ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ
ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ
ΤΟΜΕΑΣ ΗΛΕΚΤΡΙΚΩΝ ΒΙΟΜΗΧΑΝΙΚΩΝ ΔΙΑΤΑΞΕΩΝ
& ΣΥΣΤΗΜΑΤΩΝ ΑΠΟΦΑΣΕΩΝ

Πρόβλεψη Αποτελεσμάτων Αγώνων Τένις με Χρήση Μηχανικής Μάθησης και Ανάπτυξη Στοιχηματικής Στρατηγικής

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

Φρίξος Δ. Νικολουλόπουλος

Επιβλέπων: Βασίλειος Ασημακόπουλος

Καθηγητής Ε.Μ.Π.

Υπεύθυνος: Αρτέμιος-Ανάργυρος Σεμένογλου

Υποψήφιος Διδάκτωρ Ε.Μ.Π.

Περίοδος και τόπος αξιολόγησης της παρούσης εργασίας,

Αθήνα, Μάρτιος 2023

ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ
ΤΟΜΕΑΣ ΗΛΕΚΤΡΙΚΩΝ ΒΙΟΜΗΧΑΝΙΚΩΝ ΔΙΑΤΑΞΕΩΝ & ΣΥΣΤΗΜΑΤΩΝ ΑΠΟΦΑΣΕΩΝ

Πρόβλεψη Αποτελεσμάτων Αγώνων Τένις με Χρήση Μηχανικής Μάθησης και Ανάπτυξη Στοιχηματικής Στρατηγικής

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

Φρίξος Δ. Νικολουλόπουλος

Επιβλέπων: Βασίλειος Ασημακόπουλος

Καθηγητής Ε.Μ.Π.

Υπεύθυνος: Αρτέμιος-Ανάργυρος Σεμένογλου

Υποψήφιος Διδάκτωρ Ε.Μ.Π.

Τα μέλη της εξεταστικής Επιτροπής που ενέκριναν την παρούσα εργασία,

(Υπογραφή)	(Υπογραφή)	(Υπογραφή)
……	……	……
Ασημακόπουλος	**Ψαρράς**	**Ασκούνης**
Βασίλειος	**Ιωάννης**	**Δημήτριος**

Ημερομηνία έγκρισης της παρούσης εργασίας από την επιτροπή,

Αθήνα, …… Μαρτίου 2023

ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ
ΤΟΜΕΑΣ ΗΛΕΚΤΡΙΚΩΝ ΒΙΟΜΗΧΑΝΙΚΩΝ ΔΙΑΤΑΞΕΩΝ & ΣΥΣΤΗΜΑΤΩΝ ΑΠΟΦΑΣΕΩΝ

(Υπογραφή)

……..

Φρίξος Νικολουλόπουλος
Διπλωματούχος Ηλεκτρολόγος Μηχανικός και Μηχανικός Υπολογιστών

Ευχαριστίες

Η διπλωματική αυτή εργασία εκπονήθηκε στα πλαίσια των ερευνητικών δραστηριοτήτων της Μονάδας Προβλέψεων και Στρατηγικής. Η μονάδα υπάγεται στον Τομέα Βιομηχανικών Διατάξεων και Συστημάτων Αποφάσεων της Σχολής Ηλεκτρολόγων Μηχανικών & Μηχανικών Η/Υ, του Εθνικού Μετσόβιου Πολυτεχνείου.

Σε αυτό το σημείο, θα ήθελα να ευχαριστήσω τον καθηγητή κ. Βασίλειο Ασημακόπουλο, του τομέα Ηλεκτρικών και Βιομηχανικών Διατάξεων και Συστημάτων Αποφάσεων, για την ευκαιρία που μου έδωσε να ασχοληθώ σε βάθος με την πρόβλεψη αποτελεσμάτων αγώνων τένις με χρήση μηχανικής μάθησης και την ανάπτυξη στοιχηματικής στρατηγικής.

Επίσης, θα ήθελα να ευχαριστήσω τους καθηγητές κ. Ψαρρά Ιωάννη και κ. Ασκούνη Δημήτριο για την συμμετοχή τους στην Επιτροπή εξέτασης της εργασίας.

Φυσικά, ευχαριστώ θερμά και τον υποψήφιο διδάκτορα κ. Αρτέμιο-Ανάργυρο Σεμένογλου για την συνεχή βοήθεια και καθοδήγηση καθ’ όλη την διάρκειας της εργασίας.

Τέλος, θα ήθελα να εκφράσω την ευγνωμοσύνη μου στους γονείς μου, Δημήτρη και Ελένη, για την αμέριστη στήριξη τους σε ό,τι κι αν χρειάστηκε όλα αυτά τα χρόνια.

Φρίξος Νικολουλόπουλος,

Αθήνα, Μάρτιος 2023

Περίληψη

Σε αυτή την εργασία μελετάται η αποδοτικότητα μοντέλων Δένδρων Αποφάσεων, Τυχαίου Δάσους και LightGBM με δεδομένα από το ATP Dataset στο πρόβλημα της εκτίμησης Φαβορί σε αγώνες Τένις, καθώς και η σύγκριση των στρατηγικών Απλού Στοιχήματος σε Φαβορί, Στοιχήματος κατά Martingale και Επιλεκτικού Στοιχηματισμού ως προς την αξιοπιστία της πρόβλεψης του νικητή και ως προς την απόδοση του αγώνα. Τα αποτελέσματα δείχνουν πως τα μοντέλα Μηχανικής Μάθησης, αν και τα πάνε καλύτερα, δεν έχουν πολύ μεγάλη διαφορά από την εκτίμηση του Φαβορί με βάση την απόδοση που δίνουν οι Bookmakers στον αγώνα. Ως προς τις στρατηγικές, η μόνη μέθοδος που είχε αξιόπιστα αποτελέσματα για κέρδος ήταν η μέθοδος Martingale, η οποία όμως εκθέτει σε ρίσκο ένα σημαντικό αρχικό κεφάλαιο, το ύψος του οποίου είναι άγνωστο στον Παίκτη και μπορεί να φτάσει τις δεκάδες χιλιάδες ευρώ, ακόμα και εάν το αρχικό ποντάρισμα ξεκινάει από ένα ευρώ. Το κέρδος αυτής της στρατηγικής πετυχαίνει μέγιστο κέρδος της τάξης του 8% της συνολικής επένδυσης.

Λέξεις-Κλειδιά:

Τένις, Μηχανική Μάθηση, Δέντρα Αποφάσεων, Λήψη Αποφάσεων, Αθλητικός Στοιχηματισμός, Στοιχηματικές Στρατηγικές, Στοιχηματικά Μοντέλα, ATP Dataset

Abstract

The purpose of this Thesis is the study of the efficiency of machine learning models, namely Decision Trees, Random Forests and LightGBM, using data from the ATP Dataset tackling the problem of determining the Favourites of Tennis Matches, as well as the comparison between different betting strategies, namely Simple Betting on Favourites, Martingale Model and Selective Betting based on Prediction Tenability for the Estimated Winning Athlete and the Match's Rate of Return, which is provided by the Bookmakers. The results seem to indicate that although machine learning models achieve a slightly better accuracy between a Bettor estimating the result of a match using the Bookmaker's odds alone, the difference is not significant enough. Regarding the Betting Strategies, the only method that achieved Net Gain with a significant reliability was the Martingale Method, which on the other hand demands the exposure of a significant starting capital on risk, the hight of which can reach even tens of thousands of euros, even though the starting bet for each simulation is a single euro. This betting strategy reaches a maximum gain percentage of 8% of the total invested capital.

Keywords:

Tennis, Machine Learning, Decision Trees, Decision-Making, Sport Betting, Betting Strategies, Betting Models, ATP Dataset

Περιεχόμενα

Λίστα Πινάκων

Λίστα Γραφημάτων

Εισαγωγή

Είναι γεγονός πως η έκρηξη που παρατηρείται στον κλάδο της πληροφορικής σήμερα οφείλεται στην ραγδαία ανάπτυξη των τεχνολογιών της Μηχανικής Μάθησης. Στη αιχμή αυτού του δόρατος γνώσης βρίσκονται οι τεχνολογικοί κολοσσοί της εποχής, οι οποίοι διαμοιράζουν δωρεάν εργαλεία που μπορεί να χρησιμοποιήσει οποιοσδήποτε. Ειδικότερα για τον τομέα της Μηχανικής Μάθησης και Ανάλυσης Δεδομένων είναι πλέον διαδεδομένα τα εργαλεία του ομίλου **Alphabet (μητρική της εταιρίας Google),** όπως η βιβλιοθήκη **scikit-Learn (γνωστή επίσης ως skLearn)** με μεθόδους Μηχανικής Μάθησης για την γλώσσα προγραμματισμού Python, η πλατφόρμα Colab που προσφέρει περιβάλλοντα ανάπτυξης και εκτέλεσης κώδικα στο **υπολογιστικό νέφος (cloud computing),** η πλατφόρμα Kaggle που δρα τόσο ως αποθετήριο δεδομένων όσο και ως φορέας διαγωνισμών Μηχανικής Μάθησης και η υπηρεσία αποθήκευσης και συγχρονισμού νέφους Google Drive. Δεν είναι τυχαίο λοιπόν που όλα τα παραπάνω εργαλεία χρησιμοποιήθηκαν στην παρούσα εργασία. [1] [2] [3] [4] [5]

Η Μηχανική Μάθηση είναι μια υποκατηγορία του κλάδου της Τεχνητής Νοημοσύνης, και κατ' επέκταση, της Επιστήμης Υπολογιστών, κατά την οποία αξιοποιούνται εμπειρικά ή ιστορικά δεδομένα από έναν αλγόριθμο για να εξαχθεί κάποια στατιστική εκτίμηση κατηγοριοποίησης ή πρόβλεψης, χωρίς να χρειαστεί να έχει ο αλγόριθμος κάποια βαθύτερη γνώση για το σύστημα που μελετά. Κοινώς, κατά την Μηχανική Μάθηση, αν ένα υπολογιστικό σύστημα τροφοδοτηθεί με τα κατάλληλα δεδομένα μπορεί να παράγει ένα μοντέλο γνώσης, το οποίο μπορεί να κάνει προβλέψεις με βάση τα δεδομένα που του δόθηκαν. Προφανώς, υπάρχουν περιορισμοί στο τι δεδομένα μπορούν να συλλεγούν και τι μπορεί να επιτύχει ένας αλγόριθμος, τα οποία θα μας απασχολήσουν στα επόμενα κεφάλαια. Το σημαντικό όμως είναι πως αν δοθούν αρκετά δεδομένα αγώνων κάποιου σπορ, υπάρχει αλγόριθμος, τέτοιος ώστε ένας υπολογιστής, υπό τις κατάλληλες συνθήκες, θα μπορούσε να προβλέψει τον νικητή. Για την εργασία αυτή θα περιοριστούμε στον αλγόριθμο και τα μοντέλα των Δέντρων Απόφασης και συναφείς τους επεκτάσεις, τα οποία θα αναπτυχθούν περαιτέρω στο αντίστοιχο κεφάλαιο. [6] [7]

Ο κλάδος που ασχολείται με την παραγωγή προβλέψεων ονομάζεται Τεχνικές Προβλέψεων. Οι Τεχνικές Προβλέψεων είναι ένας διεπιστημονικός κλάδος των Μαθηματικών, της Στατιστικής, των Οικονομικών και της Επιστήμης Υπολογιστών. Συνδυάζοντας μεθόδους από όλες αυτές τις επιστήμες, ο κλάδος αυτός είναι υπεύθυνος για την εκτίμηση, μελέτη και κατάστρωση στρατηγικών και μεθόδων λήψης αποφάσεων. Στόχος του κλάδου αυτού είναι η εφαρμογή των εκτιμήσεων που προκύπτουν για βελτιστοποίηση του κέρδους και ελαχιστοποίηση του ρίσκου επενδύσεων. Οι προβλέψεις μπορούν να ταξινομηθούν σε Κριτικές, Στατιστικές και Προβλέψεις Προϋπολογισμού. Κρητική Πρόβλεψη ορίζεται η πρόβλεψη η οποία γίνεται βάση της εμπειρίας ενός ατόμου ή μιας ομάδας. Η Στατιστική Πρόβλεψη είναι αποτέλεσμα ανάλυσης δεδομένων με στοχαστικές μεθόδους. Η Πρόβλεψη Προϋπολογισμού είναι μια ιδανική προσέγγιση που εκτιμά ή προσδοκά μία οικονομική οντότητα, όπως ένας ιδιώτης ή μια επιχείρηση. Στην εργασία αυτή, αφού υλοποιήσουμε μια μέθοδο πρόβλεψης αγώνων, θα εστιάσουμε στις στοιχηματικές στρατηγικές, όπως το σύστημα Martingale. [8] [9]

Στόχος της εργασίας αυτής είναι η εύρεση στοιχηματικών μεθόδων που να αποφέρουν μακροπρόθεσμα κέρδος μέσω μοντέλων Μηχανικής Μάθησης για πρόβλεψη των αποτελεσμάτων των αγώνων. Τα μοντέλα θα προκύψουν μετά από εκπαίδευση επάνω σε δεδομένα αγώνων Τένις. Η εργασία μπορεί χωριστεί σε 3 μέρη. Τα κεφάλαια 1 και 2 καλύπτουν το θεωρητικό υπόβαθρο του Τένις, του Στοιχήματος και της Μηχανικής Μάθησης που θα χρησιμοποιηθούν στην εργασία. Τα κεφάλαια 3 και 4, καλύπτουν το πειραματικό σκέλος που αφορά την ανάλυση και προεπεξεργασία των δεδομένων, την εκπαίδευση των προβλεπτικών μοντέλων καθώς και συγκριτική ανάλυση των αποτελεσμάτων τους. Στο $5^{ο}$ κεφάλαιο, περιγράφεται το πειραματικό σκέλος, το οποίο αφορά την προσομοίωση στοιχηματικών στρατηγικών αξιοποιώντας τα μοντέλα που αναπτύχθηκαν.

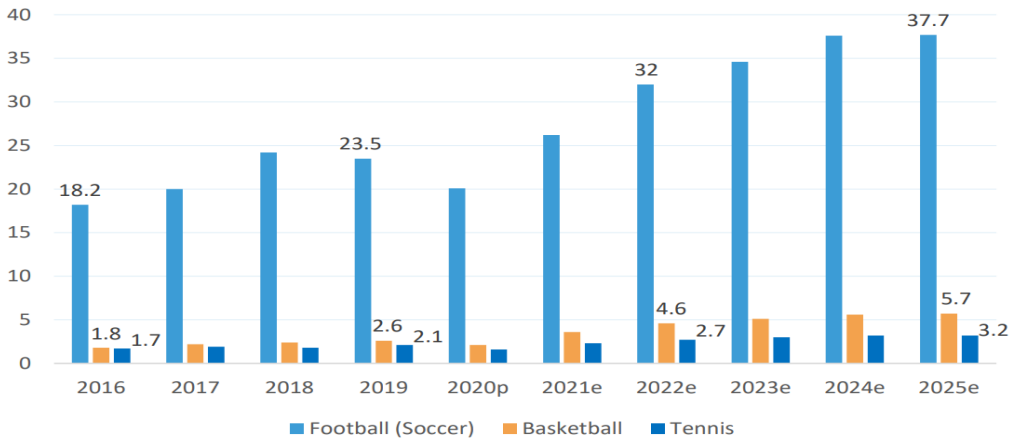
Τέλος, στο κεφάλαιο 6 γίνεται ένας εποπτικός σχολιασμός επί των συνολικών αποτελεσμάτων καθώς και συζήτηση πεδίων ενδιαφέροντος που χρήζουν περαιτέρω έρευνας.

Κεφάλαιο 1: Πρόβλεψη Αποτελέσματος και Μοντέλα Στοιχηματικών Στρατηγικών στον Αθλητισμό

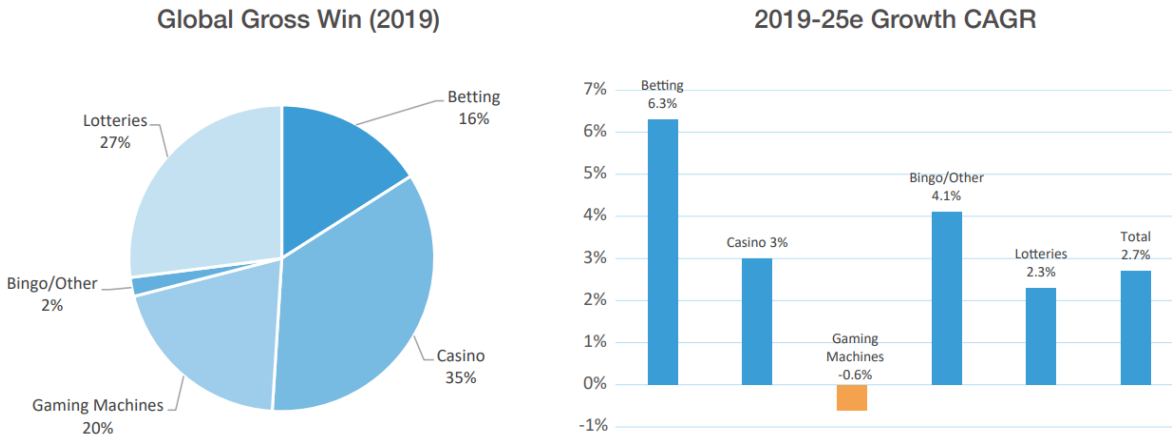
1.1 Εισαγωγή στον Στοιχηματισμό

Τα τυχερά παιχνίδια είναι εδραιωμένα από την αρχαιότητα και συνεχίζουν να έχουν απήχηση έως και σήμερα.

Από τα πιο δημοφιλή διαχρονικά είναι τα στοιχήματα στον αθλητισμό, όπως για παράδειγμα, οι ιπποδρομίες και τα αθλήματα Ολυμπιακών Αγώνων. Σήμερα, τα πιο διαδεδομένα παγκοσμίως, ως προς το πόσοι άνθρωποι στοιχηματίζουν σε αυτά, είναι κατά φθίνουσα σειρά το **ευρωπαϊκό ποδόσφαιρο (Football),** η **καλαθοσφαίριση ή μπάσκετ (Basketball)** και η **αντισφαίριση ή Τένις (Tennis)**. Σε οικονομικό επίπεδο, ενδιαφέροντα είναι τα δεδομένα και τα μεγέθη από την έκθεση της **Παγκόσμιας Οργάνωσης Στοιχηματικής Εντιμότητας (International Betting Integrity Association, IBIA)** του 2020. [10] [11] [12]



Εικόνα 1: Προσδοκόμενα Παγκόσμια Ακαθάριστα Κέρδη ανά Έτος για το Ποδόσφαιρο (Soccer), Μπάσκετ (Basketball) και Τένις (Tennis) σε δισεκατομύρια δολλάρια Ηνωμένων Πολιτειών (US $ bn). Πηγή: Παγκόσμια Οργάνωση Στοιχηματικής Εντιμότητας (IBIA), Έκθεση του 2020 [12]



Εικόνα 2: Ακαθάριστα Μερίδια Κερδών στα Παιχνίδια Τύχης Παγκοσμίως (αριστερά). Το μεγαλύτερο μερίδιο κατά φθίνουσα σειρά έχουν τα καζίνο (Casino), οι κληρώσεις/λοταρίες (Lotteries), τα μηχανικά τυχερά παιχνίδια (Gaming Machines), τα στοιχήματα (Betting) και τέλος τα υπόλοιπα. Εκτιμώμενη Μεταβολή Μεριδίου στα Τυχερά Παιχνίδια Παγκοσμίως (δεξιά). Η μεγαλύτερη αύξηση κατά φθίνουσα σειρά εκτιμάται στα στοιχήματα με 6.3%, στο Bingo (ή άλλα τυχερά παιχνίδια), στα καζίνο κατά 3% και στις λοταρίες κατά 2.3%. Στα μηχανικά τυχερά παιχνίδια προβλέπεται μείωση κατά 0.5%. Το συνολικό μερίδιο αγοράς εκτιμάται πως θα αυξηθεί για τον Τζόγο κατά 2.7%. Πηγή: Παγκόσμια Οργάνωση Στοιχηματικής Εντιμότητας (IBIA), Έκθεση του 2020. [12]

Το Τένις, συγκεκριμένα, είναι το πιο δημοφιλές από τα ατομικά αθλήματα, με την **Παγκόσμια Ομοσπονδία Τένις (International Tennis Federation, ITF)** να διοργανώνει περίπου 1500 διαφορετικά πρωταθλήματα κάθε χρόνο. Ο συνδυασμός της δημοτικότητας του αθλήματος και του στοιχήματος, η σύντομη διάρκεια των αγώνων, η συχνότητα με την οποία γίνονται μεγάλα τουρνουά, και η ζωντανή τους **κάλυψη μέσω διαδικτύου (live-streaming)** από στοιχηματικές πλατφόρμες καθιστούν το Τένις ιδανικό άθλημα για στοιχηματικά παιχνίδια. [12] [13]

Ταυτόχρονα όμως, ίδια η δυνατότητα για πολλά αλλεπάλληλα παιχνίδια σε σύντομο χρονικό διάστημα συμπεριλαμβάνει ένα μεγάλο ρίσκο που εγκυμονεί σε όλα τα τυχερά παιχνίδια, δηλαδή την πιθανότητα εθισμού και χρεοκοπίας. Επιπλέον παράγοντα ρίσκου συμπεριλαμβάνουν οι **στημένοι αγώνες (match fixing),** δηλαδή η επιρροή του αποτελέσματος του αγώνα μέσω απειλής, δωροδοκίας ή άλλων παράνομων έμμεσων και άμεσων τακτικών. Η IBIA εκτιμά πως 25 εκατ. USD είναι η ζημιά ετησίως που οφείλεται σε στημένους αγώνες. [12]

1.2 Το άθλημα της αντισφαίρισης: Τένις

Η **Αντισφαίριση ή Τένις (Tennis)** είναι ένα σύγχρονο Ολυμπιακό άθλημα του 19ου αιώνα, με προέλευση από την Ευρώπη.

Ένας αγώνας διεξάγεται με δύο (ή και τέσσερις) παίκτες να σχηματίζουν δύο αντίπαλες ομάδες (για αγώνες μονού ή διπλού) εντός ενός ειδικού συμμετρικού γηπέδου σχήματος παραλληλογράμμου, με ένα δίχτυ να διαχωρίζει τον χώρο των δύο αντιπάλων στην μέση της μεγάλης πλευράς του γηπέδου.

Τα γήπεδα διαχωρίζονται ανάλογα με τον τύπο της επιφάνειας τους, σε σκληρά, χωμάτινα ή γήπεδα με χόρτο. Πραγματοποιούνται αναμετρήσεις τόσο σε ανοικτό όσο και σε κλειστό στάδιο, ανάλογα και με την εποχή.

Εικόνα 3: Στιγμιότυπο από έναν αγώνα Τένις ανάμεσα σε δύο αθλητές στους Ολυμπιακούς Αγώνες του Τόκιο το 2020. Πηγή: https://olympics.com/

Κάθε αγώνας χωρίζεται σε σετ, ενώ κάθε σετ διαμορφώνεται από αριθμό «games». Για να κερδίσει κάποιος ένα σετ, πρέπει να φτάσει πρώτος στα 6 νικηφόρα «games» έχοντας δύο διαφορά απ' τον αντίπαλο του. Σε περίπτωση που το σκορ φτάσει στο 5-5, το σετ ολοκληρώνεται στα 7 κερδισμένα. Στο ενδεχόμενο βέβαια, που υπάρξει νέα ισοπαλία σε 6-6, τότε διεξάγεται η

διαδικασία του «Tie break», όπου το σκορ κρίνεται στους 7 κερδισμένους πόντους (με δύο διαφορά).

Κάθε παίκτης μπορεί να χτυπήσει την μπάλα μόνο με τη ρακέτα του. Για να μετρήσει ένας πόντος, πρέπει η μπάλα να έχει ακουμπήσει τουλάχιστον μια φορά εντός των γραμμών του αντιπάλου και οπουδήποτε αλλού εκτός από ρακέτα. Σε αγώνες «Best of 3», ο παίκτης που θα κερδίσει πρώτος δύο σετ, είναι ο νικητής. Σε αγώνες «Best of 5», όπως σε πρωταθλήματα Grand Slam, ο νικητής κρίνεται στα 3 νικηφόρα σετ. Σε αυτό το άθλημα δεν μπορεί να υπάρξει ισοπαλία. Σε αγώνες διπλού, υπάρχει και το σύστημα «No Advantage», κατά το οποίο όταν ένα «game» φτάσει σε ισοπαλία 40-40, η ομάδα που υποδέχεται επιλέγει πλευρά και στον επόμενο πόντο, κρίνεται ο νικητής του «game». Ακόμα, σε αναμετρήσεις «διπλού», όταν το αποτέλεσμα είναι 1-1 σετ, τότε δεν διεξάγεται 3^{ο} σετ, αλλά η διαδικασία «Super Tie Break», το οποίο είναι παρόμοιο με την διαδικασία του απλού «Tie Break», με την διαφορά ότι η νίκη κρίνεται στους δέκα κερδισμένους πόντους.[14]

1.3 Πρόβλεψη Αποτελέσματος και Διαχείριση Ρίσκου σε Αγώνες Τένις

Οι συνήθεις παράμετροι που πρέπει να λάβει υπόψιν του ένας Παίκτης[1], ο οποίος θέλει να στοιχηματίσει σε αγώνες Τένις είναι:

- η Εκτίμηση του Αποτελέσματος ενός Αγώνα
- ο Τύπος Στοιχηματικού Παιχνιδιού
- ο Υπολογισμός Απόδοσης και η Επιλογή Κατάλληλης Στοιχηματικής Πλατφόρμας
- η Συνετή Διαχείριση Ρίσκου και Κεφαλαίου του Παίκτη.

Η Εκτίμηση Αποτελέσματος ενός αγώνα είναι μια αρκετά δύσκολη διαδικασία, ιδιαίτερα στο Τένις, καθώς οι αθλητές είναι δύο το πλήθος – αντίπαλοι μεταξύ τους – και ακόμα και αμυδρές επιπλοκές στην ψυχοσωματική τους κατάσταση ή στην συγκέντρωσή τους μπορεί να καθορίσουν το αποτέλεσμα του αγώνα. Επιπλέον παράγοντες που επηρεάζουν τον αγώνα, πέρα από την κατάσταση και την φόρμα του αθλητή, είναι:

- η επιφάνεια του γηπέδου – καθώς κάποιος αθλητής μπορεί να έχει πλεονέκτημα
- ο τύπος του τουρνουά – καθώς μια σημαντική διοργάνωση μπορεί να επηρεάσει την κράση και την αποφασιστικότητα του αθλητή
- το αθλητικό ιστορικό, το οποίο μπορεί να δείχνει αν ένας αθλητής τα πηγαίνει καλά αυτήν την περίοδο
- οι καιρικές συνθήκες, οι οποίες μπορεί να επηρεάζουν τις επιδόσεις των Αθλητών με διαφορετικό τρόπο
- αν ένας αθλητής είναι δεξιόχειρας, αριστερόχειρας ή αμφιδέξιος

και πολλοί άλλοι παράγοντες.

[1] Προς αποφυγήν σύγχυσης, στην υπόλοιπη εργασία «Παίκτης» θα ονομάζεται αυτός που στοιχηματίζει σε κάποιον αγώνα, ενώ ο αθλητής που αγωνίζεται θα αναφέρεται ως «Αθλητής».

Στην σύγχρονη βιβλιογραφία, η πρόβλεψη του νικητή μπορεί να γίνει με την χρήση υπολογιστών αλλά και αλγορίθμων Μηχανικής Μάθησης, οι οποίοι θα αναλυθούν στο επόμενο κεφάλαιο. Οι στοιχηματικές εταιρίες που βγάζουν τις αποδόσεις, χρησιμοποιούν αντίστοιχες μεθόδους, μόνο που λόγω της οικονομικής τους υπεροχής και πρόσβασης σε ανθρώπινο δυναμικό, δεδομένα και πόρους, μπορούν να εκτιμήσουν με καλύτερη ακρίβεια τον νικητή Αθλητή σε σύγκριση με κάποιον μεμονωμένο ιδιώτη Παίκτη, με κάποια στατιστικά σημαντική επανάληψη. [13] [15]

Υπάρχουν διάφοροι Τύποι Στοιχηματικού Παιχνιδιού. Ο πιο γνωστός είναι η εκτίμηση του νικητή Αθλητή στοιχηματίζοντας - «παίζοντας» σε ένα και μόνο αθλητικό γεγονός. Αυτός ο τύπος αποκαλείται **Μονό Αποδεκτό Στοίχημα (Single Bet)**. Ένας άλλος τύπος, είναι το Στοίχημα **Παρολί (Parlay ή Accumulator Bet).** Σε αυτόν τον τύπο στοιχήματος, ένας Παίκτης μπορεί να δηλώσει δύο ή και περισσότερα στοιχήματα στο ίδιο δελτίο/κουπόνι. Για να πληρωθεί όμως θα πρέπει όλα τα στοιχήματα που έλαβε μέρος να είναι επιτυχημένα. Σε αυτή την εργασία θα εστιάσουμε σε «Single Bet» Τύπους Παιχνιδιού.

Ο Υπολογισμός Κατάλληλης Πλατφόρμας και μέγιστης απόδοσης στην ουσία αποτελεί μια έρευνα αγοράς για τις αποδόσεις που δίνουν στις διάφορες πλατφόρμες οι διάφορες **Στοιχηματικές Εταιρίες (Bookmakers)**. Παράγοντες που επηρεάζουν τις αποδόσεις είναι:

- η πρόσβαση που έχει ο παίκτης σε διάφορες πλατφόρμες
- ο χρόνος που θα αφιερώσει στην αναζήτηση της καλύτερης απόδοσης
- τα εργαλεία και οι πόροι που έχει στην διάθεσή του
- η συμμετοχή κερδών των εταιριών στα κέρδη ή αλλιώς η γκανιότα
- οι αποδόσεις που έχουν θέσει οι bookmakers με βάση τα δεδομένα που έχουν στην διάθεσή τους. [13] [15]

Η Συνετή Διαχείριση Ρίσκου και Κεφαλαίου αποτελεί από μόνη της έναν ολόκληρο ξεχωριστό κλάδο των Οικονομικών Επιστημών. Τόσο ο **Στοιχηματισμός (Betting),** όσο και η **Διαχείριση Οικονομικών Επενδύσεων (Financial Investment)** χρησιμοποιούν κοινή ορολογία από τα Οικονομικά, όπως **«επένδυση» (investment), «διασφάλιση» (safeguard), «πλάνο δράσης» (plan)** και **«ασφάλεια» (insurance),** καθώς και οι δύο έννοιες κάνουν εκτιμήσεις για το μέλλον και παίρνουν ρίσκα με σκοπό να διασφαλίσουν και να μεγιστοποιήσουν τα κέρδη τους. Παρόλα αυτά, ένας επενδυτής χαίρει κοινωνικής εκτίμησης, ενώ ο Στοιχηματισμός αντιμετωπίζεται ως δραστηριότητα χαμηλότερης κοινωνικής τάξης.

Η διαφορά έγκειται στο πως οι Οικονομικοί Παίκτες ζυγίζουν και αντιμετωπίζουν το ρίσκο, με επαγγελματίες επενδυτές να αναλύουν και να περιορίζουν τα ρίσκα που παίρνουν αξιοποιώντας εξειδικευμένους επιστημονικούς συμβούλους και εργαλεία, ενώ αντίθετα πολλοί Παίκτες του Στοιχηματισμού φαίνεται να παίρνουν αχρείαστα ρίσκα, με περισσότερο συναισθηματικά κριτήρια, όπως ενθουσιασμό, παρά κάποιο πλάνο δράσης ή κάποια λογική ανάλυση των πιθανοτήτων και των ποσών των κερδών τους και των ζημιών που θα υποστούν σε περίπτωση ήττας. Στην περίπτωση που ο Στοιχηματισμός αντιμετωπιστεί εξίσου υπεύθυνα με μία οικονομική επένδυση, μπορεί να μελετηθεί εξίσου καλά με τα εργαλεία των μαθηματικών,

οικονομικών και των υπολοίπων επιστημών που έχουν στην διάθεσή τους οι ειδικοί στον χώρο των Επιχειρησιακών Προβλέψεων και Διαχείρισης Οικονομικών Επενδύσεων. [8] [13] [15]

1.4 Στοιχηματικές Στρατηγικές

Ο κόσμος του στοιχήματος βρίθει από τακτικές και μοντέλα, τόσο ανάμεσα σε κοινούς παίκτες όσο και ανάμεσα σε επιχειρηματικούς ομίλους και ακαδημαϊκούς κύκλους. Σε αυτή την εργασία θα εστιάσουμε σε τρεις το πλήθος στρατηγικές, οι οποίες είναι:

- Ο Αφελής Στοιχηματισμός
- Ο Στοιχηματισμός στα Φαβορί
- Η Μέθοδος Martingale

1.4.1 Αφελής Στοιχηματισμός

Η πιο απλή προσέγγιση είναι **ο Αφελής Στοιχηματισμός (Naïve Betting, Gambling)**, υπό την έννοια του ότι ο Παίκτης ποντάρει εντελώς τυχαία, χωρίς να λάβει κανένα δεδομένο υπόψιν του. Καθώς αυτή η πρακτική δεν έχει κάποιο λογικό υπόβαθρο, παρότι είναι εντός των επιλογών ενός στοιχηματικού παίκτη, δεν μπορούμε να την θεωρήσουμε στοιχηματική στρατηγική και σίγουρα δεν αποτελεί συνετή διαχείριση ρίσκου που θα αποφέρει κέρδος με ασφάλεια, αφού η προσέγγιση για ένα θετικό αποτέλεσμα, έγκειται καθαρά στην τύχη. [14] [11] [12]

1.4.2 Στοιχηματισμός στα Φαβορί

Μια κάπως καλύτερη, αλλά εξίσου απλή, στρατηγική είναι **ο Στοιχηματισμός στα Φαβορί (Betting on Favorites)**, δηλαδή στους Αθλητές που εκτιμάται πως θα νικήσουν τον αγώνα. Εκτίμηση όμως δεν πραγματοποιεί μόνο ο Παίκτης που θα στοιχηματίσει, αλλά και οι Bookmakers που ορίζουν τις **Αποδόσεις (Rate of Return)**.

