

Model dubokog učenja za optimizaciju portfelja

Ercegović, Dora

Master's thesis / Diplomski rad

2024

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **University of Zagreb, Faculty of Electrical Engineering and Computing / Sveučilište u Zagrebu, Fakultet elektrotehnike i računarstva**

Permanent link / Trajna poveznica: <https://um.nsk.hr/um:nbn:hr:168:833243>

Rights / Prava: [In copyright](#)/[Zaštićeno autorskim pravom.](#)

Download date / Datum preuzimanja: **2025-04-01**



Repository / Repozitorij:

[FER Repository - University of Zagreb Faculty of Electrical Engineering and Computing repository](#)



SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

DIPLOMSKI RAD br. 547

MODEL DUBOKOG UČENJA ZA OPTIMIZACIJU PORTFELJA

Dora Ercegović

Zagreb, lipanj 2024.

SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

DIPLOMSKI RAD br. 547

MODEL DUBOKOG UČENJA ZA OPTIMIZACIJU PORTFELJA

Dora Ercegović

Zagreb, lipanj 2024.

DIPLOMSKI ZADATAK br. 547

Pristupnica: **Dora Ercegović (0036522883)**

Studij: Računarstvo

Profil: Znanost o mrežama

Mentor: doc. dr. sc. Stjepan Begušić

Zadatak: **Model dubokog učenja za optimizaciju portfelja**

Opis zadatka:

Problem optimizacije portfelja najčešće se rješava kao zadatak kvadratičnog programiranja, gdje su najbitnije komponente u funkciji cilja varijanca i očekivanje povrata razmatranog portfelja. Zbog toga je procjena tih parametara iz financijskih vremenskih nizova ključan korak, a greške u procjenama, su česte zbog dinamičke prirode tržišta i niskog omjera signala i šuma, pogotovo za vektor očekivanja. Zbog toga su u zadnjim desetljećima razvijene razne alternativne heuristike za optimizaciju portfelja koje zaobilaze problem procjene očekivanih povrata. Jedan od novijih smjerova istraživanja uključuje i korištenje modela strojnog učenja, pogotovo dubokih modela, koji imaju kapacitet izravno iz podataka učiti optimalne alokacije imovina. U sklopu ovog diplomskog rada potrebno je prvo implementirati standardne metode i pristupe za optimizaciju portfelja, koristeći uzoračke procjene očekivanja i kovarijance te algoritme kvadratičnog programiranja. Potom je potrebno razviti modele dubokog učenja koji iz podataka uče optimalne alokacije imovina. Pritom je potrebno dizajnirati značajke za takve modele i implementirati funkciju cilja prilagođenu prirodi problema, te primijeniti odgovarajuće algoritme za učenje modela. Potrebno je također implementirati i okvir za ispitivanje razvijenih modela na povijesnim tržišnim podacima, te razvijene modele ispitati i primijeniti statističku analizu rezultata u okviru financijskih mjera performanci.

Rok za predaju rada: 28. lipnja 2024.

Zahvaljujem svom mentoru, doc. dr. sc. Stjepanu Begušiću, na uloženom vremenu, prenesenom znanju i pruženoj podršci prilikom izrade ovog diplomskog rada.

SADRŽAJ

1. Uvod	1
2. Problem optimizacije portfelja	2
2.1. Statistička svojstva financijskih vremenskih nizova	2
2.2. <i>Mean-variance</i> optimizacija	4
2.2.1. Efikasna granica	5
2.2.2. Portfelj maksimalnog Sharpeovog omjera	6
2.2.3. Nedostatci <i>mean-variance</i> optimizacije	8
3. Duboko učenje	9
3.1. Povratne neuronske mreže	12
3.1.1. LSTM mreže	13
3.2. Učenje dubokih modela	16
4. Implementacija	18
4.1. Korišteni skup podataka	18
4.2. <i>Mean-variance</i> optimizacija	20
4.3. Modeli dubokog učenja	20
4.3.1. Aproksimacijski model	21
4.3.2. Prediktivni model	22
4.3.3. Detalji implementacije u Tensorflow okviru	23
5. Rezultati	24
5.1. <i>Mean-variance</i> optimizacija	24
5.2. Modeli dubokog učenja	24
5.2.1. Aproksimacijski model	25
5.2.2. Prediktivni model	30
6. Zaključak	36

Literatura	37
Sažetak	40
Abstract	41

1. Uvod

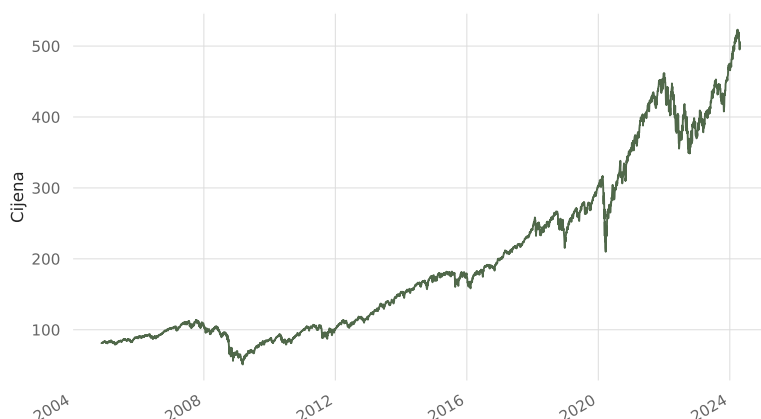
U današnjem dinamičnom i nepredvidivom financijskom okruženju, pronalazak optimalnog portfelja vrijednosnica predstavlja jedan od ključnih izazova za investitore. Tradicionalni pristupi optimizaciji portfelja, poput Markowitzove teorije portfelja [1], premda su značajno doprinijeli razvoju financijske teorije, pokazuju određena ograničenja. Sve veća količina dostupnih podataka te značajan napredak u području računalne tehnologije otvorili su vrata novim metodologijama, među kojima se ističu algoritmi strojnog učenja. Duboko učenje, kao podskup strojnog učenja, koristi umjetne neuronske mreže s višeslojnim strukturama koje su sposobne modelirati složene nelinearne odnose u podacima. Primjena modela dubokog učenja u financijama, između ostalog za problem optimizacije portfelja, postala je sve popularnija zbog njihovih sposobnosti analize velikih količina složenih podataka i identifikacije suptilnih obrazaca koji mogu biti nevidljivi tradicionalnim metodama [2].

Ovaj diplomski rad za cilj ima implementaciju modela dubokog učenja koji iz podataka uče optimalne alokacije imovina temeljem Sharpeovog omjera, odnosno omjera povrata i rizika portfelja. Nakon teorijskog uvoda u problem optimizacije portfelja i područje dubokog učenja, bit će predstavljena implementacija modela te analizirani rezultati na povijesnim tržišnim podacima, uz usporedbu s rezultatima *mean-variance* optimizacije.

2. Problem optimizacije portfelja

2.1. Statistička svojstva financijskih vremenskih nizova

Financijske vrijednosnice predstavljaju imovinu kojom se trguje na burzama na kojima cijenu određuju tržišni sudionici. To mogu biti dionice, obveznice, opcije, ETF-ovi (engl. *exchange traded funds*) itd. Tržišna cijena vrijednosnice trenutna je cijena po kojoj ona može biti kupljena ili prodana, a uvjetovana je ponudom i potražnjom za tom vrijednosnicom. Kako bi se mogla pratiti cijena vrijednosnice, potrebno ju je uzorkovati u vremenu, primjerice svaki dan, i tako kreirati vremenski niz koji je zatim moguće detaljnije analizirati. Pri uzorkovanju na dnevnoj bazi najčešće se prate zaključna cijena (engl. *closing price*) i prilagođena zaključna cijena (engl. *adjusted closing price*). Zaključna cijena zadnja je cijena po kojoj se trguje financijskim instrumentom, dok prilagođena zaključna cijena još uzima u obzir dividende i podjelu dionica (engl. *stock split*). Slika 2.1 prikazuje graf kretanja prilagođene zaključne cijene za SPDR S&P 500 ETF (SPY) u zadnjih 20 godina.

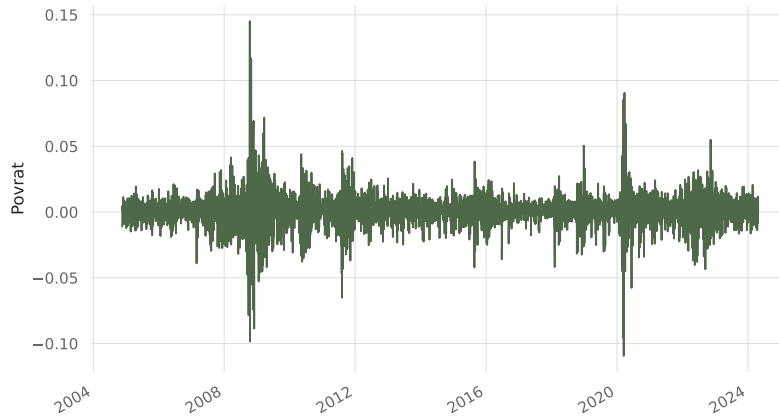


Slika 2.1: Prilagođena zaključna cijena SPDR S&P 500 ETF-a (SPY) u periodu između 2004. i 2024. godine

Stopa povrata, ili povrat imovine, definira se kao omjer razlike između vrijednosti imovine u promatranom i prošlom trenutku te vrijednosti imovine u prošlom trenutku:

$$r_i(t) = \frac{p(t) - p(t-1)}{p(t-1)} = \frac{p(t)}{p(t-1)} - 1. \quad (2.1)$$

Na slici 2.2 prikazani su dnevni povрати SPDR S&P 500 ETF-a (SPY) u zadnjih 20 godina.



Slika 2.2: Dnevni povрати SPDR S&P 500 ETF-a (SPY) u periodu između 2004. i 2024. godine

Pri donošenju investicijskih odluka, najbitnije mjere su očekivani povrat i njegova standardna devijacija, to jest volatilitnost [3]. Očekivani povrat označavamo s:

$$\mu = E[r_i(t)], \quad (2.2)$$

$$E[r_i(t)] = \frac{1}{T} \sum_{t=1}^T r_i(t). \quad (2.3)$$

Rizičnost očekivane stope povrata mjerimo standardnom devijacijom:

$$\sigma = \sqrt{E[r_i^2(t)] - \mu^2}, \quad (2.4)$$

a također se definira varijanca kao kvadrat standardne devijacije: σ^2 . Niža volatilitnost označava manji rizik za investitora.

Ako raspoložemo portfeljem koji se sastoji od n vrijednosnica, njegov očekivani povrat računa se kao vagani prosjek očekivanih prinosa pojedinih investicija u portfelju:

$$E[r_p] = \sum_{i=1}^n w_i \cdot E[r_i]. \quad (2.5)$$

S w_i označavamo udio ulaganja i u portfelju. Za portfelj bez kratkih pozicija vrijedi:

$$\sum_{i=1}^n w_i = 1, \quad (2.6)$$

$$w_i \geq 0, i = 1, \dots, n. \quad (2.7)$$

Varijanca portfelja sastavljenog od n vrijednosnica dana je formulom:

$$\sigma_p^2 = \sum_{i=1}^n w_i^2 \sigma_i^2 + \sum_{\substack{i,j=1 \\ i \neq j}}^n w_i w_j \sigma_{i,j}, \quad (2.8)$$

gdje se kovarijanca prinosa $\sigma_{i,j}$ za dvije investicije i i j definira kao:

$$\sigma_{i,j} = E[(r_i - E[r_i]) \cdot (r_j - E[r_j])]. \quad (2.9)$$

Zbog kompaktnosti, za definiranje očekivanog povrata E i varijance portfelja V možemo koristiti matrični zapis:

$$E = \mathbf{w}^T \boldsymbol{\mu}, \quad (2.10)$$

$$V = \mathbf{w}^T \boldsymbol{\Sigma} \mathbf{w}, \quad (2.11)$$

gdje je \mathbf{w} vektor udjela pojedinačnih vrijednosnica u portfelju, a $\boldsymbol{\Sigma}$ matrica kovarijanci:

$$\mathbf{w} = (w_1, \dots, w_n)^T, \quad (2.12)$$

$$\boldsymbol{\mu} = (\mu_1, \dots, \mu_n)^T, \quad (2.13)$$

$$\boldsymbol{\Sigma} = \begin{bmatrix} \sigma_{11} & \cdots & \sigma_{1n} \\ \vdots & \ddots & \vdots \\ \sigma_{n1} & \cdots & \sigma_{nn} \end{bmatrix}. \quad (2.14)$$

2.2. Mean-variance optimizacija

Mean-variance optimizacija dio je Markowitzeve moderne teorije portfelja predstavljene u [1]. Moderna teorija portfelja (engl. *modern portfolio theory*, MPT) pretpostavlja da su povrati vrijednosnica normalno distribuirani te da preference prema različitim investicijama ovise samo o očekivanom povratu i njegovoj varijanci [4]. Također, pretpostavlja se da su investitori averzivni prema riziku, odnosno da će pri izboru između dvije investicije jednakog očekivanog prinosa odabrati onu manje rizičnosti. Markowitzev model prvi je kvantitativno opisao rizik te pokazao da vrijedi diverzificirati, odnosno ulagati u portfelj vrijednosnica, kao i kako taj portfelj optimizirati [5].

