Detectarea anomaliilor în tranzacționarea cu acțiuni

Coman Tudor, 351 Eftimie Petre-Laurențiu, 351
Luculescu Ștefan, 311

Rezumat

Această lucrare explorează tehnici pentru detectarea anomaliilor în tranzacțiile bursiere și analizează impactul acestora asupra eficienței pieței financiare. Scopul principal al studiului este identificarea și caracterizarea evenimentelor neobișnuite care pot afecta performanța activelor financiare. Pentru a valida eficacitatea metodelor propuse, se folosesc seturi de date reale provenind de pe diverse piețe financiare. Sunt prezentate atât metode din analiza semnalelor, cât şi din analiza statistică.

Cuprins

Capitolul 1

Introducere

1.1 Definiţia anomaliilor

O **anomalie** este o entitate ce diferă semnificativ de restul enităţilor din setul de date. Definiţia lui Hawkins este urmatoarea[4]: *"O anomalie este o observaţie ce deviază atât de mult faţă de restul observaţiilor, încât să creeze suspiciunea că a fost generată de un mecanism diferit".*

1.2 Datele folosite

Pentru testarea modelelor, am folosit date reale de pe bursă, cu ajutorul bibliotecii `yfin`, ce este un Python SDK pentru API-ul oferit de Yahoo Finance. Din aceste date, am extras timestamp-ul (în acest caz, trunchiat la zi) şi preţul zilnic la care s-a închis tranzacţionarea ("Close"), pentru a obţine seria de timp.

1.3 Indicatorul beta

Volatilitatea acţiunilor este o măsură importantă în analiza seriilor de timp determinate de evoluţia preţurilor acestora, fiind indicatorul principal şi un standard în industrie pentru exprimarea riscului asociat cu o investiţie în acele acţiuni. Aceasta oferă investitorilor şi analiştilor financiari o imagine asupra incertitudinii şi dinamicii pieţei.

O măsură relativă (faţă de piaţă) pentru volatilitate este indicatorul β. Acţiunile cu valori $\beta > 1$ sunt mai volatile decât S&P 500, iar cele mai mici (dar pozitive) sunt mai puţin volatile. O valoare egală indică o strânsă corelare cu piaţa. În acelaşi timp, β poate lua şi valori negative, caz în care corelaţia este inversă (de exemplu, dacă $\beta = -1.0$, atunci acţiunea respectivă are o corelaţie inversă 1 la 1 cu piaţa). [1]

Formula pentru calcularea indicatorului este următoarea [1]:

$$\beta = \frac{Cov(R_e, R_m)}{Var(R_m)}$$

unde R_e - rentabilitatea acţiunii, R_m - rentabilitatea pieţii, iar Cov şi Var sunt notaţiile uzuale pentru covarianţa şi varianţa dintre două variabile.

1.4 Online versus offline

Prin detecţia de anomalii **offline** se întelege detecţia anomaliilor pe întreaga serie de timp, iar prin detecţia de anomalii **online** se întelege detecţia anomaliilor în ultimul punct al seriei (cel mai recent).

1.4.1 Detecţia anomaliilor offline

Prin această abordare, se analizează întreaga serie de timp pentru a identifica modele neobişnuite sau schimbări neaşteptate.

Această metodă poate fi utilă, de exemplu, în cazul detectării anomaliilor mai complexe, care nu reies dintr-un singur punct de date (a se vedea o anomalie de acest tip în analiza făcută mai jos pentru AAPL). În plus, este o opţiune folosită dacă se doreşte analiza datelor istorice, pentru a descoperi cum au influenţat anumite evenimente (crize financiare, războaie şamd.) pieţele de capital.

1.4.2 Detecţia anomaliilor online

În general, detecţia anomaliilor online (în ultimul punct al seriei) se face prin compararea valorii acestuia cu rezultate statistice, istorice sau date de un model predictiv.

Un exemplu de caz în care această metodă ar fi aleasă în detrimentul primeia ar fi în cazul trading-ului *high-frequency* (la intervale foarte scurte de timp), unde eficienţa computaţională poate face diferenţa între o tranzacţie profitabilă sau o tranzacţie care generează o pierdere.

1.4.3 Comparaţie

Din punct de vedere al contextului pe care îl deţine modelul, metoda offline este clar una mai bună. În schimb, această metodă are o latenţă în detectare (fiind necesară analiza unei întregi serii de timp), abordarea online fiind una mai bună dacă se doreşte eficienţă computaţională.

Capitolul 2

Metode statistice

2.1 Eliminarea trendului

2.1.1 Metoda regresiei polinomiale

Metoda constă în presupunerea că trendul seriei este un polinom de grad mic (2, 3, 4, 5) și calcularea polinomului de regresie. Pașii metodei sunt:

1) Se consideră valorile seriei de timp $v = [v_0, v_1, ..., v_{n-1}]^T$ și momentele de timp $t = [t_0, t_1, ..., t_{n-1}]^T$ la care au fost înregistrate valorile v.

2) Se determină trendul seriei sub forma unui polinom P de grad mic $d \in \{2, 3, 4, 5\}$ prin rezolvarea sistemului liniar $P(t_i) = v_i, i \in \{0, 1, ..., n-1\}$ în sensul celor mai mici pătrate.

