ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ

ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

ΤΟΜΕΑΣ ΗΛΕΚΤΡΙΚΩΝ ΒΙΟΜΗΧΑΝΙΚΩΝ ΔΙΑΤΑΞΕΩΝ ΚΑΙ ΣΥΣΤΗΜΑΤΩΝ ΑΠΟΦΑΣΕΩΝ

Πρόβλεψη Αποτελεσμάτων Αγώνων Μπάσκετ με Χρήση Τεχνικών Μηχανικής Μάθησης

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

του

Σπυρίδωνος Ι. Αρμενιάκου

Επιβλέπων : Δημήτριος Ασκούνης
Καθηγητής Ε.Μ.Π.

Αθήνα, Οκτώβριος 2022

ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ
ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ
ΤΟΜΕΑΣ ΤΕΧΝΟΛΟΓΙΑΣ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΥΠΟΛΟΓΙΣΤΩΝ

Πρόβλεψη Αποτελεσμάτων Αγώνων Μπάσκετ με Χρήση Τεχνικών Μηχανικής Μάθησης

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

του

Σπυρίδωνος Ι. Αρμενιάκου

Επιβλέπων : Δημήτριος Ασκούνης
Καθηγητής Ε.Μ.Π.

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την 12η Οκτωβρίου 2022.

.....	.....	.....
Δημήτριος Ασκούνης	Ιωάννης Ψαρράς	Χρυσόστομος Δούκας
Καθηγητής Ε.Μ.Π.	Καθηγητής Ε.Μ.Π.	Καθηγητής Ε.Μ.Π.

Αθήνα, Οκτώβριος 2022

.....

Σπυρίδων Ι. Αρμενιάκος

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

Περίληψη

Το μπάσκετ αποτελεί ένα από τα πιο δημοφιλή αθλήματα παγκοσμίως, με αγώνες να λαμβάνουν χώρα κάθε μέρα σε όλο τον κόσμο. Το γεγονός αυτό βοηθάει στη δημοσιοποίηση πολλών δεδομένων και στατιστικών σχετικά με τους αγώνες και τους παίχτες, όπως είναι το τελικό σκορ, τα ποσοστά ευστοχίας, η θέαση κλπ. Η Μηχανική Μάθηση με τεράστια πρόοδο τα τελευταία χρόνια προσπαθεί να αναλύσει δεδομένα, να τα συσχετίσει και να δημιουργήσει προβλέψεις σχετικά με αυτά. Σκοπός της παρούσας διπλωματικής εργασίας είναι η εύρεση dataset με αγώνες μπάσκετ, η ανάλυσή του, η δημιουργία προβλέψεων με την χρήση Τεχνικών Μηχανικής Μάθησης και Βαθιάς Μάθησης και τέλος η σύγκριση των αποτελεσμάτων των διαφορετικών μεθόδων.

Λέξεις Κλειδιά: Ταξινόμηση, Επιλογή - Εξαγωγή Χαρακτηριστικών, Μηχανική Μάθηση, Ρύθμιση Υπερπαραμέτρων, NBA, Προβλέψεις

Abstract

Basketball is one of the most popular sports in the world, with matches taking place every day all over the world. This fact helps in releasing a lot of data and statistics about matches and players, such as final score, hit rate, viewership, etc. Machine Learning with huge progress in recent years tries to analyze data, correlate it and make predictions about with these. The purpose of this thesis is to find a dataset with basketball games, analyze it and finally create predictions using Machine Learning and Deep Learning techniques and compare the results of the different methods.

Keywords: Classification, Features Selection - Feature Engineering, Machine learning, Hyperparameter Tuning, NBA, Prediction, Sports analytics

Στην Αλέκα

Ευχαριστίες

Με την εκπόνηση της παρούσας διπλωματικής εργασίας ολοκληρώνεται ένας πολύ σημαντικός κύκλος σπουδών και ζωής στη Σχολή Ηλεκτρολόγων Μηχανικών & Μηχανικών Υπολογιστών. Είμαι πολύ χαρούμενος και θα ήθελα να ευχαριστήσω όλους τους ανθρώπους που με βοήθησαν, ο καθένας με τον τρόπο του, να φτάσω ως εδώ.

Αρχικά, θα ήθελα να ευχαριστήσω τον κ. Δημήτριο Ασκούνη, Καθηγητή Ε.Μ.Π., που ήταν ο επιβλέπων της εργασίας και μου έδωσε την ευκαιρία να καταπιαστώ με το συγκεκριμένο θέμα, που τόσο πολύ επιθυμούσα. Συγκεκριμένα μπόρεσα να ασχοληθώ με το άθλημα του μπάσκετ που αγαπώ από μικρό παιδί, αυτή τη φορά από μια εντελώς διαφορετική θέση και εξελίσσοντας τις γνώσεις μου. Στη συνέχεια, θα ήθελα να ευχαριστήσω τους κυρίους Ιωάννη Ψαρρρά, Καθηγητή Ε.Μ.Π., και Χρυσόστομο Δούκα, Αναπληρωτή Καθηγητή Ε.Μ.Π., για την τιμή που μου έκαναν να αποτελέσουν την εξεταστική επιτροπή.

Θα ήθελα ακόμη να ευχαριστήσω τους συμφοιτητές και φίλους που απέκτησα στα χρόνια φοίτησής μου, με τους οποίους συνεργαστήκαμε σε πολλές εργασίες, διαβάσαμε πολλές ώρες μαζί και με βοήθησαν ως το τέλος της μακράς αυτής διαδρομής.

Τέλος, οφείλω ένα τεράστιο ευχαριστώ στους φίλους μου οι οποίοι είναι κοντά μου εδώ και 20 περίπου χρόνια, στη σύντροφό μου που με στηρίζει αδιάκοπα, στην οικογένειά μου που μου έδωσε όλα τα εφόδια να επιτύχω τα όνειρά μου. Ειδικότερα, ευχαριστώ τον αδέρφο μου που πάντα αποτελεί για εμένα πρότυπο.

Η διπλωματική αυτή εργασία είναι αφιερωμένη στον πιο αγαπημένο, δυνατό άθρωπο που έχω γνωρίσει, τη μητέρα μου, η οποία «έφυγε» από τη ζωή τον περασμένο χειμώνα.

Πίνακας περιεχομένων

1 *Εισαγωγή*

1.1 Τίτλος

Το ανθρώπινο είδος, εδώ και χιλιάδες χρόνια, προσπαθεί ασταμάτητα να ανακαλύψει τι πρόκειται να συμβεί στο μέλλον του, να προβλέψει αυτά που αγνοεί αλλά και τα επακόλουθα των πράξεων και των αποφάσεων του. Θα μπορούσε να ειπωθεί ότι το μέλλον έχει ιστορία[1]. Κάνοντας μια σύντομη αναδρομή, οι άνθρωποι προσπαθούσαν πάντα να μάθουν περισσότερα για τη μορφή των πραγμάτων που θα ακολουθήσουν. Αυτές οι προσπάθειες, ενώ στόχευαν στον ίδιο σκοπό, διέφεραν στο χρόνο και στο χώρο με πολλούς σημαντικούς τρόπους, με πιο προφανή τη μεθοδολογία, δηλαδή τον τρόπο με τον οποίο έγιναν και ερμηνεύτηκαν οι προβλέψεις. Από τους πρώτους πολιτισμούς, η πιο σημαντική διάκριση σε αυτή την προσπάθεια ήταν μεταξύ ατόμων που έχουν ένα εγγενές χάρισμα ή ικανότητα να προβλέπουν το μέλλον, και συστημάτων που παρέχουν κανόνες για τον υπολογισμό των μελλοντικών συμβάντων.

Με την πάροδο του χρόνου η τάση του ανθρώπου να μελετά τα φαινόμενα που συμβαίνουν γύρω του, να καταγράφει στοιχεία σχετικά με αυτά, να προσπαθεί βάσει των γνώσεων και των εμπειριών του να ερμηνεύσει τον κόσμο ενισχύθηκε. Αυτό τον οδήγησε στην εξέλιξη των μέσων και των μεθόδων συλλογής των δεδομένων και εκμετάλλευσής τους στην πρόβλεψη του μέλλοντος.

Από τον περασμένο αιώνα η τεχνολογία και οι εξελίξεις στον τομέα της πληροφορικής (προβλεφθείσες, τουλάχιστον σε κάποιο βαθμό, από το νόμο του Moore) άρχισαν να παρέχουν ισχυρότερα εργαλεία και συστήματα για την προσπάθεια προβλέψεων[2]. Παρατηρείται έκτοτε μια ολοένα και αυξανόμενη εφαρμογή, έως και εξάρτηση, του ανθρώπου από αυτά τα υπολογιστικά συστήματα. Οι περισσότερες πτυχές της ανθρώπινης καθημερινότητας είναι πλέον άρρηκτα συνδεδεμένες με την τεχνολογία, το ίντερνετ, τους υπολογιστές και τις λοιπές «έξυπνες» συσκεύες.

Το πάντρεμα του αθλητισμού και των νεότερων τεχνολογιών είναι ένα τέτοιο χαρακτηριστικό παράδειγμα[3]. Η επιστήμη των δεδομένων, τα Analytics, και οι εφαρμογές τους στον αθλητισμό είναι πολύ ανεπτυγμένα σε σύγκριση με πολλούς άλλους κλάδους. Οι αθλητικοί οργανισμοί βρίσκονται στην πρώτη γραμμή της συλλογής δεδομένων εδώ και αρκετά χρόνια. Οι ομάδες έχουν τα εργαλεία και τις υποδομές να λάβουν όσο το δυνατόν περισσότερες πληροφορίες για να αποκτήσουν ανταγωνιστικό πλεονέκτημα έναντι ενός αντιπάλου. Οι αθλητές και οι προπονητές είναι πιο δεκτικοί στη χρήση δεδομένων για τη βελτίωση της απόδοσής τους. Τα προβλήματα είναι καλά καθορισμένα στον αθλητισμό και τα δεδομένα χρησιμεύουν ως πρόσθετη πληροφορία στη γνώση και τη διαίσθηση των ειδικών για την επίλυση αυτών των προβλημάτων. Ο μετασχηματισμός της αθλητικής βιομηχανίας συνεχίζεται, καθώς οι σύλλογοι, τα πρωταθλήματα, οι ραδιοτηλεοπτικοί φορείς και οι επαγγελματίες παίκτες βλέπουν όλο και περισσότερο την αξία των προηγμένων αναλυτικών

στοιχείων για τον εντοπισμό μετρήσεων και μοτίβων που μπορεί να μην είναι προφανή στο μάτι του παραδοσιακού σκάουτερ ή μάνατζερ.

Οι εξελίξεις στην υπολογιστική ισχύ, την τεχνολογία cloud και τεχνολογίες όπως η Όραση Υπολογιστών, η Μηχανική Μάθηση, η προηγμένη ασύρματη συνδεσιμότητα και οι φορητοί αισθητήρες μεταμορφώνουν τον τρόπο με τον οποίο οι ομάδες προπονούνται, ανταγωνίζονται και διαχειρίζονται την πορεία τους. Οι εταιρείες μεταμορφώνουν την αθλητική σκηνή με ακριβείς μετρήσεις που αφορούν στην απόδοση, την υγεία και την ασφάλεια των παικτών, επιτρέποντας στους προπονητές και το ιατρικό επιτελείο να λαμβάνουν ακριβείς αποφάσεις. Ωστόσο, μια τεράστια αύξηση του όγκου δεδομένων δεν μπορεί να είναι χρήσιμη χωρίς τη σωστή ερμηνεία.

Πλέον, η συλλογή των δεδομένων, η ανάλυση και η επεξεργασία τους με προβλεπτικά μοντέλα Μηχανικής Μάθησης έχει διευρυνθεί σε τέτοιο βαθμό που μεγάλοι αθλητικοί οργανισμοί από διαφορετικά αθλήματα βασίζουν τις καίριες αποφάσεις τους στις επιστημονικές αυτές μεθόδους. Δεν περιορίζονται μόνο στην πρόβλεψη ενός θετικού ή μη αποτελέσματος στον επόμενα αγώνα που θα δώσουν, ούτε στην πρόβλεψη ενός τραυματισμού για κάποιον αθλητή, όπως αναφέρθηκε παραπάνω.

Η πρόβλεψη των δυνατών σημείων, των αδυναμιών και των τάσεων των αντίπαλων ομάδων μπορεί να βοηθήσει στην ανάπτυξη της σωστής στρατηγικής για οποιαδήποτε κατάσταση παιχνιδιού. Η επιστήμη των δεδομένων στον αθλητισμό μπορεί να συνεισφέρει στη μεγιστοποίηση των νικών, προσφέροντας αξιόπιστες πληροφορίες σχετικά με το τι πιθανότατα θα συμβεί μετά από κάθε απόφαση σε έναν αγώνα για την εξαγωγή της καλύτερης απόδοσης. Πρόσφατα, πολλοί αθλητικοί οργανισμοί έχουν επενδύσει σε αθλητικές αναλύσεις των οποίων τα αποτελέσματα είναι πολύ ενθαρρυντικά. Η κύρια στόχευση είναι στην κατασκευή μοντέλων που βασίζονται σε τεχνολογικά εργαλεία για τη διαχείριση της κόπωσης των παικτών, των τραυματισμών, του scouting, της ανάλυσης πριν από τον αγώνα, της ανάλυσης μετά τον αγώνα και της στρατολόγησης προπονητών.

Αυτή είναι μόνο η κορυφή του παγόβουνου. Η εξάρτηση από τα αθλητικά αναλυτικά στοιχεία θα πολλαπλασιαστεί με την εμφάνιση προηγμένων συσκευών παρακολούθησης και ρύθμισης συλλογής δεδομένων. Μερικοί από τους αναδυόμενους τομείς περιλαμβάνουν τη βιομηχανία φορητών συσκευών, την ιατρική βιομηχανία, τις ασφάλειες, τα στοιχήματα και τη βιομηχανία τυχερών παιχνιδιών. Είναι στιγμή οι αθλητικοί οργανισμοί να επενδύσουν σε αθλητικές αναλύσεις ή να αναζητήσουν υποστήριξη από προηγμένες εταιρείες αναλυτικών στοιχείων για να παραμείνουν ανταγωνσιτικοί στη σύγχρονη εποχή.

1.2 Αντικείμενο διπλωματικής

Το μπάσκετ αποτελείται από δύο διαφορετικά αποτελέσματα, τη νίκη της γηπεδούχου ομάδας και τη νίκη της φιλοξενούμενης ομάδας. Στην παρούσα εργασία θα δημιουργηθούν μοντέλα που προβλέπουν το τελικό αποτέλεσμα αγώνων μπάσκετ από το NBA (National Basketball Association) των Ηνωμένων Πολιτειών της Αμερικής. Ο στόχος της εργασίας είναι η ανάπτυξη μοντέλων Μηχανικής Μάθησης που θα επιτυγχάνουν όσο το δυνατόν υψηλότερη ορθότητα (accuracy) στις προβλέψεις τους.

Η δημιουργία αυτών των μοντέλων θα πραγματοποιηθεί με βασικούς αλγορίθμους Μηχανικής Μάθησης. Τα δεδομένα που χρειάστηκαν για το σκοπό αυτό, βρέθηκαν στο διαδίκτυο, καλύπτουν ένα χρονικό εύρος περίπου 20 ετών και περιέχουν σημαντική πληροφορία. Τα περισσσότερα από τα δεδομένα αυτά χρησίμευσαν επίσης στην εξαγωγή νέων χαρακτηριστικών τα οποία ήταν δυσεύρετα έως και ανύπαρκτα. Στη συνέχεια, επιλέχθηκαν τα σημαντικότερα από αυτά με γνωστές μεθόδους που θα αναφερθούν στη συνέχεια, ώστε να εκπαιδευτούν τα διάφορα μοντέλα. Ακολουθήθηκε επίσης η τεχνική της Αναζήτησης Πλέγματος (Grid Search) για τη ρύθμιση των υπερπαραμέτρων των μοντέλων, βήμα που είναι πολύ σημαντικό για την επίτευξη της καλύτερης δυνατής ορθότητάς τους. Συνολικά θα αναπτυχθούν 10 ξεχωριστά μοντέλα τα οποία στη συνέχεια θα συγκριθούν τόσο μεταξύ τους όσο και με αντίστοιχα από τη διεθνή βιβλιογραφία.

1.3 Οργάνωση κειμένου

Στο πρώτο κεφάλαιο δόθηκε μία σύντομη εισαγωγή και περιγραφή του προβλήματος, που πραγματεύεται η παρούσα εργασία, ενώ ορίστηκαν και οι στόχοι της.

Στο δεύτερο κεφάλαιο παρουσιάζεται το θεωρητικό υπόβαθρο των αλγορίθμων Μηχανικής Μάθησης και των μεθόδων Επιλογής Δεδομένων που εφαρμόζονται στην εργασία. Πιο συγκεκριμένα, κάθε αλγόριθμος αναλύεται συνοπτικά και περιγράφονται οι παράμετροί του που χρησιμοποιήθηκαν σε αυτή την εργασία

Στο τρίτο κεφάλαιο περιγράφεται και παρουσιάζεται το dataset από το οποίο προέκυψαν τα χαρακτηριστικά με τα οποία θα εκπαιδευτούν στη συνέχεια τα μοντέλα. Επιπλέον, παρουσιάζονται και οι συσχετίσεις μεταξύ των χαρακτηριστικών. Γίνεται ακόμη μια σύντομη περιγραφή του πρωταθλήματος του NBA.

Στο τέταρτο κεφάλαιο αναλύονται οι διαφορετικές υλοποιήσεις και τα αποτελέσματά τους. Για κάθε υλοποίηση παρατίθενται τα reports των training και test set, τα features που αποτελούν κάθε φορά το σύνολο εκπαίδευσης και με τη χρήση ποιας μεθόδου αυτά προέκυψαν. Επιπλέον, παρουσιάζεται διαγραμματικά η ορθότητα κάθε μοντέλου ανά σεζόν και συνολικά για τις 4 σεζόν του test set. Τέλος, γίνεται μία σύγκριση μεταξύ των διαφορετικών υλοποιήσεων που αναπτύχθηκαν για την παρούσα διπλωματική, καθώς επίσης και μία δεύτερη σύγκριση με τα αποτελέσματα κάποιων άλλων ερευνητικών εργασιών της διεθνούς βιβλιογραφίας.

Στο πέμπτο, και τελευταίο κεφάλαιο, παρουσιάζονται τα κυριότερα αποτελέσματα και οι παρατηρήσεις που προέκυψαν από την παρούσα εργασία, ενώ γίνεται αναφορά σε πιθανές μελλοντικές επεκτάσεις.

2 Θεωρία

2.1 Μηχανική Μάθηση – Θεωρητικό υπόβαθρο

Ο όρος Μηχανική Μάθηση επινοήθηκε για πρώτη φορά τη δεκαετία του 1950, όταν ο πρωτοπόρος της Τεχνητής Νοημοσύνης Άρθουρ Σάμουελ κατασκεύασε το πρώτο σύστημα αυτοεκπαίδευσης για να παίζει το επιτραπέζιο παιχνίδι ντάμα. Παρατήρησε ότι όσο περισσότερο έπαιζε το σύστημα, τόσο καλύτερο γινόταν στο συγκεκριμένο παιχνίδι. Σύμφωνα με τον Άρθουρ Σάμουελ, οι αλγόριθμοι Μηχανικής Μάθησης επιτρέπουν στους υπολογιστές να μαθαίνουν από δεδομένα, ακόμη και να βελτιώνονται, χωρίς να είναι ρητά προγραμματισμένοι για αυτό.

Η Μηχανική Μάθηση (ML) είναι μια κατηγορία αλγορίθμων που επιτρέπει στις εφαρμογές λογισμικού να γίνονται πιο ακριβείς στην πρόβλεψη των αποτελεσμάτων χωρίς να προγραμματίζονται ξεκάθαρα. Η βασική προϋπόθεση της Μηχανικής Μάθησης είναι η κατασκευή αλγορίθμων που μπορούν να λαμβάνουν δεδομένα εισόδου και να χρησιμοποιούν στατιστική ανάλυση για την πρόβλεψη ενός αποτελέσματος. Παράλληλα, ενημερώνονται τα αποτελέσματα καθώς γίνονται διαθέσιμα συνεχώς νέα δεδομένα.

Τροφοδοτούμενη από την πρόοδο στις στατιστικές μεθόδους και την επιστήμη των υπολογιστών, καθώς και από τα καλύτερα σύνολα δεδομένων και την ανάπτυξη των νευρωνικών δικτύων, η μηχανική μάθηση έχει πραγματικά απογειωθεί τα τελευταία χρόνια. Σήμερα, είτε το καταλαβαίνουμε είτε όχι, η μηχανική μάθηση είναι παντού – αυτοματοποιημένη μετάφραση, αναγνώριση εικόνας, τεχνολογία φωνητικής αναζήτησης, αυτοοδηγούμενα αυτοκίνητα και όχι μόνο.

Οι υλοποιήσεις Μηχανικής Μάθησης είναι αρκετές, ανάλογα με τη φύση του «σήματος» ή της «απόκρισης» μάθησης που διατίθεται σε ένα σύστημα εκμάθησης. Ωστόσο στην παρούσα διπλωματική εργασία εφαρμόζονται αποκλειστικά τεχνικές Επιβλεπόμενης Μάθησης(Supervised Learning): οι αλγόριθμοι Επιβλεπόμενης Μάθησης και τα επιβλεπόμενα μοντέλα μάθησης κάνουν προβλέψεις με βάση τα δεδομένα εκπαίδευσης που συνοδεύονται από μια ετικέτα. Κάθε δείγμα εκπαίδευσης περιλαμβάνει μια είσοδο και μια επιθυμητή έξοδο. Ένας επιβλεπόμενος αλγόριθμος μάθησης αναλύει αυτά τα δείγματα δεδομένων και εξάγει ένα συμπέρασμα – βασικά, μια έμπειρη εικασία κατά τον προσδιορισμό των ετικετών για μη ορατά δεδομένα. Αυτή είναι η πιο κοινή και δημοφιλής προσέγγιση στη Μηχανική Μάθηση. Ο λόγος που ορίζεται ως «επιβλεπόμενο» είναι επειδή αυτά τα μοντέλα πρέπει να τροφοδοτούνται με μη αυτόματο τρόπο δειγματοληψίας δεδομένων για να εκπαιδευτούν. Τα δεδομένα επισημαίνονται με τις ετικέτες για να ενημερώνουν το μηχάνημα με ποια μοτίβα (παρόμοιες λέξεις και εικόνες, κατηγορίες δεδομένων κ.λπ.) θα πρέπει να αναζητά και να αναγνωρίζει τις συνδέσεις.

Η διαδικασία Μηχανικής Μάθησης περιλαμβάνει τρία βήματα:

1. Τροφοδότηση με δεδομένα εισόδου εκπαίδευσης ενός μοντέλου. Στην περίπτωσή μας, αυτό θα μπορούσε να είναι τα βασικά στατιστικά στοιχεία των ομάδων στα παιχνίδια που έχουν αγωνιστεί τα τελευταία χρόνια.
2. Επισημαίνονται τα δεδομένα εκπαίδευσης με την επιθυμητή έξοδο.
3. Τεστάρεται το μοντέλο τροφοδοτώντας το με δοκιμαστικά(ή μη ορατά) δεδομένα. Οι αλγόριθμοι εκπαιδεύονται να συσχετίζουν διανύσματα χαρακτηριστικών με ετικέτες που βασίζονται σε επισημασμένα δείγματα και στη συνέχεια μαθαίνουν να κάνουν προβλέψεις κατά την επεξεργασία μη ορατών δεδομένων.

Η παρούσα εργασία χρησιμοποιεί δεδομένα από αγώνες του Αμερικανικού Πρωταθλήματος Μπάσκετ, το NBA[4], με σκοπό να προβλέψει το τελικό αποτέλεσμα επόμενων αγώνων της ίδιας διοργάνωσης με τη μεγαλύτερη δυνατή ορθότητα. Για το σκοπό αυτό χρησιμοποιήθηκαν οι ακόλουθοι αλγόριθμοι: Gaussian Naive Bayes, k-Nearest Neighbors, Random Forest, Support Vector Machines, Multilayer Perceptron, XGBoost, Stacking και Voting Classifiers, καθώς και ένα Recurrent Neural Network με LSTM (RNN with LSTM), οι οποίοι παρουσιάζονται συνοπτικά παρακάτω.

● Gaussian Naïve Bayes (GNB)

Ο Naïve Bayes[5] είναι ένας πιθανοτικός αλγόριθμος μηχανικής μάθησης ο οποίος μπορεί να χρησιμοποιηθεί σε αρκετά προβλήματα ταξινόμησης και βασίζεται στο θεώρημα Bayes. Ο όρος Naïve ή αφελής, χρησιμοποείται καθώς ο αλγόριθμος ενσωματώνει στα μοντέλα του χαρακτηριστικά που είναι ανεξάρτητα μεταξύ τους, γεγονός που είναι δύσκολο να ισχύει μεταξύ των χαρακτηριστικών ένος dataset όπως το δικό μας.

Το θεώρημα Bayes υπολογίζει την πιθανότητα να συμβεί ένα ενδεχόμενο A υπό τη συνθήκη ότι ένα άλλο ενδεχόμενο B έχει ήδη συμβεί. Μαθηματικά, η υπό συνθήκη πιθανότητα του A δεδομένου του B δίνεται από τον παρακάτω τύπο :

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

όπου P(B|A) είναι η πιθανότητα να πραγματοποιηθεί το ενδεχόμενο B με την υπόθεση ότι ισχύει το A.

Αντίστοιχα, με δεδομένα μιας μεταβλητής κατηγορίας (κλάσης) y και ένα εξαρτώμενο διάνυσμα χαρακτηριστικών x_1 μέχρι x_n , σύμφωνα με το θεώρημα του Bayes θα ισχύει :

$$P(y|x_1, \dots, x_n) = \frac{P(y)P(x_1, \dots, x_n|y)}{P(x_1, \dots, x_n)}$$

Ισχύει ότι $P(y)P(y) = \prod_{i=1}^{n} P(x_i|y, x_1, \dots, x_{i-1}, x_{i+1}, \dots, x_n)$ και κάνουμε την αφελή υπόθεση ότι το χαρακτηριστικό x_i για κάθε i εξαρτάται μόνο από την κλάση y και όχι από οποιοδήποτε άλλο χαρακτηριστικό

$$P(y, x_1, \dots, x_{i-1}, x_{i+1}, \dots, x_n) = P(x_i|y)$$

αυτό οδηγεί στην απλοποίηση :

$$P(y|x_1, \dots, x_n) = \frac{P(y)\prod_{i=1}^{n} P(x_i|y)}{P(x_1, \dots, x_n)}$$

Με δεδομένη είσοδο, το $P(x_1, \dots, x_n)$ είναι σταθερό. Συνεπώς μπορούμε να χρησιμοποιήσουμε τον ακόλουθο κανόνα ταξινόμησης :

$$P(y|x_1, \dots, x_n) \propto P(y) \prod_{i=1}^{n} P(x_i|y)$$

$$\Downarrow$$

$$\hat{y} = arg\ arg\ P(y) \prod_{i=1}^{n} P(x_i|y)$$

Το $P(y)$ είναι η υπόθεσή μας και ισούται με τη σχετική συχνότητα της κλάσης y στο training set. Το $P(x_i|y)$ είναι η πιθανοφάνεια, δηλαδή η πιθανότητα του δείγματος, με δεδομένη την υπόθεσή μας και μπορεί επίσης να υπολογιστεί απλά από το training set. Οι διάφοροι Naive Bayes classifiers διαφοροποιούνται κυρίως από τις υποθέσεις που κάνουν ως προς την κατανομή $P(x_i|y)$. Η κλάση $\hat{y}$ που ανατίθεται σε ένα νέο δείγμα είναι αυτή που μεγιστοποιεί το δεξί μέλος της σχέσης.

Ο Gaussian Naïve Bayes είναι μια γρήγορη και απλή τεχνική ταξινόμησης η οποία μπορεί να δώσει αρκετά ικανοποιητικά αποτελέσματα με καλά επίπεδα ακρίβειας.

● k – Nearest Neighbors (kNN)

Ο kNN[6] είναι ένας μη παραμετρικός ταξινομητής βασισμένος σε παραδείγματα (instance-based). Η αρχή λειτουργίας του είναι πολύ απλή. Για ένα νέο δείγμα προς ταξινόμηση, πρώτα υπολογίζονται οι k πλησιέστεροι γείτονές του (στον ν-διάστατο χώρο των χαρακτηριστικών εισόδου) με βάση κάποια συνάρτηση απόστασης, συνήθως ευκλείδεια :

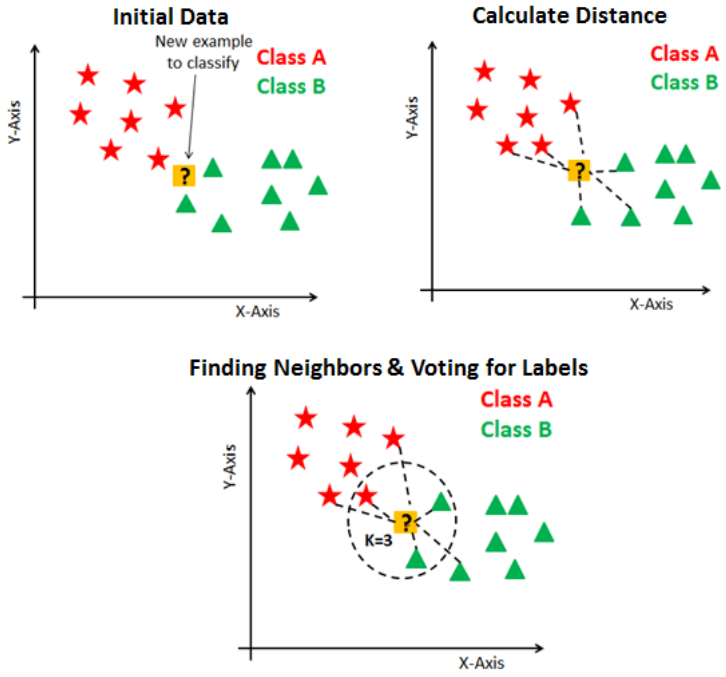
$$d(x, x') = \sqrt{(x_1 - x_1')^2 + (x_2 - x_2')^2 + \cdots + (x_n - x_n')^2}$$

Η κλάση του νέου δείγματος θα είναι η κλάση της πλειοψηφίας των k γειτόνων (επιλέγεται k περιττό γενικά), είτε απλά υπολογισμένη (άθροισμα) είτε (αντίστροφα) ζυγισμένη με βάση την απόσταση του κάθε γείτονα.
Ο kNN δεν έχει πρακτικά φάση εκπαίδευσης. Ωστόσο, για να ταξινομηθεί ένα νέο δείγμα στην φάση test, πρέπει να συγκριθεί την απόστασή του με κάθε δείγμα του train set. Αυτό σημαίνει ότι για την ταξινόμηση είναι απαραίτητα όλα τα δείγματα εκπαίδευσης (εξού και η ονομασία «instance-based», ενώ στον Naive Bayes χρειάζονται μόνο οι παράμετροι μ και σ^2). Αυτό σημαίνει ότι ο kNN είναι πιο απαιτητικός και σε χώρο (αποθήκευση όλων των δειγμάτων) και σε χρόνο (υπολογισμός όλων των αποστάσεων για κάθε νέο δείγμα).

<u>Υπερπαράμετρος k</u>

Το k της γειτονιάς του kNN είναι μια υπερπαράμετρος του ταξινομητή. Μια άλλη υπερπαράμετρος για παράδειγμα, είναι η συνάρτηση της απόστασης. Οι υπερπαράμετροι είναι επιλογές που γίνονται από τον σχεδιαστή του συστήματος και δεν είναι εφικτό να γνωρίζει κανείς τις βέλτιστες τιμές τους αν πρώτα δεν αξιολογηθούν εμπειρικά σε δεδομένα. Στην περίπτωση του kNN, το k ελέγχει το trade-off μεταξύ απόκλισης και διακύμνανσης. Έαν τεθεί μικρό k, π.χ. k=1, προκύπτει ένας ταξινομητής με υψηλή

διακύμανση και χαμηλή απόκλιση. Ο ταξινομητής τείνει να αγνοεί τη συνολική κατανομή και αποφασίζει μόνο από το κοντινότερο δείγμα. Στην περίπτωση k=1 το σύνορο απόφασης (decision boundary) περνά από τις μεσοκάθετους γειτονικών δειγμάτων διαφορετικής κλάσης. Αν τεθεί μεγαλύτερο k, δημιουργείται ένας ταξινομητής με χαμηλότερη διακύμανση και υψηλότερη απόκλιση. Θα ταξινομίσει λάθος περισσότερα αποκλίνοντα δείγματα (outliers) αλλά θα σέβεται περισσότερο τη συνολική κατανομή.



