Studium Licencjackie

Kierunek **Metody ilościowe w ekonomii i systemy informacyjne**

Imię i nazwisko autora **Rafał Chocianowski**

Nr albumu **109420**

Prognozowanie wyników meczów NBA za pomocą algorytmów uczenia maszynowego

Praca licencjacka

Napisana w Instytucie

Ekonometrii

Pod kierunkiem naukowym

prof.dr.hab. Michał Rubaszek SGH

Warszawa 2023

Spis treści

Wstęp

Sztuczna inteligencja to stosunkowo nowy, lecz coraz bardziej istotny temat w kontekście jej powszechnego zastosowania w licznych dziedzinach życia codziennego, w tym w sporcie. Celem niniejszej pracy jest przedstawienie zastosowania algorytmów uczenia maszynowego w kontekście prognozowania wyników meczów ligi National Basketball Association (NBA). Amerykańska liga od lat przyciąga uwagę miłośników koszykówki, a w ostatnich latach także fanów analizy danych, którzy starają się przewidywać wyniki rozgrywek na podstawie swojej wiedzy oraz doświadczenia w tym zakresie. Obecnie, dzięki szerszemu dostępowi do rozległych baz danych, możemy z coraz większą dokładnością przewidywać m.in. kto będzie laureatem nagród indywidualnych, takich jak ROTY[1]. Z drugiej strony ogromna ilość informacji prowadzi do powstania problemu z wyborem odpowiednich zmiennych potrzebnych do przeprowadzenia analiz. Rozpowszechniony szum informacyjny sprawia, że przeprowadzenie wnioskowania wymaga coraz większej wiedzy oraz mocy obliczeniowej komputerów. NBA jest największym związkiem koszykarskim na świecie. W sezonie 2021/2022 zanotował on zysk w wysokości ponad 10 miliardów dolarów, co jest wartością ponad dwa razy większą od zysku z sezonu 2014/2015[2]. Świadczy to o dynamicznym rozwoju ligi. Za tym idzie wzrost wydatków na analizę danych: Merrimack College wskazuje że, w roku 2021 prawie każda drużyna posiada własny dział analityki[3].Wraz z rozwojem technologii i coraz większą ilością danych, zespoły koszykarskie zaczęły wykorzystywać algorytmy uczenia maszynowego w celu uzyskania przewagi nad swoimi rywalami.

W niniejszej pracy celem jest prognoza wyniku meczu. W odróżnieniu od piłki nożnej, w meczach koszykówki nie ma możliwości zakończenia spotkania remisem, co ułatwi nam analizę i pozwoli skorzystać z modeli wykorzystujących zmienne binarne. Zadaniem modelu będzie prognozowanie czy drużyna grająca we własnej hali wygra mecz albo patrząc z drugiej strony, czy drużyna przyjezdna poniesie porażkę.

W pierwszym rozdziale przedstawię charakterystykę danych pochodzących z NBA oraz dokładnie przyjrzę się organizacji sezonu zasadniczego. Następnie scharakteryzuje wykorzystane przeze mnie dane dotyczące ostatnich sezonów oraz przekształcenia jakich

[1] Nagroda przyznawana najlepszemu zawodnikowi, który rozgrywa swój pierwszy sezon w NBA
[2] https://www.statista.com/statistics/193467/total-league-revenue-of-the-nba-since-2005/
[3] https://online.merrimack.edu/nba-analytics-changing-basketball/

dokonałem w celu wykonania dokładniejszych prognoz. Dodatkowo, na podstawie analiz, spróbuję znaleźć zależności i wyliczyć statystyki, które pomogą w późniejszej budowie modelu i ocenie jego jakości.

Drugi rozdział dotyczyć będzie zastosowanych metod badawczych i budowy modelu. Opiszę działanie modelu, czyli algorytm regresji logistycznej oraz lasu losowego. Następnie przedstawię warunki podziału na dane treningowe oraz testowe, a także wszystkie usprawnienia, które wpłynęły pozytywnie na jakość modelu. Zbadam także kryteria oceny jakości modelu krzywą ROC, pole AUC oraz macierz błędu.

W rozdziale trzecim porównam wyniki modelów między sobą oraz ocenie jakość ich predykcji w porównaniu do modeli dostępnych publicznie. Następnie zaprezentuje jakich zmian można dokonać, aby jakość modelu oraz dokładność predykcji była wyższa.

Rozdział 1. Charakterystyki ligi NBA

Dla prawdziwego fana koszykówki gracze z National Basketball Association budzą podziw i uznanie. Ich ponadprzeciętne warunki atletyczne oraz umiejętności sprawiają, że ich występy oglądają miliony osób na całym świecie. Według danych z 2022 roku każde spotkanie sezonu zasadniczego w USA ogląda przed telewizorami przeciętnie 1,6 mln widzów. Do tego trzeba doliczyć osoby, które korzystają z innych źródeł w Internecie, które nie są mierzone[4]. Zaczynając swoją przygodę z koszykówką wielu z nas marzyło o karierze „za oceanem". Koszykówką interesuję się już od ponad 10 lat i przez długi okres mojego życia sam trenowałem ten sport profesjonalnie w klubie. Dzięki temu cennemu doświadczeniu wydaję mi się, że posiadam intuicję oraz wiedzę, która pomoże mi przy dokładnej analizie czynników wpływających na wygraną. Poniżej przedstawię, dlaczego wybór padł akurat na ligę zza oceanu. Pierwszym czynnikiem wpływającym na moją decyzję jest to, że NBA jest ligą o bardzo zróżnicowanej strukturze drużynowej, w skład ligi wchodzi 29 zespołów ze Stanów Zjednoczonych oraz 1 z Kanady. Sezon zasadniczy składa się z 82 meczów dla każdej drużyny dzięki czemu mamy dużą liczbę obserwacji, gdyż łączna liczba spotkań podczas jednego sezonu regularnego wynosi 1230. Kolejnym powodem wyboru amerykańskiej ligi jest to, że zespoły składają się z zawodników z różnych krajów i kultur, a każdy z nich ma swoje unikalne umiejętności i statystyki. Każdego roku w czerwcu odbywa się Draft, czyli proces naboru nowych zawodników z uczelni wyższych albo prosto z liceum. Do NBA trafia co sezon 60 zawodników, spośród których niektórzy w przeciągu 2-3 lat odmieniają sytuację drużyny w tabeli[5]. Wszystko to sprawia, że analiza danych w NBA jest bardzo wymagająca, ale jednocześnie fascynująca. Następnym powodem wyboru ligi „zza oceanu" jest fakt, że jest to jedna z najbardziej zaawansowanych koszykarskich lig sportowych pod względem technologii i analizy danych. NBA zawsze było w czołówce w wykorzystywaniu innowacyjnych technologii do analizy gry i poprawy wyników. Jednym z najistotniejszych powodów wyboru NBA do analizy danych jest ich dostępność. Oficjalna strona NBA udostępnia ogromną ilość danych statystycznych, takich jak liczba punktów, skuteczność rzutów, liczba bloków oraz niezliczoną ilość o wiele bardziej skomplikowanych statystyk m.in. liczbę ataków na kosz danej drużyny czy skuteczność podań. Te dane są zbierane podczas każdego meczu i są następnie

[4] Mowa o serwisach, które nielegalnie streamują mecze
[5] Chodzi o ranking drużyn - pierwsze 8 z nich dostaje się do fazy pucharowej, czyli Playoffów

udostępniane do publicznego użytku. Ponadto, wiele firm i organizacji prywatnych gromadzi dane dotyczące poszczególnych drużyn i graczy spoza NBA. Istotnymi informacjami z punktu widzenia skautów może być ich wzrost, waga, celność rzutów w ostatnich 2 minutach rozgrywek oraz liczba kontuzji. Ostatnim powodem wyboru NBA jest to, że doświadczenie związane z analizą danych ma wiele zastosowań poza samą ligą. Na przykład, algorytmy uczenia maszynowego stosowane w niniejszej pracy mogą znaleźć zastosowanie w przypadku analizy innych lig koszykarskich. Ponadto, stosowane w tej pracy algorytmy regresji logistycznej oraz lasu losowego mogą być wykorzystane prognozowanie wyników w innych dziedzinach sportu.

Do analizy wykorzystam dane z sezonów 2018-2019 oraz 2019-2020. Wybór ten został podyktowany rozpoczęciem pandemii Covid-19, która spowodowała przesunięcie terminarza gier następnego sezonu oraz skrócenia przerwy między finałami i startem sezonu zasadniczego. Powyższe zmiany mogły negatywnie odbić się na statystykach poszczególnych graczy oraz drużyn w latach 2020-21 oraz późniejszych.

1.1. Jak analiza danych zmieniła NBA

Przetwarzanie danych od paru lat stało się jednym z głównych elementów rozgrywek NBA, gdzie niemal każda decyzja opiera się na analityce. Zespoły wykorzystują zaawansowaną technologię głównie na trzy sposoby:

- dobór odpowiedniej taktyki,
- przewidywanie i unikanie kontuzji zawodników,
- skauting.

Dobór odpowiedniej taktyki.

Rok 2009 był przełomowy dla analityków danych zatrudnionych w lidze, ponieważ zgodnie z wolą komisarza NBA Davida Sterna, na boisku każdej drużyny zostało zainstalowanych 6 kamer. System ten nazywa się *SportVu* i pozwala zbierać dane o ruchach zawodników oraz ich pozycji na boisku w czasie rzeczywistym z częstotliwością 25 klatek na

sekundę[6]. Dane te są analizowane przez modele uczenia maszynowego, aby zarekomendować odpowiednią taktykę. Przed wprowadzeniem tej zaawansowanej technologii, drużyny były w stanie zbierać tylko podstawowe statystyki dotyczące swoich graczy, takie jak zdobyte punkty, trafione rzuty osobiste czy faule. Natomiast obecnie dzięki zainstalowanemu systemowi wideo, analitycy drużyny mają dostęp do danych szczegółowych, takich jak częstotliwość, z jaką gracze atakują dwutaktem na prawą lub lewą stronę. Analiza nagrań pomogła trenerom zaplanować strategie defensywne, np. na podstawie zaawansowanych statystyk zadecydować, którego zawodnika oddelegować do obrony gwiazdy zespołu przeciwnego. Analiza usprawnia także ustawienia ofensywne. Trener może przekazać zespołowi w jakim dokładnie miejscu na boisku powinien stać każdy zawodnik, żeby wyegzekwować zagrywkę np. po wyjściu piłki za linię końcową.

Przewidywanie i unikanie kontuzji zawodników.

Jedną z najważniejszych kwestii dla każdego trenera oraz zespołu jest to, żeby zawodnicy nie odnosili kontuzji w trakcie trwania długiego sezonu zasadniczego. Aby zapobiec urazom, obecnie drużyny zbierają zaawansowane dane za pomocą urządzeń przenośnych. Na konferencji prasowej w 2021 roku, komisarz NBA Adam Silver przekazał, że gracze będą mieli obowiązek nosić specjalne urządzenia pomiarowe nie tylko podczas gier, ale także podczas treningów, aby zmierzyć wydajność i zmęczenie[7]. Interesująca rutyną jest pobieranie próbek śliny od każdego zawodnika, ponieważ zawiera ona wskaźniki zmęczenia. Rewolucyjnym zastosowaniem takiej analityki danych jest "odpoczywanie" zawodników nawet bez "oczywistych" uzasadnień w celu uniknięcia kontuzji. Taka praktyka nazywa się w *NBA load management*[8] i jest ona szeroko krytykowana przez fanów koszykówki. Idea jest taka, że im bardziej zmęczony jest gracz, tym bardziej jest podatny na kontuzje. Na przykład badania wykazały, że prawdopodobieństwo doznania kontuzji jest mniejsze, jeśli zawodnicy odpoczywają przez 30 dni po rozegraniu 30 prostych spotkań[9]. Dlatego też właściciele klubów NBA wraz z personelem medycznym podejmują niekorzystne dla widowiska i fanów decyzje, aby dać odpocząć kluczowym zawodnikom w celu zoptymalizowania poziomu zmęczenia na przestrzeni długiego sezonu zasadniczego.

[6] https://randerson112358.medium.com/how-data-transformed-the-nba-1cbc8b24e130
[7] https://online.merrimack.edu/nba-analytics-changing-basketball/
[8] https://www.sportscasting.com/what-load-management-how-affect-nba-players/
[9] https://d3.harvard.edu/platform-digit/submission/how-data-analytics-is-revolutionizing-the-nba/

Skauting

Trenerzy polegają teraz na analityce przy dokonywaniu wyborów w Drafcie. Wybór w Drafcie jest z punktu widzenia każdej drużyny bardzo ważną decyzją. Szansa na wybór w pierwszej trójce loterii zdarza się często raz na kilka sezonów, ponadto co roku poziom zawodników trafiających do ligi jest różny. Zdecydowanie się na nieodpowiedniego gracza może sprawić, że zespół na lata wypadnie z walki o mistrzostwo. Obecnie analizowane są dane liczbowe jak i wideo z uczelni wyższych i szkół średnich. Skauci podróżują często też do innych krajów np. w Europie, aby przyjrzeć się grze koszykarzy z innych lig. Jako ciekawostkę można dodać, że w drużynie Denver Nuggets zatrudniony jest polski skaut - Rafał Juć, który przyczynił się do wyboru w drafcie w 2014 roku Nikoli Jokicia, późniejszego dwukrotnego *Most Valuable Player* (MVP) ligi[10]. Obecnie trenerzy mają dostęp do nagrań zawodników z uczelni, aby przeanalizować (np. skuteczność, z jaką dany gracz kończy akcje *catch and shoot* po lewej stronie parkietu[11]). Tak jak w baseballu analitycy starają się też przewidzieć, jakim typem gracza będzie przyszły zawodnik np. ocenić jego szansę na dostanie się do All-NBA Team[12].