Optimizacija portfelja ostvaruje se minimizacijom rizika portfelja za zadani očekivani prinos r :

$$\begin{aligned} & \min_{\mathbf{w}} \quad \mathbf{w}^\top \Sigma \mathbf{w} \\ \text{uz ograničenja} \quad & \begin{cases} \mathbf{w}^\top \boldsymbol{\mu} = r, \\ \mathbf{w}^\top \mathbf{1} = 1. \end{cases} \end{aligned} \quad (2.15)$$

Ako promatramo portfelje bez kratkih pozicija, dodatno ograničenje je nenegativnost udjela vrijednosnica (2.7).

Navedeni optimizacijski problem ima kvadratnu funkciju cilja i linearna ograničenja zbog čega ga nazivamo kvadratičnim programom [6]. Cilj kvadratičnog programiranja je pronaći točku koja minimizira ciljnu funkciju, a ujedno i zadovoljava sva zadana ograničenja. Ako je funkcija koju optimiramo konveksna, onda postoji jedinstveno rješenje. U navedenom problemu optimizacije portfelja kovarijacijska matrica Σ treba biti pozitivno semidefinitna kako bi ciljna funkcija bila konveksna odnosno treba vrijediti:

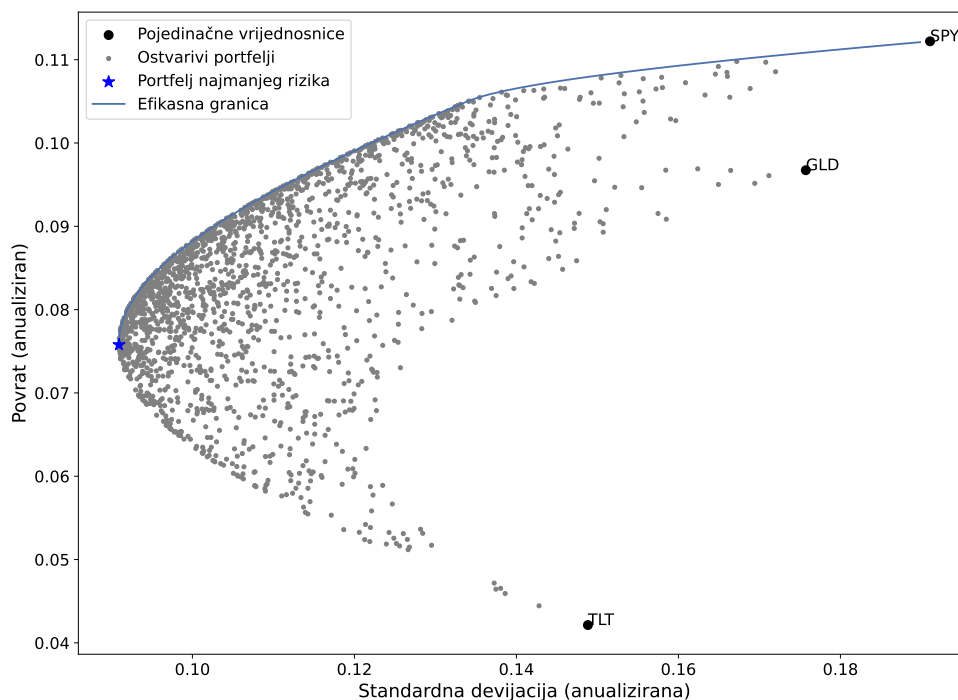
$$\mathbf{w}^\top \Sigma \mathbf{w} \geq 0. \quad (2.16)$$

Ako ovaj uvjet nije zadovoljen, pronalazak rješenja značajno je teži jer funkcija može imati nekoliko stacionarnih točaka i lokalnih minimuma [7].

2.2.1. Efikasna granica

Rješavanjem optimizacijskog problema (2.15) za svaki zadani očekivani povrat r dobivamo portfelj s najmanjom varijancom. Svi takvi portfelji čine skup minimalne varijance, a oni koji imaju očekivani prinos veći ili jednak očekivanom prinosu portfelja minimalne varijance tvore krivulju koja se naziva efikasna granica. Portfelji koji se nalaze na efikasnoj granici zovu se efikasni portfelji.

Na slici 2.3 nalazi se primjer efikasne granice za portfelje koji se sastoje od 3 vrijednosnice, konkretno ETF-ova SPY, GLD i TLT (podatci od 2004. do 2024. godine), koji će biti korišteni u implementacijskom dijelu rada. Simulirano je 2000 nasumičnih ostvarivih portfelja, odnosno onih koji zadovoljavaju 2.6, a prikazano je i gdje se ove vrijednosnice pojedinačno nalaze na rizik-povrat grafu. Efikasna granica označena je plavom krivuljom.



Slika 2.3: Primjer efikasne granice za portfelje koji se sastoje od 3 ETF-a, SPY, GLD i TLT (podatci od 2004. do 2024. godine), 2000 nasumično simuliranih ostvarivih portfelja te pojedinačne vrijednosnice na rizik-povrat grafu

Općenito, dva značajna portfelja koja se nalaze na efikasnoj granici su portfelj minimalne varijance i portfelj maksimalnog Sharpeovog omjera. U sklopu ovog rada detaljnije će se promatrati potonji.

2.2.2. Portfelj maksimalnog Sharpeovog omjera

Portfelj maksimalnog Sharpeovog omjera maksimizira omjer prinosa portfelja iznad nerizične investicije i njegovog rizika [8]:

$$S = \frac{\mu_p - r_{rf}}{\sigma_p}, \quad (2.17)$$

pri čemu je μ_p očekivani povrat portfelja, r_{rf} je stopa povrata nerizične investicije, a σ_p je standardna devijacija prinosa portfelja. Ekonomist William F. Sharpe definirao je Sharpeov omjer 1966. [9] u sklopu njegovog rada na modelu za vrednovanje kapitalne imovine (engl. *Capital Asset Pricing Model*, CAPM), koji je ekstenzija spomenute Markowitzove teorije portfelja. Prema CAPM-u svi investitori teže portfelju koji se sastoji od tangencijalnog portfelja i nerizične investicije [5]. Tangencijalni portfelj je portfelj rizične imovine koji maksimizira Sharpeov omjer. Ako ga promatramo na rizik-povrat grafu, predstavlja diralište efikasne granice i pravca tržišta kapitala (engl.

capital market line, CML), odnosno pravca s koeficijentom smjera jednakog Sharpeovom omjeru. Jednadžba pravca tržišta kapitala glasi:

$$\mu = r_{rf} + \frac{\mu_p - r_{rf}}{\sigma_p} \sigma. \quad (2.18)$$

Optimizacijski problem čijim se rješavanjem dolazi do portfelja maksimalnog Sharpeovog omjera glasi [3]:

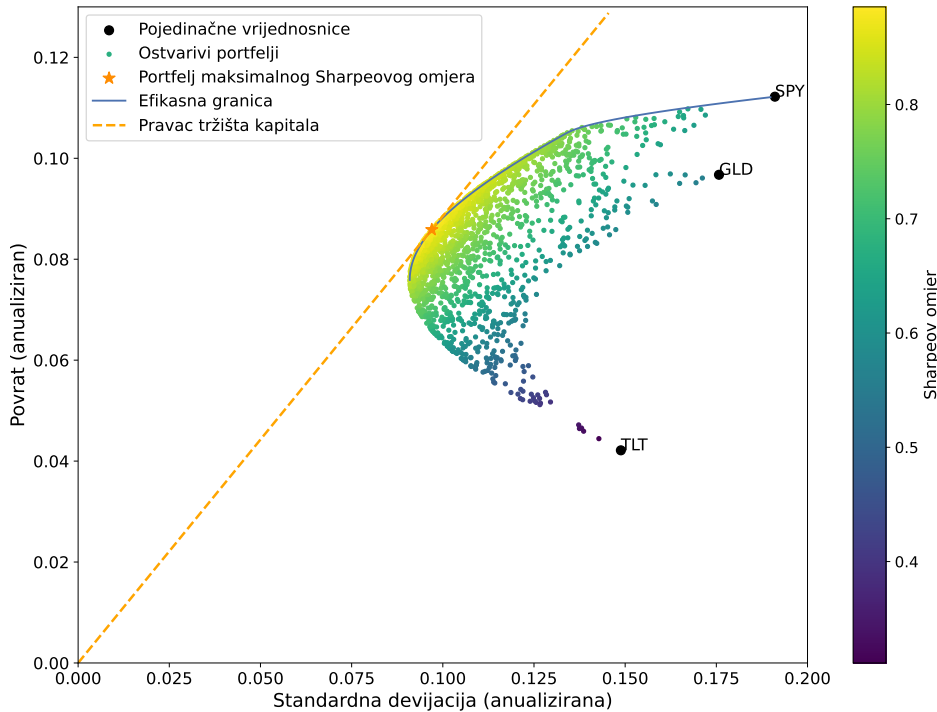
$$\max_{\mathbf{w}} \frac{\mathbf{w}^\top \boldsymbol{\mu} - r_{rf}}{\sqrt{\mathbf{w}^\top \boldsymbol{\Sigma} \mathbf{w}}} \quad (2.19)$$

uz ograničenje $\mathbf{w}^\top \mathbf{1} = 1$.

Rješavanjem problema 2.19 Lagrangeovom metodom dobivamo sljedeće analitičko rješenje:

$$\mathbf{w} = \frac{\boldsymbol{\Sigma}^{-1}(\boldsymbol{\mu} - r_{rf}\mathbf{1})}{\mathbf{1}^\top \boldsymbol{\Sigma}^{-1}(\boldsymbol{\mu} - r_{rf}\mathbf{1})}, \quad (2.20)$$

ali zahtijeva da postoji inverz kovarijacijske matrice $\boldsymbol{\Sigma}^{-1}$. Kako bi inverz postojao, matrica $\boldsymbol{\Sigma}$ treba biti pozitivno definitna, što će biti slučaj kada imamo dovoljno podataka, to jest kada vrijedi $T \geq N$, pri čemu je T duljina vremenskog perioda podataka. Na slici 2.4 označen je portfelj koji maksimizira Sharpeov omjer te je za ostale ostvarive portfelje istaknut njihov Sharpeov omjer.



Slika 2.4: Prikaz efikasne granice, pravca tržišta kapitala, portfelja koji maksimizira Sharpeov omjer i ostvarivih portfelja s istaknutim njihovim Sharpeovim omjerom (zanemarena stopa povrata nerizične investicije)

2.2.3. Nedostatci *mean-variance* optimizacije

Iako je *mean-variance* optimizacija popularna metoda za konstrukciju portfelja, pretpostavke na kojima se temelji su kritizirane jer često nisu zadovoljene u stvarnim financijskim tržištima [10]. Konkretno, pretpostavlja se da povrati prate normalnu distribuciju, a pokazano je, primjerice u [11], da distribucija povrata ima teže repove od Gaussove, što čini standardnu devijaciju kao mjeru rizika neprimjerenom.

Također, rezultat *mean-variance* optimizacije uvelike ovisi o podacima koje koristimo, odnosno očekivanim povratima i kovarijacijskoj matrici. Povijesni podatci, na koje se optimizacija oslanja, nužno ne predstavljaju točno buduće trendove. Posebice je teško predvidjeti povrate iz povijesnih podataka zato što nisu autokorelirani [12]. Za razliku od povrata, njihove varijance i kovarijance jesu autokorelirane te možemo pretpostaviti da će vrijediti u određenom budućem periodu. Ipak, korelacije se mogu promijeniti, pogotovo u trenucima nestabilnosti financijskih tržišta, zbog čega vremenski prozori za procjenu kovarijacijske matrice mogu biti ograničene veličine [12]. Ako je duljina vremenskog prozora manja od broja vrijednosnica u portfelju, kovarijacijska matrica nije pozitivno definitna. Tada se za procjenu kovarijacijske matrice mogu koristiti regularizacijske metode, neke od kojih su Ledoit i Wolf predstavili u [13].

Navedeni nedostatci, a posebice problem procjene očekivanih povrata, daju motivaciju za razvijanje alternativnih heuristika za optimizaciju portfelja, primjerice korištenjem dubokih modela.

3. Duboko učenje

Duboko učenje grana je strojnog učenja koja svoju primjenu pronalazi u izazovnim područjima s velikom dimenzionalnošću podataka: računalnom vidu, obradi prirodnog jezika, financijama itd. Duboko učenje temelji se na predstavljanju podataka složenim reprezentacijama do kojih se dolazi slijedom naučenih nelinearnih transformacija uz pomoć neuronskih mreža [14].

Osnovna gradivna jedinica neuronske mreže je neuron. Za dani ulazni vektor x izlaz neurona o definiran je kao:

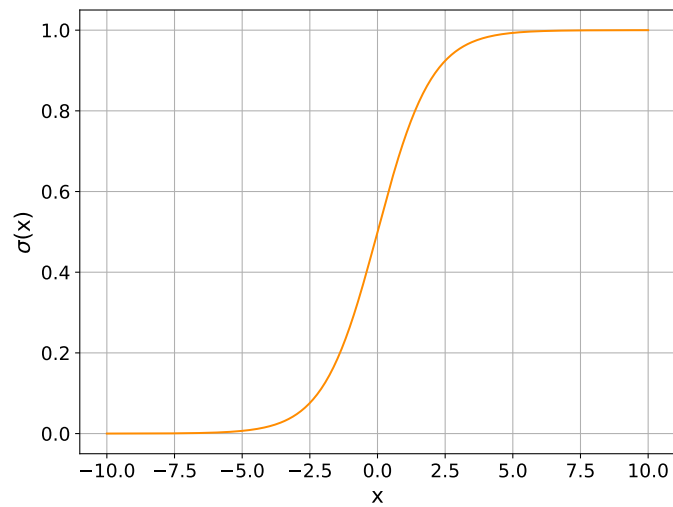
$$o = f\left(\sum_{i=1}^n w_i x_i + b\right), \quad (3.1)$$

gdje su w_i težine neurona koje određuju u kojoj mjeri svaka komponenta ulaznog vektora x pobuđuje neuron, dok je b pristranost (engl. *bias*) neurona. Oznaka f predstavlja aktivacijsku funkciju. Njome se unosi nelinearnost u model. Neke od najpoznatijih aktivacijskih funkcija su sigmoida, hiperbolni tangens i *softmax*. Sve navedene funkcije su nelinearne, monotone i kontinuirano diferencijabilne, što su bitna svojstva za učenje neuronskih mreža, posebice diferencijabilnost.