Οπτική αποτύπωση του αλγορίθμου kNN.

Source:https://www.datacamp.com/tutorial/k-nearest-neighbor-classification-scikit-learn

● Random Forest

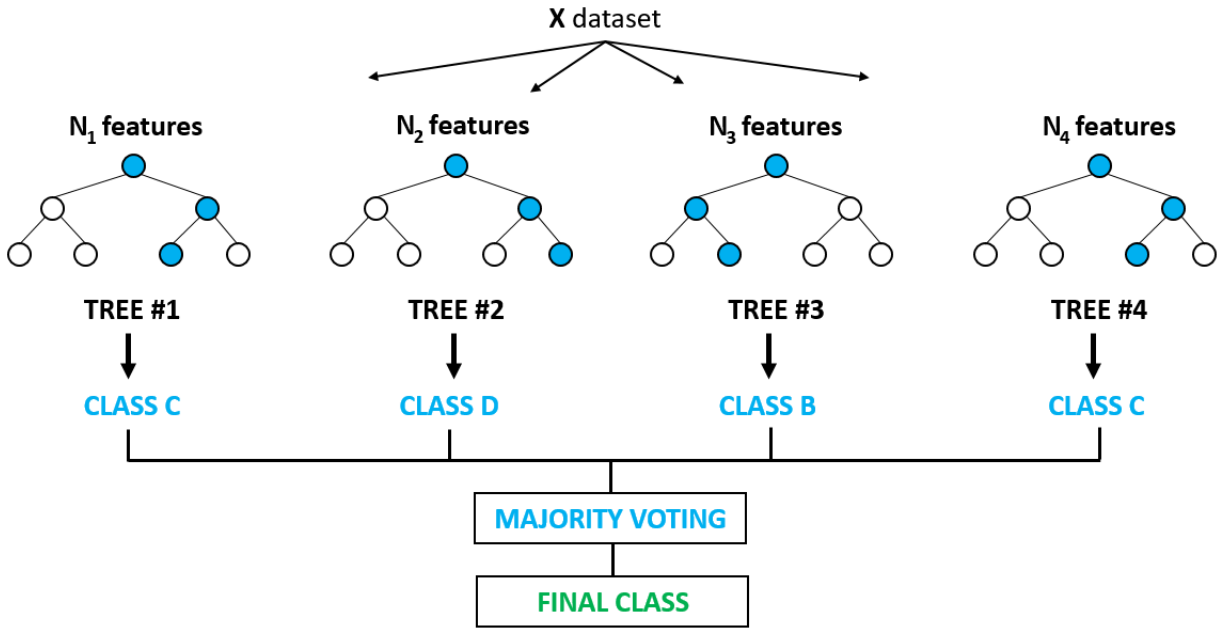
Ο Random Forest είναι ένας ισχυρός αλγόριθμος μηχανικής μάθησης που μπορεί να χρησιμοποιηθεί για μια ποικιλία τόσο Classification όσο και Regression εργασιών. Είναι μια μέθοδος συνόλου, που σημαίνει ότι ένα Random Forest μοντέλο αποτελείται από έναν αριθμό δέντρων απόφασης, που ονομάζονται εκτιμητές, και το καθένα παράγει τις δικές του προβλέψεις. Το μοντέλο συνδυάζει τις προβλέψεις των εκτιμητών για να παράγει μια πιο ακριβή πρόβλεψη.

Οι τυπικοί ταξινομητές δέντρων απόφασης έχουν το μειονέκτημα ότι είναι επιρρεπείς στην υπερβολική προσαρμογή στο σετ εκπαίδευσης. Ο σχεδιασμός συνόλου του Random Forest επιτρέπει στον ταξινομητή να αντισταθμίσει αυτό το γεγονός και να εφαρμοστεί καλά σε αόρατα δεδομένα, συμπεριλαμβανομένων δεδομένων με τιμές που λείπουν. Τα Random Forest μοντέλα είναι επίσης χρήσιμα στο χειρισμό μεγάλων συνόλων δεδομένων με υψηλή διαστατικότητα και ετερογενείς τύπους χαρακτηριστικών (για παράδειγμα, εάν μια στήλη είναι κατηγορική και μια άλλη αριθμητική). Ο Random Forest λειτουργεί πολύ καλά σε προβλήματα ταξινόμησης, αλλά η αποδοτικότητά του μειώνεται στα προβλήματα παλινδρόμησης.

Ο Random Forest μπορεί να παρομοιαστεί με ένα μαύρο κουτί : σε αντίθεση με ορισμένους πιο παραδοσιακούς αλγόριθμους μηχανικής μάθησης, είναι δύσκολο να

κοιτάξει κανείς μέσα σε έναν ταξινομητή Random Forest και να κατανοήσει το σκεπτικό πίσω από τις αποφάσεις του. Επιπλέον, μπορεί να είναι αργός στην εκπαίδευση και την εκτέλεση και να παράγει μεγάλα μεγέθη αρχείων.

Επειδή τα Random Forest μοντέλα είναι εξαιρετικά ισχυρά, εύχρηστα, καλά σε ετερογενείς τύπους δεδομένων και έχουν λίγες υπερπαραμέτρους, επιτρέπουν μια γρήγορη επισκόπηση του είδους της ακρίβειας που μπορεί εύλογα να επιτευχθεί σε ένα πρόβλημα, ακόμα κι αν η τελική λύση δεν περιλαμβάνει ένα τυχαίο δάσος.



Ένα Random Forest μοντέλο με 4 δέντρα

Source:https://medium.com/@ar.ingenious/applying-random-forest-classification-machine-learning-algorithm-from-scratch-with-real-24ff198a1c57

Θα αναφερθούν παρακάτω οι παράμετροι που τροποποιήθηκαν κατά τη δημιουργία του μοντέλου μας :

- ***n estimators***
  Η παράμετρος που ορίζει το πλήθος των Δέντρων Αποφάσεων, που θα λάβει υπόψιν του ο αλγόριθμος.

- ***max depth***
  Η παράμετρος που ορίζει το μέγιστο βάθος κάθε δέντρου απόφασης στο δάσος μας. Μεγαλύτερο βάθος συνεπάγεται και μεγαλύτερο όγκο πληροφορίας για την εκπαίδευση του μοντέλου.

- ***min samples split***
  Η παράμετρος που ορίζει το ελάχιστο πλήθος δειγμάτων που απαιτούνται για τον διαχωρισμό ενός εσωτερικού κόμβου στο Δέντρο Αποφάσεων.

- ***min samples leaf***

Η παράμετρος που ορίζει το ελάχιστο πλήθος δειγμάτων που οφείλουν να υπάρχουν σε έναν κόμβο-φύλλο.

- ***max features***

  Η παράμετρος που ορίζει το πλήθος των χαρακτηριστικών τα οποία θα ληφθούν υπόψη στον καλύτερο δυνατό διαχωρισμό που θα επιλέξει το κάθε Δέντρο Απόφασης.

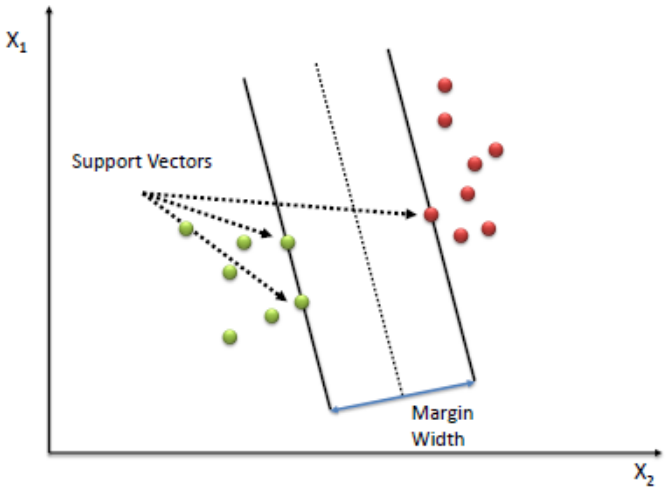
- ***bootstrap***

  Αν η παράμετρος τεθεί στην τιμή False, τότε κάθε Δέντρο Απόφασης θα χρησιμοποιήσει όλο το σύνολο δεδομένων εκπαίδευσης για το χτίσιμό του. Αν η τιμή είναι True, τότε κάθε Δέντρο επιλέγει τα δείγματα με επανατοποθέτηση, δημιουργώντας διαφοροποίηση μεταξύ των Δέντρων, ως προς το σύνολο εκπαίδευσης.

• Support Vector Machines (SVM)

Τα SVM[8] ανήκουν στην κατηγορία των μη-παραμαετρικών, διαχωριστικών(discriminative) ταξινομητών οι οποίοι δε στηρίζονται στην εύρεση υποκείμενων κατανομών (δηλαδή των παραμέτρων τους), αλλά στην εύρεση μιας ευθείας ή καμπύλης (σε δισδιάτατο χώρο) ή πολύπτυχο(manifold) σε περισσότερες διαστάσεις που θα διαχωρίζουν τις κατηγορίες μεταξύ τους.

Για παράδειγμα, ακολουθεί μια απλή περίπτωση όπου υπάρχουν δύο κλάσεις δεδομένων εύκολα διαχωρίσιμες μεταξύ τους : ο ταξινομητής θα προσπαθήσει να φτιάξει μια ευθεία γραμμή η οποία θα διαχωρίζει τα δύο σύνολα δεδομένων, δημιουργώντας έτσι ένα μοντέλο ταξινόμησης. Για την απλή περίπτωση που εξετάζεται εδώ θα μπορούσε εύκολα να βρεθεί μια τέτοια ευθεία. Ωστόσο, αμέσως εμφανίζεται ένα πρόβλημα: υπάρχουν περισσότερες από μια ευθείες, πιο σωστά, άπειρες ευθείες που διαχωρίζουν τέλεια τις δύο κλάσεις.



Ενα παράδειγμα με δύο κλάσεις δεδομένων και την ευθεία διαχωρισμού

Source:https://www.saedsayad.com/support_vector_machine.htm

Το πρόβλημα διαπιστώνεται κατά τη γενίκευση: ανάλογα με το ποια διαχωριστική ευθεία θα επιλεγεί, αν εμφανιστεί ένα νέο δείγμα-σημείο από το σύνολο δεδομένων ελέγχου, είναι πιθανό να εκχωρηθεί σε μια διαφορετική ετικέτα, ανάλογα την ευθεία.

Για τη βελτίωση της επιλογής αυτής τα SVM προσφέρουν μια μέθοδο η οποία έχει ως εξής : γύρω από κάθε ευθεία σχεδιάζεται ένα περιθώριο(margin) κάποιου πλάτους το οποίο θα φτάνει μέχρι το κοντινότερο σημείο. Η γραμμή εκείνη που μεγιστοποιεί το περιθώριο θα επιλεγεί ως το βέλτιστο μόντελο. Για τον λόγο αυτό τα SVM ορίζονται ως εκτιμητές μεγιστοποίησης του περιθωρίου(maximum margin estimator).

Στην εν λόγω περίπτωση, όπου ο πειραματισμός αφορά σε μη γραμμικώς διαχωρίσιμα δεδομένα τα SVM μπορούν να δώσουν και πάλι τη λύση συνδυαζόμενα με πυρήνες(kernels) :

- Τα Kernel Functions προβάλλουν ουσιαστικά τα δεδομένα σε υψηλότερες διαστάσεις έτσι ώστε ο γραμμικός διαχωριστής που αναζητείται να είναι επαρκής.
- Στην παρούσα εργασία έγινε χρήση της RBF(Radial Basis Function)

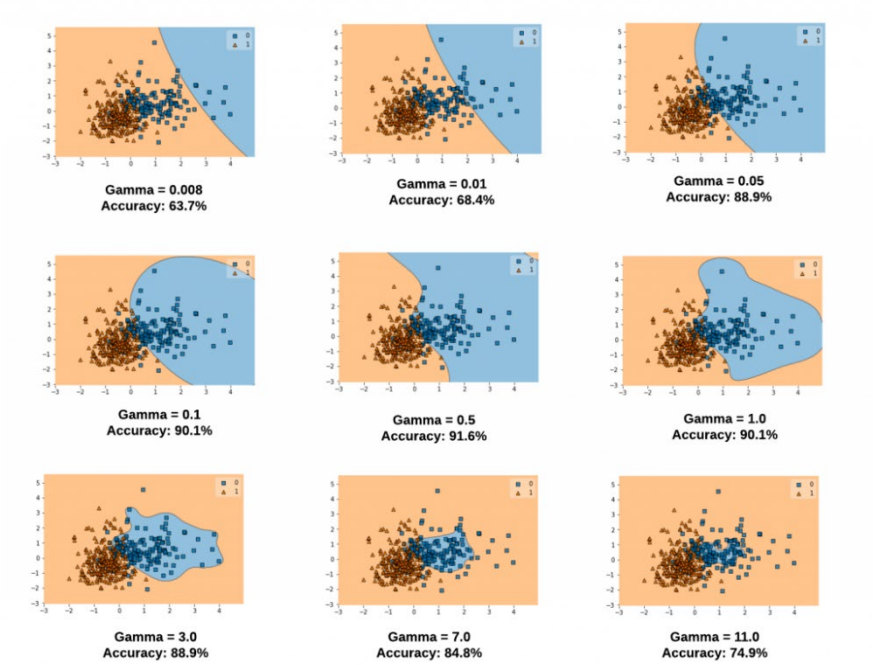
Ο αλγόριθμος SVM διαθέτει ακόμη πολλές παραμέτρους από τις οποίες επιλέχθηκαν οι παρακάτω:

→ ***C(Cost)***

Υπερπαράμετρος που ελέγχει τη σκληρότητα του περιθωρίου. Για πολύ μεγάλα C(μεγάλη πόλωση – bias), το περιθώριο είναι σκληρό και δεν μπορούν να βρεθούν καθόλου σημεία εντός του και πιθανώς αυτό να οδηγήσει σε υπερεκπαίδευση. Για μικρότερα C, το περιθώριο είναι πιο μαλακό και μπορεί να συμπεριλάβει εντός του κάποια σημεία.

→ ***g(gamma)***

Η παράμετρος gamma καθορίζει πόσο μεγάλη είναι η επιρροή ενός μεμονωμένου training παραδείγματος, με τις χαμηλές τιμές να σημαίνουν «μεγάλη» και τις υψηλές «μικρή». Οι χαμηλότερες τιμές έχουν ως αποτέλεσμα μοντέλα με χαμηλότερη ακρίβεια και το ίδιο ισχύει και με τις υψηλότερες τιμές. Είναι οι ενδιάμεσες τιμές του g που δίνουν ένα μοντέλο με καλά όρια απόφασης.



Όρια απόφασης
για διαφορετικές τιμές gamma

Source:https://vitalflux.com/svm-rbf-kernel-parameters-code-sample/

Τα παραπάνω διαγράμματα αντιπροσωπεύουν όρια απόφασης για διαφορετικές τιμές gamma με την τιμή C να ορίζεται ως 0.1 για λόγους απεικόνισης. Να σημειωθεί ότι καθώς αυξάνεται η τιμή gamma, τα όρια απόφασης ταξινομούν σωστά τα σημεία. Ωστόσο, μετά από ένα ορισμένο σημείο (Gamma = 1.0 και μετά, στο παραπάνω διάγραμμα), η ακρίβεια του μοντέλου μειώνεται. Μπορεί λοιπόν να γίνει κατανοητό ότι η επιλογή των κατάλληλων τιμών του gamma είναι σημαντική.

Όταν το gamma είναι πολύ μικρό (0,008 ή 0,01), το μοντέλο είναι πολύ περιορισμένο και δεν μπορεί να συλλάβει την πολυπλοκότητα ή τη «μορφή» των δεδομένων. Το μοντέλο που προκύπτει θα συμπεριφέρεται παρόμοια με ένα γραμμικό μοντέλο με ένα σύνολο υπερεπιπέδων που χωρίζουν τα κέντρα υψηλής πυκνότητας οποιουδήποτε ζεύγους δύο τάξεων.

Για ενδιάμεσες τιμές γάμμα (0.05, 0.1, 0.5), παρατηρεί κανείς στις γραφικές παραστάσεις ότι μπορούν να βρεθούν καλά μοντέλα.

Για μεγαλύτερες τιμές γάμμα (3.0, 7.0, 11.0) στην παραπάνω γραφική παράσταση, η ακτίνα της περιοχής επιρροής των διανυσμάτων στήριξης περιλαμβάνει μόνο το ίδιο το διάνυσμα στήριξης και κανένας βαθμός regularization με το C δεν θα μπορεί να αποτρέψει την υπερεκπαίδευση.

● Multilayer Perceptron (MLP)

Αποτελεί ένα feedforward τεχνητό νευρωνικό δίκτυο που παράγει μια έξοδο από ένα σύνολο εισόδων. Ένα MLP[9] χαρακτηρίζεται από πολλά στρώματα κόμβων εισόδου που συνδέονται ως κατευθυνόμενο γράφημα μεταξύ των επιπέδων εισόδου και εξόδου.

Πώς λειτουργεί ένα πολυστρωματικό perceptron;
Το Perceptron αποτελείται από ένα στρώμα εισόδου και ένα στρώμα εξόδου που είναι πλήρως συνδεδεμένα. Τα MLP έχουν τα ίδια επίπεδα εισόδου και εξόδου, αλλά μπορεί να έχουν πολλαπλά κρυφά επίπεδα μεταξύ των προαναφερθέντων επιπέδων, όπως φαίνεται παρακάτω.

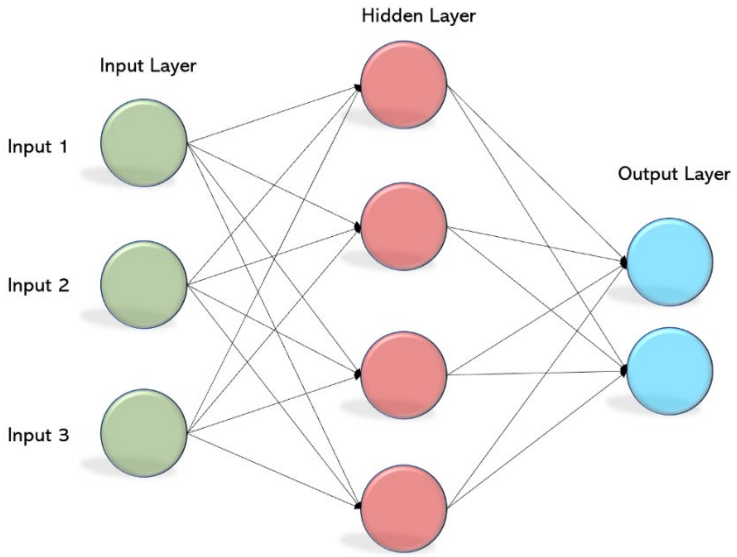
Ο αλγόριθμος για το MLP είναι ο εξής:

1. Ακριβώς όπως με το perceptron, οι είσοδοι ωθούνται προς τα εμπρός μέσω του MLP λαμβάνοντας το γινόμενο της εισόδου με τα βάρη που υπάρχουν μεταξύ του στρώματος εισόδου και του κρυφού στρώματος. Αυτό το γινόμενο αποδίδει μια τιμή στο κρυφό στρώμα. Ωστόσο, δεν προωθείται αυτήν την τιμή όπως θα γινόταν σε ένα perceptron.

2. Τα MLP χρησιμοποιούν συναρτήσεις ενεργοποίησης σε κάθε ένα από τα υπολογιζόμενα στρώματά τους. Υπάρχουν πολλές συναρτήσεις ενεργοποίησης : ReLU, σιγμοειδής συνάρτηση, tanh. Η υπολογισμένη έξοδος προωθείται στο τρέχον επίπεδο μέσω οποιασδήποτε από αυτές τις συναρτήσεις ενεργοποίησης.

3. Μόλις η υπολογισμένη έξοδος στο κρυφό στρώμα προωθηθεί μέσω της συνάρτησης ενεργοποίησης, συνεχίζει στο επόμενο στρώμα στο MLP παίρνοντας το γινόμενο με τα αντίστοιχα βάρη.

4. Στο επίπεδο εξόδου, οι υπολογισμοί είτε θα χρησιμοποιηθούν για έναν αλγόριθμο backpropagation που αντιστοιχεί στη συνάρτηση ενεργοποίησης που επιλέχθηκε για το MLP (στην περίπτωση εκπαίδευσης) είτε θα ληφθεί απόφαση με βάση την έξοδο (στην περίπτωση δοκιμής).



Δομή ενός MLP τριών στρωμάτων

Source:https://becominghuman.ai/multi-layer-perceptron-mlp-models-on-real-world-banking-data-f6dd3d7e998f

Ο αλγόριθμος backpropagation που αναφέρεται παραπάνω είναι η μέθοδος μικρορύθμισης των βαρών ενός νευρωνικού δικτύου με βάση το ποσοστό σφάλματος που λήφθηκε στην προηγούμενη εποχή (δηλαδή, επανάληψη). Ο σωστός συντονισμός των βαρών επιτρέπει να μειωθούν τα ποσοστά σφάλματος και να γίνει το μοντέλο αξιόπιστο αυξάνοντας τη γενίκευσή του. Η μέθοδος αυτή στο νευρωνικό δίκτυο είναι μια σύντομη μορφή για την «προς τα πίσω διάδοση σφαλμάτων». Είναι μια τυπική μέθοδος εκπαίδευσης τεχνητών νευρωνικών δικτύων. Αυτή η μέθοδος βοηθά στον υπολογισμό της κλίσης μιας συνάρτησης απώλειας σε σχέση με όλα τα βάρη στο δίκτυο.

Διαθέτει αρκετές παραμέτρους εκ των οποίων κάποιες επιλέχθηκαν για την παρούσα εργασία και έχουν ως εξής :

→ ***Hidden layer sizes***

Η παράμετρος αυτή ορίζει το πλήθος των νευρώνων σε κάθε κρυφό επίπεδο.

→ ***Solver***

Η παράμετρος αυτή ορίζει τη μέθοδο που θα χρησιμοποιηθεί για τη βελτιστοποίηση των βαρών.

→ ***Max iter***

Η παράμετρος αυτή ορίζει το μέγιστο πλήθος επαναλήψεων για τις οποίες θα εκτελεστεί ο solver

→ ***Alpha***

Το Alpha είναι μια παράμετρος για τον όρο κανονικοποίησης, γνωστός και ως όρος ποινής που έχει ως στόχο να αντιμετωπίσει την υπερεκπαίδευση περιορίζοντας το μέγεθος των βαρών. Η αύξηση του alpha μπορεί να διορθώσει την υψηλή διακύμανση (ένα σημάδι υπερεκπαίδευσης) επιτρέποντας τα μικρότερα βάρη. Ομοίως, η μείωση του alpha μπορεί να διορθώσει την υψηλή μεροληψία (ένα σημάδι υποεκπαίδευσης) ενθαρρύνοντας μεγαλύτερα βάρη, δυνητικά με αποτέλεσμα ένα πιο περίπλοκο όριο απόφασης.

- XGBoost

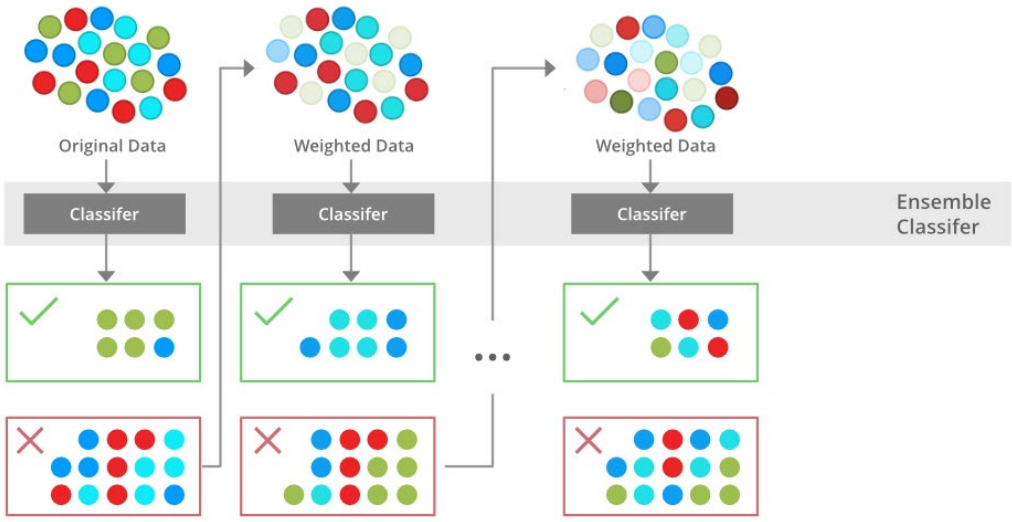
Ο XGBoost[10] συντομογραφία για τον Extreme Gradient Boosting είναι μια υλοποίηση που βελτιστοποιεί την εκπαίδευση για Gradient Boosting.

Πριν κατανοήσουμε το XGBoost, πρέπει πρώτα να κατανοήσουμε τα δέντρα, ειδικά το δέντρο αποφάσεων:

- Δέντρο αποφάσεων: Ένα δέντρο απόφασης είναι μια δομή δέντρου τύπου διαγράμματος ροής, όπου κάθε εσωτερικός κόμβος υποδηλώνει μια δοκιμή σε ένα χαρακτηριστικό, κάθε κλάδος αντιπροσωπεύει ένα αποτέλεσμα της δοκιμής και κάθε κόμβος φύλλου (τερματικός κόμβος) έχει μια ετικέτα κλάσης. Ένα δέντρο μπορεί να «εκπαιδευτεί» χωρίζοντας το σύνολο πηγών σε υποσύνολα με βάση μια δοκιμή τιμής χαρακτηριστικών. Αυτή η διαδικασία επαναλαμβάνεται σε κάθε παραγόμενο υποσύνολο με αναδρομικό τρόπο που ονομάζεται αναδρομική κατάτμηση. Η αναδρομή ολοκληρώνεται όταν το υποσύνολο σε έναν κόμβο έχει την ίδια τιμή με τη μεταβλητή-στόχο ή όταν ο διαχωρισμός δεν προσθέτει πλέον αξία στις προβλέψεις.

- Boosting: Το Boosting είναι μια μοντελοποίηση συνόλου(ensemble modelling), τεχνική που επιχειρεί να δημιουργήσει έναν ισχυρό ταξινομητή από τον αριθμό των αδύναμων ταξινομητών. Γίνεται με την κατασκευή ενός μοντέλου χρησιμοποιώντας αδύναμα μοντέλα σε σειρά. Πρώτον, δημιουργείται ένα μοντέλο από τα δεδομένα εκπαίδευσης. Στη συνέχεια κατασκευάζεται το δεύτερο μοντέλο το οποίο προσπαθεί να διορθώσει τα σφάλματα που υπάρχουν στο πρώτο μοντέλο. Αυτή η διαδικασία συνεχίζεται και προστίθενται μοντέλα

μέχρι να προβλεφθεί σωστά το πλήρες σύνολο δεδομένων εκπαίδευσης ή να προστεθεί ο μέγιστος αριθμός μοντέλων.



Οπτική αναπαράσταση της μεθόδου Boosting
Source:https://www.geeksforgeeks.org/xgboost/

- Gradient Boosting: Ο Gradient Boosting είναι ένας δημοφιλής αλγόριθμος ενίσχυσης. Κάθε προγνωστικός παράγοντας διορθώνει το σφάλμα του προκατόχου του. Τα βάρη των περιπτώσεων εκπαίδευσης δεν τροποποιούνται, αντίθετα, κάθε προγνωστικός παράγοντας εκπαιδεύεται χρησιμοποιώντας τα υπολειπόμενα σφάλματα του προκατόχου ως ετικέτες.

Εν τέλει στον XGBoost, τα δέντρα αποφάσεων δημιουργούνται σε διαδοχική μορφή. Τα βάρη παίζουν σημαντικό ρόλο στο XGBoost. Τα βάρη εκχωρούνται σε όλες τις ανεξάρτητες μεταβλητές οι οποίες στη συνέχεια τροφοδοτούνται στο δέντρο αποφάσεων που προβλέπει τα αποτελέσματα. Το βάρος των μεταβλητών που προβλέπονται λανθασμένα από το δέντρο αυξάνεται και αυτές οι μεταβλητές τροφοδοτούνται στη συνέχεια στο δεύτερο δέντρο απόφασης. Αυτοί οι μεμονωμένοι ταξινομητές/προγνωστικοί δείκτες στη συνέχεια συνδυάζονται για να δώσουν ένα ισχυρό και πιο ακριβές μοντέλο. Μπορεί να λειτουργήσει σε προβλήματα παλινδρόμησης, ταξινόμησης, κατάταξης και πρόβλεψης.

Ο αλγόριθμος διαθέτει δεκάδες παραμέτρους, ωστόσο για την εργασία επιλέχθηκαν οι παρακάτω:

→ ***max depth***: περιορίζει το βάθος στο οποίο μπορεί να αναπτυχθεί κάθε δέντρο. Η προεπιλεγμένη τιμή είναι 6, αλλά μπορούν να δοκιμαστούν άλλες τιμές εάν υπάρχει overfitting στο μοντέλο.
→ ***learning rate***: είναι μια παράμετρος κανονικοποίησης που συρρικνώνει τα βάρη χαρακτηριστικών σε κάθε βήμα ενίσχυσης.

→ ***n_estimators***: καθορίζει τον αριθμό των δέντρων απόφασης που θα ενισχυθούν. Εάν n_estimators = 1, σημαίνει ότι δημιουργείται μόνο 1 δέντρο, επομένως δεν υπάρχει καμία ενίσχυση. Δοκιμάζονται διάφορες τιμές με στόχο τη βέλτιστη απόδοση

→ ***gamma***: είναι ακόμη μια παράμετρος κανονικοποίησης για το κλάδεμα των δέντρων. Καθορίζει την απώλεια ελάχιστης μείωσης που απαιτείται για την ανάπτυξη ενός δέντρου.

→ ***min_child_weight***: καθορίζει το ελάχιστο άθροισμα βαρών όλων των παρατηρήσεων που απαιτούνται σε ένα παιδί. Αναφέρεται στο ελάχιστο «άθροισμα βαρών» των παρατηρήσεων και χρησιμοποιείται για τον έλεγχο του overfitting. Οι υψηλότερες τιμές εμποδίζουν ένα μοντέλο από σχέσεις μάθησης που μπορεί να είναι ιδιαίτερα συγκεκριμένες για το δείγμα που έχει επιλεγεί για ένα δέντρο. Οι πολύ υψηλές τιμές μπορεί να οδηγήσουν σε underfitting. Όσο μεγαλύτερο είναι το min_child_weight, τόσο πιο συντηρητικός θα είναι ο αλγόριθμος.

● Ensemble Methods – Μέθοδοι Συνόλου

1. Voting Classifier

Ο ταξινομητής ψηφοφορίας λειτουργεί σαν ένα εκλογικό σύστημα στο οποίο γίνεται μια πρόβλεψη για ένα νέο σημείο δεδομένων με βάση ένα σύστημα ψηφοφορίας των μελών μιας ομάδας μοντέλων μηχανικής μάθησης. Διακρίνουμε δύο τύπους ψηφοφορίας, hard και soft voting.

Ο τύπος hard voting εφαρμόζεται σε προβλεπόμενες ετικέτες για ψηφοφορία με κανόνα πλειοψηφίας. Αυτό χρησιμοποιεί την ιδέα του «Η πλειοψηφία φέρει την ψήφο», δηλαδή λαμβάνεται μια απόφαση υπέρ αυτού που έχει περισσότερες από τις μισές ψήφους.

Ο τύπος soft voting προβλέπει την ετικέτα της τάξης με βάση το argmax των αθροισμάτων των προβλεπόμενων πιθανοτήτων των επιμέρους εκτιμητών που απαρτίζουν το σύνολο. Η ομαλή ψηφοφορία συνιστάται συχνά στην περίπτωση ενός συνόλου καλά βαθμονομημένων/τοποθετημένων ταξινομητών.

Για παράδειγμα, εάν το μοντέλο 1 προβλέπει το Α, και το μοντέλο 2 προβλέπει το Β και το μοντέλο 3 προβλέπει το Α, τότε ο ταξινομητής ψηφοφορίας (με voting='hard') επιστρέφει Α. Σε περίπτωση ισοψηφίας, ο ταξινομητής θα επιλέξει την κατηγορία με βάση την αύξουσα σειρά.

