σύνολο δεδομένων βέλτιστα. Καλύτερα δουλεύει ο ReliefF με 100 NN ή 10% NN. Γενικά αυτό υποδηλώνει ότι οι μικτοί τύποι χαρακτηριστικών είναι ακόμα ένα θέμα για τους RBAs.

Σχήμα 4.1: Παράδειγμα πειραματικής εκτέλεσης του αλγορίθμου ReliefF

ReliefF

June 3, 2021

[1]:
```
import pandas as pd
import numpy as np
from sklearn.pipeline import make_pipeline
from skrebate import ReliefF
from sklearn.model_selection import train_test_split
```

[2]:
```
genetic_data = pd.read_csv('https://github.com/EpistasisLab/scikit-rebate/raw/
↪master/data/'
                           'GAMETES_Epistasis_2-Way_20atts_0.4H_EDM-1_1.tsv.gz',
                           sep='\t', compression='gzip')

features, labels = genetic_data.drop('class', axis=1).values,␣
↪genetic_data['class'].values

# Make sure to compute the feature importance scores from only your training set
X_train, X_test, y_train, y_test = train_test_split(features, labels)

fs = ReliefF(n_neighbors=10)
fs.fit(X_train, y_train)

for feature_name, feature_score in zip(genetic_data.drop('class', axis=1).
↪columns,
                                       fs.feature_importances_):
    print(feature_name, '\t', feature_score)
```

```
N0      0.0005833333333333331
N1      -0.007000000000000002
N2      -0.00825000000000001
N3      -0.007499999999999999
N4      -0.008666666666666677
N5      -0.013416666666666683
N6      -0.009666666666666657
N7      -0.023250000000000007
N8      -0.00466666666666667
N9      -0.013333333333333329
N10     -0.0010833333333333333
N11     -0.017249999999999974
N12     -0.012333333333333347
```

Σχήμα 4.2: Παράδειγμα πειραματικής εκτέλεσης του αλγορίθμου SURF

SURF

June 3, 2021

[1]:
```
import pandas as pd
import numpy as np
from sklearn.pipeline import make_pipeline
from skrebate import SURF
from sklearn.model_selection import train_test_split
```

[2]:
```
genetic_data = pd.read_csv('https://github.com/EpistasisLab/scikit-rebate/raw/
↪master/data/'
                           'GAMETES_Epistasis_2-Way_20atts_0.4H_EDM-1_1.tsv.gz',
                           sep='\t', compression='gzip')

features, labels = genetic_data.drop('class', axis=1).values,␣
↪genetic_data['class'].values

# Make sure to compute the feature importance scores from only your training set
X_train, X_test, y_train, y_test = train_test_split(features, labels)

fs = SURF()
fs.fit(X_train, y_train)

for feature_name, feature_score in zip(genetic_data.drop('class', axis=1).
↪columns,
                                       fs.feature_importances_):
    print(feature_name, '\t', feature_score)
```

```
N0      -0.0009142462420710952
N1      -0.002119792317067996
N2      -0.004314666966543232
N3      -0.006569346792908346
N4      -0.0038171580209947166
N5      -0.0031332358299083142
N6      -0.006127121745179951
N7      -0.004441743496920665
N8      -0.0036209403838992436
N9      -0.0011051763661404793
N10     -0.00036406751773969284
N11     -0.0021745981518854645
N12     -0.004275400285346374
```

