STUDIA UNIVERSITATIS MOLDAVIAE, 2017, nr.2(102)
Seria "Ştiinţe exacte şi economice" ISSN 1857-2073 ISSN online 2345-1033 p.43-49

CZU: 004.434:004.82

PREZICEREA PERFORMANŢELOR STUDENŢILOR FOLOSIND ÎNVĂŢAREA AUTOMATĂ (Machine Learning)

Maria CRISTEI, Ghenadie MARIN, Victor STELEA

Universitatea de Stat din Moldova

În prezent, învăţarea automată (*machine learning*) ocupă un loc important în inteligenţa artificială, preocupându-se de dezvoltarea algoritmilor ce permit unui sistem informatic să înveţe date, reguli şi algoritmi. Învăţarea automată presupune în primul rând identificarea şi implementarea unei modalităţi cât mai eficiente de reprezentare a informaţiilor, în sensul facilitării căutării, reorganizării şi modificării acestora. În acest sens, în prezentul articol se descrie utilitatea şi aplicabilitatea tehnicilor de învăţare automată supervizată la problemele de predicţie şi implementarea acestora în dezvoltarea aplicaţiilor informatice. Aplicaţia elaborată este unică prin felul ei de executare a modelului *machine learning* de predicţie. Metodologia folosită în aplicaţia elaborată este mixtă, cuprinzând tehnologii complexe de ultimă oră: mediul de dezvoltare Jupyter Notebook, limbajul de programare Python împreună cu cele mai populare librării ale acestuia utilizate în *machine learning*, instrumente de dezvoltare a aplicaţiei web Flask.

Cuvinte-cheie: *învăţare automată, inteligenţă artificială, sisteme inteligente.*

PREDICTING STUDENT PERFORMANCE USING MACHINE LEARNING

At present, machine learning occupies an important place in artificial intelligence, and is concerned with the development of algorithms that allow an information system to learn data, rules, and algorithms. Automatic learning involves first and foremost the identification and implementation of a more efficient way of representing information in order to facilitate search, reorganization and change. In this respect, this article describes the utility and applicability of supervised automated learning techniques to prediction problems and their implementation in the development of computer applications. The elaborate application is unique in its way of executing the Machine learning prediction model. The methodology used in the developed application is mixed, including state-of-the-art complex technologies: the Jupyter Notebook development environment, the Python programming language along with its most popular bookkeeping libraries used in machine learning, Flask web application development tools.

Keywords: *machine learning, artificial intelligence, intelligent systems.*

Consideraţii generale privind învăţarea automată *(machine learning)*

De-a lungul timpului, savanţii au definit diferit noţiunea de *inteligenţă artificială*. Este de remarcat că toate definiţiile pot fi împărţite în patru clase, în funcţie de cum este înţeleasă „inteligenţa":

a) sisteme care gândesc la fel ca oamenii;
b) sisteme care se comportă la fel ca oamenii;
c) sisteme care gândesc raţional;
d) sisteme care se comportă raţional [1].

Inteligenţa artificială este un concept care se referă la capacitatea unui sistem de a acţiona într-un mod „inteligent", cum ar fi: să înveţe singur, să se adapteze şi să reacţioneze în situaţii total noi. Practic, aceasta ar însemna că inteligenţa artificială ar ieşi din limitele în care a fost programată de cineva şi ar acţiona independent, indiferent de consecinţe (pozitive sau negative). Astăzi un astfel de sistem nu există în realitate. Scopurile de bază ale cercetărilor în inteligenţa artificială sunt evidenţiate de subdomeniile acesteia: prelucrările simbolice, procesarea simbolică şi limbajele de procesare simbolică, procesarea limbajului natural, raţionamentul automat şi demonstrarea teoremelor etc.

Un sistem înzestrat cu „inteligenţă" este capabil să înveţe pentru a-şi îmbunătăţi interacţiunea cu mediul. Mediul oferă stimuli sau informaţie elementului de învăţare, care foloseşte această informaţie pentru a îmbunătăţi cunoştinţele (explicite) din baza de cunoştinţe. Aceste cunoştinţe sunt utilizate de componenta de prelucrare (rezolvare) în rezolvarea problemei. Sistemul învaţă dacă îşi îmbunătăţeşte performanţele la îndeplinirea sarcinii pe baza experienţei. Învăţarea denotă schimbările dintr-un sistem, astfel încât acesta să poată realiza: (a) aceeaşi sarcină, mai eficient; (b) sarcini noi, posibil similare.