3) Seria fără trend este $r = [r_0, r_1, ..., r_{n-1}]^T$, $r_i = v_i - P(t_i)$.

2.1.2 Metoda regresiei polinomiale - rezolvarea sistemului

1) Polinomul P este reprezentat prin vectorul de coeficienți $c = [c_0, c_1, ..., c_d]^T$, $P = c_0 + c_1X + ... + c_dX^d$.

2) Matricea sistemului este $A \in M_{n,d+1}(\mathbb{R})$, $A_{i,j} = t_i^j$, $i \in \{0, 1, ..., n-1\}$, $j \in \{0, 1, ..., d\}$.

3) În ipoteza $n - 1 > d$ (seria de timp este lungă, iar gradul polinomului este mic), matricea A are rang $d + 1$ și sistemul $Ac = v$ admite soluție unică în sensul celor mai mici pătrate (c minimizează norma vectorului $v - Ab$, unde b este un vector din $\mathbb{R}^{d+1}$).

4) $c = (A^T A)^{-1} A^T v$.

2.2 Detectarea anomaliilor

2.2.1 Metoda z-score

Ideea metodei constă în presupunerea că valorile din seria fără trend reprezintă eșantioane dintr-o distributie normală de medie m și deviație standard sd. Apoi se calculează estimatorii de verosimilitate maximă [3], în acest caz media empirică pentru medie și varianța empirică pentru pătratul deviației standard. Pașii metodei sunt:

1) Se lucrează pe serii fără trend $r = [r_0, r_1, ..., r_{n-1}]^T$.

2) Se calculează media $m = \frac{1}{n} \sum_{i=0}^{n-1} r_i$ și deviația standard $sd = \sqrt{\frac{1}{n} \sum_{i=0}^{n-1} (r_i - m)^2}$.

3) Se calculează $z = [z_0, z_1, ..., z_{n-1}]^T$, $z_i = \frac{1}{sd}(r_i - m)$.

4) Se consideră anomalii valorile z_i cu modulul mai mare decât un anumt prag (de obicei 2 sau 3 sau chiar și mai mic).

2.2.2 Metoda medie mobilă

Ideea metodei constă în calcularea mediei pe o fereastră glisantă, în loc de toată seria, în scopul de a neglija posibile oscilații ale seriei. Pașii metodei sunt:

1) Se lucrează pe serii fără trend $r = [r_0, r_1, ..., r_{n-1}]^T$.

2) Se calculează media mobilă pe fiecare fereastră glisantă de lungime ws, cu pas $step < ws$.

3) După ce se calculează diferența față de medie pe toate ferestrele, valorile care depășesc un anumit prag sunt considerate anomalii.

2.2.3 Metoda de deviație medie absolută

Ideea metodei constă în presupunerea că valorile din seria fără trend reprezintă eșantioane dintr-o distributie Laplace de parametri m și s. Apoi se calculează estimatorii de verosimilitate maximă [2], în acest caz mediana empirică pentru m și deviația absolută medie pentru s.

Pașii metodei sunt:

1) Se lucrează pe serii fără trend $r = [r_0, r_1, ..., r_{n-1}]^T$.

2) Se calculează mediana seriei de timp. Mediana (p) este o valoare aleasă astfel încât jumătate din valori să fie $\geq p$ și jumătate din valori să fie $\leq p$ (în practică vom alege mijlocul vectorului sortat).

3) Se calculează deviația absolută medie $s = \frac{1}{n} \sum_{i=0}^{n-1} |r_i - p|$.

4) Se calculează $z = [z_0, z_1, ..., z_{n-1}]^T$, $z_i = \frac{1}{s}(r_i - p)$.

5) Se consideră anomalii valorile z_i cu modulul mai mare decât un anumt prag.

2.2.4 Metoda procentuală

Presupunem că apar anomalii pe un anumit procent (cunoscut) din serie (de exemplu, procent aproximat de o metodă Fourier). Se aplică una dintre metodele anterioare cu diferența că pragul se ajustează pentru a obține procentul dorit de anomalii.

2.3 Alegerea pragurilor

Pentru calcularea pragurilor pentru fiecare metodă se folosește metoda procentuală. Folosind seria fără trend $r = [r_0, r_1, ..., r_{n-1}]^T$, verificăm ce amplitudine au frecvențele 3, 4 și 5 (frecvențele 0, 1 și 2 reprezintă eventuale componente ale trendului ce nu au fost eliminate, iar frecvențele mai mari decât 5 reprezintă sezonalitatea și zgomotul). Implementare:

1) Se lucrează cu serii fără trend $r = [r_0, r_1, ..., r_{n-1}]^T$.

2) Se calculează transformata Fourier a seriei $z = [z_0, z_1, ..., z_{n-1}]^T$.

3) Se calculează suma amplitudinilor frecvențelor 3, 4, 5 (care pot indica anomalii).

4) Dacă una dintre aceste frecvențe este frecvența de amplitudine maximă, se neglijează (reprezintă probabil o componentă importantă a sezonalității).

