**KAUNO TECHNOLOGIJOS UNIVERSITETAS**

**INFORMATIKOS FAKULTETAS**

Kompiuterių katedra

**Intelektikos pagrindai**

**Projekto ataskaita**

**Atliko:**

Mantvydas Razulevičius IFF – 6/12

Raimonda Rukaitė IFF – 6/12

Tomas Čižauskas IFF – 6/12

**Dėstytojai:**

lekt. BUDNIKAS Germanas

doc. PAULAUSKAITĖ-TARASEVIČIENĖ Agnė

**KAUNAS, 2019**

Turinys

[Bendri nurodymai 3](#_Toc9934089)

[1. **Duomenų surinkimas** (iš tekstinio dokumento) 3](#_Toc9934090)

[2. **Duomenų paruošimas ir valymas** (<pasirinktas sudėtingumas>) 4](#_Toc9934091)

[3. **Dimensijų sumažinimas** (pilnai suprogramuotas) 4](#_Toc9934092)

[4. **Įžanginiai eksperimentai, patikrinantys ar dimensijų sumažinimą verta naudoti** (<pasirinktas sudėtingumas>) 6](#_Toc9934093)

[**5.** **2-ojo mašininio mokymosi metodo su mokytoju panaudojimas(pilnai suprogramuotas metodas)** 7](#_Toc9934094)

[**6.** **Mašininio mokymosi be mokytojo metodo panaudojimas (pilnai suprogramuotas metodas)** 8](#_Toc9934095)

[7. **Mokymosi be mokytojo metodo gautų rezultatų pritaikymas (programoje) / pakomentavimas ataskaitoje** (<pasirinktas sudėtingumas>) 8](#_Toc9934096)

[**8.** **Kryžminė patikra** 8](#_Toc9934097)

[**9.** **Literatūra** 8](#_Toc9934098)

Projekto ataskaitos turinys

# Bendri nurodymai

1. Santrauka

Šiam darbui atlikti naudojome žaidimo „PUBG“ statistiką. Ji aprašyta faile „stats.csv“. šis duomenų rinkinys buvo paimtas iš literatūros sąraše esančios [1] nuorodos.

Problema:

Panaudoti šie mašininio mokymo metodai, bei gauti jų tikslumas:

* k-nearest neighbor – 36%
* backpropagation – y%

1. Komandos narių atsakomybės

|  |  |  |
| --- | --- | --- |
| Vardas, pavardė, užsiėmimo laikas, data | Atsakomybės projekte | Parengti ataskaitos skyriai |
| Mantvydas Razulevičius, Antradienis 15:30 |  | ?? |
| Raimonda Rukaitė, Antradienis 15:30 | ?? |  |
| Tomas Čižauskas, Antradienis 15:30 | ?? |  |

# **Duomenų surinkimas** (iš tekstinio dokumento)

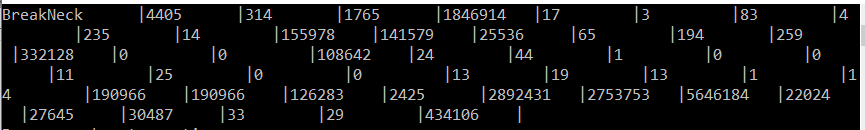
Duomenų failą sudaro 48 stulpeliai duomenų. 47 iš jų skaitiniai, o vienas tekstinis skirtas žaidėjo vardui. Iš viso faile yra 87898 eilučių duomenų.

Duomenų rinkinio trumpas aprašas: paskirtis, **įvesties atributų skaičius**, tikslo klasių/savybių skaičius, **rinkinio dydis.**

Duomenų šaltinis.

Kiekvieno atributo/ klasės reikšmių diapazonas ar reikšmių išvardinimas (išskyrus komentarus).

Pavyzdinis duomenų įrašo pavyzdys.



Pav 1 duomenų pavyzdys

# **Duomenų paruošimas ir valymas** (<pasirinktas sudėtingumas>)

Kadangi mašininio mokymosi modeliui būtina paduoti tik skaitines (*numeric*) reikšmes, šiame skyriuje aprašykite kiekvieno neskaitinio atributo transformaciją (transformacijos taisyklę) į skaitinį formatą. Pateikti duomenų pavydžius prieš transformaciją ir po jos.

Jei naudojama reikšmių normalizacija – pateikti formulę ir duomenų pavyzdį prieš normalizaciją ir po jos.

Pateikti nutolusių reikšmių aptikimo taisykles ir/ar panaudotus metodus. Pateikti duomenų rinkinio fragmentą su pažymėtomis nukrypusiomis reikšmėmis.

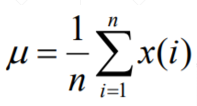
Jei naudojamas reikšmių diapazono keitimas į intervalus (pvz. jei atributas kinta intervale [100, 10000000], atributo reikšmes galime vaizduoti pasirinktais intervalais: 1-asis, 2-asis, ir pan., kur *i*-ojo intervalo diapazonas pasirenkamas atsižvelgiant į duomenis).