Chociaż ciężko jest znaleźć bezpośrednią korelację pomiędzy stosowaniem analityki danych a wygraną, jasne jest, że drużyny zaczęły czerpać korzyści z jej stosowania. Największą zmianą w grze w ostatnich latach jest wzrost znaczenia rzutów trzypunktowych w wyniku przeprowadzenia prostej kalkulacji. Modele wykazały, że rzuty trzypunktowe trafiane są średnio ze skutecznością 35%, a co za tym idzie, przeciętnie przynoszą więcej punktów na akcje niż rzuty dwupunktowe. Kalkulacja polega na tym, aby osiągnąć podobną oczekiwaną liczbę punktów na akcje gracze musieliby trafiać z półdystansu oraz spod kosza ze skutecznością powyżej 52,5%, co w praktyce jest bardzo trudne na przestrzeni 82 gier w sezonie. W rezultacie, liczba rzutów trzypunktowych oddawanych przez drużynę wzrosła ze średnio 18 prób na mecz w sezonie 2010/2011 do 35,2 w sezonie 2021/2022, co stanowi wzrost o 96%[13].

Drużyny są także coraz skuteczniejsze w przygotowaniu obrony pod konkretnego przeciwnika. Na przykład analitycy z zespołu Toronto Raptors stworzyli program, który w czasie rzeczywistym pokazuje jakie pozycje powinien zająć każdy gracz w obronie, biorąc pod

[10] https://dziendobry.tvn.pl/styl-zycia/rafal-juc-polski-lowca-talentow-dla-nba-na-czym-polega-praca-skauta-da328352-5317220
[11] Akcja polega to na tym, że gracz łapie piłkę i bez wykonania kozła od razu oddaje rzut
[12] W skład tej drużyny wchodzą najlepsi zawodnicy danego sezonu regularnego
[13] https://www.basketball-reference.com/leagues/NBA_stats_per_game.html

uwagę tendencje statystyczne graczy drużyny przeciwnej. Natomiast podczas przerw na żądanie gracze z zespołu Portland Trail Blazers oglądają na tabletach powtórki akcji z naniesionymi poprawkami w ustawieniu[14]. NBA wykorzystuje również metody wnioskowania Bayesowskiego, aby ustalić o ile lepsza jest ogólna obrona drużyny, gdy dany zawodnik jest na boisku, a także aby ocenić ogólną jakość obrony drużyny. Te dane są przydatne do oceny, którzy zawodnicy zniechęcają do najbardziej efektywnych typów rzutów, np. trzypunktowych i wsadów.

Podsumowując, analiza danych spowodowała, że trenerzy posiadają informację z jakiego miejsca na boisku gracze mają najlepszą oraz najgorszą skuteczność, jaki procent akcji kończą lewym albo prawym dwutaktem, itd.. Tak dokładne zrozumienie słabszych i mocniejszych stron każdego zawodnika pozwala na lepsze przygotowanie taktyczne, co jest szczególnie widoczne podczas *Playoffów*. Prowadzi to także do tego, że *roastery*[15] są zawężone, lepiej dopasowane i bardziej efektywne na boisku. Innym aspektem wzrostu znaczenia analizy danych jest jej wpływ na rolę trenerów. W przyszłości może ona być sprowadzona do tworzenia dobrej atmosfery w szatni i motywowania zawodników, natomiast o stronę taktyczna będzie dbał zespół analityków.

1.2. Zbiór danych modelu - NBA

Zbiór danych został pobrany z oficjalnej strony NBA[16]. Analizę będę opracowywał na podstawie sezonów 2018/19 oraz 2019/2020. W wyniku rozprzestrzeniania się pandemii nie wszystkie mecze sezonu zostały rozegrane. Sezon regularny został zawieszony 11 marca 2020 roku i wznowiony 30 lipca 2020 roku. Powrócił on w zmodyfikowanym formacie, który obejmował tylko 22 drużyny grające w środowisku "bańki" w *ESPN Wide World of Sports Complex* w Orlando na Florydzie[17].Co prawda każda drużyna do 11 marca powinna rozegrać około 72 mecze, ale niektóre zespoły rozegrały mniej spotkań ze względu na korekty

[14] https://bleacherreport.com/articles/1934273-ipop-how-big-data-will-transform-coaching-in-the-nba
[15] Lista zawodników biorących udział w meczu
[16] https://www.nba.com/stats/teams/boxscores-advanced?Season=2019-20; https://www.nba.com/stats/teams/boxscores-advanced?Season=2018-19
[17] https://www.sportingnews.com/us/nba/news/nba-bubble-rules-teams-schedule-orlando/zhap66a9hcwq1khmcex3ggabo

terminarza. W tabeli 1.1 przedstawiona została kompletna liczba spotkań rozegrana przez każdą drużynę w sezonie 2019-2020.

Tabela 1.1 Łączna liczba spotkań rozegrana przez każda drużynę w sezonie 2019-2020.

Nazwa drużyny	liczba meczów
Detroit Pistons	66
Indiana Pacers	73
Miami Heat	73
Milwaukee Bucks	73
New York Knicks	66
Orlando Magic	73
Philadelphia 76ers	73
Toronto Raptors	72
Washington Wizards	72
Dallas Mavericks	75
Denver Nuggets	73
Golden State Warriors	65
Houston Rockets	72
LA Clippers	72
Los Angeles Lakers	71
Memphis Grizzlies	73
Minnesota Timberwolves	64
New Orleans Pelicans	72
Oklahoma City Thunder	72
Phoenix Suns	73
Portland Trail Blazers	74
Sacramento Kings	72
San Antonio Spurs	71
Utah Jazz	72
Atlanta Hawks	67
Boston Celtics	72
Brooklyn Nets	72
Charlotte Hornets	65
Chicago Bulls	65
Cleveland Cavaliers	65

W analizowanych danych nie ma żadnych braków oraz wartości zerowych. Do budowy mojego modelu wziąłem pod uwagę 26 zmiennych. Dane składają się dodatkowo ze zmiennych jakościowych: nazwy drużyny grająca u siebie i na wyjeździe, datę spotkania oraz sezon, w którym mecze się odbywały. Do analizy użyję spotkań pochodzących z sezonu zasadniczego, który ze względu na swoją charakterystykę jest bardziej nieprzewidywalny w porównaniu do *Playffów* oraz oferuje o wiele większą liczbę statystyk.

Kolejnym aspektem, o którym warto wspomnieć jest fakt, że liga NBA podzielona jest na dwie konferencje (Wschodnią i Zachodnią). 15 drużyn wchodzi w skład każdej konferencji, która następnie podzielona jest na trzy dywizje po 5 drużyn. W tabeli 1.2 przedstawiam obecny podział na dywizje w NBA.

Tabela 1.2 Podział na dywizje NBA.

Atlantic	Central	Southeast	Northwest	Southwest	Pacyfic
Toronto Raptors	Cleveland Cavaliers	Miami Heat	Oklahoma City Thunder	San Antonio Spurs	Golden State Warriors
Boston Celtics	Indiana Pacers	Atlanta Hawks	Portland Trail Blazers	Dallas Mavericks	Los Angeles Clippers
New York Knicks	Detroit Pistons	Charlotte Hornets	Utah Jazz	Memphis Grizzlies	Sacramento Kings
Brooklyn Nets	Chicago Bulls	Washington Wizards	Denver Nuggets	Houston Rockets	Phoenix Suns
Philadelphia 76ers	Milwaukee Bucks	Orlando Magic	Minnesota Timberwolves	New Orleans Pelicans	Los Angeles Lakers

Dywizje i konferencje są określone przez położenie geograficzne miast w Stanach Zjednoczonych. Zespoły które są położone niedaleko siebie grają w tej samej dywizji. Każda drużyna gra następująca ilość spotkań:

- Cztery mecze (2 domowe, 2 wyjazdowe) przeciwko każdej z pozostałych czterech drużyn (łącznie 16 spotkań)
- Cztery mecze (2 domowe, 2 wyjazdowe) z sześcioma z pozostałych 10 drużyn (łącznie 24 spotkania)
- Trzy mecze z pozostałymi czterema drużynami– 1 mecz u siebie i 2 na wyjeździe przeciwko 2 drużynom oraz 2 mecze u siebie i 1 na wyjeździe przeciw 2 drużynom (łącznie 12 spotkań)

- 2 mecze (1 domowy, 1 wyjazdowy) przeciwko każdej z 15 drużyn w drugiej konferencji (łącznie 30 spotkań).

Istnieje dodatkowo pięcioletnia rotacja, która określa które drużyny z konferencji *Out of Division* grały tylko 3 razy, w której szczegóły nie będę się zagłębiać na potrzeby niniejszej pracy.

Przed przedstawieniem zmiennych, które zostały uwzględnione w modelu dodatkowo wyjaśnię kilka statystyk użytych podczas analizy:

- *True Shooting Percentage* (TS%) – miara uwzględniająca ilość zdobytych punktów w odniesieniu do liczby rzutów i ich wartości punktowej.[18] Do jej obliczenia stosuję się następującą formułę:

PTS / (2 FGA + 0,88 FTA)[19]

- *Offensive rating*- liczba punktów zdobyta na drużynę na 100 posiadań piłki.
- *Defensive rating* – liczba punktów tracona na drużynę na 100 posiadań piłki przeciwnika.
- TOV – liczba strat
- STL – liczba przechwytów
- REB – liczba zbiórek
- BLK – liczba bloków

Przed dodaniem danych do modelu wszystkie zmienne, które nie zostały wyrażone w wartościach procentowych, zostały przeskalowane, aby zawierały się w przedziale od 0 do 1. Powyższa technika została zastosowana, w celu zwiększenia jakości prognoz.

Przed wybraniem zmiennych do budowy modelu przedstawię ogólną analizę, ukazującą które wskaźniki według mnie powinny mieć największy wpływ na rezultat meczu:

1. Ze względu na większy doping z strony lokalnych fanów oraz brak zmęczenia wynikającego z podróży do hali przeciwnika, drużyny grające u siebie mają istotną przewagę. Jest wiele badań, które wskazują na to, że drużyny na

[18] https://probasket.pl/nba-czym-sa-zaawansowane-statystyki-rzutowe-analiza-i-wyjasnienie/
[19] FTA – liczba rzutów wolnych, FGA – liczba rzutów w danym meczu (bez rzutów wolnych)

wyjeździe popełniają więcej strat, mają mniej punktów z kontrataku oraz popełniają więcej fauli.[20]

2. W trakcie trwania długiego sezonu forma drużyn może ulegać zmianie pod wpływem takich czynników jak zmęczenie, kontuzje, serie meczów wyjazdowych, dlatego zmienne wzięte do budowy modelu będą oznaczone dodatkową 10 np. HOME_REB_10, która oznacza, że są to średnie wyniki z ostatnich 10 rozegranych spotkań we własnej hali dla danej drużyny.
3. Do budowy modelu skorzystałem z danych zespołowych, ponieważ według mnie są to najbardziej miarodajne i łatwo dostępne statystyki. Analiza zawierająca wpływ pojedynczych zawodników jest bardziej skomplikowana. Co więcej często przed meczem do samego końca nie wiemy czy dana gwiazda wystąpi, dlatego ciężko byłoby zawrzeć takie zmienne w modelu.

Dokładny opis zmiennych użytych do budowy modeli przedstawię poniżej:

Team_Home: jest to zmienna wskazująca nazwę drużyny grającej we własnej hali.

Team_Away: jest to zmienna wskazująca nazwę drużyny grającej na wyjeździe.

Date: jest to zmienna wyrażająca datę spotkania. Przyjmuję wartości od 2018-10-16 do 2020-08-13.

Season: jest to zmienna wskazująca lata, na przestrzeni których rozgrywał się dany sezon NBA. Przyjmuję wartości 2018-19 albo 2019-20.

RESULT: jest to zmienna oznaczająca 0, jeżeli drużyna przyjezdna wygra oraz 1, jeżeli drużyna grająca we własnej hali wygra. Przyjmuje wartości binarne.

Home_Score: jest to zmienna wskazująca liczbę punktów zdobytą przez drużynę grająca we własnej hali.

Home_FG_PCT_10: jest to zmienna opisująca średnią skuteczność procentową rzutów z pola drużyny z ostatnich 10 meczów rozegranych we własnej hali. Zawiera się w przedziale od 0 do 1.