2. Stacking Classifier

Η μέθοδος stacking περιλαμβάνει το συνδυασμό των προβλέψεων από πολλαπλά μοντέλα μηχανικής μάθησης στο ίδιο σύνολο δεδομένων. Αρχικά καθορίζονται / δημιουργούνται ορισμένα μοντέλα μηχανικής εκμάθησης που ονομάζονται εκτιμητές βάσης στο σύνολο δεδομένων, στη συνέχεια τα αποτελέσματα από αυτούς τους βασικούς εκπαιδευτές χρησιμεύουν ως είσοδος στον stacking ταξινομητή. Ο ταξινομητής μπορεί να μάθει πότε οι βασικοί εκτιμητές μπορούν να είναι αξιόπιστοι ή όχι. Η στοίβαξη επιτρέπει να εκμεταλλευθεί η ισχύς κάθε μεμονωμένου εκτιμητή χρησιμοποιώντας την έξοδο του ως είσοδο σε έναν τελικό εκτιμητή.

Κατά τη χρήση του Stacking Classifier, μπορεί κανείς να επιλέξει να εφαρμόσει cross-validation στο βασικό επίπεδο μάθησης ή σε αυτό στον τελικό εκτιμητή. Χρησιμοποιώντας τον stacking ταξινομητή από το sklearn.ensemble, οι βασικοί εκπαιδευόμενοι εκτιμητές τοποθετούνται στο πλήρες X σύνολο δεδομένων ενώ ο τελικός εκτιμητής εκπαιδεύεται χρησιμοποιώντας cross-validated προβλέψεις των βασικών «μαθητών».

Η στοίβαξη πολλαπλών επιπέδων είναι επίσης δυνατή, όπου κάποιος χτίζει στρώματα βασικών εκτιμητών πριν κατασκευαστεί ένας τελικός εκτιμητής. Έχοντας αυτήν τη δυνατότητα η παρούσα εργασία περιέχει πειράματα 2-stage και 3-stage Stacking Classifiers οι οποίοι και θα αναλυθούν περαιτέρω σε επόμενο κεφάλαιο.

- AdaBoost

Ο AdaBoost (Adaptive Boosting) είναι μια πολύ δημοφιλής τεχνική ενίσχυσης που στοχεύει στο συνδυασμό πολλών αδύναμων ταξινομητών για τη δημιουργία ενός ισχυρού ταξινομητή.

Τώρα, μπορεί να αναρωτηθεί κάποιος τί είναι ένας «αδύναμος» ταξινομητής; Ένας αδύναμος ταξινομητής είναι αυτός που αποδίδει καλύτερα από την τυχαία εικασία, αλλά εξακολουθεί να έχει κακή απόδοση στον προσδιορισμό κλάσεων σε αντικείμενα. Για παράδειγμα, ένας αδύναμος ταξινομητής μπορεί να προβλέψει ότι όλοι οι άνω των 40 ετών δεν θα μπορούσαν να τρέξουν έναν μαραθώνιο, αλλά οι άνθρωποι που πέφτουν κάτω από αυτήν την ηλικία θα μπορούσαν. Τώρα, μπορεί να έχει πάνω από 60% ακρίβεια, αλλά θα εξακολουθεί να ταξινομεί εσφαλμένα πολλά σημεία δεδομένων!

Αντί να είναι ένα μοντέλο από μόνο του, ο AdaBoost μπορεί να εφαρμοστεί πάνω από οποιονδήποτε ταξινομητή για να μάθει από τα μειονεκτήματά του και να προτείνει ένα πιο ακριβές μοντέλο. Συνήθως αποκαλείται ο «καλύτερος έξω από το κουτί ταξινομητής» για αυτόν τον λόγο.

Χρήσιμη είναι επίσης και η κατανόηση του πώς λειτουργεί με τα Decision Stumps. Αυτά είναι σαν τα δέντρα σε ένα τυχαίο δάσος, αλλά όχι «πλήρως ανεπτυγμένα». Έχουν έναν κόμβο και δύο φύλλα. Ο AdaBoost χρησιμοποιεί ένα δάσος τέτοιων stumps και όχι δέντρων.

Τα stumps από μόνα τους δεν είναι καλός τρόπος για τη λήψη αποφάσεων. Ένα δέντρο με πλήρη ανάπτυξη συνδυάζει τις αποφάσεις από όλες τις μεταβλητές για να προβλέψει την τιμή στόχο. Ένα stump, από την άλλη πλευρά, μπορεί να χρησιμοποιήσει μόνο μία μεταβλητή για να λάβει μια απόφαση.

Ένα αντιπροσωπευτικό παράδειγμα για την εσωτερική λειτουργία του αλγόριθμου AdaBoost παρατίθεται στη συνέχεια βήμα-βήμα, εξετάζοντας διάφορες μεταβλητές, για να προσδιοριστεί εάν ένα άτομο είναι «σε καλή κατάσταση» (με καλή υγεία) ή όχι.

Ένα παράδειγμα του πώς λειτουργεί το AdaBoost:

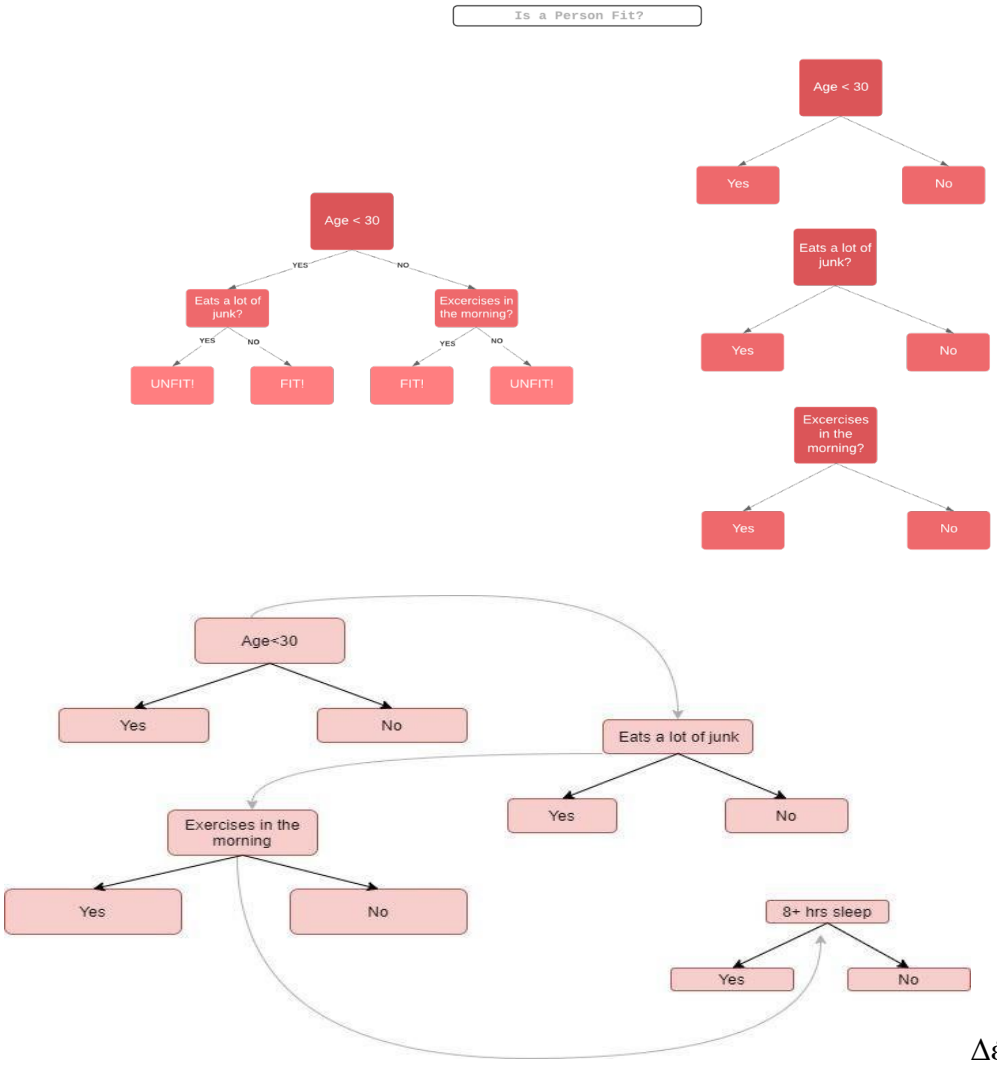
Βήμα 1: Ένας ασθενής ταξινομητής (π.χ. ένα decision stump) δημιουργείται πάνω από τα δεδομένα εκπαίδευσης με βάση τα σταθμισμένα δείγματα. Εδώ, τα βάρη κάθε δείγματος δείχνουν πόσο σημαντικό είναι να ταξινομηθεί σωστά. Αρχικά για το πρώτο stump δίνουμε σε όλα τα δείγματα ίσα βάρη.

Βήμα 2: Δημιουργείται ένα decision stump για κάθε μεταβλητή και φαίνεται πόσο καλά κάθε stump ταξινομεί τα δείγματα στις κατηγορίες-στόχους τους.

Για παράδειγμα, στο παρακάτω διάγραμμα ελέγχεται η ηλικία, η κατανάλωση πρόχειρου φαγητού και η άσκηση. Θα εξεταστεί πόσα δείγματα έχουν ταξινομηθεί σωστά ή λανθασμένα ως Κατάλληλα ή Ακατάλληλα για κάθε μεμονωμένο stump.

Βήμα 3: Αποδίδεται μεγαλύτερο βάρος στα λανθασμένα ταξινομημένα δείγματα, ώστε να ταξινομηθούν σωστά στο επόμενο stump απόφασης. Το βάρος αποδίδεται επίσης σε κάθε ταξινομητή με βάση την ακρίβεια του ταξινομητή, που σημαίνει υψηλή ακρίβεια = μεγάλο βάρος!

Βήμα 4: Επαναλαμβάνεται το Βήμα 2 μέχρι να ταξινομηθούν σωστά όλα τα σημεία δεδομένων ή να επιτευχθεί το μέγιστο επίπεδο επανάληψης.



Δέντρα αποφάσεων του AdaBoost

Source:https://www.how2shout.com/what-is/what-is-adaboost-boosting-technique.html

Σε αντιστοιχία με τα υπόλοιπα μοντέλα που παρατίθενται στην παρούσα εργασία και των οποίων οι παράμετροι καθορίστηκαν με την τεχνική του grid search, έτσι συνέβη και στην περίπτωση του AdaBoost. Πιο συγκεκριμένα οι παράμετροι που χρησιμοποιήθηκαν είναι :

→ ***n_estimators***: καθορίζει το μέγιστο αριθμό εκτιμητών στους οποίους τερματίζεται η ενίσχυση. Σε περίπτωση τέλειας εφαρμογής, η διαδικασία εκμάθησης διακόπτεται έγκαιρα. Οι τιμές πρέπει να βρίσκονται στην περιοχή [1, inf).

→ ***learning_rate***: καθορίζει το βάρος που εφαρμόζεται σε κάθε ταξινομητή σε κάθε επανάληψη ενίσχυσης. Ένα υψηλότερο ποσοστό μάθησης αυξάνει τη συμβολή κάθε ταξινομητή. Υπάρχει μια αντιστάθμιση μεταξύ των παραμέτρων learning_rate και n_estimators. Οι τιμές πρέπει να βρίσκονται στο εύρος (0.0, inf).

● Artificial Neural Network (ANN)

Ένα τεχνητό δίκτυο νευρώνων (νευρωνικό δίκτυο)[13] είναι ένα υπολογιστικό μοντέλο που μιμείται τον τρόπο που λειτουργούν τα νευρικά κύτταρα στον ανθρώπινο εγκέφαλο. Τα τεχνητά νευρωνικά δίκτυα (ANN) χρησιμοποιούν αλγόριθμους εκμάθησης που μπορούν ανεξάρτητα να κάνουν προσαρμογές – ή να μάθουν, κατά μία έννοια – καθώς λαμβάνουν νέα δεδομένα. Αυτό τα καθιστά ένα πολύ αποτελεσματικό εργαλείο για μη γραμμική μοντελοποίηση στατιστικών δεδομένων. Τα ANN Βαθιάς Μάθησης διαδραματίζουν σημαντικό ρόλο στη μηχανική μάθηση (ML) και υποστηρίζουν το ευρύτερο πεδίο της τεχνολογίας τεχνητής νοημοσύνης (AI).

Ένα τεχνητό νευρωνικό δίκτυο έχει τρία ή περισσότερα στρώματα που είναι διασυνδεδεμένα. Το πρώτο στρώμα αποτελείται από νευρώνες εισόδου. Αυτοί οι νευρώνες στέλνουν δεδομένα στα βαθύτερα στρώματα, τα οποία με τη σειρά τους θα στείλουν τα τελικά δεδομένα εξόδου στο τελευταίο στρώμα εξόδου. Όλα τα εσωτερικά στρώματα είναι κρυμμένα και σχηματίζονται από μονάδες που αλλάζουν προσαρμοστικά τις πληροφορίες που λαμβάνονται από επίπεδο σε επίπεδο μέσω μιας σειράς μετασχηματισμών. Κάθε επίπεδο λειτουργεί και ως στρώμα εισόδου και εξόδου που επιτρέπει στο ANN να κατανοεί πιο σύνθετα αντικείμενα. Συλλογικά, αυτά τα εσωτερικά επίπεδα ονομάζονται νευρικό στρώμα.

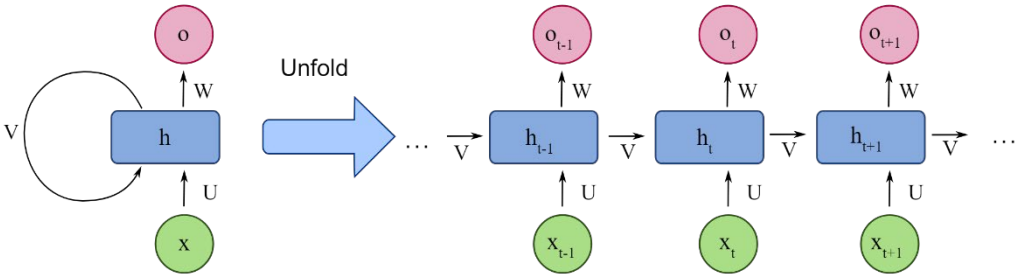
Ένα επιπλέον σύνολο κανόνων εκμάθησης κάνει χρήση της backpropagation, μιας διαδικασίας μέσω της οποίας το ANN μπορεί να προσαρμόσει τα αποτελέσματα εξόδου του λαμβάνοντας υπόψη τα σφάλματα. Μέσω της backpropagation, κάθε φορά που η έξοδος επισημαίνεται ως σφάλμα κατά τη διάρκεια της εποπτευόμενης φάσης εκπαίδευσης, οι πληροφορίες αποστέλλονται προς τα πίσω. Κάθε βάρος ενημερώνεται ανάλογα με το πόσο ήταν υπεύθυνα για το σφάλμα.

Ένα ANN έχει πολλά πλεονεκτήματα, αλλά ένα από τα πιο αναγνωρισμένα είναι το γεγονός ότι μπορεί πραγματικά να μάθει από την παρατήρηση συνόλων δεδομένων. Με αυτόν τον τρόπο, το ANN χρησιμοποιείται ως εργαλείο προσέγγισης τυχαίων συναρτήσεων.

Ένας εξειδικεύμενος τύπος τεχνητού νευρωνικού δικτύου είναι το RNN, Recurrent Artificial Network, το οποίο χρησιμοποιείται για χρονοσειρές ή διαδοχικά δεδομένα. Τα νευρωνικά δίκτυα εμπρόσθιας τροφοδοσίας (Feedforward NNs) χρησιμοποιούνται όταν τα σημεία δεδομένων είναι ανεξάρτητα μεταξύ τους. Στην περίπτωση διαδοχικών σημείων

δεδομένων εξαρτώνται το ένα από το άλλο. Σε αυτήν την περίπτωση, θα πρέπει να τροποποιηθούν τα νευρωνικά δίκτυα για να ενσωματώσουν εξαρτήσεις μεταξύ σημείων δεδομένων. Τα RNNs έχουν την έννοια της μνήμης, η οποία τους βοηθά να αποθηκεύουν καταστάσεις ή πληροφορίες προηγούμενων εισόδων για να δημιουργήσουν την επόμενη ακολουθία εξόδου.

Αποθηκεύει την έξοδο ενός συγκεκριμένου στρώματος και το τροφοδοτεί πίσω στην είσοδο για να προβλέψει την έξοδο του στρώματος. Όπως δείχνει η παρακάτω εικόνα, μπορεί να μετατραπεί ένα κανονικό νευρωνικό δίκτυο προώθησης σε RNN. Οι κόμβοι στα διαφορετικά στρώματα του νευρωνικού δικτύου συμπιέζονται για να σχηματίσουν ένα ενιαίο στρώμα. Στην παρακάτω εικόνα, τα U, V και W είναι οι παράμετροι του δικτύου.



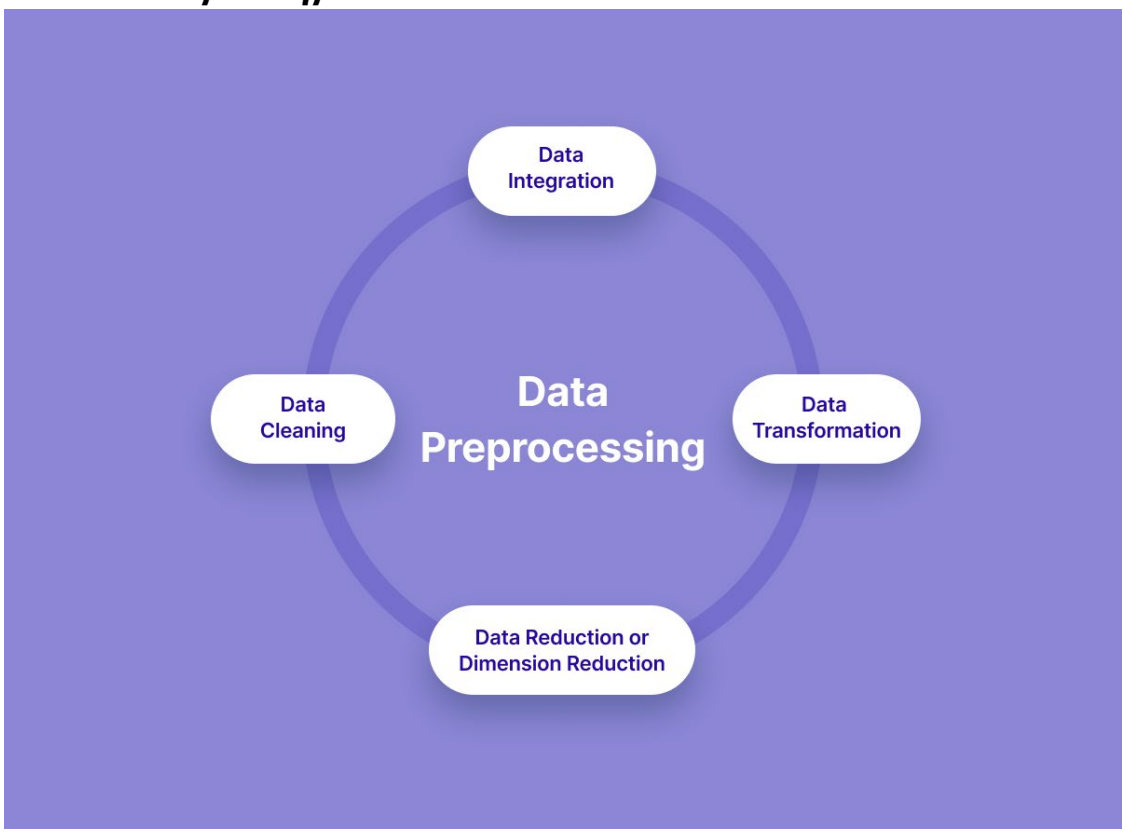
Μετατροπή ενός κλασικού NN σε RNN
Source:https://commons.wikimedia.org/wiki/File:Recurrent_neural_network_unfold.svg

Εδώ, x είναι το επίπεδο εισόδου, h είναι το κρυφό στρώμα και o είναι το επίπεδο εξόδου. Τα U, V και W είναι οι παράμετροι δικτύου που χρησιμοποιούνται για τη βελτίωση της απόδοσης του μοντέλου. Σε οποιαδήποτε δεδομένη στιγμή (t), η τρέχουσα είσοδος είναι ένας συνδυασμός εισόδου στο x(t) και x(t-1). Η έξοδος επαναφέρεται στο δίκτυο για να βελτιωθεί η έξοδος.

Στην παρούσα εργασία γίνεται χρήση μιας αρχιτεκτονικής RNN που ονομάζεται Long short-term memory (LSTM). Το LSTM έχει συνδέσεις ανάδρασης(feedback), οι οποίες δεν υπάρχουν στα feedforward νευρωνικά δίκτυα. Το LSTM μπορεί να επεξεργάζεται όχι μόνο μεμονωμένα σημεία δεδομένων, αλλά και ολόκληρες ακολουθίες δεδομένων. Η υλοποίηση ενός LSTM αποτελείται από ένα κελί, μια πύλη εισόδου, μια πύλη εξόδου και μια πύλη λήθης. Το κελί θυμάται τιμές σε αυθαίρετα χρονικά διαστήματα και οι τρεις πύλες ρυθμίζουν τη ροή των πληροφοριών μέσα και έξω από το κελί. Το LSTM είναι κατάλληλο για την επεξεργασία, την ταξινόμηση και τη δημιουργία προβλέψεων με βάση δεδομένα χρονοσειρών, καθώς μπορεί να υπάρχουν καθυστερήσεις άγνωστης διάρκειας μεταξύ σημαντικών γεγονότων σε μια χρονοσειρά. Το LSTM είναι η καλύτερη δυνατή λύση σήμερα για την επίλυση προβλημάτων που σχετίζονται με ακολουθίες και σειρές. Το μόνο μειονέκτημα του LSTM είναι ο χρόνος που απαιτείται για την εκπαίδευση ενός μοντέλου.

Η Python διαθέτει τη βιβλιοθήκη Keras – Tensorflow που εξυπηρετεί στη δημιουργία νευρωνικών δικτύων και μοντέλων βαθιάς μάθησης. Στην εργασία αυτή η οποία είναι ουσιαστικά ένα πρόβλημα δυαδικής ταξινόμησης προτιμήθηκαν συγκεκριμένες συναρτήσεις ενεργοποίησης και πλήθος στρωμάτων τα οποία και θα αναπτύξουμε εκτενέστερα στο αντίστοιχο κεφάλαιο στη συνέχεια.

2.2 Προεπεξεργασία Δεδομένων – Επιλογή Χαρακτηριστικών



Ο κύκλος προεπεξεργασίας ενός συνόλου δεδομένων

Source:https://www.v7labs.com/blog/data-preprocessing-guide

2.2.1 Κύκλος Προεπεξεργασίας

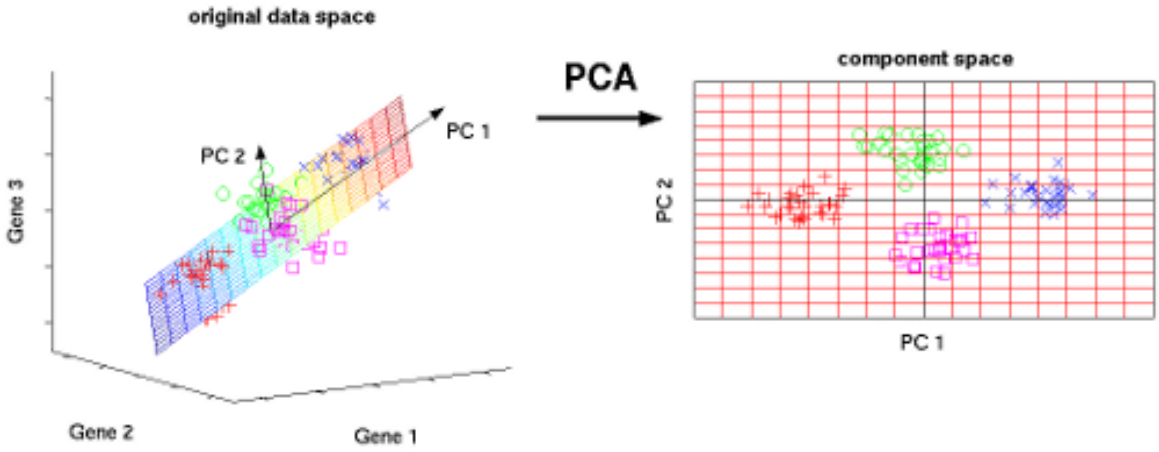
Ένας πολύ σημαντικός παράγοντας για την καλύτερη απόδοση των μοντέλων Μηχανικής Μάθησης είναι το λεγόμενο data preprocessing ή αλλιώς προεπεξεργασία δεδομένων. Η διαδικασία αυτή εξυπηρετεί στον «καθαρισμό» και την προετοιμασία του αρχικού συνόλου δεδομένων και αποσκοπεί στην εύρεση των χαρακτηριστικών εκείνων με την περισσότερη χρήσιμη πληροφορία. Για να επιτευχθεί αυτή η συλλογή των σημαντικότερων χαρακτηριστικών είναι απαραίτητο να εφαρμοστούν κάποια βασικά βήματα – τεχνικές :

1. Data Cleaning – Καθαρισμός των δεδομένων: δίνεται προσοχή στην ύπαρξη missing values, σε λάθος τιμές χαρακτηριστικών (π.χ. αριθμός εκεί που αναμένεται string), σε πολλαπλά αντίγραφα χαρακτηριστικών ή τιμές χαρακτηριστικών.

2. Data Integration – Ενσωμάτωση Δεδομένων: σε αυτό το βήμα μπορούν να συνδυαστούν δεδομένα από διαφορετικές πηγές σε ένα ενιαίο σύνολο. Πιθανώς να χρειάζονται αλλαγές ώστε να έχουμε τη σωστή δομή στο dataset, χωρίς λανθασμένες τιμές, διπλότυπα στα χαρακτηριστικά και περιττή πληροφορία.

3. Data Transformation – Τροποποίηση Δεδομένων: αφού έχουν προηγηθεί τα προηγούμενα βήματα, ακολουθεί η τροποποίηση του τελικού dataset. Σε αυτό το στάδιο μπορούν να εφαρμοστούν πολλές τεχνικές με σημαντικότερες την κανονικοποίηση (normalization) και την κλιμάκωση(scaling) των δεδομένων οι οποίες έχουν εφαρμοστεί και σε κάποια μοντέλα στην παρούσα διπλωματική. Μια ακόμη ιδέα είναι η χρήση Μεθόδων Επιλογής Χαρακτηριστικών(Feature Selection Methods) μερικές από τις οποίες θα αναλύσουμε και στη συνέχεια, καθώς φάνηκαν ιδιαίτερα χρήσιμες κατά τη δημιουργία των ταξινομητών.

4. Data Reduction – Μείωση Δεδομένων: ένα τελικό βήμα που μπορεί να βοηθήσει σε περιπτώσεις με μεγάλο όγκο δεδομένων. Στη συγκεκριμένη εργασία δοκιμάστηκε μια ευρέως διαδεδομένη τεχνική για τη μείωση της διαστατικότητας, η PCA(Principal Component Analysis).



Οπτική αναπαράσταση της εφαρμογής της PCA

Source:https://dimensionless.in/principal-component-analysis-in-r/

Αναλυτικότερα: η PCA επιτυγχάνει τη μείωση της διαστατικότας ακολουθώντας τα παρακάτω βήματα:

I. Standardization: επειδή υπάρχουν πολλά χαρακτηριστικά με πολύ μεγάλες διαφορές στις απόλυτες και στις μέσες τιμές τους, θα πρέπει με κάποιον τρόπο να αμβλυνθούν οι διαφορές αυτές. Μια λύση τη δίνει η τεχνική του standardization και συγκεκριμένα με την διαίρεση με τη διαφορά μεγίστου-ελαχίστου (feature scaling) οπότε οι τιμές όλων των χαρακτηριστικών κλιμακώνονται γραμμικά στο διάστημα [0,1]. Η κλιμάκωση σε [0,1] είναι λιγότερο ευαίσθητη σε πολύ μικρές αποκλίσεις και επίσης σε αραιά (sparse) διανύσματα χαρακτηριστικών (δηλαδή με πολλές μηδενικές τιμές) η εφαρμογή της διατηρεί τα μηδέν, κάτι που μπορεί να είναι καθοριστικό για την ταχύτητα εκπαίδευσης.

   Η μετατροπή μεγίστου ελαχίστου γίνεται με τον τύπο:

$X' = X - Xmin \ / \ Xmax - Xmin$.
Το sklearn προσφέρει τον μετασχηματιστή (Imputer) MinMax Scaler ο οποίος και εκτελεί την παραπάνω διαδικασία κανονικοποίησης.

Η μετατροπή σε standard score γίνεται με τον τύπο:
$z = X - \mu \ / \ \sigma$,
όπου: μ είναι η μέση τιμή του χαρακτηριστικού και σ η απόκλιση.
Η μετατροπή σε standard score είναι απαραίτητη σε πολλούς ταξινομητές για να συμπεριφερθούν σωστά. Επίσης είναι πιο ανθεκτική από την min-max σε τιμές outliers δηλαδή σποραδικές τιμές που είναι πολύ μακριά απο τη μέση τιμή και τις υπόλοιπες τιμές του χαρακτηριστικού (η min-max θα συμπιέσει τις περισσότερες τιμές σε ένα μικρό διάστημα). Το sklearn προσφέρει τον μετασχηματιστή (Imputer) Standard Scaler ο οποίος και εκτελεί την παραπάνω διαδικασία κανονικοποίησης.

II. Υπολογισμός του Πίνακα Συσχέτισης(Covariance Matrix): από αυτόν τον πίνακα βρίσκονται οι γραμμικώς συσχετισμένες μεταβλητές.

III. Υπολογισμός των Ιδιοδιανυσμάτων και των Ιδιοτιμών του Πίνακα Συσχέτισης: τα ιδιοδιανύσματα οδηγούν προς την κατεύθυνση των πιο ασυσχέτιστων μεταβλητών. Το ιδιοδιάνυσμα με τη μεγαλύτερη ιδιοτιμή θα αποτελεί και την πρώτη κύρια συνιστώσα της νέας βάσης του χώρου που θα δημιουργηθεί. Εν τέλει θα υπάρξει ένα νέο σύνολο μεταβλητών-κυρίων συνιστωσών που είναι γραμμικά ασυσχέτιστες.

2.2.2 Μέθοδοι Επιλογής Χαρακτηριστικών

Η επιλογή χαρακτηριστικών είναι η διαδικασία μείωσης του αριθμού των μεταβλητών εισόδου κατά την ανάπτυξη ενός μοντέλου πρόβλεψης. Είναι επιθυμητό να μειωθεί ο αριθμός των μεταβλητών εισόδου τόσο για τη μείωση του υπολογιστικού κόστους της μοντελοποίησης όσο και, σε ορισμένες περιπτώσεις, για τη βελτίωση της απόδοσης του μοντέλου.

Οι μέθοδοι επιλογής χαρακτηριστικών[14] που βασίζονται σε στατιστικά περιλαμβάνουν την αξιολόγηση της σχέσης μεταξύ κάθε μεταβλητής εισόδου και της μεταβλητής στόχου χρησιμοποιώντας στατιστικά και την επιλογή εκείνων των μεταβλητών εισόδου που έχουν την ισχυρότερη σχέση με τη μεταβλητή στόχο. Αυτές οι μέθοδοι μπορεί να είναι γρήγορες και αποτελεσματικές, αν και η επιλογή των στατιστικών μετρικών εξαρτάται από τον τύπο δεδομένων τόσο των μεταβλητών εισόδου όσο και των μεταβλητών εξόδου.

Λαμβάνοντας αυτά τα στοιχεία υπόψιν δοκιμάστηκαν αρκετές διαφορετικές μέθοδοι στην παρούσα διπλωματική. Στη συνέχεια αναλύονται κάποιες εξ αυτών οι οποίες απέδωσαν καλύτερα κατά τις δοκιμές των μοντέλων Μηχανικής Μάθησης.

- **Univariate feature selection:** η συγκεκριμένη μέθοδος εξετάζει κάθε χαρακτηριστικό ξεχωριστά για να προσδιορίσει την ισχύ της σχέσης του χαρακτηριστικού με τη μεταβλητή στόχο. Λειτουργεί επιλέγοντας τα καλύτερα χαρακτηριστικά με βάση μονομεταβλητές στατιστικές δοκιμές. Αυτές οι μέθοδοι είναι απλές στην εκτέλεση και κατανόηση και γενικά είναι ιδιαίτερα καλές για την καλύτερη κατανόηση των δεδομένων (αλλά όχι απαραίτητα για τη βελτιστοποίηση του συνόλου χαρακτηριστικών για καλύτερη γενίκευση).

