Fakultet elektrotehnike, računarstva i informacijskih tehnologija Osijek

Računarstvo usluga i analiza podataka

SEMINARSKI RAD

„Predviđanje uspjeha iz matematike“

Petar Lenart

Vilim Svjetličić

Luka Šarčević

Osijek, 2022.

**Sadržaj**

[1. Uvod 1](#_Toc94548860)

[2. Opis problema 2](#_Toc94548861)

[2.1. Korišteni podaci 2](#_Toc94548862)

[2.2. Korišteni postupci strojnog učenja 4](#_Toc94548863)

[3. Opis programskog rješenja 6](#_Toc94548864)

[3.1. Model strojnog učenja 6](#_Toc94548865)

[3.2. Način korištenja API-ja 12](#_Toc94548866)

[3.3. Klijentska aplikacija 13](#_Toc94548867)

[4. Zaključak 15](#_Toc94548868)

[5. Poveznice i literatura 15](#_Toc94548869)

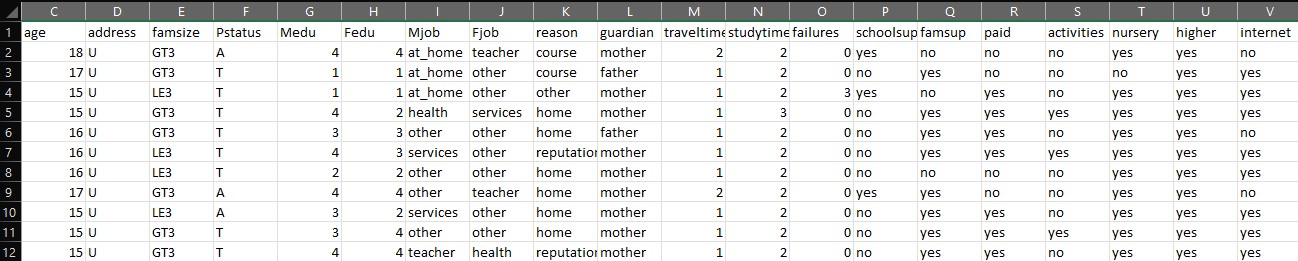
# Uvod

U današnje doba, učenici u školama opterećeni su brojnim predmetima iz kojih je potrebno postići zadovoljavajući uspjeh. Opseg gradiva sve je veći, a na uspjeh studenata utječe više čimbenika. Matematika je jedan od predmeta s lošom reputacijom i puno učenika ju ne voli te smatraju kako nisu dobri u rješavanju matematičkih zadataka. Zašto je tako? Je li matematika preteška ili ju nastavnici ne znaju dobro objasniti? Moraju li učenici biti nadareni za matematiku kako bi bili uspješni ili je to samo mit? Provedena su razna istraživanja o utjecaju brojnih čimbenika na rezultate, poput dobi, spola, socio-ekonomskog statusa, edukaciji roditelja ili izvannastavnih aktivnosti. Razvoj društva i pojava moderne tehnologije omogućili su nam predviđanje uspjeha iz navedenog predmeta ovisno o iznad navedenim čimbenicima, a upravo to je problem kojim se bavimo unutar ovoga projekta.

# Opis problema

Tema ovog projekta bila je predviđanje uspjeha iz matematike u ovisnosti o demografskim i socijalnim čimbenicima. Predviđanje je odrađeno(modelirano) korištenjem postupka linearne regresije. Korišteno je nekoliko algoritama za navedeni postupak, rezultati istih su uspoređeni, opisani te je odabran najbolji koji je korišten u modelu strojnog učenja. Nakon izrade modela potrebno ga je istrenirati i testirati te objaviti web servis. Također, potrebno je osmisliti rješenje pomoću kojeg će korisnik vršiti interakciju s navedenim servisom. Odlučili smo se za izradu desktop aplikacije, odnosno Windows Forms aplikacije u koju korisnik može unijeti tražene čimbenike, a klikom na gumb aplikacija mu vraća predviđenu vrijednost finalne ocjene.

