Kratkotrajna kretanja u cijenama dionica

Porin Ćustić i Ante Mijoč

***Sažetak*—Naš zadatak je bio koristeći podatke cijena raznih dionica, sektorskih i ekonomskih podataka pronaći model koji predviđa hoće li cijena određene dionice rasti ili padati u idućih 60 minuta. Zadatak je proizašao iz INFORMS Data Mining natjecanja održanog 2010. godine. Koristili smo više klasifikatora i napravili probir varijabli pri čemu smo uspjeli dobiti dobre rezultate.**

***Keywords*—dionice, predviđanje cijena.**

I. Uvod

Najveća želja sudionika u dioničkim tržištima, bilo da su to trgovci, analitičari, investitori ili investicijski fondovi, je predvidjeti kretanje cijene neke dionice. Problem kojim smo se mi bavili je bio upravo taj - problem predviđanja kratkotrajnog kretanja cijene jedne određene ali nepoznate dionice. Projekt je proizašao iz INFORMS Data Mining natjecanja održanog 2010. na web stranici Kaggle [1]. U ovom radu ćemo opisati tijek i rezultate našeg rada. Sama struktura rada je sljedeća: prvo ćemo opisati problem i skup podataka, potom ćemo navesti neke metode koje su koristili sami natjecatelji natjecanja, pa ćemo navesti i opisati metode koje smo mi koristili, te na kraju rezultate koji su ocjenjeni u skladu sa pravilima natjecanja.

II. Opis problema

Zadatak projekta je bio da izgradimo model koji će na temelju različitih podataka kao što su cijene drugih dionica, sektorskih podataka i ekonomskih podataka predvidjeti hoće li cijena naše promatrane nepoznate dionice za 60 minuta porasti ili se smanjiti. Preciznije, pošto su podaci bili podijeljeni u vremenske serije, trebalo je za svaku vremensku seriju t reći hoće li promatrana dionica u trenutku $t + 60min$ porasti ili se smanjiti. U bankama i investicijskim kućama se ulaže mnogo truda kako bi se predvidjelo kretanje cijena dionica ili financijskih instrumenata pa je jasno da je ovakav zadatak zanimljiv čak i 7 godina nakon što je samo ovo natjecanje završilo.

III. Opis skupa podataka

Svi podaci su osigurani u sklopu natjecanja, te nije bilo potrebno, niti moguće, dohvaćati druge podatke. Sami podaci koje smo koristili su podaci kroz 609 ne nužno nezavisnih varijabli. Te varijable su predstavljale cijene drugih dionica, sektorske podatke, ekonomske podatke, razna predviđanja te indekse stručnjaka. Sve varijable su bile neoznačene, te sama cijena naše promatrane dionice nije bila uključena u skup podataka. Zbog činjenice da su sve varijable bile neoznačene nismo mogli primijeniti nikakvo domensko znanje ili neku teoriju iz domene financijskih tržišta. Podaci su bili podijeljeni u vremenske serije u intervalima od 5 minuta. Trening skup je sadržavao 5922, a skup za testiranje 2539 opažanja koja su se kronološki nastavljala na trening skup. Iako smo imali 609 varijabli u skupu podataka, zapravo smo imali 152 varijable sa 4 podvarijable oblika $VariableXOPEN$, $VariableXHIGH$, $VariableXLOW$ te $VariableXLAST\_PRICE$, gdje je X vrijednost između 1 i 152. Za cijene dionica su to očito bile početna cijena, najveća cijena, najmanja cijena i zadnja cijena u toj vremenskoj seriji. Za varijable koje nisu bile cijene dionica nije bilo jasno što ti podaci znače. Sami skup podataka je bio jako šarolik: za neke varijable je dosta podataka nedostajalo (od 70 do 100posto), te zbog činjenice da su varijable bile cijene raznih dionica i neki ekonomski podaci sami brojevi u skupu podataka su se dosta razlikovali. Zbog toga smo prije samog klasificiranja proveli neke metode čišćenja podataka. Podaci su došli u dvije csv datoteke: TrainingData.csv i ResultData.csv, gdje u datoteci ResultData.csv nije bila zapisana vrijednost target labele, vec se uspješnost klasifikacije ocjenjivala kroz Kaggle sučelje. Problem nam je bio što ocjenjivanje za to natjecanje više nije bilo aktivno pa smo kroz kontakt ljudi koji su organizirali INFORMS natjecanje uspjeli doći do skupa podataka za testiranje koji ima vrijednosti target labele te tako sami izvršiti ocjenu uspješnosti.

