SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

DIPLOMSKI RAD br. 733

OPTIMIZACIJA SHARPEOVOG OMJERA PORTFELJA KORIŠTENJEM MODELA DUBOKOG UČENJA

Luka Lukačević

Zagreb, veljača 2025.

SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

DIPLOMSKI RAD br. 733

OPTIMIZACIJA SHARPEOVOG OMJERA PORTFELJA KORIŠTENJEM MODELA DUBOKOG UČENJA

Luka Lukačević

Zagreb, veljača 2025.

SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

Zagreb, 30. rujna 2024.

DIPLOMSKI ZADATAK br. 733

Pristupnik: **Luka Lukačević (0036516767)**

Studij: Računarstvo

Profil: Znanost o podacima

Mentor: prof. dr. sc. Zvonko Kostanjčar

Zadatak: **Optimizacija Sharpeovog omjera portfelja korištenjem modela dubokog učenja**

Opis zadatka:

U okviru zadatka potrebno je istražiti primjenu modela dubokog učenja za optimizaciju Sharpeovog omjera portfelja. Prvo je potrebno istražiti i implementirati standardne pristupe za optimizaciju Sherepovog omjera, koji se temelje na predviđanju kovarijance povrata i na predviđanju očekivanih povrata. Nakon toga potrebno je razviti metodu koja se zasniva na izravnoj optimizaciji težina portfelja putem ažuriranja parametara modela dubokog učenja te koja tako zaobilazi potrebu za predviđanjem očekivanih povrata i kovarijance. Metodu je potrebno razviti i testirati na skupu financijskih vrijednosnica koje čine fondovi kojima se trguje na burzi (ETF-ovi) s obzirom da oni nisu izloženi individualnim rizicima pojedinih kompanija. Analiza rezultata treba uključivati i usporednu studiju predložene metode sa standardnim pristupima, s posebnim naglaskom na izvedbu u razdoblju pada i rasta tržišta, odnosno pojačane i smanjene volatilnosti na tržištu. Također, potrebno je provesti analizu osjetljivosti kako bi se procijenila važnost ulaznih značajki i istražio utjecaj različitih stopa transakcijskih troškova na izvedbu portfelja.

Rok za predaju rada: 14. veljače 2025.

Zahvaljujem svom mentoru, izv. prof. dr. sc Zvonku Kostanjčaru što me zainteresirao za područje o kojem ću pisati te pruženom povjerenju i potpori pri ovome radu. Također, zahvaljujem asistentici Jani Juroš na mnoštvu odgovora i pomoći u svakome trenutku. Zahvaljujem majci Jasmini, ocu Hrvoju, sestri Mihaeli i bratu Janku za njihovu podršku i savjet kada god je bio potreban. Posebno hvala bakama Agati i Mariji te djedovima Miji i Pavlu, koji su uvijek slavili moje uspjehe i poticali me da idem dalje. Konačno, hvala svim prijateljima i kolegama s kojima dijelim mnoštvo uspomena koje su mi uljepšale obrazovanje, a posebno mojoj Luciji.

Sadržaj

1. Uvod

Optimizacija portfelja jedan je od temeljnih izazova u financijama, s ciljem učinkovite alokacije ulaganja kako bi se postigao najveći mogući povrat za određenu razinu rizika. Portfelj je skup financijskih instrumenata koji uključuju mješavinu raznih imovina, od kojih svaka nosi različite potencijalne povrate i rizike. Veći rizik obično znači i veći potencijalni povrat, no to ne garantira bolje rezultate, već ovisi o tržišnim uvjetima i strategiji ulaganja. Investitori se susreću s kritičnim zadatkom gdje moraju odlučiti kako rasporediti svoj kapital između različitih financijskih sredstava, kao što su dionice, obveznice itd. Svaki investitor treba pronaći balans i najbolju opciju za njihovu preferenciju rizika. Kompleksnost ovog zadatka proizlazi iz dinamične prirode financijskih tržišta, međuovisnosti povrata imovina te potrebe za učinkovitim strategijama upravljanja rizikom.

Jedan od najutjecajnijih okvira za optimizaciju portfelja je moderna teorija portfelja (engl. *Modern portfolio theory - MPT*), koju je predstavio Harry Markowitz 1952 [1]. godine. MPT naglašava prednosti diverzifikacije, pokazujući da dobro diverzificiran portfelj može postići bolje omjere rizika i povrata od pojedinačnih imovina. Odabirom imovina s niskom korelacijom, investitori mogu smanjiti ukupan rizik portfelja bez žrtvovanja očekivanog povrata. Glavni dio MPT-a je *mean-variance - MV* analiza koja se temelji na pretpostavci da investitori nastoje maksimizirati očekivani povrat za zadanu razinu rizika. Iako pruža snažan teorijski okvir, MPT ima ograničenja, poput oslanjanja na normalnu distribuciju povrata, stacionarne korelacije i precizne procjene parametara, što u stvarnosti nije uvijek ostvarivo.

Jedan od alternativnih klasičnih pristupa je maksimalna diverzifikacija (engl. *Maximum diversification - MD*), koja također naglašava koncept diverzifikacije. Ova metoda ima cilj maksimizirati omjer diverzifikacije odabirom imovina koje najviše utječu na ukupno smanjenje rizika portfelja. Ipak, i ova metoda suočava se s praktičnim ograničenjima,

osobito u promjenjivim tržišnim uvjetima. Teorija optimizacije portfelja oslanja se na klasične metode, no njihove pretpostavke često ne odražavaju stvarne tržišne uvjete.
Razvojem financijskih tržišta i napretkom u području umjetne inteligencije i strojnog učenja posljednjih godina, otvoreni su novi putevi za optimizaciju portfelja. Duboko učenje, podskup strojnog učenja, pokazalo je izvanredan uspjeh u identificiranju složenih, nelinearnih obrazaca u velikim skupovima podataka. Ova sposobnost čini duboko učenje posebno prikladnim za financijske aplikacije, gdje su tržišta izrazito dinamična i pod utjecajem brojnih faktora. Među povratnim modelima dubokog učenja, povratna ćelija s dugoročnom memorijom (engl. *Long short-term memory - LSTM*) stekla je popularnost zbog svoje sposobnosti modeliranja sekvencijalnih podataka i uočavanja vremenskih ovisnosti.
Ovaj diplomski rad istražuje primjenu LSTM mreža na problem optimizacije portfelja. Primarni cilj je razviti model koji optimizira težine portfelja (udjele imovina unutar portfelja) kako bi se maksimizirao Sharpeov omjer, koji predstavlja standardnu mjeru omjera povrata i rizika. Korištenjem prediktivne moći LSTM mreža, predloženi pristup nastoji prevladati ograničenja klasičnih metoda i pružiti investitorima robusnije i fleksibilnije alate za donošenje odluka. Rad je strukturiran na sljedeći način: prvo se razmatraju klasične metode optimizacije portfelja, uključujući Markowitzevu *mean-variance* analizu i strategiju maksimalne diverzifikacije. Zatim se uvode osnove dubokog učenja, povratnih mreža i LSTM-a, ističući njihovu važnost za financijske aplikacije. Na kraju, predstavlja se implementacija predloženog modela optimizacije portfelja temeljenog na LSTM mrežama, zajedno s analizom njegovih performansi i rezultata koji se uspoređuju s MV i MD pristupima.

2. Klasične metode optimizacije portfelja

Optimizacija portfelja temelj je moderne teorije ulaganja, pružajući sustavan okvir za izgradnju portfelja koji uravnotežuje rizik i povrat. Cilj optimizacije portfelja je raspodijeliti kapital kroz skup imovina na takav način da investitor postigne najveći mogući povrat za danu razinu rizika ili obrnuto, najmanji mogući rizik za danu razinu povrata. Ovo poglavlje pokriva dvije klasične metode za optimizaciju portfelja: Markowitzevu *mean-variance* analizu i metodu maksimalne diverzifikacije. Ovi pristupi ostaju temeljni u području financija i nastavljaju utjecati i na akademska istraživanja i na praktične strategije ulaganja.
U ovom poglavlju istražit će se teorijski temelji, matematičke formulacije i praktične primjene ove dvije klasične metode. Kroz navedenu analizu nastojat će se pružiti sveobuhvatno razumijevanje ovih klasičnih metoda, njihove važnosti u suvremenom upravljanju portfeljem i njihove potencijalne primjene u različitim tržišnim uvjetima.

2.1. *Mean-variance* analiza

Mean-variance analiza jedan je od temeljnih okvira u financijama za optimizaciju portfelja. Primarni cilj MV-a je konstruirati portfelj koji maksimizira očekivane povrate za određenu razinu rizika ili obrnuto, minimizira rizik za određenu razinu povrata. Markowitzev okvir naglašava važnost koncepta diverzifikacije, pokazujući da kombiniranjem imovina s različitim profilima rizika i povrata, investitori mogu izgraditi portfelje zadovoljavajućeg povrata prilagođenog riziku. Markowitzev rad donio mu je Nobelovu nagradu za ekonomiju 1990. godine. Dvije primarne mjere ovog pristupa uključuju:

- Očekivani povrat - težinski prosjek povrata svih imovina unutar portfelja

- Rizik - mjera ukupne volatilnosti (standardna devijacija povrata) portfelja, koja izražava neizvjesnost povrata

Prva od dviju primarnih mjera ovog pristupa je očekivani povrat. Općenito, aritmetički povrat pojedine imovine definiramo na način:

$$R_t = \frac{P_t - P_{t-1}}{P_{t-1}}. \tag{2.1}$$

gdje P_t predstavlja cijenu imovine u trenutku t. Sada će biti konstruiran portfelj s n vrijednosnica. Za takav portfelj, s udjelima (težinama) $w = [w_1, ..., w_n]^T$ gdje pojedina težina w_i predstavlja udio kapitala investiranog u odgovarajuću imovinu, povrat u trenutku t glasi:

$$R_{pt} = \sum_{i=1}^{n} w_i \cdot R_{it} \tag{2.2}$$

gdje R_{it} označava povrat pojedine imovine u trenutku t. Očekivani povrat portfelja aproksimira se težinskim prosjekom povrata svih imovina unutar portfelja. Za zadani portfelj s vektorom očekivanih povrata pojedinih imovina μ očekivanje je:

$$E[R_p] = \mu_p = \sum_{i=1}^{n} E[w_i R_{it}] = \sum_{i=1}^{n} w_i \mu_i = w_i^T \mu, \tag{2.3}$$

Druga ključna mjera je rizik, odnosno u ovom kontekstu volatilnost portfelja. Volatilnost je statistička mjera disperzije povrata vrijednosnog papira, tj. ona je pokazatelj rizika. Standardno se izražava kao standardna devijacija povrata. Za spomenuti portfelj s matricom kovarijance Σ varijanca portfelja je:

$$Var(R_p) = \sigma_p^2 = \sum_{i=1}^{n} \sum_{j=1}^{n} w_i w_j Cov(R_i, R_j) = \sum_{i=1}^{n} \sum_{j=1}^{n} w_i \sigma_{ij} w_j = w^T \Sigma w. \tag{2.4}$$

Formule 2.3 i 2.4 vrijede pod uvjetom $w^T \mathbf{1} = 1$, te eventualno $w_i \geq 0, i \in \{1, ..., n\}$. Drugi uvjet je proizvoljan i nalaže nemogućnost kratkih prodaja u portfelju. Kratke prodaje

(engl. *short selling*) djeluju na principu prodaje imovine koja se ne posjeduje posuđivanjem od brokera. Za početak, investitor traži posuđivanje dionica X od svog brokera, pod uvjetom da broker drži dionice u fondu imovine svojih kupaca. Jednom posuđene dionice se prodaju u ime ulagača, a prihod od te prodaje, x_0, knjiže se na račun investitora. Važno je da račun odražava negativnu poziciju u broju dionica X jer je dug evidentiran u smislu broja posuđenih dionica, a ne njihove novčane vrijednosti. Na kraju, investitor mora ponovno otkupiti isti broj dionica po trenutačnoj tržišnoj cijeni, x_1, i vratiti ih u fond brokera. Ako je otkupna cijena x_1 niža od početne prodajne cijene x_0, investitor profitira od razlike. U suprotnome, investitor gubi novac.
Međutim, kratke prodaje nose sa sobom značajne rizike. Jedan od najvećih rizika je teorijski neograničeni gubitak. Dok su potencijalni dobici ograničeni (maksimalni dobitak je jednak prodajnoj cijeni x_0 ako cijena dionice padne na nulu), gubici mogu biti neograničeni jer cijena dionice može teoretski rasti beskonačno. Na primjer, ako cijena dionice X značajno poraste, investitor će morati platiti višu cijenu x_1 za otkup, što može rezultirati velikim gubicima. Osim toga, kratke prodaje su podložne riziku od *short squeeze*-a, situacije u kojoj nagli rast cijene dionice prisiljava investitore da otkupe dionice kako bi zatvorili svoje pozicije, što dodatno potiče rast cijene i povećava gubitke. Ovi čimbenici čine kratke prodaje riskantnom strategijom koja zahtijeva pažljivo upravljanje rizikom i duboko razumijevanje tržišnih dinamika zbog čega se često postavljam spomenuti uvjet u *mean-variance* analizi (iako su doljnje formule napisane bez dotičnog uvjeta jer nije obavezan).

U modernoj teoriji portfelja, modelira se stopa povrata imovina kao slučajne varijable. Cilj je tada optimalno odabrati težine portfelja. U kontekstu Markowitzeve teorije optimalan skup težina je onaj u kojem portfelj postiže prihvatljivu osnovnu očekivanu stopu povrata uz minimalnu volatilnost. Ovdje se varijanca stope povrata imovina uzima kao surogat za njegovu volatilnost. Ako s μ_0 označimo prihvatljivu osnovnu očekivanu stopu povrata, tada prema Markowitzevoj teoriji vrijedi da je optimalni portfelj bilo koji portfelj koji rješava sljedeći kvadratni problem:

$$\min_{w} \; w^T \Sigma w \tag{2.5}$$

uz ograničenja:

$$\begin{aligned} w^T\mu &\geq \mu_0, \\ w^T\mathbf{1} &= 1. \end{aligned} \tag{2.6}$$

Alternativni pristup ovog problema koji dovodi do istog rješenja je maksimizacija očekivanih povrata za danu maksimalnu razinu rizika (varijance) σ_0^2:

$$\max_w \; w^T\mu \tag{2.7}$$

uz ograničenja:

$$\begin{aligned} w^T\Sigma w &\leq \sigma_0^2, \\ w^T\mathbf{1} &= 1. \end{aligned} \tag{2.8}$$

2.1.1. Efikasna granica i teorem dva fonda

Rješenje prethodnog problema dat će nam portfelj čiji je rizik minimalan s obzirom na povrat koji je veći ili jednak μ_0. U praksi, željeli bismo imati bolje razumijevanje kompromisa povrata i rizika budući da želimo maksimizirati povrat uz minimiziranje rizika. Kada bismo iterirali kroz sva moguća očekivanja povrata portfelja i riješili minimizacijski problem uz spomenuta ograničenja, dobili bismo portfelje s najmanjom mogućom varijancom za svako očekivanje povrata od kojih se onaj s najmanjom varijancom naziva portfelj minimalne varijance. Skup svih takvih točaka $(\sigma_p, E[R_p])$, čiji je očekivani prinos veći ili jednak očekivanom prinosu portfelja minimalne varijance, nazivamo efikasna granica. Portfelji koji se nalaze na efikasnoj granici zovu se efikasni portfelji ili efikasni portfelji s obzirom na očekivanje i varijancu. Kažemo da je investitor koji se pozicionira na efikasnoj granici efikasan investitor u Markowitzevom smislu. Portfelji koji se nalaze ispod efikasne granice su suboptimalni jer daju manju razinu povrata za isti rizik.

