VILNIAUS UNIVERSITETAS
MATEMATIKOS IR INFORMATIKOS FAKULTETAS
MATEMATINĖS STATISTIKOS KATEDRA

Krepšinio rungtynių baigčių prognozavimas

Forecasting Outcomes of Basketball Games

Magistro baigiamasis darbas

Atliko:	Ignas Deltuva	(parašas)
Darbo vadovas:	doc. dr. Rūta Levulienė	(parašas)
Recenzentas:	doc. dr. Viktor Skorniakov	(parašas)

Vilnius – 2017

Santrauka

Šiame darbe Nacionalinės Krepšinio Asociacijos (angl. NBA) rungtynių rezultatams pritaikyti regresiniai, neuroninių tinklų modeliai naudojant 2010-2016 sezonų duomenis. Atliekama analizė remiasi komandos pajėgumu, intensyvaus grafiko, varžovų praeities akistatų faktoriais. Darbe komandos pajėgumas interpretuojamas kaip komandos stiprumo, rangavimo turnyrinėje lentėleje indikatorius sezono metu. Komandos galimybės įvertinamos pasezoniui, leidžiant nusakyti tam tikrą pokytį organizacijoje- pasikeitimą komandoje dėl krepšininkų traumų, suspendavimo, dėl pasikeitimų komandos vadovybėje ar rotacijoje, arba tiesiog pasikeitus NBA politikai, lemiančiai sezono biudžeto, žaidėjų algų kepurės klausimus. Modelis taip pat bando nusakyti ir aikštės pranašumo įtaką. Visos šios analizės tikslas- prognozuoti ateities rungtynių baigtis ir rezultatus. Darbe sudaryti modeliai keliais būdais- mažiausio kvadrato, didžiausio tikėtinumo ir apibrendrinto mažiausių kvadratų metodais, pasiremta ir neuroninių tinklų modeliu bandant nuspėti rungtynių nugalėtoją, gauti rezultatai aptariami ir pasiūlymai pateikiami tolimesniam darbui- kitaip tariant, žingsnis link tobulesnės modelių prognozės.

Raktiniai žodžiai: tiesinės regresijos modelis, rungtynių baigčių prognozė, komandos pajėgumas, krepšinis, neuroniniai tinklai

Summary

In this master thesis, NBA data has been extracted, located into the database and, afterward, regression, neural network models have been applied for the 2010-2016 season data. The analysis is based on the team's strength, an ability to play back-to-back games. Applied models also take into account factors such as home court advantages, flight travels, which could indicate an exhaustion, a downgrade in the team's performance during the game. Also we strongly consider adding defence indicators like how many rebounds a team is able to grab and how many points a team allows the opponents to score, indicating whether a team relies on defence or offense. An estimated team's strength could have an interpretation of changes in the team's management, roster structure, significant player trades or even the budget the team managed to increase or reduce during the off-season. Our main objective is to forecast the game outcome as precise as possibile using linear regression and neural network models, which try to predict a winning team. Results have been discussed and suggestions have been made for the futher, more accurate analysis in the future work.

Keywords: Linear Regression Model, Outcome Forecast, Team Strength, Basketball, Neural Networks

Turinys

Įvadas

Kiekvienais metais atsiranda vis daugiau ir daugiau sporto entuziastų, bandančių pritaikyti statistines žinias, sukauptą patirtį ir kuo tiksliau nuspėti rungtynių rezultatus joms dar neprasidėjus. Viena iš galimų priežasčių- sportinis azartas, sporto populiarumas, pritaikytų sprendimų pasidalinimas internetinėje erdvėje ir įvairios statistikos prieinamumas paprastam mirtingajam- bent dėl keletos iš minėtų argumentų magistriniame darbe nagrinėjama krepšinio rungtynių baigčių prognozės problematika.

Krepšinis- viena iš populiariausių sporto šakų, bent jau stereotipinio lietuvio akimis, taigi ne nuostabu, kad Nacionalinė Krepšinio Asociacija (angl. NBA) kasnakt sukausto milijonus žiūrovų prie televizoriaus ar (šiuolaikiškiau būtų galima sakyti) monitoriaus ekrano. Paprastam, bent kažkiek besidominčiam sportu, darbu su duomenimis bei jų analize studentui ar specialistui tai reiškia didelius duomenų kiekius, kurie yra daugeliu atveju lengvai prieinami. Iš tiesų, šiais moderniais laikais vien NBA lyga, remdamasi inovacijų gausa, pateikia vis daugiau įvairios statistikos, kuri kartais, galbūt, ir gali pasirodyti juokinga, bet kokiu atveju, kalbame apie milžinišką duomenų kiekį, paruoštą tiek elementariam grafiniam atvaizdavimui, tiek gilesnei duomenų statistinei analizei. Kaip įdomų pavyzdį galėtume pateikti įdiegtą sistemą, kuri, naudodama 6 kameras, iš skirtingų kampų užfiksuoja kiekvieno žaidėjo trajektorijas aikštėje, visą šią informaciją apie žaidėjo, kamuolio trajektorijas laike galime rasti gulinčią viename iš NBA serverių.

Taigi, šis darbas- ne išimtis ir pagrindinis dėmesys skiriamas informacijos ištraukimui iš neapdorotų, bet laisvai prieinamų duomenų, tiesiai iš oficialaus NBA puslapio, kuriuos, sukrovus į duomenų bazę, bandoma atrinkti pagrindinius indikatorius, aiškinančiuosius kintamuosius, rungtynių baigčių prognozei. Prognozės modeliavimas iš esmės remiasi tiesiniais regresiniais modeliais, bandant įvertinti komandų pajėgumus, komandų praeities akistatos, intensyvaus rungtynių tvarkaraščio veiksnius, kurie turėtų padėti atspėti komandų baigtį ir įvertinti potencialų nugalėtoją. Darbe vienas iš tikslų- išlaikant paprastumą, kuo tiksliau nuspėti rungtynių baigtį dar mačui neprasidėjus. Darbo eigoje įvairių metodais sudaryti modeliai yra palyginimai ir iš jų standartiniais modelio tinkamumo principais (MSE, MAE) parenkamas geriausias variantas, kurį dar spėsime aptarti. Darbo išsikelta užduotis- įvertinti galutinį taškų skirtumą tarp žaidžiančių komandų, kalbant bendriau- bandyti atspėti rungtynių nugalėtoją.

