Универзитет у Београду

Електротехнички факултет



Софтверско инжењерство великих база података

Извештај пројектног задатка

|  |  |
| --- | --- |
| Ментор: | Кандидат: |
| ас. мс Стефан Тубић, дипл. инж. ел. и рач. | Душан Крџић 2021/3299 |

Београд, Јануар 2023.

Садржај

[Садржај 2](#_Toc442888948)

[1. Увод 3](#_Toc442888949)

[2. Опис базе 4](#_Toc442888950)

[3. Циљ (задатаk пројекта) 6](#_Toc442888951)

[4. Анализа базе података и карактеристика 7](#_Toc442888960)

[5. Алгоритми за учење и предикцију 21](#_Toc442888960)

[6. коначни резултат 26](#_Toc442888960)

[5. закључак 27](#_Toc442888960)

[Литература 3](#_Toc442888974)

1. Увод

Овај документ представља извештај на одрађени пројекат из предмета Софтверско инжињерство великих база података на мастер студијама Електротехничког факултета на смеру Софтверско инжињерство.

Пројекат је писан на програмском језику Пајтон у Јупитер окружењу.

Тема пројекта је ,,Ceлекција шутева Кобија Брајанта”. Циљ пројекта је предвидети успешност шута (погодак или промашај) Кобија Брајанта на основу различитих парамерата (тип шута, одакле је шут изведен тј позиција на терену, временских одредница итд.).

База података је са сајта *Kaggle* [*Kobe Bryant Shot Selection*](https://www.kaggle.com/competitions/kobe-bryant-shot-selection).

1. Опис базе

У оквиру текста самог проблема, постоје подаци. База података се састоји од укупно 30697 података (података), подељених у 25 карактеристика (колона).

Подаци за анализу:

1. *action\_type* – тип акције којом је Коби шутнуо на кош. У питању је текстуални податак. У бази постоје укупно 57 различитих типова акција.
2. *combined\_shot\_type* – комбиновани тип шута. Текстуални податак, укупно 6 различитих типова.
3. *game\_event\_id* – Идентификација догађаја. Целобројни тип. Укупно 620 различитих догађаја.
4. *game\_id* – Идентификација меча. Целобројни тип. Укупно 1559 различитих утакмица.
5. *lat* – Географска ширина са које је направљен шут. Децимални тип.
6. *loc\_x* – Позиција на x oси где је направљен шут, где координатни систем представља сам терен, са координатним почетком (0,0) на месту коша. Целобројни тип.
7. *loc\_y -* Позиција на y oси где је направљен шут, где координатни систем представља сам терен, са координатним почетком (0,0) на месту коша. Целобројни тип.
8. *lon* – Географска дужина где је направљен шут. Децимални тип
9. *minutes\_remaining –* У ком минуту до краја меча је направљен шут. Целобројни тип.
10. *period* – Период меча у ком је направљен шут. Целобројна вредност.
11. *playoffs* – Плајоф. Целобројна вредност. Две вредности 0 и 1.
12. *season* – Сезона. Текстуални податак. 20 различитих сезона.
13. *seconds\_remaining –* Секунде до краја меча када је направљен шут. Целобројни тип.
14. *shot\_distance* – Дистанца од коша одакле је направљен шут. Целобројни тип.
15. *shot\_made\_flag* – Циљна карактеристика задатка. 3 вредности. 0 – промашај, 1 – погодак, NaN – неодређено. Циљ задатка јесте одредити за NaN вредности вероватноћу поготка на основу осталих карактеристика.
16. *shot\_type* – тип шута. Тесктуални података, 2 вредности.
17. *shot\_zone\_area –* локација шута на терену. Текстуални податак, 6 различитих вредности.
18. *shot\_zone\_basic* -локација шута на терену. Текстуални податак, 7 различитих вредности.
19. *shot\_zone\_range* *-* локација шута на терену. Текстуални податак, 5 различитих вредности.
20. *team\_id –* Идентификација тима за који је Коби играо кад је направљен шут. Једеинствена целобројна вредност јер је Коби играо само за један тим.
21. *team\_name* – Име тима за који је Коби играо кад је направљен шут. Јединствени текстуални податак јер је Коби играо само за један Тип ЛА Лајкерси.
22. *game\_date* – Датум кад је направљен шут. Састоји се из информације о месецу и години.
23. *matchup* – Податак који се састоји из тима за који је Коби играо, противничког тима и кодном знаку. Два кодна знака *@* и *vs*. Први значи да је утакмица играна на противничком терену, док други значи да је играна на домаћем.
24. *opponent* – Противник. Текстуални податак. 33 различита противника.
25. *shot\_id* – Идентификација шута, целобројни податак.