IV. Metodologija natjecatelja

Kroz forume natjecanja na Kaggleu smo imali priliku saznati malo o načinu na koji su natjecatelji prišli ovom problemu. Glavne stavke koje smo nalazili kod većine natjecatelja su bile sljedeće: preprocesiranje (čišćenje) podataka, probir varijabli, korištenje lagged podataka, logističke regresije [4] i metode potpornih vektora (SVM) [5]. Što se tiče preprocesiranja bilo je nekoliko ideja: od brisanja dijela podataka zbog nedostatka podataka ili čudnih vrijednost, preko popunjavanja praznina srednjim vrijednostima, do stvaranja novih varijabli umjesto četiri podatka za svaku varijablu, čime bi se automatski dolazilo do velikog smanjenja broja varijabli. Većina natjecatelja je polučila dobre uspjehe samo koristeći preprocesiranje i logističku regresiju, uz tek na kraju malo profinjavanje rezultata koristeći SVM. Mnogi natjecatelji su tvrdili za pojedine varijable da je to upravo dionica koju promatramo iako su to voditelji natjecanja čak i nakon kraja natjecanja opovrgavali.

V. Metodologija

U radu smo koristili računalni program Weka za analizu podataka [10] te programski jezik Python i Python biblioteku za strojno učenje scikit-learn [6]. U prvom dijelu projekta smo pretežno koristili Weku za početnu analizu podataka i brzo i jednostavno pokretanje nekih klasifikatora. Pred kraj projekta smo se prebacili na korištenje programskog jezika Python, u kojem smo izradili i konačno rješenje.

U praktični dio projekta smo krenuli sa preprocesiranjem podataka. Kao što smo već naveli, skup podataka je bio jako šarolik i za dosta varijabli je bio veliki postotak vrijednosti koje nedostaju. Odlučili smo izbaciti sve varijable u kojima je veći dio vrijednosti nedostajao jer smo vjerovali da će nam veliki postotak nedostajućih vrijednosti znatno utjecati na naš model. Nakon tog izbacivanja ostalo je samo nekoliko varijabli sa malim postotkom praznih vrijednosti (manje od 1posto) i te vrijednosti smo popunili aritmetičkom sredinom ostalih vrijednosti dane varijable. Također, bilo je nekoliko varijabli koje su imali vrijednosti samo za jedan od stupaca $OPEN$, $HIGH$, LOW, $LAST\_PRICE$ i te vrijednosti su pripadale jednoj od sljedećih skupina: bile su sve jednake; bile su u velikim grupama jednake sa malim razlikama; bile su sastavljene samo od 0 i 1 sa jednom vrijednosti prevladavajućom. Pošto nam nije bilo jasno što predstavljaju ti podaci, a vjerovali smo da nam neće biti od velikog značaja zbog strukture podataka u tim stupcima, onda smo i te varijable izbacili iz skupa podataka. Nakon toga smo od početnih 609 došli na 476 varijabli. Također, u nekim treniranjima klasifikatora smo koristili i normalizirane podatke.

Potaknuti prezentacijom drugoplasiranog natjecatelja [3] smo krenuli u projekt trenirajući logističku regresiju u Weki na cijelom skupu podataka. Međutim, unatoč njegovim tvrdnjama da već logistička regresija na cijelom skupu podataka bez ikakvog uplitanja u podatke daje dobre rezultate mi smo cijelo vrijeme dobivali loše rezultate koji su bili u razini slučajnog pogađanja. Pokušali smo i sa SVM-om, te baggingom ali ništa nije davalo značajnije rezultate.