În funcţie de diferenţa dintre nivelul informaţiei oferite de mediu şi cel al informaţiei din baza de cunoştinţe, pot fi identificate patru tipuri de învăţare: (1) învăţarea prin memorare; (2) învăţarea prin instruire; (3) învăţarea

prin inducţie (din exemple); (4) învăţarea prin analogie. Astfel, un sistem inteligent poate învăţa fie din greşeli, prin autoperfecţiune, ghidat de profesor, generalizând, prin analogie, sau din exemple pozitive sau negative.

Metodele şi tehnicile de achiziţionare a cunoştinţelor sunt *manuale* (analistul de cunoştinţe) şi *automate* (machine learning). Domeniul învăţării automate formalizează aceste metode şi caută aplicarea lor la sistemele automate.

Sistemele de învăţare automată reprezintă varianta de inteligenţă artificială aplicabilă în prezent: în baza unor algoritmi matematici ce sunt capabili să opereze cu un volum foarte mare de date, să înveţe singure date, reguli, algoritmi şi să-şi perfecţioneze acţiunile. Învăţarea automată presupune în primul rând identificarea şi implementarea unei modalităţi cât mai eficiente de a reprezenta informaţii, în sensul facilitării căutării, reorganizării şi modificării acestora. Diferenţa esenţială faţă de inteligenţa artificială „veritabilă", care ar trebui să acţioneze complet independent, rezidă în faptul că *sistemele bazate pe învăţare automată îşi desfăşoară activitatea sub controlul uman şi în cadrul parametrilor stabiliţi de cercetători*. Aplicarea sistemelor inteligente bazate pe învăţare automată este destul de vastă: *recunoaştere de imagini şi semnal vocal* (recunoaşterea scrisului de mână, detecţia feţelor, înţelegerea limbajului vorbit); *vedere artificială* (detecţia obstacolelor, recunoaşterea amprentelor); *supraveghere bio*; *controlul roboţilor; predicţia vremii; diagnosticare medicală; detecţia fraudelor.*

Studiul învăţării automate a dus la descrierea a numeroase metode, variind după scop, date de antrenament, a strategiei de învăţare şi a modalităţii de reprezentare a datelor. Astfel, tipologia sistemelor inteligente (SI) bazate pe învăţare automată este următoarea:

- *în funcţie de scopul urmărit*:
  - ✓ SI pentru predicţii, care au ca scop predicţia ieşirii pentru o intrare nouă, folosind un model învăţat anterior;
  - ✓ SI pentru regresii, care au ca scop să estimeze forma unei funcţii uni- sau multivariată, folosind un model învăţat anterior;
  - ✓ SI pentru clasificare, care au ca scop clasificarea unui obiect în una sau mai multe categorii (clase) – cunoscute sau nu anterior, pe baza caracteristicilor (atributelor, proprietăţilor) lui;
  - ✓ SI pentru planificare, care au ca scop generarea unei succesiuni optime de acţiuni pentru efectuarea unei sarcini;
- *în funcţie de experienţa acumulată în timpul învăţării*:
  - ✓ SI cu învăţare supervizată;
  - ✓ SI cu învăţare nesupervizată;
  - ✓ SI cu învăţare activă;
  - ✓ SI cu învăţare cu întărire;
- *în funcţie de modelul învăţat (algoritmul de învăţare)*:
  - ✓ Arbori de decizie;
  - ✓ Reţele neuronale artificiale;
  - ✓ Algoritmi evolutivi;
  - ✓ Maşini cu suport vectorial;
  - ✓ Modele Markov ascunse.