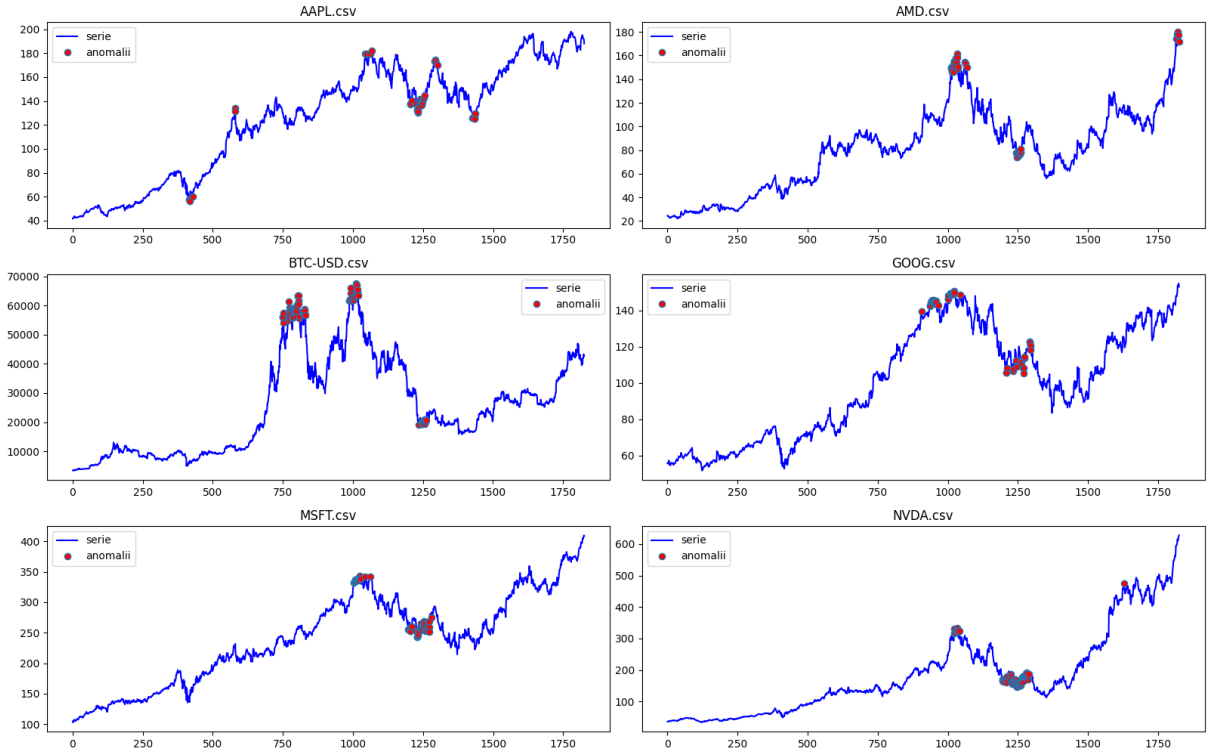
5) Se împarte suma obținută la suma tuturor frecvențelor. Această valoare va fi folosită ca *procent* de anomalii.

6) Se caută pragul optim pentru fiecare metodă în parte. Se stabilește limita inferioară a pragului la 1. Apoi se folosește observația că numărul de anomalii scade cu creșterea pragului ($anomalii(prag)$ este o funcție descrescătoare). Se pornește cu limita superioară a pragului setată la 2 și această limită se tot dublează cât timp dă procent de anomalii mai mare decât $procent$. În final se caută pragul optim între limita inferioară (=1) și limita superioară prin metoda bisecției.

2.4 Concluzii

Metodele statistice se dovedesc a fi în același timp simple și eficiente la detectarea anomaliilor când se lucrează cu seria generală, așa cum se vede din testarea lor pe un set de date. Din punct de vedere al timpului de execuție, complexitățile algoritmilor sunt: $O(n)$ pentru determinarea trendului, $O(n)$ pentru z-score și medie mobilă, $O(nlogn)$ pentru deviație absolută, $O(nlogn)$ pentru determinarea procentului în cazul metodei procentuale. Căutarea efectivă a pragului are aceeași clasă de complexitate ca metoda pentru care se caută pragul.

Imaginea de mai jos exemplifică rularea metodei deviației absolute cu prag determinat automat pe un set de date despre prețurile a 6 categorii de acțiuni.



Capitolul 3

Transformata Fourier pentru găsirea regiunilor cu anomalii

3.1 Scopul metodei

Putem folosi această metodă pentru a găsi regiuni cu puncte anormale dintr-o serie de timp, anume regiuni în care punctele au caracteristici similare doar că foarte diferite faţă de vecinii lor din afara regiunii [5].

Vom utiliza această idee doar pentru găsirea anomaliilor în date înregistrate în trecut, fapt ce ne-ar ajuta să analizăm impactul pe care anumite evenimente l-au avut asupra bursei de valori, precum o criză economică, şi astfel am putea prezice mai bine în viitor ce efect ar avea un eveniment similar. Metoda nu are ca scop prezicerea unor noi valori de pe bursă.

3.2 Ideea din spatele metodei

Transformata Fourier presupune că semnalul pe care îl analizează este **periodic** sau măcar foarte apropiat de unul periodic. Astfel, ar fi posibil să modelăm acest semnal folosind o sumă de sinusoide de frecvenţe diferite pentru seriile de timp complexe sau chiar cu o singură sinusoidă în cazul banal.