Sigmoida je funkcija čija je kodomena interval $(0, 1)$. Njena jednažba glasi:

$$\sigma(x) = \frac{1}{1 + e^{-x}}, \quad (3.2)$$

a graf je prikazan na slici 3.1.

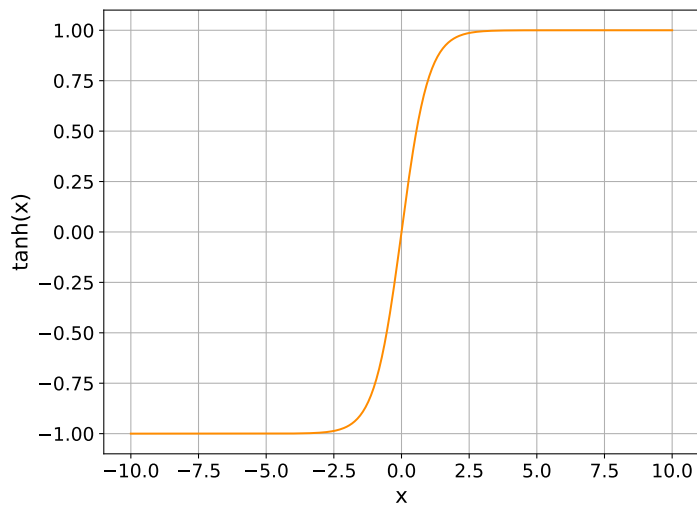


Slika 3.1: Graf sigmoide

Hiperbolni tangens također je sigmoidalna funkcija (u obliku slova S), čija kodomena je interval $\langle -1, 1 \rangle$. Njena jednađba glasi:

$$\tanh(x) = \frac{e^{2x} - 1}{e^{2x} + 1}, \quad (3.3)$$

a pripadni graf nalazi se na slici 3.2.

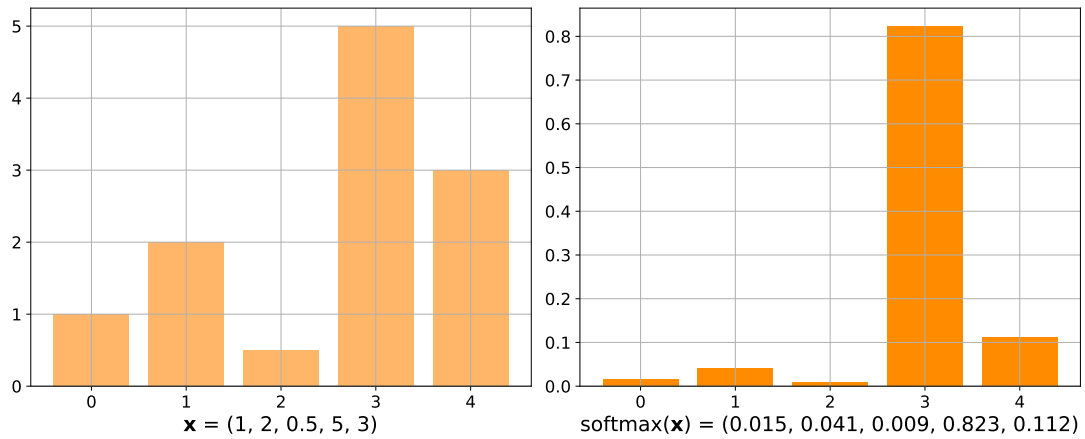


Slika 3.2: Graf hiperbolnog tangensa

Softmax funkcija najčešće se koristi u zadnjem sloju neuronske mreže za probleme višeklasne klasifikacije. Ova funkcija prima K -dimenzionalni vektor x te ga preslikava

u K -dimenzionalni vektor čije se komponente zbrajaju u 1, a k -ta komponenta izlaznog vektora jednaka je:

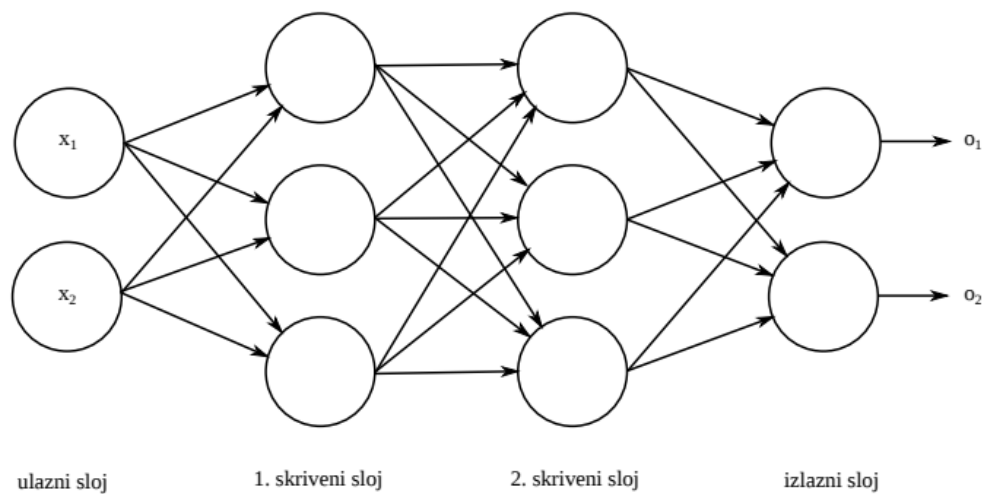
$$\text{softmax}(\mathbf{x})_k = \frac{e^{x_k}}{\sum_j e^{x_j}}. \quad (3.4)$$



Slika 3.3: Primjer djelovanja *softmax* funkcije na vektor \mathbf{x}

Osnovna neuronska mreža sastoji se od tri vrste slojeva neurona:

- prvi, ulazni sloj,
- proizvoljan broj skrivenih slojeva,
- posljednji, izlazni sloj.



Slika 3.4: Slojevita mreža arhitekture $2 \times 3 \times 3 \times 2$ [15]

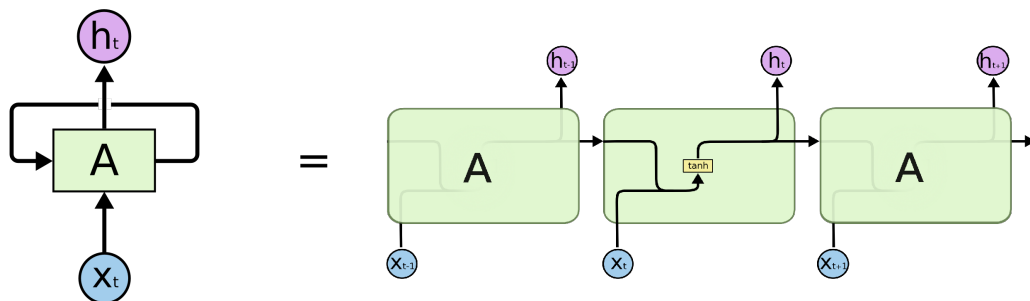
Neke od najpoznatijih arhitektura dubokih modela su:

- duboka unaprijedna mreža (engl. *deep feedforward neural network*),
- konvolucijska neuronska mreža (engl. *convolutional neural network*, CNN),
- povratna neuronska mreža (engl. *recurrent neural network*, RNN).

Odabir arhitekture dubokog modela uvelike ovisi o podacima kojima raspolažemo. U ovom radu detaljnije će se promatrati vrsta povratne neuronske mreže: mreža s povratnim ćelijama s dugoročnom memorijom (*long short-term memory*), ili kraće LSTM mreža, koja se pokazala kao najbolja za modeliranje dnevnih financijskih podataka [10].

3.1. Povratne neuronske mreže

Povratne neuronske mreže su modeli posebno prilagođeni za obradu sekvencijalnih podataka. Za razliku od unaprijednih mreža kod kojih podatci putuju u samo jednom smjeru, od ulaznog sloja prema izlaznom, u povratnim neuronskim mrežama oni dolaze i iz prethodnog vremenskog koraka. Ova sposobnost povratnih neuronskih mreža da pamte prethodne podatke omogućena je povratnim vezama.



Slika 3.5: Prikaz osnovne povratne neuronske mreže [16]

Na slici 3.5 je "odrolavanjem" prikazano kako podatak iz prethodnog trenutka utječe na trenutni izlaz. A označava RNN ćeliju koja predstavlja skriveno stanje, koje se tipično aktivira tangensom hiperbolnim:

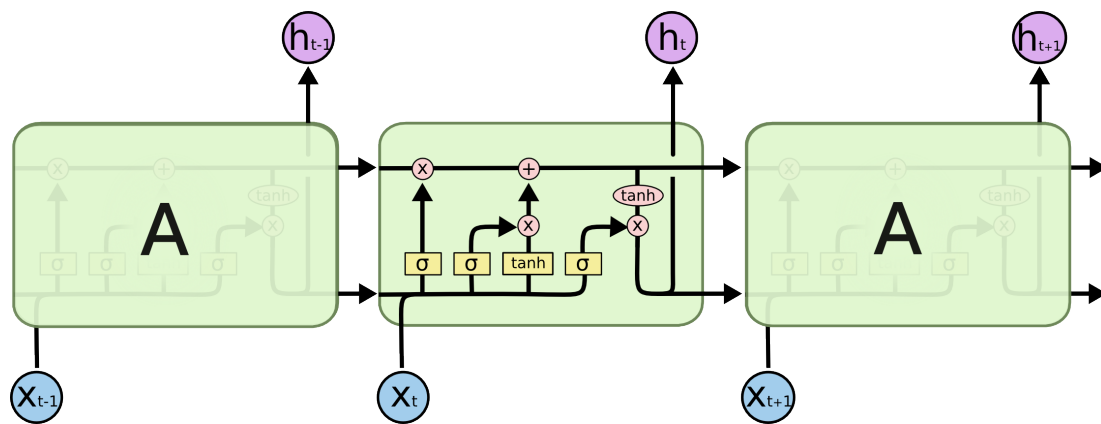
$$h_t = \tanh(W_{hh}h_{t-1} + W_{xh}x_t + b_h), \quad (3.5)$$

pri čemu su W_{hh} , W_{xh} , b_h parametri modela. Matrica W_{xh} projicira ulaz u prostor reprezentacije stanja, dok matrica W_{hh} modelira evoluciju stanja [17].

Glavni nedostatak povratnih modela je što konzistentno podbacuju pri učenju dugih veza [17]. Razlog tome je što su gradijenti povratnih modela podložni numeričkoj nestabilnosti. Zbog dijeljenja parametara u uzastopnim operacijama, može doći do iščezavanja ili eksplozije gradijenata.

3.1.1. LSTM mreže

LSTM mreža predstavljena je 1997. u članku Hochreitera i Schmidhubera [18] kao rješenje problema iščezavajućeg i eksplodirajućeg gradijenta kod povratnih neuronskih mreža. Predložena je LSTM ćelija s arhitekturom ilustriranom na slici 3.6.



Slika 3.6: LSTM ćelija (na slici označena s A) [16]

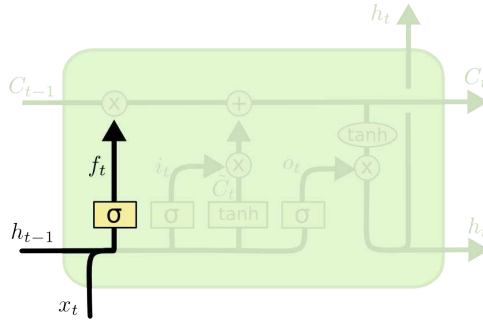
Kod LSTM-a, skriveno stanje h_t koristi se samo za računanje izlaza te se uvodi stanje ćelije C_t koje samo pamti viđenu informaciju.