Για παράδειγμα, σε μια στοιχηματική πλατφόρμα ένας αγώνας Α μπορεί να έχει απόδοση 1.2, το οποίο σημαίνει ότι σε περίπτωση που ο Παίκτης στοιχηματίσει στον νικητήριο Αθλητή, για κάθε ένα ευρώ που πόνταρε θα λάβει 1.20€. Κάποιος άλλος αγώνας Β μπορεί να έχει απόδοση 2.1 το οποίο σημαίνει ότι σε περίπτωση που ο Παίκτης στοιχηματίσει στον νικητήριο Αθλητή, για κάθε ένα ευρώ που πόνταρε, θα λάβει 2.10€. Προφανώς, σε περίπτωση πονταρίσματος στον αθλητή που έχασε, ο Παίκτης χάνει το ακριβές ποσό των χρημάτων που πόνταρε. Το γεγονός ότι οι Bookmakers δίνουν 1.2 στον αγώνα Α και 2.1 στον αγώνα Β, δείχνει πως ο Αθλητής του αγώνα Α θεωρείται Φαβορί, ενώ ο Αθλητής του αγώνα Β θεωρείται **στοιχηματικό Αουτσάιντερ (Outsider** ή **Underdog)**, δηλαδή Αθλητής που εκτιμάται από τους Bookmakers πως θα χάσει. Η μέθοδος Στοιχηματισμού στα Φαβορί λοιπόν, ουσιαστικά απαιτεί από έναν Παίκτη να παίζει μόνο στους Αθλητές που οι Bookmakers πιστεύουν ότι θα νικήσουν, δηλαδή στους αγώνες που έχουν χαμηλές αποδόσεις και τις μικρότερες πιθανότητες αποτυχίας. [14] [15]

Να σημειωθεί πως παρά τις σχετικά χαμηλές πιθανότητες για αποτυχία πρόβλεψης του νικητή της στρατηγικής των Φαβορί, στον αθλητισμό, και ιδιαίτερα στο τένις, οι εκτιμήσεις είναι δύσκολο να γίνουν με ακρίβεια. Συνεπώς, κάποια από τα υποτιθέμενα Φαβορί δεν θα είναι πράγματι νικητές. Είναι αρκετά πιθανό λοιπόν, ένας Παίκτης να νικήσει 9 στους 10 αγώνες και

παρόλα αυτά να έχει τελικά ζημιά. Αυτό συμβαίνει γιατί οι αγώνες είχαν πολύ χαμηλή απόδοση σε σύγκριση με τα χρήματα που στοιχηματίστηκαν ανά αγώνα. Για παράδειγμα, αν όλοι οι αγώνες είχαν απόδοση 1.1, και με δεδομένο ότι ο Παίκτης στοιχημάτιζε με 1€ κάθε φορά, αυτό σημαίνει ότι για κάθε ένα ευρώ που πόνταρε, ο Παίκτης είχε καθαρό κέρδος 0.10€. Αφού κέρδισε 9 αγώνες, τα κέρδη του τότε ήταν 0.90€. Σε έναν αγώνα όμως έχασε, άρα έχασε μια φορά 1.00€. Άρα, τα τελικά κέρδη βγαίνουν,

$$(Κέρδη) - (Ζημίες) = (Τελική\ Μεταβολή)$$

$$9 \cdot 0.10€ - 1 \cdot 1.00€ =$$

$$0.90€ - 1.00€ = -0.10€$$

δηλαδή, αντί για κέρδη, τελικά ο Παίκτης είχε ζημιά 0.10€, παρά το πλήθος των αγώνων που κέρδισε. Αυτό είναι το τρωτό σημείο της στρατηγικής των Φαβορί, το οποίο θα το αναλύσουμε και πειραματικά σε επόμενα κεφάλαια. Αυτή η στρατηγική θα αποτελέσει την **Βασική Στρατηγική Πρόβλεψης (Benchmark Prediction)** και ταυτόχρονα τη **Βασική Στρατηγική Σύγκρισης (Benchmark Strategy)**, δηλαδή την μέθοδο με βάση την οποία θα εκτιμήσουμε όλες τις υπόλοιπες μεθόδους πρόβλεψης και στρατηγικής.

Στους υπολογισμούς αυτούς θα πρέπει να συμπεριληφθεί και η **Γκανιότα (Vigorish)**. Στην ουσία πρόκειται για μια εταιρική προμήθεια που ορίζουν οι Bookmakers εις βάρος των κερδών του Παίκτη, ώστε να εξασφαλιστεί ότι θα παραμείνουν κερδισμένες ανεξαρτήτως αποτελέσματος. Ο πιο διαδεδομένος μαθηματικός τύπος για την γκανιότα είναι ο εξής:

$$Vigorish = 1 - \frac{1}{\frac{1}{RoRA1} + \frac{1}{RoRD} + \frac{1}{RoRA2}}$$

Όπου,

- $Vigorish$ η γκανιότα
- $RoRA1$ η απόδοση που δίνει η εταιρία για τον Αθλητή 1
- $RoRD$ η απόδοση που δίνει η εταιρία για ισοπαλία
- $RoRA2$ η απόδοση που δίνει η εταιρία για τον Αθλητή 2

Καθώς, όπως έχουμε ήδη αναφέρει, στο Τένις δεν υπάρχει ισοπαλία, ο παραπάνω τύπος μετατρέπεται στη μορφή:

$$Vigorish = 1 - \frac{1}{\frac{1}{RoRA1} + \frac{1}{RoRA2}}$$

Με αυτόν τον τρόπο προκύπτει το ποσοστό της γκανιότας, που αντιστοιχεί και στην προμήθεια των Bookmaker. Για παράδειγμα, σε έναν αγώνα με αποδόσεις 1.5 στον άσο (δηλαδή στην περίπτωση που νικήσει ο Αθλητής 1) και 2.5 στο διπλό (δηλαδή στην περίπτωση που νικήσει ο Αθλητής 2), τότε διαμορφώνεται η εξής μαθηματική πράξη για να ορίσουμε τη γκανιότα:

$$1 - \left(\frac{1}{1.50} + \frac{1}{2.50}\right) \cdot 100\% = 6.25\%.$$ [16]

1.4.3 Μέθοδος Martingale

Μία άλλη στρατηγική, που είναι ευρέως γνωστή στη βιβλιογραφία, είναι **η Μέθοδος** ή **το Μοντέλο Μάρτινγκεϊλ (Martingale Model)**.

Κατά την μέθοδο αυτή, ένας Παίκτης ποντάρει συνεχόμενα μέχρι να κερδίσει μία φορά, αυξάνοντας συνεχώς το ποσό των χρημάτων που ποντάρει στους επόμενους αγώνες, συμπεριλαμβάνοντας το ποσό των χρημάτων που έχασε. Υπό την προϋπόθεση ότι υπάρχει πιθανότητα να νικήσει ο Παίκτης έστω μία φορά και πως η απόδοση θα είναι τουλάχιστον της τάξης του 1.00 ή μεγαλύτερη, ο παίκτης είναι μαθηματικά βέβαιο και υπολογίσιμο πως θα λάβει τα χρήματα του πίσω μετά από N αγώνες, με το N να εξαρτάται από τις πραγματικές πιθανότητες των αγώνων και τις αποδόσεις τους. Και αυτή η στρατηγική όμως είναι δύσκολο να εφαρμοστεί.

Παρότι δεν είναι τόσο αφελής όσο οι προηγούμενες, και παρά το γεγονός ότι είναι μαθηματικά βέβαιη η θεωρητική επιστροφή των χρημάτων που στοιχηματίστηκαν, και μάλιστα με εξασφαλισμένο κέρδος, το πλήθος των αγώνων που μεσολαβεί μέχρι την τελική νίκη μπορεί να είναι αποτρεπτικό. Είτε λόγω χρόνου εξαιτίας του τεράστιου πλήθους αγώνων, είτε λόγω έλλειψης κεφαλαίου προς επένδυση σε επόμενα στοιχήματα λόγω της γεωμετρικής αύξησής του, ένας Παίκτης σε ένα ρεαλιστικό σενάριο θα αναγκαζόταν πιθανώς να παραιτηθεί. Πόσο μάλλον δε, αν ληφθεί υπόψιν και ο κίνδυνος των στημένων παιχνιδιών, ο οποίος θα αυξήσει παραπάνω τον εκτιμώμενο αριθμό αγώνων μέχρι την αναμενόμενη νίκη. Παρόλα αυτά, το μοντέλο αυτό είναι αρκετά υποσχόμενο λόγω της μαθηματικής του ακρίβειας και θα μελετηθεί επίσης πειραματικά σε επόμενα κεφάλαια. [9] [12]

1.5 Μετρικές Αξιολόγησης Στοιχηματικών Στρατηγικών

Για να μπορέσουμε να αποφανθούμε για το ποια στρατηγική υπερτερεί των υπολοίπων θα χρειαστεί να ορίσουμε κάποιες **Μετρικές[2] (Metrics),** οι οποίες ποσοτικοποιούν την απόδοση των στρατηγικών που χρησιμοποιήθηκαν. Στην παρούσα εργασία χρησιμοποιούνται δύο μετρικές στρατηγικών, η μετρική Διαφοράς Κεφαλαίου και η μετρική Καθαρού Κέρδους.

1.5.1 Μετρική Διαφοράς Κεφαλαίου

Η μετρική **Διαφοράς Κεφαλαίου (Net Change in Capital, NCC)** ουσιαστικά υπολογίζει την μεταβολή των χρημάτων μεταξύ δύο χρονικών περιόδων και δίνεται από τη διαφορά:

[2] Προς αποφυγήν σύγχυσης, διαφοροποιούμε τις Μετρικές των Στοιχηματικών Στρατηγικών που παρουσιάζονται σε αυτό το κεφάλαιο, από τις Μετρικές Απόδοσης Μοντέλων Μηχανικής Μάθησης που παρουσιάζονται στο επόμενο κεφάλαιο.

$$NCC = C(t_2) - C(t_1), \qquad t_1 < t_2$$

Όπου,

- NCC η Διαφορά Κεφαλαίου
- $C(t)$ το ύψος του Κεφαλαίου την χρονική περίοδο t
- t_1, t_2 δύο χρονικά στιγμιότυπα

Θεωρώντας ως πρώτο χρονικό στιγμιότυπο, μια χρονική στιγμή πριν την έναρξη, και έχοντας αντίστοιχα ως δεύτερο χρονικό στιγμιότυπο το πέρας της στοιχηματικής δραστηριότητας ο παραπάνω τύπος παίρνει την μορφή,

$$NCC = C(t_{end}) - C(t_{start}), \qquad t_1 < t_2$$

Αυτή η μετρική απεικονίζει πόσα χρήματα κέρδισε ή έχασε ένας παίκτης στο τέλος της στοιχηματικής του δραστηριότητας, σε σχέση με το αρχικό του κεφάλαιο.

1.5.2 Μετρική Απόδοσης Επενδύσεων

Η απόδοση επενδύσεων **(Rate of Return, ROI)** είναι μια μετρική που υπολογίζει τα κέρδη ως προς τα χρήματα που δαπανήθηκαν. Συγκεκριμένα πρόκειται για τον λόγο:

$$ROI = \frac{\sum \quad (gains(t_i) - losses(t_i))}{\sum \quad losses(t_i)}$$

Όπου,

- ROI η Απόδοση Επενδύσεων
- t_i η i-οστή χρονική στιγμή
- $gains(t_i)$ τα κέρδη τη χρονική στιγμή i
- $losses(t_i)$ τα κόστη/ζημίες τη χρονική στιγμή i

Κεφάλαιο 2: Μηχανική Μάθηση και Ταξινομητές Δέντρων Αποφάσεων

2.1 Εισαγωγή στη Μηχανική Μάθηση

Η **Μάθηση (Learning)** είναι μία από τις πιο χαρακτηριστικές ιδιότητες των νοημόνων όντων. Οι μηχανισμοί, υπό τους οποίους η λειτουργία της μάθησης είναι εφικτή, έχουν απασχολήσει αρκετά επιστήμονες, ψυχολόγους και φιλόσοφους, αλλά εξακολουθούν να μην είναι πλήρως κατανοητοί από την Γνωστική Ψυχολογία, τον αρμόδιο δηλαδή επιστημονικό κλάδο της Ψυχολογίας που μελετά το ζήτημα. Κατά την **Επαγωγική Μάθηση (Inductive Learning)**, ο άνθρωπος, μέσω του συνδυασμού των αισθητήριων οργάνων του, της επεξεργαστικής του δύναμης και αφαιρετικής του ικανότητας, δημιουργεί μια απλουστευμένη απεικόνιση του κόσμου, δηλαδή ένα **Νοητικό Μοντέλο (Mental Model)**. Παράλληλα, ο άνθρωπος μπορεί να οργανώνει νέες δομές με βάση τις εμπειρίες και τις παρατηρήσεις του, δημιουργώντας **Νοητικά Πρότυπα (Mental Patterns)**, συσχετίζοντας επαγωγικές και απαγωγικές συλλογιστικές. Συνεπώς, ο υποκλάδος της **Τεχνητής Νοημοσύνης (Artificial Intelligence, AI)** της **Επιστήμης Υπολογιστών (Computer Science),** εμπνευσμένη από την Γνωστική Ψυχολογία, προσπαθεί να εργαλειοποιήσει, ποσοτικοποιήσει και ψηφιοποιήσει αυτές τις νοητικές διεργασίες. Το αποτέλεσμα αυτού του εγχειρήματος είναι η **Μηχανική Μάθηση**.

Ακολουθεί οι ορισμός της Μηχανικής Μάθησης, όπως απαντάται στο βιβλίο «Τεχνητή Νοημοσύνη» της Κατερίνας Γεωργούλη [17]:

> *«Μηχανική Μάθηση ονομάζεται η ικανότητα ενός υπολογιστικού συστήματος να δημιουργεί μοντέλα ή πρότυπα από ένα σύνολο δεδομένων... Ένα πρόγραμμα υπολογιστή λέμε ότι μαθαίνει από την εμπειρία E ως προς κάποια κλάση εργασιών T και μέτρο απόδοσης P, αν η απόδοση του σε εργασίες από το T, όπως μετριέται από το P, βελτιώνεται μέσω της εμπειρίας E.»*

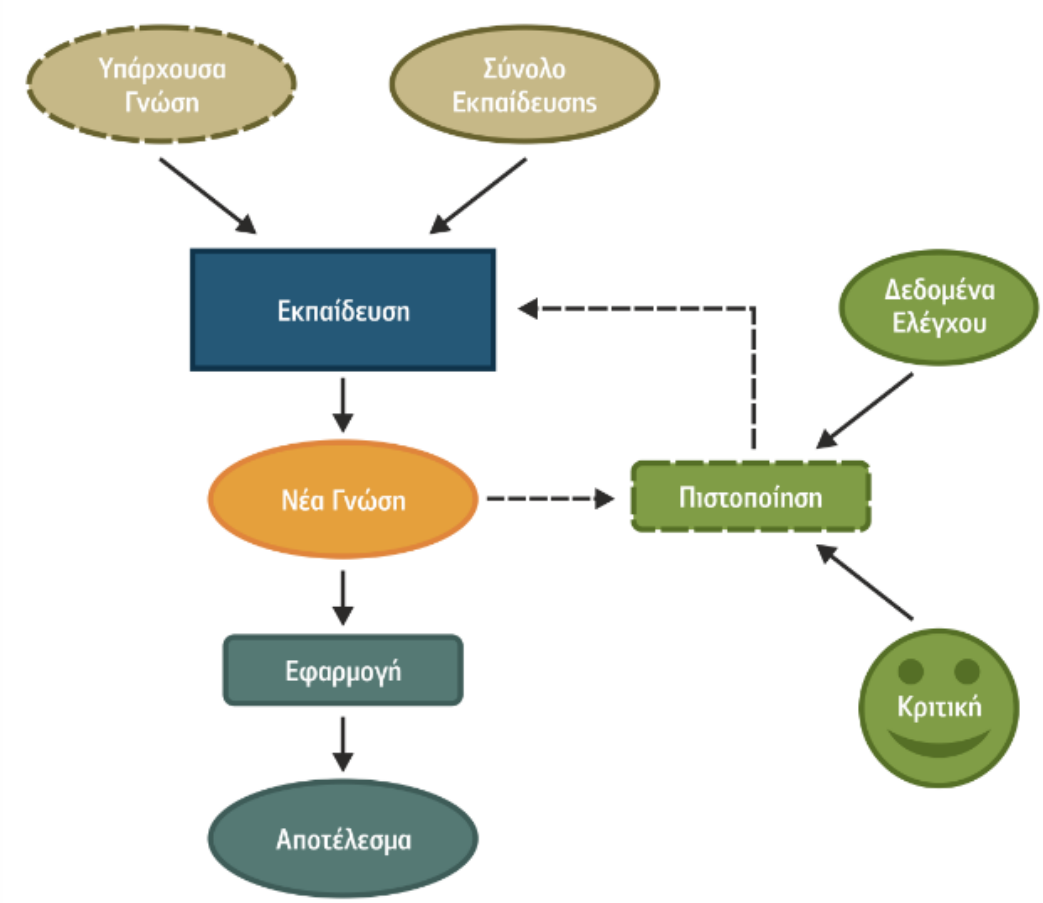
Κάθε αλγόριθμος Μηχανικής Μάθησης δέχεται ως είσοδο ένα **Σύνολο Δεδομένων Εκπαίδευσης (Training Dataset)**. Τα δεδομένα αυτά εισάγονται σε μια διαδικασία **Εκπαίδευσης (Training)** ή **Προσαρμογής (Fitting)**, όπως επίσης απαντάται στην βιβλιογραφία, υπό την έννοια του ότι ο αλγόριθμος προσαρμόζει την συνάρτηση που κατασκευάζει με βάση τα δεδομένα που δέχτηκε.

Η νέα συνάρτηση που κατασκευάζεται αποτελεί ένα νέο Νοητικό Μοντέλο γνώσης, το οποίο ονομάζεται **Μοντέλο (Model)**. Το μοντέλο που προκύπτει, για να εφαρμοστεί, θα πρέπει να δεχθεί σαν είσοδο νέα δεδομένα, είτε από αυτά που είχε εκπαιδευτεί, είτε εντελώς καινούρια, από τα οποία θα παραχθεί ένα αποτέλεσμα, το οποίο θα προκύψει από επαγωγή των νέων δεδομένων και βάση της εκπαίδευσης επάνω στα αρχικά δεδομένα εκπαίδευσης τα οποία δέχτηκε.

Προτού βγει το μοντέλο όμως στην παραγωγή, το μοντέλο που έχει προκύψει από το στάδιο εκπαίδευσης, περνά από ένα **Στάδιο Πιστοποίησης (Validation)**, το οποίο στάδιο, είτε είναι μια διαδικασία εμφολευμένη στον ίδιο τον αλγόριθμο, είτε έγκειται στην ευχέρεια του προγραμματιστή. Αυτό το στάδιο είναι απαραίτητο επειδή τα λογικά συμπεράσματα μιας επαγωγικής διαδικασίας δεν είναι πάντα ορθά[3]. Κατά το στάδιο της Πιστοποίησης, ένα σύνολο δεδομένων έχει κρατηθεί ως **Σύνολο Δεδομένων Ελέγχου (Test Dataset)**, το οποίπ χρησιμοποιείται ως στατιστικό δείγμα των αποτελεσμάτων του παραχθέντος μοντέλου. Για να είναι αξιόπιστη η διαδικασία πιστοποίησης, θα πρέπει τα δεδομένα ελέγχου να είναι ένα αντιπροσωπευτικό στατιστικό δείγμα και η διαδικασία πιστοποίησης να είναι μεθοδική, καθώς είναι εύκολο να υπάρξει παρανόηση από λανθασμένη επιλογή δεδομένων ελέγχου ή διαδικασίας πιστοποίησης. Τις διαδικασίες πιστοποίησης τις μελετά εκτενώς ο κλάδος της **Επιστήμης Δεδομένων (Data Science)**.

Να σημειωθεί επίσης σε αυτό το σημείο ότι τόσο η συλλογή, η επεξεργασία και εμπορική χρήση δεδομένων όσο και η Τεχνητή Νοημοσύνη εγείρουν ηθικά και νομικά ζητήματα, τα οποία μελετούν οι κλάδοι της Φιλοσοφίας και της Νομικής αντίστοιχα.

[3] Παραδείγματος χάριν, «Ένας Αθλητής έχει νικήσει σε κάθε αγώνα που έχει παίξει, συνεπώς θα νικήσει και στον επόμενο». Αυτή η δήλωση είναι ένα παράδειγμα Προβλεπτικής Επαγωγής (Predictive Induction), η οποία, εν τέλει, μπορεί και να μην ισχύει. Κάθε αγώνας θεωρείται ανεξάρτητο στατιστικό γεγονός. Για να κάνουμε μια καλύτερη πρόβλεψη για την επίδοση του Αθλητή σε επόμενους αγώνες, χρειαζόμαστε περισσότερα δεδομένα, όπως ποιος είναι ο αντίπαλός του, το πρόσφατο ιστορικό τραυματισμών του, κτλ.



Εικόνα 4: Το γενικευμένο κλειστό σύστημα ροής των σταδίων της Μηχανικής Μάθησης. Τα σχήματα και διαδρομές με διακεκομμένες γραμμές δεν απαντώνται απαραίτητα σε κάθε περίπτωση. Πηγή: «Τεχνητή Νοημοσύνη» της Κατερίνας Γεωργούλη. [17]

Η Μηχανική μάθηση με την σειρά της μπορεί να ταξινομηθεί σε τρείς κατηγορίες ανάλογα με τις λειτουργίες που καλείται να επιτελέσει. Οι κατηγορίες αυτές είναι η **Επιβλεπόμενη Μάθηση (Supervised Learning)**, η **Μη-Επιβλεπόμενη Μάθηση (Unsupervised Learning)** και η **Ενισχυτική Μάθηση (Reinforcement Learning)**.

Κατά την **Επιβλεπόμενη Μάθηση**, η οποία θα μας απασχολήσει σε αυτή την εργασία, ο αλγόριθμος κατασκευάζει μια συνάρτηση που απεικονίζει δεδομένα εισόδου του **Συνόλου Εκπαίδευσης (Train Set)** σε γνωστές επιθυμητές εξόδους, με απώτερο στόχο τη γενίκευση της συνάρτησης αυτής για εισόδους με άγνωστη έξοδο. Η Επιβλεπόμενη Μάθηση χρησιμοποιείται σε προβλήματα **Ταξινόμησης (Classification)**, **Πρόγνωσης (Prediction)** αλλά και **Διερμηνείας (Interpretation).**

Κατά τη **Μη-Επιβλεπόμενη Μάθηση**, ο αλγόριθμος κατασκευάζει ένα μοντέλο για κάποιο σύνολο εισόδων υπό την μορφή παρατηρήσεων, χωρίς να γνωρίζει τις επιθυμητές

εξόδους. Απαντάται σε προβλήματα **Ανάλυσης Συσχετισμών (Association Analysis)** και **Ομαδοποίησης (Clustering).**

Κατά την **Ενισχυτική Μάθηση**, ο αλγόριθμος μαθαίνει μια στρατηγική ενεργειών μέσα από άμεση αλληλεπίδραση με το περιβάλλον. Χρησιμοποιείται κυρίως σε προβλήματα **Σχεδιασμού (Planning)**, όπως για παράδειγμα ο **έλεγχος κίνησης ρομπότ (Robotics & Control Systems)** και βελτιστοποίηση εργασιών σε **εργοστασιακούς χώρους (Industrial Optimizations).**

Οι **Αλγόριθμοι Επιβλεπόμενης Επαγωγικής Μάθησης (Supervised Inductive Learning Algorithms)** αναπτύχθηκαν με σκοπό να καταλήξουν σε επικρατούσες αποφάσεις που προκύπτουν από μια δειγματοληψία παρατηρήσεων σε **Προβλήματα Ταξινόμησης (Classification Problems)** και **Παρεμβολής (Regression Problems)**. Τα αποτελέσματα των αλγορίθμων αυτών είναι μοντέλα πρόβλεψης διακριτών τάξεων. Το παραγόμενο μοντέλο είναι μια **Συνάρτηση Πρόγνωσης (Predictor Function)**, στόχος της οποίας είναι οι απεικόνιση των δεδομένων εισόδου σε προκαθορισμένες επιθυμητές εξόδους. Απώτερος στόχος είναι η γενίκευση της συνάρτησης αυτής και σε άγνωστα δεδομένα, δηλαδή να είναι σε θέση να απεικονίσει στη σωστή έξοδο και δεδομένα εισόδου που δεν είχε δεχθεί κατά την εκπαίδευση.

Ας δούμε λίγο πιο αναλυτικά λοιπόν πως δομείται μια συνάρτηση πρόγνωσης. Το πεδίο ορισμού της συνάρτησης είναι κάθε **Δείγμα (Sample)** που η συνάρτηση μπορεί να δεχθεί σαν είσοδο. Μια είσοδος είναι μια αλληλουχία δεδομένων, των οποίων το πλήθος και η σειρά έχει ορισθεί από τα δεδομένα του προβλήματος. Αυτή η αλληλουχία δεδομένων θα μπορούσε να χαρακτηριστεί ως μια **Λίστα (List)** ή ένα **Πολυδιάστατο Διάνυσμα (Multivariable Vector)** χρησιμοποιώντας τις έννοιες του Προγραμματισμού και της Γραμμικής Άλγεβρας, αντίστοιχα. Κάθε μεταβλητή σε αυτό το Πολυδιάστατο Διάνυσμα ενός Δείγματος αποτελεί ένα **Χαρακτηριστικό (Feature)** ή αλλιώς ένα **Γνώρισμα (Attribute)** και αντιπροσωπεύει κάποιο δεδομένο του Δείγματος που θα χρησιμοποιηθεί από την συνάρτηση πρόγνωσης. Ένα Δείγμα συνεπώς μπορεί να έχει πολλά Χαρακτηριστικά. Κάθε Δείγμα όμως έχει παραπάνω πεδία, ένα ή περισσότερα, αλλά ιδανικά συνήθως λιγότερα από τα Χαρακτηριστικά.

Τα πεδία αυτά αποκαλούνται **Ετικέτες (Labels)** και αναπαριστούν τις επιθυμητές τιμές που θέλουμε να έχει σαν έξοδο η Συνάρτηση Πρόγνωσης έχοντας σαν είσοδο το συγκεκριμένο Δείγμα. Το σύνολο όλων των δειγμάτων καλείται **Ολικό Σύνολο Δεδομένων (Total Dataset)**. Τα Δείγματα του Ολικού Συνόλου Δεδομένων χωρίζονται σε **Σύνολο Δεδομένων Εκπαίδευσης (Train Dataset)** και **Σύνολο Δεδομένων Ελέγχου (Test Dataset)**. Η συνάρτηση που απεικονίζει τα Χαρακτηριστικά ενός Δείγματος εισόδου σε μια τιμή εξόδου, καλείται **Συνάρτηση Στόχου (Goal Function)**.

Η τιμή που επιστρέφει η Συνάρτηση Στόχου βάσει των Χαρακτηριστικών ενός Δείγματος Εισόδου λέγεται **Μεταβλητή Στόχου (Goal Variable)**. Κατά την εκπαίδευση στην Επιβλεπόμενη Μάθηση, μια **Συνάρτηση Απώλειας (Loss Function)** εντοπίζει την διαφορά της Μεταβλητής Στόχου των Χαρακτηριστικών ενός Δείγματος εισόδου, που εκτίμησε το μοντέλο, από την Ετικέτα του Δείγματος, δηλαδή την επιθυμητή έξοδο.

Σε μαθηματικό συμβολισμό:

- x, ένα δείγμα σε μορφή ενός διανύσματος q των Χαρακτηριστικών του
- $h(x;\ q)$, η συνάρτηση πρόγνωσης που θέλουμε να μάθουμε, δηλαδή το μοντέλο μας
- f, η συνάρτηση στόχου
- $D = \{(x, f(x))\}$, το σύνολο εκπαίδευσης με x Δείγματα και $f(x)$ ετικέτες
- $distance(\dots)$, η συνάρτηση που υπολογίζει την διαφορά της πραγματικής από την προβλεπόμενη τιμή

Έχουμε τα ζεύγη $D = \{(x, f(x))\}$, που είναι γνωστά, αλλά ο τρόπος, δηλαδή η συνάρτηση, με τον οποίο συσχετίζονται μας είναι άγνωστη. Αυτό που ψάχνουμε είναι μία συνάρτηση $h(x;\ q)$ που υλοποιεί μια γενικευμένη απεικόνιση από το x στο f.

Αν η $h(x;\ q)$ είναι «αρκετά κοντά» στο $f(x)$ για όλα τα δείγματα x του Συνόλου Δεδομένων Ελέγχου, συνεπάγεται πως τα Χαρακτηριστικά q είναι οι παράμετροι που πρέπει να ληφθούν υπόψιν από την Συνάρτηση Πρόβλεψης $h(x; q)$. Και η συνάρτηση Απώλειας μπορεί να δοθεί εμπειρικά ως:

$$E(h) = \sum_{x} \quad distance[h(x;\ q), f]$$

Η Συνάρτηση Απώλειας E επιστρέφει το άθροισμα όλων των διαφορών που αφορούν τα ζεύγη εκπαίδευσης μέσα στο D.

Ένα από τα πολύ βασικά προβλήματα της Μηχανικής Μάθησης είναι πως τα μοντέλα που προκύπτουν πολύ συχνά είναι πολύ επιρρεπή στα δεδομένα που είχαν διαθέσιμα, με αποτέλεσμα να μην μπορούν να γενικεύσουν ή να κάνουν ακόμα και τελείως λάθος εκτιμήσεις. Αυτό το φαινόμενο ονομάζεται **Υπερπροσαρμογή (Overfitting)** και η σωστή επιλογή και επεξεργασία του Συνόλου δεδομένων, αλλά και τρόπου εκπαίδευσης των μοντέλων μπορεί να το περιορίσει.

Η παρούσα εργασία θα εξετάσει Προβλήματα Ταξινόμησης με χρήση Αλγορίθμων Επιβλεπόμενης Επαγωγικής Μάθησης. Για να καταστούν οι προηγούμενες έννοιες προσιτές στον αναγνώστη, στόχος του πρώτου βήματος της εργασίας είναι από ένα Σύνολο Δεδομένων Δειγμάτων Αγώνων Τένις, να εξαχθεί μια συνάρτηση πρόβλεψης της Ετικέτας Νικητή Αθλητή, όπου «1» αντιστοιχεί σε νίκη για τον πρώτο Αθλητή και «2» για τον δεύτερο Αθλητή. Το σύνολο Δεδομένων αυτό, θα χωρισθεί για την εκπαίδευση και πιστοποίηση του μοντέλου σε Δείγματα Αγώνων Δεδομένων Εκπαίδευσης και Δείγματα Αγώνων Δεδομένων Ελέγχου αντίστοιχα. Το Δείγμα κάθε Αγώνα θα έχει Χαρακτηριστικά πεδία όπως, τα σετ που νίκησε κάθε Αθλητής, την κατάταξη του κάθε Αθλητή στο αντίστοιχο πρωτάθλημα, το είδος του εδάφους όπου έλαβε χώρα ο Αγώνας, κτλ.