[20] https://bleacherreport.com/articles/1520496-how-important-is-home-court-advantage-in-the-nba

Home_FG3_PCT_10: jest to zmienna opisująca średnią skuteczność procentową rzutów za trzy punkty drużyny z ostatnich 10 meczów rozegranych we własnej hali. Zawiera się w przedziale od 0 do 1.

Home_FT_PCT_10: jest to zmienna opisująca średnią skuteczność procentową rzutów osobistych drużyny z ostatnich 10 meczów rozegranych we własnej hali. Zawiera się w przedziale od 0 do 1.

Home_REB_10: jest to zmienna opisująca średnią liczbę zbiórek zdobytych przez drużynę z ostatnich 10 meczów rozegranych we własnej. Zmienna została przeskalowana do wartości z przedziału od 0 do 1.

Home_AST_10: jest to zmienna opisująca średnią liczbę asyst zdobytych przez drużynę z ostatnich 10 meczów rozegranych we własnej hali. Zmienna została przeskalowana do wartości z przedziału od 0 do 1.

Home_TOV_10: jest to zmienna opisująca średnią liczbę popełnionych strat przez drużynę z ostatnich 10 meczów rozegranych we własnej hali. Zmienna została przeskalowana do wartości z przedziału od 0 do 1.

Home_STL_10: jest to zmienna opisująca średnią liczbę przechwytów piłki uzyskanych przez drużynę z ostatnich 10 meczów rozegranych we własnej hali. Zmienna została przeskalowana do wartości z przedziału od 0 do 1.

Home_BLK_10: jest to zmienna opisująca średnią liczbę bloków uzyskanych przez drużynę z ostatnich 10 meczów rozegranych we własnej hali. Zmienna została przeskalowana do wartości z przedziału od 0 do 1.

Home_OFF_RATING_10: jest to zmienna opisująca średni ofensywny rating drużyny z ostatnich 10 spotkań rozegranych we własnej hali. Zmienna ta wyraża liczba punktów zdobytych przez drużynę grająca u siebie na 100 posiadań piłki. Zmienna została przeskalowana do wartości z przedziału od 0 do 1.

Home_DEF_RATING_10: jest to zmienna opisująca średni defensywny rating drużyny z ostatnich 10 spotkań rozegranych we własnej hali. Zmienna ta wyraża liczba punktów straconych przez drużynę grająca u siebie na 100 posiadań piłki przeciwnika. Zmienna została przeskalowana do wartości z przedziału od 0 do 1.

Home_TS_PCT_10: jest to zmienna uwzględniająca liczbę zdobytych punktów w zależności od rodzaju rzutu i ich liczby. Oznacza ona średnią skuteczność drużyny z ostatnich 10 spotkań rozegranych we własnej hali. Zawiera się w przedziale od 0 do 1.

Away_Score: jest to zmienna wskazująca liczbę punktów zdobytą przez drużynę grająca na wyjeździe.

Away_FG_PCT_10: jest to zmienna opisująca średnią skuteczność procentową rzutów drużyny grającej z ostatnich 10 meczów rozegranych na wyjeździe. Zawiera się w przedziale od 0 do 1.

Away_FG3_PCT_10: jest to zmienna opisująca średnią skuteczność procentową rzutów za trzy punkty drużyny z ostatnich 10 meczów rozegranych na wyjeździe. Zawiera się w przedziale od 0 do 1.

Away_FT_PCT_10: jest to zmienna opisująca średnią skuteczność procentową rzutów osobistych drużyny z ostatnich 10 meczów rozegranych na wyjeździe. Zawiera się w przedziale od 0 do 1.

Away_REB_10: jest to zmienna opisująca średnią liczbę zbiórek zdobytych przez drużynę z ostatnich 10 meczów rozegranych na wyjeździe. Zmienna została przeskalowana do wartości z przedziału od 0 do 1.

Away_AST_10: jest to zmienna opisująca średnią liczbę asyst zdobytych przez z ostatnich 10 meczów rozegranych na wyjeździe. Zmienna została przeskalowana do wartości z przedziału od 0 do 1.

Away_TOV_10: jest to zmienna opisująca średnią liczbę popełnionych strat przez drużynę z ostatnich 10 meczów rozegranych na wyjeździe. Zmienna została przeskalowana do wartości z przedziału od 0 do 1.

Away_STL_10: jest to zmienna opisująca średnią liczbę przechwytów piłki uzyskanych przez drużynę z ostatnich 10 meczów rozegranych na wyjeździe. Zmienna została przeskalowana do wartości z przedziału od 0 do 1.

Away_BLK_10: jest to zmienna opisująca średnią liczbę bloków uzyskanych przez drużynę z ostatnich 10 meczów rozegranych na wyjeździe. Zmienna została przeskalowana do wartości z przedziału od 0 do 1.

Away_OFF_RATING_10: jest to zmienna opisująca średni ofensywny rating drużyny z ostatnich 10 spotkań rozegranych na wyjeździe. Zmienna ta wyraża liczba punktów zdobytych przez drużynę grająca na wyjeździe na 100 posiadań piłki. Zmienna została przeskalowana do wartości z przedziału od 0 do 1.

Away_DEF_RATING_10: jest to zmienna opisująca średni defensywny rating drużyny z ostatnich 10 spotkań rozegranych na wyjeździe. Zmienna ta wyraża liczba punktów straconych przez drużynę grająca na wyjedzie na 100 posiadań piłki przeciwnika. Zmienna została przeskalowana do wartości z przedziału od 0 do 1.

RESULT_10: jest to zmienna opisująca procentową liczbę wygranych drużyny we własnej hali na przestrzeni ostatnich 10 meczów. Zawiera się w przedziale od 0 do 1.

1.3. Analiza i wizualizacja danych

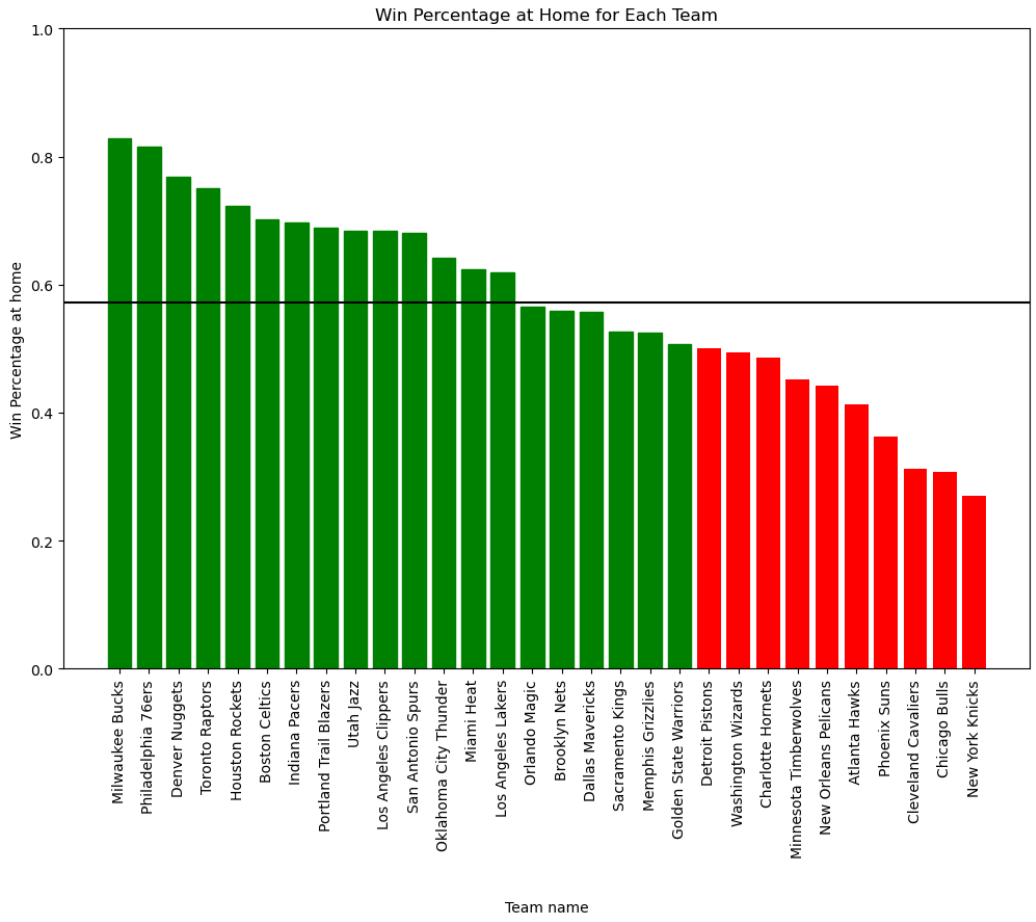
W poniższym podrozdziale postaram się przedstawić charakter zbioru danych oraz uzasadnienie, dlaczego skorzystałem z modelu regresji logistycznej oraz lasu losowego. Rozpoczynając budowę modelu wykorzystałem wszystkie zmienne dotyczące statystyk drużynowych, które zostały opisane w rozdziale „Charakterystyki ligi NBA".

Zmienna objaśniana

Analizę rozpocznę od zbadania zmiennej objaśnianej. W modelu zmienna ta jest nazwana „RESULT" i przyjmuje wartość 1, jeżeli gospodarz meczu wygra i 0, jeżeli przegra. Zgodnie z intuicją możemy stwierdzić, że drużyna mająca przewagę parkietu, czyli grająca u siebie wygrywa częściej. Oryginalny zbiór danych zawiera 2285 rekordów z czego 1310 zostało oznaczanych 1 co oznacza, że drużyny grające u siebie wygrywają w 57,33% przypadków podczas badanego okresu. Z własnych obserwacji mogę powiedzieć, że odsetek znacząco powyżej 50 procent nie powinien nas dziwić biorąc pod uwagę postawę kibiców wobec drużyny przyjezdnej oraz gwizdki sędziów, które często są bardziej przychylne dla gospodarzy. Kolejnym czynnikiem, który wpływa pozytywnie na liczbę wygranych we własnej hali jest

brak podróży pomiędzy miastami, co sprawia, że zawodnicy są bardziej wypoczęci. Na wykresie 1.1 przedstawię rozkład wygranych we własnej hali dla każdej drużyny.

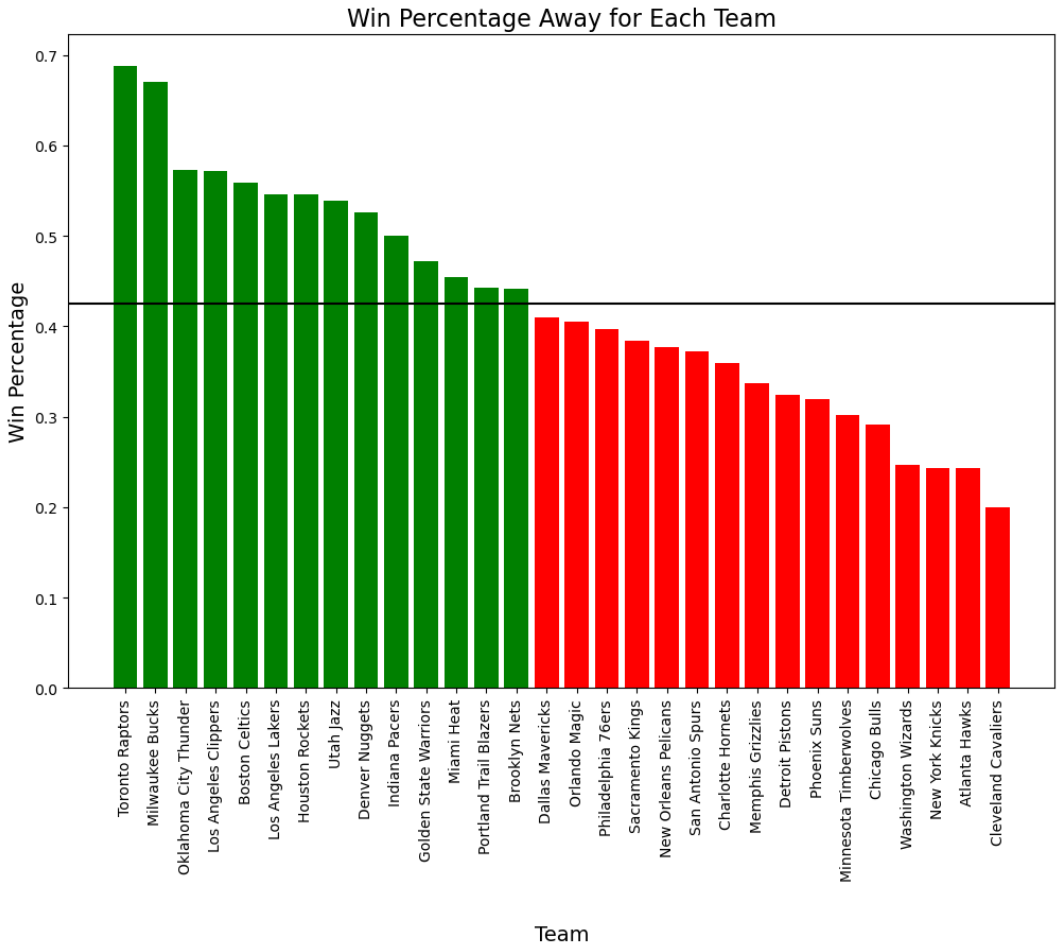
Wykres 1.1 Struktura wyników meczy rozegranych we własnej hali podczas sezonów 2018-19 / 2019-20.