Nakon toga smo se prebacili na korištenje Pythona te probali iskoristiti ideju koju smo dobili kroz ocjenu projektnog prijedloga, a to je korištenje lagged podataka. Sagradili smo lagged varijable target varijablu za od 1 do 12 koraka u prošlost. Uzeli smo taj broj koraka jer su podaci bili podijeljeni u vremenske serije od 5 minuta a mi smo predviđali rast ili pad cijene dionice za 60 min = 12 x 5min. Tada smo pokrenuli logističku regresiju na svim podacima uz dodatak tih lagged target varijabli i prvi put dobili rezultate bolje od slučajnog pogađanja. Svi rezultati pojedinih metoda su navedeni u idućem poglavlju. Nakon toga smo pokrenuli logističku regresiju i SVM samo na podacima varijable $Variable74$ koju su i mnogi natjecatelji istaknuli kao ključnu u rješavanju problema i dobili još bolje rezultate. Dodatno smo uspjeli poboljšati rezultat kroz uključivanje lagged podataka $Variable74$ a najbolje rezultate smo dobili logističkom regresijom na podacima koji se sastoje od sadašnjih i lagged vrijednosti $Variable74$, ali bez lagged podataka target varijable.

VI. Rezultati

Prvo kažimo nešto o evaluaciji rješenja. Evaluacija za natjecanje se provodila kroz Kaggle sustav, ali kako smo već naveli, natjecanje više nije bilo aktivno za evaluaciju pa smo sami izračunavali rezultat koristeći vrijednosti target labele za test skup podataka. Evaluacija se vršila kroz tzv. Area under the ROC curve (AUC). Za izračunavanje smo koristili scikit-learnovu funkciju metrics.roc_auc_score. Napomena, svugdje smo za lagged podatke uzimali korake od 1 do 12.

Rezultati:

- Logistička regresija na svim podacima - oko 0.50 (slučajno pogađalo)
- SVM na svim podacima - oko 0.51
- Logistička regresija na Variable74 - oko 0.49
- Logistička regresija na svim podacima uz lagged podatke target labele - 0.677388355976
- Logistička regresija na Variable74 uz lagged podatke target labele - 0.877990464935
- SVM na Variable74 uz lagged podatke target labele - 0.879196690839
- Logistička regresija na Variable74 sa lagged podacima svih varijabli - 0.907147856618
- SVM na Variable74 sa lagged podacima svih varijabli - 0.879495020195
- Logistička regresija na Variable74 sa lagged podacima samo od Variable74 - **0.924187052506**
- SVM na Variable74 sa lagged podacima samo od Variable74 - 0.7922738434

Iz rezultata se jasno vidi da je ovo odličan uspjeh, pogotovo u području kao što je predviđanje cijena dionica, gdje bi ovakav model, na ovakvim podacima, donio iznimnu uspješnost u trgovanju.

Literatura

[1] INFORMS 2010, https://www.kaggle.com/c/informs2010, (pristupano 26. lipnja 2017.)

[2] How I did it: The top three from the 2010 INFORMS Data Mining Contest, https://blog.kaggle.com/2010/10/11/how-i-did-it-the-top-three-from-the-2010-informs-data-mining-contest/, (pristupano 26. lipnja 2017.)

[3] How I did it: The top three from the 2010 INFORMS Data Mining Contest, https://kaggle2.blob.core.windows.net/forum-message-attachments/39806/1091/INFORMS%20Presentation%20PDF.pdf, (pristupano 26. lipnja 2017.)

[4] Logistic regression, https://en.wikipedia.org/wiki/Logistic_regression, (pristupano 26. lipnja 2017.)

[5] Support vector machine, https://en.wikipedia.org/wiki/Support_vector_machine, (pristupano 26. lipnja 2017.)

[6] Sckit-learn, http://scikit-learn.org, (pristupano 26. lipnja 2017.)

[7] Logistic regression, http://scikit-learn.org/stable/modules/generated/sklearn.linear-model.LogisticRegression.html, (pristupano 26. lipnja 2017.)

[8] Support vector machine, http://scikit-learn.org/stable/modules/svm.html, (pristupano 26. lipnja 2017.)

[9] Pandas, http://pandas.pydata.org/, (pristupano 26. lipnja 2017.)

[10] Weka, http://www.cs.waikato.ac.nz/ml/weka/, (pristupano 26. lipnja 2017.)