Περισσότερη ανάλυση επάνω σε Σύνολο Δεδομένων θα γίνει μετέπειτα στο αντίστοιχο κεφάλαιο.

2.2 Ταξινομητής Δένδρου Απόφασης

Τα **Δέντρα Απόφασης (Decision Trees)** είναι ο γνωστότερος **Ταξινομητής (Classifier)**, δηλαδή **Αλγόριθμος Κατηγοριοποίησης (Classification Algorithm)**, επιβλεπόμενης Επαγωγικής Μάθησης και έχει εφαρμοστεί με επιτυχία σε πολλούς τομείς όπου απαιτείται ταξινόμηση. Πιο ενδεικτικά, στην αναγνώριση προσώπων από εικόνες, στην ιατρική διάγνωση περιστατικών, σε συστήματα στενευμένης διαφήμισης και προώθησης προϊόντων και διάφορες εφαρμογές **Εξόρυξης Γνώσης (Data Mining)**.

Όπως μαρτυρά το όνομα της μεθόδου, ο αλγόριθμος κατασκευάζει μια δενδροειδής μορφή αναπαράστασης πληροφορίας συνεχόμενης κατηγοριοποίησης με τις απολήξεις, δηλαδή τα φύλλα αυτού δένδρου, να αποτελούν τις **Κατηγορίες Ταξινόμησης (Classes)** ή Ετικέτες, όπως τις αποκαλέσαμε στο προηγούμενο κεφάλαιο. Χαρακτηριστικό αυτής της δομής είναι η αλληλουχία και το σύνολο των κανόνων που την απαρτίζουν, οι οποίοι ονομάζονται Κανόνες **Ταξινόμησης (Classification Rules).**

Για να μπορέσει να λειτουργήσει ένας αλγόριθμος Επαγωγικής Μάθησης, όπως αναφέραμε προηγουμένως, θα πρέπει να υπάρχει ένα επαρκές Σύνολο Δεδομένων Εκπαίδευσης. Το Σύνολο Δεδομένων Εκπαίδευσης απαρτίζεται από ένα σύνολο Δειγμάτων, όπου κάθε Δείγμα έχει προκαθορισμένο πλήθος Χαρακτηριστικών και διακριτών Ετικετών, δηλαδή Κατηγοριών Ταξινόμησης. Τελικός σκοπός της διαδικασίας εκπαίδευσης ενός αλγορίθμου Δέντρου Απόφασης είναι ο προσδιορισμός των Κανόνων Ταξινόμησης, υπό τους οποίους τα Δείγματα Ταξινομούνται στις αντίστοιχες ετικέτες. [17] [18] [19]

Μερικά πλεονεκτήματα των Δένδρων Αποφάσεων είναι [20]:

- Η εύκολη κατανόηση της αρχής λειτουργείας, λόγω της απλής λογικής του αλγορίθμου και των οπτικοποιήσεων τους με σχήματα.
- Το μικρό υπολογιστικό κόστος για προβλέψεις σε σχέση με το κόστος για την εκπαίδευσή τους
- Η δυνατότητα τους για γενικευμένη Ταξινόμηση ακόμα και για προβλήματα με περισσότερες από δύο Κατηγορίες-Ετικέτες.
- Η εύκολη ποσοτικοποίηση των επιδόσεών τους μέσω Στατιστικής Ανάλυσης

Μερικά μειονεκτήματα των Δένδρων Αποφάσεων είναι [20]:

- Η δημιουργία πολύπλοκων δέντρων που δεν γενικεύουν καλά τα δεδομένα.
- Η αστάθεια τους. Μικρές αλλαγές στα δεδομένα μπορεί να οδηγήσουν σε τελείως διαφορετικό δέντρο.
- Τα Δένδρα που προκύπτουν είναι **προκατειλημμένα (biased)** αν μια Κατηγορία υπερτερεί στο Σύνολο Εκπαίδευσης.

- Η εύρεση του βέλτιστου Δένδρου θεωρείται NP-πλήρες[4] πρόβλημα. Οι αλγόριθμοι που τα υλοποιούν είναι άπληστοι αλγόριθμοι και δεν υπάρχει μαθηματικά αποδείξιμη εγγύηση πως το Δένδρο που προκύπτει από μία εκπαίδευση είναι βέλτιστο.

2.2.1 Παράδειγμα Εκτέλεσης ενός Δένδρου Απόφασης

Για να γίνει αντιληπτή η αρχή λειτουργίας των Δένδρων Απόφασης παρουσιάζεται παρακάτω ένα απλοϊκό παράδειγμα, όπως αυτό απαντάται στο ίδιο βιβλίο της Κατερίνας Γεωργούλη. Έστω πως μια διαφημιστική εταιρία ψάχνει να βρει έναν άνθρωπο για να χρησιμοποιήσει ως φωτογραφικό μοντέλο.

Η διαφημιστική αφίσα θα τον απεικονίζει εκτεθειμένο σε μία ηλιόλουστη παραλία για να τονίσει την αποτελεσματικότητα του αντηλιακού προϊόντος που μισθώθηκε να προωθήσει. Για να είναι πιο πειστική η διαφημιστική καμπάνια, η διαφημιστική ομάδα αποφάσισε να κάνει μια έρευνα για να συλλέξει δεδομένα και να χρησιμοποιήσει Δένδρα Απόφασης για να εντοπίσει τα Χαρακτηριστικά που πρέπει να έχει το εικονιζόμενο μοντέλο για να μην καεί. Η έρευνα είχε την μορφή τόσο παρατηρήσεων όσο και ερωτηματολογίου, όπου θα σημειώνονται τα χαρακτηριστικά της φυσιολογίας τους, το αν χρησιμοποίησαν το προϊόν και το αποτέλεσμα από την έκθεσή τους στον ήλιο.

Τα αποτελέσματά της παρατίθενται παρακάτω:

Πίνακας 1: Λίστα Δειγμάτων Παραδείγματος

Δείγμα	Μαλλιά	Μάτια	Δέρμα	Αντηλιακό	Κάηκε
Σάρα	Ξανθά	Πράσινα	Σκούρο	Όχι	ΝΑΙ
Άννα	Ξανθά	Μπλε	Σκούρο	Ναι	ΟΧΙ
Νίκος	Καστανά	Μαύρα	Σκούρο	Ναι	ΟΧΙ
Άλεξ	Ξανθά	Μαύρα	Σκούρο	Όχι	ΝΑΙ
Νταϊάνα	Κόκκινα	Πράσινα	Σκούρο	Όχι	ΝΑΙ
Τάκης	Καστανά	Μπλε	Σκούρο	Όχι	ΟΧΙ
Καίτη	Καστανά	Πράσινα	Σκούρο	Όχι	ΟΧΙ
Γιάννης	Ξανθά	Μαύρα	Σκούρο	Ναι	ΟΧΙ

Στην προκειμένη περίπτωση, τα δείγματα είναι οι διάφοροι άνθρωποι που έλαβαν μέρος στο ερωτηματολόγιο, τα Χαρακτηριστικά είναι τα Μαλλιά, Μάτια, Δέρμα και το αν χρησιμοποίησαν το αντηλιακό, ενώ το αν κάηκαν ή όχι αποτελεί την Ετικέτα. Όσο αυξάνεται το σύνολο των

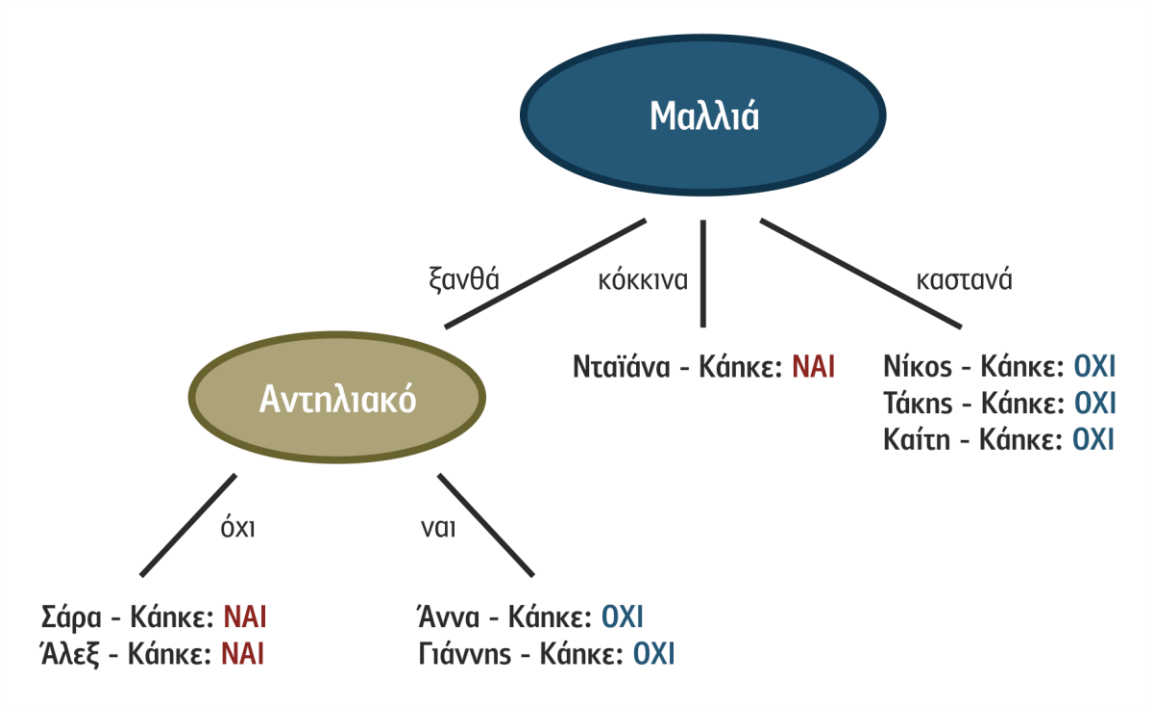
[4] Κατά την Επιστήμη Υπολογιστών τα προβλήματα διαχωρίζονται σε δυσκολία ανάλογα με την Πολυπλοκότητα τους, δηλαδή το υπολογιστικό τους κόστος. Απλά προβλήματα που έχουν το πολύ πολυωνυμική Πολυπλοκότητα για επίλυση ή απόδειξη καλούνται Πολυωνυμικά (Polynomial, P), ενώ πιο πολύπλοκα προβλήματα καλούνται Μη-Ντετερμινιστικά Πολυωνυμικά (Non-Deterministic Polynomial, NP). Δεν είναι πάντα εύκολη η κατηγοριοποίηση των προβλημάτων σε αυτές τις κατηγορίες.

Παραμέτρων, δηλαδή το πλήθος των Χαρακτηριστικών, αλλά και οι ποικίλες τιμές που αυτά μπορούν να πάρουν, οι πιθανοί συνδυασμοί αυξάνονται ραγδαία. Στο Σύνολο Δεδομένων που συλλέχθηκε απαντώνται τρία είδη Μαλλιών, τρία είδη Ματιών, δυο είδη δέρματος και δυο τιμές για το αντηλιακό, συνολικά $3 \cdot 3 \cdot 2 \cdot 2 = 36$ διαφορετικοί συνδυασμοί. Αν χρησιμοποιηθεί στοχαστικό μαθηματικό μοντέλο για να κάνει την ταξινόμηση, δεδομένου του συγκεκριμένου Συνόλου Δεδομένων των οκτώ ερωτηθέντων, μια νέα περίπτωση θα αναγνωριστεί επιτυχώς με πιθανότητα,

$$p = \frac{8}{36} \cong 0.22 = 22\%$$

Λόγω των γενικευμένων κανόνων που παράγουν τα Δέντρα Απόφασης, αποφεύγουν το πρόβλημα των πιθανοτήτων, χρησιμοποιώντας τις **Ευριστηκές (Heuristics)** που παρήγαγαν από το Σύνολο Δεδομένων στο οποίο εκπαιδεύτηκαν. Στο συγκεκριμένο παράδειγμα, ένα επιτυχημένο Δέντρο Απόφασης θα έχει την κάτωθι μορφή:

Εικόνα 5: Μια χρήσιμη αναπαράσταση του Συνόλου Δειγμάτων (χρωματιστές τελείες) και των χώρων απόφασης (διακεκομμένες γραμμές). Να σημειωθεί πως δεν μπορούν να αναπαρασταθούν αυτούσιοι χώροι μεγαλύτεροι από 3 διαστάσεις, χωρίς να καταφύγουμε σε κάποια μείωση διαστατικότητας, που όμως αλλοιώνει την γεωμετρία του χώρου δεδομένων. Ο χώρος των δειγμάτων με την γκρίζα διακεκομμένη γραμμή υποδεικνύει το σύνολο των ατόμων που κάηκαν (Σάρρα, Άλεξ, Νταϊάνα), ενώ ο υπόλοιπος χώρος είναι ο χώρος των υπόλοιπων ατόμων που δεν κάηκαν.



Εικόνα 6: Δένδρο Απόφασης για το πρόβλημα της διαφήμισης αντηλιακού. Πηγή: «Τεχνητή Νοημοσύνη» της Κατερίνας Γεωργούλη. [17]

Στο παραπάνω σχήμα γίνεται ξεκάθαρο γιατί η μέθοδος αποκαλείται Δέντρο Απόφασης. Πρακτικά, κάθε κόμβος του δέντρου αποτελεί μια διακλάδωση των κατηγοριών, βάσει της τιμής που παίρνουν τα Δείγματα για το εν λόγω Χαρακτηριστικό. Αν η διαδικασία γίνει ορθά, κάθε τελικό φύλλο του Δέντρου θα πρέπει να απαρτίζεται από δείγματα που να ανήκουν σε μία και μόνο Ετικέτα. Τα επαγωγικά, αλλά όχι νομοτελειακά αληθή συμπεράσματα που προκύπτουν από το αριστερότερο φύλλο, μπορούν να διατυπωθούν υπό μορφή ψευδοκώδικα ως εξής:

```
if(Μαλλιά == "Ξανθά" and Αντηλιακό == "ΟΧΙ"):
  Καίγεται = "ΝΑΙ"
```

Εικόνα 7: «ΑΝ (Μαλλιά Ξανθά) ΚΑΙ (Αντηλιακό ΟΧΙ) ΤΟΤΕ (ΚΑΙΓΕΤΑΙ ΝΑΙ)», ο ψευδοκώδικας που προκύπτει από το αριστερότερο φύλλο του Δέντρου Απόφασης της προηγούμενης εικόνας.

Αξίζει να σημειώσουμε εδώ πως η αλληλουχία των κατηγοριοποιήσεων δεν είναι τυχαία· διαφορετική σειρά των ερωτήσεων ανά Χαρακτηριστικό καταλήγει σε διαφορετική κατηγοριοποίηση. Η ρίζα, δηλαδή η κορυφή, του δέντρου είναι από τα Χαρακτηριστικά που έχει εκτιμήσει ο αλγόριθμος ως βέλτιστα σαν πρώτη επιλογή. Κάθε διακλάδωση έχει το όνομα ενός Χαρακτηριστικού, το οποίο απαγορεύεται να ξαναχρησιμοποιηθεί σε περεταίρω διακλάδωση.

Κάθε ακμή διακλάδωσης, φέρει μια τιμή που μπορεί να πάρει το χαρακτηριστικό και κάθε ακμή οφείλει να έχει διαφορετικό όνομα. Τέλος, κάθε φύλλο ιδανικά αντιστοιχεί σε μια κατηγορία ταξινόμησης.

Αυτή η διαδικασία θα μπορούσε να χαρακτηριστεί ως μια επαναλαμβανόμενη διχοτόμηση του Συνόλου Εκπαίδευσης, όπως ο αλγόριθμος **Επαναλαμβανόμενης Διχοτόμησης 3 (Iterative Dichotomiser 3, ID3)** που χρησιμοποιείται πλέον ευρέως για την δημιουργία δέντρων αποφάσεων. Αυτό που καθιστά τον ID3 ιδανικό είναι πως, ενώ διεκπεραιώνει την επαναλαμβανόμενη διχοτόμηση, ταυτόχρονα ψάχνει δοκιμάζοντας εξαντλητικά σε κάθε διακλάδωση κάθε Χαρακτηριστικό (που δεν έχει ήδη χρησιμοποιηθεί) το οποίο να διαχωρίζει καλύτερα τα Δείγματα. Αν το επιλεγμένο Χαρακτηριστικό διαχωρίζει πλήρως το Σύνολο Εκπαίδευσης στις Ορισμένες Ετικέτες, τότε ο αλγόριθμος τερματίζει, διαφορετικά, συνεχίζει στα υπόλοιπα παρακλάδια, που προκύπτουν από τις υπόλοιπες τιμές που μπορεί να πάρει το εν λόγω Χαρακτηριστικό. Το κάθε Χαρακτηριστικό επιλέγεται με βάση στατιστικά κριτήρια, όπως η **Εντροπία (Information Entropy)** ή **Εντροπία Shannon (Shannon Entropy)** και το **Κέρδος Πληροφορίας (Information Gain)**.

Η Εντροπία Πληροφορίας, δηλαδή ο βαθμός αβεβαιότητας ενός Συνόλου Δεδομένων, η οποία μετριέται σε bits πληροφορίας και υπολογίζεται από τη σχέση,

$$Entropy\ (S) = E(S) = \sum_{i=1}^{n} \quad - p_i\ log\ log\ (p_i)$$

Όπου,

- S το Σύνολο δεδομένων
- n το πλήθος Δειγμάτων του συνόλου
- p_i η πιθανότητα το i-οστό δείγμα να περιλαμβάνεται στο σύνολο S.

Στο παράδειγμα μας, για το χαρακτηριστικό «Μαλλιά» είχαμε:

Πίνακας 2: Χαρακτηριστικά-Ετικέτες

Μαλλιά	Κάηκε	ΔΕΝ Κάηκε	Σύνολα
Καστανά	0	3	3
Κόκκινα	1	0	1
Ξανθά	2	2	4
Συνολικά	3	5	8

Και η εντροπία κατά Shannon υπολογίζεται μέσω διωνυμικής προσέγγισης,

$$Entropy\ (Κάηκε, Μαλλιά) =$$

$$p(Καστανά) \cdot Entropy(0,3) + p(Κόκκινα) \cdot Entropy(1,0) + p(Ξανθά) \cdot Entropy(2,2) =$$

$$\frac{3}{8} \cdot 0 + \frac{1}{8} \cdot 0 + \frac{4}{8} \cdot 1 = 0.5\ bits$$

Το **κέρδος πληροφορίας** περιγράφει πόση πληροφορία περιέχεται σε ένα Χαρακτηριστικό και υπολογίζεται ως,

$$InfoGain(S, A) = G(S, A) = E(S) - \sum_{i=1}^{m} f_S(A_i) \cdot E\left(S_{A_i}\right)$$

Όπου,

- S το Σύνολο δεδομένων
- $E(S)$ η Συνάρτηση Εντροπίας για όλο το σύνολο S
- A_i η τιμή για το Χαρακτηριστικό στο i-οστό δείγμα του συνόλου S
- $f_S(A_i)$ το ποσοστό των δειγμάτων στο S που παίρνουν την τιμή A_i

Χρησιμοποιώντας την συντομογραφία της ποσότητας του δεύτερου όρου εντροπίας ως,

$$E(S, A) = \sum_{i=1}^{m} f_S(A_i) \cdot E\left(S_{A_i}\right)$$

Ο τύπος του **κέρδους πληροφορίας** μπορεί να γραφεί επίσης στην πιο απλή μορφή,

$$G(S, A) = E(S) - E(S, A)$$

Συνεπώς, το Χαρακτηριστικό Μαλλιά θα επιλεχθεί στο παράδειγμα μας, καθώς παρουσιάζει το μέγιστο Κέρδος Πληροφορίας και άρα διαχωρίζει καλύτερα το Σύνολο Δεδομένων. [17] [18] [19]

2.3 Συνδυασμοί Ταξινομητών (Ensemble of Classifiers)

Έχοντας μια εξοικείωση πλέον με την αρχή λειτουργίας ενός Ταξινομητή Δένδρου Απόφασης, μπορούμε να δούμε κάποιες ενδιαφέρουσες περιπτώσεις συνδυασμού Ταξινομητών. Κάθε Μοντέλο Μηχανικής Μάθησης έχει προκύψει από Επαγωγικές διαδικασίες, συνεπώς έχει καταλήξει σε δικές του γενικεύσεις και άρα είναι επιρρεπές στο να έχει εσφαλμένη συλλογιστική. Αυτό το πρόβλημα μπορεί να περιοριστεί χρησιμοποιώντας **Συνδυασμό Ταξινομητών (Ensemble of Classifiers)**.

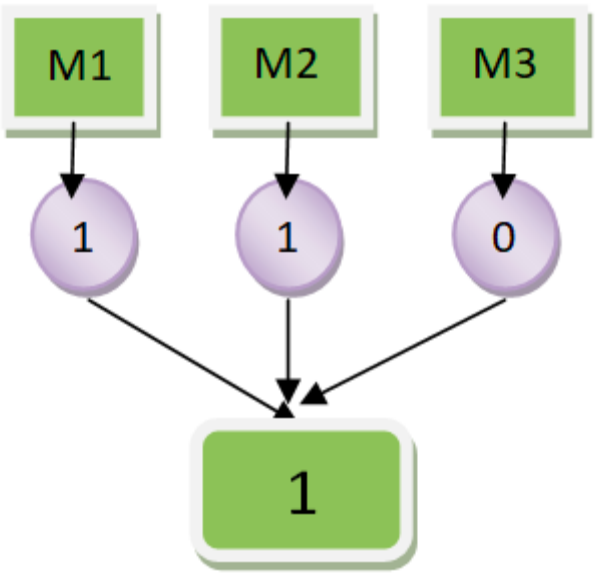
Το τι εννοούμε με τον όρο «συνδυασμό» μπορεί να ποικίλει αρκετά, καθώς μπορούμε να διαφοροποιήσουμε τους Ταξινομητές με διάφορους τρόπους. Θα μπορούσαμε για παράδειγμα να χρησιμοποιήσουμε τελείως διαφορετικούς αλγορίθμους, διαφορετικές Υπερπαραμέτρους (τιμές που «κουρδίζουν κατάλληλα ένα μοντέλο) ή/και να χρησιμοποιηθούν διαφορετικά Σύνολα Δειγμάτων Εκπαίδευσης για κάθε αλγόριθμο. Όλα αυτά θα είχαν σαν αποτέλεσμα τα μοντέλα να παράγουν διαφορετικά πρότυπα, ακόμα και εάν μερικές φορές συμφωνούν στο ίδιο τελικό αποτέλεσμα. [18]

Ένας άλλος παράγοντας που καλούμαστε να λάβουμε υπόψιν είναι το πώς να συνδυάσουμε τα μοντέλα αυτά μεταξύ τους. Οι πιο διαδεδομένοι τρόποι στην βιβλιογραφία είναι οι **Συνδυασμοί Ψηφοφορίας Ταξινομητών (Voting Ensemble Classifiers)**, τα **Πακέτα Ταξινομητών (Bag of Classifiers)** και **Ενισχυτικοί Συνδυασμοί Ταξινομητών (Boosting Classifiers)**. Στην ουσία κάθε κατηγορία προκύπτει από βελτιστοποίηση της προηγούμενής της.

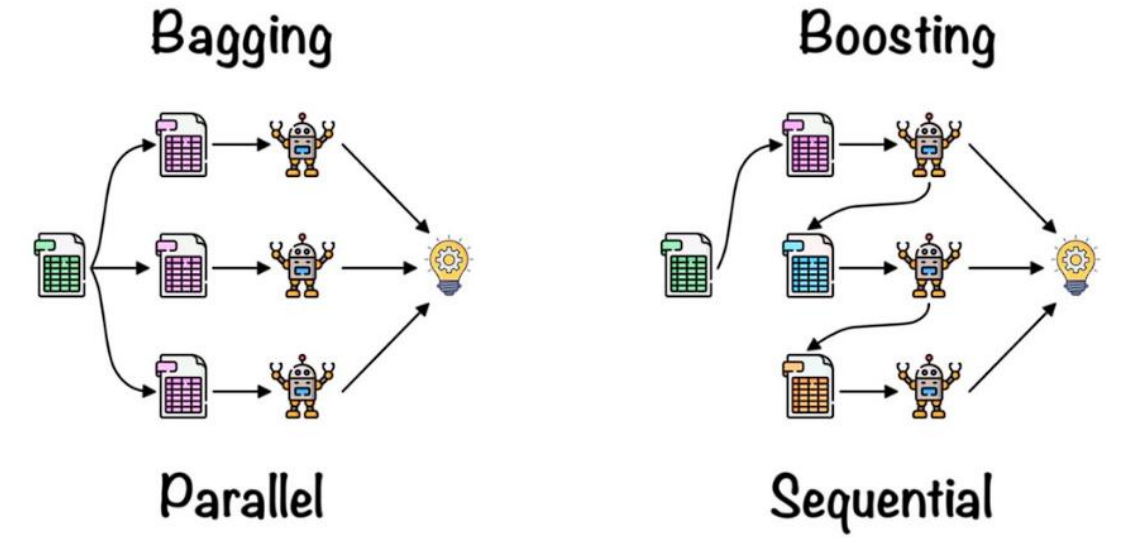
Κατά τους Συνδυασμούς Ψηφοφορίας Ταξινομητών, τα επιμέρους εκπαιδευμένα μοντέλα συνδυάζονται ως μία ομάδα και δίνουν όλα ταυτόχρονα την πρόβλεψή τους για ένα δείγμα. Η τελική πρόβλεψη της ομάδας αναδεικνύεται εκείνη την οποία έχουν επιλέξει οι περισσότεροι αλγόριθμοι. Ιδιαίτερα σε προβλήματα διττής Κατηγοριοποίησης, ο αριθμός των μοντέλων οφείλει να είναι περιττός, ώστε να αποφευχθεί το ενδεχόμενο ισοψηφίας ανάμεσα στις δύο επιλογές.

Κατά τα Πακέτα Ταξινομητών, τα μοντέλα ενός Συνδυασμού Ψηφοφορίας έχουν εκπαιδευτεί σε ελαφρά διαφορετικά Σύνολα Δεδομένων Εκπαίδευσης. Αν ένας αλγόριθμος παρουσιάσει μεγάλη διαφορά στις εκτιμήσεις που παράγει ενώ δέχτηκε μικρή διαφορά στα δεδομένα Εκπαίδευσης καλείται **Ασταθής (Unstable)**, η πιο επιστημονικά παρουσιάζει μεγάλη διασπορά εκτίμησης. Τα Δέντρα Αποφάσεων που μελετήσαμε στην προηγούμενη ενότητα τείνουν να παράγουν Ασταθή μοντέλα.

Οι Ενισχυτικοί Συνδυασμοί Ταξινομητών, με την σειρά τους, είναι Πακέτα Ταξινομητών που έχουν εκπαιδευτεί μεν σε διαφορετικά Σύνολα Δεδομένων Εκπαίδευσης, αλλά με τέτοιο τρόπο, ώστε οι αδύναμες εκτιμήσεις τους να επικαλύπτονται από τις δυνατές εκτιμήσεις των υπολοίπων. [19] [18]

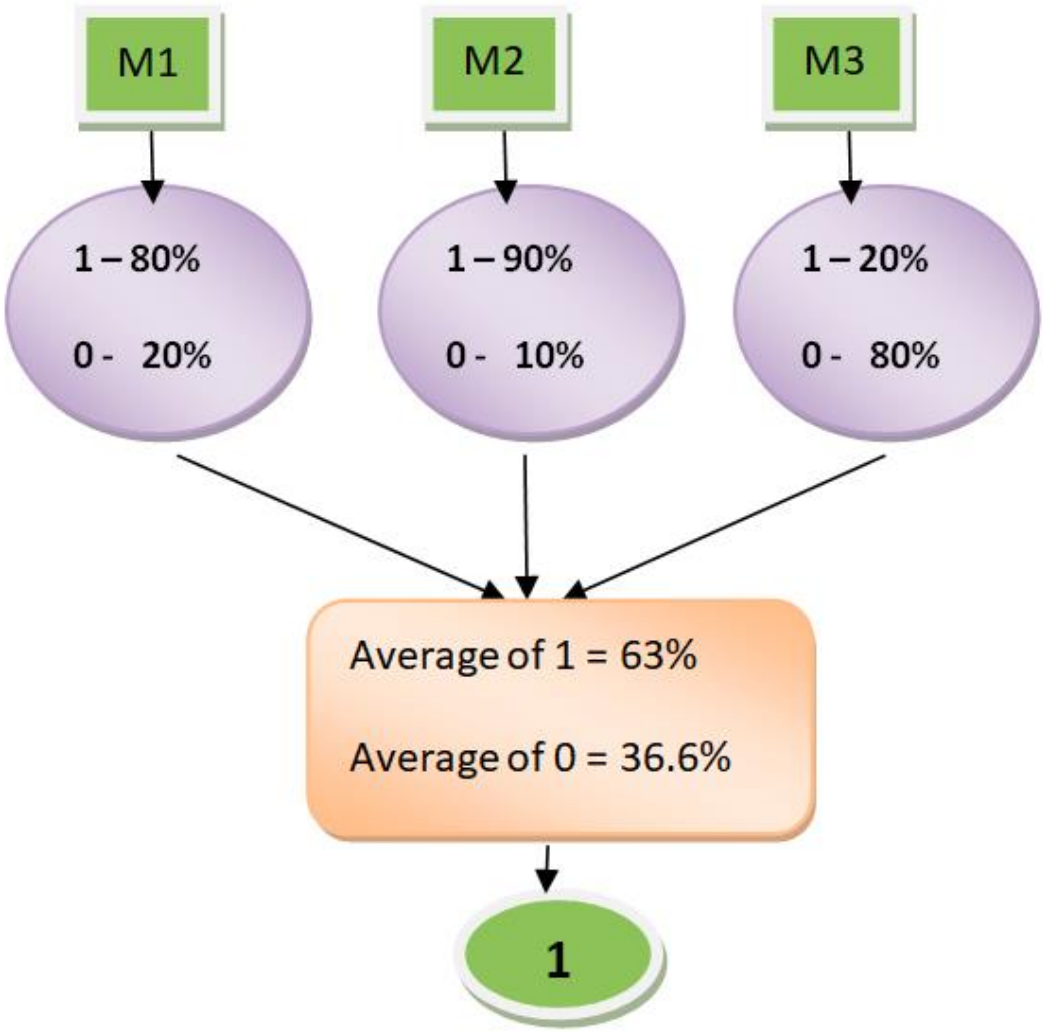


Εικόνα 8: Παράδειγμα μιας μεθόδου Συνδυασμών Ταξινομητών με Σκληρή Ψηφοφορία (Hard Voting). Τα τρία μοντέλα M1, M2 και M3 κάνουν διαφορετικές προβλέψεις για την εκτίμηση Ετικέτας του ίδιου Δείγματος. Τα μοντέλα M1 και M2 εκτίμησαν πως το Δείγμα ανήκει στην κατηγορία "1", ενώ το μοντέλο M3 κατέληξε στο αποτέλεσμα "0". Η ολική μέθοδος θα αποδώσει εν τέλη την τιμή "1" στο Δείγμα, καθώς οι ψήφοι υπερισχύουν για αυτή την κατηγορία δύο προς ένα. Πηγή: https://vitalflux.com/hard-vs-soft-voting-classifier-python-example/



Εικόνα 9: Η διαφορά προσέγγισης ανάμεσα στις μεθόδους Πακέτων (Bagging) και Ενισχυτικών (Boosting) Ταξινομητών. Οι μεν λειτουργούν παράλληλα για να καταλήξουν σε κάποια απόφαση, ενώ οι δε περνάνε σειριακά από διαδοχικές διαδικασίες εκπαίδευσης. Πηγή: https://www.analyticsvidhya.com/

Αξίζει να σημειωθεί σε αυτό το σημείο, πως παρότι οι συνδυασμοί ταξινομητών είναι αλγοριθμικά πιο σύνθετοι από μεμονωμένα μοντέλα, η πολυπλοκότητά τους δεν εγγυάται απαραίτητα καλύτερα αποτελέσματα. Αν δεν εφαρμοστούν σωστά, υπάρχει πιθανότητα ο συνδυασμός να έχει χειρότερες επιδόσεις από τα αυτοτελή. Στην διαδικασία της αξιολόγησης θα παίξουν καθοριστικό ρόλο οι μετρικές, για τις οποίες θα γίνει λόγος σε παρακάτω υποκεφάλαια.