Од 30697 података, 5000 података има неодређену карактеристику да ли је постигнут погодак или није. 25697 података је потпуно одређено, и до њих треба направити модел на основу ког ће алгоритам научити правилности и направити предикцију.

1. Циљ (Задатак пројекта)

Задатак пројекта јесте да на основу потпуних података (редови којима је *shot\_made\_flag* одређен, тј. 0 за промашај и 1 за погодак), алгоритам „научи“ како су одређене карактеристике (колоне) утицале на успешност поготка, и то „научено“ примени да предвиди вероватноћу поготка за оне колоне којима је *shot\_made\_flag* неодређен,

Како би се предикција што успешније извршила било је потребно „очистити“ сувишне податке који не утичу значајно или уопште на успешност поготка, и оне које утичу трансформисати како би алгоритам лакше направила предикцију.

1. анализа базе података и карактеристика

Анализираћемо сваку карактеристику појединачно и видети њихов утицај на колону *shot\_made\_flag,* тј. како која карактеристика утиче на проценат успешности шута, тј. погодак.

Карактеристике се посматрају из базе без *NaN* вредности, јер за *NaN* вредности тек треба претпоставити утицај и треба избећи њихов утицај на анализу. Дакле за анализу се посматрају само они редови у којима карактеристика *shot\_made\_flag* има вредности различиту од *NaN*, тј. 0 или 1.

* 1. action\_type

Представићемо утицај карактеристике *action\_type* на проценат успешности шута, тј. колико је одређени тип акције процентуално успешан.

Chart, bar chart, histogram

Description automatically generated

слика 4.1 график успешности поједине акције

Са графика се види да тип акције утиче на успешност шута. Док поједини шутеви дају велики проценат успешности. неки и нису толико успешни, равни нули. Зато дата колона ће ући у анализу.

* 1. combined\_shot\_type

Сличан поступак као 4.1

Chart, bar chart

Description automatically generated

слика 4.2 график зависност успешности шута од комбинованог типа шута

Са графика се види утицај ове карактеристике. Треба ући у анализу.

* 1. game\_event\_id, game\_id

Дате карактеристике представљају целобројне бројеве идентификације утакмице. Оне саме немају утицај на успешност шута и треба их ибацити из анализе.

* 1. lat, lon, loc\_x, loc\_y

Дате карактеристике представљају локацију шута у односу на кош*. lat, lon* представља један пар географских координата, док *loc\_x. loc\_y* представља други пар. За анлизу није потребно узети оба, јер се довољна информација може извући из само једног пара. Узећемо пар *loc\_x. loc\_y* .

A picture containing text

Description automatically generated

слика 4.4 позиција поготка и промашаја на терену. 0,0 је позиција коша

Најгушћи број погодака је у близини коша. Занимљиво је да је у близини коша и већи број промашаја, као и са позиције „тројка“. Тако да ова карактеристика има велики утицај на успешност поготка.

* 1. време до краја – minutes\_remaining, seconds remaining

Chart, line chart

Description automatically generated

слика 4.5.1 зависнот успешности шута од минута до краја меча

Text

Description automatically generated

слика 4.5.2 зависнот успешности шута од секунди до краја

Са графика се види велика варијабилност успешности шута у току времена. У последњих 3 секунди се види значајан пад перформанци. Дакле дате карактеристике ћемо спојити у једну, број секунди до краја, и узећемо само последњих 5 секунди.

* 1. period

Chart, line chart

Description automatically generated

слика 4.6 зависност успешности шута од периода меча

График успешности шута је доста променљив за ову карактеристику, тако да ћемо је уврстити у анализу.

* 1. playoff

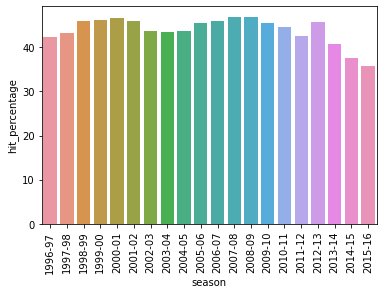
Chart

Description automatically generated

слика 4.7 график завицности успешности шута од плејофа

Са графика се види да обе вредности плејофа дају проближно исту успешност шута. Тако да ова карактеристика није пресудна за успешност. Можемо је иѕбацити из анализе.