Una dintre principalele direcţii – ***învăţarea supervizată*** **–** este o clasă de metode şi tehnici de învăţare automată prin care se induce/determină o funcţie de evaluare (şablon) dintr-un set de date (exemple, instanţe, cazuri) cu valori deja atribuite. Cu alte cuvinte, este formalizarea învăţării din exemple, adică se foloseşte un set de instanţe rezolvate ale problemei pentru a antrena sistemul în vederea rezolvării unor instanţe noi. Aceste instanţe rezolvate se numesc *instanţe de antrenament*. Formal, setul de instanţe de antrenament este o mulţime de perechi atribut-valoare ***(x,F(x))***, unde ***x*** este instanţa, iar ***F(x)*** clasa căreia îi aparţine instanţa respectivă.

Ideea constă în faptul că procesul de învăţare decurge în două etape: *antrenare* şi *testare.* Programul învăţă dintr-un set de exemple cu etichete cunoscute din setul de antrenament pentru a identifica exemple neetichetate din setul de teste cu o acurateţe cât mai mare. Un algoritm de învăţare supervizată analizează datele de antrenament şi produce o funcţie-şablon, numită *funcţie de clasificare* – în cazul în care datele de ieşire sunt dintr-o mulţime discretă (predefinită), sau *funcţie de regresie* – dacă datele de ieşire aparţin unei mulţimi continue. Funcţia produsă clasifică corect instanţele-exemplu, iar pentru un ***x*** pentru care nu se cunoaşte ***F(x)*** ea trebuie să propună/prezică o aproximare cât mai corectă a valorii ***F(x)***. Acest lucru cere de la algoritmul de învăţare să generalizeze de la datele de antrenament la orice date de intrare noi din mulţimea indicată în problemă.

Maşinile cu suport vectorial (*Support vector machines, SVM*) sunt bazate pe teoria învăţării statistice. Ideea de bază este de a mapa datele originale într-un spaţiu iniţial printr-o funcţie neliniară şi de a construi un hiperplan optim într-un spaţiu nou cu mai multe dimensiuni. Aceşti algoritmi pot fi utilizaţi atât pentru clasificare, cât şi pentru regresie. Maşinile cu suport vectorial sunt o metodă de învăţare supervizată *state-of-the-art.* Algoritmii SVM sunt folosiţi pe larg în bioinformatică datorită acurateţei înalte şi capacităţii de a se descurca cu date multidimensionale, cum ar fi descrierea genelor, flexibilitatea modelării diverselor surse de date [2].

În prezent, aplicaţiile practice ale SVM sunt numeroase. Cele mai cunoscute aplicaţii reale în care au fost implementate SVM sunt: recunoaşterea scrisului de mână, clasificarea textului şi a hipertextului, identificarea sexului din imagini şi algoritmii de ordonare a paginilor după relevanţă [3].

Analiza predictivă şi construirea unui model de predicţie al performanţelor studenţilor

Analiza predictivă este un domeniu al analizei statistice, care se ocupă cu extragerea informaţiilor din date şi folosirea acestora pentru a prezice comportamentul unor fenomene din viitor. Analiza predictivă cuprinde tehnici statistice din învăţarea automată, data mining, teoria jocurilor, prin care se caută descoperirea unor relaţii şi şabloane în volum mare de date, care pot fi folosite pentru a prezice comportamente şi evenimente. Spre deosebire de alte tehnici, analiza predictivă este anticipativă, folosind evenimentele precedente (curente şi din trecut) pentru a anticipa evenimentele din viitor.

Performanţa studenţilor este unul dintre cele mai importante scopuri ale instituţiilor de învăţământ, însă acest lucru nu este deloc simplu. Studenţii, având note joase sau chiar negative, sunt nevoiţi să repete anul academic sau, în cel mai rău caz, sunt exmatriculaţi. Aceste rezultate duc la consecinţe neplăcute atât pentru instituţia de învăţământ, cât şi pentru studenţii acesteia, implicând cheltuieli financiare semnificative. Rezultatele nesatisfăcătoare ale studenţilor pot fi consecinţa mai multor factori, printre care: scăderea motivaţiei acestora, probleme sociale (în familie, în relaţie), consumul relativ excesiv de alcool etc. Din această perspectivă, *ne propunem, prin aplicarea tehnicilor de învăţare automată, elaborarea unui sistem inteligent care, în baza modelului de predicţie, prin găsirea unor asocieri între performanţele şi obiceiurile studenţilor, să poată fi utilizat la prevenirea eşecurilor posibile în condiţii de incertitudine a stării universitare a studenţilor.*