Σχήμα 4.3: Παράδειγμα πειραματικής εκτέλεσης του αλγορίθμου TuRF

TuRF

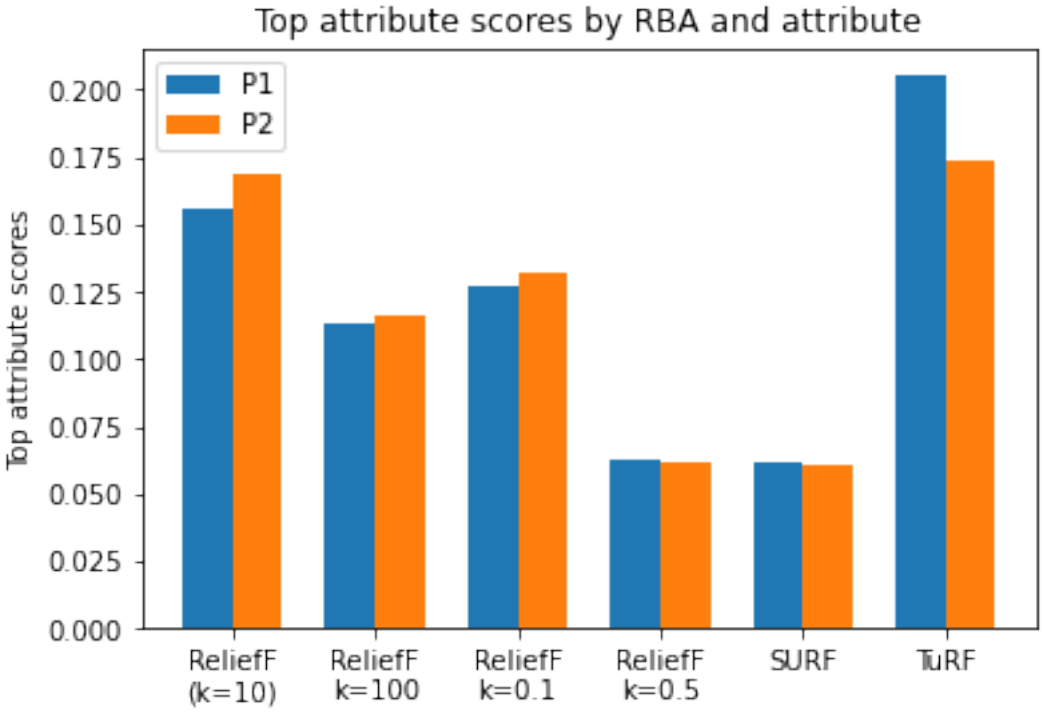
June 3, 2021

[1]:
```
import pandas as pd
import numpy as np
from sklearn.pipeline import make_pipeline
from skrebate.turf import TuRF
from sklearn.ensemble import RandomForestClassifier
from sklearn.model_selection import cross_val_score
```

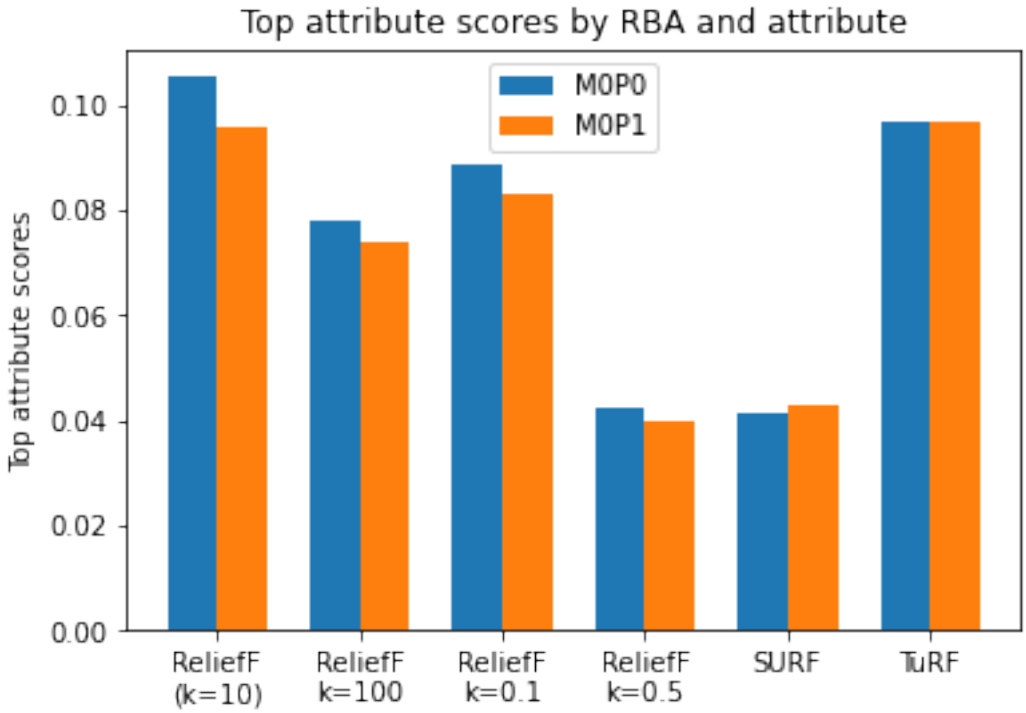
[2]:
```
genetic_data = pd.read_csv('https://github.com/EpistasisLab/scikit-rebate/raw/
↪master/data/'
                           'GAMETES_Epistasis_2-Way_20atts_0.4H_EDM-1_1.tsv.gz',
                           sep='\t', compression='gzip')

features, labels = genetic_data.drop('class', axis=1).values,␣
↪genetic_data['class'].values
headers = list(genetic_data.drop("class", axis=1))
fs = TuRF(core_algorithm="ReliefF", n_features_to_select=2, pct=0.
↪5,verbose=True)
fs.fit(features, labels, headers)
for feature_name, feature_score in zip(genetic_data.drop('class', axis=1).
↪columns, fs.feature_importances_):
    print(feature_name, '\t', feature_score)
```

```
Created distance array in 0.16866588592529297 seconds.
Feature scoring under way …
Completed scoring in 16.687040090560913 seconds.
Created distance array in 0.07659912109375 seconds.
Feature scoring under way …
Completed scoring in 9.909730911254883 seconds.
Created distance array in 0.05138111114501953 seconds.
Feature scoring under way …
Completed scoring in 7.595054864883423 seconds.
N0      -0.0010312500000000003
N1      -0.010751562499999923
N2      -0.012890624999999907
N3      -0.012890624999999907
N4      -0.012890624999999907
N5      -0.012890624999999907
N6      -0.012890624999999907
```