LSTM ćelija ima sposobnost dodavanja i uklanjanja informacija iz stanja ćelije uz pomoć struktura zvanih vrata (engl. *gates*). Postoje tri vrste vrata: vrata zaboravljanja (*forget gate*), ulazna vrata (engl. *input gate*) i izlazna vrata (engl. *output gate*). Prilikom opisa navedenih vrata u nastavku, koristit će se oznake W_{hh}, W_{xh}, b_h za parametre koji pripadaju vratima (primjerice vratima zaboravljanja pripadaju parametri: W_{fhh}, W_{fxh}, b_{fh})

Vrata zaboravljanja

Vrata zaboravljanja služe za odbacivanje prošlih informacija iz stanja ćelije. Za to koristimo sigmoidalnu funkciju koja na temelju prošlog izlaza h_{t-1} i trenutnog ulaza x_t daje realan broj između 0 i 1 koji određuje u kojoj mjeri će prošlo stanje biti zaboravljeno:

$$f_t = \sigma(W_{fhh}h_{t-1} + W_{fxh}x_t + b_{fh}). \quad (3.6)$$



Slika 3.7: Vizualizacija vrata zaboravljanja [16]

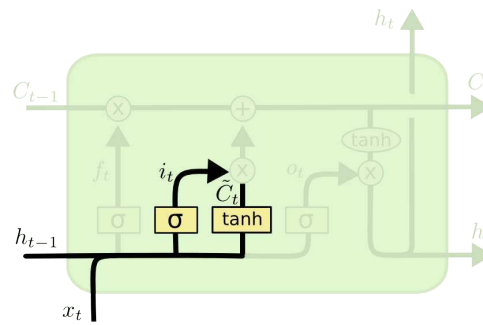
Ulazna vrata

Ulazna vrata služe za odluku koje nove informacije će se pohraniti u stanje ćelije. Odabir novih informacija odvija se u dva koraka. Prvo sigmoidalni sloj odlučuje koje će vrijednosti biti ažurirane:

$$i_t = \sigma(W_{ihh}h_{t-1} + W_{ixh}x_t + b_{ih}). \quad (3.7)$$

Zatim sloj tangensa hiperbolnog kreira vektor koji predstavlja nove informacije:

$$\tilde{C}_t = \tanh(W_{chh}h_{t-1} + W_{cxh}x_t + b_{ch}). \quad (3.8)$$



Slika 3.8: Vizualizacija ulaznih vrata [16]

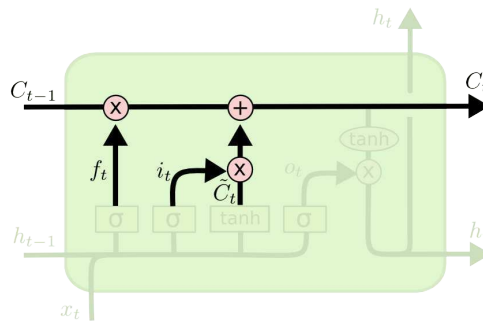
Ažuriranje stanja ćelije

Stanje ćelije ažurira se tako da se staro stanje ćelije pomnoži s f_t čime se odabrane stare informacije zaborave te se to pribroji umnošku i_t i \tilde{C}_t čime se stanju dodaju nove informacije:

$$C_t = f_t \odot C_{t-1} + i_t \odot \tilde{C}_t, \quad (3.9)$$

gdje je \odot oznaka za Hadamardov umnožak, odnosno tenzorsko množenje po elementima.

Upravo zbog ograničavanja f_t i i_t na interval $\langle 0, 1 \rangle$ u sigmoidalnim slojevima, elimini-
rani su eksplodirajući i nestajući gradijenti pri ažuriranju stanja ćelije [17].



Slika 3.9: Vizualizacija ažuriranja stanja ćelije [16]

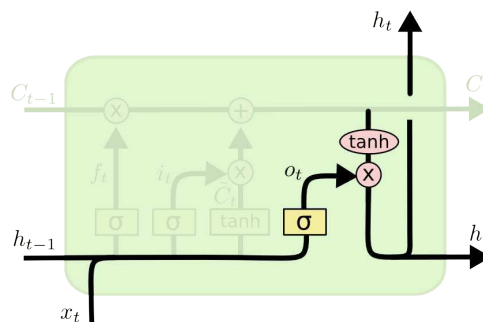
Izlazna vrata

Na kraju je potrebno odlučiti što će biti izlaz iz LSTM ćelije. Izlaz će ovisiti o stanju ćelije, ali je filtriran izlazom iz sigmoide koji odlučuje koji dijelovi stanja ćelije će se nalaziti u izlazu:

$$o_t = \sigma(W_{ohh}h_{t-1} + W_{oxh}x_t + b_{oh}). \quad (3.10)$$

Zatim stanje ćelije prolazi kroz tangens hiperbolni koji vrijednosti ograničava na inter-
val $[-1, 1]$ te se množi s izlazom sigmoide:

$$h_t = o_t \odot \tanh(C_t). \quad (3.11)$$



Slika 3.10: Vizualizacija izlaznih vrata [16]

3.2. Učenje dubokih modela

Svaki algoritam strojnog učenja ima tri glavne komponente: model, funkciju gubitka i optimizacijski postupak [19]. Model se definira kao skup funkcija h parametriziranih vektorom parametara θ :

$$\mathcal{H} = \{h(\mathbf{x}; \theta)\}_{\theta}. \quad (3.12)$$

Nakon odabira vrste modela slijedi faza učenja modela, kada se modelu predočavaju uzorci iz skupa za učenje s ciljem pronalaska optimalnih vrijednosti za θ . Da bi to postigli, potrebno je definirati što želimo da model nauči. U tu svrhu, definira se funkcija pogreške E i pripadna funkcija gubitka L (engl. *loss function*):

$$E(\theta|\mathcal{D}) = \frac{1}{N} \sum_{i=1}^N L(y^{(i)}, h(\mathbf{x}^{(i)}; \theta)), \quad (3.13)$$

gdje \mathcal{D} predstavlja skup označenih primjera, a $y^{(i)}$ točnu oznaku primjera $\mathbf{x}^{(i)}$. Funkcija gubitka govori koliko se dobiveni izlaz modela razlikuje od željenog izlaza. Neke od često korištenih funkcija pogreške su srednja kvadratna pogreška i unakrsna entropija, ali također je moguće i definirati vlastitu funkciju pogreške. Učenje modela optimizacijski je problem čiji je cilj pronalazak funkcije h^* , odnosno parametara θ^* koji minimiziraju pogrešku:

$$\theta^* = \underset{\theta}{\operatorname{argmin}} E(\theta|\mathcal{D}). \quad (3.14)$$

Iterativni optimizacijski postupak koji se najčešće koristi kod učenja dubokih modela naziva se gradijentni spust (engl. *gradient descent*) [20]. Ideja gradijentnog spusta je da se, krenuvši od neke početne točke, konkretno početnog vektora θ , postepeno "spuštamo" niz površinu funkcije $E(\theta|\mathcal{D})$ u smjeru suprotnom of gradijenta u točki θ te tako iterativno dolazimo do minimuma funkcije [21]. Vektor parametara θ u svakoj iteraciji gradijentnog spusta ažurira se na sljedeći način:

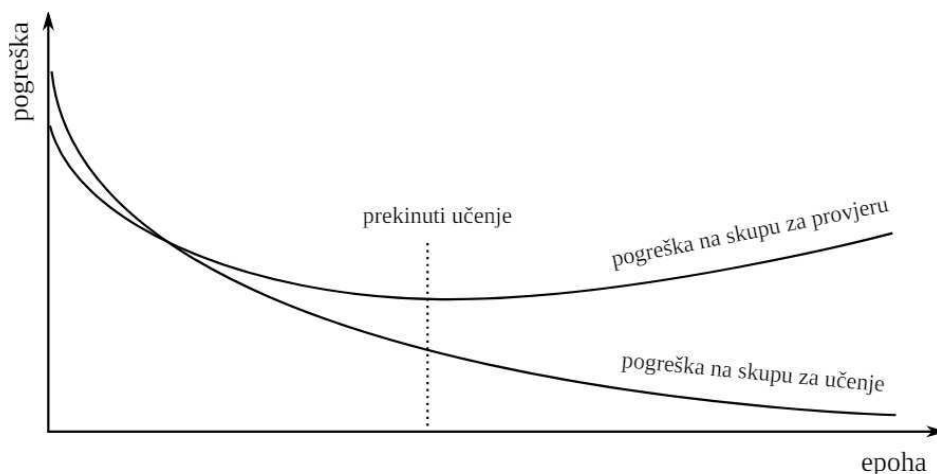
$$\theta \leftarrow \theta - \eta \nabla E(\theta|\mathcal{D}), \quad (3.15)$$

gdje je η stopa učenja koja određuje veličinu koraka koji se radi spuštajući se do minimuma funkcije. Stopa učenja je hiperparametar i potrebno ju je pažljivo odabrati tako da ne bude ni prevelika ni premala. Prevelika stopa učenja može uzrokovati divergenciju postupka, dok će u slučaju premale stope učenja postupak konvergirati, ali sporo [21]. Algoritam propagacije greške unatrag (engl. *backpropagation*) koristi se za učinkovit izračun svih parcijalnih derivacija i njihovu primjenu za određivanje iznosa kojim korigiramo svaki od parametara. Postoje tri vrste gradijentnog spusta prema načinu ažuriranja parametara [21]:

- standardni ili grupni gradijentni spust – parametri se ažuriraju na temelju svih primjera iz skupa za učenje,
- stohastički gradijentni spust – parametri se ažuriraju za svaki primjer zasebno,
- gradijentni spust s mini-grupama (engl. *mini-batches*) – parametri se ažuriraju za svaku grupu primjera proizvoljno odabrane veličine.

Adam je optimizacijski postupak koji unapređuje osnovni gradijentni spust tako što prilagođava stope učenja za svaki parametar na temelju procjena prvog (srednja vrijednost) i drugog (varijanca) momenta gradijenata [22]. Postoje još brojni optimizacijski postupci, i iako ne postoji konsenzus o tome koji algoritam koristiti, Adam često pokazuje dobre rezultate [20].

Kako se model u postupku učenja ne bi potpuno prilagodio podatcima na kojima je naučen, bitno je pratiti postupak i pustiti model da uči sve dok se sposobnost generalizacije popravlja. U tu svrhu, podatke je potrebno podijeliti na skup za učenje, skup za validaciju i skup za testiranje. Model učimo na skupu za učenje i nakon svake epohe (jednog prolaska kroz cijeli skup za učenje) računamo pogrešku na skupu za učenje i skupu za validaciju. Slika 3.11 prikazuje očekivano kretanje pogrešaka. Isprekidanom crtom označena je epoha u kojoj je potrebno prekinuti učenje jer je pronađen najbolji model. Ako se učenje nastavi dalje, pogreška na skupu za validaciju se povećava, odnosno model ulazi u prenaučenosť i gubi sposobnost generalizacije jer se prilagođava šumu u skupu za učenje. Suprotno tome, ako se učenje ranije zaustavi, model će biti podnaučen.



Slika 3.11: Kretanje pogrešaka pri učenju neuronske mreže [15]

Kada je postupak učenja gotov, kvalitetu rada modela provjeravamo na skupu za testiranje.

4. Implementacija

Programski kod, kojim su implementirani modeli dubokog učenja i *mean-variance* optimizacija, napisan je u programskom jeziku Python 3. Za izvedbu su korištene sljedeće biblioteke:

- Tensorflow [23] – razvoj modela strojnog učenja, najčešće neuronskih mreža
- NumPy [24] – proračuni s velikim višedimenzionalnim nizovima i matricama
- Matplotlib [25] – vizualizacija podataka
- pandas [26] – analiza i upravljanje podacima
- PyPortfolioOpt [27] – metode optimizacije portfelja

Kod je izvršavan uz pomoć besplatne inačice Google Colab servisa [28].

4.1. Korišteni skup podataka

Za treniranje i testiranje modela korišteni su podatci o dnevnim povratima 3 različita ETF-a, odnosno fondova kojima se trguje na burzi:

- SPDR S&P 500 ETF Trust (SPY) – prati S&P 500 indeks, koji objedinjuje 500 tržišno najvrijednijih dioničkih društava u SAD-u
- iShares 20+ Year Treasury Bond ETF (TLT) – prati indeks koji se sastoji državnih obveznica SAD-a s rokom dospijeca od 20 godina ili više
- SPDR Gold Shares ETF (GLD) – prati cijenu zlata

Slika 4.1 prikazuje kumulativne povrate za svaki od prethodno spomenutih ETF-ova u razdoblju od 2006. do 2024. godine.



Slika 4.1: Kumulativni povrati SPY, TLT i GLD ETF-ova od 2006. do 2024.

Uz to, korišteni su i podatci o dnevnim povratima još 3 ETF-a za dodatno testiranje modela:

- Vanguard Total Stock Market ETF (VTI) – prati CRSP US Total Market Index, koji obuhvaća cijelo tržište dionica SAD-a
- iShares Core US Aggregate Bond ETF (AGG) – prati Bloomberg Barclays U.S. Aggregate Bond Index, koji je sveobuhvatan referentni indeks za tržište obveznica SAD-a
- Invesco DB Commodity Index Tracking Fund (DBC) – prati DBIQ Optimum Yield Diversified Commodity Index, koji se sastoji od terminskih ugovora za 14 najtrgovanijih i najvažnijih fizičkih roba u svijetu

Slika 4.2 prikazuje kumulativne povrate za svaki od prethodno spomenutih ETF-ova u razdoblju od 2006. do 2024. godine.



Slika 4.2: Kumulativni povrati SPY, TLT i GLD ETF-ova od 2006. do 2024.

Dnevni povrati izračunati su pomoću formule 2.1 temeljem prilagođenih zaključnih cijena ETF-ova.

Jedna od glavnih prednosti ETF-ova mogućnost je ulaganja u širok spektar vrijednosnica čime se postiže diverzifikacija [29]. Dodatno ju je moguće ostvariti ulaganjem u različite kategorije ETF-ova za koje se očekuje niska razina međusobne koreliranosti. Iz tog razloga, za konstrukciju portfelja modelima dubokog učenja odabrani su SPY (dionički ETF), TLT (obveznički ETF) i GLD (robni ETF).