## Korišteni podaci

 Skup podataka korišten u ovome projektu preuzet je s UCI Machine Learning Repository arhive, sastoji se od 396 instanci podataka(jedan podatak predstavlja jednu osobu), broj značajki je 33 te nema značajki koje nedostaju. Značajke predstavljaju čimbenike koji mogu utjecati na uspijeh srednjoškolaca iz matematike u portugalskoj školi Gabriel Preirera, a prikupljeni su pomoću anketa i izvještaja.

*Slika 2.1.1. Prikaz skupa podatka na primjeru nekoliko učenika i njihovih čimbenika(značajki)*

Od trideset i tri moguće značajke korištene su:

|  |  |  |
| --- | --- | --- |
| **Značajka** | **Opis značajke** | **Moguće vrijednosti** |
| sex | Spol učenika | 'F' – žensko / 'M' - muško |
| age | Dob učenika | Od 15 do 22 |
| address | Adresa učenika | 'U' – urbano područje / 'R' – ruralno područje |
| Pstatus | Bračni status roditelja | 'T' – together(žive zajedno) / 'A' – apart(žive odvojeno) |
| Medu | Obrazovanje majke | 0 – nema edukacije  1 – osnovna škola  2 – srednja škola  3 – viša stručna sprema  4 – visoka stručna sprema |
| Fedu | Obrazovanje oca | 0 – nema edukacije  1 – osnovna škola  2 – srednja škola  3 – viša stručna sprema  4 – visoka stručna sprema |
| guardian | Skrbnik | 'mother'/'father'/'other' |
| traveltime | Vrijeme putovanja do/od škole | 1 - < 15 min  2 – 15 min do 30 min  3 – 30 min do 1h  4 - > 1h |
| studytime | Vrijeme tjedno provedeno učeći | 1 - < 2h  2 – 2h do 5h  3 – 5h do 10h  4 - > 10h |
| failures | Broj nepoloženih predmeta | Od 1 do 4 |
| schoolsup | Dodatna potpora u obrazovanju | yes/no |
| famsup | Obiteljska potpora u obrazovanju | yes/no |
| paid | Dodatni plaćeni satovi matematike | yes/no |
| activities | Izvannastavne aktivnosti | yes/no |
| nursery | Pohađanje predškolske ustanove | yes/no |
| higher | Učenik želi nastaviti s obrazovanjem | yes/no |
| internet | Pristup internetu | yes/no |
| romantic | U romantičnoj vezi | yes/no |
| famrel | Kvaliteta obiteljskih veza | Od 1(vrlo loše) do 5(odlično) |
| freetime | Slobodno vrijeme nakon škole | Od 1(vrlo malo) do 5(izrazito puno) |
| goout | Izlasci s prijateljima | Od 1(vrlo malo) do 5(izrazito puno) |
| Dalc | Konzumacija alkohola preko tjedna | Od 1(vrlo malo) do 5(izrazito puno) |
| Walc | Konzumacija alkohola preko vikenda | Od 1(vrlo malo) do 5(izrazito puno) |
| health | Vlastita ocjena renutnog zdravlja | Od 1(vrlo loše) do 5(odlično) |
| absences | Broj izostanaka u školi | Od 0 do 93 |
| G1 | Ocjena u prvom polugodištu | Od 0 do 20 |
| G2 | Ocjena u drugom polugodištu | Od 0 do 20 |
| G3 – vrijednost koja se predviđa | Krajnja ocjena | Od 0 do 20 |

*Tablica 1. – korištene značajke*

Potrebno je napomenuti kako značajka G3(koja se ujedno i predviđa) uvelike ovisi o značajkama G1 i G2 jer one predstavljaju ocjene na kraju prvog i drugog polugodišta, a tražena značajka sveukupnu zaključnu ocjenu iz matematike. Također, u portugalskom obrazovnom sustavu ocjene idu od nula do dvadeset gdje je ocjena:

20 – 17.5 : odličan

17.4 – 15.5 : vrlo dobar

15.4 – 13.5 : dobar

13.4 – 9.5 : dovoljan

9.4 – 3.5 : nedovoljan

3.4 – 0 : slab

## Korišteni postupci strojnog učenja

Često postoje dvije ili više varijabli koje su povezane, odnosno postoji određena zavisnost među njima koju želimo istražiti. Istraživanje upravo tih zavisnosti omogućuju nam regresijske tehnike. Modeli koji koriste tehnike regresije mogu biti linearni ili nelinearni. Iako su, u osnovi, linearni modeli jednostavniji, u slučaju manjeg broja podataka oni mogu dati bolje rezultate od nelinearnih modela.