Jedan od ključnih rezultata Markowitzeve teorije je teorem dva fonda (engl. *Two-Fund Theorem*), koji nalaže da se bilo koji portfelj na efikasnoj granici može konstruirati kao linearna kombinacija dva različita efikasna portfelja. Drugim riječima, ako su nam poznati efikasni portfelji A i B, bilo koji efikasni portfelj C možemo izraziti kao:

$$w_C = \alpha w_A + (1 - \alpha) w_B \tag{2.9}$$

gdje su w_A i w_B težine efikasnih portfelja A i B, a λ je udio portfelja A. Ovaj teorem uvelike pojednostavljuje konstrukciju portfelja jer investitori trebaju identificirati samo dva efikasna portfelja kako bi znali čitavu efikasnu granicu.

2.1.2. Uključivanje bezrizične imovine i teorem jednog fonda

Prethodno smo pretpostavljali da su sve imovine u našem portfelju rizične. Međutim, u praksi postoje imovine čiji je rizik toliko nizak da ih modeliramo kao bezrizične. U Hrvatskoj, bezrizična imovina uključuje trezorske zapise, koje izdaje Ministarstvo financija Republike Hrvatske. Njihov ekvivalent u Sjedinjenim Američkim Državama su Treasury Bills (engl. *T-Bills*), kratkoročni vrijednosni papiri koje izdaje američko Ministarstvo financija. Obje vrste instrumenata smatraju se praktično bezrizičnim jer su osigurane državnim jamstvom.

Sada ponavljamo Markowitzevu MV analizu uključivanjem bezrizične imovine koja ima stopu povrata r_f. Budući da je ta imovina bezrizična, njezina je varijanca jednaka nuli te je također njezina kovarijanca s bilo kojom drugom imovinom jednaka nuli. Ako zapišemo naš vektor povrata kao $\hat{x} = [r_f, x^T]^T$ gdje je $x = [R_1, ..., R_n]$, a R_i predstavlja razinu povrata rizične imovine i, onda je matrica kovarijance za $\hat{x}$ jednaka:

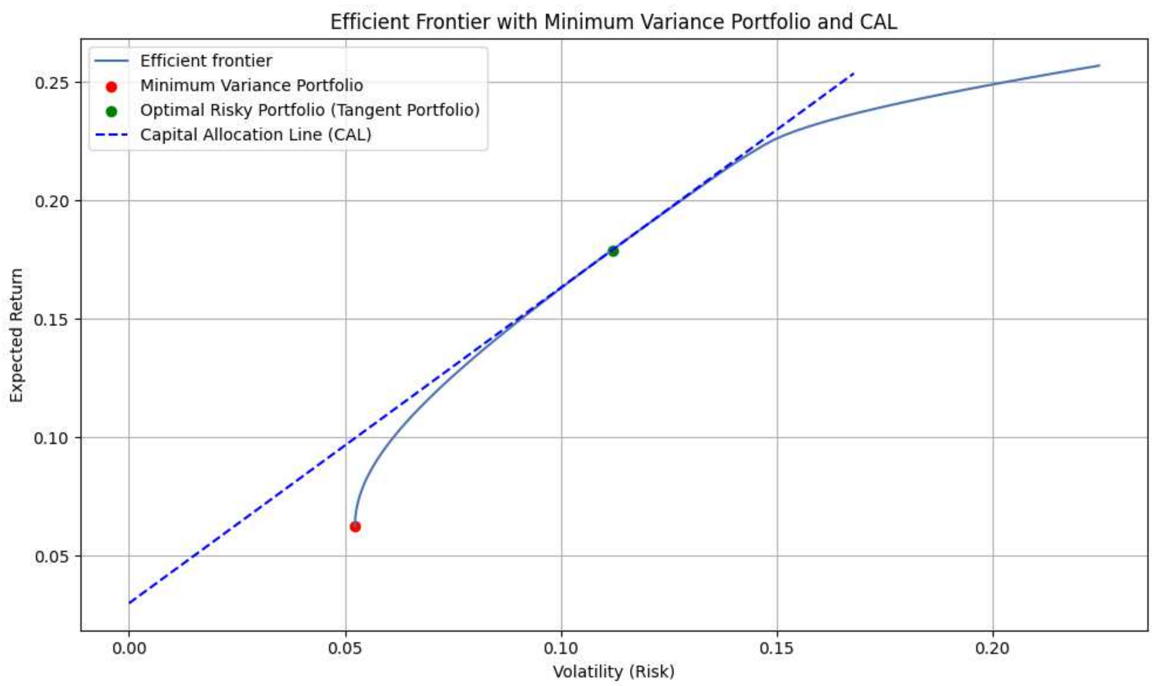
$$\hat{\Sigma} = \begin{bmatrix} 0 & 0 \\ 0 & \Sigma \end{bmatrix}, \tag{2.10}$$

gdje je Σ matrica kovarijance za x. Uzevši ovo sve u obzir, novi problem rješavamo istom minimizacijom kao 2.5, ali uz nova ograničenja:

$$\begin{aligned} r_f w_0 + w^T \mu &\geq \mu_b, \\ w_0 + w^T \mathbf{1} &= 1. \end{aligned} \tag{2.11}$$

Ovdje w_0 predstavlja udio bezrizične imovine u portfelju, a μ_0 zadovoljavajuću razinu povrata s time da pretpostavljamo $\mu_0 \geq r_f$. Također, ako je $w_0 > 0$ onda investiramo u bezrizičnu imovinu, a ako je $w_0 < 0$ onda posuđujemo bezrizičnu imovinu po kamatnoj stopi r_f. Pravac koji prolazi kroz povrat bezrizične imovine (nalazi se na osi ordinate budući da nema rizika - varijanca je jednaka nuli) i kroz neki portfelj s efikasne granice naziva se pravac alokacije kapitala (engl. *Capital Allocation Line - CAL*). Optimalni pravac alokacije kapitala je tangenta na efikasnu granicu koja prolazi kroz povrat bezrizične imovine. Taj se pravac također naziva pravac tržišta kapitala (engl. *Capital Market Line - CML*). U ovome slučaju, efikasni investitori ulagat će u portfelj s optimalnog pravca alokacije kapitala, odnosno u kombinaciju bezrizične imovine i tangencijalnog portfelja. Tangencijalni portfelj ujedno je i portfelj s najvećim Sharpeovim omjerom, tj.

$$\max_{w} \frac{\mu_p - r_f}{\sigma_p}. \tag{2.12}$$



Slika 2.1. Efikasna granica i CAL

Teorem jednoga fonda nalaže da ako odabir imovine za ulaganje uključuje bezrizičnu imovinu, tada postoji jedan portfelj F rizične imovine tako da se svaki efikasni portfelj može konstruirati kao linearna kombinacija bezrizične imovine i portfelja F. Pod uvjetima:

- svi investitori se bave MV optimizacijom
- svi investitori imaju jednake estimacije parametara (očekivani povrati, varijance i kovarijance)
- svi investitori koriste jedinstvenu bezrizičnu imovinu

tada će svi investitori ulagati u isti portfelj kojeg nazivamo tržišni portfelj. Tržišni portfelj sastoji se od težinske sume svih imovina na tržištu pri čemu je težina svake imovine proporcionalna svojoj prisutnosti na tržištu. Očekivani povrat tržišnog portfelja jednak je očekivanom povratu tržišta u cjelini.

2.1.3. Moć diverzifikacije

Diverzifikacija je strategija koja uključuje širenje ulaganja na više vrijednosnih papira kako bi se smanjio rizik. Portfelj sastavljen od samo jedne dionice izložen je dvjema glavnim vrstama rizika:

- Makroekonomski (sistematski) rizik - uključuje nepredvidive čimbenike kao što su poslovni ciklus, inflacija, kamatne stope i tečajevi, koji u određenoj mjeri utječu na sve dionice
- Rizik specifičan za tvrtku (idiosinkratski) - rizici jedinstveni za jednu tvrtku, kao što su promjene uprave ili rezultati istraživanja i razvoja, koji ne utječu na druge tvrtke.

Diverzifikacijom odnosno ulaganjem u, na primjer, još jednu tvrtku može se smanjiti utjecaj idiosinkratskog rizika. Ako jedna imovina ima loše rezultate zbog jedinstvenih okolnosti, druga može imati dobre rezultate, uravnotežujući ukupne povrate portfelja. Proširenje diverzifikacije na više imovina dodatno smanjuje idiosinkratski rizik, jer se neovisni rizici pojedinačnih kompanija kompenziraju. Međutim, diverzifikacija ne može eliminirati sistematski rizik koji proizlazi iz makroekonomskih čimbenika koji utječu na sve dionice. Ovaj preostali rizik naziva se i tržišni rizik.

Sada će biti razmotreno kako se diverzifikacijom može smanjiti idiosinkratski rizik. Koristeći naivnu strategiju diverzifikacije portfelja u kojoj se svakoj imovini unutar portfelja pridjeljuje jednak udio, odnosno $w_i = \frac{1}{n}$ za svaku imovinu portfelja. U tom slučaju jednadžba 2.4 može se raspisati na način:

$$\sigma_p^2 = \frac{1}{n}\sum_{i=1}^{n}\frac{1}{n}\sigma_i^2 + \sum_{\substack{j=1 \\ j\neq i}}^{n}\sum_{i=1}^{n}\frac{1}{n^2}\mathrm{Cov}(r_i, r_j) \tag{2.13}$$

gdje je iz prvotne jednadžbe izdvojena situacija u kojoj je $i = j$ u posebnu sumu. Valja primijetiti da sada postoji n članova sume za varijancu i $n(n-1)$ članova za kovarijancu. Ako se definira prosječna varijanca imovina kao:

$$\bar{\sigma}^2 = \frac{1}{n}\sum_{i=1}^{n}\sigma_i^2 \tag{2.14}$$

te prosječnu kovarijancu imovina na način:

$$\overline{\mathrm{Cov}} = \frac{1}{n(n-1)}\sum_{\substack{j=1 \\ j\neq i}}^{n}\sum_{i=1}^{n}\frac{1}{n^2}\mathrm{Cov}(r_i, r_j), \tag{2.15}$$

tada se može izraziti varijanca portfelja na način:

$$\sigma_p^2 = \frac{1}{n}\overline{\sigma}^2 + \frac{n}{n-1}\overline{\mathrm{Cov}}. \tag{2.16}$$

Ovdje se najbolje vidi učinak diverzifikacije. Ako u portfelju postoje nekorelirane ili slabo korelirane imovine, značajno se smanjuje varijanca portfelja. Kada su potpuno nekorelirane, varijanca portfelja svodi se samo na prvi dio jednadžbe 2.16 koji je zapravo sistematski dio. Također valja primijetiti da kako broj imovina portfelja n raste, prvi dio jednadžbe 2.16 teži u nulu ($\frac{1}{n} \to 0, n \to \infty$) dok drugi dio teži u jedan ($\frac{n}{n-1} \to 1, n \to \infty$). Zaključak ovoga je da kako broj imovina unutar portfelja raste, ukupni rizik sve više određuje međusobna kovarijanca imovina, dok doprinos individualne varijance postaje zanemariv.

2.1.4. Prednosti i nedostatci MV-a

Prednosti MV-a:

- Diverzifikacija portfelja - naglašava važnost diverzifikacije, pomažući ulagačima da postignu bolji odnos rizika i povrata od ulaganja u pojedinačnu imovinu.
- Široka primjenjivost - zbog jednostavnosti implementacije i interpretabilnosti ovo je jedan od najpoznatijih i najzastupljenijih modela u financijskoj industriji.

Neki od nedostataka MPT-a uključuju:

- Osjetljivost na ulazne parametre - Možda i najveći nedostatak cijele priče je činjenica da je efikasna granica vrlo osjetljiva na ulazne parametre koje prima što su očekivani povrati, varijance i kovarijance imovina unutar portfelja. Problem ovdje je što je vrlo teško kvalitetno procijeniti ove parametre, a čak i male promijene u njima mogu dovesti do značajno različitih efikasnih granica te u konačnici raspodjela težina unutar portfelja.
- Pretpostavka normalnosti - MPT pretpostavlja da povrati slijede normalnu distribuciju što često nije slučaj na tržištima u stvarnosti. Distribucije povrata u financijama imaju teške repove što znači da su ekstremne situacije puno češće nego u normalnoj razdiobi što rezultira većim koeficijentom zašiljenosti (engl. *kurtosis*) nego kod normalne distribucije.
- Statične korelacije - Model pretpostavlja da korelacije između imovine ostaju konstantne, ali u stvarnosti se korelacije mogu značajno promijeniti (vidi sliku 4.1.), osobito tijekom razdoblja stresa na tržištu (razdoblja nestabilnosti, nesigurnosti obično karakterizirana povećanjem volatilnosti, manjkom likvidnosti i padom cijena imovina).

2.2. Maksimalna diverzifikacija

Nadovezujući se na načela diverzifikacije, metoda maksimalne diverzifikacije pojavila se kao alternativni pristup optimizaciji portfelja. Predstavljena je 2008. godine u radu *"Toward Maximum Diversification"* [2]. Ova metoda nastoji maksimizirati omjer diverzifikacije, definiran kao omjer utežane prosječne volatilnosti pojedinačne imovine i ukupne volatilnosti portfelja. Usredotočujući se na diverzifikaciju, a ne na eksplicitne prognoze povrata, metoda MD ima za cilj konstruirati portfelje koji su otporniji na pogreške u procjeni i tržišne neizvjesnosti. Ovaj je pristup posebno privlačan u okruženjima u kojima su točna predviđanja povrata izazovna, kao što su razdoblja velike volatilnosti tržišta ili strukturnih promjena.

Neka s $X_1, ..., X_n$ budu označene rizične imovine u tržišnom prostoru U. Radi pojednostavljenja, smatrat će se da su sve imovine X_i dionice. Neka je Σ matrica kovarijance, C matrica korelacije, a $\sigma_P = [\sigma_1, ..., \sigma_n]^T$ vektor volatilnosti tih imovina. Svaki portfelj P bit će definiran svojim težinama $w = [w_1, ..., w_n]^T$ pod uvjetom $\sum_{i=1}^n w_i$. Omjer diverzifikacije $D(P)$ za neki portfelj P definira se na sljedeći način:

$$D(P) = \frac{w^T \sigma_p}{\sqrt{w^T \Sigma w}}. \tag{2.17}$$

Neka je s Γ označen set linearnih ograničenja koje se odnose na težine portfelja P. Jedno od uobičajenih ograničenja je da sve težine moraju biti pozitivne. Portfelj koji maksimizira omjer diverzifikacije u tržišnom prostoru U uz ograničenja Γ naziva se maksimalno diverzificiran portfelj, označen kao $M(\Gamma, U)$. Bolje razumijevanje kako funkcionira diverzifikacija pri konstrukciji portfelja bit će dano u sljedeća dva primjera.

Neka se tržišni svijet sastoji od dviju dionica A i B, s koeficijentom korelacije manjem od jedan te s volatilnostima redom 15% i 30%. U ovom slučaju, diverzifikacija podrazumijeva da obje dionice jednako doprinose ukupnoj volatilnosti portfelja. Radi toga, težine u maksimalno diverzificiranom portfelju bit će 66.6% za dionicu A te 33.3% za dionicu B (obrnuto proporcionalno volatilnosti).