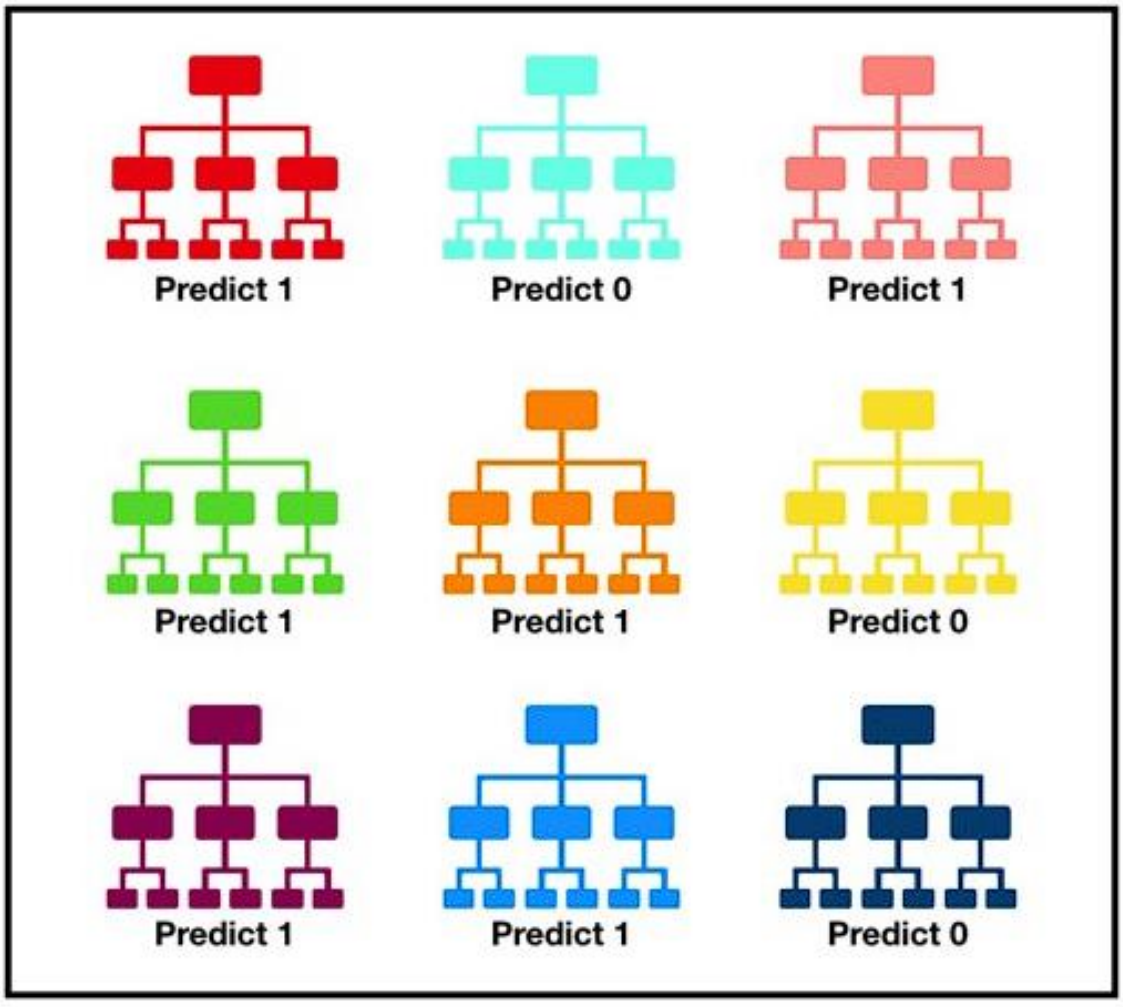


Εικόνα 10: Παράδειγμα μιας μεθόδου Συνδυασμών Ταξινομητών με Ήπια Ψηφοφορία (Soft Voting). Τα τρία μοντέλα M1, M2 και M3 κάνουν διαφορετικές προβλέψεις για την εκτίμηση της πιθανότητας η Ετικέτα του Δείγματος να ανήκει στην κατηγορία "0" ή "1". Κατά μέσο όρο, τα μοντέλα εκτιμούν πως η πιθανότητα το Δείγμα να ανήκει στην κατηγορία "1" είναι 63%, ενώ για την κατηγορία "0" είναι 36.3%. Συνεπώς, η ολική μέθοδος θα αποδώσει εν τέλη την τιμή "1" στο Δείγμα, καθώς αυτό παρουσιάζει την μέγιστη πιθανοφάνεια. Πηγή: https://vitalflux.com/hard-vs-soft-voting-classifier-python-example/

2.4 Ταξινομητής Τυχαίου Δάσους

Οι Ταξινομητές Τυχαίου Δάσους πρακτικά αποτελούνται από έναν Συνδυασμό Πακέτων Δέντρων Απόφασης Σκληρής ή **Ήπιας Ψηφοφορίας (Ensemble Bag of Decision Trees with Hard/Soft Voting)**, όρους που περιεγράφηκαν διαδοχικά στις προηγούμενες ενότητες. Κατ' αναλογία λοιπόν προκύπτει και το όνομα, υπό την έννοια πως πολλά «Δέντρα» Αποφάσεων απαρτίζουν ένα «Δάσος» Αποφάσεων. Χαρακτηρίζεται «Τυχαίο» διότι, ως Πακέτα, χρησιμοποιούν «τυχαία» διαφορετικά υποσύνολα του Συνόλου Εκπαίδευσης και διαφορετικά Χαρακτηριστικά για να καταλήξουν σε κάποια εκτίμηση. Συνεπώς, κάθε Δένδρο καταλήγει στην απόφασή του με διαφορετική προσέγγιση. [21] [22]

Με άλλα λόγια, θεωρώντας πως δεν σφάλουν όλες οι εκτιμήσεις τους στο ίδιο αποτέλεσμα, τα επιμέρους Δένδρα αλληλοπροστατεύονται από επιμέρους λάθος εκτιμήσεις. Τα επιμέρους Δένδρα ενός Ταξινομητή Τυχαίου Δάσους θεωρούνται συνεπώς ασυσχέτιστα, ως προς τον τρόπο που εξάγουν τις εκτιμήσεις τους. [21] [22]



Εικόνα 11: Ένα Τυχαίο Δάσος Αποφάσεων απαρτίζεται από πολλά Δένδρα Αποφάσεων τα οποία έχουν εκπαιδευτεί σε ξεχωριστά Χαρακτηριστικά, εξ ου και τα διαφορετικά χρώματα ανά Δένδρο. Πηγή: https://towardsdatascience.com

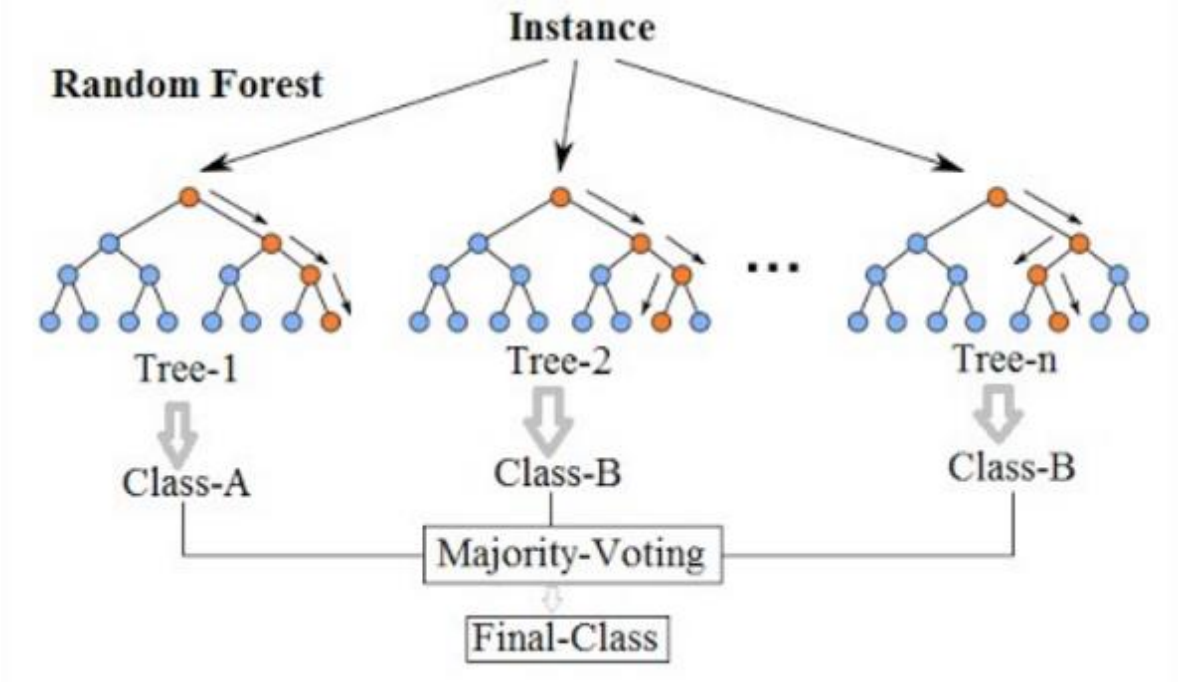
Το γεγονός πως κάθε επιμέρους δένδρο κάνει ξεχωριστούς διαχωρισμούς έχει σαν αποτέλεσμα του ότι τα Τυχαία Δάση δεν είναι επιρρεπή στον κίνδυνο της υπερπροσαρμογής, τουλάχιστον όχι τόσο όσο τα μεμονωμένα Δένδρα Αποφάσεων. Επίσης, η εφαρμογή του Τυχαίου Δάσους ενδείκνυται όταν ένας απλός ταξινομητής Δένδρου μπερδεύει μια κατηγορία με κάποια άλλη. Αυτό το γεγονός γίνεται πολύ εμφανές αν χρησιμοποιηθεί ένας Πίνακας Σύγχυσης για το πρόβλημα. Περισσότερες λεπτομέρειες για τους Πίνακες Σύγχυσης θα διευκρινιστούν στο υποκεφάλαιο που αφορά τις Μετρικές Απόδοσης.

Μερικά από τα πλεονεκτήματα των Ταξινομητών Τυχαίου Δάσους είναι πως [23]:

- Ανταποκρίνονται επαρκώς σε περιπτώσεις που κάποιες τιμές είναι κενές.
- Μπορούν να διαχειριστούν καλύτερα Σύνολα Δεδομένων όπου μία κατηγορία υπερτερεί των υπολοίπων.
- Διατρέχουν μικρότερο κίνδυνο Υπερπροσαρμογής.
- Καταφέρνουν συνήθως υψηλές επιδόσεις ακρίβειας.

Μερικά από τα μειονεκτήματα των Ταξινομητών Τυχαίου Δάσους είναι πως [23]:

- Αργούν αρκετά κατά την εκπαίδευση.
- Διατρέχουν κίνδυνο να αναπτύξουν προκαταλήψεις για κατηγορικές **Μεταβλητές (Categorical Variables)**.
- Δεν ανταποκρίνονται καλά σε πολλά το πλήθος Χαρακτηριστικά που σχηματίζουν **αραιές γεωμετρίες (a lot of Sparse Features).**



Εικόνα 12: Η αρχή λειτουργίας ενός Τυχαίου Δάσους. Τα επιμέρους Δένδρα παράγουν τις εκτιμήσεις τους, οι οποίες μέσω μίας διαδικασίας ψηφοφορίας, καταλήγουν σε ένα μια συνολιγκή τελική εκτίμηση.

2.5 Ταξινομητής Ήπιας Ενίσχυσης Βαθμίδας (Light Gradient Boosting Machine)

Οι Ταξινομητές Ήπιας Ενίσχυσης Βαθμίδας (Light Gradient Boosting Machine) είναι μια οικογένεια **Ταξινομητών Συνδυασμού Ενίσχυσης (Ensemble Boosting Classifiers)** που περιλαμβάνει γνωστούς αλγορίθμους όπως οι αλγόριθμοι: **Ενίσχυσης Βαθμίδας Δένδρου (Gradient Boosting-based Tree, GBT), Ενίσχυσης Βαθμίδας Δένδρων Αποφάσεων (Gradient Boosting Decision Trees, GBDT)** και **Μηχανές Ενίσχυσης Βαθμίδας (Gradient Boosting Machines, GBM) και RF**.

Η **Ενίσχυση Βαθμίδας (Gradient Boosting)** είναι μια τεχνική Μηχανικής Μάθησης κατά την οποία συνδυάζονται σειριακά ταξινομητές, υπό την εικασία πως κάθε επόμενο μοντέλο που θα προκύπτει, σε συνδυασμό με το προηγούμενο του, θα έχει μικρότερο στατιστικό μέσο όρο σφάλματος από το προηγούμενο. Αυτή η στοχαστική προσπάθεια μείωσης του σφάλματος καλείται **Παλινδρόμηση (Regression).** Σε αντιδιαστολή με τους προηγούμενους ταξινομητές, ο LightGBM δεν κατασκευάζει τα δένδρα του ανά επίπεδο αλλά το κάνει «κατά φύλλο», δηλαδή συνεχίζει τους διαχωρισμούς μέχρι να φτάσει σε κάποιο φύλλο του δένδρου και μετά επιστρέφει στο προηγούμενο επίπεδο για να επαναλάβει την διαδικασία. [24] [25]

Ένας ταξινομητής lightGBM είναι πιο αποδοτικός στις εκτιμήσεις του καθώς επιλέγει τον διαχωρισμό δεδομένων στα δένδρα του που έχουν την μικρότερη βαθμίδα και επιτυγχάνουν το μικρότερο δυνατό σφάλμα μέσω της **Βαθμιδοειδής Μονόπλευρης Δειγματοληψίας (Gradient-based One-Sided Sampling, GOSS)**. Μια άλλη τεχνική που χαρακτηρίζει τον ταξινομητή Ήπιας Ενίσχυσης Βαθμίδας είναι η **Αποκλειστική Συσσώρευση Χαρακτηριστικών (Exclusive Feature**

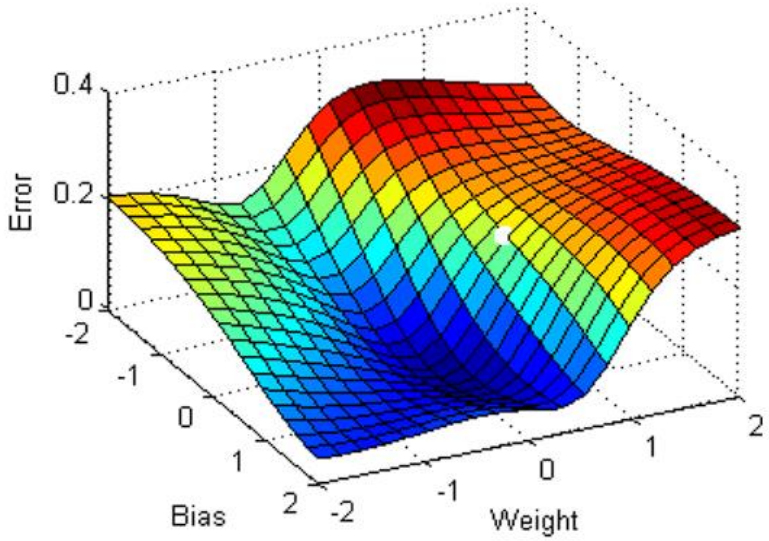
Bundling, EFB), η οποία ομαδοποιεί χαρακτηριστικά που παίρνουν διακριτές τιμές -όπως Χαρακτηριστικά που προκύπτουν από One-Hot Enconding- με σκοπό την μείωση της διαστατικότητας και τον περιορισμό των **Αραιά Κατανεμημένων Δεδομένων (Sparse Data).** [25]

Τα πλεονεκτήματα των Ταξινομητών LightGBM είναι πως [26]:

- Εκπαιδεύονται πολύ γρηγορότερα σε σχέση με τους Ταξινομητές Τυχαίου Δάσους.
- Δαπανούν πολύ λιγότερη μνήμη RAM.
- Ενδείκνυνται για μεγάλου όγκου Dataset.

Τα μειονεκτήματα των Ταξινομητών LightGBM είναι πως [26]:

- Είναι επιρρεπή σε Υπερπροσαρμογή καθώς δημιουργούν πιο περίπλοκα και βαθιά δένδρα.
- Είναι επιρρεπή σε Υπερπροσαρμογή σε μικρού μεγέθους Dataset



Εικόνα 13: Η γραφική παράσταση του σφάλματος συναρτήσει των Προκαταλήψεων (Bias) και των βαρών (Weights) ενός αλγορίθμου Παλινδρόμησης (Regression). Στόχος είναι να βρεθούν οι κατάλληλες τιμές των παραμέτρων και υπερπαραμέτρων ώστε, σε κάθε βήμα εκπαίδευσης, το σφάλμα να τείνει σε κάποιο ελάχιστο, δηλαδή εν προκειμένω, να πλησιάσει την πιο μπλε περιοχή στο γράφημα. Πηγή: https://medium.com/@hakobavjyan/stochastic-gradient-descent-sgd-10ce70fea389

2.6 Μετρικές Απόδοσης Μοντέλων Μηχανικής Μάθησης Δυαδικής Κατηγοριοποίησης

Για να μπορέσουμε να ποσοτικοποιήσουμε τις επιδόσεις των μοντέλων που περιεγράφηκαν πρωτύτερα, θα πρέπει να ορίσουμε κάποιες μετρικές. Θέλουμε να είναι αντικειμενική και αντιπροσωπευτική η αξιολόγηση των επιδόσεων, συνεπώς θα πρέπει να γίνει σε ένα στατιστικά σημαντικό σύνολο δειγμάτων, ανεξάρτητο όμως από το σύνολο δειγμάτων που χρησιμοποίησε ο αλγόριθμος για να παραγάγει το μοντέλο, αυτό που ορίσαμε σε προηγούμενη υποενότητα ως **Σύνολο Δεδομένων Ελέγχου (Test Dataset)**. Λαμβάνοντας υπόψιν πως το πρόβλημα που μας απασχολεί έχει δύο τιμές κατηγοριοποίησης, θα περιοριστούμε στις **Μετρικές Δυαδικής Κατηγοριοποίησης (Binary Classification Metrics)**. Θεωρώντας πως πληρούνται οι παραπάνω προϋποθέσεις, μπορούμε να περάσουμε στην περιγραφή των μετρικών Απόδοσης των Μοντέλων Μηχανικής Μάθησης.

2.6.1 Στατιστικά Στοιχεία Κατηγοριοποίησης και Πίνακες Σύγχυσης

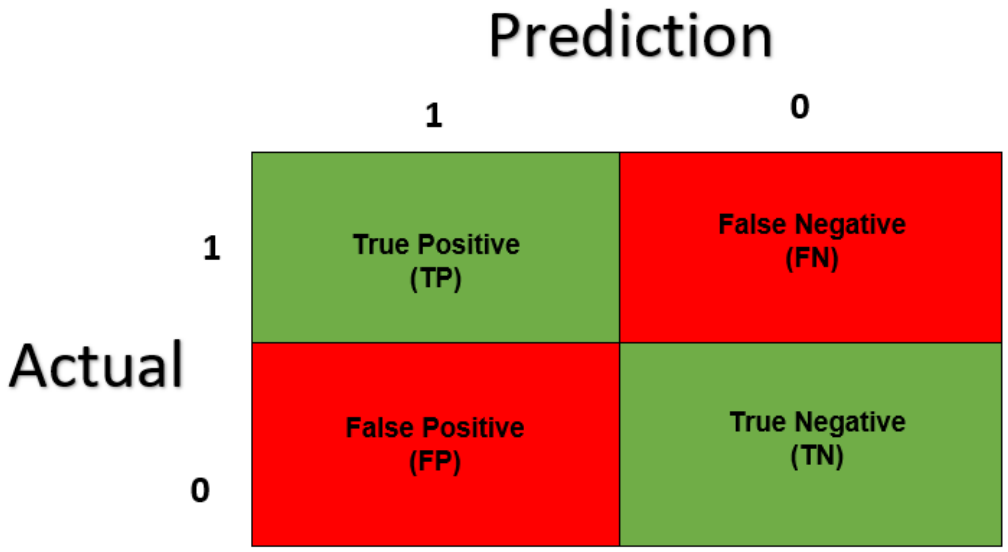
Έχοντας δώσει σαν είσοδο στο μοντέλο το Test Dataset, το μοντέλο κάνει κάποιες εκτιμήσεις. Συγκεντρώνοντας τα αποτελέσματα που εξήγαγε το μοντέλο, παίρνουμε τα παρακάτω στατιστικά στοιχεία, τα οποία ορίζουμε ως:

- **Ορθά Θετικά (True Positive, TP)**: το σύνολο των Δειγμάτων που ορθά ταξινομήθηκε από το μοντέλο στην Α κατηγορία
- **Ορθά Αρνητικά (True Negative, TN)**: το σύνολο των Δειγμάτων που ορθά ταξινομήθηκε από το μοντέλο στην Β κατηγορία
- **Εσφαλμένα Θετικά (False Positive, FP)**: το σύνολο των Δειγμάτων που ταξινομήθηκε εσφαλμένα από το μοντέλο στην Α κατηγορία
- **Εσφαλμένα Αρνητικά (False Negative, FN)**: το σύνολο των Δειγμάτων που ταξινομήθηκε εσφαλμένα από το μοντέλο στην Β κατηγορία

Το άθροισμα όλων αυτών τον στοιχείων ισοδυναμεί με το πλήθος των δειγμάτων του Test Dataset. Σε μορφή εξίσωσης έχουμε:

$$Total_{Test} = TP + TN + FP + FN$$

Με αυτά τα στοιχεία μπορεί να κατασκευασθεί ο Πίνακας Σύγχυσης που εμπεριέχει όλα αυτά τα στοιχεία. Δεν είναι τίποτα άλλο παρά μια χρήσιμη συνοπτική απεικόνιση των στατιστικών που περιγράψαμε παραπάνω. Συνήθως έχει τις **Πραγματικές Τιμές (Actual Truths, Ground Truths)** ως γραμμές και τις **Προβλέψεις (Predictions)** ως στήλες. Για παράδειγμα, με βάση την παρακάτω εικόνα, τα δείγματα που είχαν πράγματι την τιμή «1» βρίσκονται στην πρώτη γραμμή του πίνακα. Οι Τιμές που πρόβλεψε το μοντέλο ως «1» είναι στην πρώτη στήλη. Συνεπώς, τα δείγματα που μάντεψε ορθά το μοντέλο ως «1», δηλαδή το στοιχείο που ονομάσαμε True Positive, απεικονίζονται στο πράσινο κιτίο στην πρώτη γραμμή και πρώτη στήλη του παρακάτω Πίνακα Σύγχυσης [27].



Εικόνα 14: Η μορφή ενός Πίνακα Σύγχυσης Δυαδικής Κατηγοριοποίησης. Οι πραγματικές (Actual) κατηγορίες είναι οι σειρές "1" και "0" στα αριστερά, ενώ οι εκτιμήσεις/προβλέψεις (Predictions) του μοντέλου είναι οι στήλες "1" και "0" αντίστοιχα.

2.6.2 Η Μετρική Ευστοχίας

Η μετρική **Ευστοχίας (Accuracy)** είναι ο λόγος των ορθών προβλέψεων ως προς όλα τα δείγματα του Test Set. Μαθηματικά ορίζεται ως:

$$Accuracy = \frac{TP + TN}{Total_{Test}}$$

Παρότι είναι αρκετά χρήσιμη, μπορεί να μην αποτελεί πάντα καλή επιλογή σαν μετρική, καθώς βγάζει καλά αποτελέσματα για συμμετρικά Σύνολα Δεδομένων, όταν δηλαδή τα FN και FP είναι παραπλήσια. Για παράδειγμα, αν διαλέξουμε ένα Σύνολο Δειγμάτων Ελέγχου όπου τα TP είναι πολύ περισσότερα από τα TN, τότε αν βγάλει πολύ καλό ποσοστό επίδοσης, μπορεί βεβιασμένα να θεωρήσουμε ότι το μοντέλο μας κάνει γενικά σωστές προβλέψεις. Αυτό όμως δεν είναι απαραίτητα σωστό πόρισμα. Το μοντέλο μας μπορεί να είναι απλά πολύ καλό στο να μαντεύει τα TP. Αν διαλέγαμε άλλο Test Set με περισσότερα TN από TP, μπορεί το μοντέλο μας να είχε χειρότερη επίδοση ως προς την ίδια μετρική [27].

2.6.3 Η Μετρική Ακρίβειας

Η μετρική **Ακρίβειας (Precision)** είναι το ποσοστό των ορθών προβλέψεων του μοντέλου που αφορούν την κατηγορία Α ως προς όλες τις προβλέψεις του μοντέλου που εκτίμησε ως κατηγορία Α. Συντακτικά,

$$Precision = \frac{TP}{TP + FP}$$

Η συγκεκριμένη μετρική είναι πολύ χρήσιμη όταν είναι σημαντικό να κάνουμε σωστές προβλέψεις μόνο για την κατηγορία Α.

2.6.4 Η Μετρική Ανάκλησης

Η μετρική **Ανάκλησης** ή **Ευαισθησίας (Recall, Sensitivity)** είναι το ποσοστό των ορθών προβλέψεων του μοντέλου που αφορούν την κατηγορία Α ως προς όλα τα δείγματα που ανήκουν στην κατηγορία Α.

$$Recall = \frac{TP}{TP + FN}$$

Η μετρική αυτή είναι χρήσιμη όταν θέλουμε να αποφύγουμε την κατηγοριοποίηση στην κατηγορία Β. Για παράδειγμα, για διάγνωση σε μια μεταδοτική ασθένεια, θα θέλαμε ούτως η άλλως υψηλή ακρίβεια, αλλά θα είναι πιο σημαντικό να μην κατηγοριοποιήσουμε έναν ασθενή ως υγιή, γιατί τότε υπάρχει ο κίνδυνος εξάπλωσης της ασθένειας [27].

2.6.5 Η Μετρική Ειδικότητας

Η μετρική **Ειδικότητας (Specificity)** είναι το ποσοστό των ορθών προβλέψεων του μοντέλου που αφορούν την κατηγορία Β ως προς όλες τις προβλέψεις του μοντέλου που εκτίμησε ως κατηγορία Α.

$$Specificity = \frac{TN}{TP + FP}$$

Αντίστοιχα με την προηγούμενη μετρική, η μετρική Ειδικότητας είναι χρήσιμη όταν θέλουμε να αποφύγουμε την κατηγοριοποίηση στην κατηγορία Α [28].

2.6.6 Η Μετρική F1

Η μετρική **F1** είναι ο αρμονικός μέσος μεταξύ των μετρικών precision και recall και ορίζεται, ως:

$$F1 = \frac{2 \cdot Recall \cdot Precision}{Recall + Precision} = \frac{2 \cdot TP}{2 \cdot TP + FP + FN}$$

Αν μας ενδιαφέρουν και οι FP και οι FN τιμές, οι οποίες είναι μεταξύ τους άνισες, τότε αυτή η μετρική είναι κατάλληλη [28].

2.6.7 Η Μετρική Καμπύλης Δέκτη Λειτουργικού Χαρακτηριστικού

Η μετρική **Καμπύλης Δέκτη Λειτουργικού Χαρακτηριστικού (Receiver Operating Characteristic, ROC)** αφορά μία οπτικοποίηση λαμβάνοντας υπόψιν όλους τους πιθανούς πίνακες σύγχυσης. Η απεικόνιση αυτή παίρνει την μορφή:

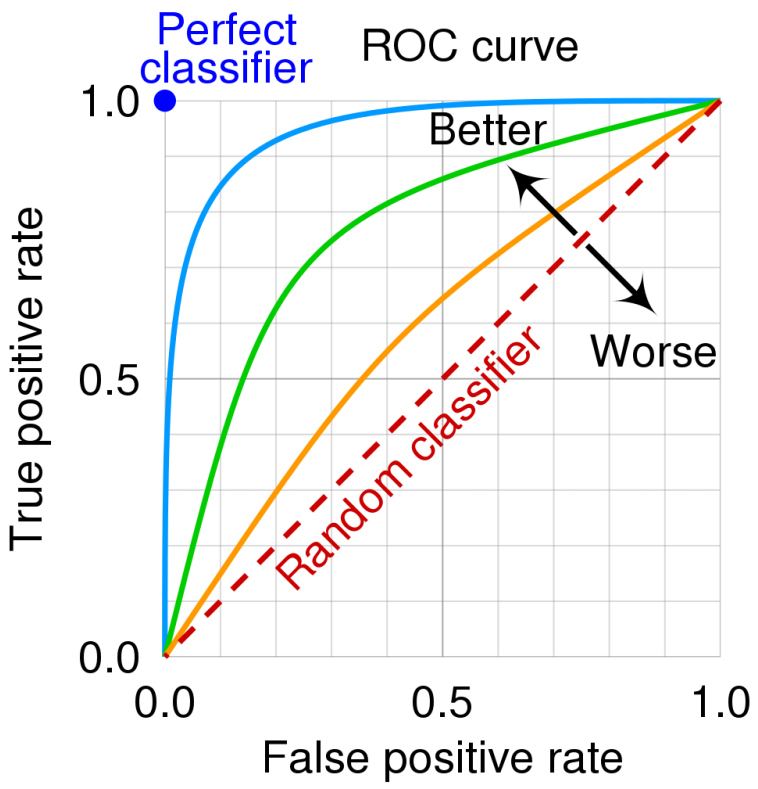
$$y = G(x), \qquad G: x \to y$$

Όπου,

$$x = 1 - Sensitivity$$

$$y = Recall$$

Κατά αυτόν τον τρόπο παράγονται γραφήματα που έχουν την παρακάτω μορφή:



Εικόνα 15: Το διάγραμμα ROC. Με θερμά χρώματα απεικονίζονται οι χειρότεροι ταξινομητές, ενώ με ψυχρά χρώματα οι βέλτιστοι. Ένας θεωρητικά τέλειος ταξινομητής θα βρισκόταν στην πάνω αριστερή γωνία του διαγράμματος.

2.6.8 Η Μετρική Στοχαστικής Βαθμονόμησης

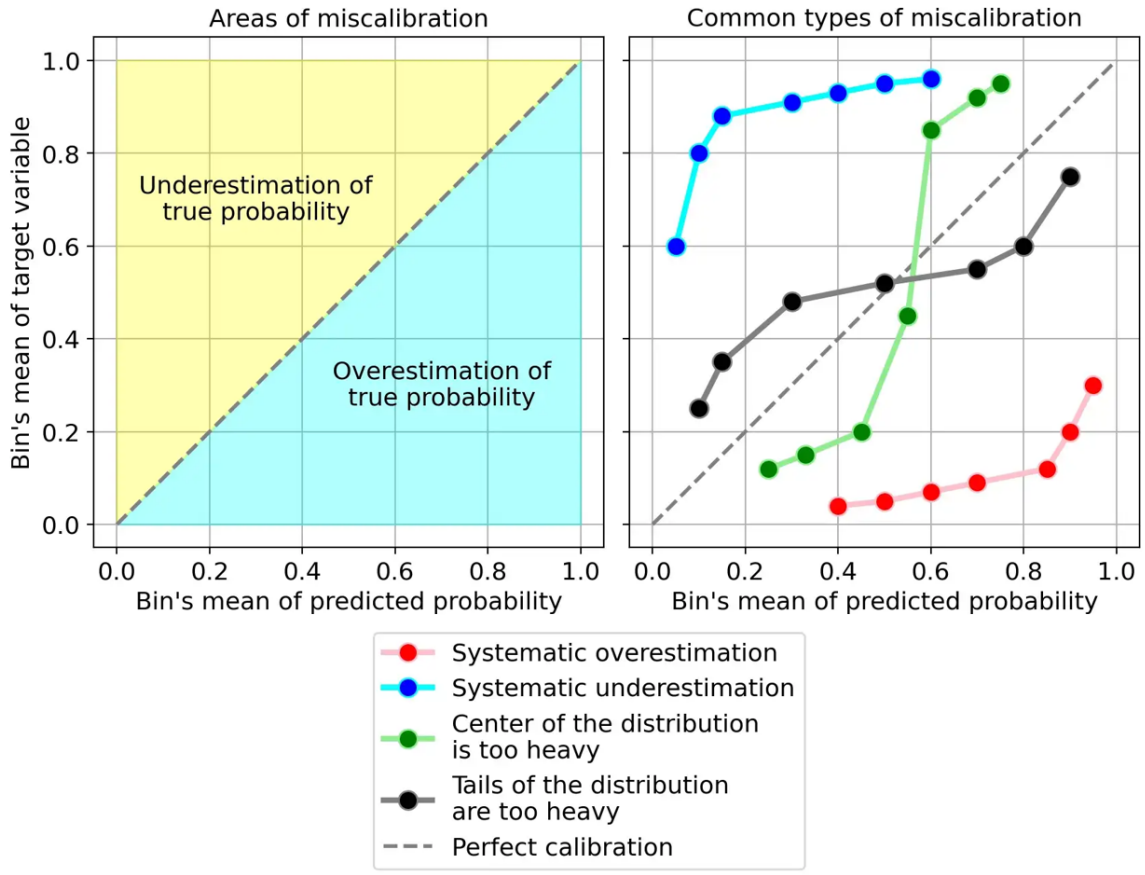
Η μετρική **Στοχαστικής Βαθμονόμησης (Probability Calibration, Prediction Probability, predict_proba)** είναι μια **Μπεσιανή (Bayesian)** εκτίμηση μιας πρόβλεψης ως προς μία κατηγορία. Ο υπολογισμός της πιθανότητας αυτής προκύπτει ευθέως από το θεώρημα του Bayes:

$$p(y_i = L_j) \cdot p(y_i = L_j) = p(f_i) \cdot p(f_i)$$

Σε μαθηματικό φορμαλισμό υπολογίζεται ως:

$$Prediction\ Probability = p(f_i)$$

Στην πράξη ερμηνεύεται ως «η πιθανότητα με την οποία η i-οστή κατηγοριοποίηση (y_i) ανήκει στην κατηγορία (L) λαμβάνοντας υπόψιν τα i-οστά δεδομένα (f_i)», δηλαδή αποτελεί ένα ποιοτικό κριτήριο κατηγοριοποίησης. Θα μπορούσε να θεωρηθεί ως η «Σιγουριά» με την οποία γίνεται μια πρόβλεψη. Στην βιβλιοθήκη Sci-Kit Learn της Python αναφέρεται ως **Πιθανότητα Πρόβλεψης (Prediction Probability)** και υπολογίζεται χρησιμοποιώντας την μέθοδο predict_proba().