Aplicăm transformata Fourier pe seria de timp dată, iar apoi inversăm operaţia doar că vom păstra doar un număr p de parametri, practic presupunând ca **magnitudinea frecvenţelor** corespunzătoare este 0. Alegem să păstrăm doar primii p parametri cu magnitudinile cele mai mari.

Apoi, folosind acest semnal, calculăm diferenţele între valorile eşantioanelor de pe aceeaşi poziţie(înregistrate la aceeaşi dată) a ambelor semnale. Punctele care au o diferenţă absolută mai mare decât media diferenţelor absolute pentru toate punctele vor fi considerate **potenţiale** anomalii. Pentru aceste potenţiale anomalii vom păstra într-un

set S diferenţa dintre valoarea seriei în punctul respectiv şi media a c vecini ai săi de pe ambele părţi. Aplicăm **z-score** pentru fiecare punct din setul S, iar punctele a căror valoare depăşeşte un prag predefinit t sunt considerate anomalii. Pentru găsirea regiunilor anormale, găsim 2 puncte consecutive cu z-score de **semn opus** care vor marca începutul regiunii, iar pentru a marca finalul, găsim 2 puncte consecutive cu z-score de semn opus, doar că în ordine inversă faţă de semnele care marchează începutul.

Punctele cu z-score pozitiv sunt asociate cu **vârfuri**, în timp ce cele cu semn negativ sunt asociate **văilor**. Din acest motiv, acest algoritm funcţionează cel mai bine atunci când datele suferă schimbări **bruşte în frecvenţă**. Dacă schimbările se întâmplă gradual, este posibil să nu mai găsim aceste diferenţe majore pentru z-score care să ne indice regiunile cu anomalii. Cu toate acestea, complexitatea de timp scăzută a algoritmului, anume $O(n \log n)$ datorită transformării Fourier rapide, face această metodă atractivă atunci când eficienţa este un criteriu important.

3.3 Reprezentarea matematică

$$X[k] = \sum_{n=0}^{N-1} x[n] \cdot e^{-j\frac{2\pi}{N}kn}$$

$$y[n] = \frac{1}{N} \sum_{k=0}^{N-1} X[k] \cdot e^{j\frac{2\pi}{N}kn}$$

cu $X[k] = 0$ dacă $k \notin P$, unde P este setul parametrilor cu cele mai mari p magnitudini. x este semnalul iniţial, X este transformarea semnalului în domeniul frecvenţă, iar y este semnalul rezultat după trunchierea frecvenţelor.

$$S = \{x[i] - medie(vecini) | abs(x[i] - y[i]) > abs(medie(x - y))\}$$

unde $vecini = \{x[j] | j \in [i - c, i + c], i \neq j\}$

$$A = \{i | abs(z[i]) > t\}$$

unde $z = \{\frac{s[i] - mean(S)}{std(S)} | s \in S\}$ este z-score.

3.4 Rezultate şi concluzii

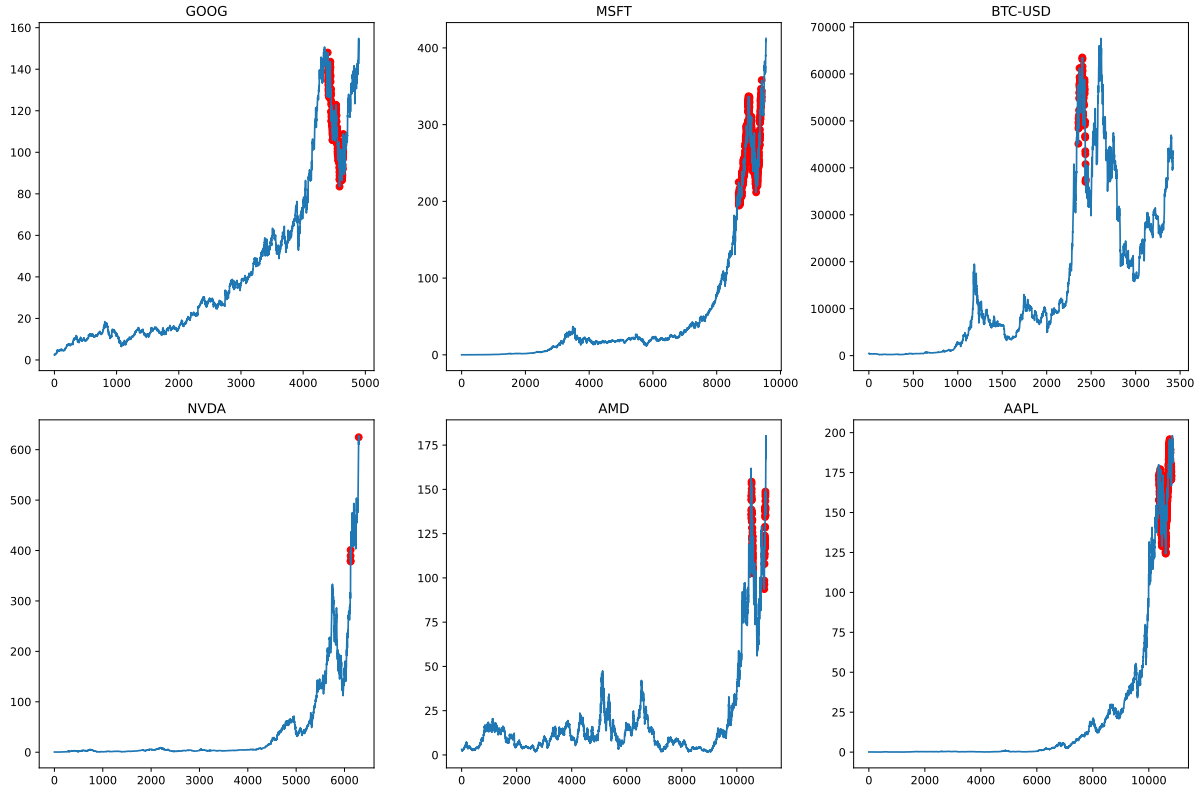
Pentru alegerea pragului t vom folosi cuantila de 0.99 din vectorul de z-score calculat, întrucât presupunem ca numărul de anomalii este mult mai mic decât cel al punctelor normale.