Sada neka se tržišni svijet sastoji od tri dionice. Neka na primjer prve dvije pripadaju istom tržišnom sektoru te imaju korelaciju 0.9, a zadnja pripada nekom drugom sektoru s korelacijom 0.1 prema svakoj od prve dvije dionce. Neka su sve volatilnosti imovina jednake. Težine maksimalno diverzificiranog portfelja bit će 25.7% za svaku od prve dvije dionice te 48.6% za zadnju dionicu. Na ovom primjeru vidi se da kada imovine imaju nisku korelaciju među sobom, optimizacija portfelja dodjeljuje veće težine tim imovinama kako bi smanjila ukupnu volatilnost. Suprotno, imovine koje su visoko korelirane s ostatkom portfelja dobijaju niže težine jer njihov zajednički rizik povećava ukupnu vo-

latilnost.
Omjer diverzifikacije za svaki dugoročni portfelj bit će strogo veći od jedan osim za portfelje sastavljene od samo jedne imovine, u tom slučaju bit će točno jedan. Ako je očekivani povrat imovina jednak njihovom riziku (volatilnosti), onda $E[R_P] = kP^T\sigma_P$ gdje je k neka konstanta, tada je maksimizacija $D(P)$ ekvivalentna maksimizaciji $\frac{E[R_P]}{\sqrt{w^T\Sigma w}}$ što je zapravo Sharpeov omjer. U tom slučaju, maksimalno diverzificirani portfelj u stvari je jednak tangencijalnom portfelju spomenutom u prošlom odjeljku.

U nastavku će biti analizirana svojstva korelacija imovina u kontekstu maksimalno diverzificiranog portfelja.Korelacija proizvoljnog portfelja P s maksimalno diverzificiranim portfeljem M može se izračunati, pri čemu se pokazuje da je portfelj M obrnuto proporcionalan s standardnom devijacijom σ i korelacijama C, što implicira da vrijedi:

$$M = \kappa\sigma^{-1}C^{-1}\mathbf{1} \tag{2.18}$$

gdje je κ konstantan faktor, σ je dijagonalna matrica volatilnosti imovina, a C matrica korelacije. Stoga vrijedi:

$$\rho_{P,M} = \frac{w^T\sigma C\sigma M}{\sigma_P\sigma_M} = \frac{w^T\sigma C\sigma\kappa\sigma^{-1}C^{-1}\mathbf{1}}{\sigma_P\sigma_M} = \frac{\sum_i w_i\sigma_i\kappa}{\sigma_P\sigma_M} = D(P)\frac{\kappa}{\sigma_M}. \tag{2.19}$$

Ovo znači da je korelacija portfelja P s maksimalno diverzificiranim portfeljem M proporcionalna omjeru diverzifikacije portfelja P, odnosno $D(P)$. Sada će biti razmotrene korelacije pojedinačnih dionica i maksimalno diverzificiranog portfelja. Omjer diverzifikacije svake pojedinačne dionice iznosi jedan jer nema diverzifikacije. Koristeći formulu 2.19 možemo izračunati korelaciju imovine i s vektorom maksimalno diverzificiranog portfelja M uz to da napravimo vektor težine $\vec{w}_i$ gdje je i-ta težina vektora jednaka jedan, a sve ostale nula. Dobivamo

$$\rho_{i,M} = \frac{\kappa}{\sigma_M}. \tag{2.20}$$

Zanimljivo je da je korelacija imovine i s maksimalno diverzificiranim portfeljem ista za svaku imovinu. Stoga možemo identificirati maksimalno diverzificirani portfelj kao onaj u kojem sva imovina ima istu pozitivnu korelaciju s njim. Poseban slučaj kada je P zapravo jednak maksimalno diverzificiranom portfelju M u formuli 2.19 dovodi nas do rezultata:

$$\varkappa = \frac{\sigma_M}{D(M)} \tag{2.21}$$

čime smo identificirali konstantu $\varkappa$. Stoga, možemo raspisati korelaciju između portfelja P i maksimalno diverzificiranog portfelja M kao omjer njihovih diverzifikacijskih omjera:

$$\rho_{P,M} = \frac{D(P)}{D(M)}. \tag{2.22}$$

Znajući ovo, možemo dobiti jednofaktorski (diverzifikacijski) model koji identificira korelaciju kao omjer diverzifikacijskih omjera:

$$R_p = \alpha_p + \frac{\sigma_P}{\sigma_M}\frac{D(P)}{D(M)}R_M + \epsilon_P \tag{2.23}$$

gdje R_P predstavlja povrat, a α_P i ϵ_P su redom konstanta i pogreška, obično povezane regresijom.

2.2.1. Prednosti i nedostatci MD-a

Prednosti ove metode su:

- Smanjeni koncentracijski rizik - MD izbjegava prekomjernu koncentraciju u određenim imovinama ili sektorima, smanjujući utjecaj nepovoljnih događaja koji utječu na jednu imovinu ili sektor.
- Robusnost unatoč nestabilnostima tržišta - fokus na niske korelacije i uravnoteženu volatilnost čini MD portfelje otpornijima tijekom tržišnih padova.

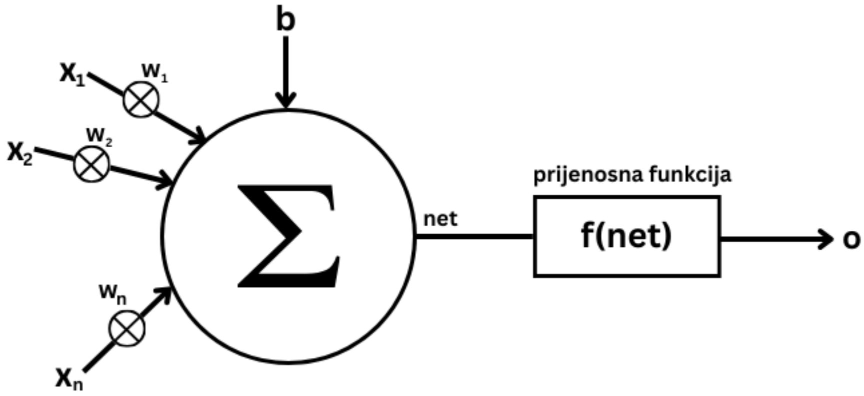
Glavna ograničenja MD-a su:

- Ovisnost o ulaznim podacima - kao i druge kvantitativne metode (kao što smo vidjeli kod Markowitzeve MV analize), MD se oslanja na točne procjene volatilnosti i korelacija, što može biti teško predvidjeti.
- Slabiji učinak na bikovskim tržištima (engl. *bull market*) - metoda daje prioritet diverzifikaciji što znači da može dodijeliti značajnu težinu imovini s nižim očekivanim prinosima ako poboljšava diverzifikaciju. Na tržištima bikova (stanje financijskog tržišta u kojem cijene nastavljaju rasti), imovina s višim očekivanim povratima često ima bolje rezultate, ali metoda maksimalne diverzifikacije mogla bi imati manju težinu od te imovine s visokim prinosima, što dovodi do neoptimalne izvedbe.
- Složenost - proces optimizacije može biti računalno intenzivan, posebno za velike portfelje s mnogo imovine.
- Ne jamči bolji učinak - iako MD smanjuje rizik, ne jamči veće povrate. Njegov uspjeh ovisi o kvaliteti ulaznih podataka i tržišnim uvjetima.

3. Duboko učenje

Duboko učenje je grana umjetne inteligencije i podskup strojnog učenja koja koristi umjetne neuronske mreže kako bi učila iz podataka. Tijekom prošlog desetljeća, duboko učenje revolucioniralo je polja kao što su obrada prirodnog jezika, računalni vid, prepoznavanje govora, pa čak i financije, zbog svoje sposobnosti modeliranja izrazito nelinearnih odnosa i otkrivanja uvida koje tradicionalni algoritmi teško postižu. Jedna od ključnih inovacija koja je duboko učenje učinila tako učinkovitim je njegova sposobnost rada s nestrukturiranim podacima, kao što su slike, tekst i zvuk. Tradicionalni pristupi strojnog učenja obično zahtijevaju opsežno pretprocesiranje i inženjerstvo značajki. Nasuprot tome, modeli dubokog učenja mogu automatski izdvojiti i optimizirati značajke kroz treniranje, čineći ih svestranijim i skalabilnijim za složene zadatke.
Duboko učenje zasniva se na dubokim neuronskim mrežama, koje u svojoj srži oponašaju način na koji ljudski mozak obrađuje informacije uz pomoć međusobno povezane mreže neurona. Svaki neuron sastoji se od ulaznih podataka x_i koji se množe sa osjetljivošću toga ulaza (težine w_i) te se ukupnoj sumi dodaje pomak b (engl. *bias*). Dobivena vrijednost propušta se kroz aktivacijsku funkciju koja je nelinearna kako bi model mogao učiti nelinearne odnose između podataka. Opisani neuron možemo vidjeti na slici 3.1.

Duboke neuronske mreže koriste umjetne neurone raspoređene u slojeve. Svaki sloj obrađuje ulazne podatke i prosljeđuje rezultat sljedećem sloju, što omogućuje modelu da nauči komplicirane značajke iz sirovih podataka. Ovakvo napredovanje izračuna kroz mrežu naziva se unaprijedni prolaz. Ulazni i izlazni slojevi neuronske mreže nazivaju se vidljivi slojevi, a svi slojevi između su skriveni i mogu biti različitih dimenzija. Ulazni sloj je mjesto gdje se unose podaci za učenje, a u izlaznom sloju se vrši predviđanje ili klasifikacija. Drugi bitan postupak naziva se propagacija pogreške unatrag (engl. *back-*

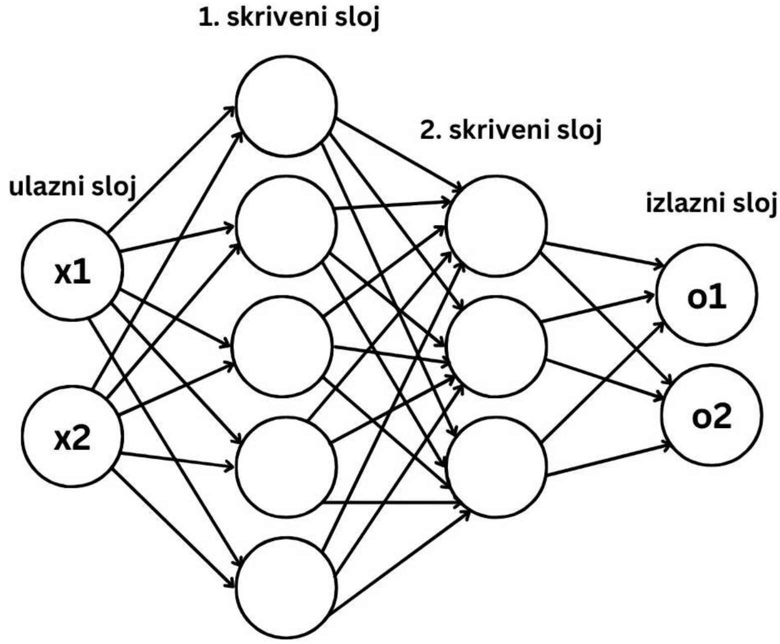


Slika 3.1. Pojedinačni neuron

propagation) koji koristi algoritme, kao npr. gradijentni spust, za izračunavanje pogreške predikcija kako bi ažurirao težine i pomake svakog neurona, pomicanjem unatrag kroz slojeve. Unaprijedni prolaz i propagacija pogreške unatrag zajedno omogućavaju neuronskoj mreži da napravi predviđanja i ispravi pogreške te s vremenom postane sve preciznija u predikcijama.

Ključni aspekt dubokog učenja je optimizacija jer se modeli uče tako da minimiziraju funkciju pogreške (engl. *loss function*). Funkcija pogreške mjeri koliko se predikcije modela razlikuju od stvarnih vrijednosti. Cilj optimizacije je pronaći skup parametara koji minimiziraju ovu funkciju pogreške. Kako su neuronske mreže obično visoko nelinearne i imaju veliki broj parametara, optimizacija postaje složen problem koji zahtijeva sofisticirane algoritme.

Jedan od najčešće korištenih algoritama za optimizaciju u dubokom učenju je gradijentni spust (engl. *gradient descent*). Ovaj algoritam iterativno ažurira parametre modela u smjeru negativnog gradijenta funkcije pogreške, što dovodi do smanjenja pogreške. Međutim, kako se modeli i skupovi podataka povećavaju, osnovni gradijentni spust postaje neučinkovit. Zbog toga su razvijene različite varijante, radi poboljšanja brzine i stabilnosti učenja, koje će biti razmotrene u nastavku.



Slika 3.2. Potpuno povezana neuronska mreža

3.1. Gradijentni spust

Gradijentni spust je iterativni optimizacijski algoritam koji se koristi za minimiziranje funkcije pogreške u dubokom učenju. Cilj algoritma je pronaći minimum funkcije pogreške $L(\theta)$ gdje su θ parametri modela. Gradijent pokazuje smjer najstrmijeg porasta funkcije pogreške. Ideja gradijentnog spusta je krenuti od neke početne točke (početnog vektora θ) i postupno se spuštati niz površinu funkcije $L(\theta)$ u smjeru suprotnome od gradijenta u točki θ. To se ponavlja dok se ne spusti u točku u kojoj je gradijent jednak nuli ili je barem dovoljno blizu nuli. Parametre θ u svakoj iteraciji gradijentnog spusta ažuriramo na ovaj način:

$$\theta_{new} = \theta_{old} - \eta \nabla L(\theta) \tag{3.1}$$

gdje je η stopa učenja koja određuje veličinu koraka koje radimo spuštajući se prema minimumu.

Gradijentni spust nakon nekog vremena doći će do točke za koju vrijedi $\nabla L(\theta) = 0$. No to ne znači da je ta točka globalni minimum funkcije pogreške. Naime, funkcija pogreške općenito može imati jedan ili više lokalnih minimuma. U tom slučaju gradijentni spust, ovisno o početnoj točki, lako može zaglaviti u nekom od lokalnih minimuma. Iz navedenog razloga u dubokom učenju se preferiraju funkcije koje su konveksne. Takve funkcije imaju samo jedan minimum, dakle, taj jedan minimum onda je sigurno i globalni minimum jer lokalnih minimuma nema. Optimizacijom konveksnih funkcija bavi se konveksna optimizacija.
Rečeno je da η određuje stopu učenja, odnosno koliko velike korake algoritam radi dok se spušta. Stopa učenja trebala bi biti odabrana tako da nije ni prevelika ni premalena. Naime, ako se odabere prevelika stopa učenja, može se dogoditi da postupak ne konvergira već da divergira. S druge strane, ako se odabere premala stopa učenja, postupak će konvergirati, ali sporo. Stopa učenja jedan je od hiperparametara koje je najteže postaviti jer značajno utječe na izvedbu modela. Zbog toga se javila ideja automatski prilagođavati te stope učenja tijekom cijelog učenja. Takvi algoritmi nazivaju se algoritmi s adaptivnom stopom učenja od kojih će biti razmotren Adam (engl. *adaptive moments*) jer je korišten u ovom radu.

3.1.1. Adam

Adam je napredni optimizacijski algoritam koji kombinira prednosti adaptivnih stopa učenja i koncept momenta za poboljšanje učinkovitosti i stabilnosti gradijentnog spusta. Naširoko se koristi u dubokom učenju jer automatski prilagođava brzinu učenja za svaki parametar i uključuje informacije o prošlim gradijentima prilikom procesa optimizacije. Adam kombinira dvije ključne ideje:

- Adaptivna stopa učenja - algoritam mijenja stopu učenja za svaki parametar na temelju magnitude prošlih gradijenata.
- Moment - algoritam koristi eksponencijalno otežane pomične prosjeke prošlih gradijenata kako bi izgladio šum gradijenata i ubrzao konvergenciju.