Z wykresu wynika, że 20 na 30 zespołów wygrało ponad 50 procent meczy rozegranych u siebie - są to słupki drużyn oznaczone kolorem zielonym. Najwyższym odsetkiem zwycięstw mogła pochwalić się drużyna *Milwaukee Bucks*, która wygrała 82,9 procent spotkań rozgrywanych we własnej hali. Czarna pozioma linia wskazuję średnią liczbę wygranych w domu. Prawie połowa, bo 14 na 30 drużyn wygrało ponad 57,33 procent spotkań rozegranych we własnej hali. Najniższy odsetek zwycięstw zanotował zespół *New York Knicks* który wygrał zaledwie 27 procent gier u siebie podczas badanego okresu. Najlepszym przykładem tego jak liga jest nieprzewidywalna oraz zmienna może być to, że w obecnym sezonie 2022-23 wspomniany wyżej zespół dostał się do *Playoffów* z 5 miejsc w Konferencji Wschodniej notując w sezonie regularnym bilans na poziomie 47-35 (23-18 we własnej hali co przekłada

się na ponad 56 procent zwycięstw). Na wykresie 1.2 będziemy mogli zaobserwować, jak prezentuje się liczba wygranych na wyjeździe na przestrzeni całego badanego okresu.

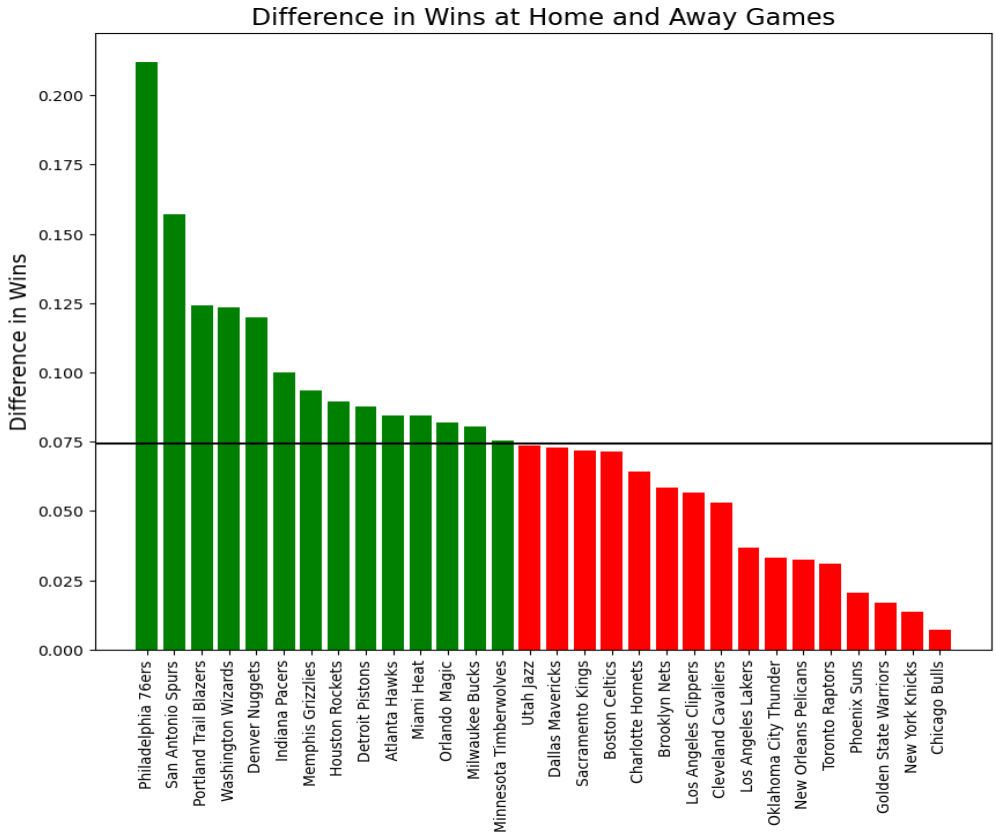
Wykres 1.2 Struktura wyników meczy na wyjeździe podczas sezonów 2018-19 oraz 2019-20.



Czternaście drużyn zanotowało odsetek wygranych na wyjeździe powyżej średniej wynoszącej 42,48%. Lider w tabeli drużyna *Milwaukee Bucks* zanotowała spadek o 15,8 punktów procentowych z poziomu 82,9 procent do 67,1 procent zwycięstw. Pierwszą drużyną jest *Toronto Raptors*, czyli zespół, który w sezonie 2018-2019 wygrał mistrzostwo NBA. Po najwyższe laury w roku 2019-2020 sięgnął zespół *Los Angeles Lakers*, który znajduje się dopiero na 6 miejscu pod względem liczby wygranych na wyjeździe w badanym okresie i aż 14 miejscu w rankingu liczby wygranych we własnej hali. Może to świadczyć o tym, że dyspozycja drużyny w sezonie regularnym nie zawsze przekłada się na późniejsze osiągnięcia w fazie pucharowej. Wskaźnik korelacji pomiędzy liczbą wygranych gier na wyjeździe, a liczbą gier wygranych u siebie jest wysoki i wynosi około 0,819 natomiast żeby dokładniej zbadać przewagę własnego parkietu, warto przyjrzeć się danym przedstawionym na kolejnym

wykresie. Na wykresie 1.3 ukazałem różnicę między liczbą wygranych w domu a na wyjeździe dla wszystkich drużyn.

Wykres 1.3 Różnica między liczba wygranych u siebie a na wyjeździe.



Analizując powyższy wykres możemy zauważyć, że 5 na 30 drużyn wygrywało średnio o 10 procent więcej spotkań we własnej hali niż na wyjeździe. Czarna pozioma linia reprezentująca średnią różnicę w badanej próbie znajduje się na poziomie 0,0743. Oznacza to, że drużyny średnio wygrywają więcej o 7,43 % gier w domu niż na wyjeździe. Nawet zespoły z samego dołu tabeli takie jak *Phoenix Suns* czy *New York Knicks*, mimo słabej dyspozycji miały wyższy odsetek zwycięstw w domu. Jedynie dla drużyny *Chicago Bulls* różnica jest prawie niezauważalna. Na pierwszym miejscu z wynikiem 21,2 procent plasuje się drużyna *Philadelphia 76ers.* Tak znacząca różnica wymagałaby dokładniejszego zbadania i przeanalizowania. Jako ciekawostkę mogę dodać, że hala drużyny *Denver Nuggets* położona jest na wysokości 1600 metrów nad poziomem morza[21] co sprawia, że ze względu na niższe

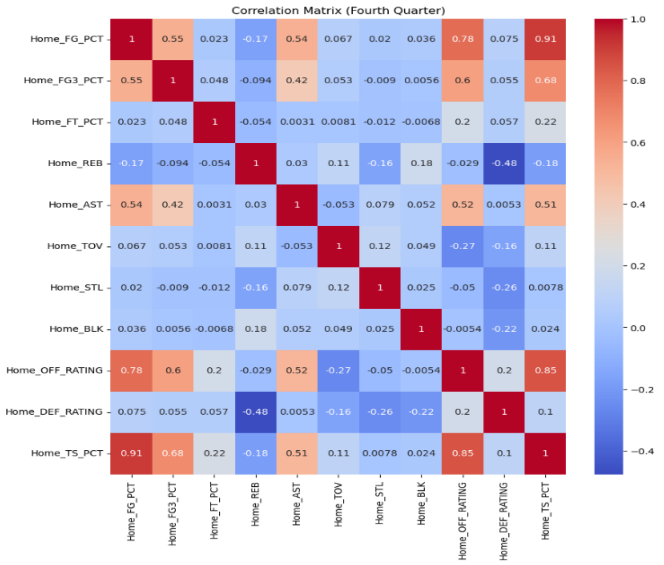
[21] https://mrbuzzer.pl/turysto-nie-zapomnij-zabrac-do-denver-maski-tlenowej/

stężenie tlenu w powietrzu gra dla drużyny przyjezdnej jest jeszcze cięższa, stąd wysokie 5 miejsce na wykresie.

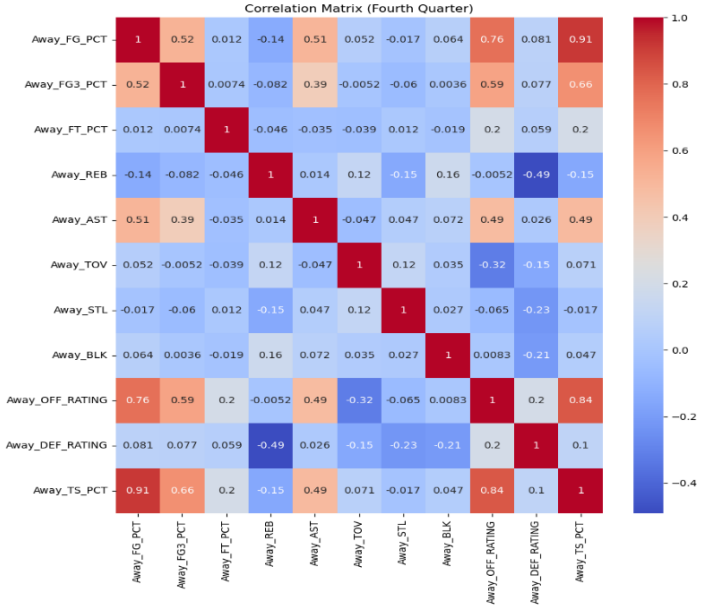
1.4. Korelacje

Następnym punktem analizy będzie zbadanie korelacji między poszczególnymi zmiennymi objaśniającymi. Na wykresie 1.4 oraz 1.5 przedstawia wyniki w postaci mapy korelacji.

Wykres 1.4 Korelacje pomiędzy zmiennymi dla meczów domowych.



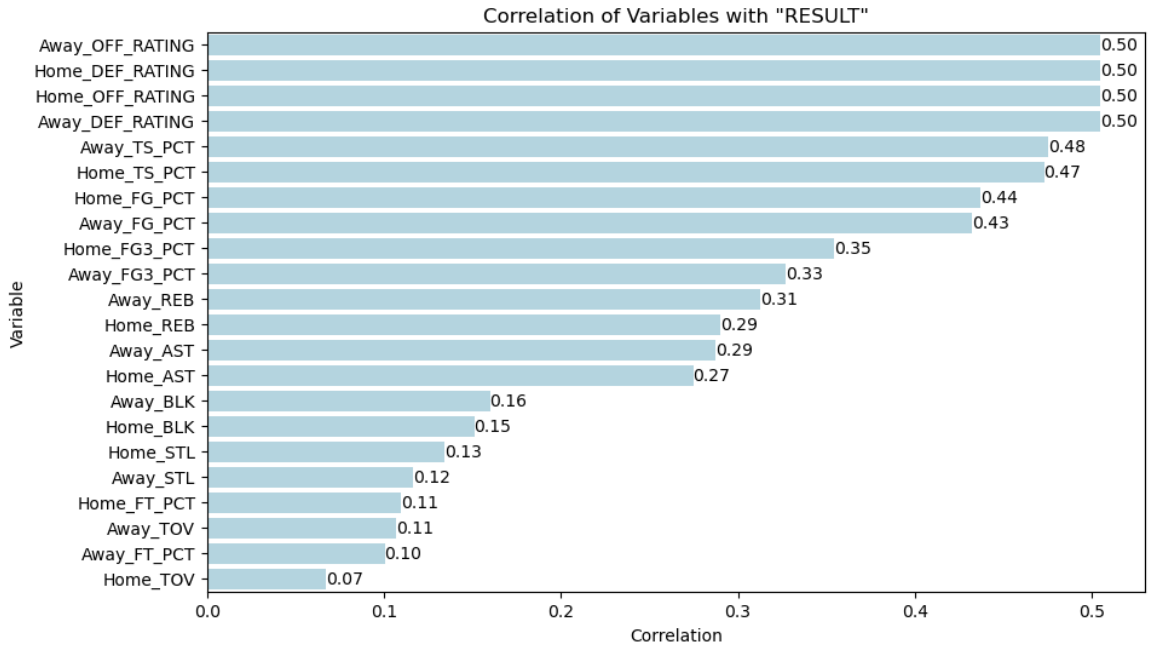
Wykres 1.5 Korelacje między zmiennymi dla meczów wyjazdowych.