1 Duomenų šaltinis

Nagrinėjama problema remiasi sezoniniais 2010-2016 metų duomenimis, pasiskolintais iš `http://stats.nba.com` puslapio. Yra sakoma, kad duomenų ištraukimas, paruošimas analizei užima daugiau laiko negu pati analizė. Tokia tendencija įsitikinome ir šiame darbe dėl keletos priežasčių: 1) vienas iš iškeltų tikslų- susikrauti norimus rodiklius į duomenų bazę, iš kurios būtų patogu norimais pjūviais nagrinėti rodiklius, rašant specifines užklausas (tokiu būdu, daugelis transformacijų, duomenų apdorojimo vyktų ne duomenų analizės, o duomenų ištraukimo bei transformavimo metu) 2) kaip ir įprasta, apdorojant duomenis, susidurta su tam tikrais nesklandumais, susijusiais su duomenų integracijos logika, duomenų srauto pakrovimo laiku(duomenų ištraukimo laikas, kuris apytiksliai užtrunka apie 40 minučių vienam sezonui) ar net analitiniu funkcijų neegzistavimu duomenų bazėje (ko gero, nereikia daug tikėtis iš atviro kodo programų- šiuo atveju, SQLite)- su visais šiais iššūkiais teko susidurti, norint turėti bent mažą dalį NBA duomenų asmeniniame kompiuteryje. Bet visi vargai, pastangos tam tikra prasme atsipirko- galime, kasdien kone kelių Python programavimo kalbos funkcijų įvykdymu susikrauti norimą informaciją apie dominamas NBA rungtynes.
Duomenų krovimo logika į duomenų bazę pasikliauna viešai prieinama NBA API informacija, kuri, žinant metaduomenų esybes (angl. entities) ir parametrus, pasiekiama JSON formatu. Nurodžius konkretų sezoną, dominančios esybės URL adresą, susisteminti automatiniai kreipimaisi leidžia pasiekti informaciją apie dominančias rungtynes, žaidėjo statistiką. Duomenų ištraukimas, iš esmės gautas, naudojant Python kompiliatoriumi ir pasirinkta atviro kodo gana lanksčia duomenų baze- SQLite, kreipiantis į nurodytą URL adresą ir eksportuojant JSON faile esančius duomenis. Visa informacija apie detalesnį duomenų apdorojimo planą ir logiką gali būti randama šio darbo priede (žr. "Priedas").
Aptarsime šiame darbe naudotas duomenų bazės lenteles, kurias naudojame užklausoms rašyti:

1 lentelė. TEAM

ID	ABBR
NAME	CITY

2 lentelė. BOXSCORE

TEAM_ID	GAME_ID
GAME_DATE_EST	TEAM_WINS_LOSSES
PTS	PTS_OT1
PTS_OT2	PTS_OT3
PTS_OT4	

3 lentelė. GAME_SUMMARY

HOME_TEAM_ID	AWAY_TEAM_ID
GAME_DATE_EST	SEASON
GAME_ID	

4 lentelė. TEAM_GENERAL_STATS

GAME_ID	TEAM_ID
PLUS_MINUS	PTS

5 lentelė. LAST_MEETING

GAME_ID	LAST_GAME_ID
LAST_GAME_DATE_EST	

6 lentelė. BOXSCORE_ADVANCED

GAME_ID	TEAM_ID
OREB_PCT	DREB_PCT
TM_TOV_PCT	OFF_RATING
DEF_RATING	AST_RATIO
PACE	

Su SQLite galime lengvai manipuliuoti turimais duomenimis- pavyzdžiui, susikurti išvestinius rodiklius- binarinius kintamuosius (angl. dummy variables). Lentelėse nurodyti tik rodikliai ir informacija apie komandą, komandų susitikimus, naudoti analizėje. Priede (žr. "Priedas") galima rasti ir pagrindinę užklausą duomenų analizei.

Naudotų duomenų aprašymas:

- GAME_ID- identifikacinis unikalus rungtynių numeris
- TEAM_ID- identifikacinis unikalus komandos numeris
- HOME_TEAM_ID- identifikacinis unikalus aikštės pranašumą turinčios komandos numeris
- AWAY_TEAM_ID- identifikacinis unikalus išvykoje žaidžiančios komandos numeris
- GAME_DATE_EST- rungtynių data

- LAST_GAME_ID- komandų akistatos praeito susitikimo identifikacinis unikalus rungtynių numeris
- CITY- komandos miestas
- ABBR- komandos sutrumpintas žymėjimas
- SEASON- sezono indikatorius (galimos reikšmės 2012, 2013, …); pavyzdžiui, 2013 atitinka 2013-2014 metų sezoną)
- TEAM_WINS_LOSSES- komandos pergalės-pralaimėjimai tam tikro sezono tarpsniu, konkrečių rungtynių metu
- PLUS_MINUS- taškų skirtumas, atsižvelgiant į komandą (teigiamas, jei komanda laimi, neigiamas kitu atveju)
- PTS- komandos surinkti taškai per rungtynes
- PTS_OT1- komandos surinkti taškai per pirmąjį pratęsimą
- PTS_OT2- komandos surinkti taškai per antrąjį pratęsimą
- PTS_OT3- komandos surinkti taškai per trečiąjį pratęsimą
- PTS_OT4- komandos surinkti taškai per ketvirtąjį pratęsimą
- OREB_PCT- kiek per rungtynes komanda atkovoja kamuolių puolime, santykinai visų puolime atkovotų kamuolių atžvilgiu
- DREB_PCT- kiek per rungtynes komanda atkovoja kamuolių gynyboje, santykinai visų gynyboje atkovotų kamuolių atžvilgiu
- OFF_RATING- kiek komanda surenka taškų per 100 atakų
- DEF_RATING- kiek komanda leidžia varžovams surinkti taškų per 100 atakų
- AST_RATIO- rezultatyvių perdavimų skaičius, nusakantis, kiek vidutiniškai per 100 atakų komandos žaidimas baigiasi rezultatyviais perdavimais
- PACE- kiek komanda atlieka atakų per 48 minutes (per standartinį rungtynių laiką, neįskaitant pratęsimų)

Sistemingai sudėliojus aprašytus rodiklius, galime išvesti papildomus kintamuosius: bandysime nusakyti kelionės įtaką rungtyniaujančiai komandai, taip pat ar kelios pergalės iš eilės atspindi emocinį komandos pakilimą žaidime (t.y., laimėjus bent dusyk iš eilės, ar didėja komandos šansai laimėti ir trečią kartą).