Adam održava dva pomična prosjeka za svaki parametar:

- Prvi moment (srednja vrijednost gradijenata) - eksponencijalno otežan prosjek prošlih gradijenata. Ovaj dio djeluje kao izglađena verzija gradijenta. Pomaže ubrzati konvergenciju smanjenjem oscilacija kada gradijenti sadrže mnogo šuma.

- Drugi moment (varijanca gradijenata) - eksponencijalno otežan prosjek kvadrata prošlih gradijenata. Ovaj dio hvata varijancu gradijenata. Adaptivno skalira stopu učenja, dopuštajući veća ažuriranja za parametre s malim gradijentima i manja ažuriranja za parametre s velikim, bučnim gradijentima.

Algoritam Adam odvija se na sljedeći način. Prvo se računa gradijent funkcije gubitka L s obzirom na parametre θ:

$$g_t = \nabla L(\theta_t). \tag{3.2}$$

Prvi moment ažurira se na način:

$$s_t = \rho_1 s_{t-1} + (1 - \rho_1) g_t. \tag{3.3}$$

Zatim se ažurira drugi moment:

$$r_t = \rho_2 r_{t-1} + (1 - \rho_2) g_t^2. \tag{3.4}$$

Nakon toga se izračunavaju prvi i drugi moment korigirani za pristranost:

$$\hat{s} = \frac{s}{1 - \rho_1}, \ \hat{r} = \frac{r}{1 - \rho_2}. \tag{3.5}$$

Završno, novi parametri računaju se na sljedeći način:

$$\theta_{new} = \theta_{old} - \eta \frac{\hat{s}}{\sqrt{\hat{r}} + \delta}. \tag{3.6}$$

gdje je η stopa učenja, a δ predstavlja malu konstantu kako bi se izbjeglo dijeljenje s nulom (preporučena vrijednost je 10^{-8}). ρ_1 i ρ_2 su hiperparametri te vrijedi $\rho_1, \rho_2 \in [0, 1)$

(tipično se postavljaju redom na 0.9 i 0.999).

3.1.2. Učenje s mini-grupama

U strojnom učenju, funkcija cilja (funkcija gubitka) obično je zbroj primjera u skupu za treniranje. Drugim riječima,ukupni je gubitak prosjek gubitaka za svaki pojedinačni primjer. Izračunavanje točnog gradijenta funkcije gubitka zahtjeva procjenu modela na svakom primjeru u skupu podataka, što je računski skupo. Kako bi se to riješilo, optimizacijski algoritmi u strojnom učenju često koriste procjenu gradijenta gdje se gradijent aproksimira korištenjem malog, nasumično uzorkovanog podskupa primjera odnosno mini-grupe (engl. *minibatch*). Ovaj pristup uravnotežuje računsku učinkovitost i točnost, čineći ga temeljem moderne optimizacije strojnog učenja.
Umjesto da se gradijent izračunava koristeći cijeli skup podataka, procjenjuje se pomoću mini-grupa. Ovo je mnogo brže i još uvijek pruža dobru aproksimaciju pravog gradijenta. Standardna pogreška srednje vrijednosti (koliko je točna naša procjena) smanjuje se s korijenom broja uzoraka ($\frac{\sigma}{n}$). To pokazuje da su prednosti korištenja većih mini-grupa sve manji, kako n raste (korijen sporo raste kako se n povećava). Većina optimizacijskih algoritama konvergira brže kada koristi približne gradijente izračunate iz malih mini-grupa. U mnogim skupovima podataka primjeri su suvišni ili vrlo slični. To znači da mala mini-grupa često može pružiti dobru procjenu gradijenta, čak i ako ne uključuje sve primjere. Važno je napomenuti da mini-grupe moraju biti odabrane nasumično kako bi se osiguralo da je procjena gradijenta nepristrana. Ako su primjeri povezani (npr. uzastopni primjeri u skupu podataka), procjena gradijenta može biti pristrana. Na primjer, u medicinskom skupu podataka, ako svi primjeri u mini-grupi dolaze od istog pacijenta, procjena gradijenta neće biti reprezentativna za cijeli skup podataka. Da bi se to izbjeglo, skup podataka treba promiješati prije odabira mini-grupa. Također, manje grupe unose šum u procjenu gradijenta, što može djelovati kao oblik regularizacije i poboljšati generalizaciju. Međutim, to može zahtijevati manju stopu učenja i više koraka treniranja.

3.2. Povratne neuronske mreže

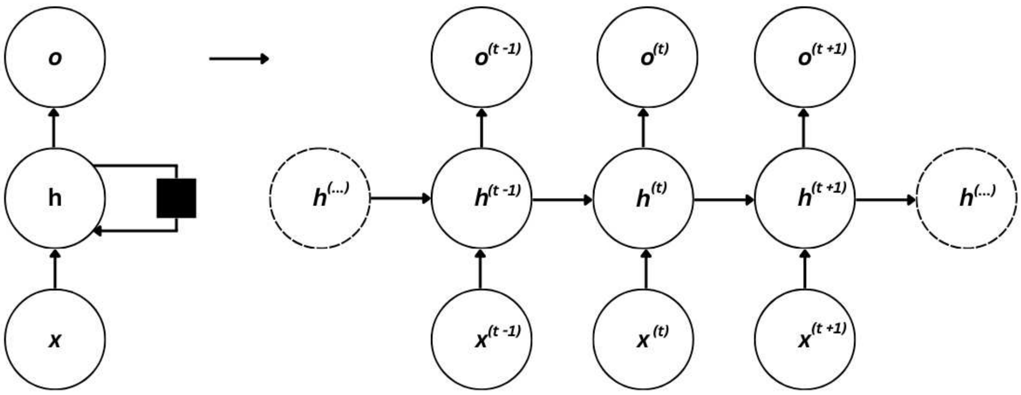
Tradicionalne neuronske mreže, kao što su potpuno povezane mreže, rade na ulaznim i izlaznim podacima fiksne veličine. Međutim, mnogi problemi u stvarnom svijetu uključuju sekvencijalne podatke, gdje su prošle informacije ključne za predviđanje budućih ishoda. Na primjer, u rečenici značenje riječi ovisi o riječima koje su bile prije nje. Ova ovisnost zahtijeva model koji može obuhvatiti vremenske odnose.

Povratne neuronske mreže (engl. *recurrent neural networks - RNN*) razvijene su za obradu sekvencijalnih podataka uvođenjem vremenske dimenzije u izračune neuronske mreže. Za razliku od standardnih mreža, RNN-ovi održavaju skriveno stanje koje nosi informacije iz prethodnih vremenskih koraka, omogućujući mreži da zadrži memoriju tijekom vremena. Ova memorija je kodirana u skrivenom stanju mreže, koje se ažurira u svakom vremenskom koraku na temelju trenutačnog unosa i prethodnog skrivenog stanja. Zbog toga su posebno prikladni za zadatke u kojima je bitan povijesni kontekst.

Da bismo prešli s višeslojnih mreža na povratne mreže, moramo iskoristiti ideju dijeljenja parametara u različitim dijelovima modela. Dijeljenje parametara omogućuje proširenje i primjenu modela na primjere različitih duljina. Kad bismo imali zasebne parametre za svaku vrijednost vremenskog indeksa, ne bismo mogli generalizirati na duljine sekvenci koje se ne vide tijekom treniranja, niti dijeliti statističku snagu kroz različite duljine sekvenci i kroz različite pozicije u vremenu. Takvo dijeljenje je posebno važno kada se određena informacija može pojaviti na više mjesta unutar niza. Na primjer, razmotrite dvije rečenice "Otišao sam u Italiju 2009." i "2009. otišao sam u Italiju." Ako se traži od modela strojnog učenja da pročita svaku rečenicu i izdvoji godinu u kojoj je pripovjedač otišao u Italiju, tražit će se da prepozna godinu 2009. kao relevantnu informaciju, bilo da se pojavljuje u šestoj ili u drugoj riječi rečenica. Pretpostavimo da smo istrenirali mrežu koja obrađuje rečenice fiksne duljine. Tradicionalna potpuno povezana mreža unaprijed bi imala zasebne parametre za svaku ulaznu značajku, pa bi morala naučiti sva pravila jezika zasebno na svakom mjestu u rečenici. S druge strane, povratna neuronska mreža dijeli istu težinu kroz nekoliko vremenskih koraka. Svaki član izlaza je funkcija prethodnih članova izlaza. Svaki član izlaza proizvodi se korištenjem istog pravila ažuriranja primijenjenog na prethodne izlaze. Ova povratna formulacija rezultira dijeljenjem parametara kroz vrlo dubok računalni graf.

RNN-ovi se također mogu primijeniti dvodimenzionalno na prostorne podatke kao što su slike, a čak i kada se primjenjuju na podatke koji uključuju vrijeme, mreža može imati veze koje idu unatrag u vremenu, pod uvjetom da se cijeli niz promatra prije nego što se dostavi mreži.

Povratni modeli inspirirani su dinamičkim sustavima. Model ažurira skriveno stanje $h^{(t)}$ na temelju ulaza $x^{(t)}$. Petlja omogućuje prijenos informacija s jednog koraka mreže na sljedeći. Iako radi ove petlje RNN-ovi mogu djelovati zbunjujućima, ispada da oni nisu toliko različiti od normalne neuronske mreže. Povratna neuronska mreža može se zamisliti kao više kopija iste mreže, od kojih svaka prosljeđuje poruku nasljedniku. Ako se odmota petlja dobiva se lančana struktura prikazana na slici 3.3.



Slika 3.3. Osnovna struktura povratne neuronske mreže

Osnovna arhitektura RNN-a sadrži:

- ulazni sloj - prima ulazne podatke u svakom vremenskom koraku
- skriveni sloj - održava skriveno stanje koje bilježi informacije iz prethodnih vremenskih koraka
- izlazni sloj - "proizvodi" izlaz u svakom vremenskom trenutku

Skriveno stanje $h^{(t)}$ u trenutku t računa se na sljedeći način:

$$h^{(t)} = g(W_{hh}h^{(t-1)} + W_{xh}x^{(t)} + b_h) = g(a(t)) \tag{3.7}$$

gdje su W_{hh}, W_{xh}, b_h parametri povratne afine transformacije. S $a(t)$ označavamo linearnu povratnu mjeru, a g je nelinearna aktivacijska funkcija (obično tangens hiperbolni ili sigmoida). Stanje $h^{(t)}$ predstavlja svu viđenu informaciju, ako je model dobro naučen h

će kodirati značenje teksta (informacija). Matrica W_{xh} projicira ulaz u prostor reprezentacija stanja te je ovaj dio odgovoran za odbacivanje nepotrebnih informacija. Matrica W_{hh} modelira evoluciju stanja, opisuje utjecaj vremena i ulaza te također odbacuje nepotrebne informacije iz stanja. Valja napomenuti da se dimenzije ulaza i skrivenog stanja mogu razlikovati. Izlazni sloj projicira skriveno stanje na prostor predikcija na način:

$$o^{(t)} = W_{hy}h^{(t)} + b_0. \tag{3.8}$$

Ovaj dio filtrira nevažne informacije te na kraju vektor $o^{(t)}$ sadrži logite (sirove, nenormalizirane rezultate koje daje završni sloj modela, obično prije primjene bilo koje aktivacijske funkcije) predikcija u trenutku t. Računanje gradijenata povratnog modela često nazivamo širenjem unatrag kroz vrijeme (engl. *backpropagation through time*). Širenje unatrag kroz vrijeme odgovara standardnoj propagaciji pogreške unazad kroz razmotani model gdje povratnu ćeliju možemo promatrati kao sloj običnog potpuno povezanog modela.

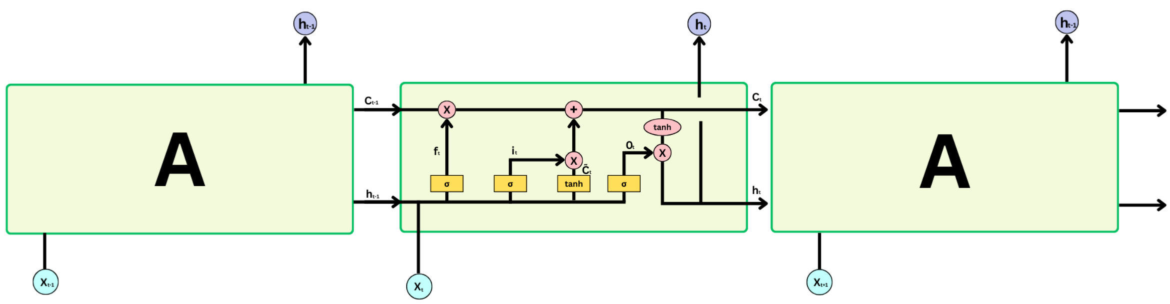
Iako vrlo moćne, povratne neuronske mreže pate od dva glavna problema, oba uzrokovana dijeljenjem parametara u uzastopnim operacijama. Prvi od njih je nestajući gradijent. Problem s nestajućim gradijentom javlja se kada gradijenti korišteni za ažuriranje parametara modela tijekom širenja unatrag kroz vrijeme postanu izuzetno mali. To otežava mreži učenje dugoročnih ovisnosti jer gradijenti za ranije vremenske korake postaju zanemarivi, zbog čega model ne može učinkovito ažurirati težine te model efektivno prestaje učiti.

Drugi problem je problem eksplodirajućeg gradijenta koji je suprotan prethodno navedenom problemu. U ovom slučaju, gradijenti rastu eksponencijalno dok se šire unatrag kroz vrijeme, što dovodi do iznimno velikih ažuriranja parametara modela. To može uzrokovati nestabilnost i divergenciju modela tijekom učenja.

Jedna od privlačnosti RNN-ova je ideja da bi mogli povezati prethodne informacije sa sadašnjim zadatkom, kao što bi korištenje prethodnih informacija u tekstu moglo informirati o razumijevanju neke rečenice dalje u tekstu. Ponekad samo trebamo pogledati nedavne informacije kako bismo obavili sadašnji zadatak. Na primjer, jezični model koji pokušava predvidjeti sljedeću riječ na temelju prethodnih. Ako se pokušava predvidjeti posljednja riječ u izrazu "oblaci su na nebu", ne treba nam dodatni kontekst - prilično je očito da će posljednja riječ biti nebo. U takvim slučajevima, gdje je razlika između relevantnih informacija i potrebnog mjesta mala, RNN-ovi mogu naučiti koristiti prošle informacije. Međutim, postoje i slučajevi u kojima nam treba više konteksta. Pokušajte predvidjeti posljednju riječ u tekstu "Odrastao sam u Italiji... tečno govorim talijanski." Nedavne informacije sugeriraju da je sljedeća riječ vjerojatno naziv jezika, ali ako želimo suziti koji jezik, potreban nam je kontekst Italije, iz (potencijalno) daleke prošlosti. Sasvim je moguće da razlika između relevantnih informacija i točke u kojoj su potrebne postane vrlo velika. Nažalost, kako ta razlika raste, RNN-ovi postaju nesposobni naučiti povezivati informacije.