Numărul de parametri p îl menţinem mic, chiar sub 0.1% din numărul total de parametri, deoarece seriile de timp de pe bursa de valori au o periodicitate aproape inexistentă şi astfel ar domina componenta medie doar că având foarte mult zgomot adăugat, deci practic am fi comparat semnalul iniţial aproape cu o linie orizontală.

Numărul c de vecini influenţează probabilitatea ca un punct ce este potenţial anomalie să fie detectat ca anomalie după testul cu z-score. Dacă includem mulţi vecini în media calculată, există o posibilitate mai mare de a include puncte ce deviază faţă de medie, deci un punct ce ar fi avut o mică şansă să fie detectat ca anomalie, acum va fi acoperit de punctele ce deviază şi ele, micşorând diferenţa rezultată. În cazul opus, dacă includem mai puţini vecini, iar aceştia diferă cu mult faţă de potenţiala anomalie, atunci va avea o şansă mai mare să treacă de testul z-score.

Mai există încă 2 parametri pe care îi numim max_local şi max_region care influenţează numărul maxim de puncte detectate ca anomalii pe care suntem dispuşi să îi lăsăm între 2 anomalii cu z-score de semn opus, respectiv numărul maxim de puncte dintr-o regiune cu anomalii. Primul parametru, intuitiv, reprezintă cât de mult vrem să lăsăm semnalul să crească sau să descrească gradual până se atinge o schimbare de semn, astfel, o valoare mai mică favorizează schimbările bruşte, pe când o valoare mai mare favorizează schimbările mai line. Al doilea parametru, practic, ne arată care este lungimea maximă a unui "plafon" cu anomalii, anume până la o schimbare de semn a z-score. În general, vom presupune că regiunile cu anomalii se întâmplă pe o perioadă scurtă de timp pentru că altfel nu ar mai fi considerate anomalii.

Mai jos se pot vedea rezultatele obţinute pentru 6 tipuri de acţiuni.



Capitolul 4

Modelul ARMA

4.1 Scopul metodei

Putem folosi această metodă pentru a prezice următoarele puncte dintr-o serie de timp, cu o acurateţe ce în mod natural va scădea cu cât încercăm să vedem mai mult în viitor, şi astfel putem să semnalăm o posibilă anomalie dacă noul eşantion pe care credem că îl vom obţine şi în realitate, deviază cu mult faţă de punctele înregistrate deja.

Pentru a decide dacă un nou punct este sau nu anomalie, vom folosi un prag care este influenţat într-o proporţie majoră de volatilitatea instrumentului, anume indicatorul beta despre care am vorbit în introducere.

Acest model funcţionează bine atunci când datele din viitor pot fi modelate cu ajutorul datelor din trecut. Dacă nu există nicio relaţie între evenimentele din trecut şi cele din viitor, atunci modelul nu va putea extrage caracteristicile necesare prezicerii.

4.2 Ideea din spatele metodei

Acest model este compus din 2 părţi, AR(AutoRegressive) şi MA(Moving-Average).

Partea autoregresivă încearcă sa modeleze punctele din seria de timp folosind o regresie liniară bazată pe un număr finit de puncte din trecut la care se adaugă şi o variabilă de eroare care se poate presupune că este independent şi identic distribuită, provenită dintr-o distribuţie Gaussiană. Numărul de puncte din trecut este notat cu p şi reprezintă singurul hiperparametru pentru această componentă.

Partea de medie glisantă încearcă să modeleze termenul eroare folosind o combinaţie liniară dintr-un număr de termeni de eroare din trecut. Despre aceste erori, putem face aceleaşi presupuneri ca mai sus. Numărul de termeni eroare din trecut este notat cu q şi reprezintă hiperparametrul pentru această componentă.

Aceste 2 componente sunt însumate pentru a obţine modelul final pe care îl vom putea folosi pentru a prezice noile puncte din seria de timp.

Găsirea hiperparamterilor optimi se poate realiza cu ajutorul funcţiei de autocorelaţie parţială pentru aflarea lui p, iar pentru q putem folosi functia de autocorelaţie. Apoi, pentru rezolvarea sistemului rezultat se poate folosi tehnica celor mai mici pătrate.