Binariniai kintamieji, išvesti iš turimų rodiklių

- LAST_MEET- 1, jei laimėjo namuose žaidžianti komanda, -1- kitais atvejais
- INTENSE_SCHEDULE- 1, jei komanda žaidžia bent du vakarus iš eilės, -1- kitais atvejais
- FLIGHT_TRAVEL- 1, jei komanda žaidžia po skrydžio, kelionės iš kito miesto, -1 kitais atvejais
- NO_FLIGHT- 1, jei komanda žaidžia namuose antrąkart iš eilės, -1- kitais atvejais

2 Modelių sudarymo metodai

2.1 Tiesinis modelis

Tiesinė regresija yra bene paprasčiausia iš taikomų regresijų. Mes modeliuojame sistemą tiesine nepriklausomų kintamųjų kombinacija tam, kad gautume kuo tiksliau nusakantį prognozės kintamąjį y_i, t.y.,

$$y_i = h(x_i, w) = w^T x_i \tag{1}$$

, čia w- svorių vektorius, x_i nepriklausomų kintamųjų vektorius.

Mūsų užduotis yra rasti svorius w, kuriais gautume geriausius parametrų įverčius, turimai duomenų aibei. Vienas iš būdų, kaip patikrinti parametrų gerumą yra rasti kiek galima mažesnes kvadratines modelio liekanas, t.y.:

$$L(w) = \Sigma_i \left(h(x_i, w) - y_i\right)^2. \tag{2}$$

Laikantis darbo tikslų ir užduočių kryptingumo, visų pirmiausia, panagrinėsime tiesinį regresijos modelį, nuo kurio ir pradėsime rungtynių baigčių prognozę.

2.1.1 Paprasta tiesinė regresija

Turint sukrautus duomenis galime pradėti duomenų analizę. Mus dominantis dydis- taškų skirtumas tarp žaidžiančių komandų, taigi nagrinėsime sumestų taškų skirtumą tarp namuose žaidžiančios ir svečių komandų. Šį atsitiktinį dydį žymėsime $y_{i,j,n}$, kur indeksai i ir j žymi ekipas: i- aikštės pranašumą turinti komanda, j- išvykoje žaidžianti ekipa, o n- tegu būna n-tosios rungtynės ($i, j \in \{1, \ldots, 30\}$ ir $n \in \{1, \ldots, N\}$). Tegu, N- sužaistų rungtynių skaičius, taigi, jeigu nagrinėjame sezono duomenis įprastomis sąlygomis, žinant kad kiekviena komanda (iš viso komandų- 30- lygšiol nekintantis dydis) per sezoną įprastomis sąlygomis (neįvykus lokautui ar neatšaukus rungtynių) sužaidžia lygiai 82 mačus, N tuomet lygus: $N = \frac{82x30}{2} = 1230$. Vadinasi, įprastomis sąlygomis, standartiniame sezone modeliuosime iš 1230 rungtynių (išskyrus kelis sezonus: 1) lokautą, įvykusį 2011-12 metais, sutrumpinusį sezoną iki 990 rungtynių, 2) Bostono maratono metu įvykdytą teroristinį išpuolį 2012-13 sezone, dėl kurio netgi buvo atšauktas mačas tarp Bostono "Celtics" ir Indianos "Pacers" klubų ir taip pat 3) šių metų sezone, rungtynės buvo nutrauktos dėl prastos kondicionavimo sistemos rungtynėse tarp vietos Philadelphijos "76ers" ir Sakramento "Kings" ekipų). Mus dominantį prognozės kintamąjį modeliuosime tokiu principu:

$$
\begin{gathered}
y_{i,j,n} = \lambda + \alpha_1(x_{1_{i,n}} - x_{1_{j,n}}) + \alpha_2(x_{2_{i,n}} - x_{2_{j,n}}) + \\
\alpha_3(x_{3_{i,n}} - x_{3_{j,n}}) + \omega_1(z_{1_{i,n}} - z_{1_{j,n}}) + \omega_2(z_{2_{i,n}} - z_{2_{j,n}}) + \\
\omega_3(z_{3_{i,n}} - z_{3_{j,n}}) + \beta_i - \beta_j + \epsilon_{i,j,n}
\end{gathered} \tag{3}
$$

, čia $\epsilon \sim \mathcal{N}(0, \sigma^2 I)$, $\epsilon_{i,j} \sim \mathcal{N}(0, \sigma^2)$.

Modelio parametrai ir kintamieji:

- λ- namų aikštės pranašumą nusakantis laisvasis narys;
- x_1- binarinis kintamasis, nusakantis, ar praeitoje komandų akistatoje buvo pasiekta pergalė (1- jei taip, -1- kitu atveju);
- x_2- binarinis kintamasis, nusakantis, ar komanda žaidžia kelias dienas paeiliui (tam tikras nuovargio indikatorius, 1- jei taip, -1- kitu atveju);
- x_3- binarinis kintamasis, nusakantis, ar namų komanda namuose žaidžia bent kelissyk paeiliui (1- jei ne, -1- jei taip);
- z_1- nepriklausomas kintamasis, kiek vidutiniškai komanda surenka taškų;
- z_2- nepriklausomas kintamasis, kiek vidutiniškai komanda praleidžia taškų į savąjį krepšį;
- z_3- nepriklausomas kintamasis, kiek komanda per 100 atakų surenka taškų;
- β_i- i-osios komandos pajėgumas.

Modelis matricinėje formoje: kintamasis y, kaip ir ϵ, yra ne kas kita kaip $N \times 1$ vektoriai, tegu $\beta = [\lambda, \alpha 1, \alpha 2, \beta_1, \ldots, \beta_{30}]^T$ yra parametrų vektorius, kuris kartu su aiškinančiųjų kintamųjų ($N \times (30 + 2)$) dydžio matrica X aprašo rungtynių rezultatų baigtį (y atsitiktinį vektorių). Matrix X yra sudaryta tokiu principu: pirmasis matricos stulpelis, kaip ir įprastai susideda iš 1 (laisvojo nario parametro įvertinimui), antrasis stulpelis- skirtumas tarp binarinių kintamųjų (angl. dummy variables) $x1$, trečiame saugomos skirtumo tarp binarinių kintamųjų $x2$ reikšmės, likę matricos X stulpeliai padės įvertinti komandų pajėgumą: 1 reikšmę priskirsime n-tųjų rungtynių ir $i + 7$ stulpelio sankirtai, -1- n-tųjų rungtynių ir $j + 7$ stulpelio sankirtai, jei tai yra akistata tarp namuose i-osios ir svečiuose j-osios žaidžiančių komandų, likusios matricos reikšmės susideda iš 0. Turint matricinę kintamųjų, parametrų išraišką, galime modelį užrašyti kompaktiškesne išraiška:

$$
y = X\beta + \epsilon \tag{4}
$$

Vis dėlto, matrica X nėra pilno rango, taigi tam, kad galėtume apskaičiuoti parametrų įvertinius, tenka priimti vieną apribojimą- tarsime, kad vienos iš komandos (tegu ta komanda būna- 1-oji) pajėgumas yra $\beta_1 = 0$, tokiu būdu, galime pašalinę 8-tąjį matricos X stulpelį, apskaičiuoti dominamus įvertinius mažiausių kvadratų metodu.