3.3. Povratna ćelija s dugoročnom memorijom

Povratnu ćeliju s dugoročnom memorijom uveli su Hochreiter i Schmidhuber 1997. kako bi se pozabavili problemima nestajanja i eksplodiranja gradijenta u osnovnim RNN-ovima. LSTM-ovi su vrsta usmjernog RNN-a koji može učinkovitije naučiti dugoročne ovisnosti uvođenjem memorijske ćelije i mehanizama usmjeravanja. Osnovna shema jednog sloja LSTM-a razmotanog kroz vrijeme prikazana je na slici 3.4.:

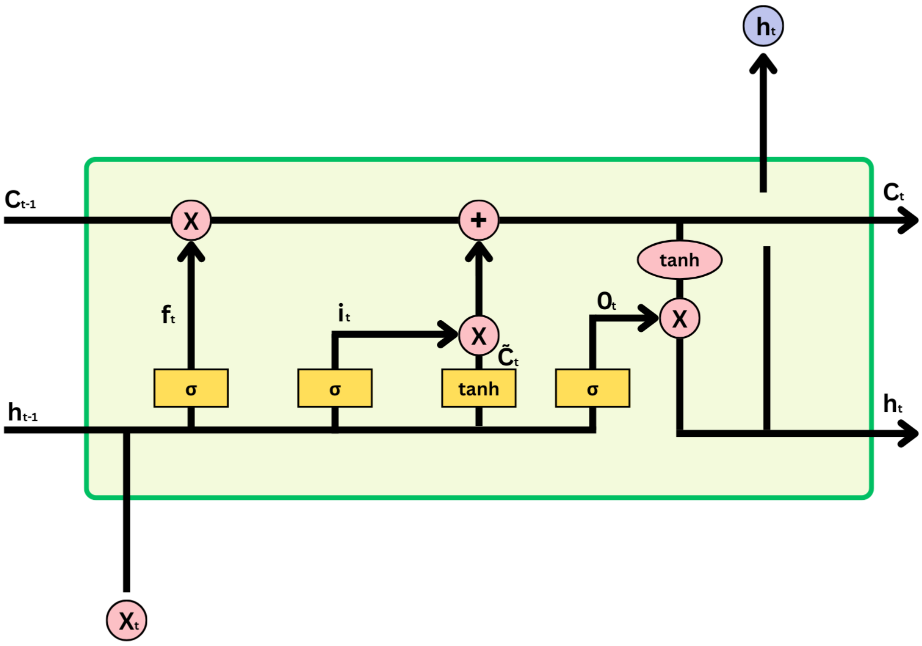


Slika 3.4. Shema LSTM mreže, slika napravljena po [3]

Ključ za LSTM je stanje ćelije $c^{(t)}$, vodoravna linija koja prolazi kroz vrh dijagrama. Stanje ćelije je nešto poput pokretne trake koja pamti viđenu informaciju te prolazi ravno niz cijeli lanac, uz samo neke manje linearne interakcije. Vrlo je lako da informacije teku nepromijenjene. LSTM ima sposobnost uklanjanja ili dodavanja informacija stanju ćelije, pažljivo reguliranim strukturama zvanim vratima. Vrata su način da se opcionalno propuste informacije. LSTM sadrži tri vrata, svaka sa svojim parametrima, koja čuvaju i kontroliraju stanje ćelije:

- Vrata zaboravljanja - kontrolira koje informacije će biti izbačene iz stanja ćelije.
- Vrata ulaza - kontrolira koje informacije će biti dodane u stanje ćelije.
- Vrata izlaza - kontrolira koje informacije izlaze iz stanja ćelije.

Sve navedeno možemo pronaći na uvećanoj slici jedne ćelije LSTM-a 3.5. Sva vrata omogućuju LSTM mrežama da selektivno zadrže ili odbace informacije dok teku kroz mrežu, čime zapravo uče dugoročne ovisnosti.



Slika 3.5. Shema LSTM mreže, slika napravljena po [3]

Prvi korak u LSTM-u je odlučiti koje informacije će se izbaciti iz stanja ćelije. Ovu odluku donosi sigmoidni sloj koji se naziva vrata zaboravljanja $f^{(t)}$ koji se dobiva na sljedeći način:

$$f^{(t)} = \sigma(W_{fhh}h^{(t-1)} + W_{fxh}x^{(t)} + b_{fh}). \tag{3.9}$$

Vrata zaboravljanja gledaju na prethodno skriveno stanje $h^{(t-1)}$ i ulaz $x^{(t)}$ te pomoću sigmoide daju broj između 0 i 1 za svaki element u stanju ćelije $c^{(t-1)}$. Sigmoidna funkcija može se probabilistički interpretirati kao dio informacije koji se želi zadržati na način da 1 predstavlja potpuno zadrži ovo, a 0 potpuno zaboravi ovo. Na primjer, jezični model koji pokušava predvidjeti sljedeću riječ na temelju svih prethodnih, stanje ćelije može uključivati spol sadašnjeg subjekta, tako da se mogu koristiti ispravne zamjenice. Kad se pojavi novi subjekt, želi se zaboraviti spol starog subjekta.

Sljedeći korak je odlučiti koje nove informacije će biti pohranjene u stanje ćelije. Ovaj dio sastoji se od dva dijela. Prvo, sigmoidni sloj zvan vratima ulaza $i^{(t)}$ odlučuje koje će se vrijednosti ažurirati. Računa se na način:

$$i^{(t)} = \sigma(W_{ihh}h^{(t-1)} + W_{ixh}x^{(t)} + b_{ih}). \tag{3.10}$$

Sljedeći je sloj koji stvara doprinos stanju ćelije $\tilde{c}^{(t)}$ i predstavlja nove informacije koje bi se mogle dodati stanju ćelije:

$$\tilde{c}^{(t)} = \tanh(W_{chh}h^{(t-1)} + W_{cxh}x^{(t)} + b_{ch}). \tag{3.11}$$

U prethodno spomenutom primjeru jezičnog modela, ovo je dio gdje se želi dodati spol novog subjekta u stanje ćelije, kako bismo zamijenili stari koji zaboravljamo. U sljedećem koraku kombinirat će se prethodna dva koraka kako bismo ažurirali stanje ćelije. Staro stanje ćelije množi se s $f^{(t)}$, kako bi se "zaboravile" nebitne informacije. Zatim se tome dodaje $i^{(t)} \odot \tilde{c}^{(t)}$ što predstavlja nove informacije skalirane za onoliko koliko je odlučeno da treba ažurirati svaku vrijednost stanja ćelije. Valja primijetiti da je stanje ćelije teško izmijeniti jer svaku promjenu moraju podržati dvoja vrata ($f^{(t)}$ i $i^{(t)}$). Stanje ćelije dobivamo na način:

$$c^{(t)} = f^{(t)} \odot c^{(t-1)} + i^{(t)} \odot \tilde{c}^{(t)} \tag{3.12}$$

gdje je $\odot$ Hadamardov produkt koji predstavlja vektorsko množenje po elementima na način:

$$a \odot b = \begin{pmatrix} a_0 b_0 \\ \vdots \\ a_n b_n \end{pmatrix}.$$

Zadnja vrata $o^{(t)}$ nazivaju se izlaznim vratima. Njihov zadatak je izvlačenje korisnih informacija iz trenutačnog stanja ćelije koje se prikazuju kao izlaz. Izlazna vrata računaju se na način:

$$o^{(t)} = \tanh(W_{ohh} h^{(t-1)} + W_{oxh} x^{(t)} + b_{oh}). \tag{3.13}$$

LSTM održava skriveno stanje, koje djeluje kao kratkoročna memorija mreže. Skriveno stanje ažurira se na temelju unosa, prethodnog skrivenog stanja (sadržano u $o^{(t)}$) i trenutnog stanja memorijske ćelije:

$$h^{(t)} = o^{(t)} \odot \tanh(c^{(t)}). \tag{3.14}$$

LSTM-ovi su revolucionirali područje dubokog učenja svojom sposobnošću učenja dugoročnih ovisnosti u sekvencijskim podacima. Zahvaljujući memorijskoj ćeliji i triju vrata (zaboravljanja, ulaza i izlaza), LSTM-ovi mogu selektivno zadržati ili odbaciti informacije, što im omogućuje da efikasno rukuju dugim sekvencama i izbjegnu probleme nestajanja ili eksplodiranja gradijenata. Ova fleksibilnost čini LSTM-ove posebno korisnima u aplikacijama kao što su obrada prirodnog jezika, prepoznavanje govora i vremenske serije (u financijama).

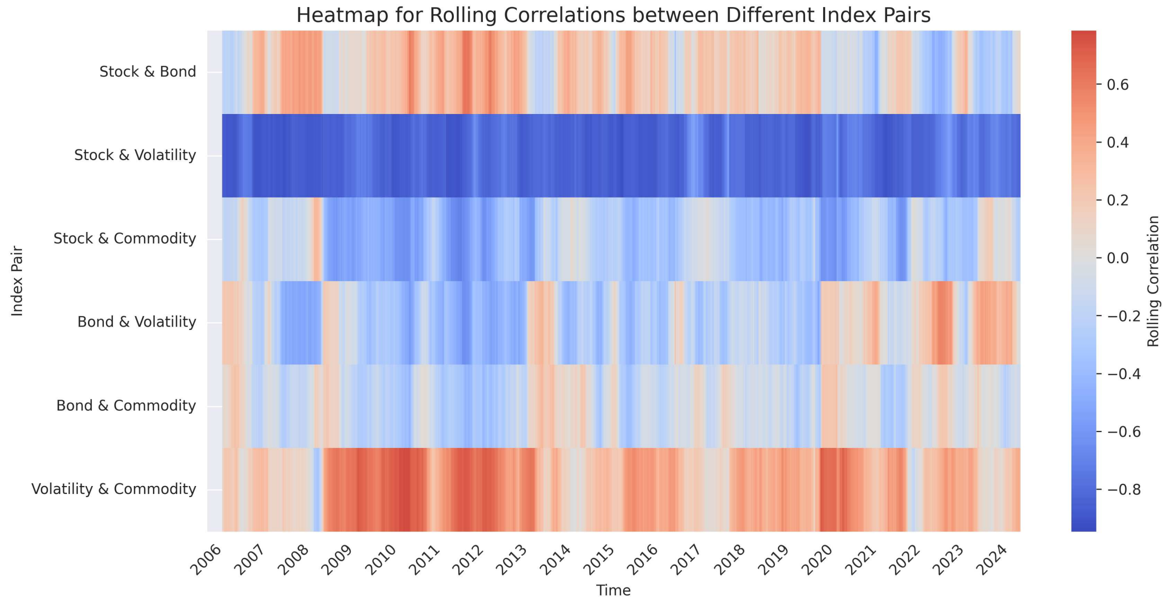
4. Implementacija

4.1. Podaci

U ovome radu koriste se ETF-ovi (engl. *exchange traded funds*). ETF je investicijski fond čijim se udjelima trguje na burzi, slično kao i dionicama pojedinačnih kompanija. ETF-ovi prate vrijednosti tržišnih indeksa koji predstavljaju promjene vrijednosti grupe dionica ili drugih financijskih instrumenata te služe kao mjerilo performansi tržišta ili određenog dijela tržišta. ETF-ovi pružaju široku izloženost tržištu uz visoku likvidnost i niske troškove. Konkretno, koriste se sljedeće četiri vrste ETF-ova:

- Indeks ukupnog tržišta dionica SAD-a (engl. *US Total Stock Market Index - VTI*): Predstavlja cjelokupno tržište dionica u SAD-u.
- Agregatni indeks obveznica SAD-a (engl. *US Aggregate Bond Index - AGG*): Prati izvedbu tržišta obveznica u SAD-u.
- Indeks roba SAD-a (engl. *US Commodity Index - DBC*): Pruža izloženost košarici roba.
- Indeks volatilnosti (engl. *Volatility Index - VIX*): Odražava tržišna očekivanja o volatilnosti u bliskoj budućnosti.

Ovi indeksi su odabrani jer imaju nisku ili čak negativnu korelaciju čime se povećava diverzifikacija portfelja. Naime, ako su vrijednosnice negativno korelirane, negativni prinos jedne ujedno implicira pozitivni prinos druge vrijednosnice čime se volatilnost portfelja smanjuje. Trgovanje pojedinačnom imovinom unutar iste klase imovina često dovodi do redundancije, budući da ona pokazuje snažnu pozitivnu korelaciju s tržišnim indeksom. Usredotočujući se na te ETF-ove, cilj je izgraditi diverzificirani portfelj s višim omjerom povrata i rizika.



Slika 4.1. Korelacije korištenih ETF-ova

Transakcijski podatci najčešće se uzorkuju u nekim vremenskim intervalima (npr. 10 min, 1 dan, ili 1 mjesec). Neka je u zadanom intervalu obavljeno n transakcija u trenutcima $t_1, t_2, ..., t_n$, s cijenama $P_1, P_2, ..., P_n$ i količinama $V_1, V_2, ..., V_n$. Podaci se uzorkuju na sljedeći način:

- Prva (open) cijena: $P_{open} = P1$
- Zadnja (close) cijena: $P_{close} = P_n$
- Najviša (high) cijena: $P_{high} = max(P_1, P_2, ..., P_n)$
- Najniža (low) cijena: $P_{low} = min(P_1, P_2, ..., P_n)$
- Ukupna količina (volume): V = $\sum_{i=1}^{n} V_i$

OHLCV (open-high-low-close-volume)– jedan od najčešće korištenih standarda u agregiranju tržišnih podataka.

Još jedna bitna informacija koja se najčešće pojavljuje u financijskim podacima je prilagođena zadnja cijena (engl. *adjusted close*). Prilagođena zadnja cijena odražava vrijednost dionice nakon obračuna korporativnih radnji, kao što su podjela dionica, dividende i ponude prava. Često se koristi u analizi povijesnog povrata kako bi se pružio točniji prikaz izvedbe dionice. Ključne prilagodbe:

- Prilagodba radi podjele dionica - Podjela dionica dijeli cijenu dionice tvrtke određenim faktorom kako bi dionice bile pristupačnije bez utjecaja na ukupnu tržišnu kapitalizaciju tvrtke. Na primjer, kod podjele dionica na tri dijela, dionica čija je cijena npr. 300 € prilagođava se na 100 €. Prošle cijene također su prilagođene kako bi se održala dosljednost.

- Prilagodba radi isplate dividendi - Dividende u novcu i dionicama smanjuju cijenu dionice, ali ne mijenjaju ukupni povrat investitora. Na primjer, ako je novčana dividenda od 1 € objavljena na dionici od 51 €, cijena se prilagođava na 50 USD. Prilagođene cijene na zatvaranju oduzimaju dividende kako bi se dobila jasnija slika povrata.

- Prilagodba za ponude prava (engl. *rights offering*) - Ponuda prava omogućuje dioničarima kupnju dodatnih dionica po sniženoj cijeni, smanjujući vrijednost postojećih dionica. Prilagođene zaključne cijene utječu na ove promjene kako bi odražavale utjecaj na vrijednost dioničara.

Prilagođena zadnja cijena najpouzdanija je i najinformativnija mjera za dugoročnu analizu, modeliranje i procjenu povijesnog trenda u znanosti o financijskim podacima. Njezina sposobnost da zabilježi ukupnu vrijednost ulaganja dok se obračunavaju korporativne akcije čini je superiornijom u odnosu na neobrađene OHLCV podatke za mnoge primjene u financijama te zbog toga koristi u ovom radu.
Skup podataka koji se koristi obuhvaća razdoblje od početka 2006. do kraja 2024. i uključuje dnevna promatranja za četiri ETF-a. Ovo razdoblje pokriva različite tržišne uvjete, pružajući čvrstu osnovu za testiranje modela.
Podaci su prikupljeni koristeći biblioteku *Yahoo Finance - yfinance* [4] u programskom jeziku Python te su podijeljeni na skup za treniranje, validaciju i testiranje. Skup za testiranje pokriva zadnje tri godine podataka (2022. - 2024.), skup za validaciju tri godine prije toga (2019. - 2021.), a skup za treniranje sve ostalo prije toga.