4.2. Mean-variance optimizacija

Mean-variance optimizacija implementirana je uz pomoć PyPortfolioOpt biblioteke. Klizećim prozorom veličine 50 s pomakom 1 krećemo se po vremenskim nizovima koji sadrže dnevne povrate promatranih vrijednosnica. U svakoj iteraciji petlje za prozor od 50 dana računa se srednji povrat i uzoračka kovarijacijska matrica. Temeljem tih podataka pronalazi se efikasna granica te se provodi optimizacija u svrhu pronalaska portfelja s maksimalnim Sharpeovim omjerom. Pri optimizaciji se definiraju ograničenja 2.6 i 2.7 kako bi rezultirala portfeljem sa samo dugim pozicijama. Rezultat optimizacije vektor je udjela vrijednosnica u optimalnom portfelju za svaki od vremenskih prozora.

4.3. Modeli dubokog učenja

Implementacija modela dubokog učenja izvedena je uz pomoć Tensorflow biblioteke i Keras API-ja. Model se sastoji od jednog sloja LSTM jedinica (engl. *units*) te izlaznog sloja od 3 neurona sa *softmax* aktivacijskom funkcijom. Na ulaz modela šalju se podatci dimenzije [50, 3], gdje prva dimenzija predstavlja broj prethodnih dnevnih povrata, a druga broj vrijednosnica. Na izlazu modela dobiva se vektor udjela imovina u portfelju (w_1, w_2, w_3) . Zbog djelovanja *softmax* funkcije na izlaz, dobiveni portfelji ne sadrže kratke pozicije. Kako bi se Sharpeov omjer direktno optimizirao pomoću gradijentnog spusta, potrebno je definirati vlastitu ciljnu funkciju. Po uzoru na [10] definirana je sljedeća ciljna funkcija:

$$L_T = -\frac{E(R_p)}{Std(R_p)}, \quad (4.1)$$

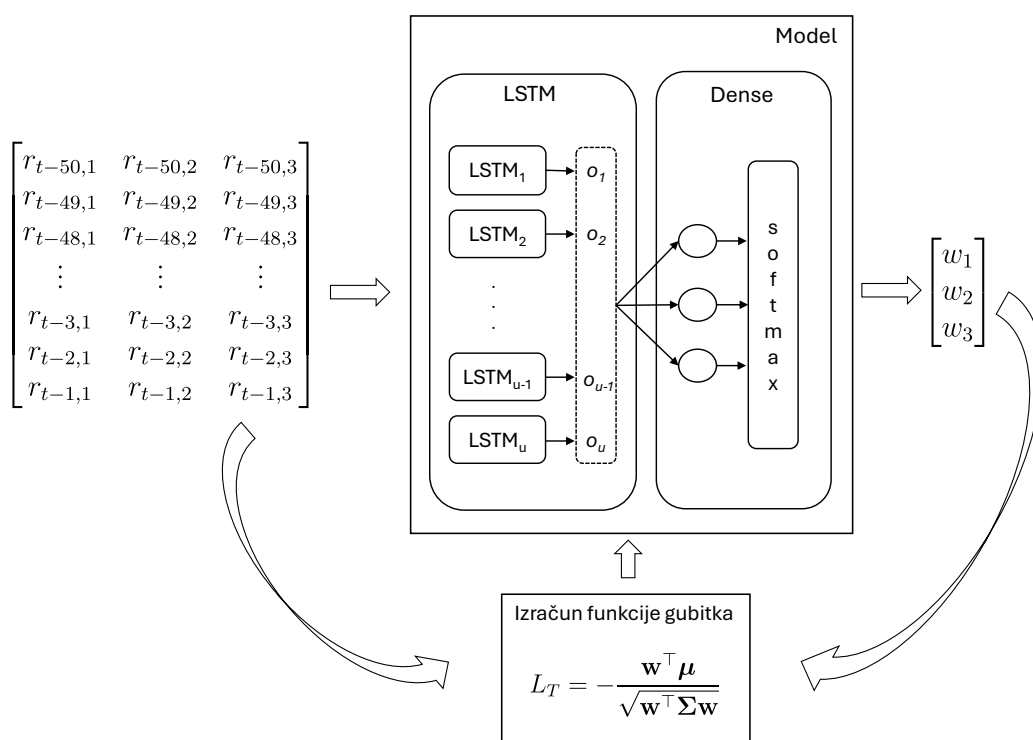
gdje su $E(R_p)$ i $Std(R_p)$ srednja vrijednost i standardna devijacija povrata portfelja. Sami izračun povrata portfelja razlikuje se za dva implementirana modela.

4.3.1. Aproksimacijski model

Kod aproksimacijskog modela, koji za cilj ima naučiti *mean-variance* optimalan portfelj, povrati portfelja se računaju tako da se ulazni podatci pomnože s udjelima dobivenim na izlazu:

$$R_p = \sum_{i=1}^{50} \sum_{n=1}^3 r_{t-i,n} \cdot w_n, \quad (4.2)$$

gdje je $r_{t-i,n}$ povrat n -te vrijednosnice u trenutku $t - i$, a w_n udio n -te vrijednosnice. Na slici 4.3 prikazana je arhitektura aproksimacijskog modela. Vidljivo je kako se za izračun funkcije gubitka koriste ulazni podatci, odnosno matrica povrata svih imovina za određeni vremenski prozor, te vektor udjela \mathbf{w} dobiven kao izlaz iz modela. Na slici se također mogu uočiti dva sloja od kojih se model sastoji: LSTM sloj koji sadrži u LSTM jedinica, te *Dense* sloj s tri potpuno povezana neurona i *softmax* aktivacijskom funkcijom.



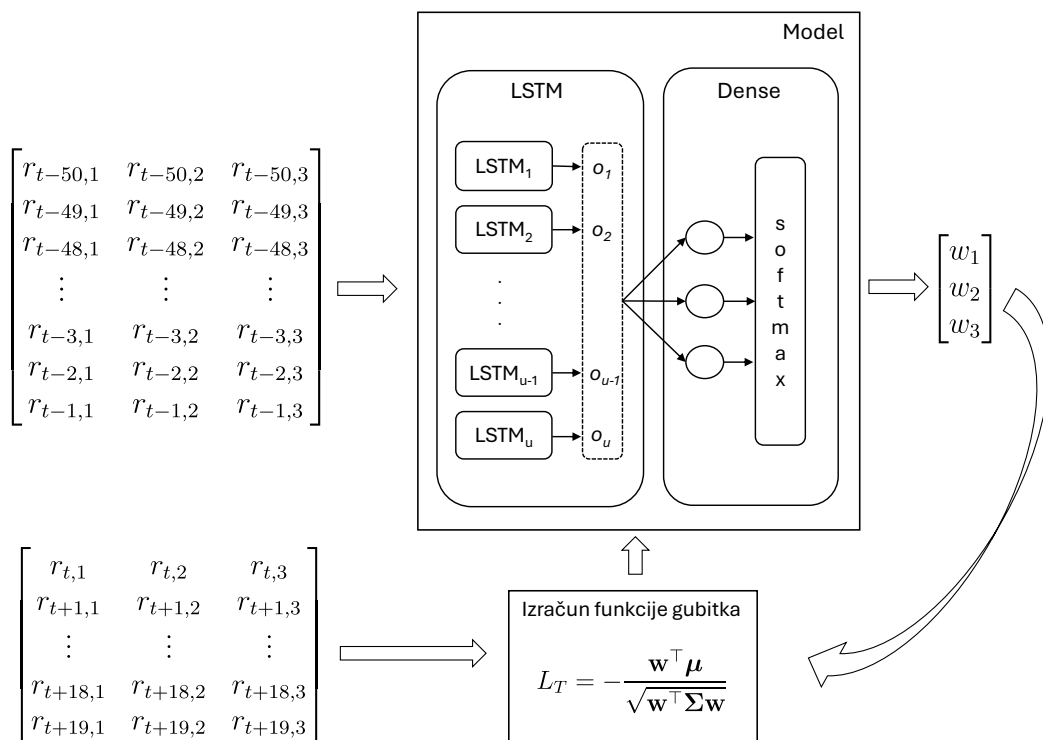
Slika 4.3: Arhitektura aproksimacijskog modela

4.3.2. Prediktivni model

Prediktivni model pokušava na temelju prethodnih povrata naučiti optimalne alokacije za određeni period u budućnosti. Stoga se za izračun povrata portfelja unutar ciljne funkcije prediktivnog modela koriste budući podatci u odnosu na ulazne podatke, konkretno 20 dana koji slijede nakon ulaznog prozora od 50 dana:

$$R_p = \sum_{i=0}^{19} \sum_{n=1}^3 r_{t+i,n} \cdot w_n, \quad (4.3)$$

pri čemu je $r_{t+i,n}$ povrat n -te vrijednosnice u trenutku $t + i$, a w_n udio n -te vrijednosnice. Na slici 4.4 prikazana je arhitektura prediktivnog modela. Vidljivo je kako se za izračun funkcije gubitka, za razliku od aproksimacijskog modela, ne koriste ulazni podatci, već povrati za budućih 20 dana u odnosu na ulazne podatke, te uz njih vektor udjela w dobiven kao izlaz iz modela. Kao i na slici 4.3, mogu se uočiti dva sloja od kojih se model sastoji: LSTM sloj koji sadrži u LSTM jedinica, te *Dense* sloj s tri potpuno povezana neurona i *softmax* aktivacijskom funkcijom.



Slika 4.4: Arhitektura prediktivnog modela

4.3.3. Detalji implementacije u Tensorflow okviru

U svrhu implementacije ciljne funkcije L_T u sklopu Tensorflow okvira, kreirana je vlastita klasa koja nasljeđuje klasu `keras.Model` te su nadjačane metode `train_step(self, data)` i `test_step(self, data)`. Definirana je metoda koja implementira ciljnu funkciju – `sharpe_loss`, a njen gradijent, koji je potreban za učenje mreže gradijentnim spustom, računa se uz pomoć `tf.GradientTape` API-ja. U procesu učenja mreže prati se Sharpeov gubitak, odnosno prosječna vrijednost negativnog neanualiziranog Sharpeovog omjera unutar epohe.

5. Rezultati

5.1. *Mean-variance* optimizacija

Provedbom *mean-variance*, kraće MV, optimizacije na način opisan u poglavlju 4.2, za svaki vremenski prozor dobivena je optimalna alokacija imovina. Za izračun povrata portfelja tijekom određenog vremenskog perioda, udjeli dobiveni optimizacijom primijenjeni su na povrate dana koji slijedi za vremenskim prozorom te su tako, pomicanjem vremenskog prozora na testnom periodu, udjeli imovina u portfelju ažurirani na dnevnoj bazi. Kao rezultat, dobiven je vremenski niz dnevnih povrata takvog portfelja iz kojeg su izračunate sljedeće metrike: očekivani povrat ($E(R)$), standardna devijacija povrata ($Std(R)$) i Sharpeov omjer. U nastavku će ovi rezultati biti korišteni kao referentna točka za testiranje rada modela dubokog učenja.

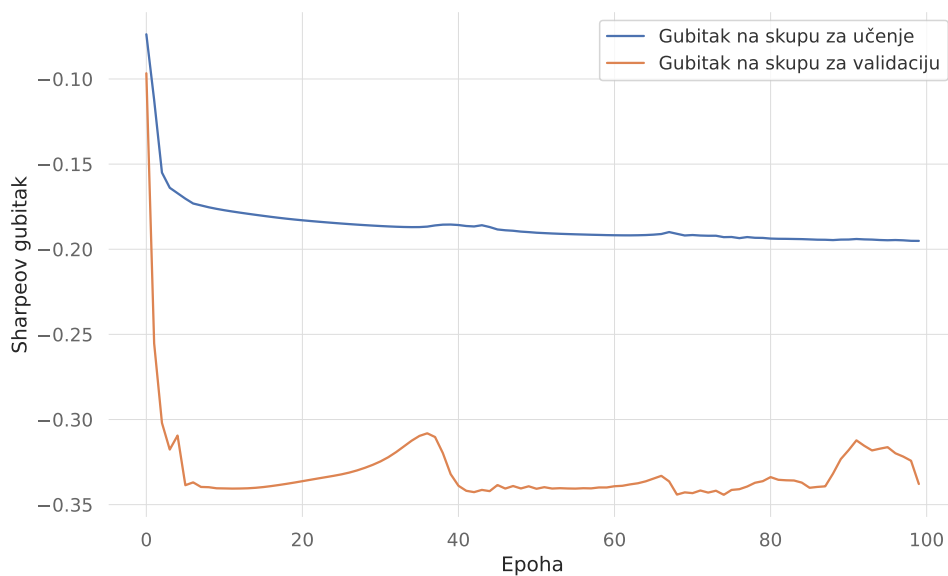
5.2. Modeli dubokog učenja

U ovom radu trenirani su modeli s jednim slojem od 32 LSTM jedinice. Isprobane su i arhitekture s većim i manjim brojem jedinica (16, 64 i 128 jedinica), no model s 32 jedinice demonstrirao je najbolje performanse. Podatci korišteni za treniranje su povrati SPY, TLT i GLD ETF-ova u postotnom obliku za period od 2006. do uključivo 2017. Od toga, 20% podataka koristi se kao skup za validaciju koji služi za optimizaciju hiperparametara i kontrolu prenaučivosti. Po uzoru na [10], za formiranje ulaza u model korišten je klizeći prozor duljine 50 s pomakom 1. Također je testirana inačica s duljinom prozora od 120 dana, no pokazala je lošije rezultate. Za učenje modela korišten je Adam optimizator, uz zadanu stopu učenja 0.001, te je proveden gradijentni spust s mini-grupama veličine 64. Modeli su trenirani na 100 epoha uz definirano rano zaustavljanje ako se Sharpeov gubitak na validacijskom skupu ne smanji nakon 50 epoha, uz pamćenje parametara modela koji daju najmanji Sharpeov gubitak na validacijskim podatcima. Valja napomenuti da se performanse modela u određenoj mjeri

razlikuju ovisno o početnom *seedu*, koji utječe na inicijalizaciju parametara mreže, te je za usporedbu modela korišten jedan nasumično odabran.