Sudarytas modelis turi ir daugiau ribojimų- priimame prielaidą, kad visos komandos intensyvus tvarkaraščio grafikas turi vienodą įtaką komandų pasirodymui. Patikrinus ir vieną kitų prielaidų, mažiausių kvadratų metodo (angl. OLS) modelio prielaidas: normalumo testai išsklaido visas dvejones apie paklaidų normalumą- paklaidos pasiskirsčiusios pagal normalųjį skirstinį.

Visgi sudarius tiesinę regresiją, pastebime nereikšmingus kintamuosius, kuriuos paeiliui (nuo labiausiai nereikšmingo kintamojo) vieną po kito šaliname iš modelio. Atsikračius nereikšmingais kintamaisiais, galutinė modelio išraiška:

$$y_{i,j,n} = \lambda + \alpha_2(x_{2_{i,n}} - x_{2_{j,n}}) + \alpha_3(x_{3_{i,n}} - x_{3_{j,n}}) + \beta_i - \beta_j + \epsilon_{i,j,n} \tag{5}$$

2.1.2 Heteroskedastiškumo problema

Intuicija kužda apie galimą paklaidų heteroskedastiškumo problemą, t.y., logiškai samprotaujant, modelio dispersija nebūtinai turi būti pastovi, kiekviena komanda būti turėti įtakos modelio dispersijai, kuri, atitinkamai atsižvelgiant į komandas gali kisti- tai reikštų nepastovią dispersiją- heteroskedastiškumą. Taigi gan pagrįstos dvejonės ir samprotavimai verčia ieškoti modelio papildymo ar patobulinimo. Vienas iš būdų, kuris padėtų nuvyti į šalį visas dvejones- modelį papildyti sąlyga, kad kiekviena komanda turi skirtingą efektą modelio paklaidų išssibarstymui, t.y., modelį aprašyti tokiu būdu:

$$y_{i,j,n} = \lambda + \alpha_2(x_{2_{i,n}} - x_{2_{j,n}}) + \alpha_3(x_{3_{i,n}} - x_{3_{j,n}}) + S_{i,n} - \S_{j,n} + \epsilon_{i,j,n} \tag{6}$$

, čia $S_{i,n} = \beta_i + \epsilon_{i,n}$, $\epsilon_{i,n} \sim \mathcal{N}(0, {\sigma_i}^2)$.

Šiuo atveju, kiekvieną komandos pajėgumą apibūdina parametras β_i ir paklaidų komponentė, kurios dideli svyravimai galėtų būti interpretuojami kaip nestabilaus žaidimo įrodymas ar bent jau pagrįsta dvejonė komandos pasirodymo galimybėmis, žaidimo nenuspėjimumu. Tagi galutinai modelį būtų galima užrašyti į pradinę išraišką, žinant, kad paklaidos turi heteroskedastiškumo problemą, t.y, iš turimos lygties 6:

$$\begin{aligned} y_{i,j,n} = \lambda + \alpha_2(x_{2_{i,n}} - x_{2_{j,n}}) + \alpha_3(x_{3_{i,n}} - x_{2_{j,n}}) + S_{i,n} - S_{j,n} + \epsilon_{i,j,n} = \\ \lambda + \alpha_2(x_{2_{i,n}} - x_{2_{j,n}}) + \alpha_3(x_{3_{i,n}} - x_{3_{j,n}}) + \beta_i - \beta_j + \epsilon_{i,n} - \epsilon_{j,n} = \\ \lambda + \alpha_2(x_{2_{i,n}} - x_{2_{j,n}}) + \alpha_3(x_{3_{i,n}} - x_{3_{j,n}}) + \beta_i - \beta_j + \epsilon_{i,j,n} \end{aligned} \tag{7}$$

Matricinis žymėjimas lieka toks pat, kaip ir ankstesniuoju atveju 4, tik šįkart $\epsilon \sim \mathcal{N}(0, \Sigma)$, bet $\Sigma \neq \sigma^2 I$, t.y., matrica Σ diagonali- įstrižainėje turinti $({\sigma_i}^2 + {\sigma_j}^2)$ narius.

Jei mūsų nagrinėjamame modelį yra heteroskedastiškumo problema, tai visgi lieka klausimas, kaip ją išspręsime? Toli ir ilgai ieškoti neteks, kadangi mums į pagalbą ateina ekonometrijoje dažnai sutinkami didžiausio tikėtinumo ir apibendrinto tiesinio modelio metodai. Didžiausio tikėtinumo metodas pritaikomas įprastai- kadangi paklaidos yra nepriklausomi vienodai pasiskirstę atsitiktiniai dydžiai (remiamės viena iš modelio sudarymo prielaidų), o apibendrinto atveju, reikia atlikti keletą žingsnių- prikabinant kintamiesiams svorius, kurie išspręstų heteroskedastiškumo problemą, taigi kalba eina apie tokią žingsnių seką:

- iš 5 lygties mažiausių kvadratų metodo būdų randame modelio liekanas ϵ;
- įsivertiname naują liekanų kvadrato modelį (žr. 8);
- gauto parametro vektoriaus $\hat{\phi}$ dispersija nusako komandos pajėgumo kintamumo charakteristiką;
- šio modelio įverčiai nusako mūsų ieškomą svorių matricą $\hat{\Sigma}$;
- turėdami svorių matricą $\hat{\Sigma}$, perskaičiuojame pradinio modelio parametrų įvertinius (žr. 9).

$$\hat{\epsilon}^2 = V\phi + \nu \tag{8}$$

, čia V- yra X matrica be pirmųjų dviejų stulpelių, t.y., $N \times 30$ matrica

$$\beta_{GLS} = (X^T \hat{\Sigma}^{-1} X)^{-1} X^T \hat{\Sigma}^{-1} y \tag{9}$$

Lieka neatsakytas klausimas: ar nagrinėjamame modelyje turime heteroskedastiškumo problemą? Į klausimą padės atsakyti hipotezės tikrinimas, t.y.:

$$H_{0,hom.} : \sigma_i = \sigma_j \forall i, j \tag{10}$$

.