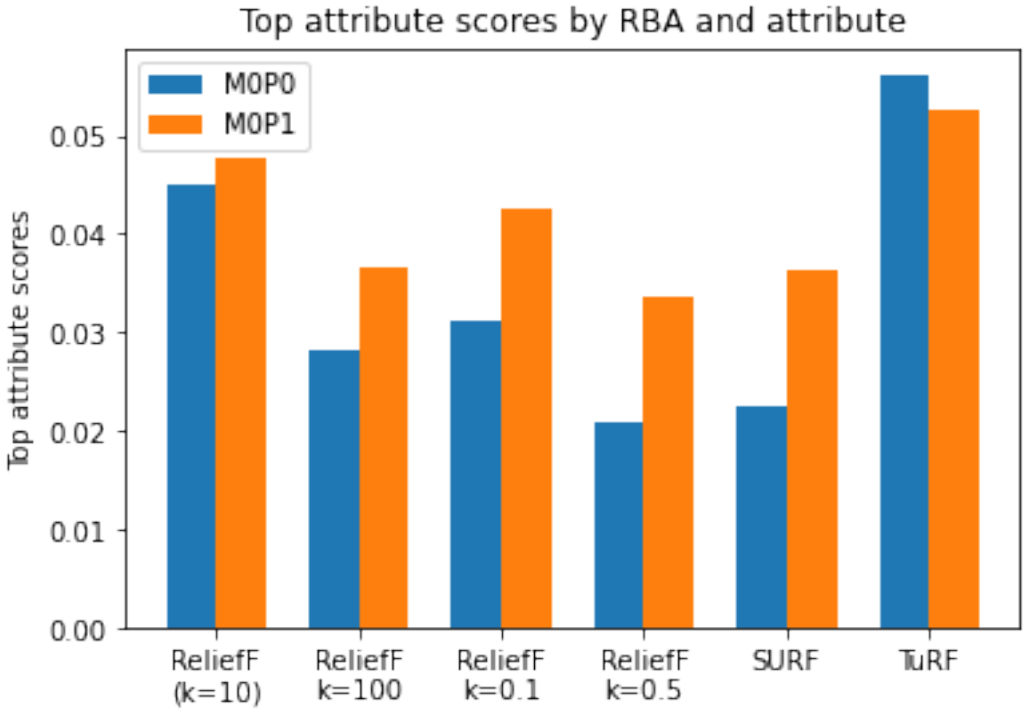
Σχήμα 4.4: Βάρη κορυφαίων χαρακτηριστικών ανά αλγόριθμο για σύνολο δεδομένων με διακριτά χαρακτηριστικά