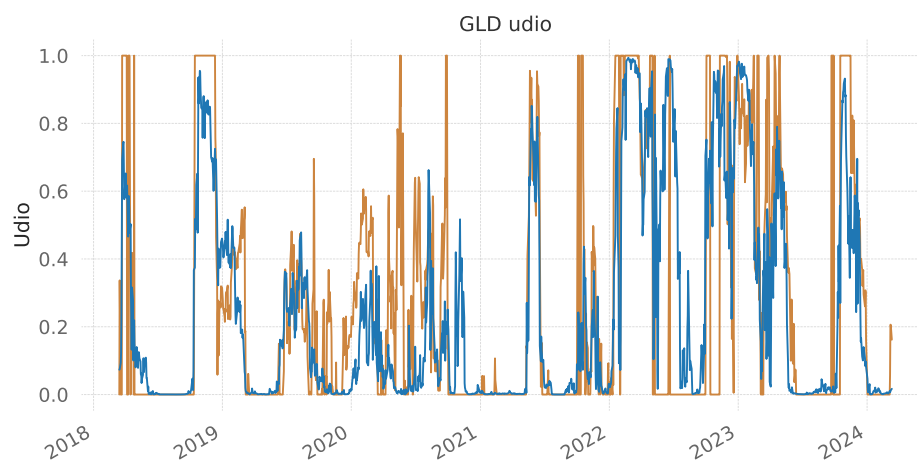
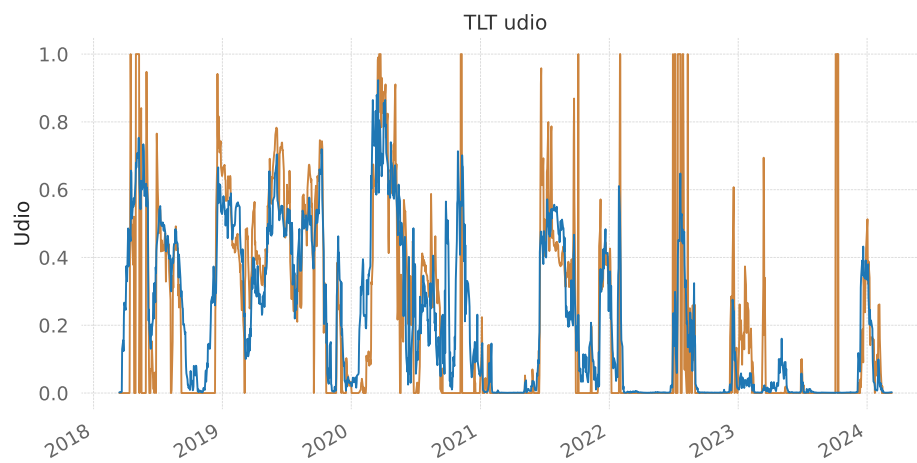
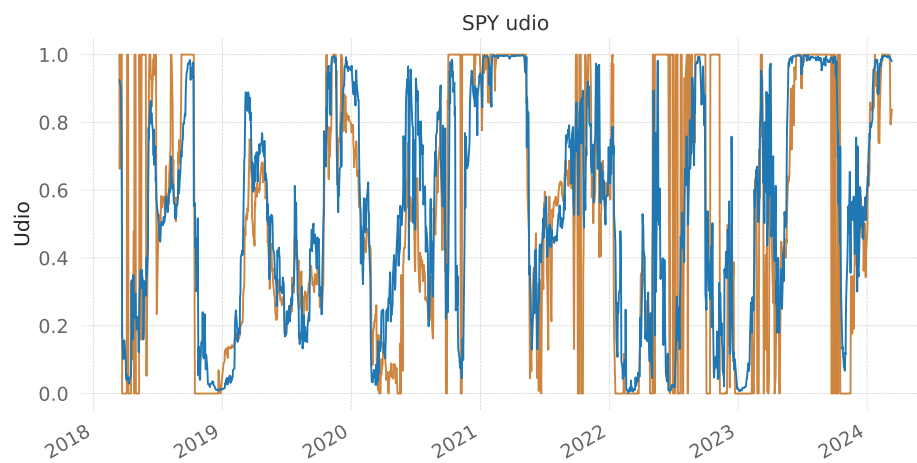
5.2.1. Aproksimacijski model

Na slici 5.1 prikazano je kretanje vrijednosti Sharpeovog gubitka na skupu za treniranje i skupu za validaciju. Vidljivo je da model vrlo brzo uspije postići nizak Sharpeov gubitak na skupu za validaciju, dok najniži gubitak postiže u 75. epohi učenja.



Slika 5.1: Kretanje vrijednosti Sharpeovog gubitka tijekom učenja aproksimacijskog modela

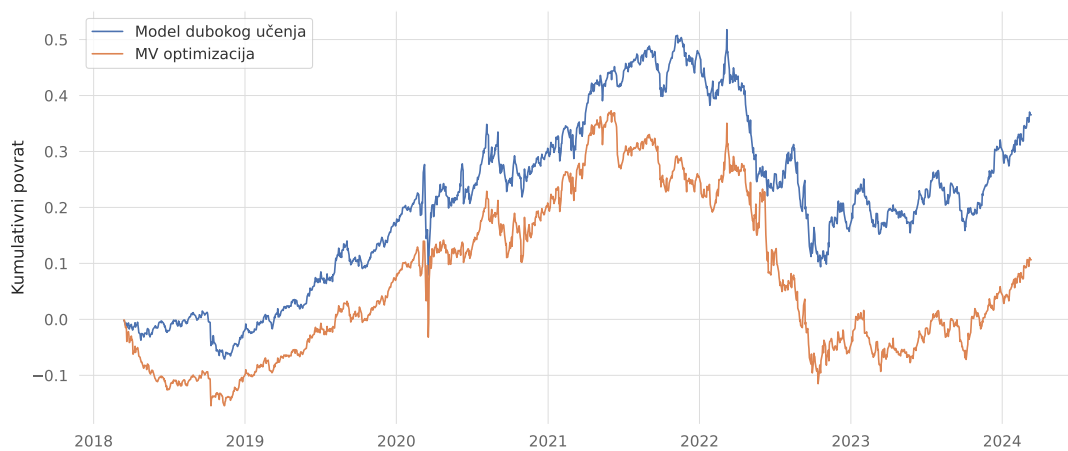
Slika 5.2 prikazuje alokacije imovina kroz testni period od 2018. do 2024., dobivene kao rezultat MV optimizacije (narančaste krivulje) i izlaza aproksimacijskog modela (plave krivulje). Vidljivo je da su rezultati ove dvije metode relativno slični, konkretno korelacija između udjela je 73.78%, ali aproksimacijski model ide u manje ekstreme.



Slika 5.2: Alokacije imovina kroz testni period od 2018. do 2024. dobivenih kao rezultat MV optimizacije (narančaste krivulje) i izlaza aproksimacijskog modela (plave krivulje)

Povrati za određeni period izračunati su način analogan onome kod MV optimizacije: udjeli koje model predviđa kao optimalne za vremenski prozor primijenjeni su na povrate dana koji slijedi, te se tako vremenski prozor pomiče za jedan dan kroz cijeli skup podataka na kojem se model testira.

Slika 5.3 prikazuje usporedbu kumulativnih povrata portfelja dobivenih MV optimizacijom i aproksimacijskim modelom na podacima za testiranje (period od 2018. do 2024. godine). Moguće je uočiti da iako povrati portfelja dobivenog aproksimacijskim modelom u velikoj mjeri prate povrate portfelja dobivenog MV optimizacijom (korelacija povrata je 85.29%), ipak su konzistentno veći od povrata MV portfelja.



Slika 5.3: Usporedba kumulativnih povrata portfelja (SPY, TLT, GLD) dobivenih MV optimizacijom i aproksimacijskim modelom

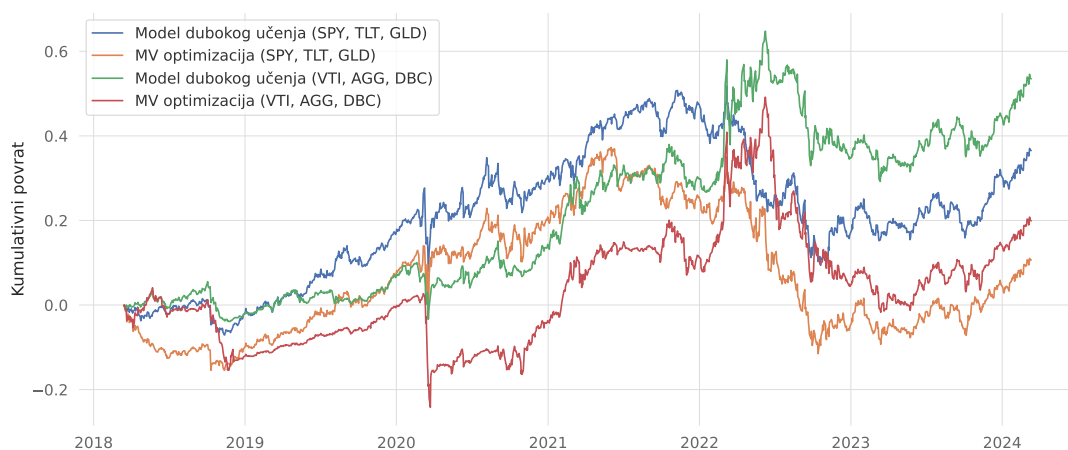
Tablica 5.1 prikazuje rad modela na podacima o povratima SPY, TLT i GLD ETF-ova na kojima je model naučen za period od 2006. do 2015. (skup podataka za učenje) te za period od 2018. do 2024. (skup podataka za testiranje). Osim toga prikazan je i rad modela na podacima o povratima VTI, AGG i DBC ETF-ova, također za periode 2006.-2015. i 2018.-2024. Ovi podatci pripadaju istim kategorijama ETF-ova (VTI i SPY su dionički ETF-ovi, TLT i AGG obveznički, a GLD i DBC robni), ali ih model nije imao pri treniranju. Ovakvim testiranjem modela provjerava se njegova sposobnost generalizacije.

Može se uočiti da svi portfelji DU, kraće za portfelje dobivene modelom dubokog učenja, nadmašuju performanse MV portfelja u vidu očekivanih povrata i rizika portfelja, a posljedično i Sharpeovog omjera.

Tablica 5.1: Očekivani povrat portfelja dobivenih aproksimacijskim modelom i MV optimizacijom, njihova standardna devijacija i Sharpeov omjer, za podatke na kojima je treniran (SPY, TLT, GLD) te za one na kojima nije (VTI, AGG, DBC), za vremenski period treniranja (2006 – 2015.) i za vremenski period testiranja (2018. – 2024.)

2006. – 2015.						
	SPY, TLT, GLD			VTI, AGG, DBC		
	E(R)	Std(R)	Sharpe	E(R)	Std(R)	Sharpe
Aproksimacijski model	0.2969	0.1156	2.25	0.1971	0.0896	2.01
MV optimizacija	0.1073	0.1230	0.78	0.0294	0.1417	0.20
2018. – 2024.						
	SPY, TLT, GLD			VTI, AGG, DBC		
	E(R)	Std(R)	Sharpe	E(R)	Std(R)	Sharpe
Aproksimacijski model	0.0632	0.1293	0.47	0.0826	0.1236	0.64
MV optimizacija	0.0277	0.1441	0.19	0.0439	0.1583	0.27

Na slici 5.4 prikazano je kretanje kumulativnih povrata sačinjenih od SPY-a, TLT-a i GLD-a te portfelja sačinjenih od VTI-a, AGG-a i DBC-a, dobivenih MV optimizacijom i aproksimacijskim modelom u periodu od 2018. do 2024. godine. Svi portfelji dobiveni aproksimacijskim modelom imaju veće kumulativne povrate tijekom testnog perioda nego oni dobiveni MV optimizacijom.



Slika 5.4: Usporedba kumulativnih povrata portfelja sačinjenih od SPY-a, TLT-a i GLD-a te portfelja sačinjenih od VTI-a, AGG-a i DBC-a, dobivenih MV optimizacijom i aproksimacijskim modelom.

Uz osnovne metrike performanse portfelja predstavljene u tablici 5.1, u tablici 5.2 na-

vedene su i one kojima se portfelj dobiven aproksimacijskim modelom uspoređuje s onim dobivenim MV optimizacijom. Konkretno, to su: analizirano praćenje odstupanja od referentne vrijednosti (engl. *tracking error*), dalje u tekstu TE, korelacija povrata te korelacija udjela. Vidljivo je da veći TE i manje korelacije povrata i udjela u odnosu na MV portfelje imaju oni sačinjeni od VTI, AGG i DBC ETF-ova, što je očekivano s obzirom da model nije učio na tim podacima.