Tehnicile de achiziţionare automată a cunoştinţelor necesită un set mare de date. Cu cât sunt mai puţine date, cu atât scade posibilitatea oricărui model de învăţare supervizată de a face vreo predicţie. Învăţarea supervizată face predicţii în baza unor fapte precedente; astfel, dacă date sunt puţine, rezultă că modelul nu are din ce învăţa. Învăţarea trebuie să ducă la formularea de suficiente „reguli", astfel încât acestea să permită rezolvarea unor probleme dintr-un spaţiu mai larg decât cel pe baza căruia s-a făcut învăţarea. Adică, învăţarea trebuie să îmbunătăţească performanţa unui sistem nu doar în rezolvarea repetată a aceluiaşi set de probleme, ci şi în rezolvarea unor probleme noi. Acest lucru presupune o generalizare a unei metode de rezolvare pentru a acoperi un număr cât mai mare de instanţe posibile, dar şi păstrarea unei specializări suficiente pentru a fi identificate corect instanţele acceptate. Aceasta se poate face fie inductiv, generalizând o problemă plecând de la un set de exemple, fie deductiv, plecând de la o bază de cunoştinţe suficiente asupra universului problemei şi extrăgând date şi reguli esenţiale. Pentru a putea face acest lucru, un algoritm de învăţare trebuie să fie capabil să selecteze acele elemente semnificative pentru rezolvarea unei instanţe viitoare a problemei.

Procesul de creare a modelului predictiv include următorii paşi:

1. ***Definirea proiectului*** – include formularea obiectivelor şi rezultatelor aşteptate pentru proiect şi traducerea acestora în obiective şi sarcini din analiza predictivă.
2. ***Explorare*** – analizarea sursei de date şi determinarea celui mai potrivit set de date şi a celei mai bune abordări pentru a construi modelul.
3. ***Pregătirea datelor*** – selectarea, extragerea şi transformarea datelor ce vor fi folosite pentru crearea modelului.
4. ***Construirea modelului*** – crearea, testarea şi validarea modelului. Evaluarea acestuia în baza unui set de date de antrenament şi satisfacerea cerinţelor proiectului.
5. ***Exploatare/utilizare*** – aplicarea regulilor (rezultatele/deciziile efectuate în model) formate pentru etichetarea unor noi date.
6. ***Gestiunea modelului*** – gestionarea şi întreţinerea modelelor construite pentru a îmbunătăţi performanţa, a promova reutilizabilitatea şi a minimiza activităţile redundante.

Exemplele (instanţele) folosite pentru acest model: un total de 1044 de studenţi, dintre care 395 au frecventat cursul „Inteligenţa artificială" şi 649 – cursul „Limba engleză", şi s-a considerat că toţi sunt consumatori de alcool (cel puţin o dată pe săptămână). Setul de date a fost preluat din baza de date de pe site-ul UCI Machine Learning (https://archive.ics.uci.edu/ml/datasets/student+performance). Instanţele sunt reprezentate printr-un număr fix (31) de atribute/caracteristici, care evaluează fiecare student şi fiecare atribut putând avea un număr limitat de valori. Funcţia obiectiv ia valori de tip discret, iar arborele de decizie reprezintă o disjuncţie de mai multe conjuncţii.

Datele din baza de cunoştinţe mai sunt caracterizate şi de acurateţea şi calitatea învăţării (metode de evaluare, măsurări de performanţă). O componentă esenţială a unui algoritm de învăţare este metoda de verificare, o metodă capabilă să confirme dacă generalizările făcute sau regulile deduse se apropie mai mult de soluţia ideală decât starea anterioară a sistemului.

Fie ***F*** funcţia pe care dorim s-o aproximăm, ***S*** un set de instanţe de antrenament ***(x,F(x))***, iar ***H*** funcţia indusă prin învăţare. Este sau nu ***H(x)*** aproape de ***F(x)***, pentru orice ***x*** din spaţiul instanţelor posibile? Argumentul standard este acela că ***H*** nu poate fi prea departe de ***F***, deoarece ***H*** clasifică corect instanţele de antrenament, deci probabil şi pe celelalte. Deci, ***H*** este *probabil aproximativ corect* (PAC). Un concept este învăţabil PAC dacă există un algoritm eficient care are o probabilitate mare de a găsi o aproximare a conceptului *probabil aproximativ corectă*. Termenul „*învăţabil PAC*" a fost introdus de Valiant (1984).