* 1. season



слика 4.8 зависност успешности шута од сезоне

* 1. shot\_distance

Graphical user interface, text

Description automatically generated

слика 4.9 зависност успешности шута од удаљености од коша

Са графика се види да удаљеност од коша утиче на успешност шута. Што је шут ближи кошу већа је вероватноћа да буде успешан. Треба уврстити у анализу.

* 1. shot\_type

Chart

Description automatically generated

слика 4.10 зависност успешности шута од типа шута

Тип шута зависи на успешност шута. „Плави“ шут је успешнији од „наранџастог“. Ову карактеристику ћемо уврстити у анализу.

* 1. shot\_zone\_area, shot\_zone\_basic, shot\_zone\_range

Chart, bar chart

Description automatically generated

слика 4.11.1 зависност успешности шута од позиције на терену

Chart, bar chart

Description automatically generated

слика 4.11.2 зависност успешности шута од позиције на терену

Chart, bar chart

Description automatically generated

слика 4.11.3 зависност успешности шута од позиције на терену

Са графика се види зависнот, успешност зависи од одређене позиције на терену. Треба уврстити ове карактеристике у анализу.

* 1. team\_id, team\_name

Дaте карактеристике су јединствене јер је Коби играо само у једном тиму. Дакле оне се могу занемарити.

**[1610612747] ['Los Angeles Lakers']**

* 1. game\_date

Из дате карактеристике се може извући година и месец меча у ком је постигнут конкретан шут.

Chart, line chart

Description automatically generated

слика 4.13.1 зависност успешности шута од године

Chart, line chart

Description automatically generated

слика 4.13.2 зависност успешности шута од месеца

Делује да месец и година делује на успешност шута. Дакле трансформисати дати податак *game\_date* у месец и годину.

* 1. matchup

Из дате карактеристике се може извући противничка екипа и да ли се игра на домаћем или противничком терену. Сам противник овде није од интереса јер постоји посебна колона за то. Овде је од интереса да ли се игра у гостима или не. Дакле дати податак ћемо прво кодирати – vs => 1 (домаћи терен), @ => 0 (противнички терен), и видети како то утиче на успешност шута.

Chart

Description automatically generated

слика 4.14. зависност успешности шута од типа терена 0 (гости), 1 (кући)

* 1. opponent

Chart

Description automatically generated with medium confidence

слика 4.15 график зависности успешности шута од противника

Са графика се види да противник не утиче толико на успешност шута. Нема великих колебања у вредности. Дакле пробаћемо да ову карактеристику занемаримо. Све вредности су између 40%-50% за успешан шут.

* 1. shot\_id

Дата карактеристика не утиче на успешност шута, то је само идентификација шута.

* 1. shot\_made\_flag

Дата карактеристика нам је циљна карактеристика, дакле нећемо је узимати за анализу.

1. Алгоритми за учење и предикцију

Пре примене самих алгоритама на базу података треба прво „очистити“ базу од сувишних податка (података који не утичу на успешност шута) или трансформисати податке тако да алгоритам изврши лакшу предикцију (закључено у претходном поглављу које су то карактеристике).

Затим податке треба поделити у две групе. Прва је тест група, од које ће алгоритам учити правилности на основу свих карактеристика, како би направио предикцију. То су они редови у бази којима је колона *shot\_made\_flag* различит од *NaN.* Друга група су резултујући подаци, тј. ред у коме је колона *shot\_made\_flag* једнака *NaN.* Те податке треба алгоритам да попуни након учења над тест подацима, са вероватноћом за успешан шут, тј погодак.

Све податке у бази које су тесктуалног типа, и алгоритам (машина) не може да протумачи треба трансфирмисати у вредност разумљиву алгоритму. За то се користи *LabelEncoder* класа, која тесктуални податак трансформише у вредност разумљиву алгоритму.

***KFold*** je класа која служи да базу података, од које се уче правилности, подели на k скупова, тако што се 1 скуп узме као валидациони тест скуп, а осталих k-1 скупова представљају тренинг скуп. Сам број k зависи од базе до базе, али у теорији се обично узима да буде 5 или 10. У овом примеру ћемо узету 10 скупова у 10 итерација, ради веће прецизности, и коначна процена ће бити средња вредност.

* 1. BOOSTING алгоритми

Дата група алгоритама прави низ модела предикције на основу датог скупа података, где сваки наредни модел покушава да исправи евентуалне грешке у предикцији оног претходног модела у низу.

Процедура се састоји тако што *KFold* класа креира k скуп података над којима алгоритам прави модел предикције. Прави се више слабијих модела предикције из којих се кроз итерације и исправљање грешака праве јачи модели. Процедура се наставља докле год се не нађе оптимални модел са најмање грешака у предикцији.