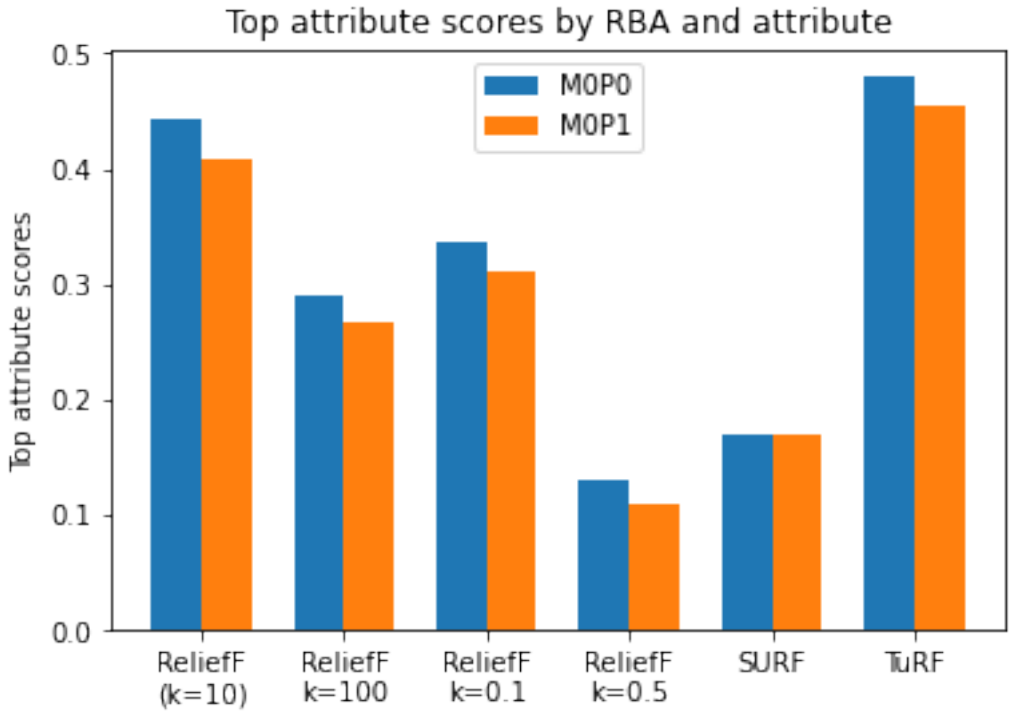
Σχήμα 4.5: Βάρη κορυφαίων χαρακτηριστικών ανά αλγόριθμο για σύνολο δεδομένων με μικτά (διακριτά και συνεχή) χαρακτηριστικά



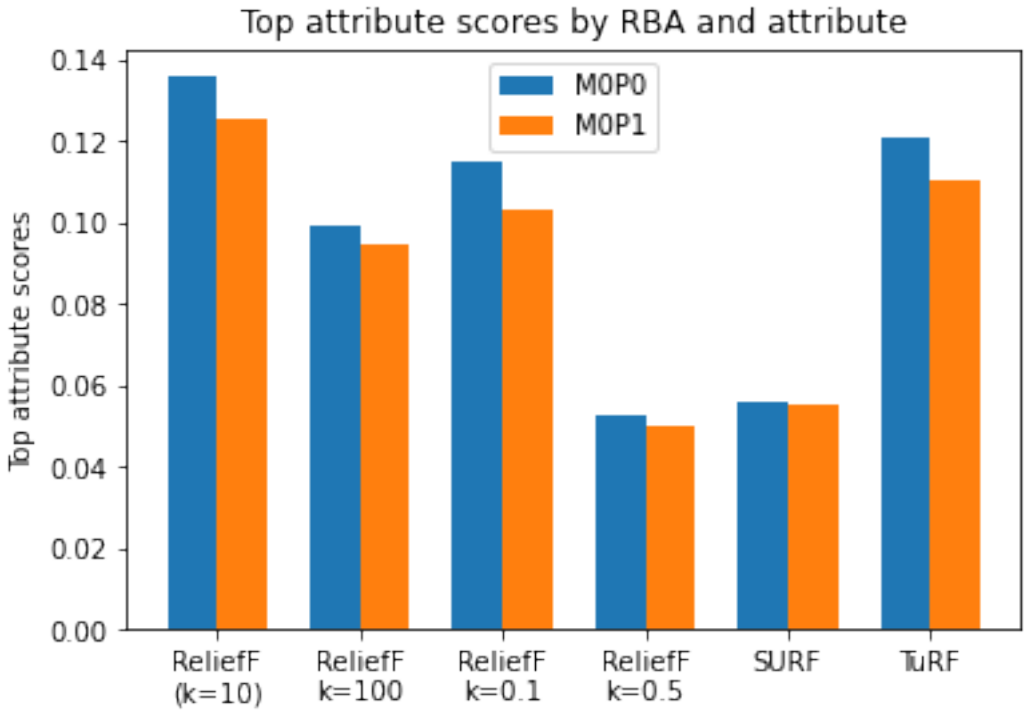
Σχήμα 4.6: Βάρη κορυφαίων χαρακτηριστικών ανά αλγόριθμο για σύνολο δεδομένων με συνεχή κλάση (παλινδρόμηση)