Jednostavan linearan regresijski model moguće je opisati na slijedeći način. Neka postoje koeficijenti α i β takvi da se za svaku novu vrijednost xi ovisna varijabla može zapisati kao:

(1)

gdje je:

– ovisna varijabla(izlaz), tj. ono što se želi predviđati.

– neovisna varijabla(ulaz), tj. ono na temelju čega se predviđa

– slučajna varijabla s normalnom raspodjelom i očekivanom vrijednošću nula

, – koeficijenti(parametri) modela

Parametri modela su nepoznati i potrebno ih je odrediti(procijeniti) temeljem podataka dostupnih za učenje. Najčešće se određuju metodom najmanjih kvadrata(engl. *least squares criterion*) koja minimizira vrijednosti kvadrata udaljenosti između opaženih podataka i regresijske krivulje(pravca).

Chart, scatter chart

Description automatically generated

*Slika 2.1.1. – primjer regresijske krivulje*

Umjetne neuronske mreže, iako poznatije po korištenju za „duboko učenje“ i modeliranje problema poput prepoznavanja slika, one se lako mogu prilagoditi problemima regresije. Regresija pomoću neuronske mreže metoda je nadziranog učenja i zahtijeva stupac s oznakom. Postoje brojni oblici i vrste neuronskih mreža, a možda najpoznatija je mreža s propuštanjem prema naprijed i prostiranjem pogreške unatrag. Ona se naziva višeslojnim perceptronom(MLP) i sastoji se od tri sloja: prvi(ulazni) sloj, središnji(skriveni) sloj i izlazni sloj. Jedna od varijacija ove mreže korištena je i u našem projektu.

Diagram

Description automatically generated Učenje pomoću stabla odluke(engl. *decision tree learning*) jedna je od metoda pristupa predviđanju korištena u statistici, rudarenju podataka i strojnom učenju. Stablo odluke(engl. *decision tree*) je model nalik stablu gdje čvor predstavlja test atributa, grana predstavlja izlaz(rezultat) navedenog testa, dok svaki list predstavlja donešenu odluku nakon izračuna svih atributa. U projektu smo koristili tzv. „boosted decision tree“ model koji nam daje poboljšani model predviđanja.

*Slika 2.1.1. – primjer stabla odluke za pitanje – je li subotnje jutro pogodno za tenis?*

# Opis programskog rješenja

Model za predviđanje uspjeha iz matematike napravljen je u Machine Learning Studiu u sklopu Microsoft Azure platforme. Napravljena je usporedba 3 regresijska algoritma predviđanja nad istim skupom podataka. Sva tri regresijska algoritma uspoređena su po performansama uz korištenje normalizacije podataka i bez normalizacije podataka te je odabran najbolji za implementaciju.

Aplikacija za prikaz rezultata omogućava unos testnih podataka i prikaz predviđenog uspjeha iz matematike. Desktop aplikacija je u cijelosti izrađena unutar Visual Studio razvojnog okruženja, koristeći programski jezik C#.

## Model strojnog učenja

Za regresijski model strojnog učenja uspoređeni su rezultati ponuđenih modela sa i bez normalizacije: Linear Regression, Neural Network Regression i Boosted Decision Tree Regression. Lijeva strana modela sadrži predviđanje podataka sa normalizacijom, a desna bez. Cilj normalizacije skupa podataka je svođenje numeričkih vrijednosti atributa na isti raspon, najčešće između 0 i 1, no to se može odrediti svojevoljno. U našem modelu podatci su normalizirani na raspon između 0 i 1.