  Το Scikit-learn API παρέχει την κλάση SelectKBest για εξαγωγή των καλύτερων χαρακτηριστικών του συνόλου δεδομένων. Η μέθοδος SelectKBest επιλέγει τα χαρακτηριστικά σύμφωνα με την k υψηλότερη βαθμολογία. Αλλάζοντας την παράμετρο 'score_func' μπορεί να εφαρμοστεί η μέθοδος τόσο για δεδομένα ταξινόμησης όσο και για δεδομένα παλινδρόμησης. Η επιλογή των καλύτερων χαρακτηριστικών είναι σημαντική διαδικασία όταν προετοιμάζει κανείς ένα μεγάλο σύνολο δεδομένων για εκπαίδευση. Βοηθά να εξαλειφθεί το λιγότερο σημαντικό μέρος των δεδομένων και να μειωθεί ο χρόνος εκπαίδευσης.

  Οι δύο παράμετροι της SelectKBest που μελετήθηκαν είναι οι score_func και το k, όπως ήδη αναφέρθηκαν. Να σημειωθεί ότι οι συναρτήσεις που μπορούν να χρησιμοποιηθούν ως παράμετροι διαφέρουν μεταξύ προβλημάτων ταξινόμησης και παλινδρόμησης. Εδώ μελετά ένα πρόβλημα ταξινόμησης, επομένως θα έχουμε τις εξής επιλογές :

  → ***score_func***: οι πιθανές συναρτήσεις είναι οι chi2, f_classif, mutual_info_classif. Το chi2 υπολογίζει τα στατιστικά χ-τετράγωνα μεταξύ κάθε μη αρνητικού χαρακτηριστικού και κλάσης. Η βαθμολογία αυτή μπορεί να χρησιμοποιηθεί για την επιλογή n χαρακτηριστικών με τις υψηλότερες τιμές για το στατιστικό χ-τετράγωνο, η οποία πρέπει να περιέχει μόνο μη αρνητικά χαρακτηριστικά. Υπενθυμίζεται ότι το τεστ χ-τετράγωνο μετρά την εξάρτηση μεταξύ στοχαστικών μεταβλητών, επομένως η χρήση αυτής της συνάρτησης «εξαλείφει» τα χαρακτηριστικά που είναι πιο πιθανό να είναι ανεξάρτητα από την τάξη και επομένως άσχετα για ταξινόμηση.

    Οι μέθοδοι που βασίζονται στο F-test υπολογίζουν τον βαθμό γραμμικής εξάρτησης μεταξύ δύο τυχαίων μεταβλητών. Από την άλλη πλευρά, οι μέθοδοι αμοιβαίας πληροφόρησης μπορούν να συλλάβουν κάθε είδους στατιστική εξάρτηση, αλλά επειδή είναι μη παραμετρικές, απαιτούν περισσότερα δείγματα για ακριβή εκτίμηση.

  → ***k***: ένας ακέραιος αριθμός που καθορίζει τα καλύτερα χαρακτηριστικά που θα διατηρηθούν για μελέτη στη συνέχεια.

- **Extra Trees Classifier:** ο Extra Trees Classifier είναι ένας τύπος εκμάθησης συνόλου (Ensemble Method) που συγκεντρώνει τα αποτελέσματα πολλαπλών αποσυσχετισμένων δέντρων αποφάσεων τα οποία συλλέγονται σε ένα «δάσος» για να εξάγει το αποτέλεσμα ταξινόμησής του. Στην ιδέα, μοιάζει πολύ με έναν ταξινομητή Random Forest και διαφέρει από αυτόν μόνο στον τρόπο κατασκευής των δέντρων απόφασης στο δάσος.

Κάθε Δέντρο Απόφασης στο Δάσος Extra Trees κατασκευάζεται από το αρχικό δείγμα εκπαίδευσης. Στη συνέχεια, σε κάθε κόμβο δοκιμής, κάθε δέντρο παρέχεται με ένα τυχαίο δείγμα k χαρακτηριστικών από το σύνολο χαρακτηριστικών, από το οποίο κάθε δέντρο απόφασης πρέπει να επιλέξει το καλύτερο χαρακτηριστικό για να χωρίσει τα δεδομένα με βάση κάποια μαθηματικά κριτήρια (συνήθως τον δείκτη Gini). Αυτό το τυχαίο δείγμα χαρακτηριστικών οδηγεί στη δημιουργία πολλαπλών αποσυσχετισμένων δέντρων αποφάσεων.
Για να πραγματοποιηθεί η επιλογή χαρακτηριστικών χρησιμοποιώντας την παραπάνω δασική δομή, κατά την κατασκευή του δάσους, για κάθε χαρακτηριστικό, υπολογίζεται συνήθως ο Δείκτης Gini αν αποφασιστεί χρησιμοποιηθεί ο Δείκτης Gini στην κατασκευή του δάσους. Αυτή η τιμή ονομάζεται Gini Importance του χαρακτηριστικού. Για την εκτέλεση της επιλογής χαρακτηριστικών, κάθε χαρακτηριστικό ταξινομείται με φθίνουσα σειρά σύμφωνα με τη Gini Importance και κανείς επιλέγει τα κορυφαία k χαρακτηριστικά που επιθυμεί.
Στην παρούσα εργασία επιλέχθηκαν τα 20 κορυφαία χαρακτηριστικά με βάση τη Gini Importance.

- **Recursive Feature Elimination:** δεδομένου ενός εξωτερικού εκτιμητή που εκχωρεί τα βάρη σε χαρακτηριστικά (π.χ. τους συντελεστές ενός γραμμικού μοντέλου), ο στόχος της αναδρομικής εξάλειψης χαρακτηριστικών (RFE) είναι η επιλογή χαρακτηριστικών εξετάζοντας αναδρομικά όλο και μικρότερα σύνολα χαρακτηριστικών. Πρώτον, ο εκτιμητής εκπαιδεύεται στο αρχικό σύνολο χαρακτηριστικών και η σημαντικότητα κάθε χαρακτηριστικού λαμβάνεται μέσω οποιουδήποτε συγκεκριμένου μέτρου(όπως coef_, feature_importances_). Στη συνέχεια, τα λιγότερο σημαντικά χαρακτηριστικά απορρίπτονται από το τρέχον σύνολο χαρακτηριστικών. Αυτή η διαδικασία επαναλαμβάνεται επανειλημμένα στο μειωμένο σύνολο έως ότου επιτευχθεί τελικά ο επιθυμητός αριθμός χαρακτηριστικών προς επιλογή.
  Το RFECV(με Cross Validation) είναι μια εναλλακτική μορφή η οποία εκτελεί RFE σε βρόχο διασταυρούμενης επικύρωσης για να βρει τον βέλτιστο αριθμό χαρακτηριστικών.

- **Select From Model:** το SelectFromModel είναι ένας μετα-μετασχηματιστής που μπορεί να χρησιμοποιηθεί μαζί με οποιονδήποτε εκτιμητή που αποδίδει σημαντικότητα σε κάθε χαρακτηριστικό μέσω ενός συγκεκριμένου χαρακτηριστικού(όπως coef_, feature_importances_) ή μέσω ενός importance_getter που μπορεί να κληθεί μετά την τοποθέτηση. Τα χαρακτηριστικά θεωρούνται ασήμαντα και αφαιρούνται εάν η αντίστοιχη σημασία των τιμών των χαρακτηριστικών είναι κάτω από την παρεχόμενη παράμετρο κατωφλίου. Εκτός από τον αριθμητικό προσδιορισμό του ορίου, υπάρχουν ενσωματωμένα ευρετικά για την εύρεση ενός ορίου χρησιμοποιώντας ένα όρισμα συμβολοσειράς. Τα διαθέσιμα ευρετικά είναι «mean», «median» και πολλαπλάσια από αυτά όπως «0.1*mean». Σε συνδυασμό με τα κριτήρια κατωφλίου, μπορεί κανείς να χρησιμοποιήσει την

παράμετρο max_features για να ορίσει ένα όριο στον αριθμό των χαρακτηριστικών που θα επιλέξει.
Στη συγκεκριμένη εργασία κι ύστερα από δοκιμές με αρκετούς εκτιμητές, αποφασίστηκε η χρήση του LassoCV: το Lasso(**L**east **A**bsolute **S**hrinkage and **S**election **O**perator) είναι ένα γραμμικό μοντέλο που υπολογίζει αραιούς συντελεστές. Είναι χρήσιμο σε ορισμένα περιβάλλοντα λόγω της τάσης του να προτιμά λύσεις με λιγότερους μη μηδενικούς συντελεστές, μειώνοντας ουσιαστικά τον αριθμό των χαρακτηριστικών από τα οποία εξαρτάται η δεδομένη λύση. Μαθηματικά, αποτελείται από ένα γραμμικό μοντέλο με έναν πρόσθετο όρο κανονικοποίησης(regularization).
Η συνάρτηση προς ελαχιστοποίηση είναι:

$$\frac{1}{2n_{samples}} \|Xw - y\|_2^2 + a\|w\|_1$$

Η εκτίμηση Lasso λύνει έτσι την ελαχιστοποίηση της ποινής των ελαχίστων τετραγώνων με πρόσθετο το $a\|w\|_1$, όπου a είναι μια σταθερά και $\|w\|_1$ είναι η l_1-νόρμα του διανύσματος συντελεστή.
Η παράμετρος a ελέγχει τον βαθμό αραιότητας των εκτιμώμενων συντελεστών.
Η βιβλιοθήκη scikit-learn δίνει τη δυνατότητα με τη συνάρτηση LassoCV να αναζητηθούν και εν τέλει να επιλεγούν οι καταλληλότερες τιμές για την παράμετρο a με τη μέθοδο cross – validation.

3 *Μελέτη Χαρακτηριστικών*

3.1 Δεδομένα

Το NBA (National Basketball Association)[15] είναι η μεγαλύτερη και σημαντικότερη ιστορικά ομοσπονδία καλαθοσφαίρισης στον πλανήτη. Πλέον, με την ταχύτατη ανάπτυξη της βιομηχανίας των sports analytics, υπάρχουν πολλά διαθέσιμα δεδομένα, στατιστικά και πληροφορίες που αφορούν τόσο τις ομάδες όσο και τον κάθε αθλητή ξεχωριστά που αγωνίζεται ή αγωνιζόταν στο πρωτάθλημα.

Είναι χρήσιμο, πριν την περαιτέρω ανάλυση των δεδομένα που χρησιμοποιήθηκαν, να αναφερθούν κάποιες βασικές πληροφορίες για τη γενικότερη διάρθρωση του πρωταθλήματος, όπως τον αριθμό των ομάδων, το πρόγραμμα των ομάδων κατά τη διάρκεια της σεζόν και το πλήθος των αγώνων. Πιο συγκεκριμένα:

- Απαρτίζεται από 30 (29 με έδρα στις ΗΠΑ και μία με έδρα στον Καναδά).
- Για την καλύτερη διεξαγωγή του πρωταθλήματος οι ομάδες έχουν χωριστεί σε δύο περιφέρειες, την Ανατολική και τη Δυτική. Αυτές με την σειρά τους χωρίζονται σε τρία μικρότερα γκρουπ ομάδων (5). Η Ανατολική Περιφέρεια χωρίζεται στις Atlantic (Ατλαντική), Central (Κεντρική) και Southeast (Νοτιοανατολική) ενώ η Δυτική στις Southwest (Νοτιοδυτική), Northwest (Βορειοδυτική) και Pacific (Ειρηνικού).

Διεξαγωγή του πρωταθλήματος:

- Κανονική περίοδος: κατά τη διάρκεια της κανονικής περιόδου, κάθε ομάδα παίζει 82 παιχνίδια, 41 εντός και ισάριθμα εκτός. Μια ομάδα αντιμετωπίζει αντιπάλους στη δική της υποπεριφέρεια τέσσερις φορές το χρόνο (16 παιχνίδια). Κάθε ομάδα παίζει με έξι από τις ομάδες των άλλων δύο υποπεριφερειών στην περιφέρειά της τέσσερις φορές (24 παιχνίδια) και με τις υπόλοιπες τέσσερις ομάδες τρεις φορές (12 παιχνίδια). Τέλος, κάθε ομάδα παίζει με όλες τις ομάδες από την άλλη περιφέρεια δύο φορές (30 παιχνίδια). Αυτή η ασύμμετρη δομή σημαίνει ότι η ισχύς του προγράμματος θα διαφέρει μεταξύ των ομάδων (αλλά όχι τόσο σημαντικά). Περί τα μέσα Απριλίου, τελειώνει η κανονική περίοδος. Κατά τη διάρκεια αυτής της περιόδου αρχίζει η ψηφοφορία για ατομικά βραβεία, καθώς και η επιλογή των τιμητικών ομάδων μετά το πέρας της σεζόν.
- Playoffs: Τα πλέι-οφ του NBA ξεκινούν τον Απρίλιο μετά το τέλος της κανονικής περιόδου με τις οκτώ κορυφαίες ομάδες σε κάθε περιφέρεια (σύνολο 16). Τα πλέι-οφ ακολουθούν τη μορφή τουρνουά. Κάθε ομάδα παίζει με έναν

αντίπαλο σε μια σειρά από επτά αγώνες(best-of-seven), με την πρώτη ομάδα που κερδίζει τέσσερα παιχνίδια να προκρίνεται στον επόμενο γύρο, ενώ η άλλη ομάδα αποκλείεται από τα πλέι-οφ. Στον επόμενο γύρο, η νικήτρια ομάδα παίζει ενάντια σε μία άλλη ομάδα που προκρίθηκε από την ίδια περιφέρεια. Όλες εκτός από μία ομάδα σε κάθε περιφέρεια αποκλείονται από τα πλέι-οφ. Ο τελευταίος γύρος των πλέι-οφ, μεταξύ των νικητών και των δύο περιφερειών, είναι γνωστός ως «Τελικοί του NBA» και διεξάγεται κάθε χρόνο τον Ιούνιο.

Στην παρούσα εργασία το αρχικό dataset είναι το NBA games data από το website https://www.kaggle.com. Το συγκεκριμένο dataset αποτελείται από 5 μικρότερα(υπό τη μορφή csv αρχείων) τα οποία παρέχουν πολλή και χρήσιμη πληροφορία με τα στατιστικά όλων των παιχνιδιών της κανονικής διάρκειας του πρωταθλήματος(χωρίς τους αγώνες των playoffs), τα στατιστικά και τα προσωπικά στοιχεία των παικτών για αυτά τα παιχνίδια, βαθμολογία των ομάδων καθώς επίσης και κάποια βασικά στοιχεία των ομάδων. Ένας βασικός λόγος που προτιμήθηκε αυτό το dataset είναι ότι καλύπτει αναλυτικά όλες τις σεζόν από το 2004 έως και το Δεκέμβριο του 2020.

Από αυτά τα αρχικά δεδομένα επιλέχθηκαν οι 12 πιο πρόσφατες σεζόν, δηλαδή η σεζόν 2007-2008 μέχρι και τη σεζόν 2018-2019 η οποία ήταν η τελευταία που ολοκληρώθηκε κανονικά πριν την έξαρση της πανδημίας Covid-19 στις Ηνωμένες Πολιτείες. Στη συνέχεια, μετά από αρκετές προσαρμογές και τροποποιήσεις, δημιουργήθηκε ένα τελικό σύνολο δεδομένων το οποίο περιείχε πολλά από τα χαρακτηριστικά που προϋπήρχαν αλλά και πολλά νέα χαρακτηριστικά που δημιουργήθηκαν για κάθε αγώνα. Μερικά από τα νέα χαρακτηριστικά είναι το ρεκόρ νικών/ηττών της κάθε ομάδας πριν τον αγώνα, μέσοι όροι βασικών στατιστικών κατηγοριών(ευστοχία σουτ, ασσίστ, ριμπάουντ κλπ.) για κάθε ομάδα στα τελευταία παιχνίδια που έχει δώσει. Επιπρόσθετα, με τους κατάλληλους συνδυασμούς στα δεδομένα, δημιουργήθηκαν νέα χαρακτηριστικά με τις διαφορές των στατιστικών της κάθε ομάδας, όπως η βαθμολογική διαφορά, η διαφορά στα ποσοστά ευστοχίας κλπ. Τέλος, προστέθηκαν και οι στοιχηματικές αποδόσεις για τον κάθε αγώνα(νίκη γηπεδούχου, ήττα γηπεδούχου), όπως δίνονταν από τις βασικές εταιρίες στοιχηματισμού των ΗΠΑ, καθώς και η διαφορά των δύο αυτών αποδόσεων. Σε αυτό το σημείο αξίζει να αναφερθεί κι η εργασία του Chenjie Cao[16] η οποία αποτέλεσε χρήσιμη έμπνευση όσον αφορά τη δημιουργία νέων χαρακτηριστικών.

Μια ακόμη επιλογή που έγιν ήταν η αφαίρεση των πρώτων δύο μηνών της κάθε αγωνιστικής περιόδου και ο λόγος για την απόφαση αυτή είναι διττός :

1. Πολλά από τα χαρακτηριστικά που δημιουργήθηκαν αποτυπώνουν τη συμπεριφορά των ομάδων στα τελευταία 3, 5 ή 7 παιχνίδια που έχουν δώσει, επομένως δε θα είχε νόημα να συμπεριληφθούν και οι προηγούμενοι αγώνες.
2. Εμπειρικά, η ιστορία έχει δείξει ότι οι ομάδες πιθανώς έχουν μη φυσιολογικά αποτέλεσματα στην αρχή της κάθε σεζόν ερχόμενες από τις διακοπές του καλοκαιριού.

Στη συνέχεια παρατίθενται τα τελικά χαρακτηριστικά που επιλέχθηκαν για την παρούσα εργασία :

ID	Σύντομη Περιγραφή
1	Τρέχον πλήθος αγώνων της γηπεδούχου ομάδας
2	Τρέχον ποσοστό νικών της γηπεδούχου ομάδας

3	Τρέχον ρεκόρ νικών εντός έδρας της γηπεδούχου ομάδας
4	Τρέχον ρεκόρ νικών εκτός έδρας της γηπεδούχου ομάδας
5	Τρέχον πλήθος αγώνων της φιλοξενούμενης ομάδας
6	Τρέχον ποσοστό νικών της φιλοξενούμενης ομάδας
7	Τρέχον ρεκόρ νικών εντός έδρας της φιλοξενούμενης ομάδας
8	Τρέχον ρεκόρ νικών εκτός έδρας της φιλοξενούμενης ομάδας
9	Ποσοστό νικών της γηπεδούχου ομάδας την προηγούμενη σεζόν
10	Ρεκόρ νικών εντός έδρας της γηπεδούχου ομάδας την προηγούμενη σεζόν
11	Ρεκόρ νικών εκτός έδρας της γηπεδούχου ομάδας την προηγούμενη σεζόν
12	Ποσοστό νικών της φιλοξενούμενης ομάδας την προηγούμενη σεζόν
13	Ρεκόρ νικών εντός έδρας της φιλοξενούμενης ομάδας την προηγούμενη σεζόν
14	Ρεκόρ νικών εκτός έδρας της φιλοξενούμενης ομάδας την προηγούμενη σεζόν
15	Ποσοστό νικών της γηπεδούχου ομάδας στα τελευταία 3 παιχνίδια
16	Σύνολο πόντων που σκόραρε η γηπεδούχος ομάδα στα τελευταία 3 παιχνίδια
17	Ποσοστό εύστοχων σουτ της γηπεδούχου ομάδας στα τελευταία 3 παιχνίδια
18	Ποσοστό εύστοχων ελευθέρων βολών της γηπεδούχου ομάδας στα τελευταία 3 παιχνίδια
19	Ποσοστό εύστοχων σουτ τριών πόντων της γηπεδούχου ομάδας στα τελευταία 3 παιχνίδια
20	Σύνολο ασίστ της γηπεδούχου ομάδας στα τελευταία 3 παιχνίδια
21	Σύνολο ριμπάουντ της γηπεδούχου ομάδας στα τελευταία 3 παιχνίδια
22	Ποσοστό νικών της φιλοξενούμενης ομάδας στα τελευταία 3 παιχνίδια
23	Σύνολο πόντων που σκόραρε η φιλοξενούμενη ομάδα στα τελευταία 3 παιχνίδια
24	Ποσοστό εύστοχων σουτ της φιλοξενούμενης ομάδας στα τελευταία 3 παιχνίδια
25	Ποσοστό εύστοχων ελευθέρων βολών της φιλοξενούμενης ομάδας στα τελευταία 3 παιχνίδια
26	Ποσοστό εύστοχων σουτ τριών πόντων της φιλοξενούμενης ομάδας στα τελευταία 3 παιχνίδια
27	Σύνολο ασίστ της φιλοξενούμενης ομάδας στα τελευταία 3 παιχνίδια
28	Σύνολο ριμπάουντ της φιλοξενούμενης ομάδας στα τελευταία 3 παιχνίδια
29	Ποσοστό νικών της γηπεδούχου ομάδας στα τελευταία 7 παιχνίδια
30	Σύνολο πόντων που σκόραρε η γηπεδούχος ομάδα στα τελευταία 7 παιχνίδια
31	Ποσοστό εύστοχων σουτ της γηπεδούχου ομάδας στα τελευταία 7 παιχνίδια
32	Ποσοστό εύστοχων ελευθέρων βολών της γηπεδούχου ομάδας στα τελευταία 7 παιχνίδια
33	Ποσοστό εύστοχων σουτ τριών πόντων της γηπεδούχου ομάδας στα τελευταία 7 παιχνίδια
34	Σύνολο ασίστ της γηπεδούχου ομάδας στα τελευταία 7 παιχνίδια
35	Σύνολο ριμπάουντ της γηπεδούχου ομάδας στα τελευταία 7 παιχνίδια
36	Μέσος όρος πόντων που σκόραρε η γηπεδούχος ομάδα στα τελευταία 5 παιχνίδια ως γηπεδούχος
37	Μέσος όρος εύστοχων σουτ της γηπεδούχου ομάδας στα τελευταία 5 παιχνίδια ως γηπεδούχος
38	Μέσος όρος εύστοχων ελευθέρων βολών της γηπεδούχου ομάδας στα τελευταία 5 παιχνίδια ως γηπεδούχος
39	Μέσος όρος εύστοχων σουτ τριών πόντων της γηπεδούχου ομάδας στα τελευταία 5 παιχνίδια ως γηπεδούχος
40	Μέσος όρος ασίστ της γηπεδούχου ομάδας στα τελευταία 5 παιχνίδια ως γηπεδούχος

41	Μέσος όρος ριμπάουντ της γηπεδούχου ομάδας στα τελευταία 5 παιχνίδια ως γηπεδούχος
42	Μέσος όρος πόντων που σκόραρε η γηπεδούχος ομάδα στα τελευταία 5 παιχνίδια ως φιλοξενούμενος
43	Μέσος όρος εύστοχων σουτ της γηπεδούχου ομάδας στα τελευταία 5 παιχνίδια ως φιλοξενούμενος
44	Μέσος όρος εύστοχων ελευθέρων βολών της γηπεδούχου ομάδας στα τελευταία 5 παιχνίδια ως φιλοξενούμενος
45	Μέσος όρος εύστοχων σουτ τριών πόντων της γηπεδούχου ομάδας στα τελευταία 5 παιχνίδια ως φιλοξενούμενος
46	Μέσος όρος ασίστ της γηπεδούχου ομάδας στα τελευταία 5 παιχνίδια ως φιλοξενούμενος
47	Μέσος όρος ριμπάουντ της γηπεδούχου ομάδας στα τελευταία 5 παιχνίδια ως φιλοξενούμενος
48	Ποσοστό νικών της φιλοξενούμενης ομάδας στα τελευταία 7 παιχνίδια
49	Σύνολο πόντων που σκόραρε η φιλοξενούμενη ομάδα στα τελευταία 7 παιχνίδια
50	Ποσοστό εύστοχων σουτ της φιλοξενούμενης ομάδας στα τελευταία 7 παιχνίδια
51	Ποσοστό εύστοχων ελευθέρων βολών της φιλοξενούμενης ομάδας στα τελευταία 7 παιχνίδια
52	Ποσοστό εύστοχων σουτ τριών πόντων της φιλοξενούμενης ομάδας στα τελευταία 7 παιχνίδια
53	Σύνολο ασίστ της φιλοξενούμενης ομάδας στα τελευταία 7 παιχνίδια
54	Σύνολο ριμπάουντ της φιλοξενούμενης ομάδας στα τελευταία 7 παιχνίδια
55	Μέσος όρος πόντων που σκόραρε η φιλοξενούμενη ομάδα στα τελευταία 5 παιχνίδια ως γηπεδούχος
56	Μέσος όρος εύστοχων σουτ της φιλοξενούμενης ομάδας στα τελευταία 5 παιχνίδια ως γηπεδούχος
57	Μέσος όρος εύστοχων ελευθέρων βολών της φιλοξενούμενης ομάδας στα τελευταία 5 παιχνίδια ως γηπεδούχος
58	Μέσος όρος εύστοχων σουτ τριών πόντων της φιλοξενούμενης ομάδας στα τελευταία 5 παιχνίδια ως γηπεδούχος
59	Μέσος όρος ασίστ της φιλοξενούμενης ομάδας στα τελευταία 5 παιχνίδια ως γηπεδούχος
60	Μέσος όρος ριμπάουντ της φιλοξενούμενης ομάδας στα τελευταία 5 παιχνίδια ως γηπεδούχος
61	Μέσος όρος πόντων που σκόραρε η φιλοξενούμενη ομάδα στα τελευταία 5 παιχνίδια ως φιλοξενούμενος
62	Μέσος όρος εύστοχων σουτ της φιλοξενούμενης ομάδας στα τελευταία 5 παιχνίδια ως φιλοξενούμενος
63	Μέσος όρος εύστοχων ελευθέρων βολών της φιλοξενούμενης ομάδας στα τελευταία 5 παιχνίδια ως φιλοξενούμενος
64	Μέσος όρος εύστοχων σουτ τριών πόντων της φιλοξενούμενης ομάδας στα τελευταία 5 παιχνίδια ως φιλοξενούμενος
65	Μέσος όρος ασίστ της φιλοξενούμενης ομάδας στα τελευταία 5 παιχνίδια ως φιλοξενούμενος
66	Μέσος όρος ριμπάουντ της φιλοξενούμενης ομάδας στα τελευταία 5 παιχνίδια ως φιλοξενούμενος
67	Πλήθος καλύτερων παικτών της γηπεδούχου ομάδας στον αγώνα με βάση το EFF(Player Efficiency Rating)

68	Συνολικό Efficiency της γηπεδούχου ομάδας στον αγώνα
69	Πλήθος απουσιών για τη γηπεδούχο ομάδα
70	Πλήθος παιχνιδιών την τελευταία εβδομάδα για τη γηπεδούχο ομάδα
71	Πλήθος παιχνιδιών εντός έδρας την τελευταία εβδομάδα για τη γηπεδούχο ομάδα
72	Πλήθος παιχνιδιών εκτός έδρας την τελευταία εβδομάδα για τη γηπεδούχο ομάδα
73	Πλήθος back to back παιχνιδιών την τελευταία εβδομάδα για τη γηπεδούχο ομάδα
74	ELO Rating της γηπεδούχου ομάδας
75	Πλήθος καλύτερων παικτών της φιλοξενούμενης ομάδας στον αγώνα με βάση το EFF(Player Efficiency Rating)
76	Συνολικό Efficiency της φιλοξενούμενης ομάδας στον αγώνα
77	Πλήθος απουσιών για τη φιλοξενούμενη ομάδα
78	Πλήθος παιχνιδιών την τελευταία εβδομάδα για τη φιλοξενούμενη ομάδα
79	Πλήθος παιχνιδιών εντός έδρας την τελευταία εβδομάδα για τη φιλοξενούμενη ομάδα
80	Πλήθος παιχνιδιών εκτός έδρας την τελευταία εβδομάδα για τη φιλοξενούμενη ομάδα
81	Πλήθος back to back παιχνιδιών την τελευταία εβδομάδα για τη φιλοξενούμενη ομάδα
82	ELO Rating της φιλοξενούμενης ομάδας
83	36-55
84	42-61
85	37-56
86	43-62
87	38-57
88	44-63
89	39-58
90	45-64
91	40-59
92	46-65
93	41-60
94	47-66
95	67-75
96	68-76
97	69-77
98	70-78
99	71-79
100	72-80
101	73-81
102	74-82
103	15-22
104	16-23
105	17-24
106	18-25
107	19-26
108	20-27
109	21-28
110	29-48
111	30-49
112	31-50
113	32-51
114	33-52

115	34-53
116	35-54
117	9-12
118	10-13
119	11-14
120	2-6
121	3-7
122	4-8
123	3-8
124	10-14
125	Μέγιστη απόδοση νίκης της γηπεδούχου ομάδας
126	Μέγιστη απόδοση νίκης της φιλοξενούμενης ομάδας
127	125-126
128	Τελικό αποτέλεσμα αγώνα (1 για νίκη γηπεδούχου, 0 για νίκη φιλοξενούμενου)

Από τον παραπάνω πίνακα χαρακτηριστικών αξίζει να επισημανθούν κάποιες μετρικές απόδοσης των ομάδων που υπολογίστηκαν εκ των υστέρων με τη βοήθεια των διαθέσιμων κύριων στατιστικών κατηγοριών του κάθε αγώνα. Αυτές είναι:

1. ELO Rating

Αρχικά πρέπει να αναφερθούν ορισμένα βασικά στοιχεία. Οι Elo[17] βαθμολογίες εξαρτώνται μόνο από το τελικό σκορ κάθε παιχνιδιού και από το πού παίχτηκε (πλεονέκτημα έδρας). Περιλαμβάνουν παιχνίδια κανονικής περιόδου και πλέι οφς. Οι ομάδες κερδίζουν πάντα πόντους Elo αφού κερδίζουν παιχνίδια και χάνουν έδαφος αφότου χάσουν παιχνίδια. Κερδίζουν περισσότερους βαθμούς για νίκες ανατροπής και για νίκη με μεγαλύτερη διαφορά. Για παράδειγμα όταν οι Ντένβερ Νάγκετς κέρδισαν 30 πόντους Elo ανατρέποντας τους No. 1 σε κατάταξη Σιάτλ Super Sonics στον πρώτο γύρο των πλέι οφς του NBA το 1994, οι Σόνικς έχασαν 30 πόντους. Οι αξιολογήσεις καθορίζονται ανά παιχνίδι και όχι ανά σεζόν. Έτσι, μπορούν να διαπιστωθούν αλλαγές στη «φόρμα» μιας ομάδας κατά τη διάρκεια του έτους. Η μακροπρόθεσμη μέση βαθμολογία Elo είναι 1500, αν και μπορεί να διαφέρει ελαφρώς σε κάθε συγκεκριμένο έτος με βάση το πόσο πρόσφατα επεκτάθηκε το πρωτάθλημα. Περισσότερο από το 90 τοις εκατό των αξιολογήσεων των ομάδων είναι μεταξύ 1300 (αρκετά κακό) και 1700 (πολύ καλό).

Επίσης υπάρχουν μερικές παράμετροι που διαφοροποιούνται για το πρωτάθλημα του NBA. Πιο συγκεκριμένα, ο παράγοντας K του Elo καθορίζει πόσο γρήγορα αντιδρά η βαθμολογία στα νέα αποτελέσματα του παιχνιδιού. Θα πρέπει να ρυθμιστεί έτσι ώστε να λαμβάνει αποτελεσματικά τα νέα δεδομένα αλλά να μην αντιδρά υπερβολικά σε αυτά. (Με μια πιο τεχνική έννοια, ο στόχος είναι να ελαχιστοποιηθεί η αυτοσυσχέτιση.) Εάν το K οριστεί πολύ ψηλά, οι βαθμολογίες θα αυξηθούν. Εάν τεθεί πολύ χαμηλά, το Elo θα χρειαστεί πολύ χρόνο για να αναγνωρίσει σημαντικές αλλαγές στην ποιότητα της ομάδας. Έχει βρεθεί ότι το βέλτιστο K για το NBA είναι 20. Είναι στο ίδιο εύρος με το K που χρησιμοποιείται για τις βαθμολογίες του πρωταθλήμτος αμερικανικού ποδοσφαίρου NFL και διεθνών πρωταθλημάτων ποδοσφαίρου, παρόλο που στο NBA παίζονται πολύ περισσότερα παιχνίδια σε σχέση με αυτά τα αθλήματα. Το γεγονός υπονοεί ότι πρέπει να δωθεί σχετικά μεγάλο βάρος στην πρόσφατη απόδοση μιας ομάδας. Ένας τρόπος για να ερμηνευτεί αυτό είναι ότι τα δεδομένα του NBA υπόκεινται σε σχετικά μικρή τυχαιότητα. Αυτό το κάνει διαφορετικό από

αθλήματα όπως το μπέιζμπολ και το χόκεϊ, καθώς σε αυτά τα αθλήματα, η προεπιλεγμένη υπόθεση θα πρέπει να είναι ότι ένα σερί νικών ή ήττων είναι κυρίως τύχη. Αυτό δεν ισχύει τόσο για το μπάσκετ. Τα σερί μπορεί να αντικατοπτρίζουν αληθινές, ίσως προσωρινές, αλλαγές στην ποιότητα της ομάδας. Αν μια ομάδα έκανε ένα σερί 19 νικών κάποια σεζόν, για παράδειγμα, θα ήταν αναμφίβολα λίγο τυχεροί, αλλά μάλλον ήταν πιο δύσκολο να ηττηθούν από ό,τι σε άλλες στιγμές εκείνης της σεζόν.