Εικόνα 16: Το Γράφημα Στοχαστικής Βαθμονόμησης. Ο άξονας x περιέχει την μέση εκτιμώμενη πιθανότητα, ενώ ο άξονας y την αντίστοιχη διασπορά. Στο πρώτο διάγραμμα (αριστερά) φαίνονται οι περιοχές υπερεκτίμησης (γαλάζια περιοχή) και υποεκτίμησης (κίτρινη περιοχή) σε σχέση με την πραγματική πιθανότητα. Το ιδανικό μοντέλο βρίσκεται επί τις κύριας διαγώνιου. Στο δεύτερο διάγραμμα (δεξιά) απεικονίζονται συνήθεις σφάλματα στα οποία υπόκεινται οι εκτιμητές, όπως Συστηματική Υπερεκτίμηση (κόκκινη γραμμή), Συστηματική Υποεκτίμηση (γαλάζια γραμμή), Υπερβολικά Συμπαγές Κέντρο Κατανομής (πράσινη γραμμή) και Υπερβολικά Συμπαγή Άκρα κατανομής (μαύρη γραμμή).

Κεφάλαιο 3: Πειραματική Μεθοδολογία

3.1 Περιγραφή Πειραματικής Διαδικασίας

Οι στόχοι του πειράματος σε αυτό το κεφάλαιο εστιάζουν στην δημιουργία εκτιμητών Μηχανικής Μάθησης για την έκβαση Αγώνων Τένις. Η βασική ιδέα είναι να εκπαιδευτούν τρία υποψήφια μοντέλα, συγκεκριμένα μοντέλα Δένδρου Αποφάσεων, Τυχαίου Δάσους και LightGBM και να συγκριθούν οι μεταξύ τους επιδόσεις, αλλά και με έναν «Αφελή» εκτιμητή, που θα προβλέπει το φαβορί αυστηρά και μόνο με την απόδοση που δίνουν οι Bookmakers. Απώτερος στόχος είναι η επιλογή του καλύτερου μοντέλου και η εφαρμογή του στις στρατηγικές στοιχήματος των επόμενων κεφαλαίων, ιδανικά με την διασφάλιση κάποιου οικονομικού κέρδους. Σε αυτό το κεφάλαιο επίσης, θα ορισθεί το σύνολο δεδομένων στο οποίο θα εκπαιδευτούν τα μοντέλα, τα Train και Test Sets καθώς και τα στάδια προεπεξεργασίας που εφαρμόστηκαν, συγκεκριμένα της σύνθεσης χαρακτηριστικών αλλά και της ψηφιοποίησης.

3.2 Περιγραφή Δεδομένων

3.2.1 Σύνολο Δεδομένων ATP

Όπως αναφέρθηκε προηγουμένως, για να λειτουργήσει ένας αλγόριθμος Μηχανικής Μάθησης, πέρα από τον ίδιο τον αλγόριθμο, είναι απαραίτητο ένα σύνολο δειγμάτων, το οποίο θα χρησιμοποιηθεί ως είσοδος στον αλγόριθμο τόσο για την εκπαίδευση όσο και για την αξιολόγηση του. Για την συγκεκριμένη εργασία επιλέχθηκαν δεδομένα από το «ATP and WTA Tennis Results and Betting Odds Data» που εδράζεται στον διαδικτυακό ιστότοπο Kaggle. Το συγκεκριμένο Σύνολο Δεδομένων απαρτίζεται από δύο υποσύνολα, το **ATP (Association of Tennis Professionals)** και το **WTA (Women's tennis Association)** που περιέχουν δεδομένα αγώνων τένις από το 2000 μέχρι το 2019. Συγκεκριμένα, το ATP αφορά ανδρικά πρωταθλήματα ενώ το WTA αφορά γυναικεία πρωταθλήματα. Για να αποφευχθούν προβλήματα προκατάληψής από ανισορροπία δεδομένων, για την συγκεκριμένη εργασία, επιλέχθηκε με τυχαίο τρόπο το ATP Dataset. [29]

Συνολικά είναι μια λίστα 54904 αγώνων με 55 πεδία δεδομένων. Στην ουσία το Dataset πρόκειται για μια λίστα αγώνων ανάμεσα σε δύο Αθλητές, μαζί με τα στοιχεία και αποτελέσματα του αγώνα και τις στοιχηματικές αποδόσεις των Bookmakers. Κάθε γραμμή αποτελεί διαφορετικό αγώνα και κάθε στήλη περιέχει τα αντίστοιχα στοιχεία. Η λίστα των στοιχείων που συγκεντρώθηκαν σε αυτό το Dataset για κάθε αγώνα είναι η εξής:

- Τοποθεσία (Location): Η τοποθεσία που έλαβε χώρα ο αγώνας
- Διοργάνωση (Tournament): Το όνομα του Πρωταθλήματος του οποίου είναι μέρος ο αγώνας
- Ημερομηνία (Date): Η ημερομηνία που έλαβε χώρα ο αγώνας
- Τύπος (Series): Ο τύπος του πρωταθλήματος (Grand Slam, Masters, International ή International Gold)
- Τύπος Γηπέδου (Court): Αν ο αγώνας έλαβε χώρα σε κλειστό ή ανοιχτό χώρο

- Επιφάνεια (Surface): Ο τύπος της επιφάνειας. Χωμάτινος (Clay), Σκληρός (Hard), Χλοοτάπητας (Carpet) ή γκαζόν (grass).
- Γύρος (Round): Ο γύρος του Αγώνα
- Μέγιστα Σετ (Best of): Ο μέγιστος αριθμός σετ που μπορούν να παιχτούν στο ματς
- Νικητής (Winner): Το όνομα του Αθλητή που νίκησε τον αγώνα
- Ηττημένος (Looser): Το όνομα του Αθλητή που έχασε τον αγώνα
- Κατάταξη Νικητή (WRank): Η κατάταξη του Νικητή Αθλητή στην διοργάνωση κατά την έναρξη του αγώνα
- Κατάταξη Ηττημένου (LRank): Η κατάταξη του Ηττημένου Αθλητή στην διοργάνωση κατά την έναρξη αρχή του αγώνα
- Πόντοι Νικητή (WPts): Οι πόντοι που έχει συγκεντρώσει ο Νικητής Αθλητής κατά την έναρξη του αγώνα
- Πόντοι Ηττημένου (LPts): Οι πόντοι που έχει συγκεντρώσει ο Ηττημένος Αθλητής κατά την έναρξη του αγώνα
- Κερδισμένα Παιχνίδια Νικητή 1 (W1): Το πλήθος των παιχνιδιών που κερδήθηκαν από τον Νικητή Αθλητή κατά το 1ο σετ
- Κερδισμένα Παιχνίδια Νικητή 2 (W2): Το πλήθος των παιχνιδιών που κερδήθηκαν από τον Νικητή Αθλητή κατά το 2ο σετ
- Κερδισμένα Παιχνίδια Νικητή 3 (W3): Το πλήθος των παιχνιδιών που κερδήθηκαν από τον Νικητή Αθλητή κατά το 3ο σετ
- Κερδισμένα Παιχνίδια Νικητή 4 (W4): Το πλήθος των παιχνιδιών που κερδήθηκαν από τον Νικητή Αθλητή κατά το 4ο σετ
- Κερδισμένα Παιχνίδια Νικητή 5 (W5): Το πλήθος των παιχνιδιών που κερδήθηκαν από τον Νικητή Αθλητή κατά το 5ο σετ
- Κερδισμένα Παιχνίδια Ηττημένου 1 (L1): Το πλήθος των παιχνιδιών που κερδήθηκαν από τον Ηττημένο Αθλητή κατά το 1ο σετ
- Κερδισμένα Παιχνίδια Ηττημένου 2 (L2): Το πλήθος των παιχνιδιών που κερδήθηκαν από τον Ηττημένο Αθλητή κατά το 2ο σετ
- Κερδισμένα Παιχνίδια Ηττημένου 3 (L3): Το πλήθος των παιχνιδιών που κερδήθηκαν από τον Ηττημένο Αθλητή κατά το 3ο σετ
- Κερδισμένα Παιχνίδια Ηττημένου 4 (L4): Το πλήθος των παιχνιδιών που κερδήθηκαν από τον Ηττημένο Αθλητή κατά το 4ο σετ
- Κερδισμένα Παιχνίδια Ηττημένου 5 (L5): Το πλήθος των παιχνιδιών που κερδήθηκαν από τον Ηττημένο Αθλητή κατά το 5ο σετ
- Κερδισμένα Σετ Νικητή (Wsets): Το πλήθος των σετ που κέρδισε συνολικά ο Νικητής Αθλητής
- Κερδισμένα Σετ Ηττημένου (Lsets): Το πλήθος των σετ που κέρδισε συνολικά ο Ηττημένος Αθλητής
- Σχόλιο (Comment): Χαρακτηρισμός που αφορά την διεξαγωγή του αγώνα. Κανονικά Εκτελεσμένος (Completed), Νίκη εκ Παραίτησης (Win via Retirement) ή Νίκη εκ Μη Συμμετοχής (Win via Walkover)
- B365W: Απόδοση για τον Νικητή Αθλητή από τον Bookmaker Bet365

- B365L: Απόδοση για τον Ηττημένο Αθλητή από τον Bookmaker Bet365
- B&WW: Απόδοση για τον Νικητή Αθλητή από τον Bookmaker Bet&Win
- B&WL: Απόδοση για τον Ηττημένο Αθλητή από τον Bookmaker Bet&Win
- CBW: Απόδοση για τον Νικητή Αθλητή από τον Bookmaker CentreBet
- CBL: Απόδοση για τον Ηττημένο Αθλητή από τον Bookmaker CentreBet
- EXW: Απόδοση για τον Νικητή Αθλητή από τον Bookmaker Expekt
- EXL: Απόδοση για τον Ηττημένο Αθλητή από τον Bookmaker Expekt
- LBW: Απόδοση για τον Νικητή Αθλητή από τον Bookmaker Ladbrokes
- LBL: Απόδοση για τον Ηττημένο Αθλητή από τον Bookmaker Ladbrokes
- GBW: Απόδοση για τον Νικητή Αθλητή από τον Bookmaker GameBookers
- GBL: Απόδοση για τον Ηττημένο Αθλητή από τον Bookmaker GameBookers
- IWW: Απόδοση για τον Νικητή Αθλητή από τον Bookmaker Interwetten
- IWL: Απόδοση για τον Ηττημένο Αθλητή από τον Bookmaker Interwetten
- PSW: Απόδοση για τον Νικητή Αθλητή από τον Bookmaker Pinnacles Sports
- PSL: Απόδοση για τον Ηττημένο Αθλητή από τον Bookmaker Pinnacles Sports
- SBW: Απόδοση για τον Νικητή Αθλητή από τον Bookmaker Sportingbet
- SBL: Απόδοση για τον Ηττημένο Αθλητή από τον Bookmaker Sportingbet
- SJW: Απόδοση για τον Νικητή Αθλητή από τον Bookmaker StanJames
- SJL: Απόδοση για τον Ηττημένο Αθλητή από τον Bookmaker StanJames
- UBW: Απόδοση για τον Νικητή Αθλητή από τον Bookmaker UniBet
- UBL: Απόδοση για τον Ηττημένο Αθλητή από τον Bookmaker UniBet
- Μέγιστη Απόδοση Νικητή (MaxW): Η μέγιστη απόδοση για τον Νικητή Αθλητή, όπως εμφανίζεται στο site Oddsportal.com
- Μέγιστη Απόδοση Ηττημένου (MaxL): Η μέγιστη απόδοση για τον Ηττημένο Αθλητή, όπως εμφανίζεται στο site Oddsportal.com
- Μέσος Όρος Αποδόσεων Νικητή (AvgW): Ο στατιστικός μέσος αποδόσεων όρος για τον Νικητή Αθλητή, όπως εμφανίζονται στο site Oddsportal.com
- Μέσος Όρος Αποδόσεων Ηττημένου (AvgL): Ο στατιστικός μέσος αποδόσεων όρος για τον Ηττημένο Αθλητή, όπως εμφανίζονται στο site Oddsportal.com

Εφ' όσων επιλέχθηκε το Dataset, το επόμενο στάδιο είναι η επιλογή Χαρακτηριστικών και Ετικετών, δηλαδή, όπως εξηγήσαμε στα προηγούμενα κεφάλαια, ποια δεδομένα θα χρησιμοποιούν τα μοντέλα για πρόβλεψη και, δεδομένου του ότι ασχολούμαστε με Επιβλεπόμενη Μηχανική Μάθηση, ποια δεδομένα θα προσπαθήσουν τα μοντέλα να εκτιμήσουν. Αποφασίστηκε πως κάθε στήλη πλην του νικητή, να χρησιμοποιηθεί ως Χαρακτηριστικό ενώ η στήλη του **Νικητή Αθλητή (WinPL)** να χρησιμοποιηθεί ως **Ετικέτα Κατηγοριοποίησης (Label).** Τα ονόματα εκτιμήθηκε πως περισσότερο θα προκαταβάλουν παρά θα βοηθήσουν τα μοντέλα, αφού δεν προσπαθούμε να κάνουμε προβλέψεις για μεμονωμένα άτομα. Συνεπώς, έχουν εξαιρεθεί από το σύνολο των δεδομένων που λαμβάνουν υπόψιν τους τα μοντέλα.

Επίσης, αποφασίστηκε πως η πληθώρα των δεδομένων θα χρησιμοποιηθεί ως **Σύνολο Εκπαίδευσης (Train Set),** με ότι περισσεύει να χρησιμοποιηθεί ως **Σύνολο Αξιολόγησης (Test Set).** Κοινώς, αποφασίστηκε πως **οι αγώνες από το 2003 έως το 2018** να χρησιμοποιηθούν ως

Train Set, ενώ όλοι **οι αγώνες του 2019** να χρησιμοποιηθούν ως **Test Set**. Δεν χρησιμοποιήθηκαν δεδομένα αγώνων από το 2000-2002, αφού θεωρήθηκαν ελλιπές.

Τα παραπάνω δεδομένα, δομημένα σε ένα αρχείο CSV, κατέβηκαν από τον ιστότοπο του Kaggle και, για την εκπόνηση της εργασίας, επεξεργάστηκαν μέσω της πλατφόρμας Google Collaboratory σε περιβάλλον με τις εξής προδιαγραφές:

- Επεξεργαστής (CPU): Intel Xeon Dual-Core @ 2.20 GHz
- Κάρτα Γραφικών (GPU): NVIDIA Tesla T4, 2560 cores, 16 GB GDDR6 VRAM, 256 bit Bus
- Γλώσσα Προγραμματισμού: Python v3.7.15
- Βιβλιοθήκη Scikit-Learn v3.1

3.2.2 Δημιουργία Σύνθετων Μεταβλητών

Παρότι το Dataset ήδη περιέχει μια σημαντική πληθώρα δεδομένων για κάθε αγώνα, έγινε από νωρίς αντιληπτό, κατά την έκβαση του πειράματος, πως κάποιες σημαντικές μετρικές απουσίαζαν για την εκτίμηση του αποτελέσματος ενός αγώνα. Συνεπώς, κρίθηκε σκόπιμο, χρησιμοποιώντας τα υπάρχοντα δεδομένα να προκύψουν στοιχεία που θα βοηθούσαν κάποιον στοιχηματικό Παίκτη να εποπτεύσει και να εκτιμήσει καλύτερα το αποτέλεσμα του κάθε αγώνα.

Συγκεκριμένα, υλοποιήθηκε μια συνάρτηση η οποία υπολογίζει το ποσοστό νίκης κάθε αθλητή για κάθε αγώνα για προαποφασισμένα χρονικά διαστήματα. Τα διαστήματα που επιλέχθηκαν ήταν:

- Οι τελευταίοι 10 αγώνες (Ratio of Last 10 Matches)
- Οι αγώνες το τελευταίο τρίμηνο (Ratio of Last 3 Months)
- Οι αγώνες το τελευταίο εξάμηνο (Ratio of Last 6 Months)
- Οι αγώνες τον τελευταίο χρόνο (Ratio of Last 1 Year)
- Οι αγώνες τα τελευταία τρία χρόνια (Ratio of Last 3 Years)

Τα δεδομένα αυτά συγχωνεύθηκαν με το αρχικό dataset, ώστε να υπάρχει και μια καταγραφή των παραπάνω στοιχείων και για τους δυο Αθλητές για κάθε αγώνα. Επίσης, προς αποφυγήν κάποιας λανθασμένης γενίκευσης ως προς την εμφάνιση των Αθλητών, αλλάχθηκε η μορφή του Dataset από Νικητή – Ηττημένο σε Αθλητής 1 και Αθλητής 2 με αλφαβητική σειρά. Με αυτό τον τρόπο, το προς εκπαίδευση μοντέλο δεν θα μπορεί να μαντέψει εκ των προτέρων τον νικητή του αγώνα με βάση του ποιος παίκτης εμφανίζεται ως Νικητής, αλλά χρησιμοποιώντας τα υπόλοιπα στοιχειά.

3.2.3 Μετασχηματισμός Δεδομένων

Σε αυτό το στάδιο χρειάστηκε να τροποποιήσουμε τις εγγραφές στο Dataset από δεδομένα τύπου **Σειρών Χαρακτήρων Κειμένου (Strings of Characters, str)** σε δεδομένα τύπου **Ακεραίων Αριθμών (Integer Numbers, int)**. Αυτό κρίθηκε απαραίτητο διότι τα μοντέλα Δένδρων Αποφάσεων της βιβλιοθήκης Scikit-Learn της παρούσας έκδοσης δεν μπορούν να διαχειριστούν καλά δεδομένα πέραν του τύπου **Αριθμών Κινητής Υποδιαστολής (Floating Point Numbers, float)** και ακεραίων. Τα Δένδρα Αποφάσεων παράγουν καλύτερα αποτελέσματα όταν μπορούν να κάνουν εύκολες διακρίσεις, συνεπώς επιλέχθηκε ψηφιοποίηση των τιμών σε ακεραίους. Αυτή η διαδικασία έγινε υλοποιώντας τη συνάρτηση **Encode_Dataset(),** η οποία παίρνει σαν όρισμα το προς ψηφιοποίηση dataset και επιστρέψει ένα Dataset το οποίο απαρτίζεται μόνο από δεδομένα τύπου int ή float με την εξής μέθοδο:

- Για εγγραφές που είχαν διακριτές τιμές, χρησιμοποιήθηκε η συνάρτηση **LabelEncoder()**, η οποία ψηφιοποιεί τις εγγραφές σε σειριακούς ακεραίους. Για παράδειγμα, το πεδίο “Τύπος Γηπέδου” ψηφιοποιήθηκε με “0” για ανοιχτό γήπεδο και “1” για κλειστό γήπεδο.
- Τα πεδία της ημερομηνίας ψηφιοποιήθηκαν ως αριθμοί που περιέχουν σαν πρώτα ψηφία την χρονιά και τα υπόλοιπα ψηφία τον μήνα. Για παράδειγμα, η ημερομηνία “15-05-2004” ψηφιοποιήθηκε ως ο ακέραιος “200405”.

3.3 Μετρικές Αξιολόγησης

Όσον αφορά τις μετρικές που χρησιμοποιήθηκαν για την Πρόβλεψη κατά την πειραματική διαδικασία, έγινε χρήση των μετρικών:

- Accuracy
- Prediction Probability

Η **μετρική Accuracy** χρησιμοποιήθηκε για μια εποπτική ποσοτικοποίηση των επιδόσεων των μοντέλων, ώστε να εκτιμηθεί η δυσκολία του προβλήματος σε όρους εύκολα αντιληπτούς από τον άνθρωπο. Από την άλλη πλευρά, η μετρική **Prediction Probability** αποδείχθηκε πολύ χρήσιμη καθώς καθιστά δυνατή μια ποιοτική εκτίμηση της «αυτοπεποίθησης» των μοντέλων σχετικά με τις προβλέψεις τους, γεγονός αρκετά σημαντικό για προβλήματα διαχείρισης ρίσκου. Ένας Παίκτης Στοιχήματος πρέπει να μπορεί να ποντάρει με κάποια αξιοπιστία και η μετρική **Prediction Probability** ποσοτικοποιεί ακριβώς αυτό.

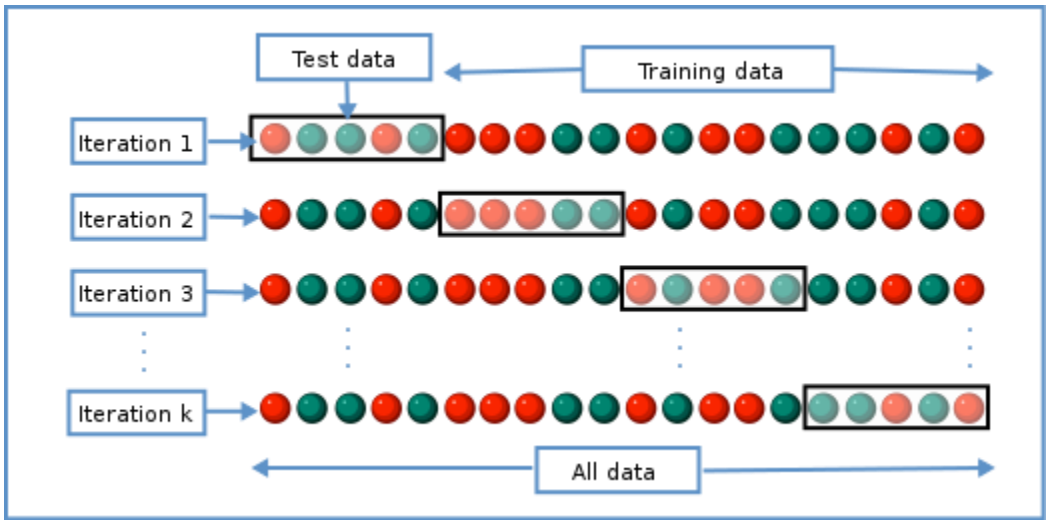
Κεφάλαιο 4: Αποτελέσματα Πρόβλεψης Αποτελέσματος Αγώνα

4.1 Βελτιστοποίηση Μοντέλων Πρόβλεψης

Ένα πολύ σημαντικό τεχνικό στάδιο προτού εφαρμοστούν τα μοντέλα Μηχανικής Μάθησης σε κάποιο πρόβλημα είναι η **Βελτιστοποίηση Υπερπαραμέτρων (Hyperparameter Tuning)**. Κάθε αλγόριθμος Μηχανικής Μάθησης, πέρα από τα δεδομένα που δέχεται ως είσοδο, δέχεται και κάποιες τιμές που τροποποιούν τον τρόπο με τον οποίο θα λειτουργήσει. Ανάλογα τον αλγόριθμο, αυτές οι τιμές μπορεί να είναι κοινές ή διαφορετικές μεταξύ άλλων αλγορίθμων. Για παράδειγμα, τα Δένδρα και τα Δάση έχουν κοινές παραμέτρους, αλλά καθώς τα Δάση απαρτίζονται από Δένδρα, έχουν μια επιπλέον παράμετρο που αφορά το πλήθος των δένδρων που θα χρησιμοποιήσουν. Στόχος είναι να βρούμε τις τιμές εκείνες για τις οποίες τα μοντέλα φαίνεται να κάνουν τις καλύτερες, ως προς κάποια μετρική αξιολόγησης, προβλέψεις.

Για να γίνουν τα πράγματα πιο συγκεκριμένα, πρέπει να οριστούν εκ των προτέρων οι διάφορες τιμές των ποικίλων Υπερπαραμέτρων που θα δοκιμαστούν στα μοντέλα. Αυτός ο συνδυασμός όλων των τιμών των Υπερπαραμέτρων ορίζει έναν **Χώρο Αναζήτησης (Search Space)** για κάθε μοντέλο, στον οποίο χώρο καλούμαστε να βρούμε ποιες τιμές-συντεταγμένες κάνουν την καλύτερη δυνατή πρόβλεψη-κατηγοριοποίηση. Στην πράξη, παράγουμε κάθε μοντέλο με κάθε δυνατό συνδυασμό Υπερπαραμέτρων και εξετάζουμε τις επιδόσεις του.

Το να εκπαιδεύσουμε όμως πολλά μοντέλα, πολλές φορές στο ίδιο Dataset εγκυμονεί τον κίνδυνο της Υπερπροσαρμογής. Όπως εξηγήσαμε σε προηγούμενο κεφάλαιο, κατά την Υπερπροσαρμογή το μοντέλο βγάζει καλές προβλέψεις μόνο για τα δεδομένα που εκπαιδεύτηκε και χάνει την δυνατότητα να γενικεύει σε δεδομένα έξω από το σύνολο εκπαίδευσης. Για να αποφευχθεί λοιπόν ο κίνδυνος της Υπερπροσαρμογής, κατά την διαδικασία της Βελτιστοποίησης εφαρμόζεται το σχήμα της **Αναζήτησης Πλέγματος (Gridsearch).**



Εικόνα 17: Εικονική αναπαράσταση της Αναζήτησης Πλέγματος (Grid Search). Κάθε γραμμή αφορά μια εποχή εκπαίδευσης και αξιολόγησης ενός συνδυασμού Υπερπαραμέτρων για ένα μοντέλο. Κάθε φορά που εκπαιδεύεται ένα μοντέλο (διαφορετική γραμμή), από το αρχικό Train Set (μια ακολουθία από πράσινες και κόκκινες σφαίρες) επιλέγονται διαφορετικοί συνδυασμοί δειγμάτων για Train και Test Sets. Στο τέλος, τα μοντέλα θα έχουν αξιοποιήσει

ολόκληρο το Train Set, αλλά ταυτόχρονα, καθώς θα έχουν εκπαιδευτεί σε ελαφρά διαφορετικά υποσύνολα, θα έχουν περιορίσει τον κίνδυνο της Υπερπροσαρμογής, ενώ ταυτόχρονα, θα έχουν δοκιμάσει τις διάφορες Υπερπαραμέτρους που τους ζητήθηκαν.

Για την συγκεκριμένη εργασία επιλέχθηκαν οι κάτωθι χώροι αναζήτησης:

Πίνακας 3: Χώρος Αναζήτησης Δένδρου Αποφάσεων

Χώρος Αναζήτησης Δένδρου Αποφάσεων

Criterion	Gini			entropy	
Max Depth	4	8	16	32	64
Max Features	sqrt			Log2	

Πίνακας 4: Χώρος Αναζήτησης Τυχαίου Δάσους

Χώρος Αναζήτησης Τυχαίου Δάσους

Criterion	Gini		entropy	
Max Depth	4	8	16	
Max Features	sqrt		Log2	
N Estimators	100	300	500	1000
Max Samples	0.8	0.85	0.9	0.95	1.0

Πίνακας 5: Χώρος Αναζήτησης LightGBM

Χώρος Αναζήτησης LightGBM

Max Depth	1	4	8	16
Max Features	sqrt		Log2	
N Estimators	100	300	500	1000
Sub Sample	0.8	0.85	0.9	0.95	1.0
Col Sample	0.8	0.85	0.9	0.95	1.0
Learning Rate	0.3	0.1	0.05	0.01	0.005	0.001

Τα μοντέλα που θα χρησιμοποιηθούν είναι τα Δένδρα Αποφάσεων, τα Τυχαία Δάση και οι Ταξινομητές Ήπιας Ενίσχυσης Βαθμίδας. Ακολουθεί η επεξήγηση των Υπερπαραμέτρων:

- **Κριτήριο (Criterion)**: Η μετρική με την οποία αποφασίζει κάθε Δένδρο ως προς ποιο χαρακτηριστικό θα κάνει τον κάθε διαχωρισμό. Επιλέχθηκαν δυο εκδοχές:
    - **Εντροπία (Entropy)**: Η ποσοτικοποίηση του κέρδους πληροφορίας του εκάστοτε διαχωρισμού και δημιουργεί διαχωρισμούς με την μικρότερη απώλεια πληροφορίας.
    - **Προσμίξεις Τζίνι (Gini Impurities)**: Η συχνότητα με την οποία ένα τυχαίο δείγμα θα κατηγοριοποιηθεί λανθασμένα, εάν γινόταν κατηγοριοποίηση υπό την

παρούσα κατανομή. Με πιο απλά λόγια, ποσοτικοποιεί την απόκλιση από την αναμενόμενη τιμή και δημιουργεί διαχωρισμούς με την μικρότερη δυνατή τιμή.

- **Μέγιστο Βάθος (Max Depth)**: Το μέγιστο βάθος, δηλαδή ο μέγιστος αριθμός διαχωρισμών, το οποίο επιτρέπεται να φτάσουν τα δένδρα.
- **Μέγιστα Χαρακτηριστικά (Max Features)**: Το μέγιστο Πλήθος των χαρακτηριστικών που θα λαμβάνονται υπόψιν σε κάθε διαχωρισμό.
- **Αριθμός Εκτιμητών (N Estimators)**: Το πλήθος των υπο-μοντέλων που θα συνδυαστούν για να κατασκευάσουν το πλήρες μοντέλο.
- **Μέγιστη Δειγματοληψία (Max Samples, Sub Sample)**: Το μέγιστο ποσοστό δειγματοληψίας από ολόκληρα τα δεδομένα που επιτρέπεται στα υπομοντέλα. Με αυτόν τον τρόπο, κάθε υπομοντέλο στο τυχαίο δάσος εκπαιδεύεται επάνω σε ελαφρά ξεχωριστά δεδομένα.

Τα αποτελέσματα του Gridsearch παρατίθενται στους παρακάτω πίνακες:

Πίνακας 6: Αποτελέσματα Gridsearch για Δένδρο και Δάσος

Μοντέλα Πρόβλεψης	**Criterion**	**Max Depth**	**Max Features**	**N Estimators**	**Max Samples**	**Train Set Accuracy (%)**	**Test Set Accuracy (%)**
Δέντρο	Entropy	4	sqrt	-	-	68.71	66.83
Δάσος	Entropy	4	sqrt	500	0.9	70.18	66.67

Πίνακας 7: Αποτελέσματα Gridsearch gia LightGBM

Μοντέλα Πρόβλεψης	**Col sample By Tree**	**Learning Rate**	**Max Depth**	**N Estimators**	**Max Sample**	**Train Set Accuracy (%)**	**Test Set Accuracy (%)**
LightGBM	0.85	0.005	4	300	0.95	71.98	66.67

4.2 Συγκριτική Ανάλυση Μοντέλων Πρόβλεψης: Πλειοψηφική Εκτίμηση Φαβορί από Bookmakers

Έχοντας βρει τις κατάλληλες παραμέτρους όπου αποδίδουν καλύτερα τα μοντέλα από το συγκεκριμένο Χώρο Αναζήτησης για το ATP Dataset, μπορούμε να περάσουμε στην εκπαίδευση και αξιολόγησή τους. Προτού όμως συμβεί αυτό, θα ήταν φρόνιμο να έχουμε και έναν εκτιμητή που να αντιπροσωπεύει την πρόβλεψη του Νικητή Αθλητή χωρίς κάποιο μοντέλο, ώστε να μπορεί να γίνει κάποια ποιοτική αξιολόγηση των προβλέψεων των μοντέλων σε σχέση με κάποια πρόβλεψη που θα έκανε κάποιος Παίκτης χωρίς την βοήθεια Μηχανικής Μάθησης.