Homogeniškumą tikrinsime pasinaudoję didžiausio tikėtinumo metodu, t.y., modelių logaritmuotą tikėtinumą (angl. log-likelihood) iš 6 modelio ir modelio neturinčio (pagal prielaidas) heteroskedastiškumo problemos (žr. 5) atvejais, kitaip tariant, atliksime tikėtinumo santykio (angl. likelihood ratio) testą LR. Pirmu atveju, gautą statistiką žymėsime LL_1 (priėmus prielaidą apie heteroskedastiškumą) ir LL_0- antru atveju, gautus rezultatus palyginsime analizuodami skirtumą tarp šių dviejų reikšmių, t.y.:

$$LR = 2(LL_1 - LL_0) \tag{11}$$

, kuris yra pasiskirstęs pagal χ^2 skirstinį su $30 - 1 = 29$ laisvės laipsniais nulinės 10 hipotezės atveju.

Atlikus tikėtinumo santykio testą pasezoniui, lyginant 2010-2016 metų sukauptą informaciją, tik sykį buvo atmesta nulinė hipotezė apie paklaidų homogeniškumą su reikšmingumo lygmeniu $\alpha = 0.05$.

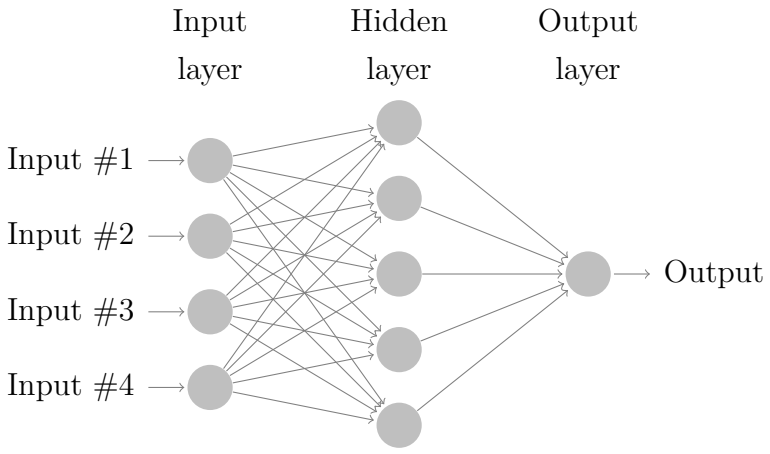
2.2 Neuroninių tinklų modelis

2.2.1 Modelio apibrėžimas

Neuroniniai tinklai gelbsti daugelyje situacijų- klasifikavimo, klasterizavimo, prognozės problemoms spręsti. Regresijos analizėje dirbtiniai neuroniniai tinklai skirti daugiasluoksnių perceptronų apmokymui tam, kad aproksimuotų funkcinius ryšius tarp kovariančių ir atsako kintamųjų (šiuo atveju, atsako kintamojo- rungtynių baigčių rezultato)- taigi šiame kontekste neuroninius tinklus galime įsivaizduoti kaip apibendrintą tiesinio modelio išplėtimą.

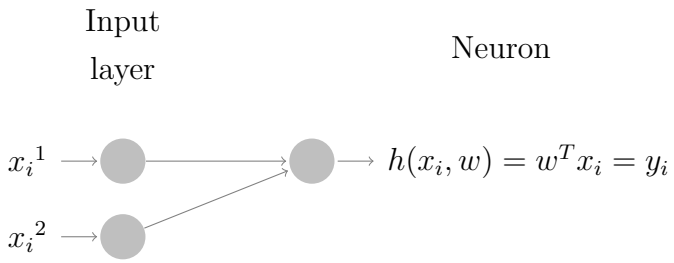
Kalbant bendrai, neuroninių tinklų terminologijos statistikoje kilmė yra glaudžiai susijusi su biologiniais procesais, tiksliau- smegenų ląstelėmis, neuronais, kurie priimdami įvairius signalus atlieka tam tikrą aktyvavimo procesą- veikiama išorinių veiksnių ir priklausomai nuo jų suformuoja ir siunčia signalus kitoms ląstelėms. Tokiu būdu, siunčiami ir gaunami signalai per neuronus įgauna tikslesnius svorius, nuo kurių priklauso galutinis smegenų inicijuotas sprendimas organizmui veikti, atlikti tam tikrą funkciją- visas šis komplikuotas smegenų funkcionavimo procesas supraprastintoje versijoje gali būti atvaizduojamas grafiškai (žr. pav. 1), kuris yra ne kas kitas kaip dirbtinių neuroninių tinklų modelio pavyzdys. Visgi tolimesniuose aptarimuose vengsime sąsajų su smegenų neuronais ir turėsime omenyje tik apie statistiko požiūriu suprantamus "dirbtinius" neuronus.

1 pav. Dirbtinių neuroninių tinklų pavyzdys



Prisiminę aptartas tiesinio modelio lygtis (žr. lygtis 1, 2), apibrėšime neuroninius tinklus tiesinei regresijai modeliuoti. Iš tiesų, elementariausias (bet vis tiek gana painus) neuroninių tinklų modelis remiasi mažiausių kvadratų regresija, kur kiekvienas neuronas naudojasi tiesine aktyvavimo funkcija, t.y.,

2 pav. Dirbtinių neuroninių tinklų pavyzdys



Toks neuronų tinklas priima x^1, x^2 reikšmes $(x_i{}^1, x_i{}^2)$, kurioms priskiria svorius w_1, w_2 ir susumuoja jas, susumavus siunčiamas signalas, dar vadinama prognoze. Tokiu būdu galime apibrėžti ir sudėtingesnį tinklą, susidedantį iš daugiau kintamųjų (tarkime, j- kintamųjų), tuomet vietoj parametrų w_1, w_2 stebėtume $w_1, \ldots, w_j$ svorių.
Įvertinimui, kaip tinklas veikia, galime pasinaudoti ir aukščiau aptartomis mažiausių kvadratų liekanų funkciją (žr. lygtį 2), kuri sumuoja kvadratines modelio liekanas panaudojus visus apmokymui skirtus duomenis.