Υπάρχουν ακόμη ορισμένες περιπτώσεις στις οποίες το Elo φαίνεται πολύ αργό για να φτάσει στην πραγματικότητα, όπως όταν ο Μάικλ Τζόρνταν έφυγε από τους Μπουλς ή ο ΛεΜπρόν Τζέιμς από τους Καβς. Αλλά ο Elo κοιτάζει μόνο τα σκορ των αγώνων και όχι τη σύνθεση του ρόστερ. Εάν αυτές είναι όλες οι πληροφορίες που υπάρχουν, η ρύθμιση του Elo ώστε να αντιδρά πιο γρήγορα σε αυτές τις περιπτώσεις θα το έκανε να αντιδράσει υπερβολικά σε άλλες.

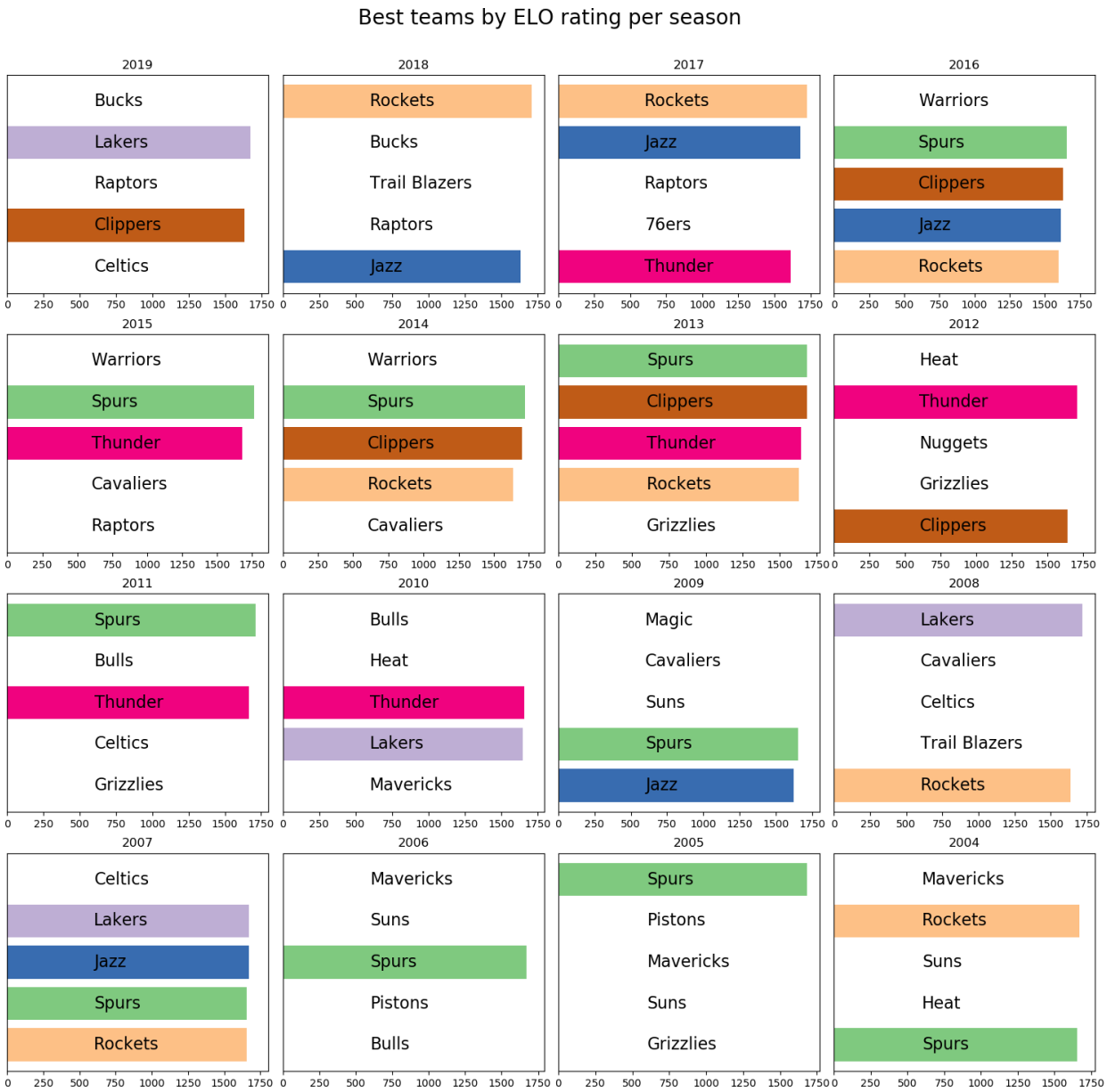
Το πλεονέκτημα έδρας ορίζεται ως ισοδύναμο με 100 βαθμούς αξιολόγησης Elo. Εκατό πόντοι Elo ισοδυναμούν με περίπου 3,5 πόντους NBA, έτσι είναι σαν να λέμε ότι η γηπεδούχος ομάδα θα ευνοείται με 3 ή 4 πόντους εάν οι ομάδες είχαν κατά τα άλλα ίσα στατιστικά. Στην πράξη, το μέγεθος του πλεονεκτήματος εντός έδρας έχει αυξηθεί και εξασθενίσει στην ιστορία του NBA. Ορισμένες ομάδες (ειδικά αυτές όπως το Ντένβερ και η Γιούτα που παίζουν σε μεγάλα υψόμετρα) είχαν ιστορικά ελαφρώς μεγαλύτερα πλεονεκτήματα εντός έδρας.

Το Elo επιτυγχάνει μια καλή ισορροπία μεταξύ των συστημάτων αξιολόγησης που αντιπροσωπεύουν το περιθώριο νίκης και αυτών που δεν το κάνουν. Ενώ οι ομάδες κερδίζουν πάντα πόντους Elo μετά από νίκες και χάνουν πόντους Elo μετά από ήττες, κερδίζουν ή χάνουν περισσότερους πόντους όταν υπάρχουν μεγαλύτερα περιθώρια νίκης. Αυτό λειτουργεί αναθέτοντας έναν πολλαπλασιαστή σε κάθε παιχνίδι με βάση το τελικό σκορ και διαιρώντας τον με το προβλεπόμενο περιθώριο νίκης μιας ομάδας υπό την προϋπόθεση ότι έχει κερδίσει το παιχνίδι. Για παράδειγμα, το περιθώριο 4 πόντων των Warriors έναντι των Rockets στο Game 1 των τελικών της Δυτικής Περιφέρειας ήταν χαμηλότερο από αυτό που θα περίμενε το Elo για μια νίκη των Warriors. Έτσι, οι Warriors κερδίζουν πόντους Elo, αλλά όχι τόσους όσους, αν είχαν κερδίσει με μεγαλύτερη διαφορά. Ο τύπος υπολογίζει τις φθίνουσες αποδόσεις. Η μετάβαση από μια νίκη 5 πόντων σε μια νίκη 10 πόντων έχει μεγαλύτερη σημασία από τη μετάβαση από μια νίκη 25 πόντων σε μια νίκη 30 πόντων. Ο ακριβής τύπος που χρησιμοποιούμε στην παρούσα διπλωματική είναι : παίρνουμε το περιθώριο νίκης μιας ομάδας, προσθέτουμε 3 πόντους και μετά υψώνουμε το αποτέλεσμα στη δύναμη του 0,8 . Διαιρούμε το αποτέλεσμα με τον ακόλουθο τύπο: 7.5+.006*(elo_diff), όπου το elo_diff αντιπροσωπεύει τη διαφορά βαθμολογίας Elo μεταξύ των ομάδων, λαμβάνοντας υπόψη το πλεονέκτημα εντός έδρας. Το Elo_diff θα πρέπει να είναι αρνητικό σε παιχνίδια που κερδίζει το αουτσάιντερ.

Τελευταία και πολύ σημαντική παράμετρος για το σύστημα Elo και κατ' επέκταση για τα παρόντα δεδομένα, είναι η μεταφορά της βαθμολογίας Elo της κάθε ομάδας στην επόμενη σεζόν. Αντί να επαναφέρει τη βαθμολογία κάθε ομάδας όταν ξεκινά μια νέα σεζόν, το Elo μεταφέρει ένα μέρος της βαθμολογίας μιας ομάδας από τη μια σεζόν στην άλλη. Στις βαθμολογίες στο NBA κρατούν τα τρία τέταρτα. Το υψηλότερο κλάσμα αντικατοπτρίζει το γεγονός ότι οι ομάδες του NBA είναι πιο συνεπείς από χρόνο σε χρόνο.

Για παράδειγμα, οι Μαϊάμι Χιτ τελείωσαν τη σεζόν 2012-13 στο NBA με βαθμολογία Elo 1754. Η βαθμολογία Elo της ομάδας για την έναρξη της σεζόν 2013-14 υπολογίζεται ως εξής:

(0.75 * 1754) + (0.25 * 1500)
Η βαθμολογία Elo κάθε ομάδας επανέρχεται στη μέση τιμή και – όπως ειπώθηκε – η μακροπρόθεσμη μέση βαθμολογία Elo είναι 1500.



Πίνακας των 5 κορυφαίων ομάδων στη βαθμολογία Elo ανά σεζόν του συνόλου δεδομένων

Η βαθμολογία ELO φαίνεται σύμφωνη με τα αποτελέσματα των αντίστοιχων σεζόν. Οι πιο σταθερές ομάδες από άποψη σεζόν στο top 5 έχουν το δικό τους χρώμα. Γίνεται προφανές ότι οι Σπερς έχουν κυριαρχήσει τα τελευταία 15 χρόνια και είναι σταθερά μία από τις καλύτερες ομάδες του πρωταθλήματος. Να σημειωθεί ότι η βαθμολογία ELO υπολογίστηκε μόνο με βάση τα αποτελέσματα της κανονικής περιόδου και όχι την επιτυχία των πλέι οφ. Αυτό γίνεται πιο σημαντικό στις επόμενες σεζόν, καθώς οι ομάδες κάνουν διαχείριση φορτίου και έχουν καταλάβει ότι το να έχουν τους παίκτες τους υγιείς και ξεκούραστους στα πλέι οφ είναι πιο σημαντικό από μερικές θέσεις κατάταξης στην κανονική περίοδο.

2. ***Player Efficiency Rating (PER)***

Το μπάσκετ είναι ένα άθλημα όπου ορισμένοι παίκτες μπορούν να έχουν ουσιαστικό αντίκτυπο στην επιτυχία της ομάδας. Επομένως, είναι σημαντικό να ενσωματωθούν αυτές οι πληροφορίες στο σύνολο δεδομένων. Αυτό ακούγεται απλό στη θεωρία, αλλά στην πράξη υπάρχει ένα σημαντικό ζήτημα που πρέπει να αντιμετωπιστεί: δεν υπάρχει συνολική μέτρηση για την αποτελεσματικότητα στο μπάσκετ. Η πιο δημοφιλής είναι η βαθμολογία απόδοσης παίκτη (PER). Είναι η βαθμολογία μπάσκετ all-in-one του John Hollinger, η οποία επιχειρεί να συγκεντρώσει ή να συνοψίσει όλη τη συνεισφορά ενός παίκτη σε έναν αριθμό. Χρησιμοποιώντας έναν λεπτομερή τύπο, ο Hollinger[18] ανέπτυξε ένα σύστημα που βαθμολογεί τη στατιστική απόδοση κάθε παίκτη.

Το PER προσπαθεί να μετρήσει την απόδοση ενός παίκτη ανά λεπτό, προσαρμόζοντας παράλληλα τον ρυθμό. Ο μέσος όρος PER του πρωταθλήματος είναι πάντα 15,00, το οποίο επιτρέπει συγκρίσεις της απόδοσης των παικτών μεταξύ των σεζόν. Το PER λαμβάνει υπόψη κατηγοριες, όπως εύστοχα σουτ εντός πεδιάς, ελεύθερες βολές, τρίποντα, ασίστ, ριμπάουντ, μπλοκ και κλεψίματα και αρνητικά αποτελέσματα, όπως χαμένες βολές, λάθη και προσωπικά φάουλ. Ο τύπος προσθέτει θετικά στατιστικά και αφαιρεί τα αρνητικά μέσω ενός συστήματος στατιστικών τιμών σημείων. Στη συνέχεια, η βαθμολογία για κάθε παίκτη προσαρμόζεται σε βάση ανά λεπτό, ώστε, για παράδειγμα, οι αναπληρωματικοί να μπορούν να συγκριθούν με τους βασικούς στις συζητήσεις για τον χρόνο παιχνιδιού. Προσαρμόζεται και για τον ρυθμό της ομάδας. Στο τέλος, ένας αριθμός συνοψίζει τα στατιστικά επιτεύγματα των παικτών για εκείνη τη σεζόν.

Όλοι οι υπολογισμοί ξεκινούν με αυτό που ονομάζεται μη προσαρμοσμένο PER (uPER). Ο τύπος είναι:

uPER = (1 / MP) *
[3P
+ (2/3) * AST
+ (2 – factor * (team_AST / team_FG)) * FG
+ (FT *0.5 * (1 + (1 – (team_AST / team_FG)) + (2/3) * (team_AST / team_FG)))
- VOP * TOV
- VOP * DRB% * (FGA – FG)
- VOP * 0.44 * (0.44 + (0.56 * DRB%)) * (FTA – FT)
+ VOP * (1 – DRB%) * (TRB – ORB)
+ VOP * DRB% * ORB
+ VOP * STL
+ VOP * DRB% * BLK
- PF * ((lg_FT / lg_PF) – 0.44 * (lg_FTA / lg_PF) * VOP)], όπου

factor = (2 / 3) – (0.5 * (lg_AST / lg_FG)) / (2 * (lg_FG / lg_FT))
VOP = lg_PTS / (lg_FGA – lg_ORB + lg_TOV + 0.44 * lg_FTA)
DRB% = (lg_TRB – lg_ORB) / lg_TRB

Για καλύτερη κατανόηση των τυπών να σημειωθούν τα εξής:
tm, το πρόθεμα, που δείχνει την ομάδα και όχι τον παίκτη.
Lg, το πρόθεμα, που δείχνει το πρωτάθλημα και όχι τον παίκτη.
Min για τον αριθμό των λεπτών που παίχτηκαν.

3P για τον αριθμό των εύστοχων σουτ τριών πόντων που πραγματοποιήθηκαν.
FG για τον αριθμό των εύστοχων σουτ εντός πεδιάς που πραγματοποιήθηκαν.
FT για τον αριθμό των εύστοχων ελεύθερων βολών που έγιναν.
VOP για την αξία της κατοχής (αλλά σε σχέση με το πρωτάθλημα, σε αυτήν την περίπτωση).
RB για τον αριθμό των ριμπάουντ: **ORB** για επιθετικό, **DRB** για αμυντικό, **TRB** για το σύνολο των ριμπάουντ, **RBP** για ποσοστό επιθετικού ή αμυντικού ριμπάουντ.

Μόλις υπολογιστεί το uPER, πρέπει να προσαρμοστεί για τον ρυθμό της ομάδας και να κανονικοποιηθεί στο πρωτάθλημα για να γίνει PER:

- pace adjustment = lg_Pace / team_Pace
- aPER = (pace adjustment) * uPER
- Το τελευταίο βήμα είναι η κανονικοποίηση του aPER. Αρχικά, υπολογίζεται ο μέσος όρος aPER (lg_aPER) του πρωταθλήματος χρησιμοποιώντας τα λεπτά που έχουν παιχθεί ως βάρη. Τελικά, θα έχουμε:
  PER = aPER * (15 / lg_aPER)

Ωστόσο, επειδή στο τελικό dataset δεν είναι διαθέσιμα όλα τα παραπάνω χαρακτηριστικά, χρησιμοποιήθηκε ένας διαφοροποιημένος τύπος υπολογισμού του Efficiency:

Efficiency = Pts + Rebs + Ast + Stl + Blk – (TO + FG_Misses + FT_Misses) / Games_Played

Η φόρμουλα δημιουργήθηκε από τον αθλητικό ρεπόρτερ και στατιστικολόγο του Κάνσας Σίτι Μάρτιν Μάνλεϊ[19].

Ένα πλεονέκτημα σε αυτό το στατιστικό είναι ότι φιλτράρει τα παιχνίδια που παίζονται, καθώς ο αριθμός όλων αυτών των στατιστικών αυξάνεται με περισσότερα παιχνίδια. Αυτό σημαίνει ότι δεν υπάρχει ανάγκη για αυθαίρετο cut-off για τους αγώνες της σεζόν. Χρησιμοποιώντας αυτό το μέτρο απόδοσης προστέθηκαν και τα ακόλουθα χαρακτηριστικά στο σύνολο δεδομένων που παρατίθεται παραπάνω:
1) Παίκτες σε μια ομάδα που είχαν βαθμολογία αποτελεσματικότητας που τους κατέτασσε στους 30 κορυφαίους την περασμένη σεζόν.
2) Ένα μέτρο της αποτελεσματικότητας της κάθε «ομάδας» αθροίζοντας την αποτελεσματικότητα της προηγούμενης σεζόν όλων των παικτών ανά ομάδα κι ανά παιχνίδι.

3.2 Μελέτη της Συσχέτισης μεταξύ των Χαρακτηριστικών

Ένα πολύ χρήσιμο βήμα στην ανάπτυξη των μοντέλων που βασίζονται σε πολλά δεδομένα είναι η κατανόηση των συσχετίσεων που έχουν τα δεδομένα μεταξύ τους. Είναι σημαντικό να εντοπισθούν αυτές οι συσχετίσεις, ώστε να γίνει σωστότερη χρήση των χαρακτηριστικών.

Η συνηθέστερη μέθοδος υπολογισμού της συσχέτισης δύο μεταβλητών X, Y είναι ο συντελεστής Pearson ή αλλίως Συντελεστής r : είναι ο λόγος μεταξύ της συνδιακύμανσης δύο μεταβλητών και του γινομένου των τυπικών αποκλίσεών τους σ_Xκαι σ_Y. Επομένως, είναι ουσιαστικά μια κανονικοποιημένη μέτρηση της συνδιακύμανσης, έτσι ώστε το αποτέλεσμα να

έχει πάντα μια τιμή μεταξύ -1 και 1. Όπως και με την ίδια τη συνδιακύμανση, το μέτρο μπορεί να αντανακλά μόνο μια γραμμική συσχέτιση μεταβλητών και αγνοεί πολλούς άλλους τύπους σχέσεων ή συσχετίσεις.

Μαθηματικά ο συντελεστής r υπολογίζεται ως εξής:

$$r = \frac{\sum_i^n (x_i - \underline{X})(y_i - \underline{Y})}{\sqrt{\sum_i^n (x_i - \underline{X})^2 \sum_i^n (y_i - \underline{Y})^2}}$$

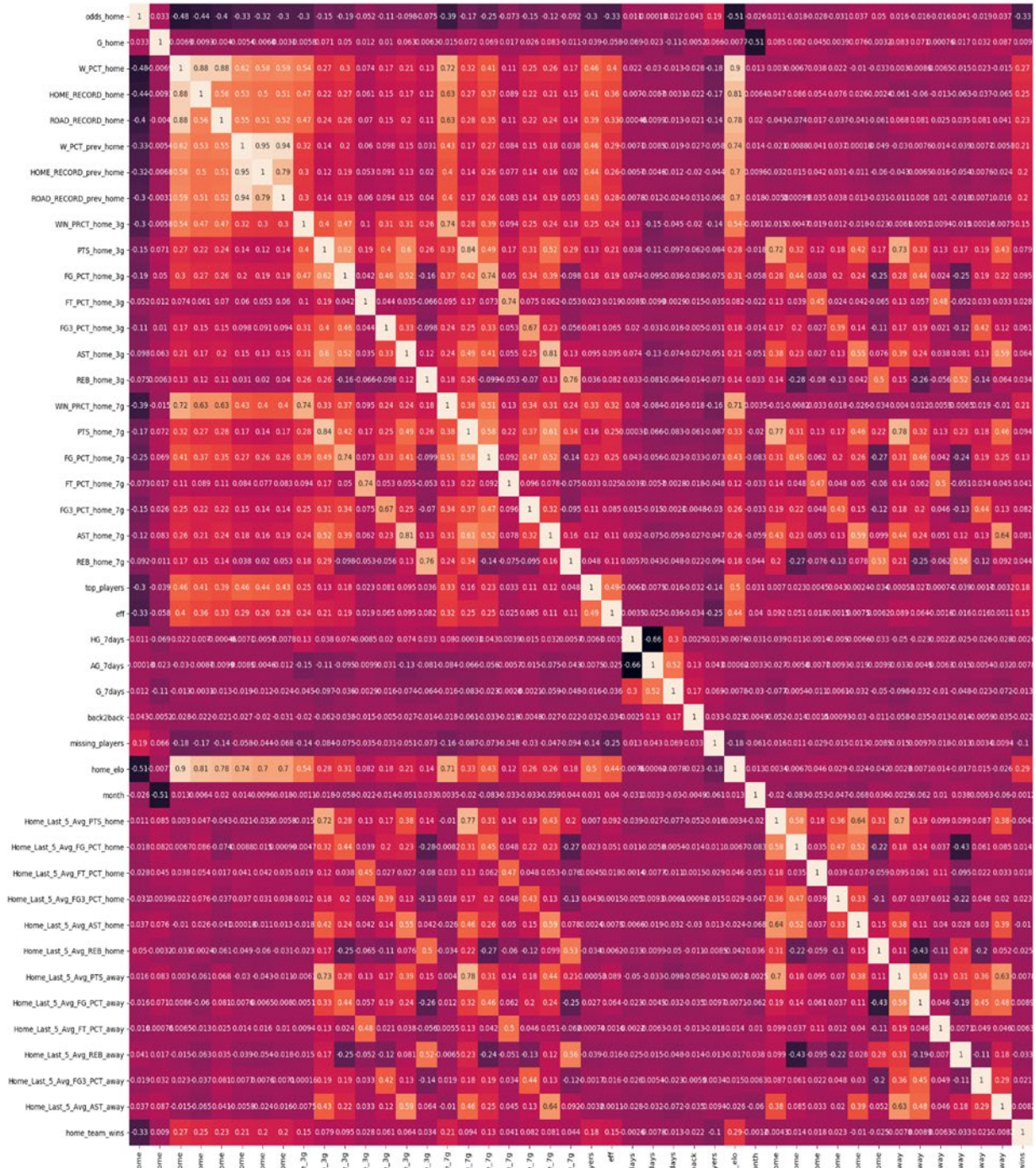
,όπου x_i , y_i είναι κάθε τιμή της μεταβλητής X και Y, ενώ $\underline{X}$, $\underline{Y}$ είναι οι μέσες τιμές των δύο μεταβλητών.

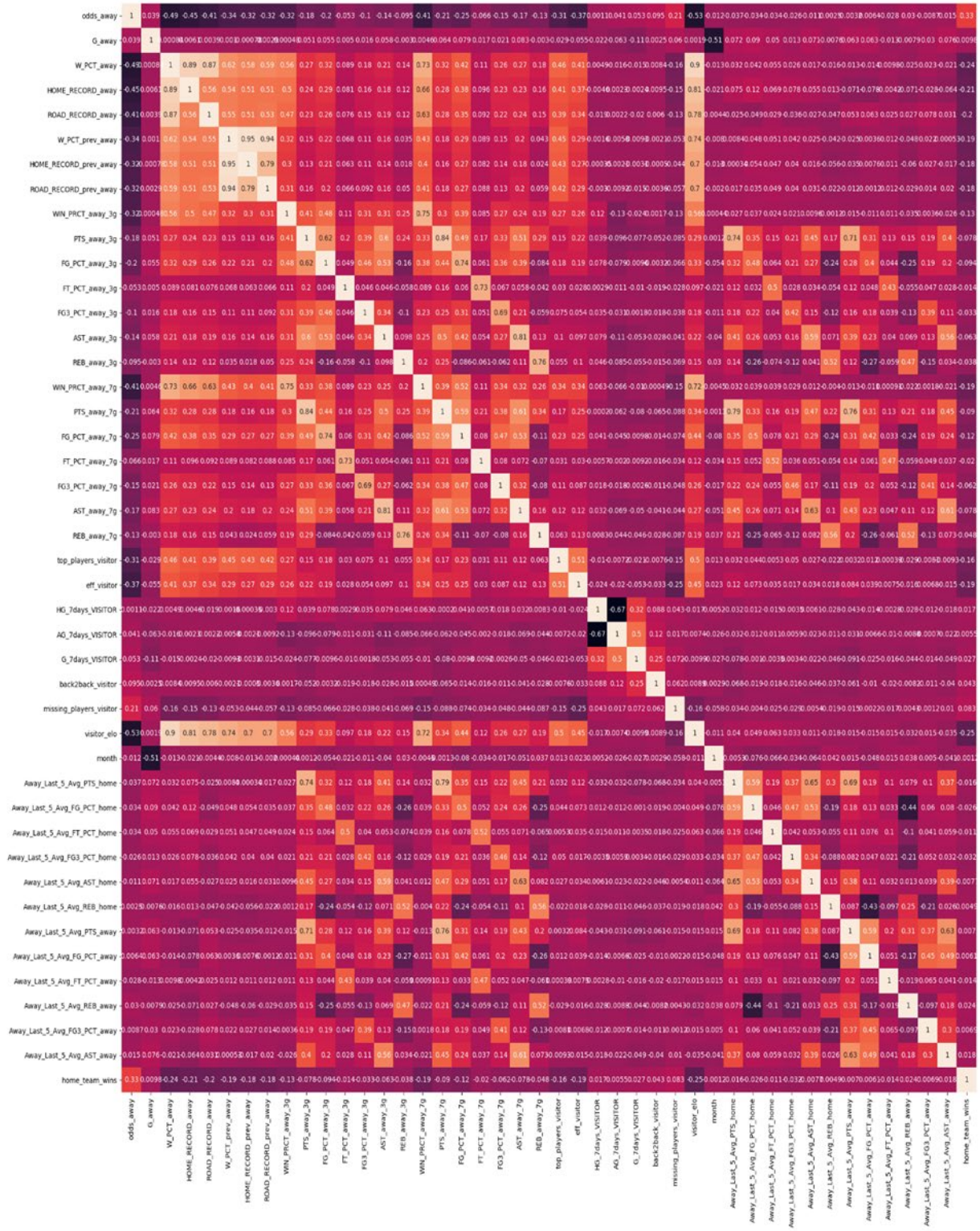
Για τις διάφορες τιμές του r ισχύει:

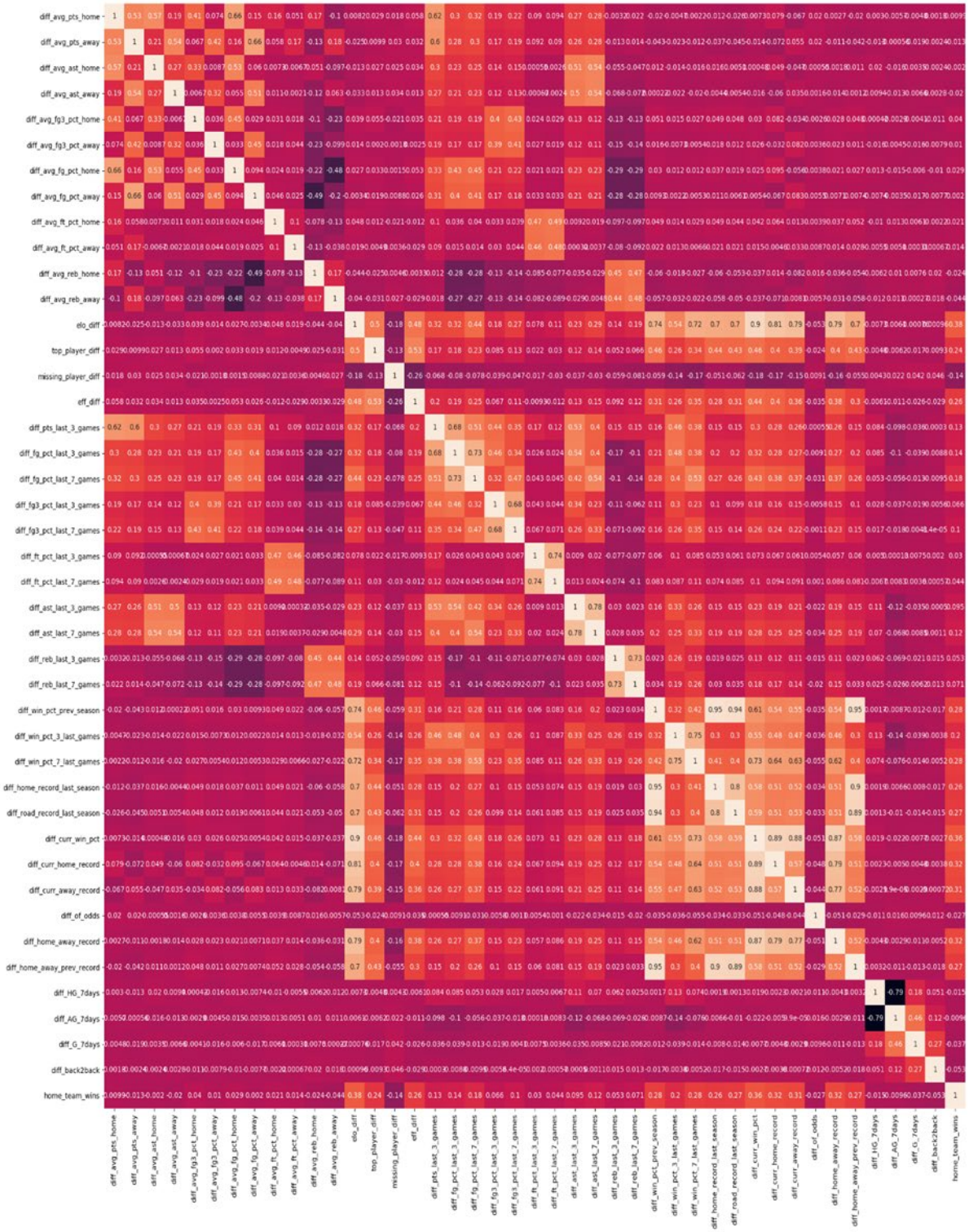
- αν r = ±1, τότε υπάρχει τέλεια θετική ή αρνητική γραμμική συσχέτιση των δεδομένων.
- αν r = 0, τότε τα δεδομένα είναι πλήρως ανεξάρτητα.
- αν r > 0, τότε τα δεδομένα είναι θετικά συσχετισμένα, δηλαδή η αύξηση του ενός συνεπάγεται και αύξηση του άλλου.
- αν r < 0, τότε τα δεδομένα είναι αρνητικά συσχετισμένα, δηλαδή η αύξηση του ενός συνεπάγεται μείωση του άλλου.

Στην παρούσα διπλωματική μάς ενδιαφέρει η συσχέτιση καθενός χαρακτηριστικού με το τελικό αποτέλεσμα του αγώνα(νίκη / ήττα γηπεδούχου). Αντίστοιχη προσοχή πρέπει να δοθεί στην παρατήρηση και των συσχετίσεων των χαρακτηριστικών μεταξύ τους, καθώς τα χαρακτηριστικά εκείνα που είναι αρκετά συσχετισμένα θα προκαλέσουν προβλήματα στην απόδοση των μοντέλων κατά την εκπαίδευσή τους.