4.2. Sharpeov omjer

Za početak potrebno je definirati ciljnu funkciju (funkciju gubitka) što je u našem slučaju Sharpeov omjer. Sharpeov omjer je mjera u financijama koja pokazuje kako je ulaganje uspješno u odnosu na rizik. Što je veći omjer rizika ulaganja, to nudi veće povrate u odnosu na rizik. Sharpeov omjer pomaže nam odrediti je li rizik koji smo preuzeli generirao dovoljno visoke povrate u usporedbi s povratima koje biste mogli ostvariti bez preuzimanja rizika. Sharpeov omjer definiramo kao:

$$\text{Sharpeov omjer} = \frac{R_p - R_f}{\sigma_p} \tag{4.1}$$

gdje R_p predstavlja povrat portfelja na nekoj frekvenciji (minutni, dnevni, mjesečni, itd...), R_f je bezrizična stopa povrata koja predstavlja povrat ulaganja u bezrizičnu imovinu (npr. trezorski zapisi ili u SAD-u T-Bills), a σ_p je standardna devijacija portfelja.
Sharpeov omjer može se koristiti za procjenu performansi pojedinačne imovine ili portfelja prilagođenog riziku. Investitor može izračunati Sharpeov omjer koristeći povijesne ili očekivane povrate. Što je veći Sharpeov omjer portfelja, to je bolja njegova izvedba prilagođena riziku. Negativan Sharpeov omjer znači da je bezrizična ili referentna stopa veća od povijesnog ili predviđenog povrata portfelja ili se u suprotnom očekuje da će povrat portfelja biti negativan.

U našem slučaju ciljna funkcija Sharpeovog omjera bit će definirana kao:

$$L = \frac{\mathbb{E}[R_p]}{\text{Std}(R_p)} \tag{4.2}$$

gdje su $\mathbb{E}[R_p]$ i $\text{Std}(R_p)$ procijenjena srednja vrijednost i standardna devijacija povrata portfelja. Specifično, za period trgovanja od $t = 1, ..., T$, možemo maksimizirati sljedeću ciljnu funkciju:

$$L_T = \frac{\mathbb{E}[R_{p,t}]}{\sqrt{\mathbb{E}[R_{p,t}^2] - (\mathbb{E}[R_{p,t}])^2}} \tag{4.3}$$

gdje očekivanje povrata računamo na sljedeći način

$$E(R_{p,t}) = \frac{1}{T}\sum_{t=1}^{T}(R_{p,t}). \tag{4.4}$$

Ovdje $R_{p,t}$ predstavlja realizirane povrate portfelja n imovina tijekom vremena t koje računamo pomoću formule 2.2, a $r_{i,t}$ je povrat imovine i izračunat izrazom 2.1

Udio imovine i predstavljamo kao $w_{i,t} \in [0,1]$ te $\sum_i^n (w_{i,t}) = 1$. U ovome pristupu, neuronska mreža (LSTM) f s parametrima θ je primijenjena za modeliranje $w_{i,t}$ za dugoročni portfelj:

$$w_{i,t} = f(\theta | x_t) \tag{4.5}$$

gdje x_t predstavlja trenutačne informacije o tržištu, a klasični korak predviđanja se zaobilazi povezivanjem ulaza (povrata) s težinama (udjelima pojedine imovine) kako bi dobili povrat portfelja te naposljetku maksimizirali Sharpeov omjer tijekom perioda trgovanja T, odnosno L_T. Međutim, dugoročni portfelj nameće ograničenja koja zahtijevaju da težine budu pozitivne i da se zbrajaju u jedan; koristi se softmax kako bi se ispunili ovi zahtjevi:

$$w_i = \frac{\exp(\tilde{w}_i)}{\sum_{j=1}^n \exp(\tilde{w}_j)}. \tag{4.6}$$

4.3. Implementacija dubokog modela

Arhitektura modela nalazi se na slici 4.2. Model se sastoji od 3 glavna bloka: ulazni sloj, LSTM sloj te izlazni sloj. Ideja ovoga dizajna je korištenje neuronskih mreža za izdvajanje značajki ulaznih imovina. Značajke izdvojene iz modela dubokog učenja pokazale su bolje rezultate od tradicionalno "ručno rađenih" značajki. Kada su značajke izdvojene, model izbacuje težine portfelja te se dobivaju realizirani povrati kako bi se maksimizirao Sharpeov omjer.

U ulaznom sloju svaka imovina označava se s A_i te postoji ukupno n vrijednosnica koje formiraju portfelj (u našem slučaju $n = 4$). Jedan ulaz čine ulančani podaci svih imovina. Ulazne značajke jedne imovine su njezine prošle cijene i povrati dimenzije $(k, 2)$, gdje je k retrospektivni prostor koji u našem slučaju iznosi 50. Također, treniranje se odvija u mini-grupama veličine 64. Spajanjem značajki svih imovina, dimenzija rezultirajućeg ulaza je $(64, k, 2 \cdot n)$ odnosno $(64, 50, 8)$. Zatim se ovakvi ulazi unose u LSTM.

Implementacija LSTM modela napravljena je u programskom jeziku Python koristeći biblioteku `PyTorch`. Koristi se model s X skrivenih stanja veličine Y. U sloju LSTM-a odvija se unaprijedni prolaz kroz model kao što je opisano u prošlom odlomku.

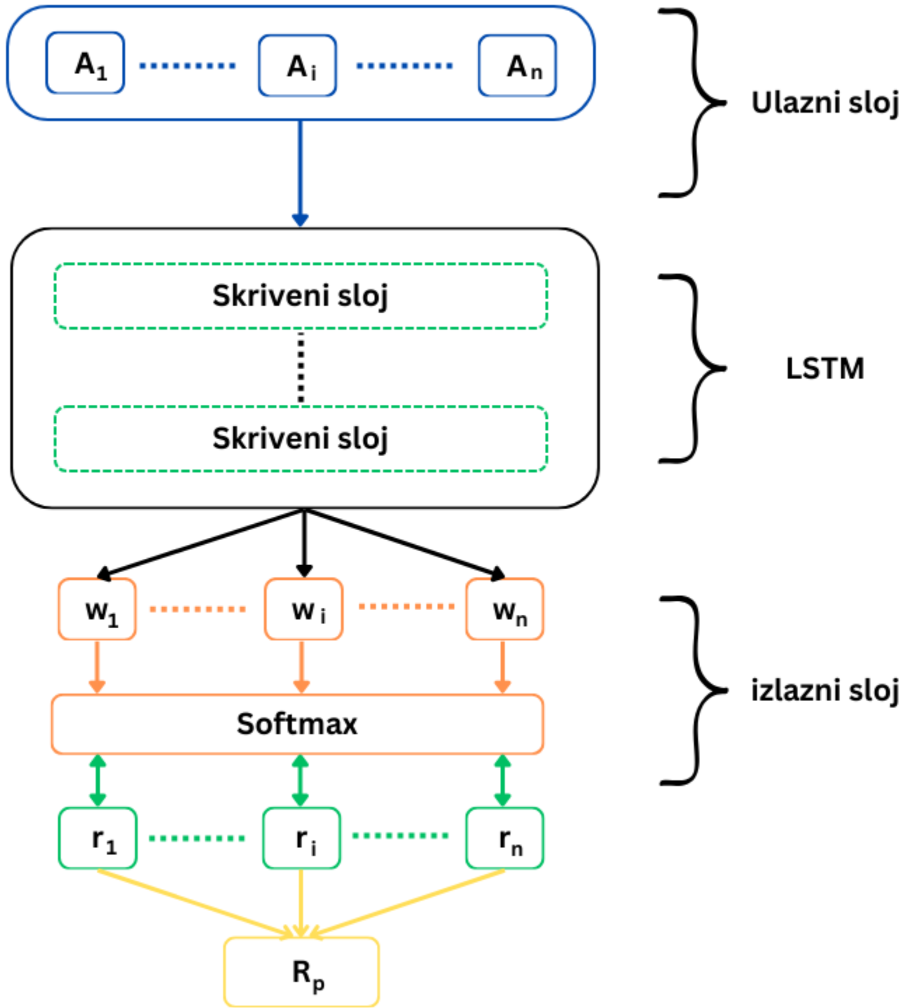
Izlaz LSTM-a daje težine $(w_1, ..., w_n)$ čiji je broj jednak broju imovina u portfelju. Na težine se primjenjuje softmax 4.6 zbog uvjeta da težine budu pozitivne te da se zbrajaju u jedan. Nakon toga se dobivene težine mogu množiti s povezanim povratima portfelja $(r_1, ..., r_n)$ kako bi izračunali realizirani povrat portfelja R_p. Kada je to dobiveno, može se izračunati Sharpeov omjer 4.2 Za optimizaciju parametara modela koristi se gradijentni uspon s optimizatorom Adam (sa stopom učenja $\eta = 0.001$) kako bi se maksimizirao Sharpeov omjer. U stvari će se u kodu koristiti gradijenti spust negativnog Sharpeovog omjera, što je zapravo ista stvar kao spomenuto.

Treniranje se odvija u 100 epoha. Tijekom svake epohe prođe se kroz cijeli skup za učenje te se model testira na validacijskom skupu i zabilježava se rezultat (Sharpeov omjer). Parametri najbolje epohe koriste se na kraju za testiranje modela.

Za treniranje i testiranje modela koristi se postupak zvan "Walk-forward backtesting" s pomičnim prozorom. Ova metoda često se koristi za optimizaciju i testiranje strategija trgovanja. Tradicionalni backtesting evaluira strategiju na fiksnom vremenskom periodu u prošlosti. Iako je takav pristup koristan, ima svoja ograničenja. Često dovodi do prenaučenosti, gdje strategija dobro radi u nekom prošlom vremenskom periodu, ali ne uspijeva u sadašnjosti. Walk-forward analiza zaobilazi taj problem tako što kontinuirano ažurira parametre pomičući se kroz podatke inkrementalno. Ovakav dinamičan pristup preciznije simulira stvarne uvjete na tržištu.

Postupak se koristi na sljedeći način. Prvo se odredi veličina pomičnog prozora (u ovom radu postavljen na 50) te veličina skupa za testiranje (u ovom radu iznosi 1). Drugim riječima, koristi se pedeset dana za učenje modela kako bi se dobile težine portfelja koje se koriste na sljedećem danu kombinirajući ih s povratima da bi se izračunao povrat portfelja. Zatim se na njemu računa funkcija gubitka, odnosno negativni Sharpeov omjer, te se odvija korak optimizacije (ažuriranje parametara modela). Zatim se pomični prozor pomiče za jedan dan te se proces ponavlja za sve podatke u skupu za učenje.

Testiranje se odvija koristeći isti postupak s istom dimenzijom ulaza (ista veličina minigrupe i retrospektivnog prozora) i s istim pomakom dok se ne prođe kroz cijeli skup za testiranje čime se dobiva povrat portfelja za svaki taj trenutak.



Slika 4.2. Arhitektura modela, slika napravljena po [5]

4.4. Implementacija metoda *mean-variance* i maksimalne diverzifikacije

Implementacije metoda *mean-variance* i maksimalne diverzifikacije napravljene su koristeći `PyPortfolioOpt` biblioteku u programskom jeziku Python. Obje metode koriste pomični prozor veličine 50. Na tom intervalu metoda *mean-variance* procjenjuje srednji očekivani povrat i uzoračku kovarijacijsku matricu. Pomoću njih se računa efikasna granica i rješava minimizacijski 2.5 (maksimizacijski 2.7) problem koristeći ograničenja 2.6 (odnosno 2.8 za maksimizaciju) uz dodatak ograničenja 2.1. koji ne dopušta kratke prodaje. Optimizacija se provodi kako bi se dobio portfelj s maksimalnim Sharpeovim

omjerom, odnosno težine takvog portfelja. Dobivene težine se množe sa povratima idućeg dana kako bi se izračunao ukupni povrat portfelja te se postupak ponavlja nakon pomaka za 1 (1 dan), dok se ne prođe kroz cijeli skup za testiranje (postupak ne zahtijeva treniranje jer se parametri direktno procjenjuju iz zadanog skupa za testiranje).
S druge strane metoda maksimalne diverzifikacije uzima povrate u vremenskom intervalu pomičnog prozora te računa i maksimizira omjer diverzifikacije. Dobivaju se težine, nakon čega je proces jednak kao i za *mean-variance* metodu.

5. Rezultati

5.1. Metrike

Osim Sharpeovog omjera kao glavne metrike koja se nastoji maksimizirati, koriste se i druge metrike za evaluaciju rezultata testiranja. One su očekivani povrat $E(R)$, standardna devijacija povrata $Std(R)$, negativna devijacija (engl. *downside deviation* - $DD(R)$), odnosno standardna devijacija negativnih povrata portfelja i Sortinov omjer. Spomenute mjere su anualizirane, odnosno predstavljaju godišnji prosjek pojedine mjere. Osim navedenih, koriste se i postotak pozitivnih povrata ($\% of + Ret$), omjer pozitivnih i negativnih povrata ($Ave.P/Ave.L$), te maksimalni propad (engl. *maximum drawdown* - MDD), koji predstavlja najveći pad vrijednosti portfelja od vrhunca do najniže točke.

5.1.1. Sortinov omjer

Sortinov omjer je varijacija Sharpeovog omjera koji razlikuje štetnu volatilnost od ukupne ukupne volatilnosti korištenjem negativne devijacije umjesto ukupne standardne devijacije povrata portfelja:

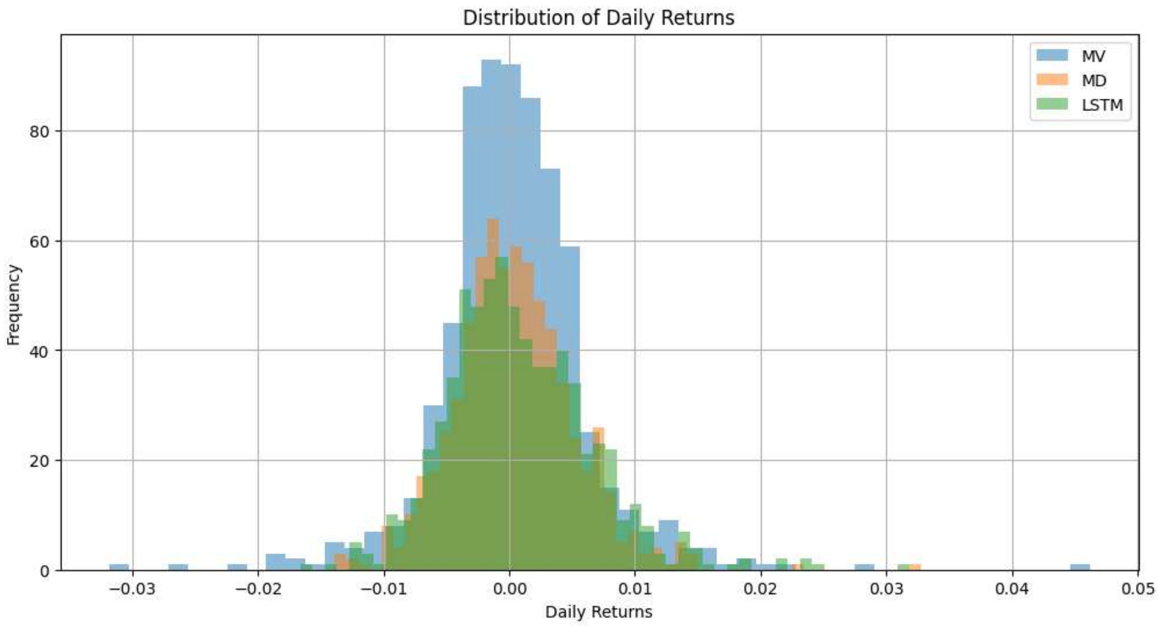
$$Sortino = \frac{R_p - r_f}{\sigma_d} \tag{5.1}$$

gdje R_p predstavlja povrat portfelja, r_f bezrizičnu kamatnu stopu, a σ_d negativnu devijaciju povrata portfelja.