Presupunerea aflată la baza acestei justificări este aceea că instanţele de antrenament şi instanţele de testare sunt uniform distribuite în spaţiul instanţelor posibile ale problemei. Acest lucru este fundamental în justificarea oricărui rezultat al unei învăţări supervizate, prezentată grafic în Figura 1.

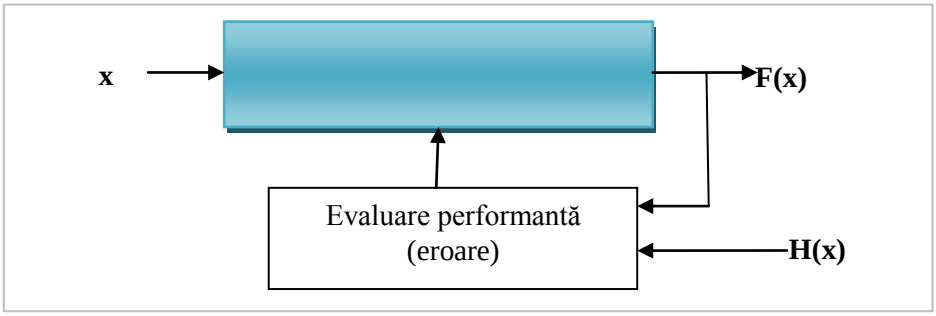


Fig.1. Învăţarea supervizată.

După construirea mai multor modele de regresie diferite, există o multitudine de criterii prin care pot fi evaluate şi comparate. În continuare, vom menţiona cele mai importante măsurări de evaluare folosite a modelului predictiv de regresie elaborat. Vom nota cu ***p*** vectorul valorilor prezise, cu ***a*** – vectorul valorilor aşteptate, cu ***n*** – numărul de instanţe.

Pentru măsurarea erorii medii pătrate (EMP), care poate fi comparată doar între modelele de regresie, a căror eroare măsurată are aceeaşi unitate de măsură, folosim [4]:

$$EMP = \frac{\sum_{i=1}^{n}(p_i - a_i)^2}{n} \quad [1]$$

Pentru a compara două modele pentru care erorile sunt măsurate în unităţi diferite folosim eroarea relativă pătrată (ERP):

$$ERP = \frac{\sum_{i=1}^{n}(p_i - a_i)^2}{\sum_{i=1}^{n}(a - a_i)^2} \quad [2]$$

Eroarea medie absolută (EMA) are aceeaşi unitate ca şi datele originale şi poate fi comparată numai între modelele ale căror erori sunt măsurate în aceleaşi unităţi:

$$EMA = \frac{\sum_{i=1}^{n}|p_i - a_i|}{n} \quad [3]$$

La fel ca ERP, eroarea relativă absolută (ERA) poate fi comparată între modele ale căror erori sunt măsurate în unităţi diferite.

$$ERA = \frac{\sum_{i=1}^{n}|p_i - a_i|}{\sum_{i=1}^{n}|a - a_i|} \quad [4]$$

Selectarea tehnologiilor informatice pentru realizarea aplicaţiei

Metodologia folosită în realizarea informatică a aplicaţiei elaborate a fost mixtă, bazându-se pe tehnologii software de ultimă oră. Decizia noastră a avut ca prim argument utilizarea instrumentelor dedicate implementării algoritmilor din *machine learning* şi analiza datelor, fără necesitate de mari resurse. În acest sens, astăzi una dintre cele mai populare tehnologii este *Jupyter Notebook* (iniţial fiind numit *IPython*, mai apoi redenumit în Jupyter). Ca limbaj de programare a fost ales *Python* şi distribuţia Anaconda, ca tehnologii inovatoare ce s-au remarcat prin simplitate, portabilitate şi robusteţe şi au început să fie utilizate pentru producerea şi dezvoltarea unui număr mare de aplicaţii software, sub diverse forme, implementate în diferite domenii. Limbajul Python poate fi folosit pentru programarea *obiect-orientată*, *funcţională* şi *procedurală.* Ca web framework pentru dezvoltarea aplicaţiei s-a folosit *Flask.*

Rezultate finale obţinute

În urma pregătirii şi analizei setului de date, cu ajutorul IPython API citim, executăm şi transformăm *notebook*-urile Jupyter în HTML. Din pagina de start a aplicaţiei (Fig.2) suntem invitaţi să selectăm şi să încărcăm fişierul cu date pentru care se va efectua predicţia.