Z wykresów 1.4 i 1.5 wynika, że najsilniej ze sobą skorelowanymi zmiennymi są Home_TS_PCT ze zmienną Home_FG_PCT oraz Away_TS_PCT ze zmienną Away_FG_PCT - ich współczynniki korelacji są identyczne i wynoszą 0,91. Tak wysoki wynik ma związek z tym, że do wyliczenia *True Shooting Percentage* wykorzystujemy skuteczność z pola, czyli *Field Goal*(FG). Na drugim miejscu z korelacją na poziomie 0,85 znajdują się zmienne Home_TS_PCT oraz Home_OFF_RATING identycznie jak w powyżej wymienionym wypadku do wyliczenia ratingu ofensywnego potrzebna jest nam ilość zdobytych punktów na 100 posiadań piłki, która jest tym wyższa, im wyższy jest wskaźnik oznaczający *True Shooting Percentage*. Kolejnymi zmienną z poziomem korelacji powyżej 0,5 są zmienne Home_AST ze zmienną Home_FG_PCT. Oznacza to, że wraz ze wzrostem skuteczności rzutów liczba asyst rośnie. Porównując ze sobą dwie tabelki możemy zaobserwować, że dla zmiennych z oznaczeniem „Home" względem „Away" korelacja jest często większa o około 0,02 punktu, co może być kolejnym dowodem świadczącym o sile przewagi własnego parkietu. Najniższą korelacją wynoszącą -0,48 charakteryzują się zmienne oznaczające zbiórki i defensywny rating. Brak dobrego zastawienia własnej tablicy powoduje, że drużyny pozwalają na zwiększoną ilość ofensywnych zbiórek co przekłada się na niższy rating defensywny, ponieważ po zbiórce zawodnik ma dodatkową szansę na zdobycie punktów. Kolejnymi zmiennymi, które mają ujemna korelacją są straty piłki i ranking ofensywny. W tym przypadku analogicznie więcej strat prowadzi do mniejszej ilości akcji co przekłada się na niższy ranking ofensywny. Interesującą zależność przedstawia korelacja wynosząca -0,22 pomiędzy zmienna Home_BLK- oznaczająca liczbę bloków, a zmienną Home_DEF_RATING. Tak nieintuicyjna relacja może wynikać z tego, że drużyny, które mają więcej bloków, pozwalają na więcej oddanych rzutów spod własnego kosza[22] co może negatywnie wpływać na ich rating defensywny. Na wykresie 1.6 ukażę które zmienne mają najwyższą korelacje ze zmienną objaśnianą „RESULT".

[22] Z analizy przeprowadzonej prze Justina Jacoba w 2017 roku wynika, że około 90-95 procent zablokowanych rzutów to rzuty 2-punktowe. Patrząc dokładnie na rozkład bloków każdej drużyny możemy zobaczyć, że najwięcej bloków są to rzuty spod samego kosza w promieniu 2 metrów od obręczy. https://squared2020.com/2017/12/20/making-blocks-count/

Wykres 1.6 korelacja między zmiennymi zależnymi a zmienną niezależną.



Większość wyników uzyskanych w powyższej tabeli jest zgodna z wcześniejszymi przypuszczeniami. Im wyższa jakość gry drużyny domowej jak i przyjezdnej, tym wyższe prawdopodobieństwo ich zwycięstwa. Zmiennymi, które mają największy wpływ na wynik gry to rating defensywny i ofensywny. Terminy ofensywny i defensywny rating odnoszą się do ilości punktów zdobytych przez drużynę i przez jej przeciwników, odpowiednio na 100 posiadań piłki. Normalizacja do 100 posiadań pozwala porównanie drużyn, ponieważ tempo gry może zniekształcić porównanie całkowitej liczby punktów na mecz. Oznacza to, że do wyliczenia tego wskaźnika nie jest istotna łączna liczba rzutów na kosz, która może być zasadniczo różna pomiędzy drużynami. Na kolejnych miejscach znajdują się charakterystyki opisujące skuteczność rzutową, tutaj zależność wydaje się prosta, im wyższa skuteczność tym większa szansa na wygraną. Znacząca rola zbiórek, mimo zwiększonego tempa gry w ostatnich latach, może zaskakiwać niektórych obserwatorów. W moim zbiorze danych nie dokonałem rozróżnienia między zbiórkami ofensywnymi a defensywnymi, natomiast z badań przeprowadzonych podczas sezonu 2019-20 wynika, że jedno posiadanie piłki generuje średnio 1,1 punktu a posiadanie z ofensywna zbiórką 1,19 punktu[23]. Wysoki współczynnik korelacji oznacza, że wyżej wymienione zmienne będą miały największy wpływ na przewidywanie wyników meczy. Najniższą korelacją względem zmiennej objaśnianej wykazują zmienne

[23] https://fansided.com/2020/01/28/nylon-calculus-nba-rebound-tracking/

dotyczące liczby przechwytów, strat oraz trafionych rzutów osobistych. Z klasyfikacji ze strony *Teamrankings* wynika, że w analizowanych przeze mnie sezonach odsetek punktów jaki został zdobyty na linii rzutów wolnych wynosił od 13,8 procent do 19,7 procent w zależności od drużyny[24]. Oznacza to, że nieznaczne różnice w skuteczności pomiędzy zespołami, nie mają istotnego wpływu na wygraną biorąc pod uwagę cały sezon, a nie pojedyncze mecze, gdzie każdy punkt może decydować o zwycięstwie.

[24] https://www.teamrankings.com/nba/stat/percent-of-points-from-free-throws?date=2020-10-12

Rozdział 2. Opis metody badawczej i wybór modelu

Przedstawione w niniejszej pracy modele mogą znaleźć zastosowanie w przewidywaniu wyników pojedynczego meczu sezonu regularnego NBA. W celu określenia wartości zmiennych objaśniających dla danej obserwacji potrzebna jest znajomość wyników 10 poprzednich kolejek rozegranych zarówno u siebie jak i na wyjeździe. Modele nie mogą zatem zostać wykorzystane do predykcji wyników wszystkich spotkań sezonu zasadniczego.

Przy użyciu regresji logistycznej oraz lasu losowego stworzę dwa modele, w których zawarte zostaną średnie statystyki drużynowe z ostatnich 10 meczy rozegranych we własnej hali jak i na wyjeździe wraz ze średnią zmiennej „RESULT", która będzie odpowiadać procentowi zwycięstw drużyny grającej u siebie w ostatnim okresie. W tym rozdziale przedstawię oszacowanie parametrów modelu oraz ocenię jakość modelu oraz jakość predykcji.

2.1. Selekcja zmiennych do budowy modelu

W celu selekcji zmiennych objaśniających na początku wybrałem 10 z nich, które wykazały się największą korelacją ze zmienną „RESULT". Po przeanalizowaniu jakości modelu m.in. krzywej ROC oraz macierzy błędu, model ten okazał się gorszy od modelu, w którym zostały wykorzystane wszystkie zmienne. Następnie korzystając z funkcji w *SequentialFutureSelector* w Pythonie określiłem odgórnie liczbę zmiennych jaką chcę zawrzeć w mojej analizie. Poniżej przedstawię, jak działa wyżej wymieniona funkcja:

1. W pierwszym kroku w zależności od tego czy wybierzemy opcje *forward* czy *backward* albo zaczniemy od pustego zbioru albo od umieszczenia wszystkich zmiennych, które będą potem stopniowo eliminowane z modelu.
2. Wybór liczby zmiennych modelu jest dopasowywany do danych treningowych z wykorzystaniem aktualnego zestawu cech.
3. W następnym etapie następuje ocena dobranych parametrów przy użyciu określonego kryterium oceny na danych walidowanych krzyżowo przy użyciu parametru cv.

4. W oparciu o parametr *forward=True,* algorytm wybiera najlepszą cechę do dodania. Odwrotnie jest w wypadku, gdy wybierzemy opcje *forward=False* wtedy algorytm eliminuje najmniej dopasowaną zmienną.
5. Algorytm pracuje do momentu, gdy wybierze wybraną przez nas odgórnie liczbę cech wskazaną w parametrze o nazwie *n_features_to_select.*

Korzystając z poniższej funkcjonalności zdecydowałem się na wybór od 5 do 20 zmiennych i analizowałem wynik krzywej ROC, miary *Accuracy* oraz macierzy błędu, a także oszacowania parametrów regresji logistycznej oraz ich istotność. Po przeanalizowaniu wyników najlepszym rozwiązaniem okazał się model zawierający 20 zmiennych wraz ze zmienną objaśnianą. Poniżej przedstawione zmienne zostały następnie użyte do budowy modeli:

1. *Home_FG_PCT_10,*
2. *Home_FG3_PCT_10,*
3. *Home_FT_PCT_10,*
4. *Home_REB_10,*
5. *Home_AST_10,*
6. *Home_TOV_10,*
7. *Home_STL_10,*
8. *Home_BLK_10,*
9. *Home_TS_PCT_10,*
10. *RESULT_ 10,*
11. *Away_FG_PCT_10,*
12. *Away_FG3_PCT_10,*
13. *Away_FT_PCT_10,*
14. *Away_REB_10,*
15. *Away_AST_10,*
16. *Away_TOV_10,*
17. *Away_STL_10,*
18. *Away_BLK_10,*
19. *Away_OFF_RATING_10,*
20. *RESULT.*

2.2. Trening modelu

Następnym etapem będzie podzielenie modelu na zbiór testowy oraz treningowy. Ze względu na to, że w modelach znajdują się zmienne wyliczające średnią z ostatnich 10 spotkań rozegranych we własnej hali oraz na wyjeździe trzeba było usunąć 600 meczy(po 300 meczów ze startu każdego analizowanego sezonu) ze względu na braki danych. Z tego powodu łączna liczba obserwacji spadła z poziomu 2285 do 1685. Do fragmentacji mojego zbioru użyłem funkcji *train_test_split* z biblioteki *skcikt-learn* w Pythonie.

Poniżej przedstawię, jak działa ta funkcja:

1. W pierwszym kroku dzielimy zbiór zmiennych na dwie grupy - w pierwszej będziemy mieli nasze zmienne zależne natomiast w drugim zmienną objaśnianą.
2. W drugim kroku, przy użyciu parametru *test_size*, określamy podział próby na zbiór testowy i treningowy. W mojej analizie wybrałem parametr 0,2 co oznacza, że 20% jest zbiorem testowym na którym potem będziemy sprawdzać skuteczność predykcji, a 80 procent obserwacji służy do treningu.
3. Funkcja *train_test_split* zwraca cztery tablice wyjściowe: *X_train, X_test, y_train i y_test. X_train* oraz *y_train* reprezentują zbiór treningowy, który będzie użyty do trenowania modelu uczenia maszynowego. *X_test* i *y_test* reprezentują zbiór testowy, który będzie użyty do oceny wydajności wytrenowanego modelu.
4. Zestawy treningowe i testowe są generowane przez losowe tasowanie danych wejściowych i dzielenie ich na podstawie współczynnika *test_size*. Punkty danych w zbiorach treningowych i testowych są niezależne i losowo wybrane z oryginalnego zbioru danych, aby zapewnić bezstronną ocenę wydajności modelu.
5. Po uzyskaniu zbiorów treningowych i testowych, można je wykorzystać do wytrenowania modelu uczenia maszynowego na danych *X_train* i *y_train*, a następnie ocenić wydajność modelu na danych *X_test* i *y_test* przy użyciu różnych metryk oceny, aby ocenić jego dokładność i zdolność do generalizacji.
6. Na koniec jakość predykcji będzie oceniony miarą „*Accuracy*”. Losowanie zbioru treningowego zostanie powtórzone 10 razy, następnie wyniki zostaną uśrednione. Proces ten nazywany jest walidacją krzyżową i pozwala na lepszą ocenę jakości modelu niż proste podzielenie danych na zbiór treningowy i testowy. W trakcie walidacji krzyżowej, każda część zbioru danych wykorzystywana jest zarówno do treningu, jak i

testowania modelu, co pozwala na uzyskanie bardziej reprezentatywnych wyników. Ponadto, walidacja krzyżowa pozwala na wykorzystanie większej ilości danych do treningu modelu, co może prowadzić do lepszych wyników i bardziej optymalnego doboru parametrów.

Funkcję tą zastosowałem zarówno dla regresji logistycznej jak i lasu losowego. Poniżej przedstawię dokładniej jak działają dwa wyżej wymienione algorytmy.

2.3. Algorytm regresji logistycznej

Regresja logistyczna jest algorytmem klasyfikacyjnym, który pozwala przewidywać prawdopodobieństwo przynależności do danej klasy (np. zwycięstwo zespołu w meczu NBA) [25]. Wartość dopasowania regresji logistycznej to prawdopodobieństwo, które należy do przedziału od 0 do 1. W niniejszej pracy zastosowałem regresję logistyczną binarną, gdzie jedna z klas odpowiada za zwycięstwo, a druga za porażkę. Rozpoczynając tworzenie modelu regresji logistycznej, należy najpierw dostosować funkcję logistyczną do danych wejściowych. Współczynniki funkcji logistycznej są szacowane na podstawie danych treningowych, które następnie są wykorzystywane do przewidywania wyników dla danych testowych. Poniżej przedstawię oszacowania parametrów regresji logistycznej.

[25] https://www.lukaszderylo.pl/blog/regresja-logistyczna.html

Tabela 2.1 Oszacowania parametrów regresji logistycznej.