Tablica 5.2: Metrike usporedbe aproksimacijskog modela s MV portfeljima: *tracking error* (TE), korelacija povrata (R) i korelacija udjela, za podatke na kojima je treniran (SPY, TLT, GLD) te za one na kojima nije (VTI, AGG, DBC), za vremenski period treniranja (2006 – 2015.) i za vremenski period testiranja (2018. – 2024.)

2006. – 2015.						
	SPY, TLT, GLD			VTI, AGG, DBC		
	TE	Kor. R	Kor. udjela	TE	Kor. R	Kor. udjela
DU ↔ MV	0.0673	0.8559	0.8604	0.1196	0.5402	0.6333
2018. – 2024.						
	SPY, TLT, GLD			VTI, AGG, DBC		
	TE	Kor. R	Kor. udjela	TE	Kor. R	Kor. udjela
DU ↔ MV	0.0755	0.8529	0.7378	0.1049	0.7497	0.6189

Model treniran na permutacijama imovina

Uz ranije predstavljeni model koji je naučen na jednom poretku imovina, testirane su i performanse aproksimacijskog modela koji je treniran na svih 6 permutacija imovina. Ovakvo treniranje za cilj ima stvaranje modela koji je agnostičan prema poretku imovina, odnosno model s većom sposobnošću generalizacije. U tablici 5.3 navedena je srednja vrijednost sa standardnom devijacijom rezultata testiranja modela na različitim permutacijama imovina za očekivani povrat portfelja, njegovu standardnu devijaciju i Sharpeov omjer. Prikazani su rezultati za aproksimacijski model treniran na svim poredcima imovina, oznaka (perm.), te za onaj treniran na jednom poretku.

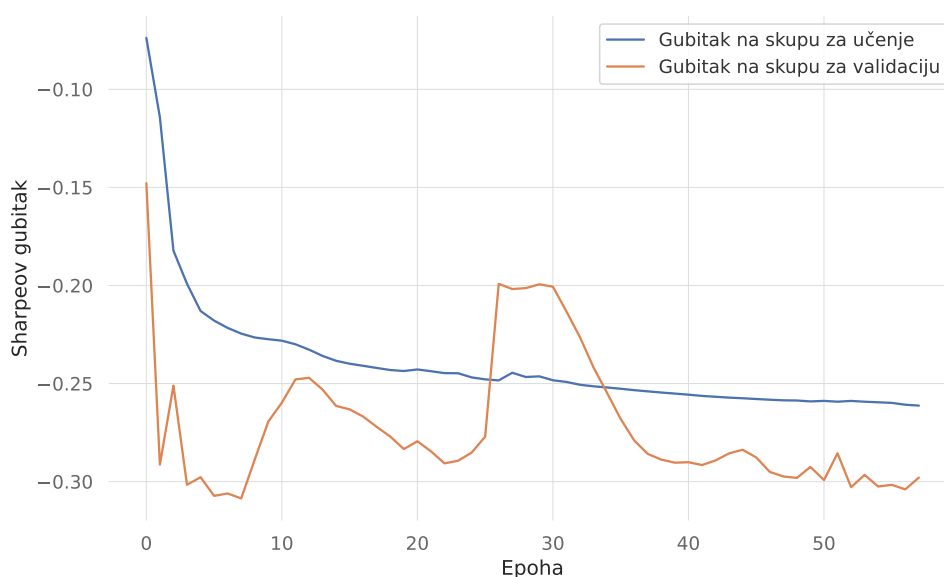
Iako i model koji je treniran na samo jednoj permutaciji imovina pokazuje dobre rezultate pri testiranju na različitim poredcima imovina, treniranjem modela na permutacija postiže se manja razlikama u performansama portfelja dobivenih s različitim redoslijedom imovina u ulaznim podacima. Osim toga portfelji dobiveni aproksimacijskim modelom treniranom na permutacijama postižu bolji Sharpeov omjer (permutacija s najlošijim Sharpeovim omjerom: 0.54).

Tablica 5.3: Srednja vrijednost \pm standardna devijacija rezultata testiranja na različitim permutacijama imovina za očekivani povrat portfelja, njegovu standardnu devijaciju i Sharpeov omjer. Prikazani rezultati za aproksimacijski model treniran na svim poredcima imovina, oznaka (perm.), te za onaj treniran na jednom poretku.

	2018. – 2024.		
	SPY, TLT, GLD		
	E(R)	Std(R)	Sharpe
Aproksimacijski model (perm.)	0.0775 ± 0.0054	0.1288 ± 0.0015	0.58 ± 0.04
Aproksimacijski model	0.0718 ± 0.0148	0.1264 ± 0.0038	0.55 ± 0.10

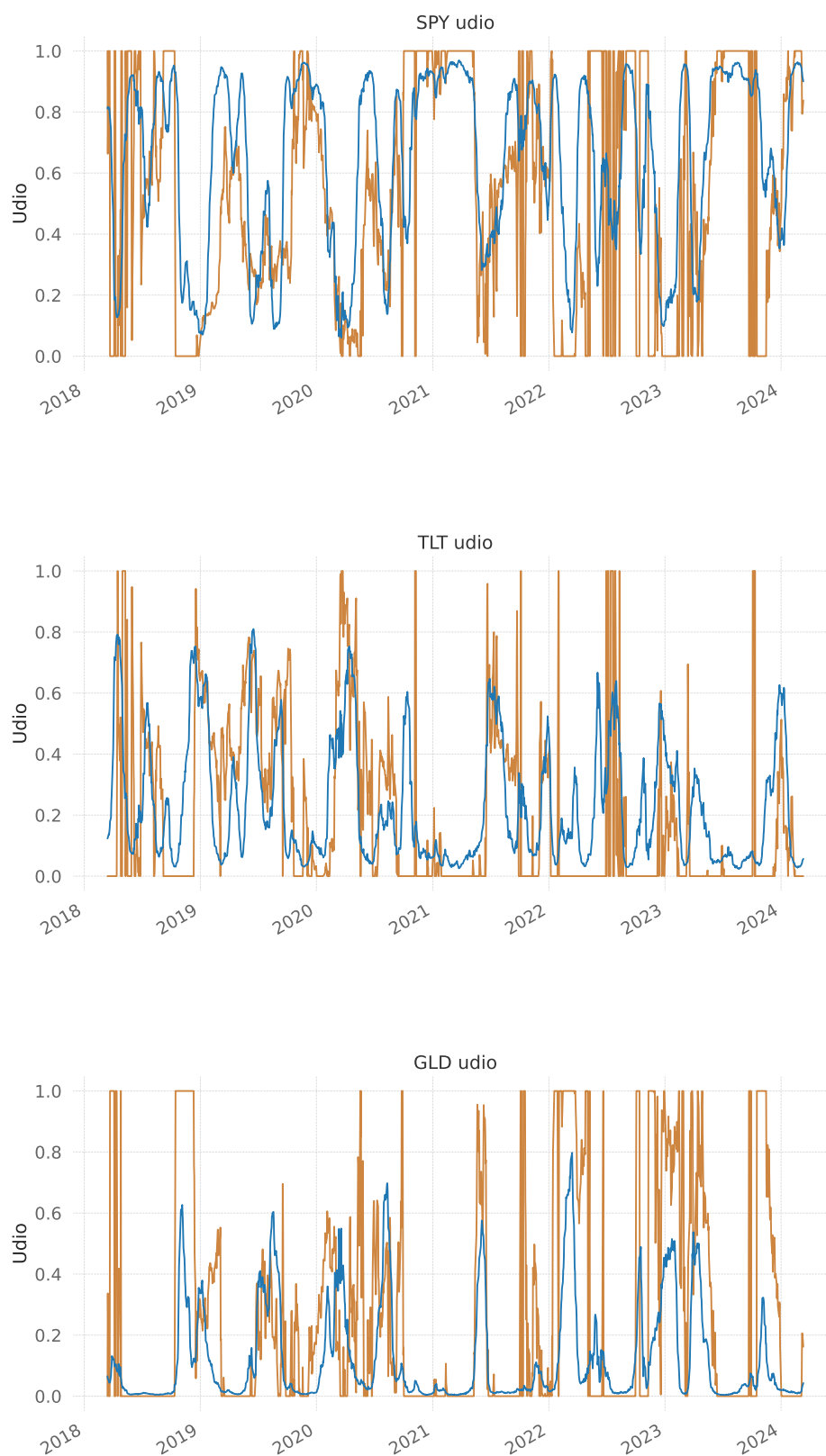
5.2.2. Prediktivni model

Na slici 5.5 prikazano je kretanje vrijednosti Sharpeovog gubitka na skupu za treniranje i skupu za validaciju. Kao i aproksimacijski model, prediktivni model vrlo brzo postiže niske vrijednosti Sharpeovog gubitka, a već u 8. epohi najnižu.



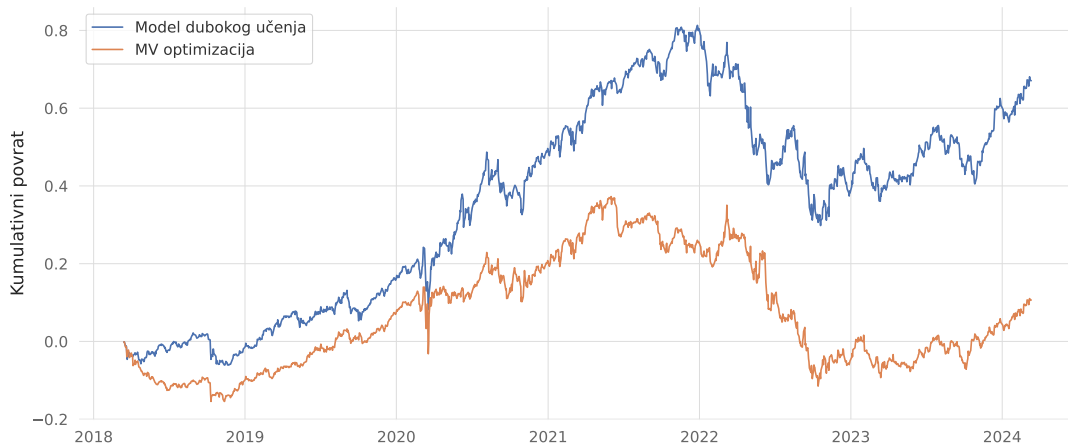
Slika 5.5: Kretanje vrijednosti Sharpeovog gubitka tijekom učenja modela

Slika 5.6 prikazuje alokacije imovina (SPY, TLT i GLD) kroz testni period od 2018. do 2024., dobivene kao rezultat MV optimizacije (narančaste linije) i izlaz prediktivnog modela (plave linije). Korelacija udjela značajno je manja u odnosu na aproksimacijski model (55.25% naspram 73.78%), te je također vidljivo da prediktivni model ima manje naglih promjena u alokacijskoj strategiji i manje je sklon ekstremima.



Slika 5.6: Alokacije imovina kroz testni period od 2018. do 2024. dobivenih kao rezultat MV optimizacije (narančaste linije) i izlaza prediktivnog modela (plave linije)

Slika 5.7 prikazuje usporedbu kumulativnih povrata portfelja dobivenih MV optimizacijom i prediktivnim modelom na podacima za testiranje. Korelacija povrata s MV portfeljem nešto je manja u odnosu na aproksimacijski model (75.71% za prediktivni model naspram 85.29% za aproksimacijski), ali portfelj dobiven prediktivnim modelom nadmašuje performanse kako onog dobivenog MV optimizacijom, tako i onog dobivenog aproksimacijskim modelom.



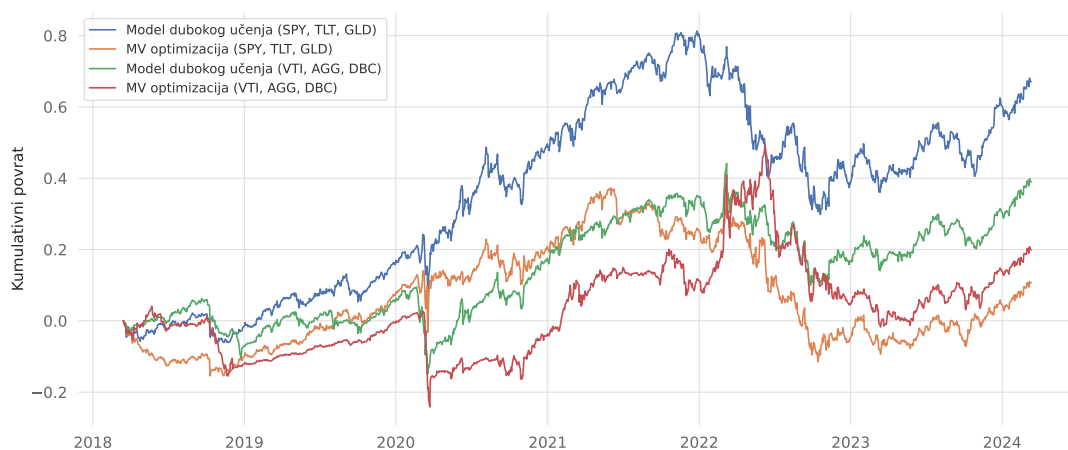
Slika 5.7: Usporedba kumulativnih povrata portfelja (SPY, TLT, GLD) dobivenih MV optimizacijom i prediktivnim modelom

U tablici 5.4 može se uočiti da svi portfelji dobiveni prediktivnim modelom nadmašuju performanse MV portfelja u testnom periodu (2018. – 2024.) u vidu očekivanih povrata i rizika portfelja, a posljedično i Sharpeovog omjera. Također, valja spomenuti da portfelji dobiveni prediktivnim modelom postižu bolje rezultate u odnosu na one dobivene aproksimacijskim modelom na testnom periodu od 2018. do 2024. godine.