# **Dimensijų sumažinimas** (pilnai suprogramuotas)

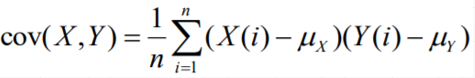
Naudotas esminių komponenčių analizės (PCA) metodas.

Dimensijų sumažinimas mums leidžia atsisakyti nedaug reikšmės pridedančių komponenčių. Sumažinus dimensijas komponentes galime vizualizuoti grafike. PCA yra geras būdas tai padaryti neprarandant daug informacijos. PCA vaizdavimas grafike parodo komponenčių koreliaciją – elementai kurie stipriai korealiuoja yra šalia vienas kito.

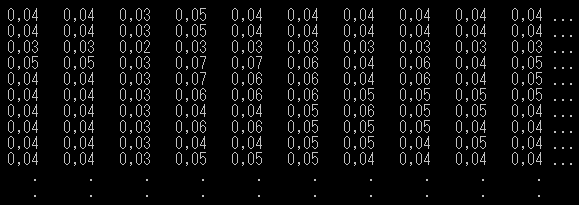
Skaičiuojant PCA komponentes išpradžių suskaičiuojamos duomenų stulpelių vidutinės reikšmės



Tada kiekvienas duomenų elementas pastumiamas per suskaičiuotą vidutinę reikšmė. Toliau apskaičiuojama kovarijacijos matrica.



Gaunama 47 x 47 kovarijacijos matrica nusakanti visų komponenčių kovarijaciją.



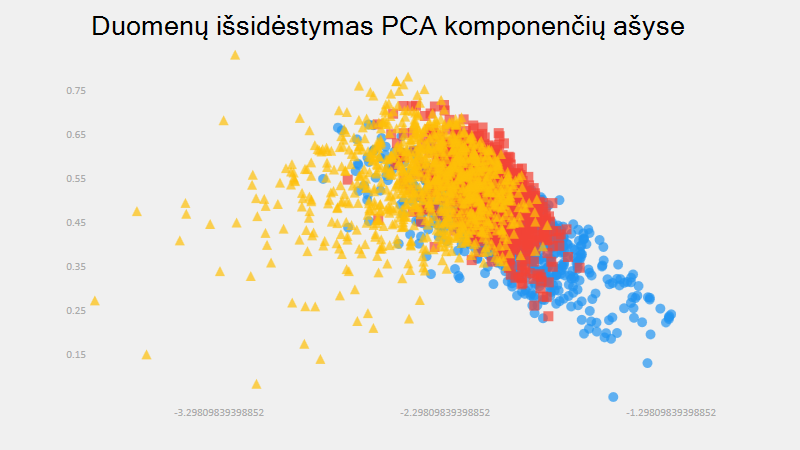
pav. 1 Kovarijacijos matricos ištrauka

Vėliau skaičiuojamos kovarijacijos matricjos tikrinės reikšmės ir tikriniai vektoriai. Kuo didesnė tikrinė reikšmė tuo svarbesnė komponentė.

Lentelė 1 kovarijacijos matricos tikrinės reikšmės

|  |
| --- |
| DenseVector 47-Complex  (1,95431, 0) (0,00202604, 0) (9,48049E-05, 0) (3,8299E-06, 0)  (0,130125, 0) (0,00145954, 0) (0,000111396, 0) (2,21673E-06, 0)  (0,0986652, 0) (0,000869644, 0) (5,75436E-05, 0) (9,73498E-07, 0)  (0,0349396, 0) (0,000777506, 0) (4,33466E-05, 0) (1,54054E-06, 0)  (0,0299071, 0) (0,000524917, 0) (3,66252E-05, 0) (6,13352E-07, 0)  (0,0111932, 0) (0,00068202, 0) (3,22581E-05, 0) (2,04007E-07, 0)  (0,00958111, 0) (0,000341968, 0) (2,6733E-05, 0) (6,56369E-09, 0)  (0,00607169, 0) (0,00032011, 0) (1,90407E-05, 0) (2,79564E-10, 0)  (0,00447445, 0) (0,000234922, 0) (1,1207E-05, 0) (7,54994E-18, 0)  (0,0040846, 0) (0,000209452, 0) (9,57868E-06, 0) (4,34275E-18, 0)  (0,0033558, 0) (0,000129017, 0) (5,03293E-06, 0) (-2,87078E-18, 0)  (0,00221488, 0) (0,000160107, 0) (3,66566E-06, 0) |

Galime pastebėti, jog tik pačios pirmosios komponentės laiko didžiausią duomenų informaciją. Todėl pašalinus paskutines komponentes daug informacijos neprarasime.



pav. 2 Duomenų išsidėstymas 2 svarbiausiose komponentėse po dimensijų sumažinimo.

Šiame pavyzdyje po dimensijų sumažinimo duomenų dimensijos tampa 87898 x 2. Grafike parodomas 3000 elementų išsidėstymas dviejose komponentėse. Elementai priskirti į grupes: geltona – žaidėjas padaręs 0 – 50 nužudymų, raudona atvaizduoja 50 – 150 nužudymų, mėlyna atvaizduoja žaidėjus su daugiau nei 150 nužudymų.