4.3 Reprezentarea matematică

ARMA(p, q):

$$X_t = \varepsilon_t + \sum_{i=1}^{p} \phi_i X_{t-i} + \sum_{j=1}^{q} \theta_j \varepsilon_{t-j}$$

AR(p):

$$X_t = \varepsilon_t + \sum_{i=1}^{p} \phi_i X_{t-i}$$

MA(q):

$$X_t = \mu + \varepsilon_t + \sum_{j=1}^{q} \theta_j \varepsilon_{t-j}$$

unde X_t este seria de timp, ε_t este termenul eroare, μ este media semnalului X_t pe care o presupunem 0 de obicei, iar ϕ_i şi θ_j sunt parametrii pentru regresia liniară a modelului AR, respectiv pentru combinaţia liniară a modelului MA.

4.4 Implementare

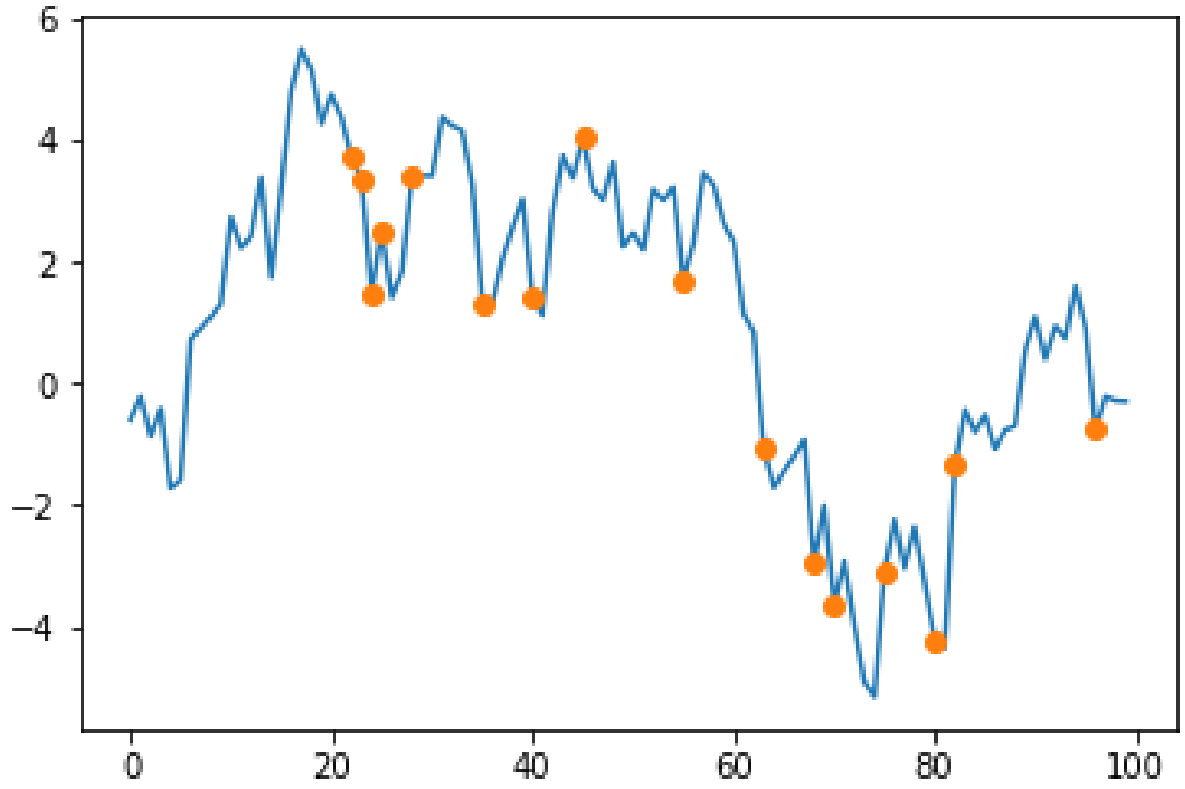
Metoda ARMA se implementează în python cu ajutorul funcției ARIMA din *statsmodels*. Ținând cont că ARMA(p,q) = ARIMA(order=(p,0,q)), se stabilește p la o zecime din lungimea seriei de timp și un prag superior pentru q ($qmax$, probabil 5 pentru că ARIMA rulează lent).
Se construiește modelul ARMA(p,q) pentru fiecare q de la 1 la $qmax$ și se prezice următoarea valoare a seriei. Media valorilor obținute reprezintă predicția.
Dacă modulul diferenței dintre valoarea reală a seriei și predicție depășește un anumit prag, se consideră anomalie.
Pragul se stabilește în funcție de predicțiile obținute pentru fiecare valoare a lui q.

Aplicare pentru ultima valoare a seriei
Modelul ARMA se construiește pentru toată seria înafară de ultima valoare sau pe ultimele $k\%$ valori înafară de ultima. Se aplică pașii anteriori pentru ultima valoare a seriei.

Aplicare pe seria generală
Pentru fiecare *start* de la $2p$ până la $n-1$ se aplică același algoritm ca la aplicarea pentru ultima valoare a seriei trunchiate până la *start*, cu diferența că pragul care indică ano-

meliile poate fi setat ulterior dacă se cunoaște dinainte procentul de anomalii din serie.