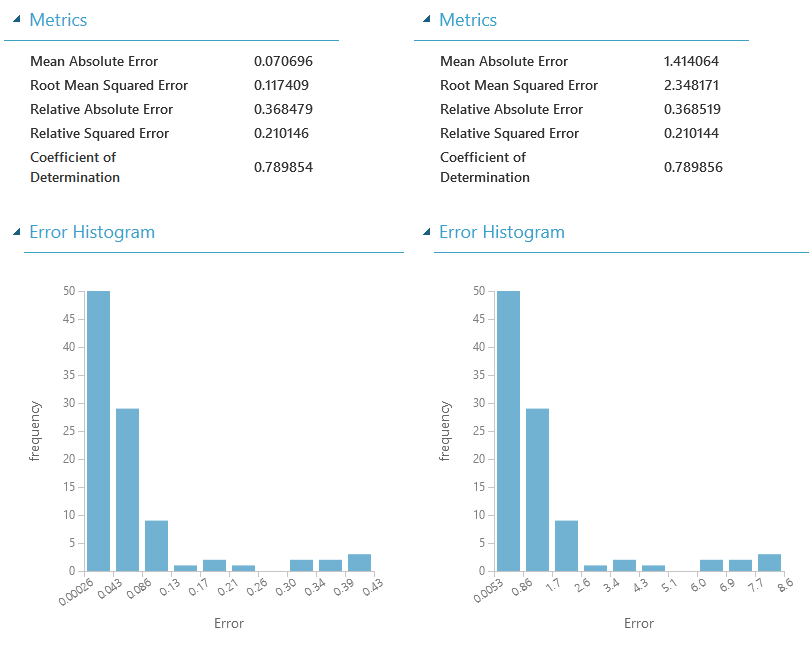
Slika na kojoj se prikazuje tekst, karta, na zatvorenom, nekoliko

Opis je automatski generiran

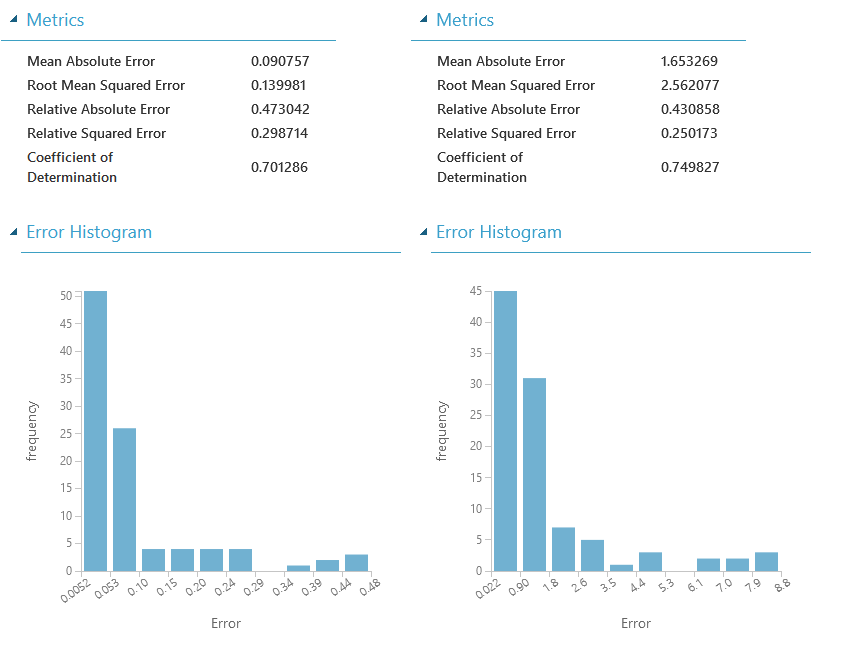
*Slika 3.1.1. Slika modela u cijelosti*

U modelu korišteni su elementi: „Select Column in Dataset“ za odabir podataka koje želimo koristiti u algoritmima regresije, „Split Data“ za podjelu podataka na testne podatke i podatke koji se upotrijebe za učenje modela. Podatci su podijeljeni po redovima, gdje se 75% redova koristi za učenje modela, a ostalih 25% za testiranje naučenog modela. Element „Train Model“ koristi se za odabir podatka (stupca) koji se želi predviđati. U ovom slučaju to je stupac G3 tj. završna ocjena iz matematike. Element „Score Model“ služi za tablični i grafički prikaz predviđenih ocjena i stvarnih ocjena iz testnih podataka kao što je prikazano na slici 3.1.8. Posljednji korišteni element je „Evaluate Model“ i prikazuje sve parametre za ocjenjivanje efikasnosti modela kao i „Error Histogram“. Ovaj element važan je za usporedbu različitih algoritama predviđanja i odabir najboljeg.