Για τον σκοπό αυτό αναπτύχθηκε η συνάρτηση **Prediction_Per_BetCorp()**, η οποία σε συνδυασμό με την συνάρτηση **predict_Win_by_Corp_Vote()** εφαρμόζουν την Πλειοψηφική Εκτίμηση Φαβορί από Bookmakers. Με απλά λόγια, η συνάρτηση –Παίκτης- βλέπει τι απόδοση δίνει κάθε στοιχηματική και στοιχηματίζει στο “Φαβορί”. Μικρότερη απόδοση συνεπάγεται περισσότερες πιθανότητες να νικήσει ο συγκεκριμένος αθλητής. Αν έχουμε δεδομένα αποδόσεων μόνο για τον ένα αθλητή, μπορούμε να μαντέψουμε τις αποδόσεις για τον αντίπαλο του. Για παράδειγμα, αν ένας παίκτης έχει απόδοση 1.7, τότε πιθανότατα ο αντίπαλος του έχει απόδοση 2.3. Δηλαδή, αν δεν έχουμε αρκετά δεδομένα, μπορούμε να εκτιμήσουμε ποιος Αθλητής είναι το Φαβορί από το πόσο απέχει η απόδοση που του έδωσε κάποιος Bookmaker από το 2.

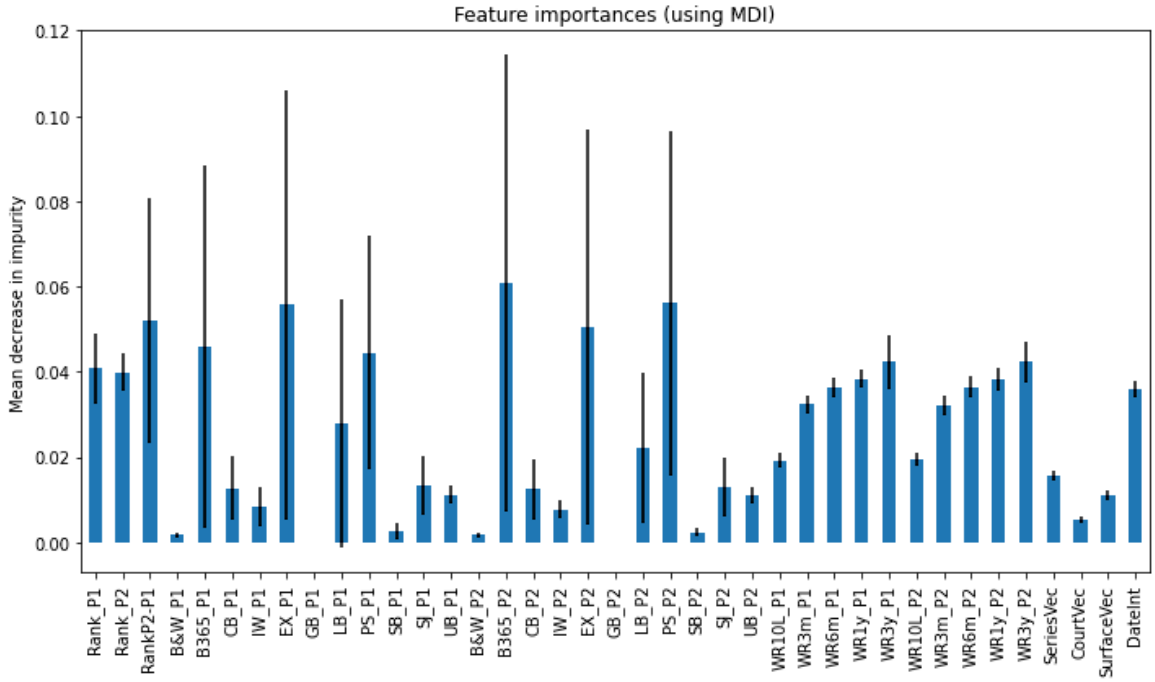
Αφού το Dataset συμπεριλαμβάνει 11 Bookmakers και οι Αθλητές είναι 2 το πλήθος, υπό την υπόθεση πως όλες οι Bookmakers δίνουν αποδόσεις και για τους δυο παίκτες, τότε για κάθε αγώνα έχουμε 22 διαφορετικές αποδόσεις. Συνεπώς, αν μετρήσουμε τις προβλέψεις που κάνει κάθε Στοιχηματική για κάποιον αγώνα, μπορεί να εκτιμηθεί ποιον Αθλητή θεωρούν οι Bookmakers Φαβορί και άρα μπορεί κάποιος παίκτης να στοιχηματίσει σε αυτόν. Συντακτικά αυτό μπορεί να γραφεί ως:

$$\hat{y}_{bench} = mode\{Bet_1(x), Bet_2(x), \dots, Bet_{22}(x)\}$$

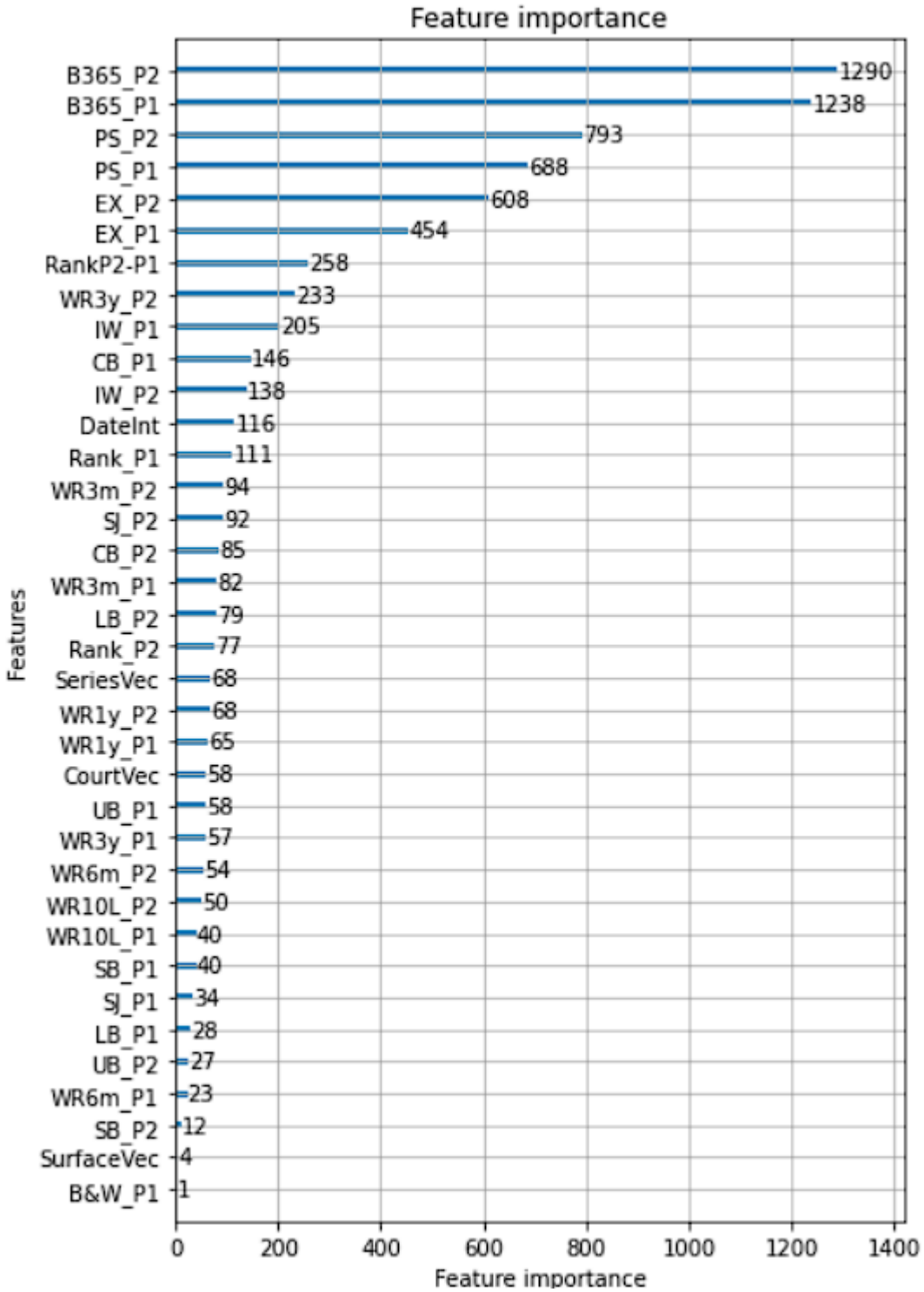
4.3 Ανάλυση Σημαντικότητας Μεταβλητών

Έχοντας κάνει τις απαραίτητες αλλαγές στο Dataset είναι δυνατό σε αυτό το σημείο να γίνει μια επισταμένη στατιστική ανάλυση για να εκτιμηθεί η **Σημαντικότητα των Features (Feature Importance)** που θα χρησιμοποιήσουν τα μοντέλα. Αυτή η ανάλυση προέκυψε χρησιμοποιώντας τις μεθόδους της βιβλιοθήκης Scikit-Learn για Τυχαία Δάση και τις βιβλιοθήκης LightGBM για ταξινομητές Ήπιας Ενίσχυσης Βαθμίδας.

Τα αποτελέσματα και των δύο μεθόδων συγκλίνουν ως προς την Σημαντικότητα των Χαρακτηριστικών με τα πιο σημαντικά να κρίνονται οι αποδόσεις των εταιριών Bet365, Pinnacles Sports και Expect. Επίσης, η διαφορά στην κατάταξη μεταξύ των δύο Αθλητών και η αναλογία των επιτυχημένων αγώνων τους για τα τελευταία τρία χρόνια, έναντι των χαμένων.



Γράφημα 1: Οι μετρικές Feature Importance όπως προκύπτουν από τη βιβλιοθήκη της Scikit-Learn, μέσω Τυχαίου Δάσους



Γράφημα 2: Οι μετρικές Feature Importance όπως προκύπτουν από την βιβλιοθήκη των LightGBM

4.4 Αποτελέσματα και Σχολιασμός

Παρακάτω παρατίθενται τα αποτελέσματα επιδόσεων των μοντέλων:

Πίνακας 8: Αποτελέσματα Προβλέψεων και Χρόνοι Εκπαίδευσης για Όλα τα Μοντέλα

Μοντέλα Πρόβλεψης	**Απλά**		**Βελτιστοποιημένα**	
	Accuracy (%)	**Time (min)**	**Accuracy (%)**	**Time (min)**
Benchmark	66,41	0,006	-	-
Δένδρο	57,08	0,023	66,83	0,09
Δάσος	66,76	0,372	66,68	45,61
LightGBM	66,80	0,020	66,68	10,02

Εκ πρώτης όψεως, τα αποτελέσματα των μοντέλων δεν φαίνονται πολύ ενθαρρυντικά. Το ποσοστό ακρίβειας που επιτυγχάνεται από τα μοντέλα Μηχανικής Μάθησης είναι με τα βίας καλύτερο από την «απλή» εκτίμηση κάποιου παίκτη με βάση τα φαβορί των Bookmaker.

Αν συμπεριληφθεί και ο χρόνος -και κατά συνέπεια η υπολογιστική ισχύς και το ενεργειακό κόστος- ο οποίος χρειάστηκε για να βελτιστοποιηθούν τα μοντέλα και να αποφανθούν για τον νικητή κάθε αγώνα, φαίνεται πως η μέθοδος που ακολουθήθηκε είναι συγκρίσιμη, αν όχι χειρότερη, από την απλή εκτίμηση.

Λόγοι που μπορεί να συμβαίνει αυτό είναι οι εξής:

- Οι στοιχηματικές εταιρίες είναι ήδη πολύ καλό Χαρακτηριστικό εκτίμησης της έκβασης ενός Αγώνα. Συνεπώς, ακόμα και η απλή εκτίμηση ενός Παίκτη παρουσιάζει αξιόλογα ποσοστά επιτυχίας. Ενδείξεις για αυτή την αιτία αποτελούν οι εικόνες των Feature Importance που παρουσιάστηκαν στο κεφάλαιο Στατιστική Ανάλυση του Dataset.
- Το πλήθος ή το ποιόν των Χαρακτηριστικών δεν είναι κατάλληλο. Χωρίς να λαμβάνονται υπόψιν περισσότερα δεδομένα για τις αθλητικές επιδόσεις των αθλητών, τα μοντέλα περιορίζονται στα στοιχεία που έχουν για την έκβαση του αγώνα.
- Πιθανώς, το συγκεκριμένο Dataset να μην είναι επαρκές για τον σκοπό για τον οποίο χρησιμοποιήθηκε.

Από την άλλη πλευρά, στην Μηχανική Μάθηση δεν είναι όλα τα προβλήματα αντιμετωπίσημα/επιλύσιμα στον ίδιο βαθμό. Κάποια εξ' αυτών των προβλημάτων, αγγίζουν ποσοστά ακρίβειας της τάξης του 90%, ενώ άλλα προβλήματα οριακά αγγίζουν το 60%. Το συγκεκριμένο πρόβλημα για αυτό το Dataset φαίνεται να είναι της τάξης του 70%, το οποίο φαίνεται συνηθισμένο στην βιβλιογραφία. Όπως αναφέραμε στα πρώτα κεφάλαια, η έκβαση αποτελεσμάτων στον αθλητισμό είναι ένα πολυπαραγοντικό και δύσκολο εν γένει πρόβλημα. Το ποσοστό της τάξης του 70% φαίνεται να είναι αρκετό για να μπορέσει κάποιος να στοιχηματίσει με κάποια άνεση.

Παρόλα αυτά, όπως αναφέρθηκε σε προηγούμενα κεφάλαια και όπως θα γίνει αντιληπτό στην συνέχεια, η μετρική Ακρίβειας δεν είναι η μόνη ένδειξη της ποιότητας των

μοντέλων. Μια άλλη πολύ σημαντική μετρική είναι η σιγουριά με την οποία γίνεται μια πρόβλεψη. Μπορεί δηλαδή τα μοντέλα μας, εν τέλει να μην κάνουν πολύ πιο συχνά επιτυχημένες προβλέψεις, αλλά κάνουν σωστές προβλέψεις με περισσότερη αυτοπεποίθηση, με αποτέλεσμα να είναι εφικτό να αποφύγουμε κάποιες παρακινδυνευμένες εκτιμήσεις. Αυτό όμως θα σχολιαστεί στο επόμενο κεφάλαιο που αφορά τις στοιχηματικές στρατηγικές.

Κεφάλαιο 5: Εφαρμογή Στοιχηματικών Στρατηγικών υποβοηθούμενη από Μοντέλα Μηχανικής Μάθησης

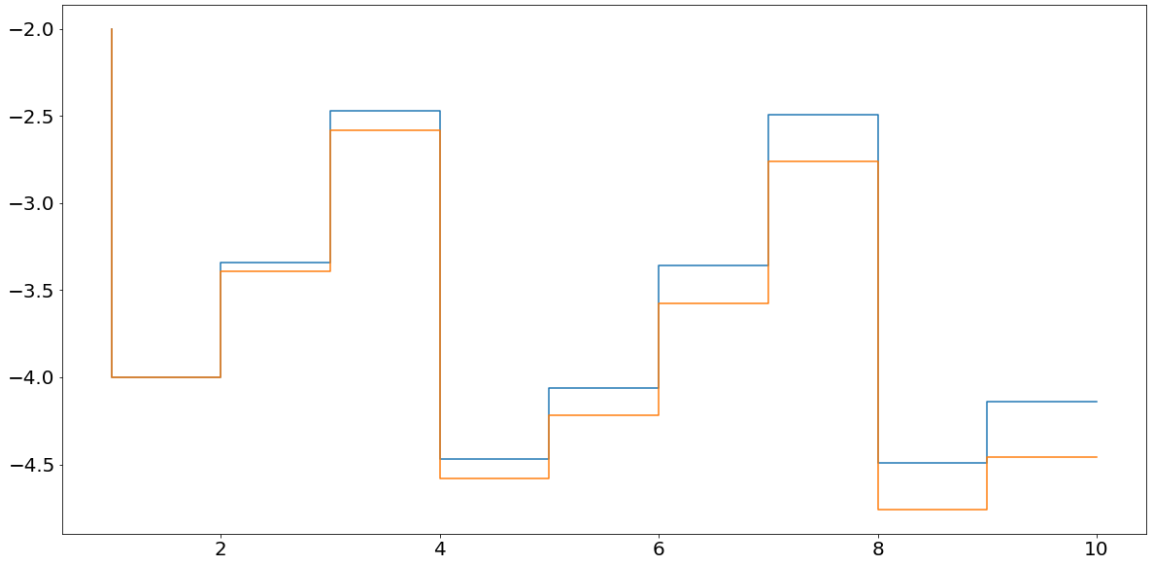
5.1 Απλό Στοίχημα σε όλους τους αγώνες σε όλες τις στοιχηματικές, με ένα ευρώ ανά στοίχημα

Έχοντας πλέον εκπαιδευμένα και βελτιστοποιημένα μοντέλα Μηχανικής Μάθησης για πρόβλεψη αποτελεσμάτων αγώνων Τένις, μπορούμε πλέον να περάσουμε στην ανάπτυξη στοιχηματικών στρατηγικών. Οι στρατηγικές που θα μελετήσουμε αξιολογούνται με βάση τις μετρικές στρατηγικών που αναφέρθηκαν στα προηγούμενα κεφάλαια.

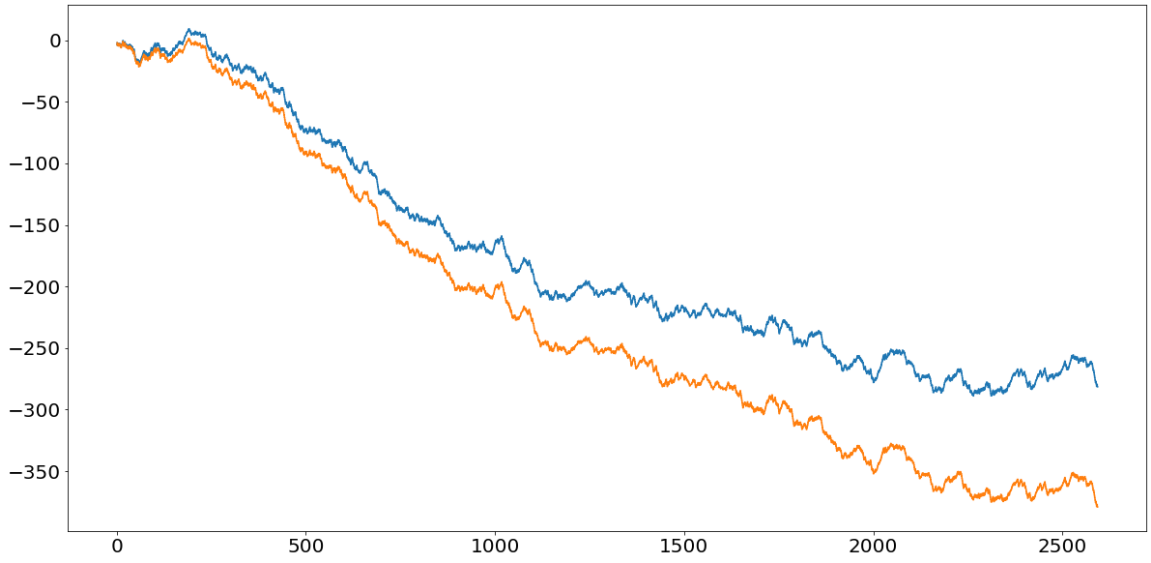
Μια τελείως τυχαία μέθοδος επιλογής αθλητή, πέραν του ότι δεν παράγει καλά αποτελέσματα, δεν αποτελεί στρατηγική υπό την έννοια ότι δεν μπορεί κάποιος ενδιαφερόμενος παίκτης να την ακολουθήσει και να αναμένει πως θα έχει κέρδος μετά από Κ αγώνες με κάποια αξιοπιστία. Ιδανικά, θέλουμε μια μέθοδο που να υποδηλώνει με κάποιον σαφή τρόπο σε ποιον αθλητή αξίζει να στοιχηματίσει ένας Παίκτης για κάθε αγώνα ξεχωριστά.

Μία κάπως καλύτερη και σαφέστερη Στρατηγική είναι ο Παίκτης να στοιχηματίσει σε όλους τους αγώνες στο εκτιμώμενο Φαβορί ή στο εκτιμώμενο αουτσάιντερ, παίζοντας ένα ευρώ σε κάθε αγώνα. Τα αποτελέσματα παρατίθενται στην πιο κάτω εικόνα.

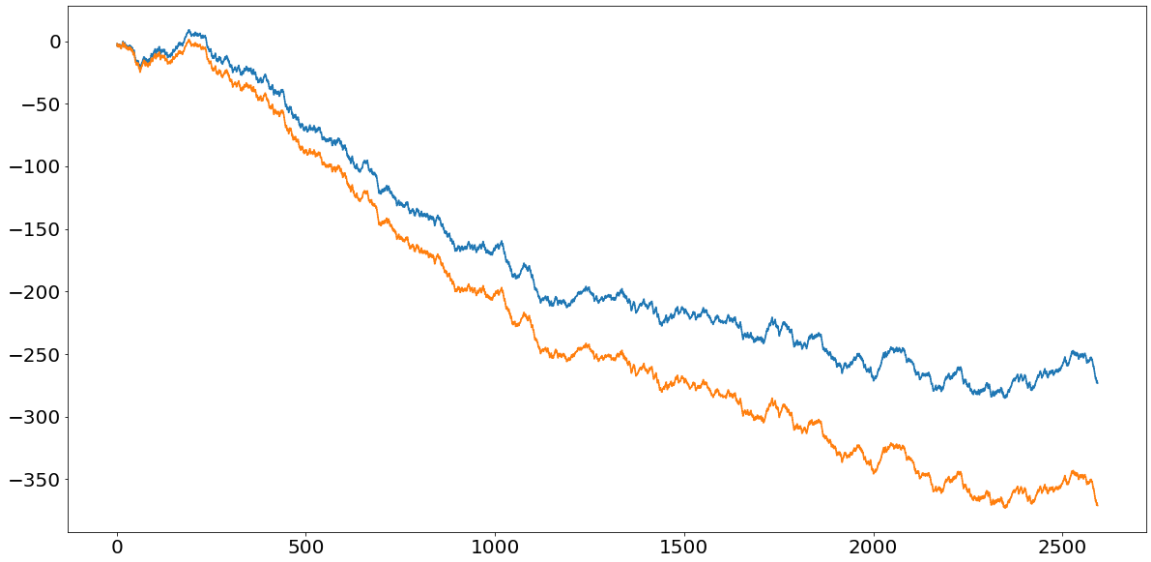
Μια βελτίωση της προηγούμενης τακτικής είναι ο παίκτης να αξιοποιήσει τα παραγμένα μοντέλα Μηχανικής Μάθησης και να στοιχηματίσει στον Εκτιμώμενο Νικητή Αθλητή που αυτά υποδεικνύουν. Συνεπώς, η μέθοδος προσπαθεί να κάνει μια καλύτερη εκτίμηση του φαβορί. Ο παίκτης θα στοιχηματίσει σε όλους τους αγώνες, και σε κάθε αγώνα έναντι όλων των διαθέσιμων Bookmakers για τον αγώνα αυτό, παίζοντας ένα ευρώ σε κάθε στοίχημα.



Γράφημα 3: Ενδεικτικό Διάγραμμα Κεφαλαίου-Αγώνα Αγώνα με Απλό Στοίχημα σε Όλους τους Αγώνες σε Όλες τις Στοιχηματικές, με Ένα Ευρώ ανά Στοίχημα με πρόβλεψη κατά Benchmark για τους πρώτους 10 αγώνες. Η μπλε γραμμή αφορά προσομοίωση κατά την οποία η γκανιότα δεν λήφθηκε υπόψιν, ενώ με πορτοκαλί είναι η γραφική που αφορά την προσομοίωση στην οποία η γκανιότα λήφθηκε υπόψιν.



Γράφημα 4: Διάγραμμα Κεφαλαίου-Αγώνα με Απλό Στοίχημα σε Όλους τους Αγώνες σε Όλες τις Στοιχηματικές, με Ένα Ευρώ ανά Στοίχημα με πρόβλεψη κατά Benchmark, χωρίς (μπλέ) και με (πορτοκαλί) γκανιότα.



Γράφημα 5: Διάγραμμα Κεφαλαίου-Αγώνα με Απλό Στοίχημα σε Όλους τους Αγώνες σε Όλες τις Στοιχηματικές, με Ένα Ευρώ ανά Στοίχημα με πρόβλεψη κατά Τυχαίο Δάσος, χωρίς (μπλε) και με (πορτοκαλί) γκανιότα.

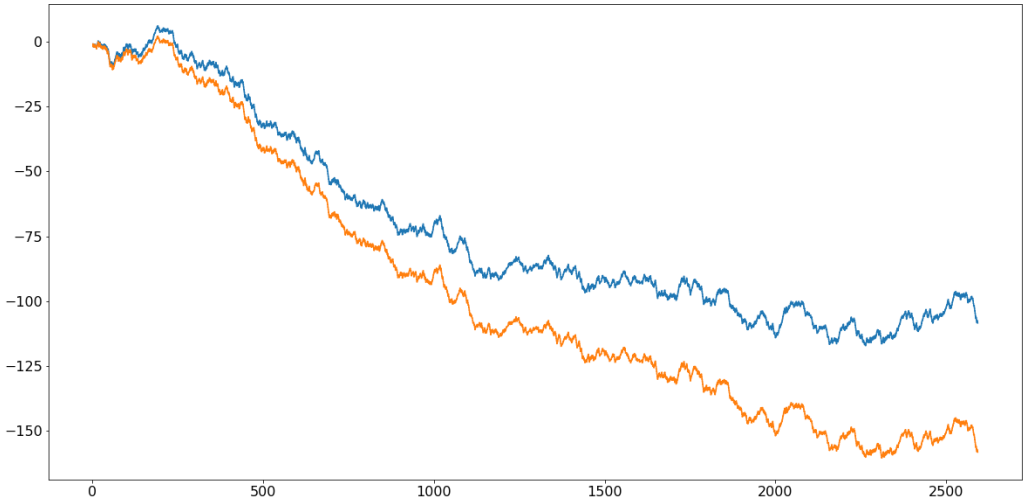
Τα αποτελέσματα παρατέθηκαν στις προηγούμενες εικόνες. Σε κάθε γράφημα με μπλε χρώμα είναι το τρέχον κεφάλαιο του παίκτη μετά από κάθε αγώνα, ενώ στο γράφημα με πορτοκαλί έχει συνυπολογιστεί η «γκανιότα», δηλαδή αντιπροσωπεύει το καθαρό κέρδος του παίκτη.

Μια πρώτη παρατήρηση είναι πως όλα τα γραφήματα διαγράφουν καθοδική πορεία. Αυτό συμβαίνει αφενός διότι τα κέρδη από επιτυχημένους αγώνες είναι πολύ μικρότερα κατά μέσο όρο από το ποσό που στοιχηματίζεται ανά αγώνα, αφετέρου διότι για κάθε αγώνα λαμβάνονται πολλά στοιχήματα με διαφορετικές αποδόσεις. Συνεπώς, διαδοχικές ήττες, ακόμα και εάν είναι λίγες το πλήθος, μπορεί να δημιουργούν μακροπρόθεσμα ζημιά.

Η ομοιότητα των δύο διαγραμμάτων είναι εκπληκτική, παρότι αμφότερα χρησιμοποιούν διαφορετικούς μηχανισμούς πρόβλεψης. Αυτό το φαινόμενο μπορεί να αποδοθεί στο γεγονός πως, παρότι τα μοντέλα μηχανικής μάθησης δέχονται σαν είσοδο περισσότερα δεδομένα, βασίζουν τις προβλέψεις τους στα ίδια χαρακτηριστικά με την benchmark μέθοδο, δηλαδή στις αποδόσεις των bookmakers. Αυτό έχει σαν αποτέλεσμα να κάνουν πανομοιότυπες εκτιμήσεις σχετικά με τον νικητή και άρα οι γραφικές παραστάσεις και των δύο μεθόδων να έχουν την ίδια συμπεριφορά.

5.2 Απλό Στοίχημα σε Όλους τους Αγώνες στην Καλύτερη Στοιχηματική, με Ένα Ευρώ ανά Αγώνα

Αντί να παίξουμε σε όλες τις στοιχηματικές εταιρίες, μπορούμε απλά να παίξουμε στην καλύτερη απόδοση που δίνει ένας και μοναδικός Bookmaker. Κρατώντας τα υπόλοιπα δεδομένα ίδια, παίρνουμε τα εξής αποτελέσματα από την προσομοίωση.



Γράφημα 6: Διάγραμμα Κεφαλαίου-Αγώνα με Απλό Στοίχημα σε Όλους τους Αγώνες στην Καλύτερη Στοιχηματική, με Ένα Ευρώ ανά Στοίχημα με πρόβλεψη κατά Benchmark, χωρίς (μπλε) και με (πορτοκαλί) γκανιότα.

Γράφημα 7: Διάγραμμα Κεφαλαίου-Αγώνα με Απλό Στοίχημα σε Όλους τους Αγώνες στην Καλύτερη Στοιχηματική, με Ένα Ευρώ ανά Στοίχημα με πρόβλεψη κατά Τυχαίο Δάσος, χωρίς (μπλε) και με (πορτοκαλί) γκανιότα.

Τα αποτελέσματα των προσομοιώσεων συνοψίζονται στον παρακάτω πίνακα:

Πίνακας 9: Αποτελέσματα Προσομοίωσης Απλού Στοιχήματος Χωρίς Ρήτρα Σιγουριάς

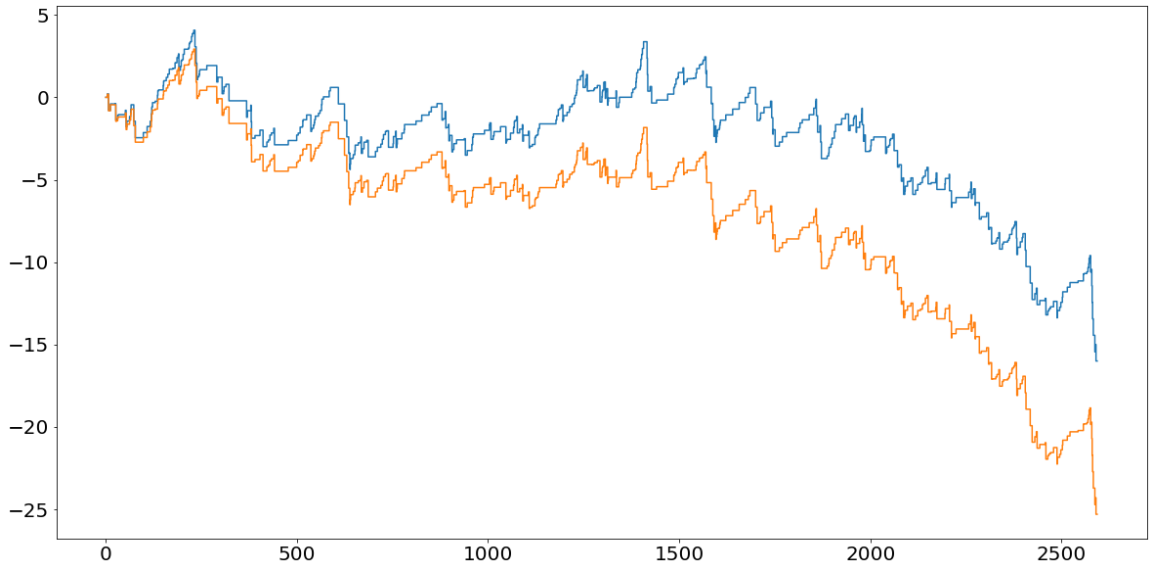
Μοντέλα Πρόβλεψης	**Capital**				**Ποσοστιαίο Κέρδος Ανά Επένδυση (%)**
	Final	**Average**	**Max**	**Min**	
Benchmark	-158.12	-96.77	2.00	-160.33	-6.09
Δένδρο	-192.00	-117.56	2.26	-193.36	-7.40
Δάσος	-120.27	-70.61	9.47	-126.65	-4.07
LightGBM	-111.20	-59.44	6.51	-112.33	-4.28

Σε αυτή την προσομοίωση παρατηρείται πάλι μια φθίνουσα πορεία, αλλά αυτή τη φορά το μοντέλο του Τυχαίου Δάσους φαίνεται να ανταποκρίνεται καλύτερα και να διαφοροποιείται από το benchmark μοντέλο. Αυτό μπορεί να εξηγηθεί διότι αυτή τη φορά παίζεται ένα και μοναδικό παιχνίδι ανά αγώνα και μάλιστα στην καλύτερη δυνατή απόδοση, συνεπώς είναι πιο εύκολο να αντισταθμιστούν οι ζημιές από αποτυχημένα παιχνίδια.