Nazwa zmiennej	Oszacowania parametrów	Średni błąd oszacowań	p-value
const	-3,6737	4.562	0.421
Home_FG_PCT_10	-3,666	6.687	0.584
Home_FG3_PCT_10	2,6233	2.511	0.367
Home_FT_PCT_10	0,5868	1.550	0.705
Home_REB_10	2,4325	1.503	0.106
Home_AST_10	1,0804	1.061	0.309
Home_TOV_10	-3,5226	1.918	0.066
Home_STL_10	2,3452	1.736	0.177
Home_BLK_10	0,9482	1.236	0.443
Home_TS_PCT_10	4,5531	6.621	0.492
RESULT_ 10	0,7754	0.562	0.167
Away_FG_PCT_10	3,1842	6.544	0.627
Away_FG3_PCT_10	1,6826	2.650	0.525
Away_FT_PCT_10	1,5642	1.875	0.404
Away_REB_10	-1,74	1.564	0.266
Away_AST_10	-1,2558	1.240	0.311
Away_TOV_10	-1,5859	1.847	0.391
Away_STL_10	0,9937	1.942	0.609
Away_BLK_10	-0,1676	0.995	0.866
Away_OFF_RATING_10	-3,0658	2.774	0.269

Tak jak możemy zauważyć w powyższej tabeli dla przyjętej wartości p-value 0,05 żadna ze zmiennych nie jest istotna. Ważne jest jednak, aby mieć na uwadze że p-value nie daje bezpośredniej informacji o wielkości efektu lub praktycznej istotności wyników. Niskie wartości p mogą wskazywać na statystyczną istotność, ale niekoniecznie na praktyczne znaczenie związku między zmiennymi. Dlatego zawsze warto również interpretować wyniki w szerokim kontekście. Natomiast w moim wypadku oszacowania parametrów wydają się być niezgodne z intuicją np. skuteczność rzutowa drużyny grającej u siebie negatywnie wpływa na oszacowania zmiennej RESULT, która przyjmuję 1 jeżeli drużyna grająca we własnej hali wygra mecz. Zgodnie z analizą najsilniejszy pozytywny wpływ na prawdopodobieństwo wygranej zespołu grającego we własnej hali ma zmienna *Home_TS_PCT_10.*

W przypadku modelu regresji logistycznej binarnej, wartość progowa jest wykorzystywana do klasyfikacji wyników - w omawianym modelu wartością progową wynosi 0,5. Jeśli prawdopodobieństwo przynależności do klasy oznaczającej zwycięstwo jest większe niż

wartość progowa, to wynik jest klasyfikowany jako zwycięstwo. W przeciwnym razie, wynik jest klasyfikowany jako porażka. Regresja logistyczna jest jednym z popularnych algorytmów klasyfikacyjnych, które są stosowane w przewidywaniu wyników meczów NBA[26]. Poniżej przedstawię krok po kroku jak działa algorytm w Pythonie:

1. Pierwszym krokiem, po podzieleniu danych na testowe oraz treningowe będzie użycie gotowej implementacji algorytmu dostępnej w bibliotece *sklearn* o nazwie *LogisticRegression.*
2. Następnie przeprowadzę proces uczenia modelu regresji logistycznej na danych treningowych używając do tego wspomnianej walidacji krzyżowej.
3. Po dopasowaniu modelu możemy przystąpić do predykcji na grupie testowej. Model będzie przyjmował wejściowe cechy nowych danych i na ich podstawie będzie przewidywał przynależność do jednej z dwóch klas 0 albo 1.
4. Po dokonaniu predykcji przeprowadzę ocenę modelu używając wyniku miary „*Accuracy*" oraz krzywej ROC, pola AUC i macierzy błędów.
5. Następnie po zbadaniu wyjściowego modelu dokonam paru zmian w celu poprawy jakości predykcji stosując proces optymalizacji *Hyperparameter Tuning* i regularyzację[27].
6. Na końcu porównam wynik modelu bazowego z modelem po dokonaniu usprawnień.

Parametry, które poddam dostrojeniu w algorytmie uczenia maszynowego:

1. „*penalty*" - Ten parametr określa typ regularyzacji, który ma być zastosowany do modelu. Regularyzacja grzbietowa, czyli inaczej normalizacja L2 jest metodą, która znajduję zastosowanie w uczeniu maszynowym w celu ograniczenia nadmiernego dopasowania modelu. Metoda ta jest użyteczna, gdy istnieją silnie ze sobą skorelowane zmienne. Tak jak możemy zauważyć na wykresach z rozdziału 1, kilka zmiennych użytych do budowy modelu jest silnie ze sobą skorelowana, dlatego metoda ta wydaję się być dobrym rozwiązaniem. Silna korelacja zmiennych przyczynia się do tego, że macierz korelacji nie jest macierzą jednostkową, czyli jedno z założeń Gaussa Markowa

[26]https://wyoscholar.uwyo.edu/articles/thesis/Development_of_a_logistic_regression_model_to_predict_the_outcome_of_NBA_games/13701274
[27] Techniki regularyzacji pomagają zmniejszyć prawdopodobieństwo nadmiernego dopasowania i uzyskać idealny model.

nie jest spełnione i oszacowanie MNK są obciążone. Regresja grzbietowa poprzez dodanie parametru lambda do funkcji błędu, zmniejsza wielkość współczynników przy parametrach. Metoda ta jest skuteczna, gdy w danych jest dużo szumu, ponieważ zapobiega to zbytniej wrażliwości modelu na poszczególne obserwacje.

2. „*max_iter*" – Określa maksymalna liczbę iteracji.

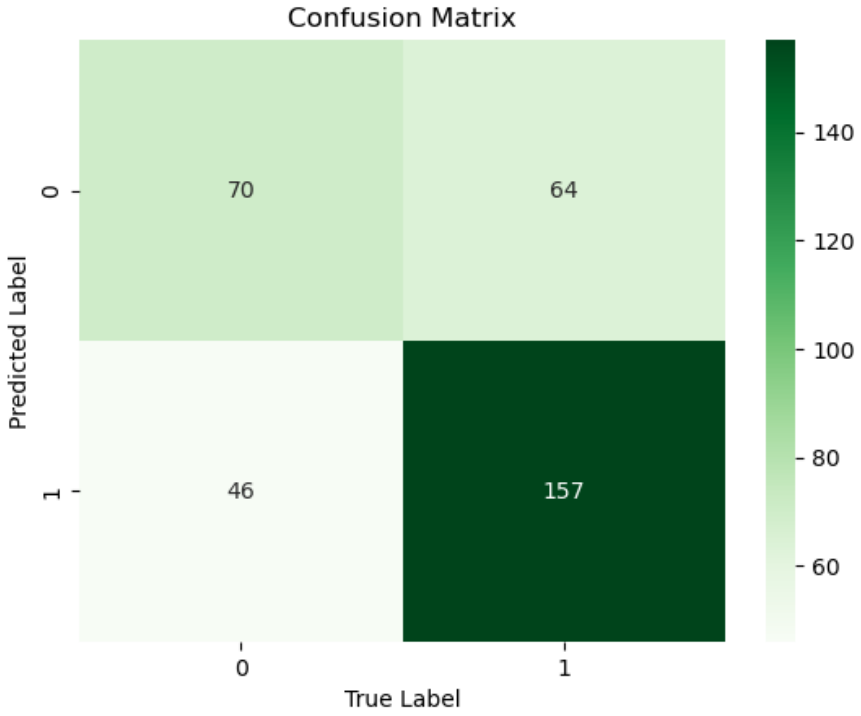
3. C: Ten parametr reprezentuje odwrotność siły regularyzacji. Mniejsze wartości C powodują silniejszą regularyzację, podczas gdy większe wartości powodują słabszą.

W celu dobrania jak najtrafniejszych parametrów dokonałem tak zwanego „*Hyperparameters Tuning*", to znaczy poszukiwanie optymalnych parametrów dla algorytmu. W celu przeprowadzenia analizy korzystałem z pakietu *RandomizedSearchCV* oraz *randint* z biblioteki *Scipy*. Optymalizacja parametrów może być dokonana tylko po podaniu określonych wartości parametrów do przeszukania. Poniżej przedstawię wartości parametrów po zastosowaniu dostrojenia:

- *Max_iter* = 1396
- *Penalty* = L2
- C=3

Po doborze wyżej wskazanych parametrów trafność modelu na zbiorze testowym wynosiła 67,36 % natomiast na zbiorze treningowym 62,76%. Model bazowy natomiast bez zmian parametrów regresji logistycznej, skalowania oraz procesu walidacji krzyżowej osiągnął wyniki na poziomie 60,53 % trafności na zbiorze testowym oraz 62,83% trafności na zbiorze treningowym. Wprowadzanie algorytmu walidacji krzyżowej, skalowanie oraz dostrojenie parametrów przyczyniło się do poprawy efektywności predykcji o około 7 punktów procentowych. Zbliżona wartość parametru „*Accuracy*" dla zbioru testowego oraz treningowego oznacza, że model jest w stanie dobrze generalizować na nowych, nieznanych danych, co jest celem uczenia maszynowego - tworzenia modeli zdolnych do efektywnego przewidywania na nowych danych spoza zbioru treningowego.

Wykres 2.1 Macierz błędu dla regresji logistycznej.



Wykres 2.2 ukazuje macierz błędów, znaną również jako macierz pomyłek. Jest to tabela używana do oceny wyników klasyfikacji modelu. Macierz błędów prezentuje liczbę przypadków, które zostały sklasyfikowane jako prawdziwie pozytywne, fałszywie pozytywne, prawdziwie negatywne oraz fałszywie negatywne. Poniżej przedstawię pełen opis wszystkich możliwych wyników dla macierzy błędu dla modelu używającego klasyfikacji binarnej:

- *True Positive (TP)* to liczba obserwacji poprawnie zaklasyfikowanych do klasy pozytywnej.
- *False Positive (FP)* to liczba obserwacji błędnie zaklasyfikowanych do klasy pozytywnej.
- *True Negative (TN)* to liczba obserwacji poprawnie zaklasyfikowanych do klasy negatywnej.
- *False Negative (FN)* to liczba obserwacji błędnie zaklasyfikowanych do klasy negatywnej.

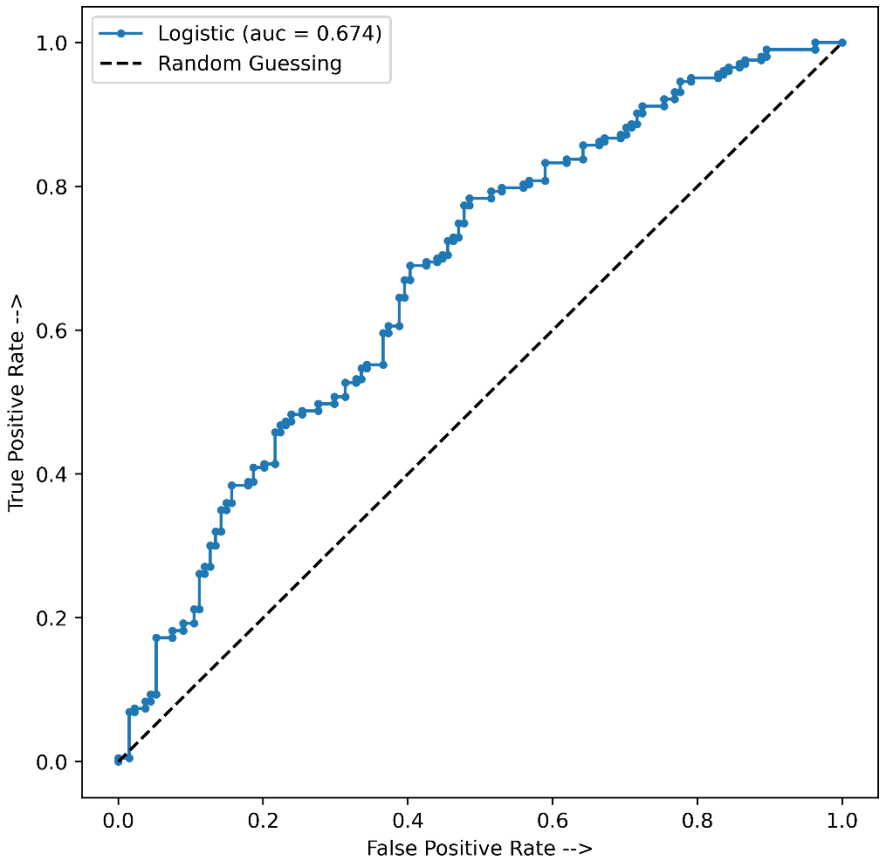
Macierz błędów może być używana do obliczenia różnych metryk oceny jakości modelu, takich jak precyzja, F1-score, specyficzność, wartość predykcyjna pozytywna. Poniżej przedstawię wzory dzięki którym możemy obliczyć specyficzność oraz czułość[28]:

TPR *(True Positive Rate)* = TP/(TP+FN) – czułość – zdolność klasyfikatora do prawidłowego identyfikowania klasy pozytywnej.

TNR *(True Negative Rate)* = TN/(FP+TN) – specyficzność – zdolność klasyfikatora do prawidłowego identyfikowania klasy negatywnej.

Dla modelu regresji logistycznej używając biblioteki *Scikitlearn* obliczyłem czułość, która wynosi 0,603 oraz swoistość na poziomie 0,710. Przedstawione wyniki świadczą o tym, że model lepiej radzi sobie z przewidywaniem porażek niż wygranych.