Tuomet panaudosime gradientinio nusileidimu (angl. gradient descent), apibrėžtai kvadratinių liekanų funkcijai (dar kitaip vadinamai praradimo funkcija (angl. loss function)) $L(w)$, ir kurią žymėsime $\bigtriangledown_w L(w)$, tam, kad minimizuotume bendras apmokymo duomenų liekanas. Minimizavimas remiasi išvestinių skaičiavimu atitinkamai atsižvelgiant į svorius $w_{j\to k}$ (čia taip žymėsime perėjimą iš j-ojo į k-tąjį neuroną), bendresnė išraiška turės tokį pavidalą:

$$\begin{gathered}\frac{\partial}{\partial w_{j\to k}} L(w) = \frac{\partial}{j \to k} \Sigma_i \left(h(x_i, w) - y_i\right)^2 \\ \Sigma_i \frac{\partial}{j \to k} \left(h(x_i, w) - y_i\right)^2 = \\ \Sigma_i 2(h(x_i, w) - y_i) \frac{\partial}{j \to k} h(x_i, w)\end{gathered} \tag{12}$$

Minimizavus liekanas, svoriai yra atnaujinami naudojant standratiniu gradientiniu nusileidimu, t.y.:

$$w = w - \gamma \bigtriangledown_w L(w) \tag{13}$$

, čia γ parenkamas parametras.

Po sudėtingų ir ilgų skaičiavimų galų gale randamas geriausias modelio variantas.

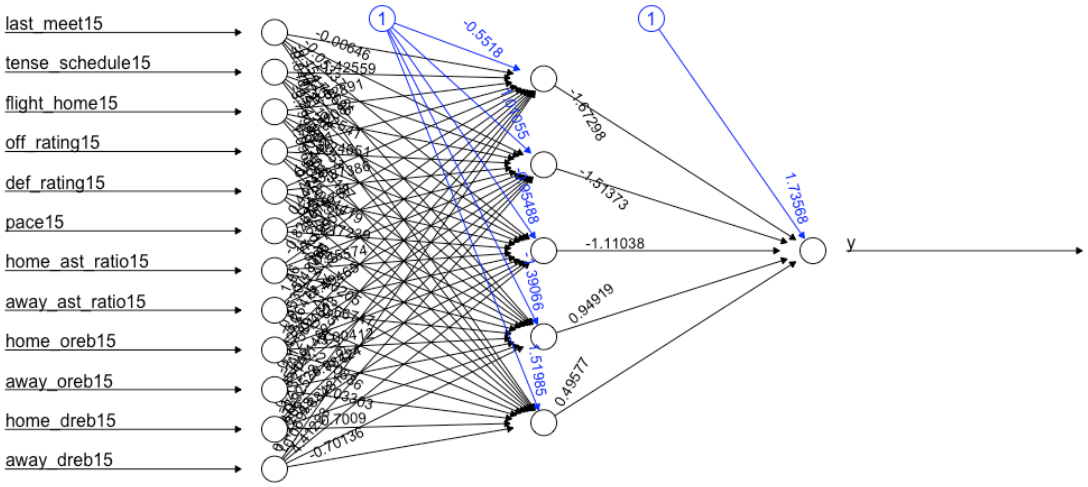
2.2.2 Neuroninių tinklų pritaikymas

Prieš pradėdami taikyti dirbtinių neuroninių tinklų modelį, reikia atlikti tam tikrus pakeitimus- paspartinančius funkcinių ryšių aproksimacijų skaičiavimus (kadangi neuroniniai tinklų modelyje parametrai nėra lengvai konfiguruojami ir apmokomi).
Pirmiausia, apdorosime duomenims taip, kad jie būtų tinkamesni mūsų sudaromam mo-

deliui. Daugelyje sutinkamų neuroninių tinklų modelių taikyme yra siūloma normalizuoti atsako kintamuosius pries apmokant neuroninius tinklus. Tai gana svarbus žingsnis dėl kelių priežasčių: duomenų nenormalizavimas gali turėti įtakos apmokymo procesui ir netgi jį dar labiau pasunkinti (priminsime, kad neuroninių tinklų apmokymas ir taip nėra paprastas procesas), kadangi daugeliu atveju algoritmas gali nespėti konverguoti pasibaigus nustatytam iteracijų skaičiui. Pasirinkome vadinamą minimumo-maksimumo (angl. min-max) normavimo būdą, kuris duomenis sukelia į $[0, 1]$ intervalą- dažniausiai toks normavimo būdas duoda geresnius rezultatus.

Neuroninių tinklų modeliavime nėra griežtos taisykles, nusakančios kiek paslėptų sluoksnių (angl. hidden layers) ir neuronų reikia parinkti, nors, savaime aišku, yra keletas priimtų nykščio taisyklių (angl. rules of thumb). Bet dažniausiai- vienas sluoksnis yra pakankamas daugumai neuroninių tinklų modelių pritaikymui. Neuronų skaičius priklauso nuo nepriklausomų kintamųjų skaičiaus ir išvesties (kiek rezultatų grąžins neuroninių tinklų modelis). Įprastai naudojamasi nerašyta taisykle- neuronų skaičius atitinka turimų nepriklausomų kintamųjų 2/3 dalį. Bet šiame darbe testuosime daugiau nei kelis atvejus (parinkę skirtingą neuronų skaičių) ir ieškosime priimtiniausio sprendimo, kadangi neturime jokių garantijų, jog praktikoje naudojamos taisyklės geriausiai tiks ir mūsų nagrinėjamam modeliui. Todėl testuojant modelį išbandyti įvairūs scenarijai (išnagrinėti atvejai, kai $l \in \{1, 2, 3\}$, o $nn \in \{3, \ldots, 7\}$, čia l žymime sluoksnių, o nn- neuronų skaičius). Pagal mažiausią MSE reikšmę geriausias pasirodė modelis su $l = 1$ ir $nn = 5$.

3 pav. Geriausias NN modelis



Ko gero, žvelgiant į grafiką (žr. pav. 3) daug pasakyti apie modelio pritaikomumą ir svorius negalime. Tačiau užtenka paminėti, kad apmokymo algoritmas konvergavo ir todėl turimas modelis paruoštas tolimesniam naudojimui- t.y., prognozei.

Dabar galime bandyti prognozuoti krepšinio rungtynių baigčių rezultatus testavimo duomenų aibei ir paskaičiuoti MSE. Būtent kitame skyriuje ir apžvelgsime rezultatus lyginant su tiesiniais modeliais.