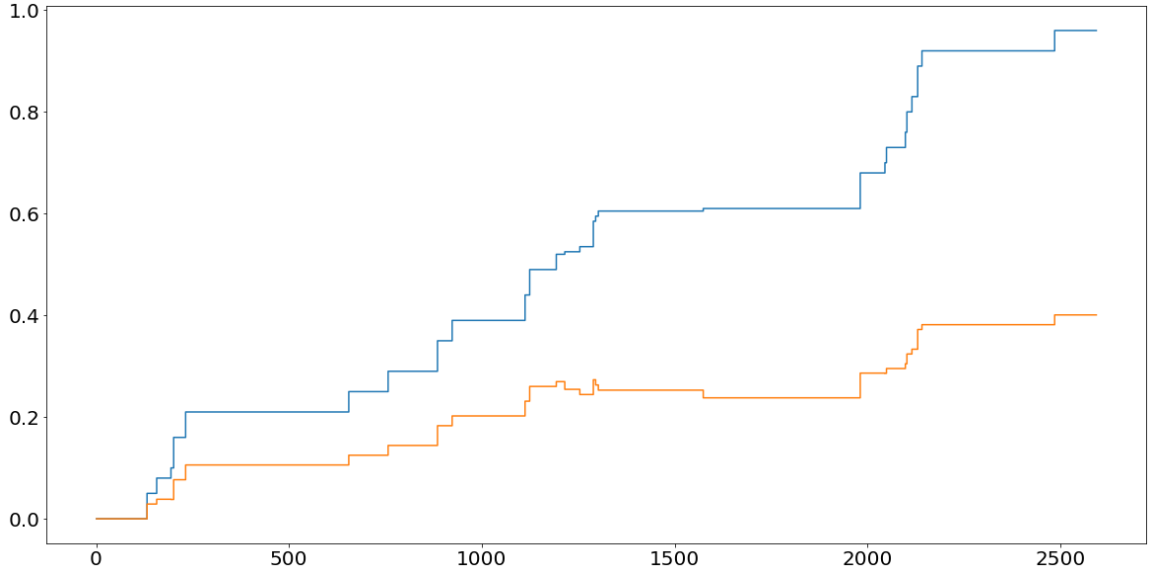
Αξιοσημείωτη επίδοση παρατηρείται επίσης και στο μοντέλο LightGBM, το οποίο επιτυγχάνει ελάχιστα καλύτερο τελικό κέρδος και κέρδος κατά μέσο όρο, αλλά μικρότερο μέγιστο κέρδος. Αυτό το γεγονός θα μπορούσε να χαρακτηρίσει την παρούσα υλοποίηση του LightGBM ως έναν πιο συντηρητικό στοιχηματικό παίκτη.

5.3 Απλό στοίχημα στην καλύτερη στοιχηματική σε αγώνες με ρήτρα σιγουριάς εκτίμησης, με ένα ευρώ ανά στοίχημα

Ένας πιο ρεαλιστικός παίκτης δεν θα στοιχημάτιζε σε όλους τους αγώνες, αλλά σε αυτούς που θα εκτιμούσε το φαβορί με την καλύτερη αξιοπιστία. Αυτό μπορεί να συμπεριληφθεί στην προσομοίωση με μια απλή διαδικασία ρήτρας, καθώς τα εκπαιδευμένα μοντέλα μπορούν να υπολογίσουν την στοχαστική αξιοπιστία των προβλέψεων που κάνουν, βάσει των δεδομένων στα οποία εκπαιδεύτηκαν όπως εξηγείται στο Κεφάλαιο 2.



Γράφημα 8: Διάγραμμα Κεφαλαίου-Αγώνα με Απλό Στοίχημα στην Καλύτερη Στοιχηματική με Ρήτρα Σιγουριάς Εκτίμησης 0.77, με Ένα Ευρώ ανά Στοίχημα με πρόβλεψη κατά Δένδρο, χωρίς (μπλε) και με (πορτοκαλί) γκανιότα.



Γράφημα 9: Διάγραμμα Κεφαλαίου-Αγώνα με Απλό Στοίχημα στην Καλύτερη Στοιχηματική με Ρήτρα Σιγουριάς Εκτίμησης 0.77, με Ένα Ευρώ ανά Στοίχημα με πρόβλεψη κατά Τυχαίο Δάσος, χωρίς (μπλε) και με (πορτοκαλί) γκανιότα.

Τα αποτελέσματα των προσομοιώσεων συνοψίζονται στους παρακάτω πίνακες:

Πίνακας 10: Αποτελέσματα Απλού Στοιχήματος Με Ρήτρα Εκτίμησης 0.77 Χωρίς Γκανιότα

Απλό Στοίχημα Στην Καλύτερη Στοιχηματική με Ρήτρα Εκτίμησης 0.77 Χωρίς Γκανιότα

Μοντέλο	Καθαρό Κέρδος (€)	Κέρδη (€)	Ζημίες (€)	Αριθμός Αγώνων (#)	Τάξη Κεφαλαίου (€)	Μέγιστο Ποσό Σε Στοίχημα (€)	Ποσοστιαίο Κέρδος Ανά Επένδυση (%)
Benchmark	-	-	-	-	-	-	-
Δένδρο	-16	85	101	481	481	1	-3.33
Δάσος	0.96	0.96	0	27	27	1	3.56
LightGBM	-15.09	54.91	70	475	475	1	-3.18

Πίνακας 11: Αποτελέσματα Απλού Στοιχήματος Με Ρήτρα Εκτίμησης 0.77 με Γκανιότα

Απλό Στοίχημα Στην Καλύτερη Στοιχηματική με Ρήτρα Εκτίμησης 0.77 Με Γκανιότα

Μοντέλο	Καθαρό Κέρδος (€)	Κέρδη (€)	Ζημίες (€)	Αριθμός Αγώνων (#)	Τάξη Κεφαλαίου (€)	Μέγιστο Ποσό Σε Στοίχημα (€)	Ποσοστιαίο Κέρδος Ανά Επένδυση (%)
Benchmark	-	-	-	-	-	-	-
Δένδρο	-25.3	75.7	101	481	481	1	-5.26
Δάσος	0.4	0.4	0	27	27	1	1.48
LightGBM	-24.29	45.71	70	475	475	1	-5.11

Με αυτή την μέθοδο παρατηρείται δραματική βελτίωση στην επίδοση των μοντέλων, όσον αφορά το τελικό καθαρό κέρδος. Παρόλα αυτά οι αριθμοί δεν πρέπει να μας μπερδέψουν καθώς και σε αυτή την προσομοίωση, το πλήθος των αγώνων δεν παραμένει σταθερό και το κεφάλαιο που στοιχηματίζεται είναι σαφώς μικρότερο.

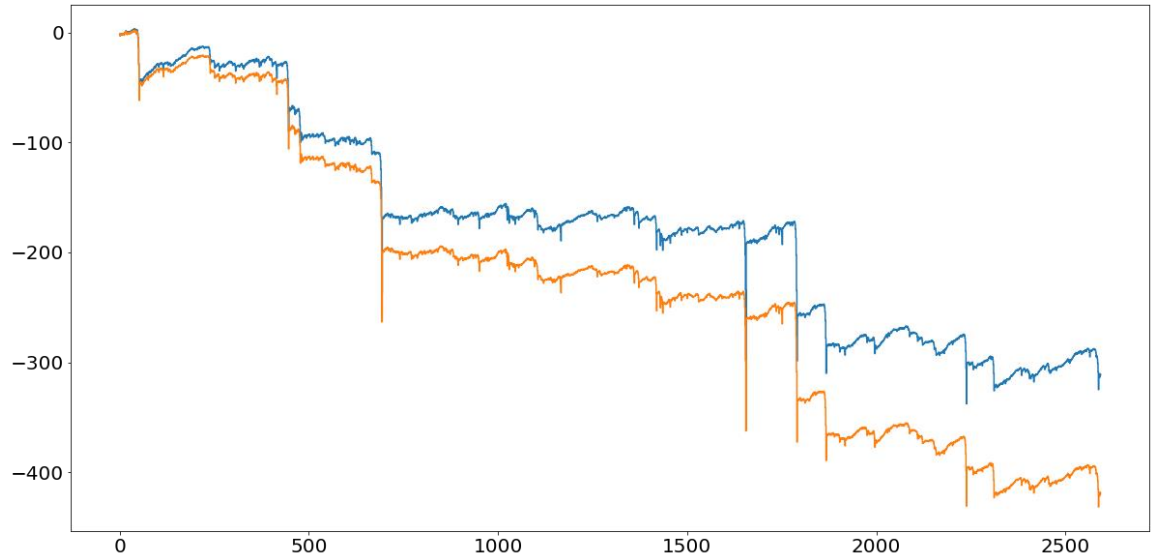
Τα μοντέλα Δένδρου και LightGBM παρουσιάζουν αντίστοιχες επιδόσεις, με το LightGBM να επιτυγχάνει ελαφρώς καλύτερες τιμές, αλλά και τα δυο να εξακολουθούν να μην καταφέρνουν να επιτύχουν θετικό κέρδος, έστω και αν κατάφεραν να περιορίσουν σημαντικά την ζημία των προηγούμενων προσομοιώσεων.

Αντίθετα, το μοντέλο του Τυχαίου Δάσους είναι το πρώτο που πετυχαίνει θετικό κέρδος. Το κέρδος όμως αυτό, είναι μηδαμινό μπροστά στο κεφάλαιο που στοιχηματίστηκε για να το παραγάγει. Το Τυχαίο Δάσος στην προκειμένη προσομοίωση παρουσιάζει μια μονότονη (ή σχεδόν μονότονη στην περίπτωση της γκανιότας) αύξηση. Αυτό προφανώς είναι μια τυχαία συμπεριφορά του μοντέλου και αποτέλεσμα του ότι τέθηκε υψηλό κατώφλι για την αξιοπιστία της πρόβλεψης. Σε καμία περίπτωση όμως δεν είναι ενδεικτικό της συμπεριφοράς του μοντέλου στη γενική περίπτωση.

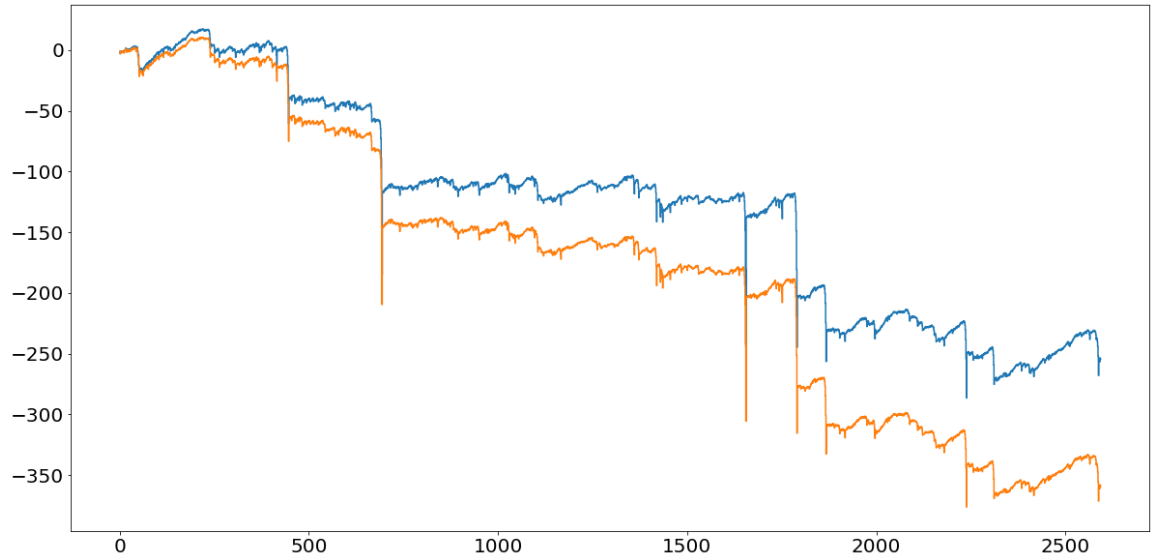
5.4 Στοίχημα με μέθοδο Martingale σε όλους τους αγώνες με διπλασιασμό Πονταρίσματος σε κάθε ήττα

Ας μελετήσουμε τώρα την επίδοση της στοιχηματικής μεθόδου Martingale, η οποία παρουσιάστηκε λεπτομερώς στο Κεφάλαιο 1.

Σε αυτή την προσομοίωση, ο Παίκτης παίζει σε όλους τους αγώνες, στοιχηματίζοντας ένα ευρώ στην καλύτερη απόδοση σε κάθε αγώνα. Θυμίζουμε πως σε κάθε ήττα το ποσό που θα ποντάριστεί, διπλασιάζεται, με σκοπό να ισοζυγιστεί η ζημία των προηγούμενων αγώνων. Σε περίπτωση νίκης, το επόμενο ποντάρισμα ξεκινάει ξανά από το αρχικό ποσό του ενός ευρώ.



Γράφημα 10: Διάγραμμα Κεφαλαίου-Αγώνα με Μέθοδο Martingale στην Καλύτερη Στοιχηματική, με Ένα Ευρώ ανά Στοίχημα, διπλασιασμό σε Ήττα, με πρόβλεψη κατά Benchmark, χωρίς (μπλε) και με (πορτοκαλί) γκανιότα.



Γράφημα 11: Διάγραμμα Κεφαλαίου-Αγώνα με Μέθοδο Martingale στην Καλύτερη Στοιχηματική, με Ένα Ευρώ ανά Στοίχημα, διπλασιασμό σε Ήττα, με πρόβλεψη κατά Τυχαίο Δάσος, χωρίς (μπλε) και με (πορτοκαλί) γκανιότα.

Τα αποτελέσματα των προσομοιώσεων συνοψίζονται στους παρακάτω πίνακες:

Πίνακας 12: Αποτελέσματα Στοιχήματος κατά Martingale Χωρίς Γκανιότα

Στοίχημα κατά Martingale Σε Κάθε Αγώνα Στην Καλύτερη Στοιχηματική Χωρίς Γκανιότα

Μοντέλο	Καθαρό Κέρδος (€)	Κέρδη (€)	Ζημίες (€)	Αριθμός Αγώνων (#)	Τάξη Κεφαλαίου (€)	Μέγιστο Ποσό Σε Στοίχημα (€)	Ποσοστιαίο Κέρδος Ανά Επένδυση (%)
Benchmark	-311.32	1686.68	1998	2593	5705	128	-5.46
Δένδρο	-257.27	1679.73	1937	2593	5583	128	-4.61
Δάσος	-254.49	1641.51	1896	2593	5478	128	-4.65
LightGBM	-267.65	1635.35	1903	2593	5484	128	-4.88

Πίνακας 13: Αποτελέσματα Στοιχήματος κατά Martingale με Γκανιότα

Στοίχημα κατά Martingale Σε Κάθε Αγώνα Στην Καλύτερη Στοιχηματική Με Γκανιότα

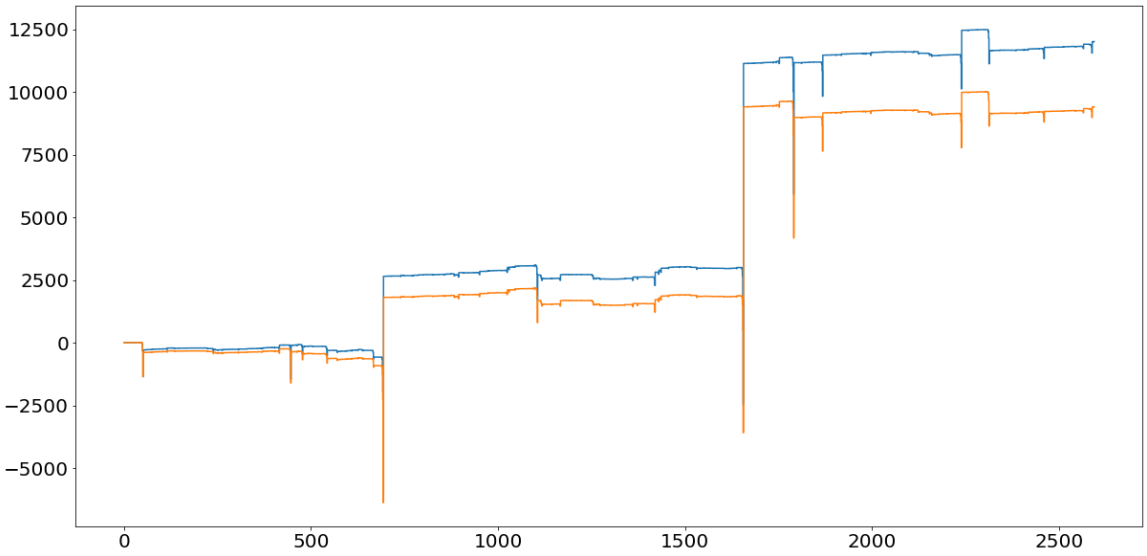
Μοντέλο	Καθαρό Κέρδος (€)	Κέρδη (€)	Ζημίες (€)	Αριθμός Αγώνων (#)	Τάξη Κεφαλαίου (€)	Μέγιστο Ποσό Σε Στοίχημα (€)	Ποσοστιαίο Κέρδος Ανά Επένδυση (%)
Benchmark	-419.19	1578.81	1998	2593	5705	128	-7.35
Δένδρο	-363.78	1573.22	1937	2593	5583	128	-6.52
Δάσος	-254.49	1641.51	1896	2593	5478	128	-6.55
LightGBM	-371.98	1531.02	1903	2593	5484	128	-6.78

Για άλλη μια φορά παρατηρούμε πως το διάγραμμα του Τυχαίου Δάσους ταυτίζεται με αυτό των Benchmark προβλέψεων, το οποίο οφείλεται στον κοινό τρόπο με τον οποίο τα δύο μοντέλα κάνουν τις προβλέψεις τους. Η διαφορά όμως τώρα έγκειται στην μορφή των γραφικών. Παρατηρείται και πάλι φθίνουσα πορεία, καθώς μετά από σειριακές ήττες ο απλός διπλασιασμός των κερδών δεν είναι αρκετός για να καλύψει τις ζημιές που δημιουργούνται από τις ήττες. Η καλύτερη επίδοση παρουσιάζεται για άλλη μια φορά στο μοντέλο του Τυχαίου δάσους, όσον αφορά πάντα το καθαρό κέρδος.

5.5 Στοίχημα με επιθετική Martingale με τετραπλασιασμό πονταρίσματος σε κάθε ήττα, ρήτρα σιγουριάς και ρήτρα αποδόσεων

Ο τελευταίος κύκλος προσομοιώσεων της παρούσας εργασίας αφορά την διερεύνηση μιας πιο επιθετικής (κατά Martingale) στρατηγικής, η οποία ιδανικά έχει ως στόχο να αποφέρει κέρδη με κάποια αξιοπιστία. Για τον σκοπό αυτό, έγινε εφαρμογή ρήτρας τόσο στην αξιοπιστία των μοντέλων, όσο και στην απόδοση του εκάστοτε παιχνιδιού. Επίσης, επιλέχθηκε ο συντελεστής Martingale αντί για 2 να γίνει 4.

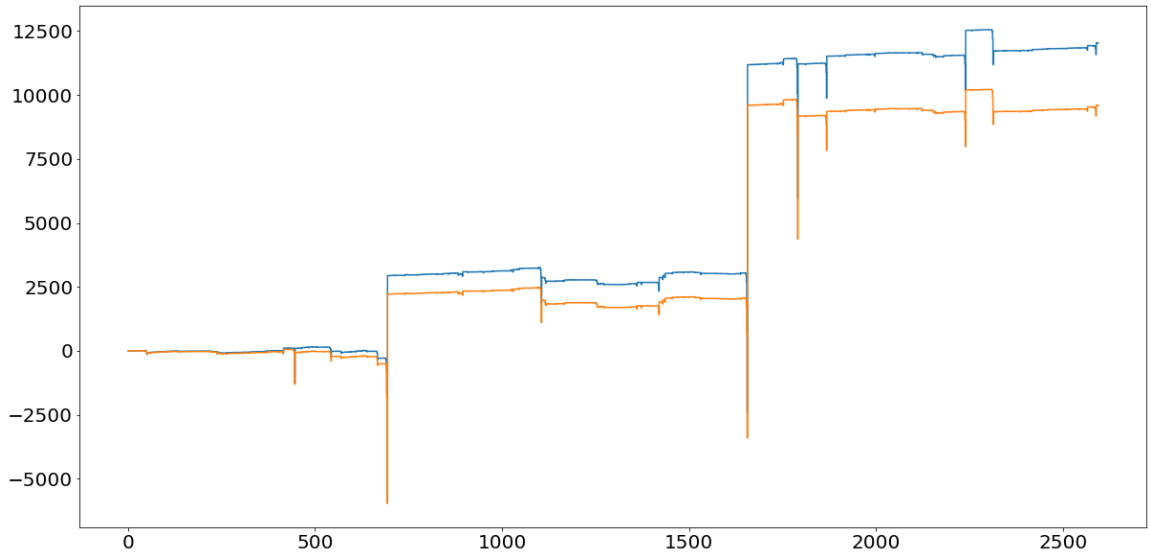
Συνεπώς, σε κάθε διαδοχική ήττα το επόμενο ποντάρισμα τετραπλασιάζεται αντί να διπλασιάζεται.



Γράφημα 12: Διάγραμμα Κεφαλαίου-Αγώνα με Μέθοδο Martingale στην Καλύτερη Στοιχηματική, με Ένα Ευρώ ανά Στοίχημα, Ρήτρα Απόδοσης Αγώνα, Ρήτρα Εκτίμησης Μοντέλου, Τετραπλασιασμό σε Ήττα, με πρόβλεψη κατά Benchmark, χωρίς (μπλε) και με (πορτοκαλί) γκανιότα.

Έχοντας αξιολογήσει ορθά το πρόβλημα που παρουσιάστηκε στις προηγούμενες προσομοιώσεις, είναι εμφανές πως η αύξηση του πολλαπλασιαστή Martingale είχε τα επιθυμητά αποτελέσματα. Με διαδοχικούς τετραπλασιασμούς σε κάθε ήττα, η επόμενη νίκη συνήθως είναι αρκετή για να υπερκαλύψει τις ζημίες.

Αξίζει να παρατηρήσουμε πως υπάρχουν μερικές περιοχές στο διάγραμμα όπου μετά από διαδοχικές ήττες, δεν παρατηρείται απότομη αύξηση. Αυτό συμβαίνει διότι ο επόμενος αγώνας μετά από ένα σερί χαμένων αγώνων (ηττών) έχει πολύ μικρή απόδοση. Αυτό έχει σαν αποτέλεσμα τα κέρδη μας, παρότι πολλαπλασιάζονται με έναν μεγάλο αριθμό να μην φτάσουν την τιμή της ζημίας.



Γράφημα 13: Διάγραμμα Κεφαλαίου-Αγώνα με Μέθοδο Martingale στην Καλύτερη Στοιχηματική, με Ένα Ευρώ ανά Στοίχημα, Ρήτρα Απόδοσης Αγώνα, Ρήτρα Εκτίμησης Μοντέλου, Τετραπλασιασμό σε Ήττα, με πρόβλεψη κατά Τυχαίο Δάσος, χωρίς (μπλε) και με (πορτοκαλί) γκανιότα.

Πίνακας 14: Αποτελέσματα Στοιχήματος κατά Επιθετική Martingale Με Ρήτρα Σιγουριάς 0.51, x4 ανά Ήττα, Χωρίς Γκανιότα

Στοίχημα κατά Martingale Στην Καλύτερη Στοιχηματική Με Ρήτρα Σιγουριάς 0.51, Ρήτρα Απόδοσης 1.15, x4 Ανά Ήττα, Χωρίς Γκανιότα

Μοντέλο	Καθαρό Κέρδος (€)	Κέρδη (€)	Ζημίες (€)	Αριθμός Αγώνων (#)	Τάξη Κεφαλαίου (€)	Μέγιστο Ποσό Σε Στοίχημα (€)	Ποσοστιαίο Κέρδος Ανά Επένδυση (%)
Benchmark	12012.92	43069.92	31057	2593	118036	16384	10.18
Δένδρο	11935.04	41789.04	29854	2593	113377	16384	10.53
Δάσος	12038.49	40962.49	28924	2570	109421	16384	11.00
LightGBM	12087.53	41188.53	29101	2561	110066	16384	10.98

Πίνακας 15. Αποτελέσματα Στοιχήματος κατά Επιθετική Martingale Με Ρήτρα Σιγουριάς 0.51, x4 ανά Ήττα, με Γκανιότα

Στοίχημα κατά Martingale Στην Καλύτερη Στοιχηματική Με Ρήτρα Σιγουριάς 0.51, Ρήτρα Απόδοσης 1.15, x4 Ανά Ήττα, Με Γκανιότα

Μοντέλο	Καθαρό Κέρδος (€)	Κέρδη (€)	Ζημίες (€)	Αριθμός Αγώνων (#)	Τάξη Κεφαλαίου (€)	Μέγιστο Ποσό Σε Στοίχημα (€)	Ποσοστιαίο Κέρδος Ανά Επένδυση (%)
Benchmark	9411.94	40468.94	31057	2593	118036	16384	7.97
Δένδρο	9428.80	39282.80	29854	2593	113377	16384	8.32
Δάσος	9603.30	38533.30	28924	2570	109421	16384	8.78
LightGBM	9644.46	38745.46	29101	2561	110066	16384	8.76

Κεφάλαιο 6: Συμπεράσματα και Προεκτάσεις

6.1 Συμπεράσματα

6.1.1 Συμπεράσματα ως προς τα μοντέλα Μηχανικής Μάθησης

Το συμπέρασμα των αποτελεσμάτων από αυτή την εργασία όσον αφορά τα μοντέλα προβλέψεων, είναι πως παρότι τα μοντέλα Μηχανικής Μάθησης Δάσους Αποφάσεων, Τυχαίου Δάσους και LightGBM παρουσιάζουν μικρές διαφοροποιήσεις στις εκτιμήσεις που κάνουν, με τον ταξινομητή του Τυχαίου Δάσους να διαπρέπει συγκριτικά σε αρκετές προσομοιώσεις, οι τελικές εκτιμήσεις διαφέρουν απειροελάχιστα από την benchmark εκτίμηση που δεν χρησιμοποιεί Μηχανική Μάθηση.

Βλέπουμε ωστόσο, πως καλιμπράροντας περισσότερο τα μοντέλα και κάνοντας καλύτερη προεπεξεργασία στο ATP dataset που χρησιμοποιήθηκε, θα μπορούσε να επιτευχθεί ακόμα καλύτερη επίδοση στα μοντέλα Μηχανικής Μάθησης, αλλά σε καμία περίπτωση και πάλι δεν θα διέφεραν σημαντικά από τις Benchmark προβλέψεις.

Μια πιθανή ερμηνεία αυτού του αποτελέσματος είναι πως αφενός οι αποδόσεις των bookmakers είναι αρκετά ενδεικτικές του αποτελέσματος του αγώνα, αφετέρου αφού τόσο τα μοντέλα όσο και η Benchmark μέθοδος χρησιμοποιούν τα ίδια δεδομένα με τον ίδιο σχεδόν τρόπο, θα καταλήξουν και σε παραπλήσια εκτίμηση. Συνεπώς, η βελτίωση των μοντέλων δεν θα επηρεάσει σημαντικά την επίδοση τους.

6.1.2 Συμπεράσματα ως προς τις Στοιχηματικές Στρατηγικές

Σε ότι αφορά τις στρατηγικές που χρησιμοποιήθηκαν, η αποτυχία επίτευξης κέρδους στο σταθερό Στοίχημα στο Αουτσάιντερ ήταν αναμενόμενη. Παρότι το Αουτσάιντερ δίνει σταθερά υψηλή απόδοση, η συχνότητα με την οποία νικάει είναι τόσο μικρή που δεν δίνει ρεαλιστικά περιθώρια νίκης σε κάποιον Παίκτη Στοιχήματος.

Αναμενόμενη ήταν και η αποτυχία στο Απλό Στοίχημα σταθερά στα Φαβορί και κατά συνέπεια και της Benchmark μεθόδου πρόβλεψης. Αφενός λόγω False Positive και False Negative κατηγοριοποιήσεων, αφετέρου λόγω του ότι το κέρδος στο Φαβορί είναι πολλές φορές μηδαμινό συγκριτικά με το ποντάρισμα. Εξ ‘ου και οι αρνητικές επιδόσεις των προσομοιώσεων ακόμα και με πρόβλεψη μέσω Μηχανικής Μάθησης.

Η μέθοδος Martingale με διπλασιασμό, όπως είδαμε, δεν κατάφερε να επιφέρει κάποιο κέρδος. Αυτό συνέβη διότι ο συντελεστής απόδοσης των Φαβορί ήταν πολύ μικρός και το ποσό πονταρίσματος δεν μπόρεσε ποτέ να γίνει αρκετά μεγάλο ώστε τα κέρδη να αποσβέσουν την ζημιά. Αντίθετα, η μέθοδος Martingale με τετραπλασιασμό παρουσιάζει μεγάλη επιτυχία σε όλες τις προσομοιώσεις, καθώς το επιθετικό ποντάρισμα καταφέρνει να υπερκεράσει τις ζημίες από τις ήττες των υπόλοιπων αγώνων, παρά τις χαμηλές αποδόσεις τους, με το εν λόγω Κέρδος ανά Συνολικό Κεφάλαιο Στοιχήματος να κυμαίνεται στα επίπεδα του 8%.

Στον αντίλογο, η στρατηγική Martingale απαιτεί ένα πολύ σημαντικό αρχικό κεφάλαιο, καθώς οι ζημιές στις προσομοιώσεις πριν την μέγιστη απόδοση των κερδών φτάνουν την τάξη

των δεκάδων χιλιάδων ευρώ. Παρότι η πιθανότητα για κέρδος είναι στατιστικά και θεωρητικά πάντα πιθανή και παρότι το τελικό ποσό κέρδους μπορεί να φτάσει ακόμα και τα 40.000€, η στρατηγική προϋποθέτει ότι ο Παίκτης θα έχει την οικονομική δυνατότητα να εκτεθεί σε αυτά τα μεγάλα ποσά επένδυσης και να συνεχίσει να επενδύει χωρίς να ξέρει πότε ακριβώς θα συμβεί η απόσβεση των επενδύσεων του. Στο πείραμα της παρούσας εργασίας αυτό συνέβη αρκετές φορές εντός της διάρκειας ενός έτους, αλλά αυτό δεν είναι κάτι το οποίο είναι εγγυημένο πως θα συμβαίνει κάθε έτος με την ίδια συχνότητα. Κοινώς, ο Παίκτης βρίσκεται πάντα στην δύσκολη θέση του να πρέπει να ποντάρει μεγάλα ποσά, το οποίο παράλληλα του δημιουργεί και μια δυσκολία στην ψυχολογία αφού διαχειρίζεται το ρίσκο.

Επίσης, αρκετές εταιρίες έχουν περιορισμούς στα μέγιστα ποσά πονταρίσματος, που μπορεί να στοιχηματίσει ένας Παίκτης, αποτρέποντας με αυτόν τον τρόπο πιθανές στρατηγικές παικτών, όπως για παράδειγμα την χρήση της στρατηγικής Martingale.

Συνοψίζοντας, θα μπορούσαμε να αποφανθούμε, πως η επιθετική μέθοδος Martingale είναι για τους παραπάνω λόγους προβληματική στρατηγική και δεν μπορεί να χαρακτηρισθεί πρακτικά αξιόπιστη.