Kako bismo utvrdili trebamo li koristiti normalizaciju, uspoređuju se rezultati predviđanja sva tri regresijska algoritma sa i bez normalizacije. Utvrđeno je da normalizacija ne donosi bolje rezultate predviđanja u našem modelu. Na slikama 3.1.2., 3.1.3. i 3.1.4. primjećuje se da je relativna apsolutna pogreška manja u algoritmima bez normalizacije. Poželjno je da relativna apsolutna pogreška ima vrijednosti što bliže nuli. Također, koeficijent determinacije je veći kod algoritama bez normalizacije. Koeficijent determinacije može imati vrijednosti od 0 do 1 te korisniku govori koliko promjena u jednoj varijabli ima utjecaja na drugu pri predviđanju vrijednosti. Što je koeficijent bliže vrijednosti 1 to je model pouzdaniji u predviđanju budućih vrijednosti.



*Slika 3.1.2. Usporedba algoritma Linear Regression sa (lijevo) i bez (desno) normalizacije*



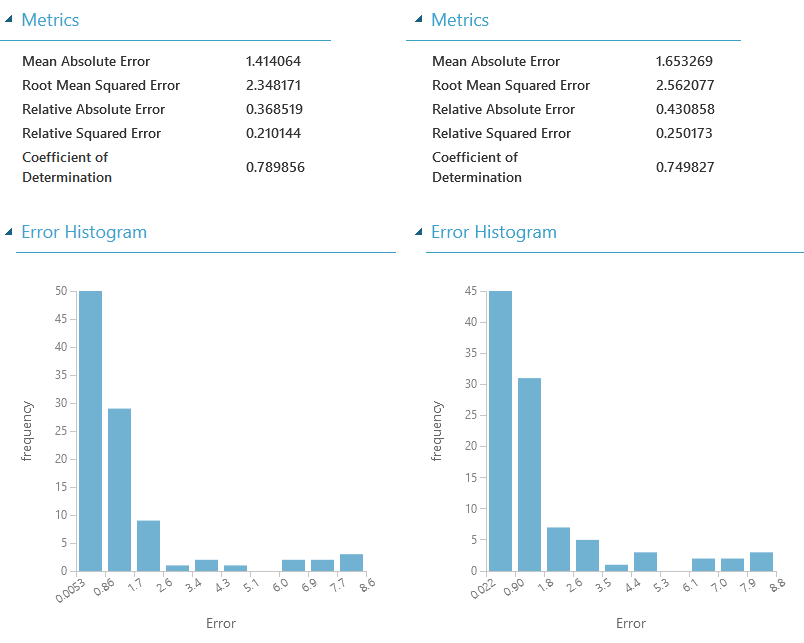
*Slika 3.1.3. Usporedba algoritma Neural Network Regression sa (lijevo) i bez (desno) normalizacije*



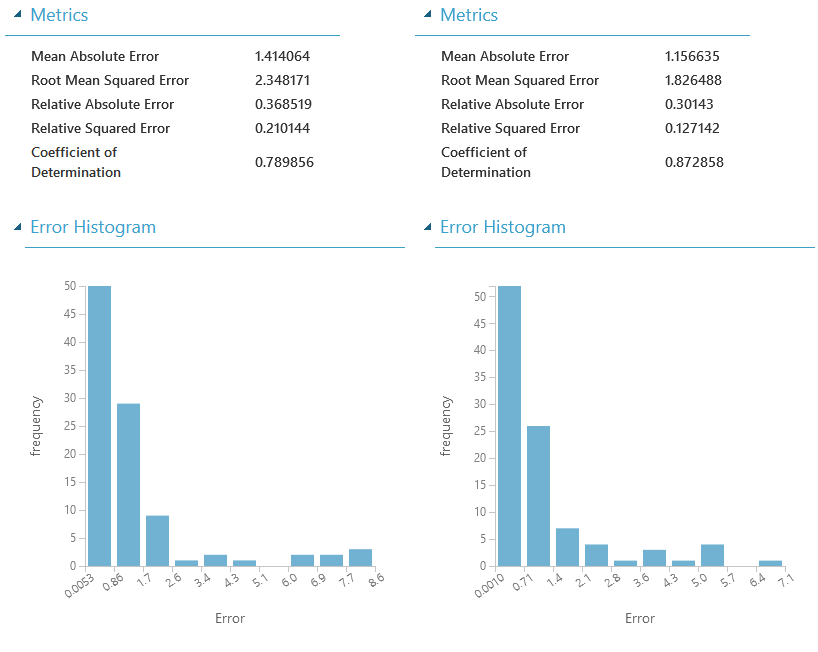
*Slika 3.1.4. Usporedba algoritma Boosted Decision Tree Regression sa (lijevo) i bez (desno) normalizacije*