Fig.2. Pagina de start a aplicaţiei.

În urma studiului comparativ, se poate afirma cu siguranţă că aşa tip de aplicaţie nu a fost până în prezent elaborată, deoarece cel puţin un argument este faptul că tehnologia API este relativ nouă (motiv ce explică lipsa documentaţiei de exploatare). Aplicaţia elaborată, combinând o interfaţă web cu un model de *machine learning*, facilitează enorm utilizarea modelului de predicţie de către diverşi utilizatori.

În urma încărcării şi exploatării setului de date, au fost obţinute rezultate/preziceri destul de pozitive (Fig.3, 4), folosind ca caracteristici pentru predicţie performanţele (notele) anului 1 (A1) şi ale anului 2 (A2).

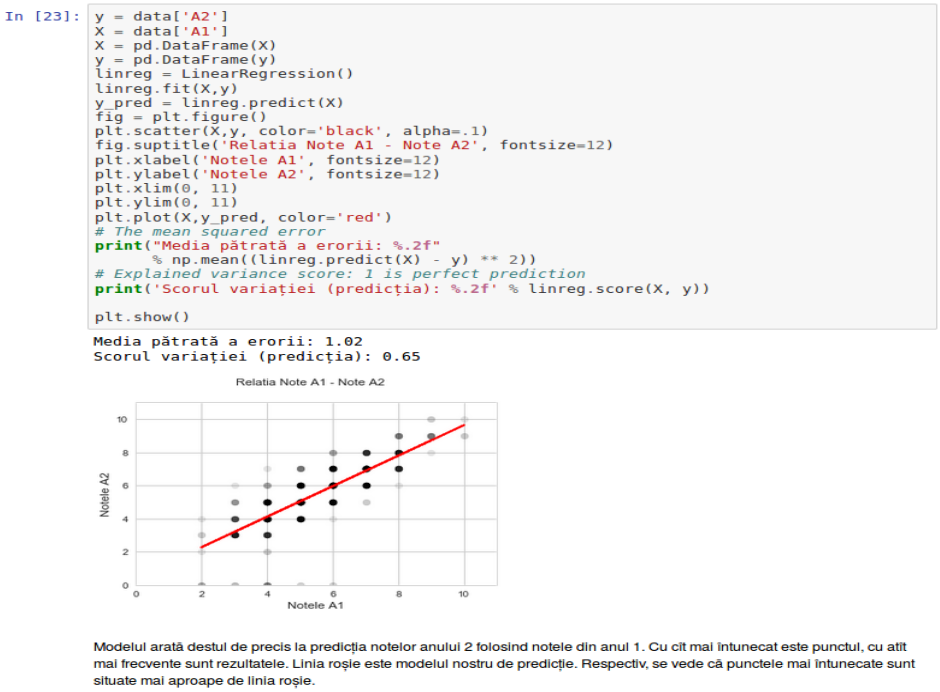


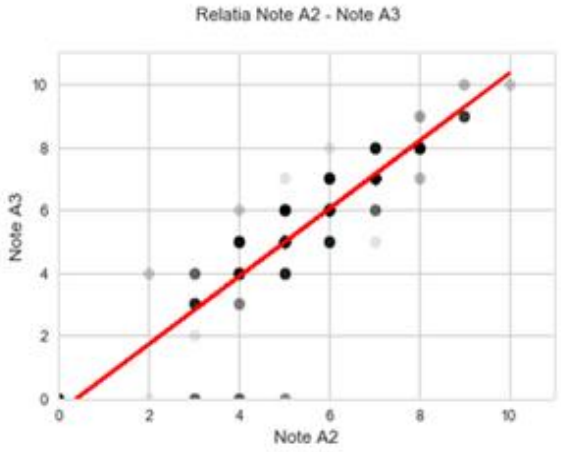
Fig.3. Rezultatele predicţiei prin **regresia liniară**, folosind caracteristicile **A1** şi **A2**.