Exemplu de rulare



Capitolul 5

Prophet

5.1 Despre Prophet

Prophet [6] este un model open-source dezvoltat de Facebook pentru analiza seriilor de timp, în special pentru prognozare ("forecasting"). Această librărie separă componentele de trend, sezoniere şi reziduale, folosind în spate regresie non-liniară. Descompunerea seriei de timp se face în felul următor:

$$y(t) = g(t) + s(t) + h(t) + \epsilon_t$$

unde y este seria completă de timp, g este funcţia de trend, s este funcţia de sezonalitate, h este funcţia care modelează efectele vacanţelor ce pot apărea în mod neregulat asupra uneia sau mai multor zile, iar ϵ_t este termenul de eroare reziduală [6].

5.2 Formatul predicţiilor

Din punct de vedere al predicţiilor, pentru fiecare punct din prognoza generată de Prophet, avem 3 valori: `yhat`, `yhat_lower` şi `yhat_upper`. Ultimele două valori reprezintă limitele intervalului de incertitudine, în timp ce `yhat` este valoarea estimată pe baza acestor valori. Pe baza datelor din trecut, am calculat o eroare absolută (diferenţa dintre valoarea actuală si cea prezisă), respectiv un factor de incertitudine (diferenţa între capetele intervalului de incertitudine). Am considerat anomalii acele puncte pentru care eroarea era mai mare decât pragul de incertitudine, înmulţit cu un factor setat.

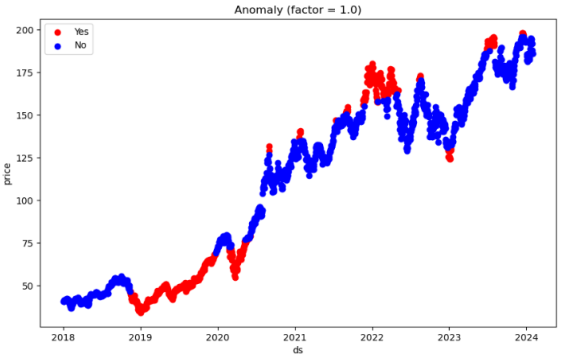
5.3 Găsirea factorului optim, rezultate şi concluzii

Prima încercare a fost aplicarea metodei procentuale pe Prophet, dar aceasta nu a mers pentru că este funcţională doar dacă numărul de anomalii depinde monoton de un singur parametru. În continuare, factorul a fost determinat din mai multe rulări (şi rezultatele cele mai bune au fost obţinute pe intervalul $[1.25, 1.5]$, i.e o eroare mai mare cu 25% - 50% decât intervalul de incertitudine).

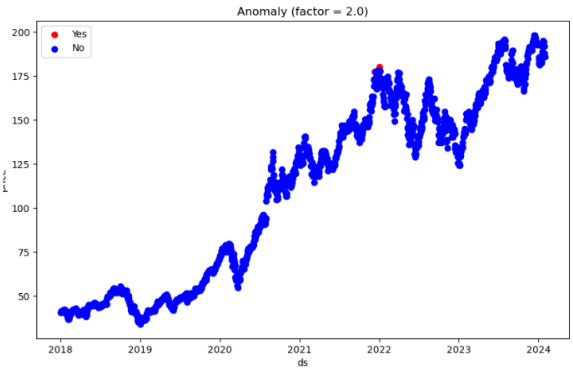
Totodată, pentru a stabili relevanţa rezultatelor obţinute din aceste rulări, am ales mai multe instrumente financiare, cu diverse volatilităţi si sezonalităţi, selectându-le cu ajutorul indicatorului β. Am comparat rezultatele obţinute de Prophet pentru detectarea de anomalii cu ajutorul a trei instrumente: Ethereum (este cunoscut faptul că pe piaţa de criptomonede volatilitatea este mult mai mare), AAPL (o acţiune cu volatilitate medie, având $\beta = 1.29$ la momentul redactării) şi VZ (o acţiune cu volatilitate mică, având $\beta = 0.38$ la momentul redactării). Graficile prezentate arată ultimii 5 ani din seriile de timp, dar Prophet a fost antrenat cu întreg setul de date disponibil pe Yahoo Finance.

Din punct de vedere al măsurării erorilor, modelul a obţinut rezultate foarte bune, obţinând valoarea erorii mediii absolute procentuale în jurul valorilor 0.1% - 0.2% în cazul acţiunilor şi în jur de 1% pentru criptomonede.

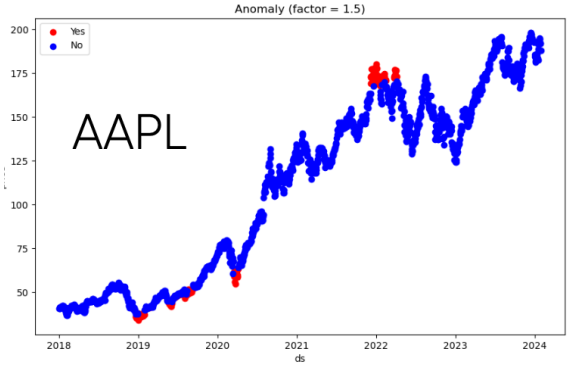
Am început prin a selecta valorile 1 şi 2 pentru factor pe acţiunea AAPL, pentru a vedea cum se comportă modelul.