Normalizacija je korisna u skupovima podataka koji imaju veliki raspon vrijednosti, a to u našem modelu nije slučaj. Zbog toga koristimo regresijske algoritme bez normalizacije podataka. Sada je potrebno utvrditi koji algoritam je najbolji za korišteni skup podataka.

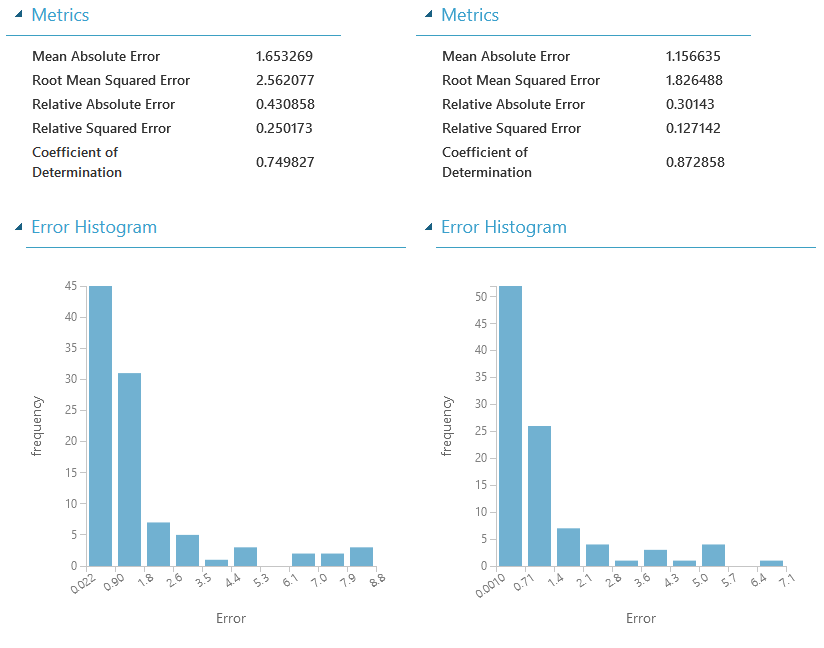
Na slikama 3.1.5., 3.1.6. i 3.1.7. prikazane su usporedbe svih triju algoritama. Odabrani algoritam je Boosted Decision Tree Regression zbog boljih performansi od ostala dva algoritma. Svi parametri koji služe za usporedbu dvaju algoritama bolji su kod navedenog. Odabrani algoritam ima najmanju srednju i relativnu pogrešku te najveći koeficijent determinacije od sva tri algoritma. Na slikama 3.1.6. i 3.1.7. vidi se usporedba performansi odabranog algoritma sa ostala dva. Na slikama su također vidljive i vrijednosti parametara svih algoritama.