Analizând dacă această regresie funcţionează şi pentru predicţia performanţelor (notelor) anului 3 (A3), obţinem următoarele rezultate:

In [24]:
```
y = data['A3']
X = data['A2']
X = pd.DataFrame(X)
y = pd.DataFrame(y)
linreg = LinearRegression()
linreg.fit(X,y)
y_pred = linreg.predict(X)
fig = plt.figure()
plt.scatter(X,y, color='black', alpha=.1)
fig.suptitle('Relatia Note A2 - Note A3', fontsize=12)
plt.xlabel('Note A2', fontsize=12)
plt.ylabel('Note A3', fontsize=12)
plt.xlim(0, 11)
plt.ylim(0, 11)
plt.plot(X,y_pred, color='red')
# The mean squared error
print("Media pătrată a erorii: %.2f"
      % np.mean((linreg.predict(X) - y) ** 2))
# Explained variance score: 1 is perfect prediction
print('Scorul variației (predicția): %.2f' % linreg.score(X, y))

plt.show()
```

```
Media pătrată a erorii: 1.03
Scorul variației (predicția): 0.76
```



Lucrează chiar mai bine pentru notele anului 3. Aceasta este din cauza corelației dintre A2 și A3 (~ 87%) mai mari decît celei dintre A1 și A2 (~ 80%).

Concluzii

Nu am avut posibilitatea de a găsi careva corelații semnificative între obiceiurile studenților și notele acestora la universitate. Aceasta poate fi datorită dimensiunii prea mici a seturilor de date, sau poate datorită vreunei variabile care noi nu o considerăm în aceste date. Dar se pare că există un model similar pentru fiecare dintre perioadele de note (A1, A2 și A3), arătînd o oarecare continuitate în rezultatele universitare. Însă mai trebuie careva cercetare în acest domeniu pentru a putea avea rezultate mai clare.

Fig.4. Rezultatele predicției prin **regresia liniară**, folosind caracteristicile **A2** și **A3**.

Concluzii

Există numeroase probleme în care nu s-au exploatat îndeajuns metodele şi tehnicile de învăţare automată existente. Una dintre aceste probleme este previziunea performanţelor studenţilor. Deşi s-au întreprins încercări de a utiliza învăţarea automată pentru astfel de predicţie, această problemă rămâne în continuare un subiect important de cercetare. Studiul de caz din acest articol a demonstrat cu succes că pot fi aplicate tehnici de învăţare automată pentru astfel de probleme. Algoritmii de învăţare folosesc date precedente ca să construiască modele matematice care să fie capabile să facă predicţii de performanţă universitară.

Procesul de predicţie este unul complex, care necesită un volum mare de informaţii, pe care specialistul nu poate să le deţină în totalitate. Elaborarea aplicaţiei informatice prin implementarea tehnicilor de învăţare automată permite realizarea de experimente pentru studierea problemei de predicţie a performanţelor studenţilor.

În final, conchidem că utilizarea tehnologiilor de *machine learning* şi, în special, a modelelor de predicţie poate facilita enorm rezolvarea anumitor probleme (cum ar fi intervenţia la timp a unor factori de decizie pentru a preîntâmpina dificultăţile cu care se confruntă unii studenţi în ce priveşte continuarea studiilor), ceea ce va duce la rezolvarea lor la timp.

Referinţe:

1. RUSSEL, S.J., NORVIG, P. *Artificial Intelligence: A Modern Approach, Prentice Hall*. New Jersey, 1995, p.4-8, 567-570.
2. BEN-HUR, A., WESTON, J. *A User's Guide to Support Vector Machines*. Colorado State University, Colorado, NEC Labs America, Princeton, p.1-3.
3. KAR, P. *Support Vector Machines and their Applications*. Indian Institute of Technology Kanpur, 2009, p.56-70.
4. SAYAD, S. *Model Evaluation – Regression*. University of Toronto, 2010, http://chem-eng.utoronto.ca/~datamining/dmc/model_evaluation_r.htm

Prezentat la 13.07.2017