Σχήμα 4.7: Βάρη κορυφαίων χαρακτηριστικών ανά αλγόριθμο για σύνολο δεδομένων με πολλές (τρεις) κλάσεις



Σχήμα 4.8: Βάρη κορυφαίων χαρακτηριστικών ανά αλγόριθμο για σύνολο δεδομένων με απουσιάζουσες τιμές



Κεφάλαιο 5

Επίλογος

5.1 Σύνοψη και συμπεράσματα

Αυτή η εργασία αφορούσε τη μελέτη, την εκτέλεση και την αξιολόγηση αλγορίθμων επιλογής χαρακτηριστικών σε σύνολα δεδομένων που αντιστοιχούν σε προβλήματα της Βιοπληροφορικής. Συγκεκριμένα, (1) αξιολογήθηκε η ReBATE ως ένα ανοιχτού κώδικα, φιλικό προς τον χρήστη, και ευέλικτο προς τους τύπους δεδομένων πακέτο λογισμικού για την εφαρμογή μιας ποικιλίας από RBAs, (2) αξιολογήθηκαν στρατηγικές επέκτασης των RBAs για μια ποικιλία από τύπους συνόλων δεδομένων (που δεν έχουν κατά ανάγκη καθαρά δεδομένα ή διακριτά χαρακτηριστικά και κλάση), και (3) αναγνωρίστηκαν γνωστοί ή αναμενόμενοι λόγοι για διαφορές στην αποτελεσματικότητα και την απόδοση των RBAs.

Τα αποτελέσματα αυτής της μελέτης υποστηρίζουν τα ακόλουθα συμπεράσματα: (1) οι RBAs λειτουργούν υπό την ύπαρξη μοτίβων ετερογενούς συσχέτισης, (2) ο αριθμός των γειτόνων που χρησιμοποιείται στη βαθμολόγηση των RBAs είναι ένας κρίσιμος παράγοντας που εξαρτάται από το πρόβλημα ως προς την επιτυχία του αλγορίθμου, (3) οι υλοποιημένες επεκτάσεις τύπων δεδομένων της ReBATE ήταν επιτυχείς, ωστόσο παρατηρήθηκαν κάποιες απώλειες επίδοσης σε σύνολα δεδομένων με μικτά (διακριτά και συνεχή) χαρακτηριστικά, και (4) το κύριο μειονέκτημα του ReliefF είναι ότι ο χρήστης πρέπει να προσδιορίσει μια παράμετρο k η οποία τα αποτελέσματα δείχουν ότι μπορεί να επηρεάσει δραματικά την επιτυχία ανάλογα με το θόρυβο (δηλ. τη κληρονομικότητα σε γενετικά σύνολα δεδομένων), τον αριθμό των δειγμάτων εκπαίδευσης, το μέγεθος του χώρου χαρακτηριστικών, και/ή το πλήθος των απουσιάζουσων τιμών.