3 Modelio parinkimas, prognozė

Nors heteroskedastiškumo testas sufleruoja apie liekanų homogeniškumą (daugeliu atveju), prognozę sudarysime naudodami visais metodais. Gautus prognozės rezultatus lyginsime standartiniais modelio gerumo būdais- MSE, MAE reikšmėmis. Prognozuosime 2015-2016, taip pat ir 2016-17 metų sezono rungtynių baigtis. Kadangi 2016-17 metų sezonas dar net nespėjo įpusėti, komandos sužaidė kiek mažiau nei po 40 rungtynių (priminsime, kad sezonas susideda iš 41 rungtynių namuose ir 41 rungtynių išvykoje) šiame sezone sukaupėme informacijos iki šių metų pirmos dienos, tai turės įtakos prognozių rezultatams, nes imties dydis modelio parametrų įvertinių radimui ir skaičiavimams (angl. in-sample) nebus didesnis nei 20 rungtynių vienai komandai (kalbame apie 200 pirmų rungtynių, iš kurių spręsime apie likusių rungtynių nugalėtojus). Skamba gal kiek komplikuotai, bet visgi 2016-17 metų sezonas iš visų labiausiai intriguoja.

7 lentelė. 2015-16 sezonas

	OLS	GLS	MLE
λ	2.46	2.438	2.49
	(0.34)	(0.36)	(0.34)
α_2	-1.21	-1.48	-1.22
	(0.53)	(0.61)	(0.59)
α_3	0.45	0.401	0.46
	(0.23)	(0.19)	(0.22)

$$MAE = \Sigma_{n=1}^{N^*} |y_{i,j,n} - \hat{y}_{i,j,n}| \tag{14}$$

, čia N^* prognozuojama sezono dalis (angl. out-of-sample).

Tam, kad surastume geriausią modelį, reikia patikimų, daug kur sutinkamų modelio tinkamumo rodiklius- būtent tokie MAE ir MSE mums ir pasirodė. Prognozę 2015-16 metų sezonui atliksime tokiu principu: duomenis skelsime į dvi lygias dalis (po 615 rungtynes), viena įvertiniams surasti, likusi- tikrinti prognozės tikslumą, taigi pirmiausia įvertinsime parametrų įvertinių $\widehat{\lambda}, \widehat{x1}, \widehat{x2}, \widehat{\beta_i} \forall i \in \{1, \ldots, 30\}$ reikšmes iš pirmo sezono dalies (pirmų 615 rungtynių), tada turėdami parametrų įverčius, likusioms 615 rungtynėms bandysime nuspėti taškų skirtumą tarp komandų. Modelio metodus palyginsime MAE (žr. 14) ir MSE (žr. 15) rodikliais.

$$MSE = \Sigma_{n=1}^{N^*} \left(y_{i,j,n} - \hat{y}_{i,j,n}\right)^2 \tag{15}$$

, čia N^* prognozuojama sezono dalis (angl. out-of-sample).

3.1 Modelių analizė

8 lentelė. 2015-16 sezono komandų reitingavimas

$\widehat{\beta_{OLS}}$	$\widehat{\beta_{MLE}}$	$\widehat{\beta_{GLS}}$	Team
14.59	17.38	15.16	Spurs
14.86	15.01	15.13	Warriors
11.29	11.59	11.71	Thunder
9.79	10.52	9.67	Cavaliers
8.37	7.92	8.10	Raptors
8.37	7.95	7.77	Clippers
7.61	6.34	7.29	Hawks
6.98	7.57	6.69	Celtics
6.43	4.76	6.62	Hornets
5.95	4.60	5.81	Jazz
5.70	5.05	5.66	Heat
5.00	2.78	5.26	Trail Blazers
4.46	6.24	5.22	Pistons
5.63	7.22	4.92	Pacers
4.53	3.33	4.90	Rockets
4.08	4.25	4.21	Mavericks
3.60	2.97	3.34	Wizards
2.67	5.19	2.65	Bulls
1.70	2.42	2.03	Kings
2.14	2.81	1.95	Grizzlies
2.28	3.43	1.87	Magic
1.18	0.89	1.35	Nuggets
0.53	0.01	1.02	Timberwolves
0.45	0.94	0.83	Pelicans
1.10	2.69	0.75	Knicks
0.00	0.00	0.00	Bucks
-2.41	−0.25	−2.18	Suns
-3.21	−2.06	−3.30	Nets
-5.08	−5.39	−4.92	Lakers
-6.08	−6.59	−5.60	Sixers

Iš lentelės matyti, kad modeliai pagal įvertintą stiprumą ganėtinai gerai sureitinguoja ekipas-stipriausios 2015-2016 metų NBA komandos rikiuojasi lentelės viršuje. Nors čempionų žiedus praeitame sezone matavosi Cleveland "Cavaliers" krepšininkai (pripažinsime, gal kiek ir netikėtas rezultatas), bet sezono metu neabejotinai visa galva pranašesnės buvo San Antonio "Spurs" ir Golden State "Warriors" komandos, kurios išlaikė komandinį, stabilų žaidimą

viso sezono metu. Priminsime, kad "Spurs" komandai priklauso namuose iš eilės iškovotų pergalių skaičius- San Antonijaus krepšininkai, turėdami aikštės pranašumą, juo pasinaudojo ir varė varžovus į neviltį net 40 kartų iš 41 (namuose nusileista tik kartą- pakartotas geriausias NBA rezultatas žaidžiant namų aikštėje). "Warriors" dvyliktukas nustebino net ir didžiausius ekspertus- pasiekdami 73 pergales (iš 82 rungtynių)- ko iki tol nepavyko padaryti nei vienai komandai (net ir Michealo Jordano Chicago "Bulls" superkomandai). Taigi nieko neturėtų stebinti, kad komandos reitingavime dalinasi pirmąsias vietas, o lentelės dugne- dvi prasčiausią rezultatą parodžiusios komandos- Los Angeles "Lakers" ir Philadelphia "76ers".

Iš lentelės (žr. lentelę 7) matome, kokie namų aikštės pranašumo, λ, intensyvaus tvarkaraščio, α_2, praeitos akistatos reikšmė dabartinei dvikovai, α_1, veiksnių įvertiniai taikytais metodais. Visais metodais- gauti įvertiniai pernelyg nesiskiria.