Sortinov omjer razlikuje se od Sharpeovog omjera po tome što uzima u obzir samo standardnu devijaciju negativnog rizika, a ne cjelokupnog (pozitivnog + negativnog) rizika. Budući da se Sortino omjer usredotočuje samo na negativno odstupanje povrata portfelja od srednje vrijednosti, smatra se da daje bolji uvid u performanse portfelja prilagođene riziku budući da je pozitivna volatilnost prednost. Sortinov omjer koristan je način za ulagače, analitičare i upravitelje portfelja za procjenu povrata ulaganja za određenu razinu lošeg rizika.

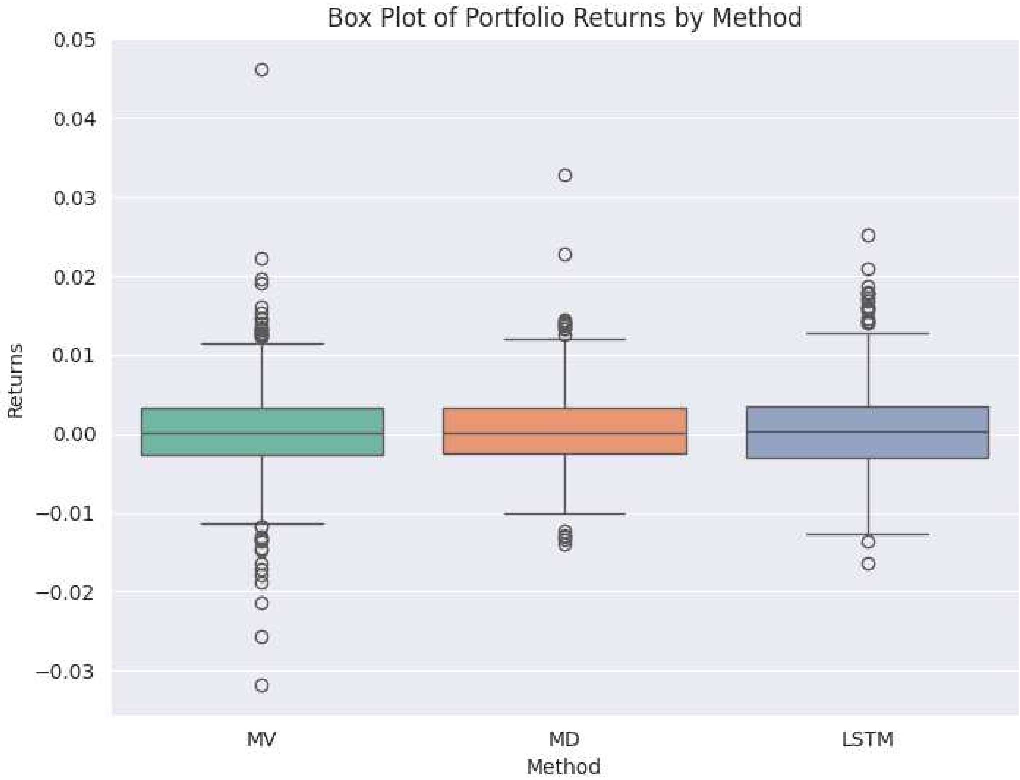
5.2. Usporedba rezultata

Slika 5.1. prikazuje distribucije povrata portfelja za svaku metodu. Sve distribucije se čine prilično normalnima s različitim razinama spljoštenosti i iskrivljenosti. Distribucija MV povrata je najviše koncentrirana oko nule što znači da ima najmanju volatilnost. LSTM portfelj ima malo širi raspon od druga dva dok je MD portfelj negdje između.



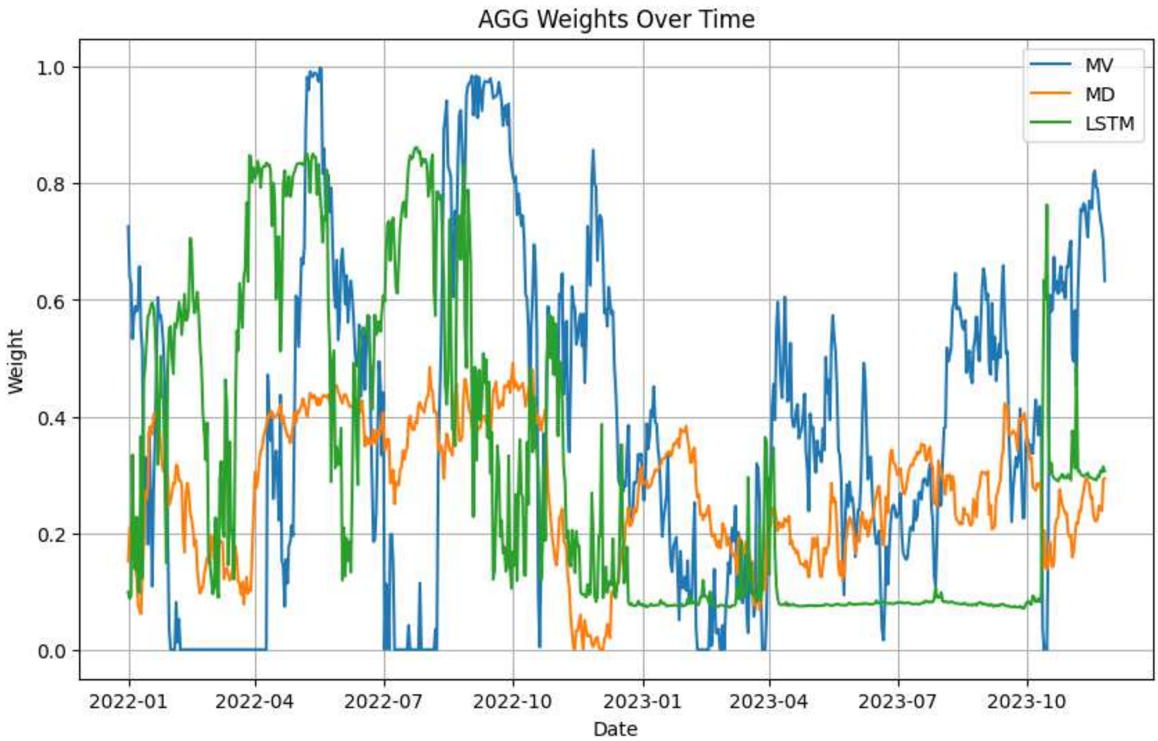
Slika 5.1. Distribucije dnevnih povrata portfelja za svaku metodu

Na slici 5.2. prikazan je *box plot* povrata za svaku metodu. Vidi se da LSTM model ima širi raspon što ukazuje na veću varijabilnost. Prisutnost stršećih vrijednosti, osobito u pozitivnom rasponu, može ukazivati na povremene velike dobitke, ali negativni stršeći podaci mogu signalizirati rizik (vidi se da LSTM ima najmanje stršećih vrijednosti). Može se primijetiti da je medijan povrata LSTM-a veći od druge dvije metode, što podupire njegovu superiornu izvedbu.

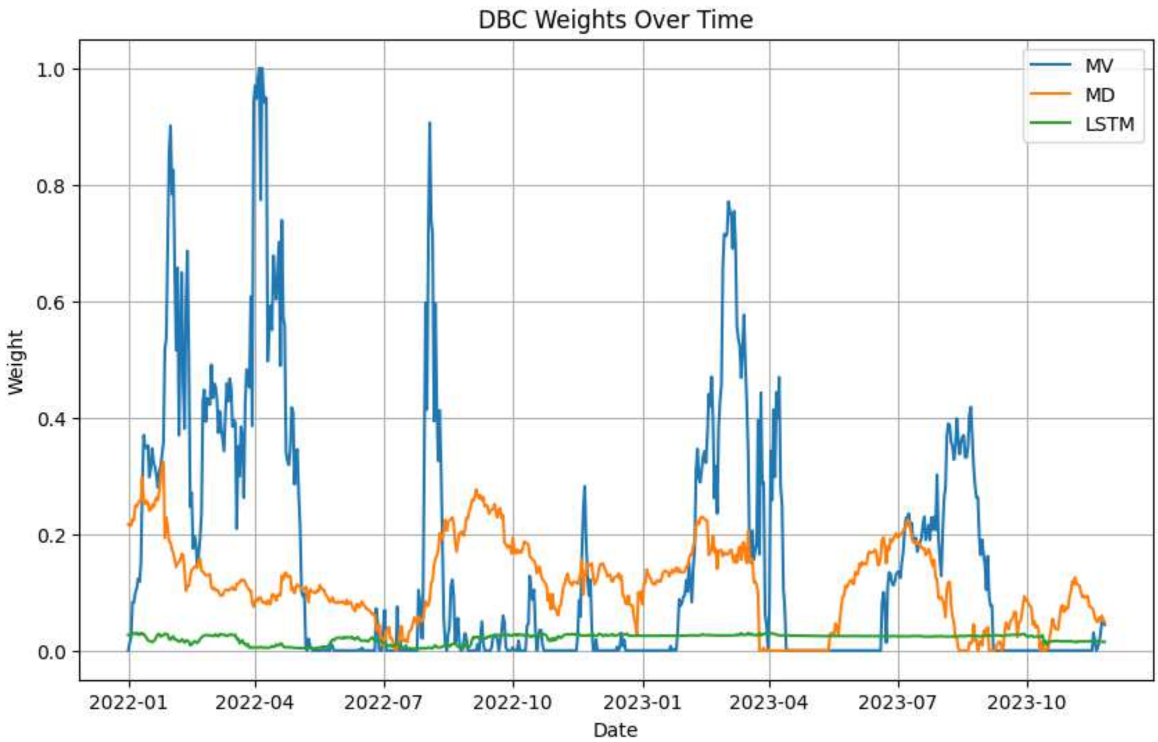


Slika 5.2. *Box plot* dnevnih povrata portfelja za svaku metodu

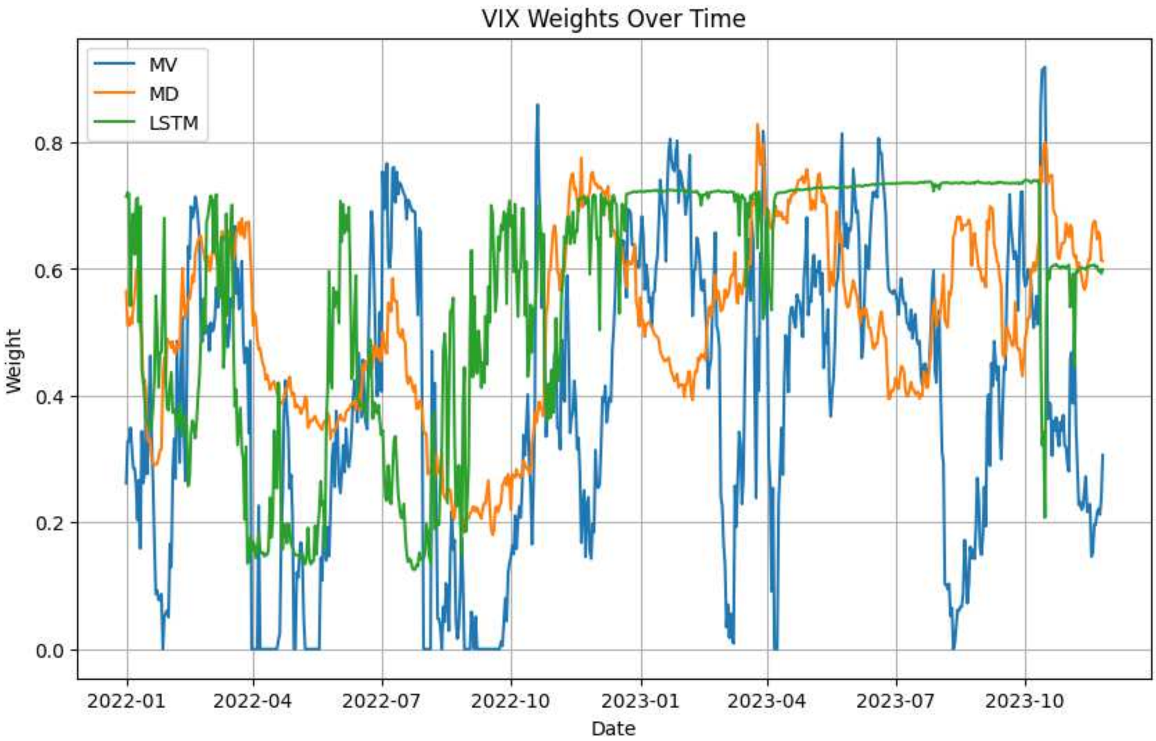
Na sljedećim slikama prikazane su težine svake imovine za pojedinu metodu na skupu za testiranje.