5.2 Μελλοντικές επεκτάσεις

Τα αποτελέσματα αυτής της εργασίας ενθαρρύνουν τη συνέχεια της μελέτης και της αξιολόγησης των RBAs. Προτάσεις για μελλοντική έρευνα αποτελούν: (1) η σύγκριση εναλλακτικών ή νέων στρατηγικών για το χειρισμό απουσιάζουσων τιμών, παλινδρόμησης, και μικτών τύπων χαρακτηριστικών, (2) η αξιολόγηση άλλων RBAs που υλοποιεί η ReBATE όπως οι SURF*, MultiSURF, MultiSURF*, (3) η εξέταση εναλλακτικών στρατηγικών για τη προσαρ-

μογή της επιλογής γειτόνων σε διαφορετικά προβλήματα, και (4) η ολοκλήρωση υποσχόμενων RBAs με επαναληπτικούς RBAs (π.χ. TuRF) για τη κλιμάκωση σε πολύ μεγάλους χώρους χαρακτηριστικών.

Βιβλιογραφία

[1] K. Kira, L. Rendell. The Feature Selection Problem: Traditional Methods and a New Algorithm. In *AAAI-92 Proceedings*.

[2] I. Kononenko, I. Simec, M. Robnik-Sikonja. Overcoming the myopia of inductive learning algorithms with RELIEFF. *Applied Intelligence*, 7(1):39-55, 1997.

[3] I. Kononenko. Estimating attributes: analysis and extensions of RELIEF. In *Mach Learn ECML-94*, 784:171–182, 1994.

[4] K. Kira, L. Rendell. A practical approach to feature selection. In *Proceedings of the ninth international workshop on machine learning*, 249-256, 1992.

[5] M. Robnik-Sikonja, I. Kononenko. Theoretical and Empirical Analysis of ReliefF and RreliefF. In *Machine Learning*, 53:23-69, 2003.

[6] S.J. Hong. Use of contextual information for feature ranking and discretization. In *IEEE Transactions on Knowledge and Data Engineering*, 9(5):718-730, 1997.

[7] J.H. Moore, B.C. White. Tuning ReliefF for Genome-Wide Genetic analysis. In *Lecture Notes in Computer Science*, 4447:166-175, 2007.

[8] C.S. Greene, N.M. Penrod, J. Kiralis, J.H. Moore. Spatially Uniform ReliefF (SURF) for computationally-efficient filtering of gene-gene interactions. In *BioData Mining*, 2(1):5, 2009.

[9] R.J. Urbanowicz, M. Meeker, W. La Cava, R.S. Olson, J.H. Moore. Relief-based feature selection: Introduction and review. In *Journal of Biomedical Informatics*, 85:189-203, 2018.

[10] R.J. Urbanowicz, R.S. Olson, P. Schmitt, M. Meeker, J.H. Moore. Benchmarking relief-based feature selection methods for bioinformatics data mining. In *Journal of Biomedical Informatics*, 85:168-188, 2018.

[11] M. Robnik-Sikonja, I. Kononenko. An adaptation of relief attribute estimation in regression. In *Machine Learning: Proceedings of the Fourteenth International Conference (ICML97)*, 296-304, 1997.

[12] I. Kononenko, M.R. Sikonja. Non-myopic feature quality evaluation with (r) relieff. In *Computational Methods for Feature Selection*, 169-191, 2008.

[13] Wikipedia Foundation, Inc. Relief (feature selection).

[14] J.H. Moore, F.W. Asselbergs, S.M. Williams. Bioinformatics challenges for genome-wide assosiation studies. In *Bioinformatics*, 26(4):445-455, 2010.

[15] A. Todorov. An Overview of the RELIEF Algorithm and Advancements. In *Statistical Approaches to Gene x Environment Interactions, MIT Press*, 95-116, 2016.

[16] B.A. McKinney, B.C. White, D.E. Grill, P.W. Li, R.B. Kennedy, G.A. Polland, A.L. Oberg. Reliefseq: a gene-wise adaptive-k nearest neighbor feature selection tool for finding gene-gene interactions and main effects in mrna-seq gene expression data. In *PloS one*, 8(12), e81527, 2013.

[17] B. Draper, C. Kaito, J. Bins. Iterative relief. In *Computer Vision and Pattern Recognition Workshop*, 6, 62-62, 2003.

[18] Y. Sun, J. Li. Iterative relief for feature weighting. In *Proceedings of the 23rd international conference on Machine learning ACM*, 913-029, 2006.

[19] R.J. Palma-Mendoza, D. Rodriguez, L. de-Marcos. Distributed ReliefF-based feature selection in Spark. In *Knowledge and Information Systems*, 57, 1-20, 2018.