6.2 Προεκτάσεις

Ερευνητές που θα ήθελαν να εντρυφήσουν περισσότερο στην **«Πρόβλεψη Αποτελεσμάτων ή/και Ανάπτυξη Στρατηγικής Στοιχήματος για Αγώνες Τένις»**, είναι πολύ πιθανόν να ενδιαφέρονται για τα παρακάτω:

- Ένα πεδίο έρευνας είναι η δημιουργία ή χρήση κατάλληλου Dataset. Το dataset που χρησιμοποιήθηκε στην παρούσα εργασία, εν τέλει βασίστηκε στις αποδόσεις, και άρα τις εκτιμήσεις των bookmakers. Θα είχε ενδιαφέρον να χρησιμοποιηθεί ένα dataset που να περιέχει περισσότερα features που αφορούν τον κάθε αγώνα τένις ή τον κάθε παίκτη.
- Τα μοντέλα που χρησιμοποιήθηκαν ήταν σχετικά απλές εφαρμογές αλγορίθμων της οικογένειας των Δένδρων Αποφάσεων. Ίσως πιο πολύπλοκοι αλγόριθμοι, όπως τα **Νευρωνικά Δίκτυα (Neural Networks),** να είχαν πιο σύνθετη ανάλυση και αποτέλεσμα. Επίσης, ενδιαφέρον θα είχε να μελετηθεί η επίδοση των **Νευρωνικών Γλωσσικών Μοντέλων (Neural Language Models)** σε αυτού του είδους τα προβλήματα.
- Η παρούσα εργασία ασχολήθηκε με τις πιο γνωστές στρατηγικές στοιχήματος. Άλλες εργασίες θα μπορούσαν να διερευνήσουν διαφορετικές στρατηγικές στοιχηματισμού. Για παράδειγμα, θα μπορούσε να αναζητηθεί η βέλτιστη ρήτρα απόδοσης που να ισοσταθμίζει το ρίσκο με την απόδοση κέρδους ή ακόμα καλύτερα να ορίζεται αυτή δυναμικά ανάλογα των αγωνιζόμενων αθλητών.
- Εν τέλει, οι προβλέψεις από τα μοντέλα Μηχανικής Μάθησης αποδεικνύεται πως είναι συγκρίσιμες με τις αποδόσεις των Bookmakers, πιθανώς διότι και οι ίδιοι οι Bookmakers χρησιμοποιούν αντίστοιχες μεθόδους για τον προσδιορισμό τον αποδόσεων. Άρα, η τακτική της εκπαίδευσης μοντέλων για εκτίμηση νικητή ίσως να μην είναι τόσο προσοδοφόρα. Μια εναλλακτική προσέγγιση που απαντάται στην βιβλιογραφία είναι η

εκπαίδευση μοντέλων Μηχανικής Μάθησης για εκτίμηση, όχι του νικητή σε μια αναμέτρηση, αλλά του αγώνα στον οποίο οι Bookmakers έχουν εκτιμήσει λανθασμένα το φαβορί ή έχουν αποδώσει λανθασμένα μεγάλη απόδοση σε έναν αμφίρροπο αγώνα.

Μελλοντικές εργασίες θα μπορούσαν ενδεχομένως να μελετήσουν το πρόβλημα από αυτή την σκοπιά. Δηλαδή, αμφισβητώντας τα φαβορί των Bookmakers βάσει αποδόσεων.[30]

Βιβλιογραφία

[1] S.-J. K. Sang-Youn LEE, «Analysis of Google's success factors and direction,» *Korean Journal of Artificial Intelligence,* pp. 11-16, 05 December 2022.

[2] D. Cournapeau, «Scikit-Learn,» Google Summer of Code Project, [Ηλεκτρονικό]. Available: https://scikit-learn.org/. [Πρόσβαση 11 October 2022].

[3] G. B. Pérez Fernando, «Google Colab,» Google Project Jupyter, [Ηλεκτρονικό]. Available: https://colab.research.google.com/. [Πρόσβαση 11 October 2022].

[4] A. J. Goldbloom, «Kaggle,» Kaggle Inc., [Ηλεκτρονικό]. Available: https://www.kaggle.com/. [Πρόσβαση 11 10 2022].

[5] «Google Drive,» Google LLC, [Ηλεκτρονικό]. Available: https://www.google.com/drive/. [Πρόσβαση 11 10 2022].

[6] F. M. Behrouz Forouzan, Εισαγωγή στην Επιστήμη των Υπολογιστών, Κλειδάριθμος, 2008.

[7] M. Kirk, Thoughtful Machine Learning with Python – A Test Driven Approach, O'Reilly, 2017.

[8] Β. Α. Φώτιος Πετρόπουλος, Επιχειρησιακές Προβλέψεις, Συμμετρία, 2013.

[9] D. Williams, Probability with Martingales, 1991, Cambridge Mathematical Textbooks.

[10] R. Lanciani, «Gambling and Cheating in Ancient Rome,» *The North American Review,* pp. 97-105, July 1892.

[11] S. M. Gainsbury, «Gambling on the Olympics,» *International Gambling Studies,* pp. 1-4, April 2010.

[12] International Betting Integrity Association, H2 Gambling Capital, «An Optimum Betting Market: A Regulatory, Fiscal & Integrity Assesment,» 2020.

[13] «Bookmakers.gr,» [Ηλεκτρονικό]. Available: https://www.bookmakers.gr/4300/stoixima-tennis/. [Πρόσβαση 13 Οκτώβριος 2022].

[14] B. Newcombe, Tennis Rules: A Players Guide, Sterling Pub, 1997.

[15] J. Buchdahl, σε *Fixed Odds and Sports Betting*, High Stakes Publishing, 2003.

[16] F. E. Moya, «Statistical Methodology for Profitable Sports Gambling,» Anahuac University, 2012.

[17] Κ. Γεωργούλη, Τεχνητή Νοημοσύνη, Κάλλιπος, 2015.

[18] E. Alpaydin, Introduction to Machine Learning, The MIT Press, 2014.

[19] C. Bishop, Pattern Recognition and Machine Learning, Springer, 2006.

[20] «scikit-learn.org,» [Ηλεκτρονικό]. Available: https://scikit-learn.org/stable/modules/tree.html. [Πρόσβαση 08 11 2022].

[21] T. K. Ho, «Random Decision Forests,» *IEEE,* 1995.

[22] T. K. Ho, «A Data Complexity Analysis of Comparative,» *Springer-Verlang Pattern Analysis & Applications,* pp. 102-112, 2002.

[23] «https://corporatefinanceinstitute.com/,» [Ηλεκτρονικό]. Available: https://corporatefinanceinstitute.com/resources/data-science/random-forest/. [Πρόσβαση 09 11 2022].

[24] J. Friedman, «Greedy Function Approximation: A Gradient Boosting Machine,» *The Annals of Statistics,* τόμ. 29, αρ. 05, pp. 1189-1232, 2001.

[25] G. Ke, «LightGBM: A High Efficient Gradient Boosting Decision Tree,» *NIPS,* 2017.

[26] [Ηλεκτρονικό]. Available: https://www.kaggle.com/general/264327. [Πρόσβαση 09 11 2022].

[27] «https://www.yourdatateacher.com,» [Ηλεκτρονικό]. Available: https://www.yourdatateacher.com/2021/06/07/precision-recall-accuracy-how-to-choose/#:~:text=We%20use%20precision%20when%20we,us%20a%20great%20competitive%20advantage.. [Πρόσβαση 09 11 2022].

[28] [Ηλεκτρονικό]. Available: https://towardsdatascience.com/accuracy-recall-precision-f-score-specificity-which-to-optimize-on-867d3f11124. [Πρόσβαση 09 11 2022].

[29] «Kaggle: ATP and WTA Tennis Results and Betting Odds Data,» [Ηλεκτρονικό]. Available: https://www.kaggle.com/datasets/hakeem/atp-and-wta-tennis-data. [Πρόσβαση 13 11 2022].

[30] «www.vantage-ai.com,» [Ηλεκτρονικό]. Available: https://www.vantage-ai.com/en/blog/beating-the-bookies-with-machine-learning. [Πρόσβαση 25 11 2022].

Παράρτημα Α: Πηγαίος Κώδικας Πειραματικού Σκέλους σε Python για τις Στοιχηματικές Στρατηγικές

Α.1: Απλό Στοίχημα σε Όλους τους Αγώνες στην Καλύτερη Στοιχηματική, με Ένα Ευρώ ανά Στοίχημα, με ή χωρίς Ρήτρα Σιγουριάς

Ο αλγόριθμος για την προσομοίωση του Απλού Στοιχήματος σε Όλους τους Αγώνες, στην Καλύτερη Στοιχηματική, με ένα ευρώ ανά στοίχημα, με η χωρίς Ρήτρα Σιγουριάς:

- Απλό Στοίχημα, διότι ο Παίκτης στοιχηματίζει σε μια λίστα αγώνων με σταθερό ποσό πονταρίσματος
- Καλύτερη Στοιχηματική, διότι λαμβάνεται υπόψιν μόνο η μέγιστη απόδοση αγώνα που δίνεται από τους Bookmakers
- Ένα Ευρώ, λόγω του σταθερού στοιχηματικού ποσού
- Ρήτρα Σιγουριάς, ένα κατώφλι για την στατιστική εκτίμηση της πρόβλεψης που έλαβε χώρα από κάποιο μοντέλο.

Ο αλγόριθμος υλοποιήθηκε ως μια συνάρτηση η οποία δέχεται ως ορίσματα:

- Το αρχικό χρηματικό κεφάλαιο του παίκτη (startingMoney)
- Το χρηματικό ποσό ανά αγώνα (betPerGame)
- Το κατώφλι σιγουριάς από το οποίο και πάνω δεχόμαστε να παίξουμε τον αγώνα (betZeroClause)
- Το ποσοστό της γκανιότας που κρατά η πλατφόρμα επί των κερδών (betCorpTax)
- Τα features που δέχεται το μοντέλο για κάθε αγώνα (X_dataset)
- Τα πραγματικά αποτελέσματα νικητή για κάθε αγώνα (Y_dataset)
- Οι προβλέψεις που παράγει το μοντέλο ανά αγώνα (modelPredictions)
- Η υπολογισμένη πιθανότητα-σιγουριά που δίνει το μοντέλο στην κάθε πρόβλεψη του (modelProbas)

Η συνάρτηση επιστρέψει ως αποτελέσματα:

- Το τρέχον κεφάλαιο του παίκτη ανά αγώνα
- Το κέρδη του παίκτη ανά αγώνα
- Τις Ζημίες του παίκτη ανά αγώνα
- Τα Χρήματα που στοιχηματίστικαν ανά αγώνα

Για αυτή τη προσομοίωση χρησιμοποιήθηκε ο παρακάτω πηγαίος κώδικας:

```
#startingMoney = αρχικό κεφάλαιο
#betPerGame = χρήματα που ποντάρουμε ανα αγώνα
#betZeroClause = Ποσοστό σιγουριάς κάτω απο οποίο δεν τζογάρουμε στον
αγώνα
#betOneClause = Ποσοστό σιγουριάς κάτω απο οποίο τζογάρουμε X€ στον
αγώνα, ενώ πάνω απο το οποίο τζογάρουμε 2*X€
```

```
#betCorpTax = ποσοστό κέρδους που παρακρατάται απο την στοιχηματική
(0.02)
#X_dataset = dataframe των features των αγώνων προς προσομοίωση
#Y_dataset = dataframe των labels των αγώνων προς προσομοίωση
#modelPredictions = εκτιμήσεις αποτελεσμάτων των αγώνων απο το μοντέλο
σε αυτό το datset
#modelProbas = λίστα με ποσοστιέες σιγουριές εκτιμήσεων. benchmark =
none, model = (p1, p2)

def Bet_Sim_Best_Corp_High_Proba_wBet (startingMoney, betPerGame,
betZeroClause, betOneClause, betCorpTax, X_dataset, Y_dataset,
modelPredictions, modelProbas):
  #Επικεφαλίδες αποδόσεων των στοιχηματικών εταιριών για κάθε παίκτη
  betCorpList_P1 =
['B&W_P1','B365_P1','CB_P1','IW_P1','EX_P1','GB_P1','LB_P1','PS_P1','SB
_P1','SJ_P1','UB_P1']
  betCorpList_P2 =
['B&W_P2','B365_P2','CB_P2','IW_P2','EX_P2','GB_P2','LB_P2','PS_P2','SB
_P2','SJ_P2','UB_P2']
  #Μετατροπή του dataset των νικητών, σε λίστα
  realWinnersList = Y_dataset.values.tolist()

  #τα λεφτά που έχουμε (currentMoney) στην αρχή είναι το αρχικό
κεφάλαιο (starting money)
  moneyList = [] #Λίστα που αποθηκεύεται το τρέχον χρηματικό κεφάλαιο
  gainList = [] #Λιστα που αποθηκεύονται τα κέρδη
  lossList = [] #Λίστα που αποθηκεύονται οι ζημιές
  matchBetList = [] #Λίστα που αποθηκεύονται τα χρήματα που
πονταρίστηκαν
  currentMoney = startingMoney
  #Για κάθε αγώνα
  for index, row in X_dataset.iterrows():
    betValues = []
    rateOfReturn, betGain, betMod = 0, 0, 0
    #Παίζουμε τον νικητή που μας λέει το μοντέλο μας για τον αντίστοιχο
αγωνα
    expectedWinner = float(modelPredictions[index])
    #Μαθαίνουμε το αποτέλεσμα του αγώνα και τον νική του
    realWinner = float(realWinnersList[index][0])
    #Κοιτάζουμε την σιγουριά με την οποία μαντεύει το φαβορί το μοντέλο
    if (len(modelProbas)!=0):
      winnerProba = max(modelProbas[index])
      #αν το μοντέλο μας είναι όσο σίγουρο θέλουμε (αλλιώς default 0)
      if (winnerProba >= betZeroClause):
        betMod = 1 #τζογάρω στο ματς
        #if (winnerProba >= betOneClause):
```

```
53          #  betMod = 2 #τζογάρω τα διπλα
54      else:
55        winnerProba = 1
56        betMod = 1
57
58      matchBet = betPerGame*betMod #τα χρήματα που θα τζογάρω στον αγώνα
59  θα είναι
60      #είτε 0*(...) αν το μοντέλο δεν είναι πολύ σίγουρο
61      #είτε 1*(...) αν το μοντέλο είναι κάπως σίγουρο
62
63      #Αν μαντέψαμε σωστά
64      if (expectedWinner == realWinner and winnerProba >= betZeroClause):
65        #Διαλέγουμε την λίστα με τις αποδώσεις του παίκτη για αυτόν τον
66  αγώνα
67        if (realWinner == 1): betCorpList = betCorpList_P1
68        else: betCorpList = betCorpList_P2
69        #Αποθηκεύουμε τις τιμές τους στη λίστα betValues
70        for corp in betCorpList:
71          if (row[corp] !=0):
72            betValues.append(row[corp])
73        #Το rateOfReturn θα προκύπτει απο την απόδοση με την μέγιστη τιμή
74        if (len(betValues)==0):
75          rateOfReturn = 0
76        else:
77          rateOfReturn = max(betValues)
78        #Το καθαρό κέρδος θα είναι (1-Γκανιότα)*(αποδώσεις για κάθε
79  στοιχηματική)*(λεφτά που τζογάραμε)
80        betGain = (1-betCorpTax)*rateOfReturn*matchBet
81      else:
82        #Αν χάσαμε, δεν έχουμε κέρδος
83        betGain = 0
84      #
85      #Τα λεφτά που έχουμε (currentMoney) είναι τα (λεφτά που είχαμε) -
86  (λεφτα που τζογάραμε) + (λεφτά που κερδίσαμε)
87      currentMoney = currentMoney - matchBet + betGain
88      moneyList.append(currentMoney)
89
90      matchBetList.append(matchBet)
91      #Αν χάσαμε, αποθήκευσε τις ζημιές, αλλιώς, αν κερδίσαμε, αποθήκευσε
92  τα κέρδη
93      if (betGain ==0):
94        gainList.append(0)
95        lossList.append(matchBet)
96      else:
97        gainList.append(betGain-matchBet)
98        lossList.append(0)
```

```
 99      return moneyList, gainList, lossList, matchBetList
100      #Μετα από κάθε αγώνα αποθηκεύω τα κέρδη & ζημίες ώστε να εξαχθεί το
101   διάγραμμα κεφαλαίου-αγώνα
```

Α.2: Στοίχημα με μέθοδο Martingale σε Όλους τους Αγώνες με Διπλασιασμό Πονταρίσματος σε κάθε Ήττα

Κατά αντιστοιχία με την προηγούμενη προσομοίωση, έχει δομηθεί και η τρέχουσα υπό την μορφή συνάρτησης. Κατά την Martingale στρατηγική:

- Ορίζεται ένα αρχικό ποντάρισμα
- Κάθε επόμενος αγώνας μετά από μια ήττα παίζεται στο διπλάσιο ποσό
- Σε περίπτωση νίκης το ποντάρισμα επιστρέψει στην αρχική τιμή του

Η συνάρτηση που υλοποιήθηκε δέχεται ως ορίσματα:

- Το αρχικό χρηματικό κεφάλαιο του παίκτη (startingMoney)
- Το χρηματικό ποσό ανά αγώνα (betPerGame)
- Το κατώφλι σιγουριάς από το οποίο και πάνω δεχόμαστε να παίξουμε τον αγώνα (betZeroClause)
- Το ποσοστό της γκανιότας που κρατά η πλατφόρμα επί των κερδών (betCorpTax)
- Τα features που δέχεται το μοντέλο για κάθε αγώνα (X_dataset)
- Τα πραγματικά αποτελέσματα νικητή για κάθε αγώνα (Y_dataset)
- Οι προβλέψεις που παράγει το μοντέλο ανά αγώνα (modelPredictions)
- Η υπολογισμένη πιθανότητα-σιγουριά που δίνει το μοντέλο στην κάθε πρόβλεψη του (modelProbas)

Η συνάρτηση επιστρέφει ως αποτελέσματα:

- Το τρέχον κεφάλαιο του παίκτη ανά αγώνα
- Το κέρδη του παίκτη ανά αγώνα
- Τις Ζημίες του παίκτη ανά αγώνα
- Τα Χρήματα που στοιχηματίστικαν ανά αγώνα

```
#startingMoney = αρχικό κεφάλαιο
#betPerGame = χρήματα που ποντάρουμε ανα αγώνα
#minCorpClause = Ρήτρα στοιχηματικής απόδοσης κάτω απο οποίο δεν
τζογάρουμε στον αγώνα
#maxCorpClause = Ρήτρα στοιχηματικής απόδοσης πάνω απο οποίο τζογάρουμε
στον αγώνα
#betZeroClause = Ποσοστό σιγουριάς κάτω απο οποίο δεν τζογάρουμε στον
αγώνα
#betOneClause = Ποσοστό σιγουριάς κάτω απο οποίο τζογάρουμε X€ στον
αγώνα, ενώ πάνω απο το οποίο τζογάρουμε 2*X€
#betCorpTax = ποσοστό κέρδους που παρακρατάται απο την στοιχηματική
(0.02)
#X_dataset = dataframe των features των αγώνων προς προσομοίωση
#Y_dataset = dataframe των labels των αγώνων προς προσομοίωση
#modelPredictions = εκτιμήσεις αποτελεσμάτων των αγώνων απο το μοντέλο
σε αυτό το dataset
```

```
#modelProbas = λίστα με ποσοστιέες σιγουριές εκτιμήσεων. benchmark =
none, model = (p1, p2)

def Martingale_Sim (startingMoney, betPerGame, minCorpClause,
maxCorpClause, betZeroClause, betOneClause, betCorpTax, X_dataset,
Y_dataset, modelPredictions, modelProbas):
  #Επικεφαλίδες αποδώσεων των στοιχηματικών εταιριών για κάθε παίκτη
  betCorpList_P1 =
['B&W_P1','B365_P1','CB_P1','IW_P1','EX_P1','GB_P1','LB_P1','PS_P1','SB
_P1','SJ_P1','UB_P1']
  betCorpList_P2 =
['B&W_P2','B365_P2','CB_P2','IW_P2','EX_P2','GB_P2','LB_P2','PS_P2','SB
_P2','SJ_P2','UB_P2']
  #Μετατροπή του dataset των νικητών, σε λίστα
  realWinnersList = Y_dataset.values.tolist()

  #τα λεφτά που έχουμε (currentMoney) στην αρχή είναι το αρχικό
κεφάλαιο (starting money)
  moneyList = [] #Λίστα που αποθηκεύεται το τρέχον χρηματικό κεφάλαιο
  gainList = [] #Λιστα που αποθηκεύονται τα κέρδη
  lossList = [] #Λίστα που αποθηκεύονται οι ζημιές
  matchBetList = [] #Λίστα που αποθηκεύονται τα χρήματα που
πονταρίστηκαν
  currentMoney = startingMoney
  martingaleFactor = 1
  matchPlayed = False
  #Για κάθε αγώνα
  for index, row in X_dataset.iterrows():
   betValues = []
    rateOfReturn, betGain, betMod = 0, 0, 0
    #Παίζουμε τον νικητή που μας λέει το μοντέλο μας για τον αντίστοιχο
αγωνα
    expectedWinner = float(modelPredictions[index])
    #Μαθαίνουμε το αποτέλεσμα του αγώνα και τον νικητή του
    realWinner = float(realWinnersList[index][0])
    #Κοιτάζουμε την σιγουριά με την οποία μαντεύει το φαβορί το μοντέλο
    if (len(modelProbas)!=0):
      winnerProba = max(modelProbas[index])
      #αν το μοντέλο μας είναι όσο σίγουρο θέλουμε (αλλιως default 0)
      if (winnerProba >= betZeroClause):
        betMod = 1 #τζογάρω στο ματς
        #if (winnerProba >= betOneClause):
        #  betMod = 2 #τζογάρω τα διπλα
    else:
      winnerProba = 1
      betMod = 1
```

```
63
64      #-----MARTINGALE MONEY ARE INSERTED HERE
65      matchBet = betPerGame*betMod*martingaleFactor
66      #τα χρήματα που θα τζογάρω στον αγώνα θα είναι
67      #είτε 0*(...) αν το μοντέλο δεν είναι πολύ σίγουρο
68      #είτε 1*(...) αν το μοντέλο είναι κάπως σίγουρο
69      #ποντάρω κανονικά ή περισσότερα ανάλογα με τον συντελεστή
70  MARTINGALE
71      #Παίζουμε μόνο όταν τηρούνται οι ρήτρες που έχουμε βάλει
72
73      #Αν μαντέψαμε σωστά
74      if (expectedWinner == realWinner and winnerProba >= betZeroClause):
75        #Διαλέγουμε την λίστα με τις αποδώσεις του αντιστοιχου παίκτη για
76  αυτόν τον αγώνα
77        if (realWinner == 1): betCorpList = betCorpList_P1
78        else: betCorpList = betCorpList_P2
79        #Αποθηκεύουμε τις τιμές τους στη λίστα betValues
80        for corp in betCorpList:
81          if (row[corp] !=0):
82            betValues.append(row[corp])
83        #Το rateOfReturn θα προκύπτει απο την απόδοση με την μέγιστη τιμή
84  (αν δεν έχουμε καθόλου αποδώσεις, δεν παίζουμε)
85        if (len(betValues)==0):
86          rateOfReturn = 0
87          matchPlayed = False
88        else:
89          rateOfReturn = max(betValues)
90          #----MARTINGALE CLAUSE---------------
91          if (rateOfReturn >= minCorpClause and rateOfReturn <=
92  maxCorpClause):
93            #Το καθαρό κέρδος θα είναι (1-Γκανιότα)*(αποδώσεις για κάθε
94  στοιχηματική)*(λεφτά που τζογάραμε)
95            betGain = (1-betCorpTax)*rateOfReturn*matchBet
96            matchPlayed = True
97          else:
98            matchPlayed = False
99            betGain = 0
100     else:
101       #Αν χάσαμε (ή αν δεν παίξαμε), δεν έχουμε κέρδος
102       if(winnerProba >= betZeroClause): matchPlayed = True
103       else: matchPlayed = False
104       betGain = 0
105
106     #Τα λεφτά που έχουμε (currentMoney) είναι τα (λεφτά που είχαμε) -
107 (λεφτα που τζογάραμε) + (λεφτά που κερδίσαμε)
108     currentMoney = currentMoney - matchBet + betGain
```

```
109       moneyList.append(currentMoney)
110
111       matchBetList.append(matchBet)
112       #Αν χάσαμε, αποθήκευσε τις ζημιές, αλλιως, αν κερδίσαμε, αποθήκευσε
113   τα κέρδη.
114       if (betGain ==0):
115         gainList.append(0)
116         lossList.append(matchBet)
117         #----Προσάρμοσε αντίστοιχα τον συτνελεστή MARTINGALE--------
118         if (matchPlayed): martingaleFactor = matchBet*2 #martingale*2
119       else:
120         martingaleFactor = 1
121         gainList.append(betGain-matchBet)
122         lossList.append(0)
123     return moneyList, gainList, lossList, matchBetList
124     #Μετα απο καθε αγώνα αποθηκεύω τα λεφτά που έχω εκείνη τη στιγμή ωστε
125   να φτιάξω διάγραμμα ανα αγώνα
```

A.3: Στοίχημα με Επιθετική Martingale με Τετραπλασιασμό Πονταρίσματος σε Κάθε Ήττα, Ρήτρα Σιγουριάς και Ρήτρα Αποδόσεων

Δίνεται για πληρότητα και ο αλγόριθμος της επιθετικής Martingale στρατηγικής, παρότι δεν αλλάζει σημαντικά στην δομή του. Κατά την επιθετική Martingale στρατηγική:

- Ορίζεται ένα αρχικό ποντάρισμα
- Κάθε επόμενος αγώνας μετά από μια ήττα παίζεται στο τετραπλάσιο ποσό
- Σε περίπτωση νίκης το ποντάρισμα επιστρέψει στην αρχική τιμή του

Η συνάρτηση που υλοποιήθηκε δέχεται ως ορίσματα:

- Το αρχικό χρηματικό κεφάλαιο του παίκτη (startingMoney)
- Το χρηματικό ποσό ανά αγώνα (betPerGame)
- Το κατώφλι σιγουριάς από το οποίο και πάνω δεχόμαστε να παίξουμε τον αγώνα (betZeroClause)
- Το ποσοστό της γκανιότας που κρατά η πλατφόρμα επί των κερδών (betCorpTax)
- Τα features που δέχεται το μοντέλο για κάθε αγώνα (X_dataset)
- Τα πραγματικά αποτελέσματα νικητή για κάθε αγώνα (Y_dataset)
- Οι προβλέψεις που παράγει το μοντέλο ανά αγώνα (modelPredictions)
- Η υπολογισμένη πιθανότητα-σιγουριά που δίνει το μοντέλο στην κάθε πρόβλεψη του (modelProbas)

Η συνάρτηση επιστρέφει ως αποτελέσματα:

- Το τρέχον κεφάλαιο του παίκτη ανά αγώνα
- Το κέρδη του παίκτη ανά αγώνα
- Τις Ζημίες του παίκτη ανά αγώνα
- Τα Χρήματα που στοιχηματίστικαν ανά αγώνα

```
#startingMoney = αρχικό κεφάλαιο
#betPerGame = χρήματα που ποντάρουμε ανα αγώνα
#minCorpClause = Ρήτρα στοιχηματικής απόδοσης κάτω απο οποίο δεν
τζογάρουμε στον αγώνα
#maxCorpClause = Ρήτρα στοιχηματικής απόδοσης πάνω απο οποίο τζογάρουμε
στον αγώνα
#betZeroClause = Ποσοστό σιγουριάς κάτω απο οποίο δεν τζογάρουμε στον
αγώνα
#betOneClause = Ποσοστό σιγουριάς κάτω απο οποίο τζογάρουμε X€ στον
αγώνα, ενώ πάνω απο το οποίο τζογάρουμε 2*X€
#betCorpTax = ποσοστό κέρδους που παρακρατάται απο την στοιχηματική
(0.02)
#X_dataset = dataframe των features των αγώνων προς προσομοίωση
#Y_dataset = dataframe των labels των αγώνων προς προσομοίωση
#modelPredictions = εκτιμήσεις αποτελεσμάτων των αγώνων απο το μοντέλο
σε αυτό το datset
```

```
#modelProbas = λίστα με ποσοστιέες σιγουριές εκτιμήσεων. benchmark =
none, model = (p1, p2)

def Martingale_Sim (startingMoney, betPerGame, minCorpClause,
maxCorpClause, betZeroClause, betOneClause, betCorpTax, X_dataset,
Y_dataset, modelPredictions, modelProbas):
  #Επικεφαλίδες αποδώσεων των στοιχηματικών εταιριών για κάθε παίκτη
  betCorpList_P1 =
['B&W_P1','B365_P1','CB_P1','IW_P1','EX_P1','GB_P1','LB_P1','PS_P1','SB
_P1','SJ_P1','UB_P1']
  betCorpList_P2 =
['B&W_P2','B365_P2','CB_P2','IW_P2','EX_P2','GB_P2','LB_P2','PS_P2','SB
_P2','SJ_P2','UB_P2']
  #Μετατροπή του dataset των νικητών, σε λίστα
  realWinnersList = Y_dataset.values.tolist()

  #τα λεφτά που έχουμε (currentMoney) στην αρχή είναι το αρχικό
κεφάλαιο (starting money)
  moneyList = [] #Λίστα που αποθηκεύεται το τρέχον χρηματικό κεφάλαιο
  gainList = [] #Λιστα που αποθηκεύονται τα κέρδη
  lossList = [] #Λίστα που αποθηκεύονται οι ζημιές
  matchBetList = [] #Λίστα που αποθηκεύονται τα χρήματα που
πονταρίστηκαν
  currentMoney = startingMoney
  martingaleFactor = 1
  matchPlayed = False
  #Για κάθε αγώνα
  for index, row in X_dataset.iterrows():
    betValues = []
    rateOfReturn, betGain, betMod = 0, 0, 0
    #Παίζουμε τον νικητή που μας λέει το μοντέλο μας για τον αντίστοιχο
αγωνα
    expectedWinner = float(modelPredictions[index])
    #Μαθαίνουμε το αποτέλεσμα του αγώνα και τον νικητή του
    realWinner = float(realWinnersList[index][0])
    #Κοιτάζουμε την σιγουριά με την οποία μαντεύει το φαβορί το μοντέλο
    if (len(modelProbas)!=0):
      winnerProba = max(modelProbas[index])
      #αν το μοντέλο μας είναι όσο σίγουρο θέλουμε (αλλιως default 0)
      if (winnerProba >= betZeroClause):
        betMod = 1 #τζογάρω στο ματς
        #if (winnerProba >= betOneClause):
        #  betMod = 2 #τζογάρω τα διπλα
    else:
      winnerProba = 1
      betMod = 1
```

```
63
64      #-----MARTINGALE MONEY ARE INSERTED HERE
65      matchBet = betPerGame*betMod*martingaleFactor
66      #τα χρήματα που θα τζογάρω στον αγώνα θα είναι
67      #είτε 0*(...) αν το μοντέλο δεν είναι πολύ σίγουρο
68      #είτε 1*(...) αν το μοντέλο είναι κάπως σίγουρο
69      #ποντάρω κανονικά ή περισσότερα ανάλογα με τον συντελεστή
70  MARTINGALE
71      #Παίζουμε μόνο όταν τηρούνται οι ρήτρες που έχουμε βάλει
72
73      #Αν μαντέψαμε σωστά
74      if (expectedWinner == realWinner and winnerProba >= betZeroClause):
75        #Διαλέγουμε την λίστα με τις αποδώσεις του αντιστοιχου παίκτη για
76  αυτόν τον αγώνα
77        if (realWinner == 1): betCorpList = betCorpList_P1
78        else: betCorpList = betCorpList_P2
79        #Αποθηκεύουμε τις τιμές τους στη λίστα betValues
80        for corp in betCorpList:
81          if (row[corp] !=0):
82            betValues.append(row[corp])
83        #Το rateOfReturn θα προκύπτει απο την απόδοση με την μέγιστη τιμή
84  (αν δεν έχουμε καθόλου αποδώσεις, δεν παίζουμε)
85        if (len(betValues)==0):
86          rateOfReturn = 0
87          matchPlayed = False
88        else:
89          rateOfReturn = max(betValues)
90          #----MARTINGALE CLAUSE---------------
91          if (rateOfReturn >= minCorpClause and rateOfReturn <=
92  maxCorpClause):
93            #Το καθαρό κέρδος θα είναι (1-Γκανιότα)*(αποδώσεις για κάθε
94  στοιχηματική)*(λεφτά που τζογάραμε)
95            betGain = (1-betCorpTax)*rateOfReturn*matchBet
96            matchPlayed = True
97          else:
98            matchPlayed = False
99            betGain = 0
100     else:
101       #Αν χάσαμε (ή αν δεν παίξαμε), δεν έχουμε κέρδος
102       if(winnerProba >= betZeroClause): matchPlayed = True
103       else: matchPlayed = False
104       betGain = 0
105
106     #Τα λεφτά που έχουμε (currentMoney) είναι τα (λεφτά που είχαμε) -
107 (λεφτα που τζογάραμε) + (λεφτά που κερδίσαμε)
108     currentMoney = currentMoney - matchBet + betGain
```

```
109       moneyList.append(currentMoney)
110
111       matchBetList.append(matchBet)
112       #Αν χάσαμε, αποθήκευσε τις ζημιές, αλλιως, αν κερδίσαμε, αποθήκευσε
113   τα κέρδη.
114       if (betGain ==0):
115         gainList.append(0)
116         lossList.append(matchBet)
117         #----Προσάρμοσε αντίστοιχα τον συντελεστή MARTINGALE--------
118         if (matchPlayed): martingaleFactor = matchBet*4 #ήταν matchBet*2
119       else:
120         martingaleFactor = 1
121         gainList.append(betGain-matchBet)
122         lossList.append(0)
123     return moneyList, gainList, lossList, matchBetList
```