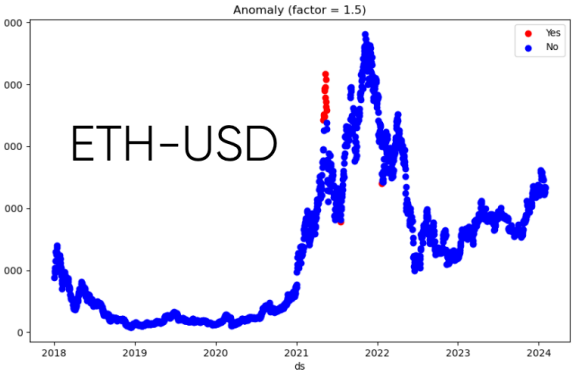


Aşa cum reiese din grafice, pentru valoarea 1 a detectat multe valori ca anomalii, în timp ce pentru valoarea 2 a detectat una singură, pe vârful de la sfârşitul anului 2021. Prin urmare, pentru a verifica dacă factorul 2 este într-adevăr prea mare pentru ca modelul să detecteze anomalii, am ales un instrument mult mai volatil, şi anume ETH-USD (Ethereum).



Chiar şi în acest caz, detectarea anomaliilor este una redusă pentru un instrument atât de volatil, aşa că următoarul experiment a implicat rularea ambelor instrumente cu factorul 1.5.

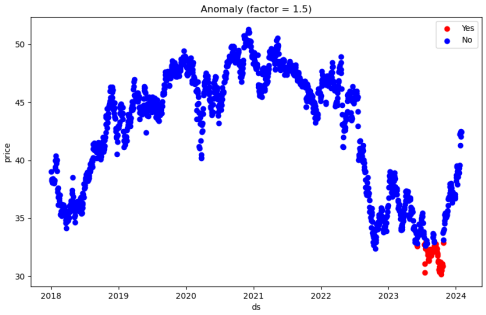


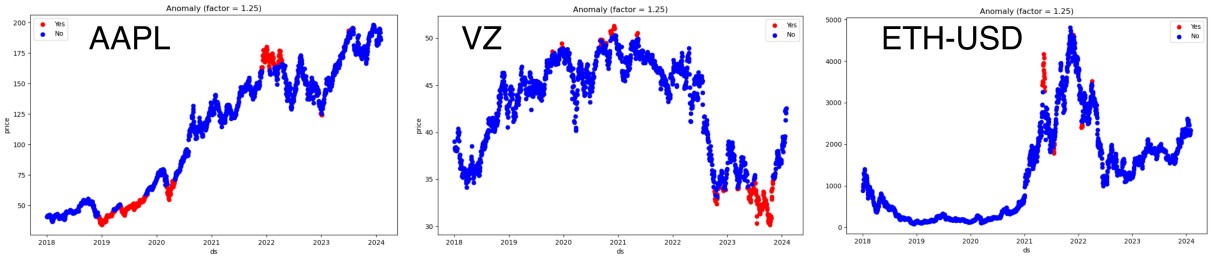


Rezultatele obţinute pentru VZ (imaginea din dreapta) confirmă faptul că 1.5 este un factor bun. Având β mic, este de aşteptat ca anomaliile detectate să fie mai puţine.

Se poate observa că modelul începe să dea rezultate de o calitate mai bună, detectând mai multe puncte de minim şi mai multe de maxim (puncte ce sunt bune pentru efectuare de tranzaţii de long, respectiv short, în contextul unui program de trading algoritmic).

Această valoare este una bună, aşa că urmează să fie validată şi pentru acţiunea VZ. Totodată, pentru a verifica dacă o valoare mai mică poate obţine rezultate mai bune, este nevoie de o rulare pe toate cele trei instrumentele folosite pâna acum, cu valoarea factorului de 1.25.





Şi pentru factorul 1.25 au fost obţinute rezultate foarte bune, dar, în mod evident, modelul este mai sensibil şi va detecta mai multe anomalii. Prin urmare, în functie de comportamentul aşteptat de către utilizator, acest prag poate fi situat între 1.25 şi 1.5.

Bibliografie

[1] *Beta: Definition, Calculation, and Explanation for Investors*, URL: https://www.investopedia.com/terms/b/beta.asp.

[2] *Estimatorii de verosimilitate maximă pentru distribuția Laplace*, URL: https://en.wikipedia.org/wiki/Laplace_distribution#Statistical_inference.

[3] *Estimatorii de verosimilitate maximă pentru distribuția normală*, URL: https://en.wikipedia.org/wiki/Normal_distribution#Estimation_of_parameters.

[4] David M Hawkins, *Identification of Outliers*, Chapman și Hall, 1980.

[5] Faraz Rasheed, Peter Peng, Reda Alhajj și Jon Rokne, „Fourier transform based spatial outlier mining", în *Proceedings of the 10th International Conference on Intelligent Data Engineering and Automated Learning*, IDEAL'09, Burgos, Spain: Springer-Verlag, 2009, pp. 317–324, ISBN: 3642043933.

[6] Sean J Taylor și Benjamin Letham, în *Forecasting at scale* (Sept. 2017), DOI: 10.7287/peerj.preprints.3190v2.