Tablica 5.4: Očekivani povrat portfelja dobivenih prediktivnim modelom i MV optimizacijom, njihova standardna devijacija i Sharpeov omjer, za podatke na kojima je treniran (SPY, TLT, GLD) te za one na kojima nije (VTI, AGG, DBC), za vremenski period treniranja (2006 – 2015.) i za vremenski period testiranja (2018. – 2024.)

2006. – 2015.						
	SPY, TLT, GLD			VTI, AGG, DBC		
	E(R)	Std(R)	Sharpe	E(R)	Std(R)	Sharpe
Prediktivni model	0.0918	0.1141	0.77	0.0728	0.1080	0.65
MV optimizacija	0.1073	0.1230	0.78	0.0294	0.1417	0.20
2018. – 2024.						
	SPY, TLT, GLD			VTI, AGG, DBC		
	E(R)	Std(R)	Sharpe	E(R)	Std(R)	Sharpe
Prediktivni model	0.0994	0.1334	0.71	0.1077	0.1252	0.82
MV optimizacija	0.0277	0.1441	0.19	0.0439	0.1583	0.27

Na slici 5.8 prikazano je kretanje kumulativnih povrata sačinjenih od SPY-a, TLT-a i GLD-a te portfelja sačinjenih od VTI-a, AGG-a i DBC-a, dobivenih MV optimizacijom i prediktivnim modelom u periodu od 2018. do 2024. godine. Kao što je ranije komentirano, za imovine SPY, TLD i GLD prediktivni model nadmašuje MV optimizaciju, dok za VTI, AGG i DBC kumulativni povrati portfelja dobivenog prediktivnim modelom na većini vremenskog perioda premašuju one portfelja dobivenog MV optimizacijom.



Slika 5.8: Usporedba kumulativnih povrata portfelja sačinjenih od SPY-a, TLT-a i GLD-a te portfelja sačinjenih od VTI-a, AGG-a i DBC-a, dobivenih MV optimizacijom i prediktivnim modelom.

U tablici 5.5 vidljivo je da veći TE i manje korelacije povrata i udjela u odnosu na MV portfelje imaju oni sačinjeni od VTI, AGG i DBC ETF-ova.

Tablica 5.5: Metrike usporedbe prediktivnog modela s MV portfeljima: *tracking error* (TE), korelacija povrata (R) i korelacija udjela, za podatke na kojima je treniran (SPY, TLT, GLD) te za one na kojima nije (VTI, AGG, DBC), za vremenski period treniranja (2006 – 2015.) i za vremenski period testiranja (2018. – 2024.)

2006. – 2015.						
	SPY, TLT, GLD			VTI, AGG, DBC		
	TE	Kor. R	Kor. udjela	TE	Kor. R	Kor. udjela
DU ↔ MV	0.0866	0.7554	0.6461	0.1215	0.5559	0.4011
2018. – 2024.						
	SPY, TLT, GLD			VTI, AGG, DBC		
	TE	Kor. R	Kor. udjela	TE	Kor. R	Kor. udjela
DU ↔ MV	0.0968	0.7591	0.5525	0.1119	0.7119	0.4083

Model treniran na permutacijama imovina

Prediktivni model također je treniran i na permutacijama imovina te su rezultati testiranja na svim poredcima imovina u ulaznim podacima predstavljeni u tablici 5.6. Može se uočiti da je prediktivni model treniran na permutacijama nestabilniji od aproksimacijskog, čiji su rezultati prezentirani u tablici 5.3. Primjerice, jedna permutacija ulaznih podataka rezultira portfeljem Sharpeovog omjera 0.73, a druga 0.5. Unatoč tome, treniranjem na svim poredcima ipak se postiže određena doza agnostičnosti prema poretku imovina, što se očituje značajno smanjenom varijabilnošću performansi portfelja dobivenih modelom treniranim na svim permutacijama u odnosu na one dobivene modelom treniranim na jednoj permutaciji.

Tablica 5.6: Srednja vrijednost \pm standardna devijacija rezultata testiranja na različitim permutacijama imovina za očekivani povrat portfelja, njegovu standardnu devijaciju i Sharpeov omjer. Prikazani rezultati za prediktivni model treniran na svim poredcima imovina, oznaka (perm.), te za onaj treniran na jednom poretku.

	2018. – 2024.		
	SPY, TLT, GLD		
	E(R)	Std(R)	Sharpe
Prediktivni model (perm.)	0.0871 ± 0.0093	0.1285 ± 0.0028	0.65 ± 0.08
Prediktivni model	0.0883 ± 0.0268	0.1248 ± 0.0074	0.67 ± 0.19

6. Zaključak

Cilj ovog rada bio je implementirati modele dubokog učenja za optimizaciju portfelja. Predložena su dva modela, LSTM arhitekture, koji na temelju povrata imovina direktno optimiziraju Sharpeov omjer portfelja, to jest omjer povrata i rizika. Prvi od njih, aproksimacijski model, ima za cilj naučiti MV optimalan portfelj. S druge strane, prediktivni model pokušava na temelju prethodnih povrata naučiti optimalne alokacije za određeni period u budućnosti. U radu je detaljnije predstavljena arhitektura navedenih modela i komentirana programska izvedba te odabrani hiperparametri. Modeli su trenirani na odabranom skupu podataka (SPY, TLT i GLD ETF-ovi u periodu između 2006. i 2015.), prvo na jednom poretku imovina, a zatim i na svim permutacijama u pokušaju postizanja agnostičnosti modela prema redoslijedu imovina u ulaznim podacima. Zatim su analizirani rezultati, kako na vremenskom periodu treniranja (2006. – 2015.), tako i na vremenskom periodu testiranja (2018. – 2024.), te je napravljena usporedba s rezultatima MV optimizacije. Aproksimacijski model nije uspio u potpunosti naučiti MV optimalan portfelj. Portfelji dobiveni i aproksimacijskim i prediktivnim modelom polučili su bolje rezultate pri testiranju u odnosu na portfelje dobivene MV optimizacijom. Također, modeli trenirani na permutacijama poboljšali su svoju sposobnost generalizacije i smanjili varijabilnost performansi portfelja dobivenih različitim poretkom imovina u ulaznim podacima.

LITERATURA

- [1] Frank Fabozzi, Harry Markowitz, i Francis Gupta. *Portfolio Selection*. 09 2008. ISBN 9780470404324. doi: 10.1002/9780470404324.hof002001.
- [2] Weinan E, Ruimeng Hu, i Shige Peng. Deep Learning in Finance. *Digital Finance*, 5(1):1–2, March 2023. URL https://ideas.repec.org/a/spr/digfin/v5y2023i1d10.1007_s42521-023-00080-2.html.
- [3] A.N. Akansu i M.U. Torun. *A Primer for Financial Engineering: Financial Signal Processing and Electronic Trading*. Elsevier Science, 2015. ISBN 9780128015612. URL <https://books.google.hr/books?id=dUkHrgEACAAJ>.
- [4] Eric Zivot. *Chapter 11 Introduction to Portfolio Theory | Introduction to Computational Finance and Financial Econometrics with R*. URL <https://bookdown.org/compfinezbook/introcompfinr/Introduction-to-Portfolio-Theory.html>. [Pristupljeno 15-06-2024].
- [5] Petra Posedel Šimović i Zvonko Kostanjčar. *Financijska matematika: CAPM*, 2024. prezentacija s kolegija Financijska matematika na FER-u.
- [6] Marko Đurasević. *Kvadratično i sekvencijalno kvadratično programiranje*. URL https://www.fer.unizg.hr/_download/repository/SQP.pdf.
- [7] Jorge Nocedal i Stephen J. Wright. *Numerical Optimization*. Springer, New York, NY, USA, 2e izdanju, 2006.
- [8] William F. Sharpe. The sharpe ratio. *Journal of Portfolio Management*, 21(1):49, 1994.
- [9] William F. Sharpe. Mutual fund performance. *The Journal of Business*, 39(1): 119–138, Siječanj 1966. URL <https://www.jstor.org/stable/2351741>.

- [10] Zihao Zhang, Stefan Zohren, i Stephen Roberts. Deep learning for portfolio optimization. *The Journal of Financial Data Science*, 2(4):8–20, Kolovoz 2020. ISSN 2640-3943. doi: 10.3905/jfds.2020.1.042. URL <http://dx.doi.org/10.3905/jfds.2020.1.042>.
- [11] Rama Cont. Empirical properties of asset returns: Stylized facts and statistical issues. *Quantitative Finance*, 1:223–236, 03 2002. doi: 10.1088/1469-7688/1/2/304.
- [12] Stjepan Begušić. *Estimation of latent factors from high-dimensional financial time series based on unsupervised learning*. Doktorska disertacija, Sveučilište u Zagrebu, Fakultet elektrotehnike i računarstva, Zagreb, 2020. URL <https://urn.nsk.hr/urn:nbn:hr:168:732129>.
- [13] Olivier Ledoit i Michael Wolf. The Power of (Non-)Linear Shrinking: A Review and Guide to Covariance Matrix Estimation. *Journal of Financial Econometrics*, 20(1):187–218, 06 2020. ISSN 1479-8409. doi: 10.1093/jjfinec/nbaa007. URL <https://doi.org/10.1093/jjfinec/nbaa007>.
- [14] Siniša Šegvić. Duboko učenje — zemris.fer.hr. URL <https://www.zemris.fer.hr/~ssegvic/du/>. [Pristupljeno 17-06-2024].
- [15] Marko Čupić. *Umjetna inteligencija: Umjetne neuronske mreže*, 05 2016. URL <http://java.zemris.fer.hr/nastava/ui/ann/ann-20180604.pdf>. skripta s kolegija Umjetna inteligencija na FER-u.
- [16] Christopher Olah. Understanding LSTM Networks, 8 2015. URL <https://colah.github.io/posts/2015-08-Understanding-LSTMs/>. [Pristupljeno 17-06-2024].
- [17] Martin Tutek, Petra Bevandić, Josip Šarić, i Siniša Šegvić. Napredni povratni modeli, 2023. URL <https://www.zemris.fer.hr/~ssegvic/du/du6recurrent2.pdf>. [Pristupljeno 17-06-2024].
- [18] Sepp Hochreiter i Jürgen Schmidhuber. Long short-term memory. *Neural Computation*, 9(8):1735–1780, 1997.
- [19] Jan Šnajder. *Strojno učenje 1: Osnovni koncepti*, 2022. URL https://www.fer.unizg.hr/_download/repository/SU1-2022-P02-OsnovniKoncepti.pdf. skripta s kolegija Strojno učenje na FER-u.

- [20] Marko Čupić. Duboko učenje: Optimizacija parametara modela. <https://www.zemris.fer.hr/~ssegvic/du/du3optimization.pdf>, 2019. [Pristupljeno 18-06-2024].
- [21] Jan Šnajder. *Strojno učenje 1: Logistička regresija*, 2022. skripta s kolegija Strojno učenje na FER-u.
- [22] Diederik P. Kingma i Jimmy Ba. Adam: A method for stochastic optimization, 2017.
- [23] Tensorflow. URL <https://www.tensorflow.org/>.
- [24] Numpy. URL <https://numpy.org/>.
- [25] Matplotlib. URL <https://matplotlib.org/>.
- [26] pandas. URL <https://pandas.pydata.org/>.
- [27] Pyportfolioopt. URL <https://pyportfolioopt.readthedocs.io/en/latest/index.html>.
- [28] Google colab. URL <https://colab.research.google.com/>.
- [29] Laurent Deville. Exchange traded funds: History, trading, and research. U *Handbook of financial engineering*, stranice 67–98. Springer, 2008.

Model dubokog učenja za optimizaciju portfelja

Sažetak

U ovom radu dan je teorijski uvod u problem optimizacije portfelja, te *mean-variance* optimizacije kao jedan od standardnih pristupa rješavanju ovog problema. Zatim je predstavljeno područje dubokog učenja, uz naglasak na LSTM mreže. Predstavljena je implementacija modela dubokog učenja, s LSTM arhitekturom, koji iz podataka o povratima uče optimalne alokacije imovina s ciljem maksimizacije Sharpeovog omjera, odnosno omjera povrata i rizika portfelja. Modeli su ispitani na povijesnim tržišnim podacima uz statističku analizu rezultata u okviru financijskih mjera performansi. Kao referentna točka za procjenu kvalitete rada modela korišteni su portfelji dobiveni *mean-variance* optimizacijom.

Ključne riječi: optimizacija portfelja, mean-variance optimizacija, duboko učenje, LSTM

Deep learning model for portfolio optimization

Abstract

This paper provides a theoretical introduction to the problem of portfolio optimization, and mean-variance optimization as one of the standard approaches to solving this problem. Next, the field of deep learning is introduced, with an emphasis on the LSTM network. The implementation of a deep learning model, with LSTM architecture, which learns optimal asset allocations from return data with the aim of maximizing the Sharpe ratio, i.e. the ratio of return and portfolio risk, is presented. The models were tested on historical market data with a statistical analysis of the results in the framework of financial performance measures. Portfolios obtained by mean-variance optimization were used as a benchmark for evaluating performances of the models.

Keywords: portfolio optimization, mean-variance optimization, deep learning, LSTM