* + 1. Еxtreme Gradient Boost

Побољшана верзија *Gradient Boost* алгоритма, омогућава паралелизацију и поседује додатне параметре који могу служити за бољу предикцију. Заснива се на стаблима одлучивања.

Резултати:

0.6076413601417615

0.6011794724332357

0.6022798719859119

0.6157184104905128

0.6066564320429253

0.5909227142944277

0.6040324264055369

0.6251442130300652

0.6064740253677422

0.5984867443790118

Mean score = 0.6058535670571131

Time = 34.329468965530396

Дати резултати представљају вероватноћу успешне предикције као и време трајања алгоритма.

* + 1. Gradient Boost

Дати алгоритам је добар за мешовите типове података, међутим проблематичан је јер не омогућава паралелизацију. Заснива се на стаблима одлучивања.

Резултати:

0.6117578221379223

0.6024661221451122

0.6071941610949528

0.6172004358429939

0.6057435113970401

0.5951398691552904

0.609139454111856

0.626940263513912

0.6105946235388928

0.6027961911788123

Mean score = 0.6088972454116784

Time = 216.5245702266693

Дати резултати представљају вероватноћу успешне предикције као и време трајања алгоритма.

* + 1. Ada boost

Заснива се на стаблима одлучивања. Креира модел и свакој карактеристици (колони) из базе података додељује тежину у стаблу одлучивања. На почетку све карактеристике имају исту тежину. Одређеним математичким калкулацијама праве нове тежине за сваку карактеристику. Карактеристике са већим тежинама имају „већу грешку“ и оне ће бити од интереса за следећи модел предикције, који ће бити јачи.

Резултати:

0.6692557297508024

0.6684283995150638

0.6688581541972094

0.6703962739141321

0.6699744278314346

0.666784000143643

0.6692700414017424

0.6715358037571298

0.6699695110907155

0.6683289764583815

Mean score = 0.6692801318060255

Time = 59.72605109214783

Дати резултати представљају вероватноћу успешне предикције, као и време трајања алгоритма.

* 1. Bagging (Bootstrap Aggregation) alogoritmi

Узима узорке из тренинг скупа података и „тренирају“ модел помоћу сваког од купа узорака. Коначни резултат је средња вредност свих подскупова.

* + 1. Bagged Decision Trees

0.7187055545266374

0.6945626763314274

0.694056674657786

0.6951787251471849

0.7328899893511757

0.6826987511516837

0.7241229064439585

0.7648308233456966

0.6925391476418512

0.7074752294761094

Mean score = 0.710706047807351

Time = 463.12094497680664

* + 1. Random Forest

0.7162053126382688

0.6717329159034794

0.6763293462515902

0.6963452349211693

0.6963820280190447

0.6815267308919903

0.698493633831166

0.7648777074792327

0.6793252971581153

0.726864650865944

Mean score = 0.7008082857960001

Time = 267.25720977783203

* + 1. Еxtra Trees

0.7279884273128766

0.70583235578464

0.6770671206843074

0.7170910117053623

0.6739502886596128

0.6670223848267476

0.6858465321341419

0.717893758390157

0.6879644576360422

0.7147635591533994

Mean score = 0.6975419896287287

Time = 271.1591341495514

1. коначни резултат

Резултати у претходном поглављу представљају вероватноћу успешности предикције на основу најрелевантнијег скупа података који смо одабрали из анализе базе података.

Следи сама предикција алгоритма на основу „наученог“. Алгоритам примењује најбољи модел који је генерисао из претходног корака и за сваки неодређени шут даје свој проценат успешности.

Резултати се чувају у посебном фајлу за сваки алгоритам који има идентификацију шута и проценат успешности.

1. Закључак

Сви примењени алгоритми дају сличан проценат поузданости предикције 0,6-0,7. Алгоритми из групе *Bagging (Bootstrap Aggregation)* имају нешто бољи проценат, јер користе мање комплексне алгоритме предикције.

Што се тиче перформанси, *Boosting* алгоритми имају далеко боље перформансе. *Extreme Gradient Boost > Ada Boost > Gradient Boost > Random Forest > Extra Trees > Bagged Decision Trees*. Осетно најбржи алгоритам је свакако *Extreme Gradient Boost* што је и очекивано јер омогућава паралелизацију проблема.

Литература

1. https://www.nvidia.com/en-us/glossary/data-science/xgboost/
2. <https://www.analyticsvidhya.com/blog/2021/09/adaboost-algorithm-a-complete-guide-for-beginners/>
3. <https://www.kaggle.com/competitions/kobe-bryant-shot-selection>