9 lentelė. 2015-16 sezono modelių palyginimas

	OLS	MLE	GLS
MAE	9.262 703	9.262 717	9.292 119
MSE	141.908 44	141.906 425	143.661 495

10 lentelė. 2015-16 sezono OLS ir NN

	OLS	NN
MSE	119.61	117.52

Iš lentelės (žr. lentelę 10) matyti, kad NN (neuroninių tinklų) modelis lenkia OLS (mažiausių kvadratų) modelį. Tačiau skirtumas nėra ženklus, didelis.

3.2 Komandos tikimybė laimėti

Gana nesunkiai galime ne tik sudarinėti prognozės modelius, bet ir apskaičiuoti tikimybes, kokie šansai, kad i komanda laimės, žinant ar komanda žaidžia namuose ar išvykoje, ar abi komandos turi intensyvų tvarkaraštį ir kaip sekėsi praeitoje varžovų akistatoje (žr. 16).

$$
\begin{aligned}
P\left(\lambda + \alpha_2(x_{2_{i,n}} - x_{2_{j,n}}) + \alpha_3(x_{3_{i,n}} - x_{3_{j,n}}) + \beta_i - \beta_j > \epsilon_{i,j,n}\right) = \\
P\left(\lambda + \alpha_2(x_{2_{i,n}} - x_{2_{j,n}}) + \alpha_3(x_{3_{i,n}} - x_{3_{j,n}}) + \beta_i - \beta_j > \epsilon_{i,n} - \epsilon_{i,n}\right) = \\
P\left(\frac{\lambda + \alpha_2(x_{2_{i,n}} - x_{2_{j,n}}) + \alpha_3(x_{3_{i,n}} - x_{3_{j,n}}) + \beta_i - \beta_j}{\sqrt{\sigma_i^2 + \sigma_j^2}} > \frac{\epsilon_{i,n} - \epsilon_{i,n}}{\sqrt{\sigma_i^2 + \sigma_j^2}}\right) = \\
\Phi\left(\frac{\lambda + \alpha_2(x_{2_{i,n}} - x_{2_{j,n}}) + \alpha_3(x_{3_{i,n}} - x_{3_{j,n}}) + \beta_i - \beta_j}{\sqrt{\sigma_i^2 + \sigma_j^2}}\right)
\end{aligned}
\tag{16}
$$

, čia Φ- standartinio normalaus kumuliatyvinė pasiskirstymo funkcija.

Dabar paskaičiuosime 2015-2016 metų komandų šansus laimėti, panagrinėsime atvejus komandų, kurių pajėgumas apylygis ir atvejus tarp aukštai išreitinguotų komandų ir vidutiniokų.

Iš tiesų pažvelgsime, kokios tikymybės komandai nugalėti vieną kitą, atsižvelgiant į tiesinio modelio komandų pajėgumus, ekipas atrinkome pagal pajėgumų lentelę (žr. lentelę 8).

11 lentelė. 2015-16 sezono tikimybės laimėti

	Komanda1	Komanda2	$P_{1@2}$	$P_{2@1}$
	"Hawks"	"Pistons"	0.47	0.75
α_2	1	0	0.59	0.6
	0	1	0.33	0.81
	"Wizards"	"Knicks"	0.54	0.61
α_3	1	0	0.63	0.53
	0	1	0.51	0.64

3.3 Apibendrinimas

Taigi palyginus modelius, išsiaiškinome, kad visais atvejais jie pernelyg nesiskiria (žvelgiant pagal MAE, MSE rodiklius). Pasakyti, kuriuo modeliu naudotis, kuris geriausiai prognozuoja pergales- ganėtinai kėblus klausimas. Pagal rezultatus, neuroninių tinklų modelis geriausias, tačiau tiesinės regresijos modelius interpretuoti yra gerokai lengviau. Prieš apsisprendžiant, reiktų išsiaiškinti, ar neturime heteroskedastiškumo problemos, jei ne- ko gero, teisingiausia rinktis paprasčiausią, bet ne prasčiausią- mažiausių kvadratų metodą, kuris duoda gana gerus įvertinius. Tolimesnėje analizėje siūloma neapsiriboti vienu modeliu ir bandyti kelis variantus iškart, susipažinus su NBA duomenų gausa, modelius būtų galima dar išplėsti papildžius naujom kovariantėmis, tiksliau nusakančiomis ne vien komandos puolimo galią, bet ir darbą gynyboje.

Išvados

NBA rezultatų baigčių prognozavimas- sudėtingas dalykas, kadangi nugalėtoją gali lemti daug smulkių detalių, kiekvienos rungtynės gali pateikti staigmenų ir lygos lyderiai bet kada gali pralaimėti prieš autsaiderius- tuo sportas ir žavus, savo nenuspėjamumu. Šiame darbe bandyta pakovoti su šiuo iššūkiu- ieškant ir konstruojant naujas kovariantes. Surinkus ir susisteminus 2010-2016 sezono duomenys, sukurtas funkcionalumas, leidžiantis kasdien išsitraukti informaciją apie rungtynes, žaidėjus, galimai naudingus statistinei analizei, rungtynių baigtims pritaikyti tiesiniai modeliai ir apskaičiuoti komandos pajėgumo, nuovargio faktoriaus įverčiai skirtingais būdais- mažiausių kvadratų, apibendrintų mažiausių kvadratų ir didžiausio tikėtinumo metodais, iš kurių geriausias buvo lygintas su neuroniniu tinklų modeliu, abiem atvejais sudarytos 2015-2016 sezono rezultatų prognozės, kurių rezultatai rodė neženklų neuroninių tinklų pranašumą.

Literatūra

[1] Renato Amorim Torres *Prediction of NBA games based on Machine Learning Methods,"* 2013.

[2] Colin F. Camerer *Does the Basketball Market Believe in the 'Hot Hand'?* 1989.

[3] Mark E. Glickman, Hal S. Stern *A state-space model for National Football League scores.*

[4] Hans Manner *Modeling and forecasting the outcomes of NBA basketball games* 2015.

[5] Michael J. Bailey *Predicting sporting outcomes: a statistical approach* 2005.

[6] Alexander Dubbs *Statstics-Free Sports Prediction* 2016.

[7] Bryan Cheng, Kevin Dade, Michael Lipman, Cody Mills *Predicting the Betting Line in NBA Games* 2014.

[8] J. Scott Armstrong *Predicting Job Performance: The Moneyball Factor* 2012.

[9] Mark E. Glickman, Hal S. Stern *Estimating team strength in the NFL* 2016.