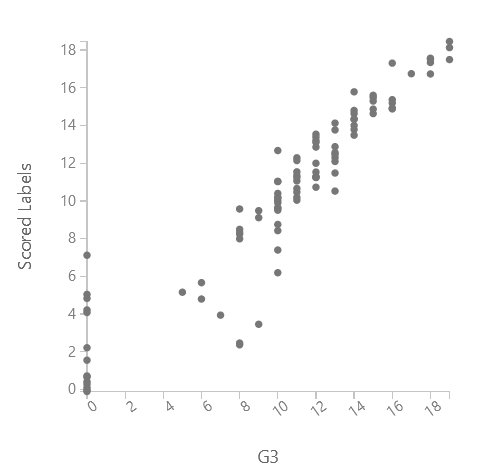


*Slika 3.1.5. Usporedba algoritma Linear regression (lijevo) i Neural Network Regression (desno)*



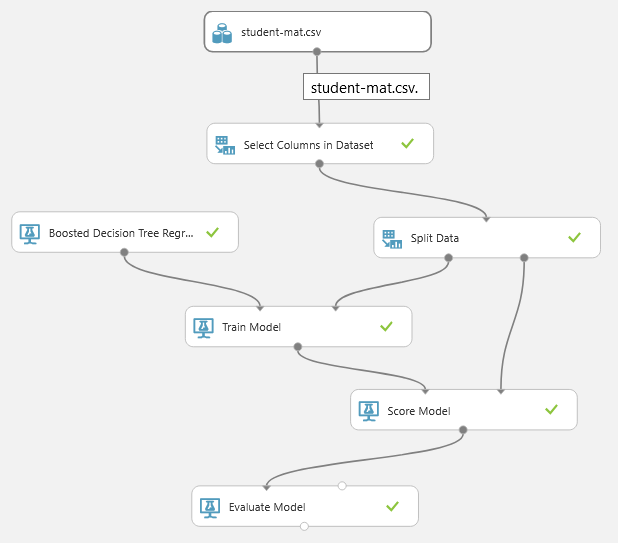
*Slika 3.1.6. Usporedba algoritma Linear regression (lijevo) i Boosted Decision Tree Regression (desno)*



*Slika 3.1.7. Usporedba algoritma Neural Network Regression (lijevo) i Boosted Decision Tree Regression (desno)*

Slika 3.1.8. Graf usporedbe predviđenih i stvarnih vrijednosti

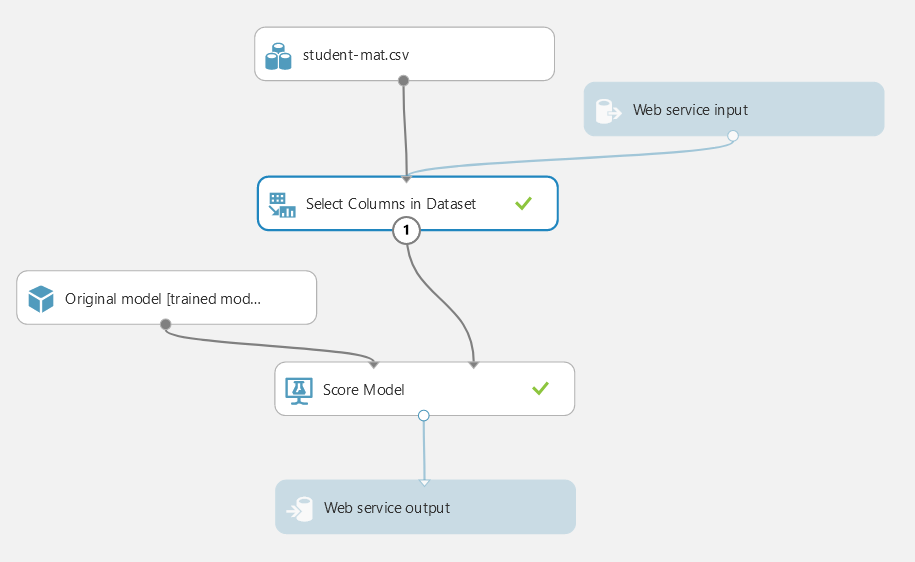
Na slici 3.1.8. prikazan je graf na usporedbe predviđenih i stvarnih vrijednosti završnih ocjena iz matematike. Na x-osi nalaze se stvarne vrijednosti, a na y-osi nalaze se predviđene vrijednosti. U slučaju da je model točno predvidio sve vrijednosti vidjela bi se linija točkica pod kutom od 45°. U našem slučaju, algoritam nije 100% točan, ali se prepoznaje linija točkica pod 45° sa manjim odstupanjima. Prema grafu, možemo zaključiti kako model predviđanja uspjeha dobro predviđa ocjene iz matematike.