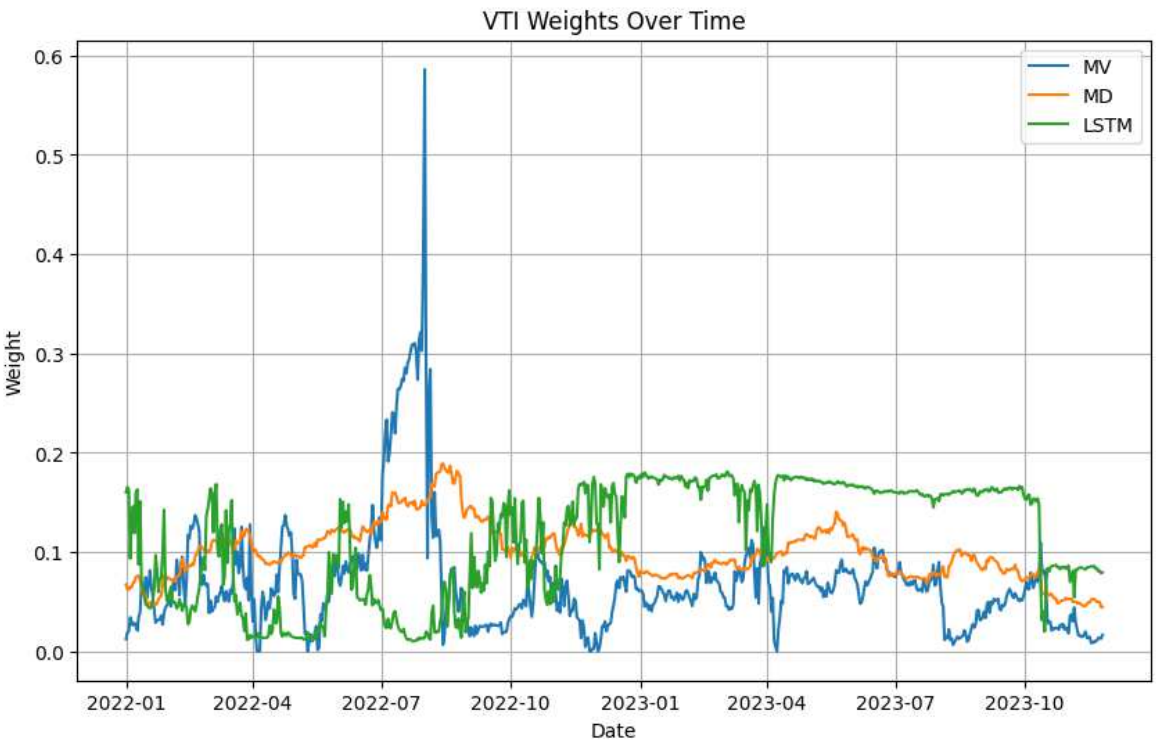
Slika 5.3. Težine AGG-a za svaku metodu



Slika 5.4. Težine DBC-a za svaku metodu

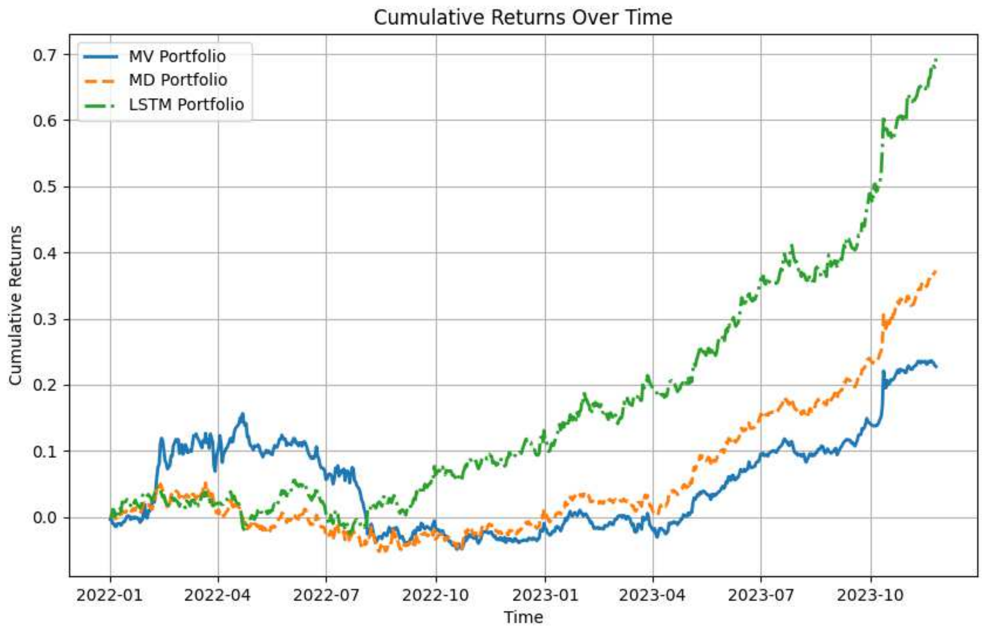


Slika 5.5. Težine VIX-a za svaku metodu



Slika 5.6. Težine VTI-a za svaku metodu

Slika 5.7. prikazuje kumulativne povrate kroz cijeli skup za testiranje. Može se vidjeti da LSTM model ostvaruje najbolje rezultate za dugoročni portfelj u ovom vremenskom razdoblju.



Slika 5.7. Kumulativni povrati portfelja

Tablica 5.1. prikazuje usporedbu spomenutih metrika za sve tri metode. LSTM pobjeđuje referentne metode MV i MD u većini metrika.

Tablica 5.1. Usporedba rezultata

	E(R)	**Std(R)**	**Sharpe**	**DD(R)**	**Sortino**	**MDD**	**% of + Ret**
MV	0.059	0.092	0.644	0.063	0.934	0.192	0.513
MD	0.119	**0.078**	1.528	**0.043**	2.780	**0.101**	**0.523**
LSTM	**0.150**	0.086	**1.737**	0.044	**3.392**	0.120	0.520

6. Zaključak

Ovaj rad istražuje primjene dubokog učenja, konkretno LSTM mreža, na problem optimizacije portfelja s ciljem maksimiziranja Sharpeovog omjera. Tradicionalne metode optimizacije portfelja, kao što su Markowitzeva *mean-variance* analiza i strategija maksimalne diverzifikacije, dugo su bile korištene za konstrukciju portfelja. Međutim, ove metode zasnivaju se na snažnim statističkim pretpostavkama, kao što su normalnost povrata i stabilne korelacije, koje često ne vrijede na stvarnim financijskim tržištima. Također, jako ovise o procjeni ulaznih parametara (očekivana vrijednost povrata i matrica kovarijance) za koje nije lagano dobiti kvalitetne procjene. Kao rezultat toga, alternativne metode koje koriste strojno učenje počele su dobivati sve više pozornosti.

U svrhu testiranja alternativnih metoda optimizacije portfelja, razvijen je LSTM model koji uči veze u ulaznim povratima imovina kako bi dobio optimalne težine istih unutar portfelja. Model je treniran na povijesnim podacima četiri ETF-a, te su njegovi rezultati testiranja uspoređeni s referentnim metodama *mean-variance* i maksimalnom diverzifikacijom. Rezultati pokazuju da se razvijeni model uspješno prilagodio tržišnim uvjetima, identificirajući složene odnose među imovinama, što je dovelo do poboljšanja performansi.

U procesu testiranja LSTM model uspio je postići Sharpeov omjer od 1.737, nadmašujući Markowitzrvu *mean-variance* metodu sa Sharpeovim omjerom 0.644 i metodu maksimalne diverzifikacije, koja je postigla Sharpeov omjer od 1.528. Ovaj rezultat ukazuje na to da metode dubokog učenja nude obećavajuće alternative za optimizaciju portfelja učinkovito identificirajući složene nelinearne obrasce i dinamične uvjete na tržištu.

Unatoč ovim obećavajućim rezultatima, postoji nekoliko ograničenja koja treba razmotriti. Izvedba modela uvelike ovisi o kvaliteti i kvantiteti ulaznih podataka, a prenaučenost i dalje može biti izazov. Dodatno, transakcijski troškovi na tržištu u stvarnom svijetu nisu eksplicitno modelirani u ovom istraživanju, što može utjecati na praktičnu primjenu.

Općenito, ovaj rad naglašava potencijal pristupa temeljenih na dubokom učenju u donošenju financijskih odluka, pokazujući da LSTM mreže mogu poslužiti kao moćan alat za optimizaciju portfelja.

Literatura

[1] H. Markowitz, "Portfolio selection", *Journal of Finance*, 1952. [Mrežno]. Adresa: https://doi.org/10.1111/j.1540-6261.1952.tb01525.x

[2] Y. Choueifaty i Y. Coignard, "Toward maximum diversification", *The Journal Of Portfolio Management*, sv. 35, 2008. [Mrežno]. Adresa: https://api.semanticscholar.org/CorpusID:49359548

[3] C. Olah, "Understanding lstm networks", https://colah.github.io/posts/2015-08-Understanding-LSTMs/, 2015., [mrežno; stranica posjećena: siječanj 2025.].

[4] "Yahoo finance", https://finance.yahoo.com/, [mrežno; stranica posjećena: studeni 2024.].

[5] Z. Zhang, S. Zohren, i S. Roberts, "Deep Learning for Portfolio Optimization", arXiv.org, Papers 2005.13665, svibanj 2020. [Mrežno]. Adresa: https://ideas.repec.org/p/arx/papers/2005.13665.html

[6] J. V. Burke, "Markowitz mean-variance portfolio theory", https://sites.math.washington.edu/~burke/crs/408/fin-proj/mark1.pdf, [mrežno; stranica posjećena: prosinac 2024.].

[7] M. Haugh, "Mean-variance optimization and the capm", https://www.columbia.edu/~mh2078/FoundationsFE/MeanVariance-CAPM.pdf, 2016., [mrežno; stranica posjećena: siječanj 2025.].

[8] Z. Bodie, A. Kane, i A. Marcus, *Investments*, ser. McGraw-Hill Education series in finance, insurance, and real estate. McGraw-Hill Education, 2018. [Mrežno]. Adresa: https://books.google.hr/books?id=3QlmvgAACAAJ

[9] D. Palomar, *Portfolio Optimization: Theory and Application*. Cambridge University Press, 2025. [Mrežno]. Adresa: https://books.google.hr/books?id=8uT20AEACAAJ

[10] I. Goodfellow, Y. Bengio, i A. Courville, *Deep Learning*, ser. Adaptive Computation and Machine Learning series. MIT Press, 2016. [Mrežno]. Adresa: https://books.google.hr/books?id=Np9SDQAAQBAJ

[11] Z. Kostanjčar i S. Begušić, “Vrednovanje imovine”, https://www.fer.unizg.hr/_download/repository/AF08%20-%20Vrednovanje%20imovine%20-%20drugi%20dio.pdf, 2024., prezentacija s predmeta Analitika financija na FER-u [mrežno; stranica posjećena: siječanj 2025.].

[12] Z. Kostanjčar i P. P. Šimović, “Capm”, www.fer.unizg.hr/_download/repository/Prezentacija_CAPM_(1)[2].pdf, 2024., prezentacija s predmeta Financijska matematika na FER-u [mrežno; stranica posjećena: prosinac 2024.].

[13] N. Lioudis, “Understanding the sharpe ratio”, https://www.investopedia.com/articles/07/sharpe_ratio.asp, 2024., [mrežno; stranica posjećena: prosinac 2024.].

[14] W. F. Sharpe, “The sharpe ratio”, 1994. [Mrežno]. Adresa: https://api.semanticscholar.org/CorpusID:55394403

[15] B. D. Bašić i M. Čupić i Jan Šnajder, “Umjetne neuronske mreže”, https://www.fer.unizg.hr/_download/repository/UI_12_UmjetneNeuronskeMreze[1].pdf, 2019., prezentacija s predmeta Uvod u umjetnu inteligenciju na FER-u [mrežno; stranica posjećena: siječanj 2025.].

[16] J. Holdsworth i M. Scapicchio, “What is deep learning?” https://www.ibm.com/think/topics/deep-learning, 2024., [mrežno; stranica posjećena: siječanj 2025.].

[17] C. Stryker, “What is a recurrent neural network?” https://www.ibm.com/think/topics/recurrent-neural-networks, 2024.

[18] GekseforGeeks, “What is lstm – long short term memory?” https://www.geeksforgeeks.org/deep-learning-introduction-to-long-short-term-memory/, 2024., [mrežno; stranica posjećena: siječanj 2025.].

[19] J. Šnajder, “Logistička regresija, skripta iz predmeta strojno učenje 1 na fer-u”, https://www.fer.unizg.hr/_download/repository/SU1-2022-P06-LogistickaRegresija.pdf, 2022., [mrežno; stranica posjećena: siječanj 2025.].

[20] M. Čupić, “Optimizacija parametara modela”, https://www.zemris.fer.hr/ ssegvic/du/du3optimization.pdf, 2019., prezentacijaspredmetaDubokouÄŊenje1naFER-u[mreÅ¿no;stranicaposjeÄĞena: sijeÄŊanj2025.].

[21] A. Ganti, “Adjusted closing price: How it works, types, pros cons”, https://www.investopedia.com/terms/a/adjusted_closing_price.asp, 2020., [mrežno; stranica posjećena: siječanj 2025.].

[22] P. News, “The future of backtesting: A deep dive into walk forward analysis”, https://www.pyquantnews.com/free-python-resources/the-future-of-backtesting-a-deep-dive-into-walk-forward-analysis#:~:text=Walk%20forward%20analysis%20is%20a,moving%20through%20the%20data%20incrementally., 2024., [mrežno; stranica posjećena: siječanj 2025.].

[23] T. N. Rollinger i S. T. Hoffman, “Sortino: A ‘sharper’ ratio”, https://www.cmegroup.com/education/files/rr-sortino-a-sharper-ratio.pdf, [mrežno; stranica posjećena: siječanj 2025.].

[24] Z. Kostanjčar i S. Begušić, “Optimizacija portfelja: problemi u praksi, alternativni pristupi i testiranje”, https://www.fer.unizg.hr/_download/repository/AF%2010%20Optimizacija%20portfelja%20u%20praksi,%20alternativni%20pristupi%20i%20testiranje.pdf, 2024., prezentacija s predmeta Analitika financija na FER-u [mrežno; stranica posjećena: siječanj 2025.].

[25] M. Jahn, “Downside deviation: Definition, uses, calculation example”, https://www.investopedia.com/terms/d/downside-deviation.asp, 2022., [Mrežno; stranica posjećena: siječanj 2025.].

[26] A. Gunjan i S. Bhattacharyya, “A brief review of portfolio optimization techniques”, *Artif. Intell. Rev.*, sv. 56, br. 5, str. 3847–3886, rujan 2022. https://doi.org/10.1007/s10462-022-10273-7

[27] J. Prigent, *Portfolio Optimization and Performance Analysis*, ser. Chapman and Hall/CRC Financial Mathematics Series. CRC Press, 2007. [Mrežno]. Adresa: https://books.google.hr/books?id=IT8IdJlPeKkC

[28] "Pyportfolioopt", https://pyportfolioopt.readthedocs.io/en/latest/index.html, [mrežno; stranica posjećena: studeni 2024.].

[29] "skfolio", https://skfolio.org/index.html, [mrežno; stranica posjećena: prosinac 2024.].

Sažetak

Optimizacija Sharpeovog omjera portfelja korištenjem modela dubokog učenja

Luka Lukačević

Ovaj rad proučava problem optimizacije portfelja te različite metode njegovog rješavanja, uključujući tradicionalne pristupe i metodu temeljenu na dubokom učenju. Najprije su predstavljene dvije temeljne strategije optimizacije portfelja: *mean-variance* optimizacija, temeljena na Markowitzevom okviru moderne teorije portfelja, i maksimalna diverzifikacija, koja ima za cilj konstruirati portfelj s maksimalnim omjerom diverzifikacije odabirom imovina koje najviše utječu na ukupno smanjenje rizika portfelja. Nakon detaljnog uvida u ove strategije, istražena je primjena dubokog učenja u optimizaciji portfelja, usredotočujući se na LSTM mreže. Model temeljen na LSTM-u uči iz povijesnih podataka kako bi ostvario optimalne udjele imovina unutar portfelja, s ciljem maksimizacije Sharpeovog omjera. Evaluacija modela na povijesnim tržišnim podacima pokazala je da pristup temeljen na dubokom učenju može pružiti konkurentne rezultate u odnosu na klasične metode, s potencijalom za bolju prilagodbu promjenjivim tržišnim uvjetima.

Ključne riječi: optimizacija portfelja; *mean-variance* analiza; maksimalna diverzifikacija; duboko učenje; LSTM

Abstract

Optimizing portfolio sharpe ratio using deep learning models

Luka Lukačević

This paper studies the portfolio optimization problem and various methods of solving it, including traditional approaches and a method based on deep learning. First, two fundamental portfolio optimization strategies are presented: mean-variance optimization, based on Markowitz's framework of modern portfolio theory, and maximum diversification, which aims to construct a portfolio with the highest diversification ratio by selecting assets that contribute the most to reducing overall portfolio risk. After a detailed examination of these strategies, the application of deep learning in portfolio optimization was explored, with a focus on LSTM networks. An LSTM-based model learns from historical data to determine optimal asset allocations within the portfolio, aiming to maximize the Sharpe ratio. The evaluation of the model on historical market data has shown that the deep learning-based approach can provide competitive results compared to classical methods, with the potential for better adaptability to changing market conditions.

Keywords: portfolio optimization; mean-variance analysis; maximum diversification; deep learning; LSTM