Παρακάτω θα παρουσιαστούν οι επιμέρους χάρτες για 3 διαφορετικά σύνολα με το χαρακτηριστικό στόχο, το τελικό αποτέλεσμα. Αυτά τα 3 σύνολα αντιπροσωπεύουν με τη σειρά: τα χαρακτηριστικά που αφορούν **τη γηπεδούχο ομάδα**, **τη φιλοξενούμενη ομάδα** και τα χαρακτηριστικά για **τις διαφορές των δύο ομάδων**. Με σκούρο μωβ χρώμα παρουσιάζονται οι αρνητικές αυτοσυσχετίσεις, ενώ με ανοιχτό ροζ οι θετικές. Το σκούρο χρώμα υποδηλώνει ότι οι δύο μεταβλητές έχουν χαμηλότερη αυτοσυσχέτιση και αντίθετα όσο πιο ανοιχτό χρώμα, τόσο υψηλότερη αυτοσυσχέτιση. Στην τελευταία γραμμή παρουσιάζεται η συσχέτιση κάθε χαρακτηριστικού με το χαρακτηριστικό στόχο (νίκη γηπεδούχου).


odds_home
G_home
W_PCT_home
HOME_RECORD_home
ROAD_RECORD_home
W_PCT_prev_home
HOME_RECORD_prev_home
ROAD_RECORD_prev_home
WIN_PRCT_home_3g
PTS_home_3g
FG_PCT_home_3g
FT_PCT_home_3g
FG3_PCT_home_3g
AST_home_3g
REB_home_3g
WIN_PRCT_home_7g
PTS_home_7g
FG_PCT_home_7g
FT_PCT_home_7g
FG3_PCT_home_7g
AST_home_7g
REB_home_7g
top_players
eff
HG_7days
AG_7days
G_7days
back2back
missing_players
home_elo
month
Home_Last_5_Avg_PTS_home
Home_Last_5_Avg_FG_PCT_home
Home_Last_5_Avg_FT_PCT_home
Home_Last_5_Avg_FG3_PCT_home
Home_Last_5_Avg_AST_home
Home_Last_5_Avg_REB_home
Home_Last_5_Avg_PTS_away
Home_Last_5_Avg_FG_PCT_away
Home_Last_5_Avg_FT_PCT_away
Home_Last_5_Avg_REB_away
Home_Last_5_Avg_FG3_PCT_away
Home_Last_5_Avg_AST_away
home_team_wins


odds_away
G_away
W_PCT_away
HOME_RECORD_away
ROAD_RECORD_away
W_PCT_prev_away
HOME_RECORD_prev_away
ROAD_RECORD_prev_away
WIN_PRCT_away_3g
PTS_away_3g
FG_PCT_away_3g
FT_PCT_away_3g
FG3_PCT_away_3g
AST_away_3g
REB_away_3g
WIN_PRCT_away_7g
PTS_away_7g
FG_PCT_away_7g
FT_PCT_away_7g
FG3_PCT_away_7g
AST_away_7g
REB_away_7g
top_players_visitor
eff_visitor
HG_7days_VISITOR
AG_7days_VISITOR
G_7days_VISITOR
back2back_visitor
missing_players_visitor
visitor_elo
month
Away_Last_5_Avg_PTS_home
Away_Last_5_Avg_FG_PCT_home
Away_Last_5_Avg_FT_PCT_home
Away_Last_5_Avg_FG3_PCT_home
Away_Last_5_Avg_AST_home
Away_Last_5_Avg_REB_home
Away_Last_5_Avg_PTS_away
Away_Last_5_Avg_FG_PCT_away
Away_Last_5_Avg_FT_PCT_away
Away_Last_5_Avg_REB_away
Away_Last_5_Avg_FG3_PCT_away
Away_Last_5_Avg_AST_away
home_team_wins


diff_avg_pts_home
diff_avg_pts_away
diff_avg_ast_home
diff_avg_ast_away
diff_avg_fg3_pct_home
diff_avg_fg3_pct_away
diff_avg_fg_pct_home
diff_avg_fg_pct_away
diff_avg_ft_pct_home
diff_avg_ft_pct_away
diff_avg_reb_home
diff_avg_reb_away
elo_diff
top_player_diff
missing_player_diff
eff_diff
diff_pts_last_3_games
diff_fg_pct_last_3_games
diff_fg_pct_last_7_games
diff_fg3_pct_last_3_games
diff_fg3_pct_last_7_games
diff_ft_pct_last_3_games
diff_ft_pct_last_7_games
diff_ast_last_3_games
diff_ast_last_7_games
diff_reb_last_3_games
diff_reb_last_7_games
diff_win_pct_prev_season
diff_win_pct_3_last_games
diff_win_pct_7_last_games
diff_home_record_last_season
diff_road_record_last_season
diff_curr_win_pct
diff_curr_home_record
diff_curr_away_record
diff_of_odds
diff_home_away_record
diff_home_away_prev_record
diff_HG_7days
diff_AG_7days
diff_G_7days
diff_back2back
home_team_wins

Με τη βοήθεια των παραπάνω πινάκων μπορούν εποπτικά να προκύψουν αρκετά συμπεράσματα για το κάθε χαρακτηριστικό ξεχωριστά, αλλά και για τις υποομάδες χαρακτηριστικών, κοιτώντας σε κάθε χάρτη την τελευταία γραμμή ή στήλη.

Όπως αναμενόταν, για τα χαρακτηριστικά που περιγράφουν τα συμπεριφορά της γηπεδούχου ομάδος, όσο αυξάνουν, πχ όσα περισσότερα εύστοχα σουτ ή ασσίστ έχε η ομάδα ή όσο περισσότερες νίκες έχει στα παιχνίδια της, είναι λογικό το τελικό αποτέλεσμα να παίρνει και χαμηλότερη τιμή, δηλαδή να τείνει προς το 1, πράγμα που αποτυπώνεται με το μωβ χρώμα. Να σημειωθεί εδώ, πως η νίκη γηπεδούχου αντιστοιχεί στον αριθμό 1 και η νίκη του φιλοξενούμενου στο 0. Αντίστοιχα, στα χαρακτηριστικά που περιγράφουν τον φιλοξενούμενο, όπως αναμενόταν η αύξησή τους οδηγεί και σε αύξηση του χαρακτηριστικού του αποτελέσματος, καθώς αυτό δείχνει ότι η φιλοξενούμενη ομάδα είναι αρκετά δυνατή, έτσι είναι λογικό και το αποτέλεσμα να είναι υπέρ της, για το λόγο αυτό και στον χάρτη αποτυπώνονται με ροζ χρώμα. Με όμοια λογική, και η κατηγορία με τις διαφορές των χαρακτηριστικών των δύο ομάδων, καθώς αύξηση της τιμής τους δείχνει υπεροχή του γηπεδούχου και άρα είναι λογικό το αποτέλεσμα να είναι 1 (νίκη γηπεδούχου). Παρατηρεί κανείς εύκολα, ότι η κατηγορία χαρακτηριστικών με τις διαφορές των δύο ομάδων παρουσιάζεται με περισσότερο ανοιχτό χρώμα σε σχέση με τις υπόλοιπες, επομένως, όπως ήταν και αναμενόμενο, αποτελεί και την σημαντικότερη κατηγορία. Ενώ τέλος, χαρακτηριστικά που είναι πανομοιότυπα ή πολύ κοντινά παρουσιάζουν υψηλή συσχέτιση.

Τα κύρια χαρακτηριστικά που ξεχώρισαν από τους χάρτες, αλλά και από τις μεθόδους επιλογής χαρακτηριστικών που πραγματοποιήθηκαν στα μοντέλα, είναι τα ακόλουθα: 2, 3, 6, 7, 74, 76, 82, 96, 102, 125, 126 τα οποία και αποτέλεσαν τη βάση στα μοντέλα μας. Ωστόσο, μετά από μια σειρά δοκιμών που έγιναν, κάποια μοντέλα απέδιδαν καλύτερα με επιπλέον ή και λιγότερα χαρακτηριστικά και έτσι, υπήρξαν κάποιες μετατροπές.

4 *Αποτελέσματα*

4.1 Εισαγωγή στα πειραματικά αποτελέσματα

Στο τρέχον κεφάλαιο εξετάζονται τα πειραματικά μοντέλα και τα αποτελέσματά τους στο τελικό dataset που διαμορφώθηκε με τις μεθόδους που περιγράφηκαν στο προηγούμενο κεφάλαιο.

Τα δεδομένα χωρίστηκαν ως εξής:

• **season 2007-08 έως 2012-13**: αποτελούν το σύνολο δεδομένων εκπαίδευσης (training set) κάθε αλγορίθμου που χρησιμοποιήθηκε.

• **season 2013-14 και 2014-15**: αποτελούν το validation set, όπου κάθε μοντέλο του ίδιου αλγορίθμου, αφού εκπαιδευθεί στο training set, «δοκιμάζεται» σε αυτό το σύνολο δεδομένων. Οι παράμετροι και τα χαρακτηριστικά του μοντέλου με την καλύτερη απόδοση επιλέγονται ώστε να χρησιμοποιηθούν στο τελικό μοντέλο.

• **season 2015-16 έως 2018-19**: αποτελούν το test set, όπου θα δοκιμαστεί η απόδοση του τελικού μοντέλου. Το τελικό μοντέλο προκύπτει από την επιλογή των επικρατέστερων παραμέτρων με εκπαίδευση στο σύνολο δεδομένων της ένωσης του training set και του validation set.

Με στόχο την επίτευξη της μέγιστης δυνατής ακρίβειας στην πρόβλεψη των αποτελεσμάτων των αγώνων υλοποιήθηκαν συνολικά 10 Μηχανικής Μάθησης. Οι 7 έχουν ήδη παρουσιαστεί αναλυτικά σε προηγούμενο κεφάλαιο, οι Gaussian Naive Bayes, k-Nearest Neighbors, Support Vector Machine, Multilayer Perceptron, Random Forest, XGBoost και Recurrent Neural Network με Long Short – Term Memory αρχιτεκτονική. Οι επιπλέον υλοποιήσεις αφορούν Ensemble μεθόδους, πιο συγκεκριμένα υπάρχει ένας Voting ταξινομητής, καθώς επίσης και δύο υλοποιήσεις Stacking ταξινομητή με 2 και 3 στάδια(stages) αντίστοιχα. Όλοι οι αλγόριθμοι μαζί με τα features που χρησιμοποιήθηκαν και τα αποτελέσματά τους θα περιγραφούν παρακάτω.

4.2 Πρωτόκολλο

4.2.1 Classification Report και μετρικές απόδοσης

Είναι μία από τις μεθόδους αξιολόγησης της απόδοσης ενός μοντέλου μηχανικής μάθησης που βασίζεται σε ταξινόμηση. Εμφανίζει το precision(ακρίβεια), το recall(ανάκληση), το f1-score και το support του μοντέλου, μετρικές τις οποίες και θα αναλύσουμε παρακάτω και βοηθούν στην καλύτερη κατανόηση της συνολικής απόδοσης του εκπαιδευμένου μοντέλου μας :

Precision: Η ακρίβεια ορίζεται ως ο λόγος των αληθινών θετικών προς το άθροισμα των αληθών και των ψευδών θετικών.

$$Precision = \frac{True\ Positive}{True\ Positive + False\ Positive}$$

Recall: Η ανάκληση ορίζεται ως ο λόγος των αληθινών θετικών προς το άθροισμα των αληθινών θετικών και των ψευδών αρνητικών.

$$Recall = \frac{True\ Positive}{True\ Positive + False\ Negative}$$

F1-score: Το F1 είναι η σταθμισμένη αρμονική μέση ακρίβεια και ανάκληση. Όσο πιο κοντά είναι η τιμή της βαθμολογίας F1 στο 1.0, τόσο καλύτερη είναι η αναμενόμενη απόδοση του μοντέλου:

$$f1score = 2 \times \frac{Recall\ *\ Precision}{Recall\ +\ Precision}$$

Support: Υποστήριξη είναι ο αριθμός των πραγματικών εμφανίσεων της κλάσης στο σύνολο δεδομένων. Δεν διαφέρει μεταξύ μοντέλων, απλώς διαγιγνώσκει τη διαδικασία αξιολόγησης απόδοσης.

Η βασικότερη μετρική απόδοσης στην παρούσα διπλωματική είναι το **Accuracy**(ορθότητα) το οποίο ορίζεται ως ο λόγος των αληθινών θετικών προς το σύνολο των στοιχείων.

$$Accuracy = \frac{True\ Positive + True\ Negative}{True\ Positive + True\ Negative + False\ Positive + False\ Negative}$$

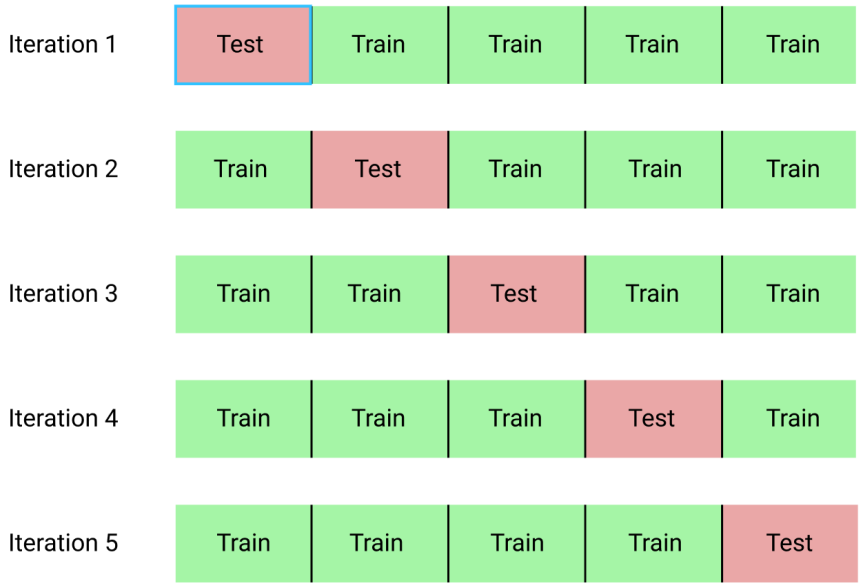
4.2.2 Grid Search (Αναζήτηση πλέγματος)

Η αναζήτηση πλέγματος είναι η διαδικασία σάρωσης των δεδομένων για τη διαμόρφωση των βέλτιστων υπερπαραμέτρων για ένα μοντέλο. Ανάλογα με τον τύπο του μοντέλου που χρησιμοποιείται, απαιτούνται και συγκεκριμένες υπερπαράμετροι. Η αναζήτηση πλέγματος δεν ισχύει μόνο για έναν τύπο μοντέλου, αλλά μπορεί να εφαρμοστεί στο σύνολο των μοντέλων Μηχανικής Μάθησης για τον υπολογισμό των καλύτερων υπερπαραμέτρων που θα χρησιμοποιηθού. Είναι σημαντικό να σημειωθεί ότι η αναζήτηση πλέγματος μπορεί να είναι εξαιρετικά δαπανηρή υπολογιστικά και μπορεί να χρειαστεί πολύς χρόνος για να εκτελεστεί. Θα δημιουργήσει ένα μοντέλο για κάθε δυνατό συνδυασμό υπερπαραμέτρων. Επαναλαμβάνει κάθε συνδυασμό υπερπαραμέτρων και αποθηκεύει ένα μοντέλο για κάθε συνδυασμό.

4.2.3 Cross – Validation

Ο στόχος είναι να αναπτυχθούν μοντέλα για πρόβλεψη σε Νέα Δεδομένα. Ένα καλό μοντέλο δεν είναι αυτό που δίνει ακριβείς προβλέψεις για τα γνωστά δεδομένα ή τα δεδομένα εκπαίδευσης, αλλά αυτό που δίνει καλές προβλέψεις για τα νέα δεδομένα που δεν χρησιμοποιήθηκαν κατά την εκπαίδευση και αποφεύγει το overfitting(υπερπροσαρμογή) και το underfitting(υποπροσαρμογή).

Μια λύση σε αυτά τα προβλήματα προσαρμογής έρχεται να δώσει η τεχνική του cross – validation και πιο συγκεκριμένα η μέθοδος k – Fold Cross Validation :



Παράδειγμα 5-fold cross validation

Source:https://towardsdatascience.com/cross-validation-explained-evaluating-estimator-performance-e51e5430ff85

Η διαδικασία έχει μια ενιαία παράμετρο που ονομάζεται k και αναφέρεται στον αριθμό των ομάδων στις οποίες πρόκειται να χωριστεί ένα δείγμα δεδομένων. Ως εκ τούτου, η διαδικασία ονομάζεται συχνά k-fold cross-validation.

Αν k=5 το σύνολο δεδομένων θα διαιρεθεί σε 5 ίσα μέρη και η παρακάτω διαδικασία θα εκτελεστεί 5 φορές, κάθε φορά με διαφορετικό σετ που δε θα χρησιμοποιηθεί για εκπαίδευση :

1. Ορίζεται μια ομάδα ως test set.

2. Ορίζονται οι υπόλοιπες ομάδες ως σύνολο δεδομένων εκπαίδευσης.

3. Εφαρμόζεται ένα μοντέλο στο σετ εκπαίδευσης και αξιολογείται στη συνέχεια στο test set.

4. Αποθηκεύεται η βαθμολογία αξιολόγησης του μοντέλου.

Στο τέλος της παραπάνω διαδικασίας συνοψίζεται η ικανότητα του μοντέλου χρησιμοποιώντας το δείγμα των βαθμολογιών αξιολόγησης του μοντέλου.

Πώς αποφασίζεται όμως η τιμή του k;

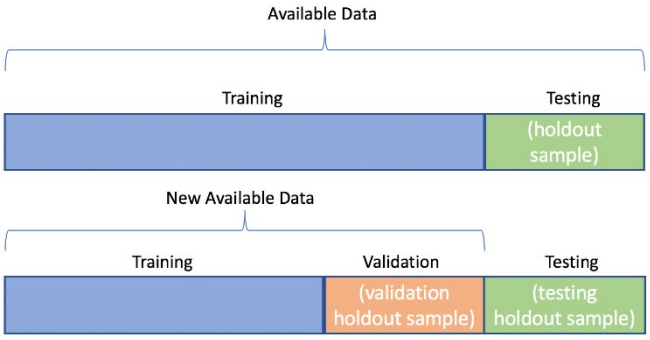
Η τιμή για το k επιλέγεται έτσι ώστε κάθε ομάδα train/test δεδομένων να είναι αρκετά μεγάλη ώστε να είναι στατιστικά αντιπροσωπευτική του ευρύτερου συνόλου δεδομένων. Εάν επιλεγεί μια τιμή για το k που δεν χωρίζει ομοιόμορφα το δείγμα δεδομένων, τότε μια ομάδα θα περιέχει ένα υπόλοιπο από τα παραδείγματα. Είναι προτιμότερο να χωριστεί το δείγμα δεδομένων σε k ομάδες με τον ίδιο αριθμό δειγμάτων, έτσι ώστε το δείγμα των βαθμολογιών του μοντέλου να είναι όλες ισοδύναμες.

Τελικά, η ορθότητα του μοντέλου θα προκύπτει από το μέσο όρο της ορθότητας της κάθε επανάληψης.

4.2.4 Διαχωρισμός σε Train – Validation – Test Sets

Μια εναλλακτική λύση για το διαχωρισμό του dataset είναι η δημιουργία ενός ενδιάμεσου set το οποίο ονομάζεται validation set. Αυτή η επιλογή αποσκοπεί στην ταχύτερη και λιγότερο απαιτητική σε υπολογιστικούς πόρους διαδικασία εκπαίδευσης των μοντέλων.
Η λογική έχει ως εξής :

- το αρχικό training dataset χωρίζεται σε δύο μικρότερα υποσύνολα, ένα καινούριο training dataset και ένα validation set.
- Στη συνέχεια το μοντέλο εκπαιδεύται στο νέο training set και δοκιμάζεται η απόδοσή του στο validation set. Αυτή η διαδικασία επαλήθευσης δίνει την απαραίτητη πληροφορία που χρειάζεται για τη ρύθμιση των υπερπαραμέτρων του εκάστοτε μοντέλου και την εύρεση του καλύτερου δυνατού συνδυασμού αυτών.
- Το τελικό μοντέλο, όπως αυτό προκύπτει από το προηγούμενο βήμα, εκπαιδεύεται στο αρχικό training set(validation + νέο training set) και δοκιμάζεται εκ νέου η απόδοσή του στο test set που ορίσαμε αρχικά.



Πώς λειτουργεί ο διαχωρισμός σε train-validation-test sets. Source:https://algotrading101.com/learn/train-test-split/

Η συνήθης αναλογία του διαχωρισμού στα τρία sets που συναντάται είναι 5:1:3, ωστόσο εξαρτάται κι από άλλους παράγοντες(π.χ. τον όγκο των δεδομένων). Στην παρούσα εργασία επιλέχθηκε η αναλογία train-validation-test να είνα 6:2:4.

4.3 Πειραματικά Αποτελέσματα

Στη συνέχεια θα παρουσιαστούν οι διαφορετικές υλοποιήσεις που εφαρμόστηκαν για τα μοντέλα Μηχανικής Μάθησης. Για κάθε μοντέλο θα περιγραφεί ποια features επιλέχθηκαν, με ποια μέθοδο επιλογής, καθώς επίσης και οποιαδήποτε άλλη επεξεργασία που πραγματοποιήθηκε στο dataset. Να επισημανθεί για μια ακόμη φορά ότι σκοπός είναι η σύγκριση της ορθότητας των προβλέψεων μεταξύ των μοντέλων. Για το λόγο αυτό παρατίθενται αναλυτικά και τα classification reports για κάθε μοντέλο. Επίσης, για κάθε υλοποίηση, παρουσιάζεται και το αντίστοιχο διάγραμμα με τις τιμές Accuracy για καθεμία σεζόν από το test set, καθώς και η τελική μέση τιμή για τις 4 σεζόν που αποτελούν το συνολικό test set. Ακολουθούν οι υλοποιήσεις:

1. Gaussian Naïve Bayes (GNB)

Ο αλγόριθμος έχει περιγραφεί προηγουμένως στο Κεφάλαιο 2. Τα χαρακτηριστικά που επιλέχθηκαν ως επικρατέστερα προέκυψαν από τη μέθοδο SelectFromModel(LassoCV) και είναι: 32, 67, 69, 74, 76, 94, 96, 125, 126. Δεν έγινε κάποια επιπλέον προεπεξεργασία στα τελικά δεδομένα εκπαίδευσης, διότι κατά τις δοκιμές δεν προσέφεραν βελτίωση στην ορθότητα των μοντέλων. Ο αλγόριθμος δε διαθέτει παραμέτρους.

Παρακάτω παρουσιάζεται το confusion matrix των προβλέψεων, η αναφορά του συστήματος για τις 4 επιλεγμένες σεζόν του NBA που αποτελούν το test set, καθώς και η αντίστοιχη για το training set:

	Πρόβλεψη ήττας γηπεδούχου	Πρόβλεψη νίκης γηπεδούχου
Ήττα γηπεδούχου	1181	754
Νίκη γηπεδούχου	738	1910

Training Set

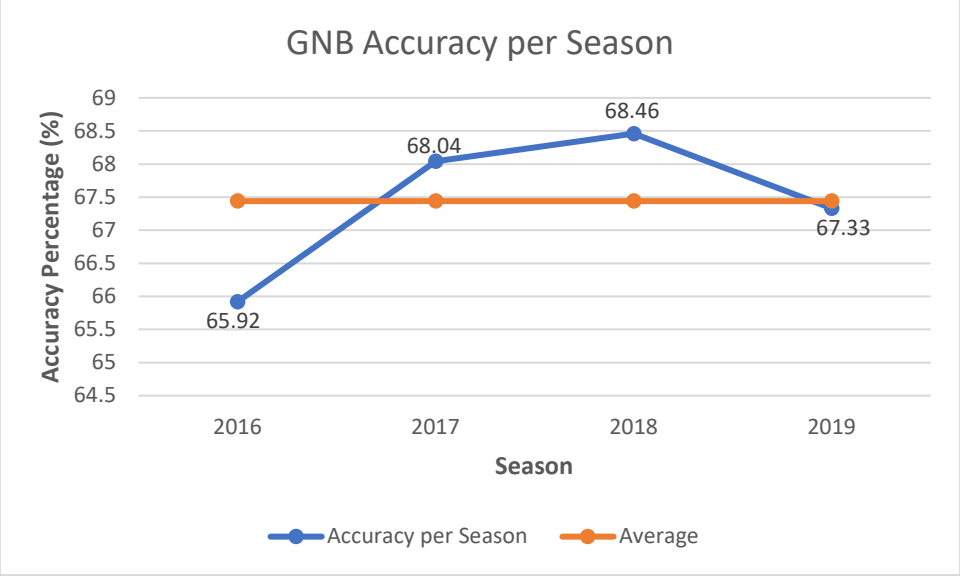
	precision	recall	f1 – score	support
Home loss	0.62	0.60	0.61	3795
Home win	0.73	0.74	0.74	5525
Accuracy			0.68	9320

Test Set

	precision	recall	f1 – score	support
Home loss	0.62	0.61	0.61	1935
Home win	0.72	0.72	0.72	2648
Accuracy			0.67	4583

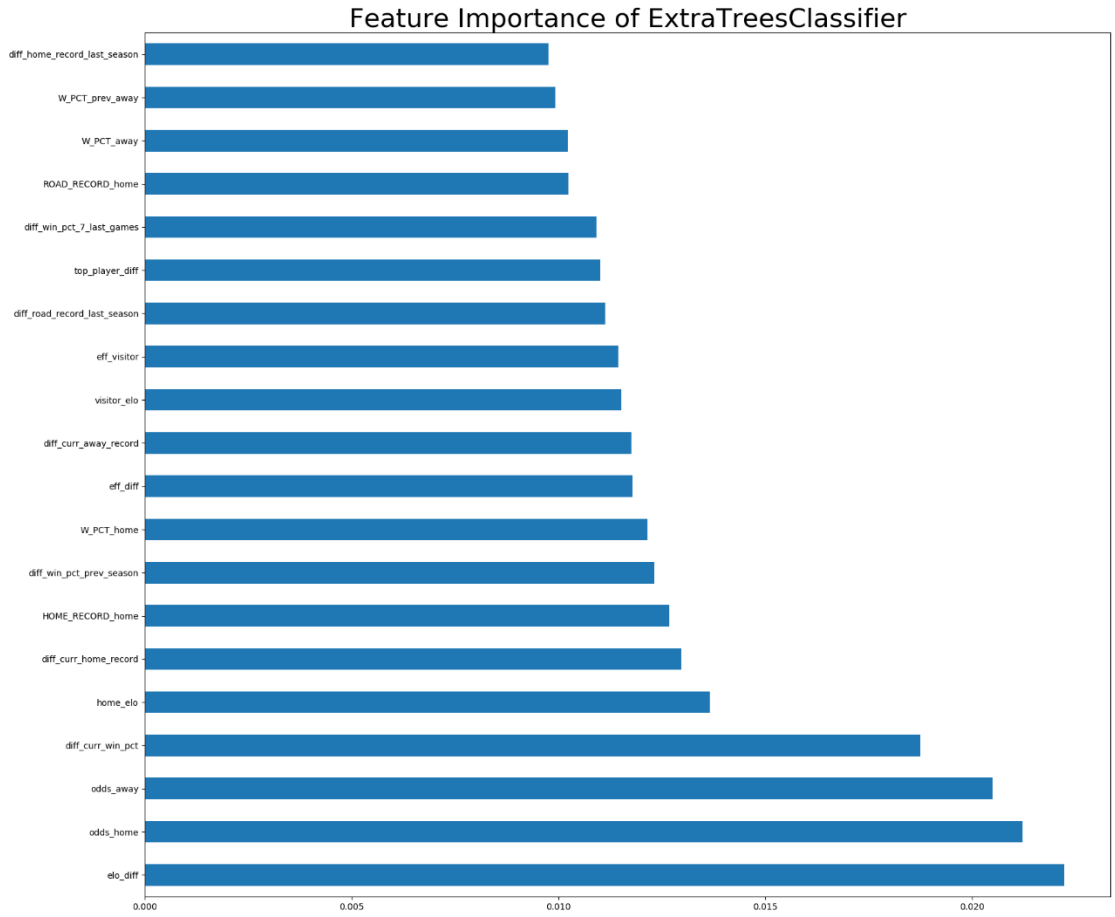
Ο αλγόριθμος, όπως φαίνεται, στο test set επιτυγχάνει υψηλό precision στην κλάση της νίκης γηπεδούχου, δηλαδή όταν ταξινομεί έναν αγώνα σε νίκη γηπεδούχου επαληθεύεται η πρόβλεψη του κατά 72%. Ακόμα, ο αλγόριθμος ανιχνεύει και ταξινομεί σωστά τους αγώνες που ανήκουν στις κλάσεις νίκη γηπεδούχου και νίκη φιλοξενούμενου, γεγονός που αποτυπώνεται και στα αντίστοιχα υψηλά f1-scores.

Η ορθότητα του αλγορίθμου φτάνει στο 67.44% .



2. *k – Nearest Neighbors (kNN)*

Ο αλγόριθμος έχει περιγραφεί προηγουμένως στο Κεφάλαιο 2. Τα επικρατέστερα χαρακτηριστικά που επιλέχθηκαν για το τελικό μοντέλο προέκυψαν από τη μέθοδο Extra Trees Classifier.



Όμως, μετά από αναζήτηση της συσχέτισης μεταξύ των 20 αυτών features, καταλήξαμε στην αφαίρεση μερικών από το σετ προς εκπαίδευση, ενώ εφαρμόστηκε και κανονικοποίηση με χρήση του Standard Scaler. Έτσι, τα τελικά χαρακτηριστικά είναι τα εξής: 2, 6, 74, 76, 82, 95, 96, 110, 117, 125, 126. Επίσης ως τιμή για την παράμετρο k του αλγορίθμου επιλέχθηκε μέσω grid search το 72.

Παρακάτω παρουσιάζεται το confusion matrix, η αναφορά του συστήματος για τις 4 επιλεγμένες σεζόν του NBA που αποτελούν το test set, καθώς και η αντίστοιχη για το training set:

	Πρόβλεψη ήττας γηπεδούχου	Πρόβλεψη νίκης γηπεδούχου
Ήττα γηπεδούχου	991	944
Νίκη γηπεδούχου	585	2063

Training Set

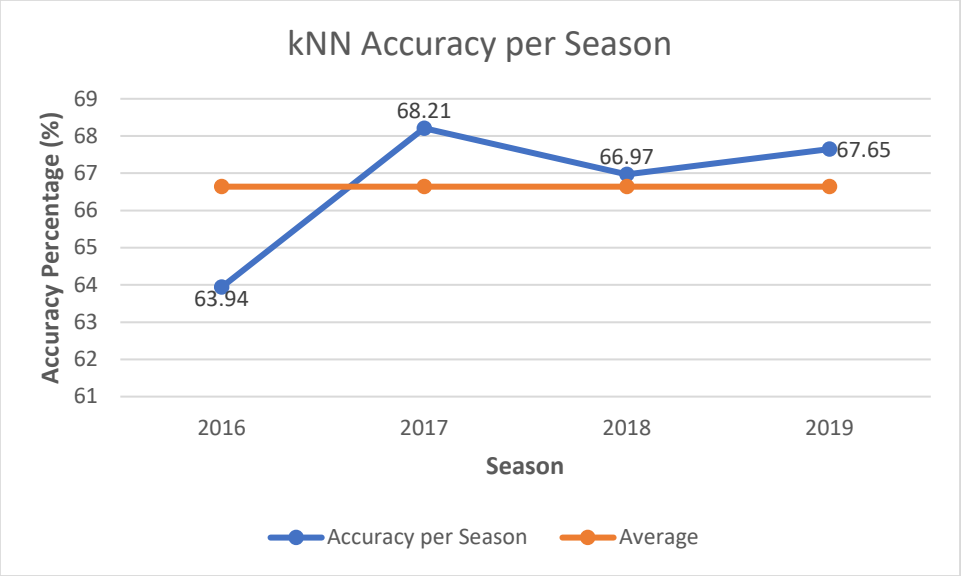
	precision	recall	f1 – score	support
Home loss	0.65	0.55	0.59	3795
Home win	0.72	0.80	0.76	5525
Accuracy			0.70	9320

Test Set

	precision	recall	f1 – score	Support
Home loss	0.63	0.51	0.56	1935
Home win	0.69	0.78	0.73	2648
Accuracy			0.67	4583

Ο συγκεκριμένος αλγόριθμος, για το test set, παρατηρείται να έχει υψηλή απόδοση στις νίκεςγηπεδούχων (0.73 f1-score), καθώς ανιχνεύει και κατηγοριοποιεί σωστά το 78% των αγώνων που έληξαν με νίκη γηπεδούχου. Ακόμα, παρατηρείται ότι η πλειοψηφία των αγώνων κατηγοριοποιείται σε νίκη γηπεδούχου, με ακρίβεια (precision) στην πρόβλεψη νίκη γηπεδούχου 69%. Η νίκη του γηπεδούχου είναι το συχνότερο αποτέλεσμα και αποτελεί περίπου το αποτέλεσμα άνω του 50% των αγώνων, για αυτό και δικαιολογείται η τάση του αλγορίθμου σε πολλές νίκες γηπεδούχου. Ο αλγόριθμος σημείωσε αρκετά υψηλή απόδοση και στην πρόβλεψη νίκη της φιλοξενούμενης ομάδας.

Η ορθότητα του αλγορίθμου φτάνει στο 66.64% .