# **Įžanginiai eksperimentai, patikrinantys ar dimensijų sumažinimą verta naudoti** (<pasirinktas sudėtingumas>)

Padaryti eksperimentus (kryžminės patikros naudoti nebūtina) su pasirinktais mokymosi metodais bei duomenų rinkiniu su pradinių dimensijų skaičiumi. Eksperimentams atlikti gali būti naudojama paprasčiaiusia platforma (pvz. Matlab). Taip atlikti identiškus eksperimentus su duomenų rinkiniu kuriam sumažinote dimensijų skaičių.

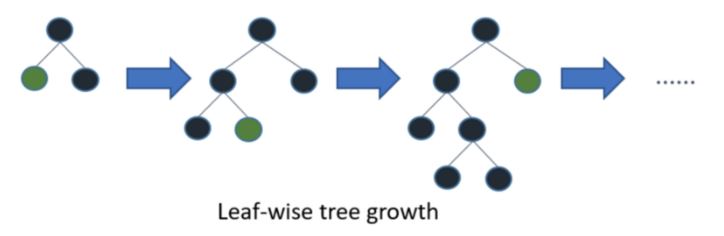
Suvestinėje Lentelėje pateikti

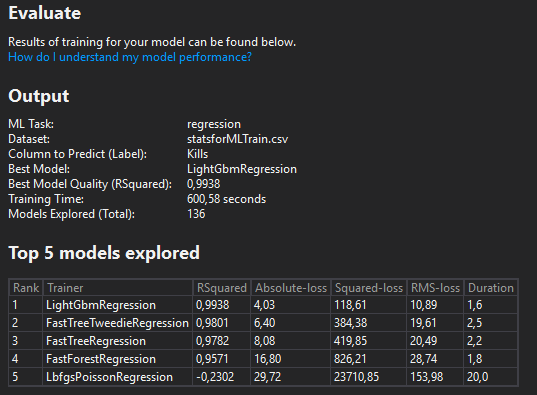
|  |  |  |  |
| --- | --- | --- | --- |
|  | NeuralNetwork Klaida | ... | Maš.Mok.Metodas\_M |
| Pradinis duomenų rinkinys | 4,40173699388703E-06 |  | <tikslumas> |
| Sumažintas duomenų rinkinys PCA būdu | 9,95875488148776E-06 |  | <tikslumas> |

Išvada dėl dimensijų sumažinimo pritaikymo tikslingumo remiantis pateikta lentele.

# **2-ojo mašininio mokymosi metodo su mokytoju panaudojimas(pilnai suprogramuotas metodas)**

Naudotas ML.net įrankis ([3] šaltinis), pasirinktas Light Gbm metodas. Šis metodas paremtas medžio mokymusi. Skirtingai nuo kitų algoritmų, šio algoritmo medis auga vertikaliai (leaf-wise). Beauginant medį metodas pasirenka medžio lapą, kuris teikia mažiausią praradimą. Šis metodas yra greitas, gali apdoroti didelius duomenis, ir užima mažiau atminties. Bet su šiuo metodu lengva pernelyg prisitaikyti prie duomenų, todėl jo nereikėtų naudoti su mažais duomenų kiekiais.





Kryžminė patikra

|  |  |
| --- | --- |
| Patikros dalių kiekis | Least squares error |
| 10 | 405,59 |
| 5 | 449,878 |

# **Mašininio mokymosi be mokytojo metodo panaudojimas (pilnai suprogramuotas metodas)**

Naudotas K-Nearest Neigbor mokymosi metodas. Apmokymui naudoti duomenys apdoroti PCA metodu. Išsirinktos dvi svarbiausios komponentės ir pagal jas buvo prognozuojama testuojamo duomens klasė.

Šis metodas klasterizuoja/klasifikuoja duomenis todėl mūsų prognozuojamą atributą – žaidėjo nužudymus, suskirstėme į 3 klases. Pirma klasė – nužudymų mažiau nei 50, antra klasė – nužudymų tarp 50 ir 150, trečia klasė – nužudymų daugiau nei 150. K reikšmė (skaičius nurodantis kiek artimiausių klasių tikrinsime) nustatytas 131.

Metodas klasifikuoja prastai – tik 36 procentų tikslumu. Turbūt reikėtų geriau paderinti žaidėjo nužudymų skaičiaus rėžius tarp klasių ir geriau parinkti K reikšmę.

# **Literatūra**

[1] <https://www.kaggle.com/lazyjustin/pubgplayerstats>

[2] <https://visualstudiomagazine.com/articles/2015/04/01/back-propagation-using-c.aspx>

[3] <https://dotnet.microsoft.com/apps/machinelearning-ai/ml-dotnet>

[4] <https://github.com/Live-Charts/Live-Charts>

[5] <https://www.mathdotnet.com/>

[6] <https://medium.com/@pushkarmandot/https-medium-com-pushkarmandot-what-is-lightgbm-how-to-implement-it-how-to-fine-tune-the-parameters-60347819b7fc>