2.2 Wykres krzywej ROC i AUC dla regresji logistycznej.



[28] https://success.openhealth.fr/pl/articles/3928135-czulosc-i-swoistosc-testu

W statystyce matematycznej krzywa ROC jest graficzną reprezentacją efektywności modelu predykcyjnego poprzez wykreślenie charakterystyki jakościowej klasyfikatorów binarnych powstałych z modelu przy zastosowaniu wielu różnych punktów odcięcia**.** [29] Każdy punkt krzywej ROC odpowiada innej macierzy błędów. Im więcej różnych punktów odcięcia zbadamy, tym więcej uzyskamy punktów na krzywej ROC. Czarna przerywana linia oznacza zgadywanie i AUC na poziomie 0,5. Dodatkowo obliczyłem Współczynnik Giniego, który dla mojego modelu wyniósł 0,348. Jest to pole powierzchni pomiędzy krzywą ROC (niebieskie połączone ze sobą kropki), a krzywą odpowiadającą przerywanej czarnej linii na wykresie. Dla idealnego modelu wartość AUC oraz Współczynniki Giniego wynoszą 1 co oznacza, że im wynik jest bliższy tej wartości tym model jest bardziej precyzyjny. Natomiast wskaźnik AUC (*Area Under Curve*) to prawdopodobieństwo, że badany model predykcyjny oceni wyżej losowy element klasy pozytywnej od losowego elementu klasy negatywnej. Dla stworzonego przeze mnie modelu regresji logistycznej wskaźnik ten wynosi 0,674.

2.4. Algorytm lasu losowego

Las losowy jest kolejnym popularnym algorytmem uczenia maszynowego. Jest to model połączenia wielu klasyfikacji uzyskany za pomocą drzewa decyzyjnego. W lesie losowym, budowany jest zespół drzew decyzyjnych, gdzie każde drzewo jest trenowane na losowym podzbiorze danych treningowych. Proces ten znany jest jako *bootstrapping*, ponieważ dane są próbkowane z oryginalnego zestawu danych treningowych, tak aby stworzyć wiele podzbiorów. Dodatkowo, przy każdym podziale drzewa decyzyjnego, tylko losowy podzbiór cech jest brany pod uwagę. Pomaga w ograniczeniu ryzyka przetrenowania modelu, a tym samymi poprawia zdolność modelu do uogólniania na wcześniej nie znanych danych. Lasy losowe są znane ze swojej zdolności do pracy z danymi wielowymiarowych ze złożonymi interakcjami między zmiennymi i znajdują zastosowanie do m.in. rozpoznawania obrazów[30]. Są one również w stanie zadowalająco radzić sobie z brakującymi wartościami w danych. Cechują się mniejszą podatnością na przetrenowanie w porównaniu do pojedynczych drzew decyzyjnych, co czyni

[29] https://mathspace.pl/matematyka/receiver-operating-characteristic-krzywa-roc-czyli-ocena-jakosci-klasyfikacji-czesc-7/
[30] https://www.robots.ox.ac.uk/~vgg/publications/2007/Bosch07a/bosch07a.pdf

je popularnym wyborem w projektach uczenia maszynowego. Teraz przedstawię krok po kroku algorytm zastosowany w Pythonie:

1. Pierwszym krokiem po podziale danych na zbiór testowy i treningowy będzie stworzenie instancji modelu *RandomForest* z biblioteki *Scikitlearn*.
2. Następnie przeprowadzony zostanie proces uczenia modelu lasu losowego na danych treningowych używając do tego procesu walidacji krzyżowej.
3. Po dopasowaniu modelu można przystąpić do predykcji na grupie testowej. Model będzie przyjmował wejściowe cechy nowych danych i na ich podstawie będzie przewidywał przynależność do jednej z dwóch klas 0 albo 1.
4. Po przeprowadzeniu predykcji zostanie dokonana ocena modelu na podstawie wyników miary „*Accuracy*" oraz krzywej ROC i macierzy błędów.
5. Następnie po zbadaniu wyjściowego modelu zostanie dokonana próba wprowadzenia zmian mających na celu poprawę jakości predykcji stosując proces optymalizacji oraz *hyperparameter tuning*.
6. Na końcu przystąpię do porównania wyniku modelu bazowego z modelem po dokonaniu usprawnień.

Parametry w algorytmie lasu losowego, które zostały poddane dostrojeniu:

- *n_estimators*: Parametr ten określa liczbę drzew decyzyjnych, które zostaną włączone do lasu losowego. Zwiększenie liczby drzew generalnie poprawia wydajność modelu do pewnego momentu, ale również zwiększa złożoność obliczeniową.

- *max_depth*: Ten parametr ustawia maksymalną głębokość drzew decyzyjnych w lesie losowym. W powyższym przypadku użyty jest do kontrolowania złożoności drzew i zapobiegania nadmiernemu dopasowaniu. Niższa wartość skutkuje mniej złożonymi drzewami z mniejszą zdolnością do uchwycenia złożonych wzorców w danych. Natomiast wysoka wartość tego parametru prowadzi do bardziej złożonych modeli, które mogą nadmiernie dopasować się do danych treningowych.

- *min_samples_split*: Ten parametr określa minimalną liczbę obserwacji wymaganych do podziału węzła podczas konstrukcji drzewa. Parametr oceni liczbę obserwacji w węźle

i jeśli liczba ta jest mniejsza od minimalnej, wtedy nie nastąpi podział i węzeł zostanie liściem.

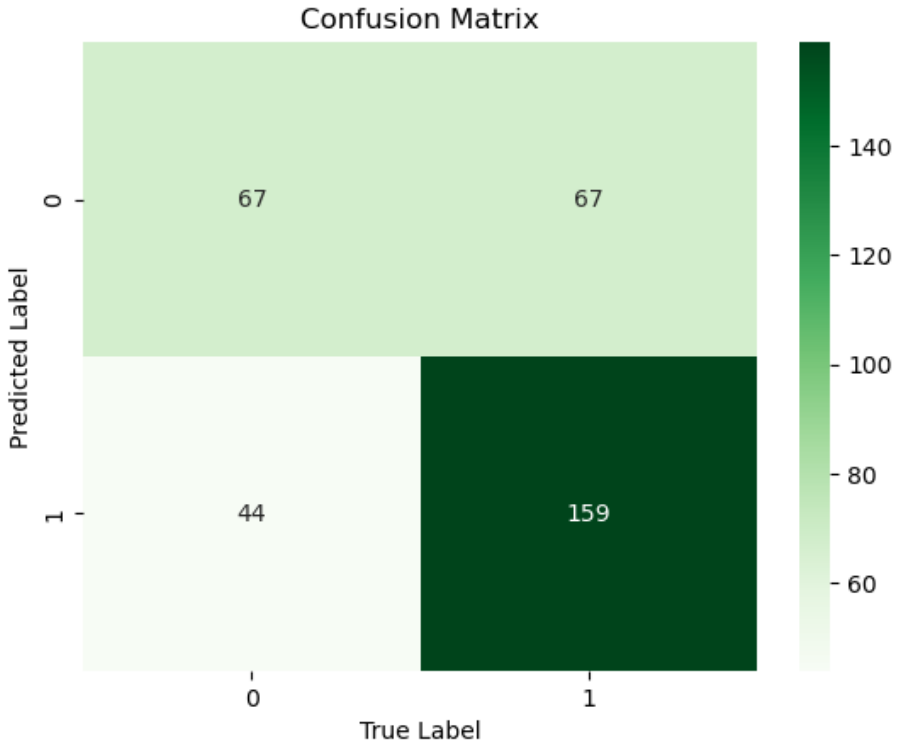
- *max_leaf_nodes:* Ten parametr określa maksymalną liczbę węzłów w każdym drzewie decyzyjnym. Może być używany do kontroli rozmiaru i złożoności drzew. Ustawienie tego parametru na odpowiednim poziomie może zapobiec nadmiernemu dopasowaniu. Natomiast za niska wartość może ograniczyć zdolność drzewa do uchwycenia złożonych wzorców w danych.

Do poszukiwania najefektywniejszej wartości parametrów użyłem wcześniej wspomnianego pakietu *RandomizedSearchCV*. Wyniki dostrojenia przedstawię poniżej:

- *max_dept*: 12,
- *max_leaf_nodes*: 931,
- *min_samples_split*: 523,
- *n_estimators*: 1628.

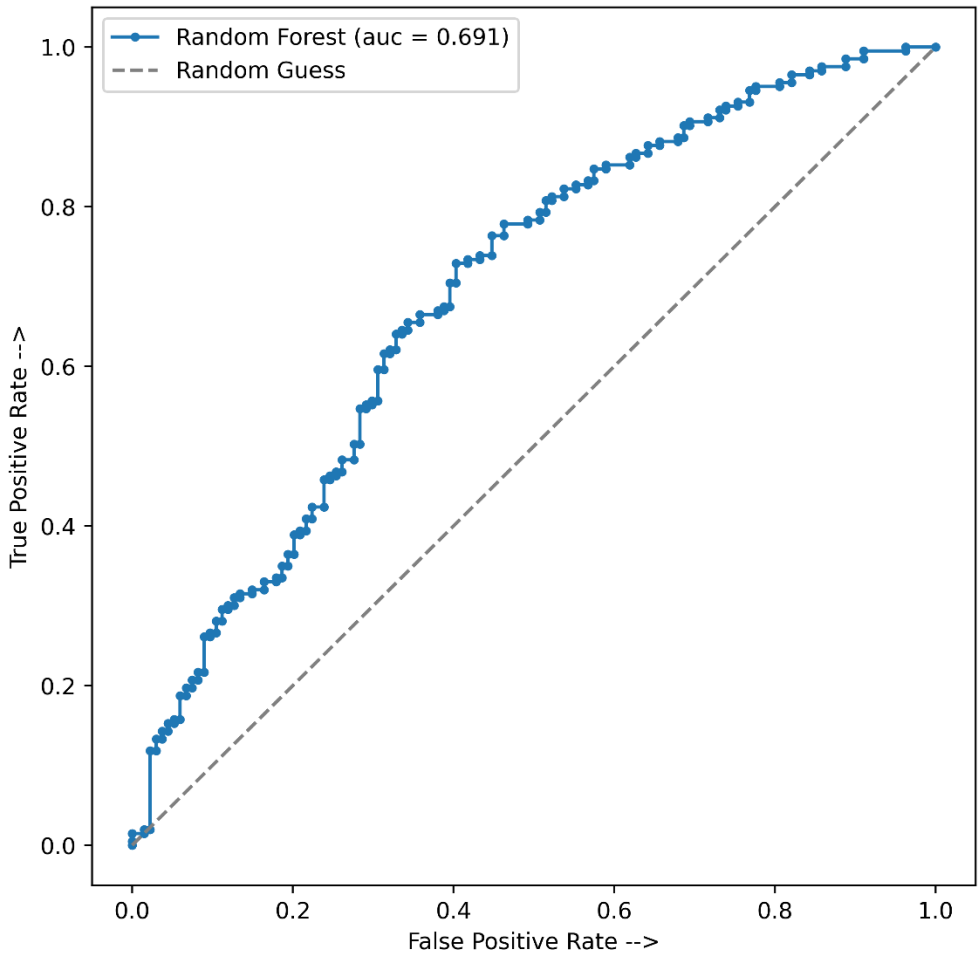
Używając powyższych parametrów trafność modelu na zbiorze testowym wyniosła 67,06% a na zbiorze treningowym 63,72%. Jest to wynik znacząco lepszy od modelu bazowego, który na zbiorze testowym osiągnął skuteczność 61,12%, a na zbiorze treningowym przewidział wszystkie wyniki poprawnie. Stuprocentowa skuteczność na zbiorze treningowym oznacza, że model bazowy jest nadmiernie dopasowany do danych szkoleniowych. Nadmierne dopasowanie występuje, gdy model jest zbyt złożony i przechwytuje szum w danych szkoleniowych, co prowadzi do słabej wydajności generalizacji na niewidzianych danych. W tym przypadku model zapamiętuje dane szkoleniowe zamiast uczyć się podstawowych wzorców i zależności w danych. W naszym głównym modelu możemy zaobserwować małą różnicę między zbiorem testowym a treningowym co oznacza, że model dobrze radzi sobie z predykcjami na wcześniej nie widzianych danych

Wykres 2.3 Macierz błędów dla lasu losowego.



Dla algorytmu lasu losowego obliczyłem również specyficzność i wrażliwość odpowiednio 0,704 oraz 0,604. Wyższy wynik dla specyficzności oznacza, że także w tym wypadku model lepiej radzi sobie z przewidywaniem porażek.

Wykres 2.4 Wykres krzywej ROC i AUC dla lasu losowego.



Nawiązując do poprzedniego podrozdziału, im wyżej położona jest krzywa ROC tym wskazuje to na dokładniej funkcjonujący model. Idealna krzywa ROC znajdowałaby się w lewym górnym rogu wykresu, wskazując na model o doskonałej dyskryminacji między dwiema klasami, w którym uzyskuje się wszystkie prawdziwe wyniki pozytywne bez fałszywych. Losowy klasyfikator na wykresie to czarna przerywana linia ukośna przebiegająca od lewego dolnego do prawego górnego rogu, wskazująca na równe szanse uzyskania wyników prawdziwie pozytywnych i fałszywie pozytywnych. Dla modelu lasu losowego pole AUC wynosi 0,691 a Współczynnik Giniego 0,382. Wyniki te są nieznacznie lepsze od modelu regresji logistycznej jednak całościowe porównanie zamieszczę dopiero w rozdziale 3.