3. Random Forest

Ο αλγόριθμος έχει περιγραφεί προηγουμένως στο Κεφάλαιο 2. Τα επικρατέστερα χαρακτηριστικά που επιλέχθηκαν για το τελικό μοντέλο προέκυψαν από τη μέθοδο RFECV και είναι: 82, 97, 125, 126. Ως παράμετροι για τον αλγόριθμο επιλέχθηκαν μέσω grid search οι εξής:
bootstrap: True, max_depth: 8, max_features: auto, min_samples_leaf: 4, min_samples_split: 8, n_estimators: 100

Παρακάτω παρουσιάζεται το confusion matrix, η αναφορά του συστήματος για τις 4 επιλεγμένες σεζόν του NBA που αποτελούν το test set, καθώς και η αντίστοιχη για το training set:

	Πρόβλεψη ήττας γηπεδούχου	Πρόβλεψη νίκης γηπεδούχου
Ήττα γηπεδούχου	1056	879
Νίκη γηπεδούχου	608	2040

Training Set

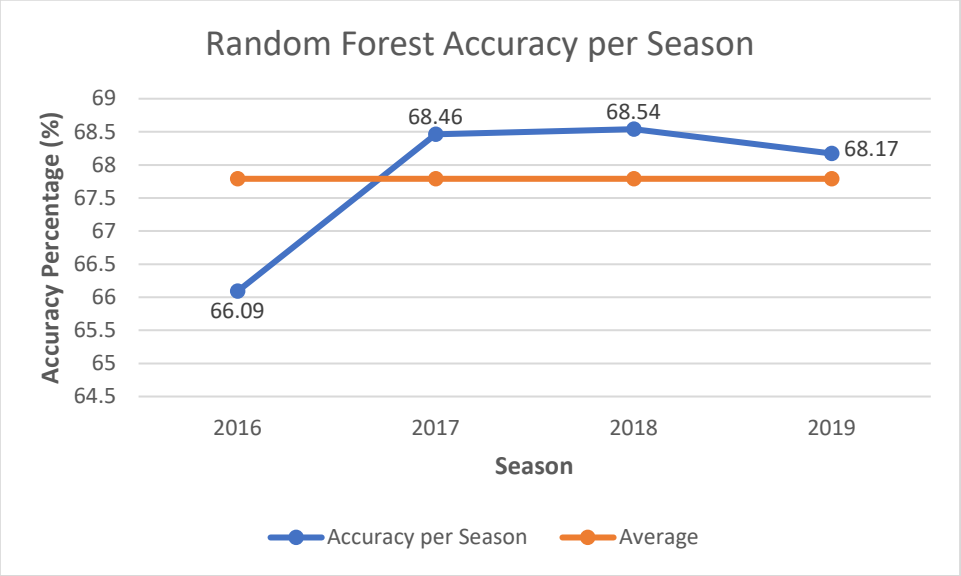
	precision	recall	f1 – score	support
Home loss	0.70	0.59	0.64	3795
Home win	0.74	0.82	0.78	5525
Accuracy			0.73	9320

Test Set

	precision	recall	f1 – score	support
Home loss	0.63	0.55	0.59	1935
Home win	0.70	0.77	0.73	2648
Accuracy			0.68	4583

Παρατηρείται, στο test set, στις δύο κλάσεις ότι και η ακρίβεια, αλλά και η σωστή ανίχνευση τους είναι ιδιαίτερα υψηλά. Πιο συγκεκριμένα, στη νίκη γηπεδούχου το recall είναι 77% και το precision 70%, δίνοντας έτσι ένα πολύ υψηλό f1-score ίσο με 73%. Όμοια και στη νίκη του φιλοξενούμενου, με τα ποσοστά να είναι λίγο πιο χαμηλά από τη νίκη γηπεδούχου. Ο αλγόριθμος δηλαδή δείχνει πολύ ικανοποιητική συμπεριφορά στις νίκες, είτε πρόκειται για τον γηπεδούχο είτε για το φιλοξενούμενο.

Η ορθότητα του αλγορίθμου φτάνει στο 67.55% .



4. Support Vector Machines (SVM)

Ο αλγόριθμος έχει περιγραφεί προηγουμένως στο Κεφάλαιο 2. Τα επικρατέστερα χαρακτηριστικά που επιλέχθηκαν για το τελικό μοντέλο προέκυψαν και πάλι από τη μέθοδο RFECV και είναι: 82, 97, 125, 126 . Ως παράμετροι για τον αλγόριθμο επιλέχθηκαν μέσω grid search οι εξής:
C: 10, gamma: 0.01, kernel: rbf.

Παρακάτω παρουσιάζεται το confusion matrix των προβλέψεων, η αναφορά του συστήματος για τις 4 επιλεγμένες σεζόν του NBA που αποτελούν το test set, καθώς και η αντίστοιχη για το training set:

	Πρόβλεψη ήττας γηπεδούχου	Πρόβλεψη νίκης γηπεδούχου
Ήττα γηπεδούχου	896	1039
Νίκη γηπεδούχου	502	2146

Training Set

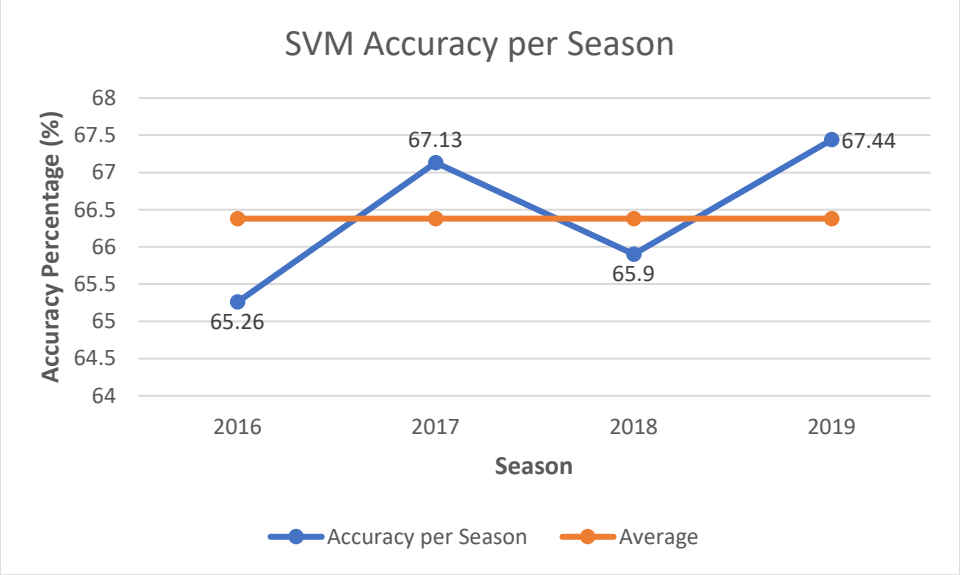
	precision	recall	f1 – score	support
Home loss	0.68	0.49	0.57	3795
Home win	0.71	0.84	0.77	5525
Accuracy			0.70	9320

Test Set

	precision	recall	f1 – score	support
Home loss	0.64	0.46	0.54	1935
Home win	0.67	0.81	0.74	2648
Accuracy			0.67	4583

Παρατηρεί κανείς, στο test set, πως η ακρίβεια στις προβλέψεις και στις δύο κλάσεις είναι αρκετά καλή, 67% για τη νίκη γηπεδούχου και 64% για τη νίκη φιλοξενούμενου. Ο αλγόριθμος εμφανίζει ένα μέτριο ποσοστό (46%) recall στην κλάση νίκης φιλοξενούμενου σε αντίθεση με το recall της κλάσης νίκης γηπεδούχου που είναι πολύ υψηλό (81%). Ο συνδυασμός αυτών των ποσοστών για τις δύο κλάσεις αποτυπώνεται και στα f1 – scores αντίστοιχα.

Η ορθότητα του αλγορίθμου φτάνει στο 66.38% .



5. *Multilayer Perceptron (MLP)*

Ο αλγόριθμος έχει περιγραφεί προηγουμένως στο Κεφάλαιο 2. Τα επικρατέστερα χαρακτηριστικά που επιλέχθηκαν για το τελικό μοντέλο προέκυψαν από τη μέθοδο Extra Trees Classifier, ενώ εφαρμόστηκε και κανονικοποίηση με χρήση του Standard Scaler. Τα τελικά χαρακτηριστικά είναι τα εξής: 2, 6, 74, 76, 82, 95, 96, 110, 117, 125, 126. Ως παράμετροι για τον αλγόριθμο επιλέχθηκαν μέσω grid search οι εξής:
alpha: 1e-05, hidden_layer_sizes: (20, 10, 5), max_iter: 100, solver: sgd .

Παρακάτω παρουσιάζεται το confusion matrix, η αναφορά του συστήματος για τις 4 επιλεγμένες σεζόν του NBA που αποτελούν το test set, καθώς και η αντίστοιχη για το training set:

	Πρόβλεψη ήττας γηπεδούχου	Πρόβλεψη νίκης γηπεδούχου
Ήττα γηπεδούχου	991	944
Νίκη γηπεδούχου	556	2092

Training Set

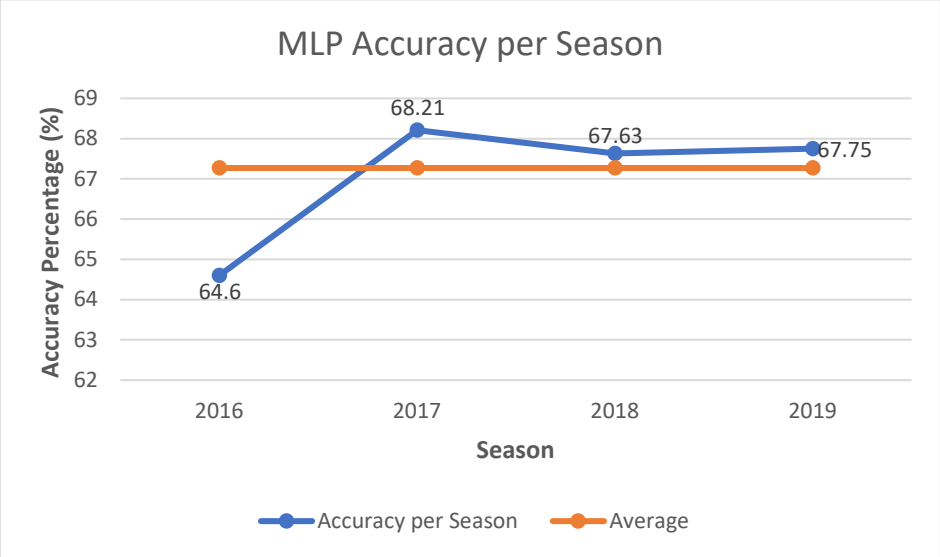
	precision	recall	f1 – score	support
Home loss	0.66	0.52	0.58	3795
Home win	0.71	0.81	0.76	5525
Accuracy			0.69	9320

Test Set

	precision	recall	f1 – score	support
Home loss	0.64	0.51	0.57	1935
Home win	0.69	0.79	0.74	2648
Accuracy			0.67	4583

Ο αλγόριθμος, στο test set, παρουσιάζει ικανοποιητική ακρίβεια στις προβλέψεις του και στις 2 κλάσεις. Ως προς την σωστή ανίχνευση και πρόβλεψη των κλάσεων, το σύστημα τα πηγαίνει ικανοποιητικά και στις 2 κλάσεις. Ειδικότερα, στην κλάση νίκης γηπεδούχου το recall είναι 79% και στην κλάση της νίκης φιλοξενούμενου 51%. Η γενικότερη απόδοση του αλγορίθμου κρίνεται ικανοποιητική με υψηλά f1-score και στις 2 κλάσεις.

Η ορθότητα του αλγορίθμου φτάνει στο 67.27% .



6. *XGBoost*

Ο αλγόριθμος έχει περιγραφεί προηγουμένως στο Κεφάλαιο 2. Τα χαρακτηριστικά που επιλέχθηκαν ως επικρατέστερα προέκυψαν από τη μέθοδο SelectFromModel(LassoCV) και είναι: 32, 67, 69, 74, 76, 94, 96, 125, 126. Ως παράμετροι για τον αλγόριθμο επιλέχθηκαν μέσω grid search οι εξής:
gamma: 0.2, learning_rate: 0.1, max_depth: 3, min_child_weight: 3, n_estimators: 100.

Παρακάτω παρουσιάζεται το confusion matrix των προβλέψεων, η αναφορά του συστήματος για τις 4 επιλεγμένες σεζόν του NBA που αποτελούν το test set, καθώς και η αντίστοιχη για το training set:

	Πρόβλεψη ήττας γηπεδούχου	Πρόβλεψη νίκης γηπεδούχου
Ήττα γηπεδούχου	1063	872
Νίκη γηπεδούχου	604	2044

Training Set

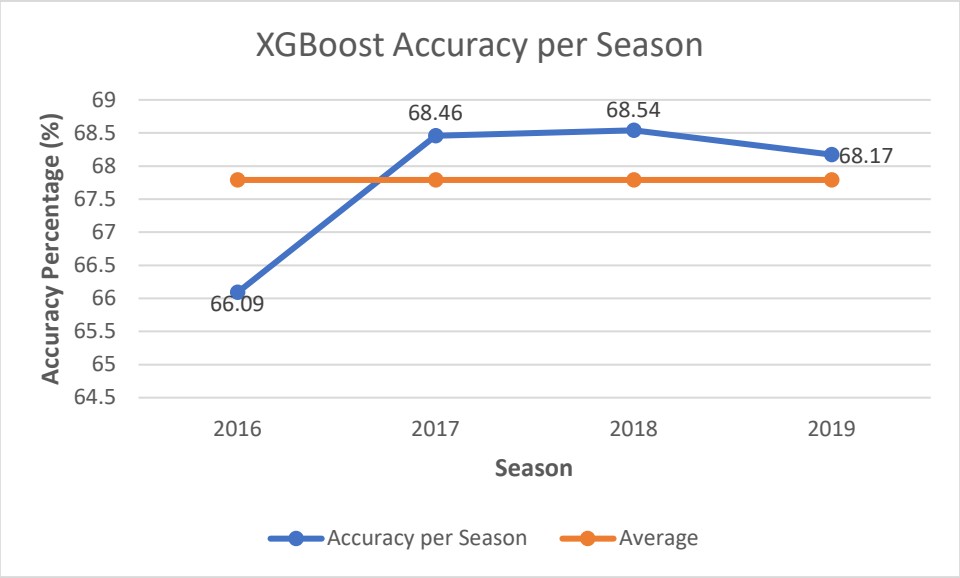
	precision	recall	f1 – score	support
Home loss	0.68	0.56	0.62	3795
Home win	0.73	0.82	0.77	5525
Accuracy			0.71	9320

Test Set

	precision	recall	f1 – score	support
Home loss	0.64	0.55	0.59	1935
Home win	0.70	0.77	0.73	2648
Accuracy			0.68	4583

Ο αλγόριθμος, στο test set, παρουσιάζει αρκετά υψηλή ακρίβεια στις προβλέψεις του και στις 2 κλάσεις. Ως προς την σωστή ανίχνευση και πρόβλεψη των κλάσεων, το σύστημα τα πηγαίνει ικανοποιητικά και στις 2 κλάσεις. Ειδικότερα, στην κλάση νίκης γηπεδούχου το recall είναι 77% και στην κλάση της νίκης φιλοξενούμενου 55%. Η γενικότερη απόδοση του αλγορίθμου κρίνεται ικανοποιητική με υψηλά f1-score και στις 2 κλάσεις.

Η ορθότητα του αλγορίθμου φτάνει στο 67.79% .



7. *Hard Voting Classifier*

Ο αλγόριθμος έχει περιγραφεί προηγουμένως στο Κεφάλαιο 2. Τα χαρακτηριστικά που επιλέχθηκαν ως επικρατέστερα προέκυψαν από τη μέθοδο SelectFromModel(LassoCV) και είναι: 32, 67, 69, 74, 76, 94, 96, 125, 126. Ως εκτιμητές – ψηφοφόροι επιλέχθηκαν, ύστερα από πολλές δοκιμές, τρία από τα παραπάνω μοντέλα με τις παραμέτρους τους, όπως περιγράφηκαν. Αυτά είναι ο XGBoost, ο kNN και ο Random Forest ταξινομητής, ενώ προστέθηκαν και βάρη (2,1,1) αντίστοιχα στην ψήφο του καθενός.

Παρακάτω παρουσιάζεται η αναφορά του συστήματος για τις 4 επιλεγμένες σεζόν του NBA που αποτελούν το test set, καθώς και η αντίστοιχη για το training set:

Training Set

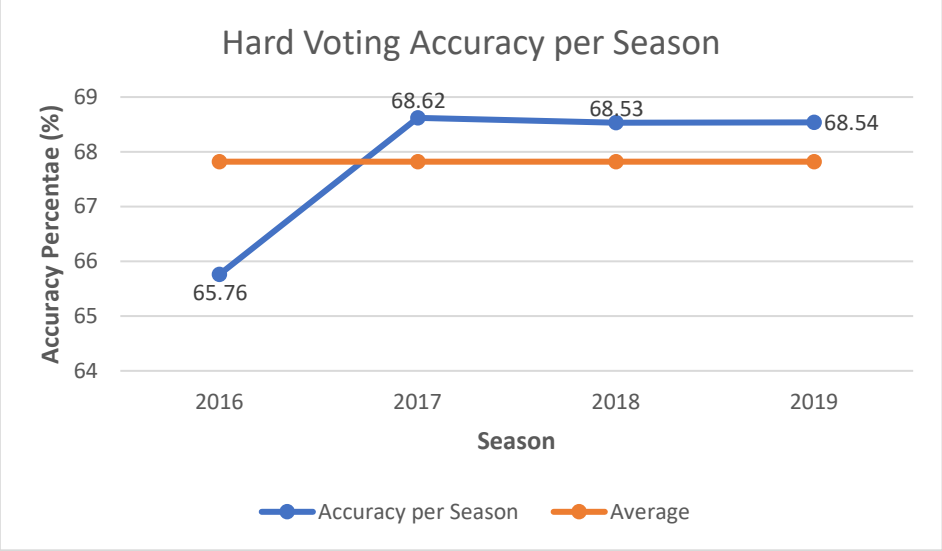
	precision	recall	f1 – score	support
Home loss	0.70	0.62	0.66	3795
Home win	0.76	0.81	0.78	5525
Accuracy			0.73	9320

Test Set

	precision	recall	f1 – score	support
Home loss	0.63	0.57	0.60	1935
Home win	0.71	0.76	0.73	2648
Accuracy			0.68	4583

Παρατηρείται, στο test set, πως ο αλγόριθμος παρουσιάζει ικανοποιητική ακρίβεια στις προβλέψεις του και στις 2 κλάσεις. Ειδικότερα, στην κλάση νίκης γηπεδούχου το recall είναι 76% και στην κλάση της νίκης φιλοξενούμενου 57%. Η συνολική απόδοση του αλγορίθμου κρίνεται ικανοποιητική με υψηλά f1-score στις 2 κλάσεις, 73% για τη νίκη γηπεδούχου και 60% για τη νίκη φιλοξενούμενου αντίστοιχα

Η ορθότητα του αλγορίθμου φτάνει στο 67.82% .



8. 2 – Stage Stacking

Ο αλγόριθμος έχει περιγραφεί προηγουμένως στο Κεφάλαιο 2. Τα χαρακτηριστικά που επιλέχθηκαν ως επικρατέστερα προέκυψαν από τη μέθοδο SelectFromModel(LassoCV) και είναι: 32, 67, 69, 74, 76, 94, 96, 125, 126. Το πρώτο επίπεδο εκτιμητών περιλαμβάνει τα μοντέλα των SVM, XGBoost, Random Forest, GNB που έχουν ήδη περιγραφεί και ως τελικός εκτιμητής προτιμήθηκε το μοντέλο του MLP.

Παρακάτω παρουσιάζεται το confusion matrix, η αναφορά του συστήματος για τις 4 επιλεγμένες σεζόν του NBA που αποτελούν το test set, καθώς και η αντίστοιχη για το training set:

	Πρόβλεψη ήττας γηπεδούχου	Πρόβλεψη νίκης γηπεδούχου
Ήττα γηπεδούχου	1045	890
Νίκη γηπεδούχου	571	2077

Training Set

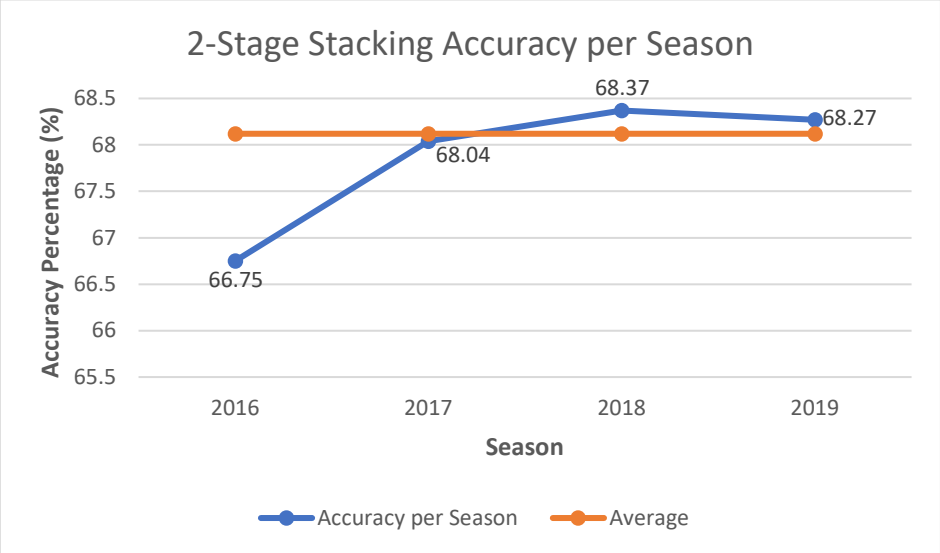
	precision	recall	f1 – score	support
Home loss	0.69	0.57	0.62	3795
Home win	0.74	0.82	0.78	5525
Accuracy			0.72	9320

Test Set

	precision	recall	f1 – score	support
Home loss	0.65	0.54	0.59	1935
Home win	0.70	0.78	0.74	2648
Accuracy			0.68	4583

Παρατηρεί κανείς, στο test set, πως η ακρίβεια στις προβλέψεις και στις δύο κλάσεις είναι αρκετά καλή, 70% για τη νίκη γηπεδούχου και 65% για τη νίκη φιλοξενούμενου. Ο αλγόριθμος εμφανίζει ένα καλό ποσοστό (54%) recall στην κλάση νίκης φιλοξενούμενου σε αντιστοίχιση με το recall της κλάσης νίκης γηπεδούχου που είναι αρκετά υψηλό (78%). Ο συνδυασμός αυτών των ποσοστών για τις δύο κλάσεις αποτυπώνεται και στα f1 – scores αντίστοιχα.

Η ορθότητα του αλγορίθμου φτάνει στο 68.12% .



9. 3 – Stage Stacking

Ο αλγόριθμος έχει περιγραφεί προηγουμένως στο Κεφάλαιο 2. Τα χαρακτηριστικά που επιλέχθηκαν ως επικρατέστερα προέκυψαν από τη μέθοδο SelectFromModel(LassoCV) και είναι: 32, 67, 69, 74, 76, 94, 96, 125, 126. Το πρώτο επίπεδο εκτιμητών περιλαμβάνει τα μοντέλα των SVM, Random Forest που έχουν ήδη περιγραφεί, στο δέυτερο επίπεδο

επιλέχθηκαν τα μοντέλα των GNB, XGBoost και ως τελικός εκτιμητής προτιμήθηκε ένα μοντέλο AdaBoost με learning_rate = 0.1 και n_estimators = 100. Τέλος, τα δεδομένα κανονικοποιήθηκαν με την εφαρμογή του MinMax Scaler ο οποίος βελτίωσε ελαφρώς την τελική τιμή της ορθότητας του αλγορίθμου.

Παρακάτω παρουσιάζεται το confusion matrix, η αναφορά του συστήματος για τις 4 επιλεγμένες σεζόν του NBA που αποτελούν το test set, καθώς και η αντίστοιχη για το training set:

	Πρόβλεψη ήττας γηπεδούχου	Πρόβλεψη νίκης γηπεδούχου
Ήττα γηπεδούχου	1165	770
Νίκη γηπεδούχου	712	1936

Training Set

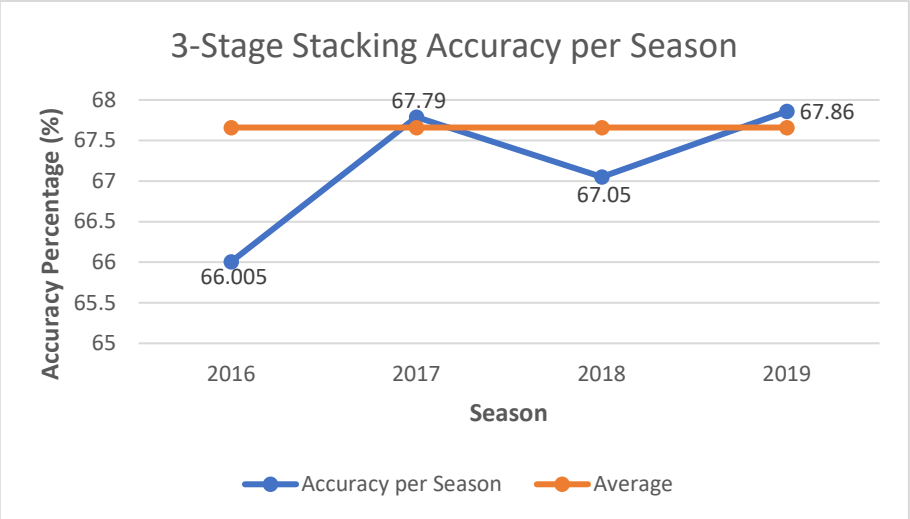
	precision	recall	f1 – score	support
Home loss	0.63	0.63	0.63	3795
Home win	0.75	0.74	0.74	5525
Accuracy			0.70	9320

Test Set

	precision	recall	f1 – score	support
Home loss	0.62	0.60	0.61	1935
Home win	0.72	0.73	0.72	2648
Accuracy			0.68	4583

Παρατηρείται, στο test set, για τον παραπάνω αλγόριθμο, ότι η ακρίβεια στις προβλέψεις και στις δύο κλάσεις είναι αρκετά καλή, 72% για τη νίκη γηπεδούχου και 62% για τη νίκη φιλοξενούμενου. Ο αλγόριθμος εμφανίζει ένα καλό ποσοστό (60%) recall στην κλάση νίκης φιλοξενούμενου όπως και στο recall της κλάσης νίκης γηπεδούχου που κρίνεται υψηλό (73%). Ο συνδυασμός αυτών των ποσοστών για τις δύο κλάσεις αποτυπώνεται και στα f1 – scores αντίστοιχα.

Η ορθότητα του αλγορίθμου φτάνει στο 67.66% .



10. RNN with LSTM

Ο αλγόριθμος έχει περιγραφεί προηγουμένως στο Κεφάλαιο 2. Στο παρόν μοντέλο εισάγονται όλα τα features, χωρίς κάποια περαιτέρω προεπεξεργασία (π.χ. κανονικοποίηση, feature selection methods).

Πριν περάσουμε στην αρχιτεκτονική του παρόντος μοντέλου, αξίζει να αναφέρουμε ορισμένες παραμέτρους/στοιχεία που συνθέτουν συνήθως ένα τεχνητό νευρωνικό δίκτυο, όπως:

- **Sequential API:** η κλάση Sequential προσφέρει έναν τρόπο δημιουργίας μοντέλων βαθιάς μάθησης όπου δημιουργείται ένα μοντέλο της κλάσης, στη συνέχεια δημιουργούνται και προστίθενται σε αυτό επίπεδα μοντέλων. Η βασική ιδέα του Sequential API είναι βασικά η διάταξη των διαφορετικών επιπέδων Keras με διαδοχική σειρά και για αυτό ονομάζεται Sequential API. Το μεγαλύτερο μέρος του ANN έχει επίσης επίπεδα σε διαδοχική σειρά και τα δεδομένα ρέουν από το ένα επίπεδο στο άλλο με τη δεδομένη σειρά έως ότου τα δεδομένα φτάσουν τελικά στο επίπεδο εξόδου. Είναι κατάλληλο για μια απλή στοίβα στρωμάτων όπου κάθε στρώμα έχει ακριβώς έναν τανυστή εισόδου και έναν τανυστή εξόδου.
- **Layers**: για τη συγκεκριμένη αρχιτεκτονική που δημιουργήθηκε στην παρούσα εργασία, όπως παρουσίαζεται παρακάτω, επιλέχθηκε να προστεθούν συνολικά 5 στρώματα ανάμεσα στην είσοδο και την έξοδο του μοντέλου. Αυτά απαρτίζονται από δύο στρώματα LSTM, δύο στρώματα Dropout κι ένα τελευταίο στρώμα Dense. Αναλυτικότερα για το κάθε επίπεδο, για το LSTM έχουμε ήδη περιγράψει τη λειτουργία του στο Κεφάλαιο 2. Όσον αφορά το Dropout επίπεδο, η λειτουργία του είναι να ορίζει τυχαία τις μονάδες εισόδου στο 0 με συχνότητα συγκεκριμένου rate(ρυθμού) σε κάθε βήμα κατά τη διάρκεια του χρόνου training, κάτι που βοηθά στην αποφυγή της υπερπροσαρμογής(overfitting). Οι είσοδοι που δεν έχουν οριστεί στο 0 κλιμακώνονται κατά 1/(1 – rate) έτσι ώστε το άθροισμα όλων των εισόδων να παραμένει αμετάβλητο. Εδώ επιλέχθηκε ως τιμή του rate το 0.2. Τέλος, το στρώμα Dense είναι ένα στρώμα που είναι βαθιά συνδεδεμένο με το προηγούμενο στρώμα του που σημαίνει ότι οι νευρώνες του στρώματος συνδέονται με κάθε νευρώνα του προηγούμενου στρώματός του. Αυτό το επίπεδο είναι το πιο συχνά χρησιμοποιούμενο στρώμα σε τεχνητά νευρωνικά δίκτυα.

  Ο νευρώνας του Dense στρώματος σε ένα μοντέλο λαμβάνει έξοδο από κάθε νευρώνα του προηγούμενου στρώματος, όπου οι νευρώνες του Dense στρώματος εκτελούν πολλαπλασιασμό μήτρας-διανύσματος. Ο πολλαπλασιασμός αυτός είναι μια διαδικασία όπου το διάνυσμα γραμμής της εξόδου από τα προηγούμενα στρώματα είναι ίσο με το διάνυσμα στήλης του Dense στρώματος. Οι τιμές κάτω από τη μήτρα είναι οι εκπαιδευμένες παράμετροι των προηγούμενων στρωμάτων και μπορούν επίσης να ενημερωθούν από την backpropagation. Η backpropagation είναι ο πιο συχνά χρησιμοποιούμενος αλγόριθμος για την εκπαίδευση των νευρωνικών δικτύων feedforward. Γενικά, η αντίστροφη διάδοση σε ένα νευρωνικό δίκτυο υπολογίζει τη διαβάθμιση της συνάρτησης απώλειας σε σχέση με τα βάρη του δικτύου για μεμονωμένη είσοδο ή έξοδο. Μπορούμε να πούμε ότι η έξοδος που προέρχεται από το dense στρώμα θα είναι ένα διάνυσμα N-διάστασης. Μπορούμε να δούμε ότι μειώνει τη διάσταση των διανυσμάτων. Έτσι, ένα dense στρώμα βασικά χρησιμοποιείται για την αλλαγή της διάστασης των διανυσμάτων χρησιμοποιώντας κάθε νευρώνα. Όπως αναφέρθηκε προηγουμένως, τα αποτελέσματα από κάθε νευρώνα των προηγούμενων στρωμάτων πηγαίνουν σε κάθε μεμονωμένο νευρώνα του dense στρώματος. Μπορούμε λοιπόν να πούμε ότι εάν το προηγούμενο στρώμα εξάγει έναν πίνακα (M x N) συνδυάζοντας αποτελέσματα από κάθε νευρώνα, αυτή η έξοδος περνά μέσα από το

dense στρώμα όπου ο αριθμός των νευρώνων πρέπει να είναι N. Ως παράμετροι χρησιμοποιήθηκαν ο αριθμός των units = 1, ο οποίος ορίζει το μέγεθος της εξόδου του dense στρώματος και πρέπει πάντα να είναι ένας θετικός ακέραιος αριθμός, καθώς αντιπροσωπεύει τη διάσταση του διανύσματος εξόδου. Δεύτερη παράμετρος είναι η συνάρτηση ενεργοποίησης την οποία και θα αναλύσουμε εκτενέστερα αμέσως.