*Slika 3.1.9. Model odabranog algoritma*

## Način korištenja API-ja

Microsoft Azure za stvoreni model kreira predložak koda za slanje i primanje podataka. Uz dodavanje API ključa u kod, program šalje informacije servisu koji ima istrenirani model i vraća rješenje u obliku JSON-a.



*Slika 3.2.1. Model web servisa izveden iz modela sa slike 3.1.8.*

Dobiveni kod pokreće se u aplikaciji pritiskom na tipku za predviđanje. Ključni dijelovi koda prikazani na slikama 3.2.2. i 3.2.3. zaduženi su za slanje i primanje informacija.



*Slika 3.2.2. Slanje podataka servisu sa klijentske aplikacije*



*Slika 3.2.3. Primanje obrađenih podataka*

JSON koji aplikacija primi sa servisa sadrži sve već prijašnje poslane informacije uz dodanu informaciju o traženom rezultatu na kraju niza. JSON je pretvoren u objekt uz pomoć web stranice koja generira kod, a unutar klase koja u nizu sadrži rezultat dodana je metoda za dohvaćanje posljednje vrijednosti niza.

## Klijentska aplikacija

Pokretanjem aplikacije otvara se prozor na kojemu se mogu odabrati svi parametri za predviđanje ocjene. Za unos binarnih informacije kao što su YES/NO odgovori koriste se radioButton-i, a za unos brojčanih parametara koristi se comboBox koji nudi padajući izbornik te jedan textBox za unos izostanaka učenika. Odabirom svih parametara potrebno je kliknuti na gumb „Predict“ koji šalje upit web servisu i predviđena ocjena se ispisuje ispod teksta „Predicted value:“. Također, ispod predviđene ocjene napisan je uspjeh učenika na način koji je opisan u poglavlju 2.1. Za pouzdano predviđanje potrebno je popuniti sve parametre, ali u slučaju da neki parametri nisu popunjeni aplikacija i dalje vraća ocjenu. Važno je napomenuti kako pouzdanost predviđene ocjene opada što je više parametara neispunjeno.

Slika na kojoj se prikazuje tekst, računalo, prijenosnik, na zatvorenom

Opis je automatski generiran

*Slika 3.3.1. Prikaz aplikacije*

Slika na kojoj se prikazuje tekst, na zatvorenom, računalo, prijenosnik

Opis je automatski generiran

*Slika 3.3.2. Prikaz predviđene ocjene*

# Zaključak

Projekt prikazuje aplikaciju koja za predviđanje rezultata koristi strojno učenje tehnikom regresije. Odabrana su tri regresijska algoritma koja su uspoređena po performansama te uz korištenje normalizacije podataka i bez iste. Rezultati su pokazali kako za korišteni podatkovni skup normalizacija nije potrebna, a kao najprikladniji algoritam pokazao se Boosted Decision Tree, koji je i korišten u aplikaciji.

Uspoređeni modeli su realizirani i istrenirani na servisu Microsoft Azure sa 75% podataka korištenih za treniranje, a 25% korištenih za testiranje točnosti modela. Izrađena je desktop aplikacija u koju korisnik može unijeti potrebne parametre za predviđanje ocjene, a aplikacija mu preko servisa vraća predviđenu vrijednost.

# Poveznice i literatura

Programskom je rješenju moguće pristupiti preko:

|  |
| --- |
| [Programsko rješenje na GitHubu](https://github.com/VilimSvjetlicic/Ruap_projekt) |
| [ML model](https://gallery.cortanaintelligence.com/Experiment/Score-prediction-model) |
| [ML model](https://gallery.cortanaintelligence.com/Experiment/Regression-comparison) |

1. <https://www.pmf.unizg.hr/_download/repository/PREDAVANJE11.pdf>
2. <https://docs.microsoft.com/en-us/azure/machine-learning/component-reference/neural-network-regression>
3. <https://docs.microsoft.com/en-us/azure/machine-learning/component-reference/boosted-decision-tree-regression>
4. <https://en.wikipedia.org/wiki/Decision_tree_learning>
5. <https://www.fer.unizg.hr/_download/repository/SU-4-Stabla-odluke.pdf>
6. <https://json2csharp.com/>