Rozdział 3. Porównanie wyników modeli

Modele uczenia maszynowego utworzone w niniejszej pracy wykorzystują średnie statystyki zespołowe z ostatnich meczy dla podstawowych oraz bardziej zaawansowanych statystyk drużynowych. Na ich podstawie udało się stworzyć modele uczenia maszynowego których wyniki przedstawię w tabeli 3.1

Tabela 3.1 Porównanie wyników predykcji pomiędzy modelami

Nazwa wskaźnika	Regresja logistyczna	Las losowy
Accuracy test	67,36%	67,06%
Accuracy training	62,76%	63,72%
Gini coefficient	0,348	0,382
AUC	0,674	0,691
Sensivity	0,603	0,604
Specificity	0,710	0,704

Z tabeli 3.1 można zaobserwować, iż model regresji logistycznej nieznacznie lepiej poradził sobie z trafnością predykcji. Na powyższy stan rzeczy może mieć wpływ kilka czynników m.in:

- Regresja logistyczna jest prostszym modelem niż las losowy, z mniejszą ilością parametrów do dostrojenia. W rezultacie, model ten może być mniej podatny na nadmierne dopasowanie do danych i bardziej stabilny w działaniu na zbiorze dotyczącym statystyk pochodzących z meczów NBA.
- Regresja logistyczna zakłada liniowy związek pomiędzy zmiennymi objaśniającymi a zmienną objaśnianą, co może znaleźć odzwierciedlenie w omawianym w niniejszej pracy zbiorze danych. Las losowy lepiej radzi sobie w sytuacji nieliniowych zależności

pomiędzy cechami a zmienną docelową, ale możliwe, że wymaga większej ilości danych i lepszego dostrojenia.

- Sposób, w jaki cechy są zaprojektowane, może również odgrywać rolę. Jeśli zmienne objaśniające są liniowo związane ze zmienną objaśnianą, wówczas regresja logistyczna może przewyższyć las losowy. Z drugiej strony, jeżeli istniałyby złożone, nieliniowe relacje pomiędzy zmiennymi objaśniającymi a zmienną docelową, wówczas las losowy mógłby mieć przewagę.

Warto zaznaczyć, że modele które stworzyłem posiadają elementy które w przyszłości można poprawić. Dlatego stworzyłem listę zmian wartych zaimplementowania, dzięki którym jakość predykcji może okazać się dokładniejsza:

- Inżynieria cech: W przypadku danych NBA, można stworzyć dodatkowe zmienne, aby uchwycić więcej informacji na temat bilansu między drużynami, dynamiki zespołu oraz siły przeciwnika. Do modelu można dodać bilans między każdymi zespołami z np. ostatnich trzech sezonów który pomógłby w przewidywaniu zwycięskiej drużyny. Kolejną zmienną, wartą utworzenia jest *ELO rating*, czyli parametr, który w dokładniejszy sposób aktualizowałaby dyspozycje zespołu po każdym meczy sezonu.

- *Ensembling*: Techniki grupowania, takie jak *bagging*, *boosting* lub *stacking*[31], mogą być stosowane do łączenia wielu modeli predykcyjnych i poprawy ogólnej wydajności. Na przykładzie niniejszej pracy, można połączyć las losowy z modelem regresji logistycznej, aby stworzyć model zespołowy, który wykorzystuje mocne strony obu metod. Dodatkowo metodami, które po przeprowadzonej przeze mnie analizie wydają się pasować do omawianego zbioru danych są mi: sieć neuronowa, *AdaBoost* oraz *XGBoost*.[32]

- Wykorzystanie większej ilości danych: Warte rozważenia może być użycie większej ilości danych do trenowania modelu predykcyjnego. Przy wykorzystaniu zmiennych z sezonów sprzed np.10 lat z pewnością warto przyjrzeć się: historycznemu bilansowi,

[31] https://www.toptal.com/machine-learning/ensemble-methods-machine-learning
[32] https://towardsdatascience.com/the-ultimate-guide-to-adaboost-random-forests-and-xgboost-7f9327061c4f

wyborom w drafcie, zmianie na stanowisku trenera. Dlatego w tym wypadku istotne wydaje się zastosowanie jednocześnie wspomnianego wcześniej *ELO rating*, ze względu na to że wyniki drużyn mogą znacząco się zmieniać na przestrzeni lat.

- Dodanie nowych zmiennych: Dobrym pomysłem jest wzbogacenie modelu o nowe zmienne korzystając z danych dostępnych publicznie. Przed każdym meczem drużyna ma obowiązek poinformować o wszystkich kontuzjach i graczach niedostępnych na dane spotkanie - jest to tak zwany *injury report*[33]. Często informacja o tym, że gwiazda zespołu nie wystąpi w danym meczu wpływa na jego rezultat. Dlatego moim zdaniem wartą rozważania metodą jest dodanie list zawodników przed meczem i wskazanie kluczowych koszykarzy w danym zespole.

[33] https://official.nba.com/nba-injury-report-2022-23-season/

Podsumowanie

Celem powyższej pracy było stworzenie modeli opartych na procesie uczenia maszynowego do prognozowania wyników meczów sezonu zasadniczego w amerykańskiej lidze koszykówki – National Basketball Association. Dane dotyczące spotkań sezonu regularnego, mimo ich ogólnodostępności, stawiają liczne wyzwania przed modelowaniem predykcyjnym. Przeglądając strony internetowe z modelami stworzonymi przez innych analityków oraz studentów odnotowałem, że celność ich predykcji waha się w przedziale 63-72 procent. Modele stworzone przeze mnie - ze skutecznością około 67 procent - wpisują się akurat pośrodku stawki. Badane przeze mnie dwa sezony regularne sprzed rozpoczęcia pandemii, były relatywnie jednorodne i cechowała je dość duża przewidywalność[34]. Drużyny z początku sezonu określone mianem faworytów spełniły oczekiwania fanów i układ tabeli w dużej części pokrywał się z przewidywaniami analityków. Dzięki temu skuteczność modeli stworzonych na potrzeby powyższej pracy była wyższa (dodając dane z przyszłego sezonu 2020-21 jakość predykcji spadła o około 3 punkty procentowe). Jedną z popularnych technik uczenia maszynowego wykorzystywanych w modelowaniu predykcyjnym NBA jest regresja logistyczna, którą zastosowałem w tej pracy. Metoda ta znalazła zastosowanie w przewidywaniu prawdopodobieństwa wyniku binarnego. Innym znanym algorytmem uczenia maszynowego używanym przy pracy z danymi pochodzącymi z NBA, który został zaimplementowany w niniejszej pracy jest algorytm lasu losowego. Lasy losowe są zespołem drzew decyzyjnych, gdzie każde drzewo jest budowane przy użyciu losowego podzbioru dostępnych zmiennych wejściowych. W celu przezwyciężenia przeszkód związanych z modelowaniem predykcyjnym zastosowałem szereg technik, w tym inżynierię zmiennych, walidację krzyżową i strojenie parametrów dzięki którym jakość moich predykcji uległa poprawie.

[34] https://bleacherreport.com/articles/2789084-nba-schedule-2018-19-team-by-team-record-predictions-and-playoff-odds

Bibliografia

[1] Christian Gough, *National Basketball Association total revenue from 2001/02 to 2021/22*, https://www.statista.com/statistics/193467/total-league-revenue-of-the-nba-since-2005/

[2] Merrimack College, *How NBA Analytics is Changing Basketball,* https://online.merrimack.edu/nba-analytics-changing-basketball/

[3] Medium**,** *How Data Transformed NBA,* https://randerson112358.medium.com/how-data-transformed-the-nba-1cbc8b24e130

[4] Digital Editors, *what is Load Management and How Does it Affects NBA Players,* https://www.sportscasting.com/what-load-management-how-affect-nba-players/

[5] Petra, *How data analytics is revolutionizing the NBA*, https://d3.harvard.edu/platform-digit/submission/how-data-analytics-is-revolutionizing-the-nba/

[6] Nastazja Bloch, *Rafał Jud- polski łowca talentów dla NBA:" Same umiejętności techniczno-taktyczne są najmniej istotne"*, https://dziendobry.tvn.pl/styl-zycia/rafal-juc-polski-lowca-talentow-dla-nba-na-czym-polega-praca-skauta-da328352-5317220

[7] *NBA League Averages- Per Game*, https://www.basketball-reference.com/leagues/NBA_stats_per_game.html

[8] Tom Sunnergren, *IPop: How Big Data Will Transform Coaching in NBA,* https://bleacherreport.com/articles/1934273-ipop-how-big-data-will-transform-coaching-in-the-nba

[9] Tadd Haislop, *NBA bubble, explained: A complete guide to the rules, teams, schedule & more for Orlando games,* https://www.sportingnews.com/us/nba/news/nba-bubble-rules-teams-schedule-orlando/zhap66a9hcwq1khmcex3ggabo

[10] Damian Puchalski, *NBA: Czym są zaawansowane statystyki rzutowe? Analiza i wyjaśnienie (część 1)*, https://probasket.pl/nba-czym-sa-zaawansowane-statystyki-rzutowe-analiza-i-wyjasnienie/

[11] Kevin Belhumeur, *How Important Is Home_Court Advantage in the NBA?,* https://bleacherreport.com/articles/1520496-how-important-is-home-court-advantage-in-the-nba

[12] Todd Whitehead, *Nylon Calculus: More lessons from NBA rebound tracking*, https://fansided.com/2020/01/28/nylon-calculus-nba-rebound-tracking/

[13] NBA Team Percent of Point from Free Throws, NBA Stats - NBA Team Percent of Points from Free Throws | TeamRankings.com

[14] Łukasz Derylo, *Regresja logistyczna – co to jest?* https://www.lukaszderylo.pl/blog/regresja-logistyczna.html

[15] Mariusz Gromada*, Receiver Operating Characteristic – Krzywa ROC – czyli ocean jakośći klasyfikacji (część 7)* https://mathspace.pl/matematyka/receiver-operating-characteristic-krzywa-roc-czyli-ocena-jakosci-klasyfikacji-czesc-7/

[16] Anthony Cabos, *Czułość i swoistość testu*, https://success.openhealth.fr/pl/articles/3928135-czulosc-i-swoistosc-testu

[17] IBM, *what is overfitting*, https://www.ibm.com/topics/overfitting

[18] Necati Demir, *Ensemble Methods: Elegant Techniques to Produce Improved Machine Learning Results*, https://www.toptal.com/machine-learning/ensemble-methods-machine-learning

[19] Julia Nikulski, *The Ultimate Guide to AdaBoost, random forest and XGBoost*, https://towardsdatascience.com/the-ultimate-guide-to-adaboost-random-forests-and-xgboost-7f9327061c4f

[20] NBA Website, *NBA Injury Report: 2022-23 Season*, https://official.nba.com/nba-injury-report-2022-23-season/

[21] Alec Nathan, *NBA Schedule 2018-19: Team-by-Team Record Predictions and Playoff Odds*, https://bleacherreport.com/articles/2789084-nba-schedule-2018-19-team-by-team-record-predictions-and-playoff-odds

[22] Zachary Campbel, *Development of a logistic regression model to predict the outcome of NBA games, https://wyoscholar.uwyo.edu/articles/thesis/Development_of_a_logistic_regression_model_to_predict_the_outcome_of_NBA_games/13701274*

[23] *Anna Bosch, Andrew Zisserman, Xavier* Munoz, *Image Classification using Random Forests and Ferns*, https://www.robots.ox.ac.uk/~vgg/publications/2007/Bosch07a/bosch07a.pdf

Streszczenie

Niniejsza praca dotyczy prognozowania wyników NBA przy użyciu dwóch algorytmów: regresji logistycznej oraz lasu losowego. W moich badaniach zastosowałem wspomniane wyżej metody aby za pomocą analizy danych pochodzących z NBA przewidzieć zwycięską drużynę pojedynczego meczu sezonu regularnego w latach 2018-29 oraz 2019-20. W pierwszym rozdziale skupiłem się na charakterystyce ligi NBA, zmianach jakie zaszły w tej organizacji pod wpływem dynamicznego rozwoju analityki oraz big data. Przyjrzałem się także bliżej zmiennym uwzględnionym w modelu oraz danym zgromadzonym na potrzeby prognoz. Następnie w rozdziale drugim przy użyciu algorytmów uczenia maszynowego przeprowadziłem podział na grupę testową oraz treningową oraz proces uczenia modelu. Za pomocą strojenia parametrów i przeskalowania zmiennych, udało mi się osiągnąć dokładniejsze predykcję. W ostatnim rozdziale przy użyciu metod klasyfikacyjnych porównałem modele między sobą i zaproponowałem kilka usprawnień, które mogą poprawić jakość stworzonych przeze mnie modeli.