- **Activation**: στα νευρωνικά δίκτυα, η συνάρτηση ενεργοποίησης είναι μια συνάρτηση που χρησιμοποιείται για τον μετασχηματισμό των τιμών εισόδου των νευρώνων. Πρακτικά εισάγει τη μη γραμμικότητα στα δίκτυα, έτσι ώστε τα δίκτυα να μπορούν να μάθουν τη σχέση μεταξύ των τιμών εισόδου και εξόδου. Υπάρχουν πολλές διαθέσιμες συναρτήσεις ενεργοποίησης. Στο συγκεκριμένο μοντέλο εφαρμόστηκαν τρεις και πιο συγκεκριμένα:
    1. ***Tanh***: υπερβολική εφαπτομένη, είναι μια συνάρτηση παρόμοια με μια συνάρτηση μη γραμμικής ενεργοποίησης που εξάγει τιμές μεταξύ -1.0 και 1.0 .
    2. ***Relu***: είναι μια τμηματικά γραμμική συνάρτηση που θα εξάγει απευθείας την είσοδο εάν είναι θετική, διαφορετικά, θα βγάζει μηδέν. Έχει γίνει η προεπιλεγμένη λειτουργία ενεργοποίησης για πολλούς τύπους νευρωνικών δικτύων, επειδή ένα μοντέλο που το χρησιμοποιεί είναι πιο εύκολο να εκπαιδευτεί και συχνά επιτυγχάνει καλύτερη απόδοση. Η συνάρτηση είναι γραμμική για τιμές μεγαλύτερες από το μηδέν, που σημαίνει ότι έχει πολλές από τις επιθυμητές ιδιότητες μιας συνάρτησης γραμμικής ενεργοποίησης όταν εκπαιδεύει ένα νευρωνικό δίκτυο χρησιμοποιώντας backpropagation.
    3. ***Sigmoid***: η συνάρτηση σιγμοειδούς ενεργοποίησης, που ονομάζεται επίσης λογιστική συνάρτηση, είναι παραδοσιακά μια πολύ δημοφιλής συνάρτηση ενεργοποίησης για νευρωνικά δίκτυα. Η είσοδος στη συνάρτηση μετατρέπεται σε τιμή μεταξύ 0.0 και 1.0. Οι είσοδοι που είναι πολύ μεγαλύτερες από 1.0 μετατρέπονται στην τιμή 1.0, παρομοίως, τιμές πολύ μικρότερες από 0.0 μετατρέπονται σε 0.0. Μαθηματικά ισχύει: sigmoid(x) = 1 / (1 + exp(-x)) .
- **Recurrent dropout:** είναι μια μέθοδος regularization, όπου οι συνδέσεις εισόδου και οι επαναλαμβανόμενες συνδέσεις με LSTM πιθανώς εξαιρούνται από την ενεργοποίηση και τις ενημερώσεις βάρους κατά την εκπαίδευση ενός δικτύου. Αυτό έχει ως αποτέλεσμα τη μείωση της υπερπροσαρμογής και τη βελτίωση της απόδοσης του μοντέλου. Ορίζεται ένας ρυθμός(rate) για το dropout ο οποίος εδώ επιλέχθηκε ίσο με 0.2
- **Loss:** οι συναρτήσεις απώλειας είναι μια από τις πιο σημαντικές πτυχές των νευρωνικών δικτύων, καθώς (μαζί με τις συναρτήσεις βελτιστοποίησης) είναι άμεσα υπεύθυνες για την προσαρμογή του μοντέλου στα δεδομένα εκπαίδευσης. Μια συνάρτηση απώλειας είναι μια συνάρτηση που συγκρίνει τον στόχο και τις προβλεπόμενες τιμές εξόδου, μετρά πόσο καλά το νευρωνικό δίκτυο μοντελοποιεί τα δεδομένα εκπαίδευσης. Κατά την εκπαίδευση, σκοπός είναι να ελαχιστοποιηθεί αυτή η απώλεια μεταξύ των προβλεπόμενων και των στοχευόμενων αποτελεσμάτων. Η συνάρτηση απώλειας που χρησιμοποιείται σε μοντέλα δυαδικής ταξινόμησης (όπως είναι και το παρόν μοντέλο), όπου το μοντέλο δέχεται μια είσοδο και πρέπει να την ταξινομήσει σε μία από τις δύο προκαθορισμένες κατηγορίες, είναι συνήθως η Binary Cross-Entropy/Log Loss. Πράγματι, ύστερα από δοκιμές με άλλες συναρτήσεις, η βέλτιστη δυνατή ορθότητα επιτεύχθηκε και στη δική μας περίπτωση με τη χρήση αυτής της συνάρτησης απώλειας. Μαθηματικά δίνεται από τον τύπο:

$$CE\ Loss = \frac{1}{n}\sum_{i=1}^{N} -(y_i \cdot \log(p_i) + (1 - y_i) \cdot \log(1 - p_i)$$

Τα νευρωνικά δίκτυα ταξινόμησης λειτουργούν βγάζοντας ένα διάνυσμα πιθανοτήτων, την πιθανότητα η δεδομένη είσοδος να ταιριάζει σε καθεμία από τις προκαθορισμένες κατηγορίες. Στη συνέχεια επιλέγεται η κατηγορία με την υψηλότερη πιθανότητα ως τελική έξοδο. Στη δυαδική ταξινόμηση, υπάρχουν μόνο δύο πιθανές πραγματικές τιμές του y, 0 ή 1. Έτσι, για να προσδιοριστεί με ακρίβεια η απώλεια μεταξύ της πραγματικής και της προβλεπόμενης τιμής, πρέπει να συγκρίνει την πραγματική τιμή (0 ή 1) με την πιθανότητα να ευθυγραμμιστεί η είσοδος με αυτήν την κατηγορία (p(i) = πιθανότητα η κατηγορία να είναι 1, (1 – p(i)) = πιθανότητα η κατηγορία να είναι 0)

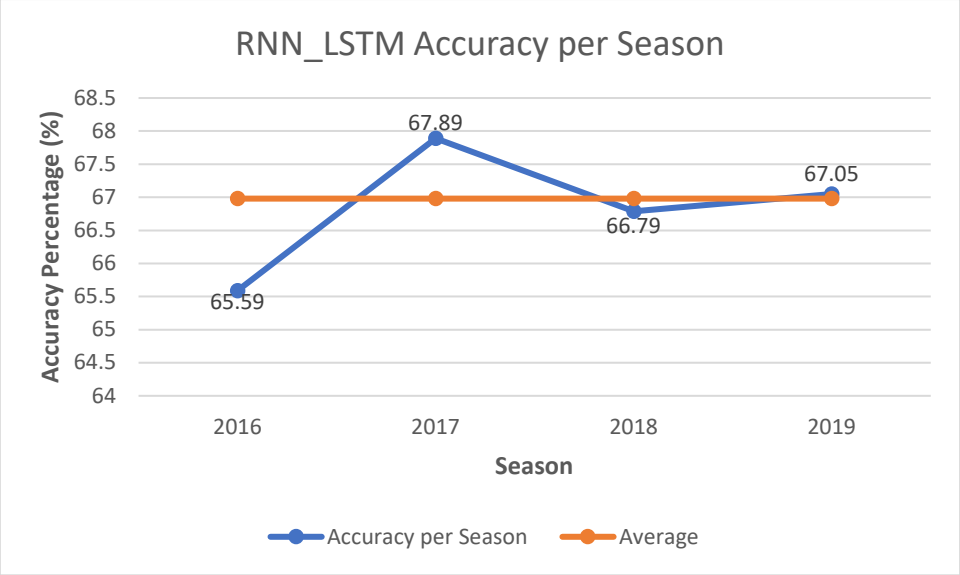
- **Optimizer:** οι βελτιστοποιητές είναι αλγόριθμοι ή μέθοδοι που χρησιμοποιούνται για την ελαχιστοποίηση μιας συνάρτησης σφάλματος (συνάρτηση απώλειας) ή για τη μεγιστοποίηση της αποδοτικότητας της παραγωγής. Οι βελτιστοποιητές είναι μαθηματικές συναρτήσεις που εξαρτώνται από τις παραμέτρους του μοντέλου, π.χ. Weights & Biases. Βοηθούν ώστε να γίνεται αντιληπτό πώς θα πρέπει να αλλάζουν τα βάρη και ο ρυθμός εκμάθησης του νευρωνικού δικτύου για να μειώνονται οι απώλειες. Υπάρχουν πολλές εναλλακτικές που χρησιμοποιούνται ευρέως για την ανάπτυξη μοντέλων Βαθιάς Μάθησης. Ωστόσο, στην παρούσα διπλωματική, εφαρμόστηκε ο optimizer Adam(Adaptive Moment Estimation), ο αλγόριθμος είναι μια περαιτέρω επέκταση της Στοχαστικής Κλίσης Καθόδου(Stochastic Gradient Descent) για την ενημέρωση των βαρών του δικτύου κατά τη διάρκεια της εκπαίδευσης. Σε αντίθεση με τη διατήρηση ενός ενιαίου ρυθμού εκμάθησης μέσω της εκπαίδευσης στο SGD, ο Adam optimizer ενημερώνει τον ρυθμό εκμάθησης για κάθε βάρος δικτύου ξεχωριστά. Ο Adam έχει πολλά πλεονεκτήματα, λόγω των οποίων χρησιμοποιείται ευρέως. Συνιστάται ως αλγόριθμος βελτιστοποίησης, καθώς είναι απλός στην εφαρμογή, έχει ταχύτερο χρόνο λειτουργίας, χαμηλές απαιτήσεις μνήμης και απαιτεί λιγότερο συντονισμό από οποιονδήποτε άλλον αλγόριθμο βελτιστοποίησης.

Τελικά, βέλτιστη δομή του μοντέλου φάνηκε να αποτελεί η ακόλουθη:

```
model = Sequential ()
model.add (LSTM (50, activation='tanh', recurrent_dropout=0.2, return_sequences=True))
model.add (Dropout (0.2))
model.add (LSTM (50, activation='relu', recurrent_dropout=0.2, return_sequences=True))
model.add (Dropout (0.2))
model.add (Dense (1, activation='sigmoid')),
```

ενώ για το βήμα του compilation επιλέχθηκαν ως παράμετροι, loss = binary_crossentropy, optimizer = adam, metrics = accuracy.

Η ορθότητα (accuracy) , στο test set, ισούται με 66.98, ενώ για το training set η αντίστοιχη τιμή ισούται με 69.69%

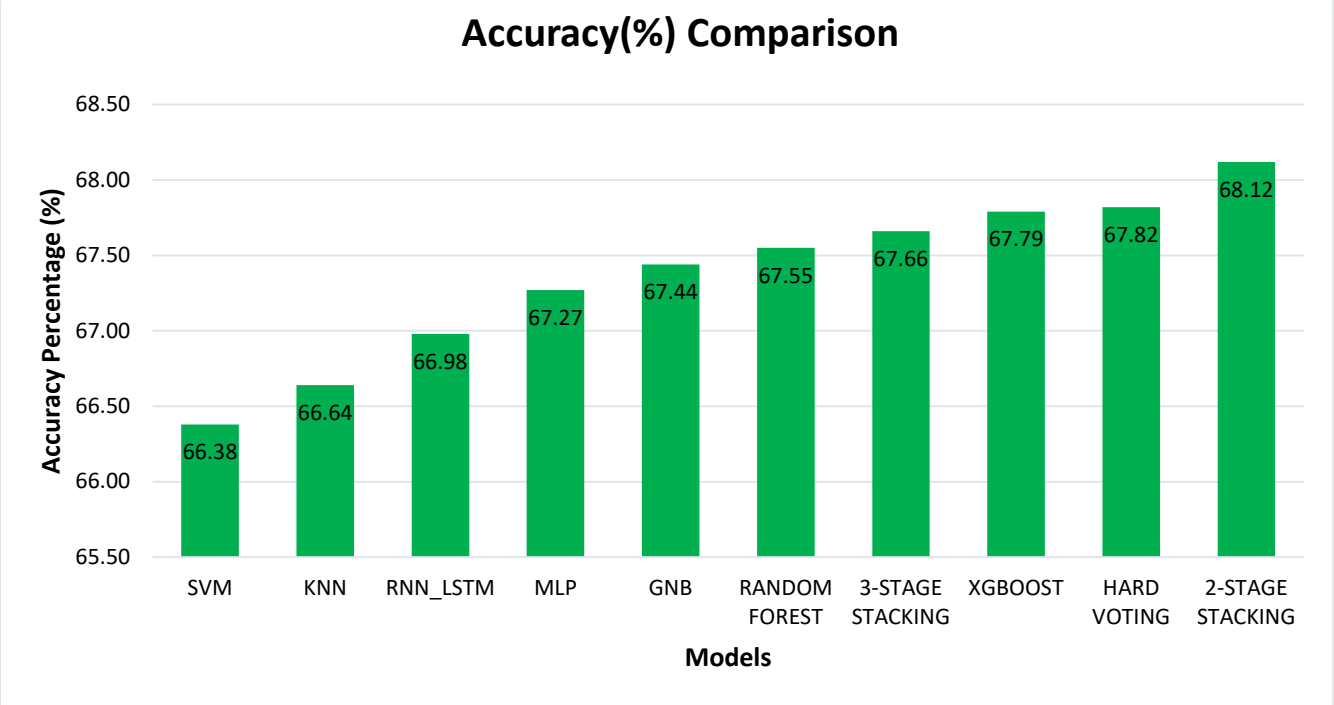


Στο σημείο αυτό να σημειωθεί, ότι η σύγκριση ανάμεσα σε Training και Test Accuracy, γίνεται για λόγους επαλήθευσης των αποτελεσμάτων τόσο κατά το στάδιο της εκπαίδευσης όσο και κατά το στάδιο των δοκιμών. Ειδικότερα, σε περίπτωση που υπήρχε πολύ υψηλό accuracy στο training set και διέφερε κατά πολύ από το ποσοστό accuracy του test set, θα προέκυπτε το συμπέρασμα ότι υπάρχει overfitting. Θα σήμαινε ότι το εκάστοτε μοντέλο έμαθε κανόνες ειδικά για το σετ εκπαίδευσης κι αυτοί οι κανόνες δεν γενικεύονται ορθότερα πέρα από το σύνολο αυτό. Επιπλέον, όπως φαίνεται από τις μετρικές των μοντέλων, αποφεύχθηκε το σφάλμα υψηλότερης ορθότητας στο test set από το training set.

4.4 Σύνοψη Μεθόδων

Με την παρουσίαση των παραπάνω αλγορίθμων και των αποτελεσμάτων τους, παρατηρεί κανείς πως τα μοντέλα πέτυχαν σε πολύ ικανοποιητικό βαθμό τον πρωταρχικό στόχο τους που δεν ήταν άλλος από την επίτευξη υψηλού ποσοστού ορθότητας στις προβλέψεις τους. Στην ενότητα αυτή θα υπάρξει σύγκριση των διαφορετικών υλοποιήσεων, ώστε να απαντηθεί ποια ήταν εν τέλει και η βέλτιστη μεταξύ αυτών.

Αρχικά, παρακάτω παρουσιάζεται το συγκριτικό γράφημα που αφορά την Ορθότητα (Accuracy) του κάθε μοντέλου που αναπτύχθηκε.

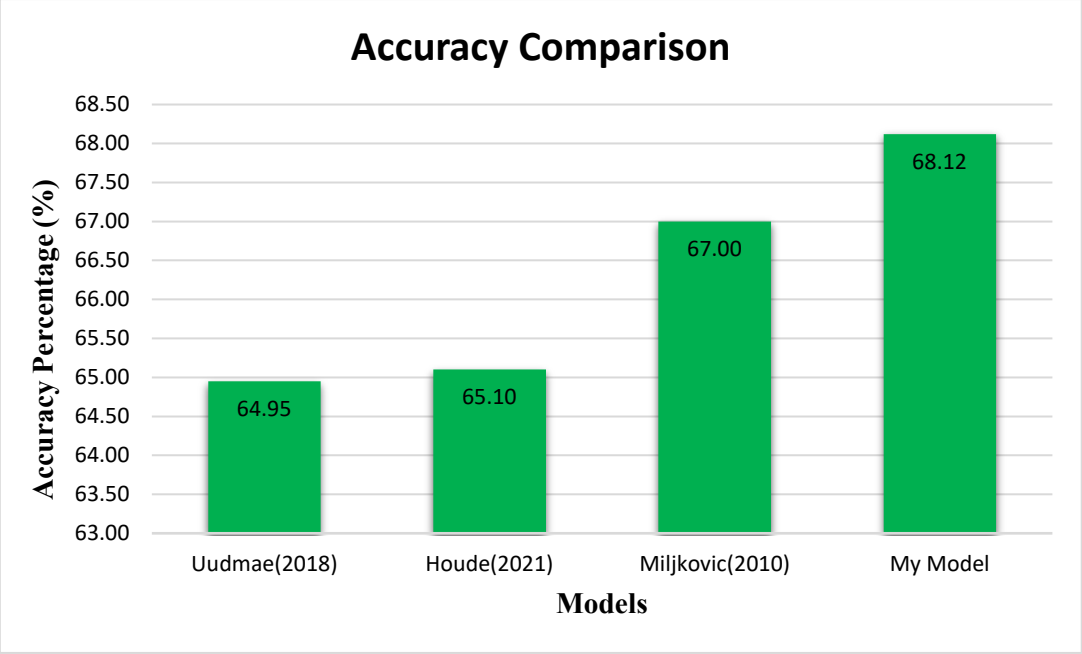


Όπως προκύπτει από το διάγραμμα, οι υλοποιήσεις που ξεχωρίζουν είναι ο 2 – Stage Stacking με ορθότητα 68.12%, το Hard Voting μοντέλο με ποσοστό ορθότητας 67.82%, ενώ πολύ κοντά ακολουθούν ο XGBoost με ορθότητα 67.79% και ο 3 – Stage Stacking με ποσοστό ορθότητας 67.66%.

4.5 Συγκριτική Ανάλυση των Αποτελεσμάτων με άλλες Εργασίες

Ιδιαίτερο ενδιαφέρον παρουσιάζει η μελέτη άλλων εργασιών της διεθνούς βιβλιογραφίας στις οποίες αναπτύσσονται μοντέλα παρόμοια με τα δικά μας και οι οποίες με τη σειρά τους στοχεύουν στην επίτευξη της υψηλότερης δυνατής ορθότητας. Όπως φάνηκε από την αναζήτησή μας, υπάρχει μεγάλη ανάπτυξη, τις δύο τελευταίες δεκαετίες περίπου, στο θέμα των sports analytics και στο συνδυασμό των δεδομένων αυτών με τεχνικές Μηχανικής Μάθησης. Οι ιδέες και οι διαφορετικές προσεγγίσεις ποικίλλουν, από βασικούς αλγορίθμους μέχρι την ανάπτυξη Fuzzy Δικτύων, καθώς επίσης Νευρικών Δικτύων Βαθιάς Μάθησης και συνδυασμό όλων των παραπάνω σε αρκετές εργασίες.

Από αυτήν την ευρεία πηγή ορισμένες υλοποιήσεις αποτέλεσαν σημαντικό κριτήριο σύγκρισης για τα δικά μας μοντέλα και το πόσο επιτυχημένα τελικά μπορούν να θεωρηθούν. Στο ακόλουθο διάγραμμα παρατίθενται τρεις διαφορετικές εργασίες των οποίων η βέλτιστη προτεινόμενη λύση τους υπολείπεται αισθητά του 2 – Stage Stacking μοντέλου που προκρίνεται ως το καλύτερο από την παρούσα εργασία.



Πιο συγκεκριμένα, ο Miljkovic στην εργασία του προτείνει ένα Naïve Bayes ταξινομητή ο οποίος δεν καταφέρνει να ξεπεράσει το 67% ποσοστό ορθότητας. Παρομοίως, ο Houde[20], με ένα μοντέλο Gaussian Naïve Bayes έφτασε το 65.10%, ενώ ο Uudmae[21], μετά από δοκιμή Linear Regression, SVM και ενός NNR(Neural Network Regression) μοντέλου, κατέληξε στο NNR με ποσοστό ορθότητας 64.95% . Όπως γίνεται εύκολα αντιληπτό, η τεχνική του 2 – Stage Stacking, αλλά και οι περισσότερες από τις τεχνικές που υλοποιήθηκαν για την παρούσα διπλωματική, επιτυγχάνουν σημαντικά καλύτερα ποσοστά ορθότητας.

5 Συμπεράσματα

5.1 Συμπεράσματα και Παρατηρήσεις

Στο προηγούμενο κεφάλαιο παρουσιάστηκαν όλες οι υλοποιήσεις και τα αποτελέσματα τους, τα οποία ήταν πολύ θετικά. Οι υλοποιήσεις επέτυχαν υψηλά ποσοστά ορθότητας κατά τη διάρκεια των 4 χρόνων που αποτέλεσαν το σετ δοκιμών, διατηρώντας αυτήν την καλή συμπεριφορά σε όλη σχεδόν τη διάρκεια. Αυτός ήταν άλλωστε και ο στόχος της εργασίας, η ανάπτυξη τεχνικών που μπορούν να οδηγήσουν σε μεγαλύτερη ορθότητα πρόβλεψης του τελικού αποτελέσματος ενός αγώνα μπάσκετ αξιοποιώντας την πληροφορία του συνόλου δεδομένων. Σε όλη τη διάρκεια της εργασίας, από την αρχή μέχρι το τέλος, προέκυψαν ενδιαφέροντα συμπεράσματα και παρατηρήσεις, τα οποία θα αναφερθούν παρακάτω.

Τα κυριότερα συμπεράσματα και παρατηρήσεις που προέκυψαν και αξίζουν να αναφερθούν:

1. Η χρήση πολλών χαρακτηριστικών δεν συνεπάγεται απαραίτητα υψηλή ορθότητα

Η χρήση πολλών features ως training set, δηλαδή περισσότερα δεδομένα, δεν οδηγεί αυτόματα και σε υψηλότερα ποσοστά ορθότητας στις προβλέψεις των μοντέλων. Όπως έχει ήδη αναφερθεί σε προηγούμενο κεφάλαιο, ο μεγαλύτερος όγκος δεδομένων πιθανότατα θα σημαίνει και μεγαλύτερη συσχέτιση μεταξύ των χαρακτηριστικών, γεγονός που εμποδίζει ένα μοντέλο να αξιοποιήσει την ουσιαστική πληροφορία που θα «κρύβεται» στο σύνολο εκπαίδευσης. Κάτι τέτοιο φάνηκε εξάλλου και σε όλες τις υλοποιήσεις στην παρούσα διπλωματική, όπου σε κάθε περίπτωση έγινε προσπάθεια να εξαγάγουμε τη χρήσιμη πληροφορία από ένα πολύ μικρό μέρος του συνόλου δεδομένων.

Αυτή η πορεία σκέψης απέδωσε τελικά σε αρκετά ικανοποιητικό βαθμό, καθώς τα μοντέλα που παρουσιάστηκαν πράγματι κατάφεραν υψηλότερη ορθότητα με λιγότερα features, συγκρινόμενα με πολλές ακόμη δοκιμές που πραγματοποιήθηκαν κατά τη διάρκεια ανάπτυξης της εργασίας, όπου το σετ εκπαίδευσης ήταν μεγαλύτερο.

2. Οι Μέθοδοι Επιλογής Χαρακτηριστικών είναι ιδιαίτερα χρήσιμες

Η εφαρμογή Feature Selection μεθόδων κρίνεται απαραίτητη σε περιπτώσεις με μεγάλο όγκο δεδομένων, όπως και στην παρούσα εργασία, η οποία αποτελείται περίπου από 13.000 παιχνίδια με το καθένα να έχει πάνω από 100 διαφορετικά χαρακτηριστικά που το περιγράφουν. Για κάθε μοντέλο δοκιμάστηκαν πολλές μέθοδοι και πολλές παράμετροι των μεθόδων αυτών οι οποίες άλλαζαν συχνά για να δώσουν το βέλτιστο αποτέλεσμα

συνδυαζόμενες με τους αλγορίθμους Μηχανικής Μάθησης που μελετώνται στην παρούσα εργασία.

Από όλους αυτούς τους ξεχωριστούς συνδυασμούς, όπως γίνεται αντιληπτό κι από το κεφάλαιο των πειραματικών αποτελεσμάτων, ξεχώρισαν κάποιες συγκεκριμένες μέθοδοι. Αυτές είναι ο ταξινομητής Extra Trees Classifier, η συνάρτηση SelectFromModel(LassoCV) και τέλος, η μέθοδος RFECV – Recursive Feature Elimination with Cross Validation. Σημαντικό επιπλέον στάδιο είναι κι αυτό της προεπεξεργασίας το οποίο και εφαρμόστηκε σε ορισμένα από τα μοντέλα και απέφερε υψηλότερα ποσοστά ορθότητας στις προβλέψεις τους. Ενδεικτικά, να υπενθυμίσουμε τη χρήση των Standard και Minmax Scalers.

3. Συγκεκριμένα χαρακτηριστικά περιέχουν «συμπυκνωμένη» πληροφορία

Τα μοντέλα που αναπτύχθηκαν χρησιμοποίησαν χαρακτηριστικά για την εκπαίδευσή τους διαφορετικά από τα συνηθισμένα, δηλαδή ποσοστά ευστοχίας μιας ομάδας ανά αγώνα, ασσίστ, ριμπάουντ και άλλα που καταγράφονται στο λεγόμενο box score. Αυτή τη φορά, όπως φάνηκε, τα επικρατέστερα χαρακτηριστικά ήταν η βαθμολογία Elo που περιγράφηκε σε προηγούμενο κεφάλαιο, το Player Efficiency, οι απουσίες – τραυματισμένοι παίκτες που είχε μια ομάδα σε έναν αγώνα, οι στοιχηματικές αποδόσεις για τη νίκη γηπεδούχου / φιλοξενούμενης ομάδας, καθώς επίσης τα χαρακτηριστικά που περιγράφουν τις διαφορές στα στατιστικά των ομάδων και χαρακτηριστικά που περιγράφουν τη συμπεριφορά των ομάδων στα τελευταία **n** παιχνίδια που έχουν παίξει.

4. Τα Μοντέλα Συνόλου είναι πιο επιτυχημένα από τα βασικά Μοντέλα Μηχανικής Μάθησης

Παρατηρήθηκε πως τα κορυφαία μοντέλα όσον αφορά το υψηλό Accuracy είναι αυτά που δημιουργήθηκαν με βάση τις Ensemble τεχνικές. Ειδικότερα, όπως έδειξε και το συνόπτικο διάγραμμα στο Κεφάλαιο **4.4 Σύνοψη Μεθόδων**, τα τρία από τα τέσσερα καλύτερα μοντέλα είναι οι 2 – Stage, 3 – Stage Stacking Classifiers και ο Hard Voting Classifier, ενώ και ο XGBoost που πέτυχε υψηλό ποσοστό ορθότητας, βασίζεται στην ιδέα του Boosting που αποτελεί επίσης μια μοντελοποίηση συνόλου.

5. Η υλοποίηση του 2 – Stage Stacking Classifier κατάφερε να ξεχωρίσει

Η τελική σύγκριση ανέδειξε ως πιο επιτυχημένη υλοποίηση αυτήν του 2 – Stage Stacking Ταξινομητή. Με ποσοστό ορθότητας 68.12% υπερβαίνει αισθητά τα υπόλοιπα μοντέλα της συγκεκριμένης διπλωματικής, καθώς και τα ποσοστά μοντέλων από άλλες εργασίες, όπως δείξαμε προηγουμένως.

5.2 Μελλοντικές Επεκτάσεις

Η Μηχανική Μάθηση και τα Analytics τα τελευταία χρόνια έχουν κερδίσει πολύ έδαφος στο χώρο του αθλητισμού γενικότερα και έχει αποκτήσει πιστούς υποστηρικτές. Η παρούσα διπλωματική προσπάθησε να αναπτύξει γνωστές τεχνικές προς ένα υψηλότερο επίπεδο, το οποίο θα παράγει προβλέψεις για τα αποτελέσματα αγώνων του πρωταθλήματος του NBA με τη μέγιστη δυνατή ορθότητα. Σε παρόμοια φιλοσοφία υπάρχουν ακόμη πολλά ανοικτά και πρόσφορα μέτωπα με τα οποία μπορεί κανείς να καταπιαστεί όπως:

- η ανάπτυξη ισχυρών νευρωνικών δικτύων Βαθιάς Μάθησης για την επίτευξη ακόμη ακριβέστερης πρόβλεψης του τελικού αποτελέσματος.

- η χρήση πρόσθετων features, τα λεγόμενα Advanced Analytics, τα οποία υπάρχουν διαθέσιμα σε βάσεις δεδομένων αλλά προς το παρόν καλύπτουν ένα μικρό εύρος περασμένων ετών.

- η πρόβλεψη για τα ατομικά βραβεία που δίνονται στους παίκτες στο τέλος της σεζόν, όπως το βραβείο του πολυτιμότερου παίκτη – MVP, του πιο βελτιωμένου παίκτη της σεζόν, την καλύτερη πεντάδα του πρωταθλήματος και αρκετά ακόμη.

- η πρόβλεψη του πίνακα των playoffs για μια σεζόν και συνολικά της πορείας μέχρι κα τους τελικούς του NBA και την ανάδειξη του πρωταθλητή.

- άλλα πρωταθλήματα μπάσκετ ανά τον κόσμο, όπως για παράδειγμα η Euroleague που προσομοιάζει στον τρόπο διεξαγωγής της με το NBA, με πολλά παιχνίδια να λαμβάνουν χώρα κατά τη διάρκεια της σεζόν.

- η εφαρμογή αντίστοιχων τεχνικών πρόβλεψης σε άλλα αθλήματα, όπως το ποδόσφαιρο, το αμερικάνικο football, το baseball και το τένις. Να αναφερθεί, ότι έχουν γίνει προσπάθειες και σε αυτά τα αθλήματα, αλλά τα περιθώρια βελτίωσης είναι μεγάλα, καθώς όλο και πληθαίνουν οι πηγές και ο όγκος λεπτομερών δεδομένων για καθένα από αυτά.

- η υλοποίηση μιας διαδικτυακής πλατφόρμας η οποία θα ενημερώνεται καθημερινά για κάθε παιχνίδι που πραγματοποιείται και θα συλλέγει τα δεδομένα που επιθυμούμε. Ο χρήστης θα μπορεί να επιλέγει ημερομηνία και αγώνες που τον ενδιαφέρουν προς πρόβλεψη, ώστε οι αλγόριθμοι που αναπτύχθηκαν να αναλάβουν να δώσουν τη βέλτιστη λύση στο ερώτημα αυτό.

Βιβλιογραφία

[1] https://www.wired.com/story/history-predicting-future/

[2] https://analyticsindiamag.com/

[3] https://www.latentview.com/blog/how-sports-analytics-is-changing-the-game/

[4] https://medium.com/@nabil_lathif/the-number-games-how-machine-learning-is-changing-sports

[5] Miljković, D., Gajić, L., Kovačević, A., & Konjović, Z. (2010, September). The use of data mining for basketball matches outcomes prediction. In *IEEE 8th international symposium on intelligent systems and informatics* (pp. 309-312). IEEE.

[6] https://www.datacamp.com/tutorial/k-nearest-neighbor-classification-scikit-learn

[7] https://medium.com/@ar.ingenious/applying-random-forest-classification-machine-learning-algorithm-from-scratch-with-real-24ff198a1c57

[8] Jain, S., & Kaur, H. (2017, September). Machine learning approaches to predict basketball game outcome. In *2017 3rd International Conference on Advances in Computing, Communication & Automation (ICACCA)(Fall)* (pp. 1-7). IEEE.

[9] McCabe, A., & Trevathan, J. (2008, April). Artificial intelligence in sports prediction. In *Fifth International Conference on Information Technology: New Generations (itng 2008)* (pp. 1194-1197). IEEE.

[10] https://www.geeksforgeeks.org/xgboost/

[11] https://machinelearningmastery.com/tour-of-ensemble-learning-algorithms/

[12] https://www.how2shout.com/what-is/what-is-adaboost-boosting-technique.html

[13] Thabtah, F., Zhang, L. & Abdelhamid, N. NBA Game Result Prediction Using Feature Analysis and Machine Learning. *Ann. Data. Sci.* **6**, 103–116 (2019). https://doi.org/10.1007/s40745-018-00189-x.

[14] https://scikit-learn.org/stable/modules/feature_selection.html

[15] https://en.m.wikipedia.org/wiki/National_Basketball_Association

[16] Cao, C. (2012). Sports data mining technology used in basketball outcome prediction.

[17] https://fivethirtyeight.com/features/how-we-calculate-nba-elo-ratings/

[18] "What is Player Efficiency Rating?". Washington Post. Retrieved 2020-09-27

[19] https://en.wikipedia.org/wiki/Efficiency_(basketball)

[20] Houde, M. (2021). Predicting the Outcome of NBA Games.

[21] Uudmae, J. (2018). Predicting NBA Game Outcomes. *Accessado em*, *28*(09).