UNIVERZITET U TUZLI

EKONOMSKI FAKULTET

FINANSIJE, BANKARSTVO I OSIGURANJE

ZAVRŠNI RAD

Prvog ciklusa studija

Primjena mašinskog učenja u optimizaciji portfolija vrijednosnih papira

Mentor: **Student:**

Dr. sci. Jasmina Okičić, vanredni profesor **Emir Smajlović, 3-179/I-15**

Tuzla, septembar, 2021. godine

Mentor rada: dr. sci. Jasmina Okičić, vanredni profesor

Rad sadrži: 34 stranice

Redni broj završnog rada _____

Sažetak

Predmet istraživanja u radu je primjena metoda mašinskog učenja u optimizaciji portfolija vrijednosnih papira.

Cilj istraživanja je analiza i komparacija standardnih metoda i metoda koje koriste mašinsko učenje, kako bi se pokušali prevazići nedostaci standardnih metoda.

Istraživanje je potvrdilo da u određenim situacijama, modeli mašinskog učenja mogu prevazići te nedostatke, ali uz određena ograničenja.

Rad, pored uvoda i zaključka, sadrži tri dijela. U prvom dijelu opisuju se standardne metode moderne portfolio teorije, osnove neuralnih mreža, osnove reinforsiranog učenja, osnove Q-učenja. Drugi dio sadrži metodologiju koja će se primjeniti na skup podataka, i opisuje se metoda dubokog Q-učenja. Treći dio sadrži rezultate dobivene primjenom obje metode.

Ključne riječi: optimizacija portfolija, mašinsko učenje,

Sadržaj

Uvod

Znatan razvoj u polju mašinskog učenja dovodi do zanimljivog pitanja koliko je ono primjenljivo u polju optimizacije portfolija. Metode koje su od ranije poznate, imaju razne pretpostavke ili ograničenja koja bi se mogla prevazići upravo korištenjem tih metoda.

Predmet istraživanja ovog rada odnosi se na primjenu metoda mašinskog učenja u optimizaciji portfolija vrijednosnih papira, konkretnije primjene metoda reinforsiranog učenja.

Opšti cilj istraživanja odnosi se na to da se teorijski objasne „klasične“ metode optimizacije portfolija vrijednosnih papira, tj. Markowitz-eva metoda, fundamenti teorije neuralnih mreža, reinforsiranog učenja, Q-učenja i dubokog Q-učenja.

Da bi opšti cilj bio ispunjen neophodno je realizovati nekoliko operativnih ciljeva:

1. opisati način optimizacije Markowitz-evom metodom, zajedno sa nedostacima te metode
2. opisati generalno osnove neuralnih mreža, osnove reinforsiranog učenja te određene metode koje će se koristiti u svrhu istraživanja, odnosno Q-učenje, i duboko Q-učenje.
3. primjeniti na skupu cijena vrijednosnih papira obje metode, a onda potom, izvršiti usporedbu rezultate primjene obje metode.

Za provođenje planiranog istraživanja koristiće se prije svega, saznanja iz oblasti optimizacije portfolija, neuralnih mreža i reinforsiranog učenja, te knjige, članci i internet izvori koji se bave istim ili sličnim temama.

U okviru određenih teorijskih razmatranja koristiti će se metoda dedukcije, analize, sinteze, i komparacije. Pri optimizaciji portfolija vrijednosnih papira koristiće se metode mašinskog učenja.

Rad je koncipiran iz 3 dijela. Prvi dio se odnosi na uvod u optimizaciju portfolija vrijednosnih papira, gdje ćemo definisati osnovne formule potrebne da bi se izvele ideje potrebne za optimizaciju Markowitz-evom metodom, definisaćemo način na koji neuralne mreže funkcionišu, zašto su korisne, onda ćemo opisati način na koji reinforsirano učenje funkcioniše kao podvrstu mašinskog učenja. Drugi dio odnosi se na metodologiju koja će se koristiti da bi se dobili rezultati, u njoj je malo detaljnije objašnjeno duboko Q-učenje, te način na koji se neuralne mreže koriste unutar te metode. Treći dio odnosi se na predstavljanje rezultata koji su dobiveni koristeći navedene metode.

1. Uvod u optimizaciju portfolija vrijednosnih papira

Počeci optimizacije portfolija vrijednosnih papira mogu se vezati za rani period razvoja finansijskih tržišta. Investitorima je bio potreban siguran alat, zasnovan na relativno novim poljima vjerovatnoće i statistike, koji bi im služio kao podrška za upravljanje njihovim portfolijima. Do tada općeprihvaćena praksa jeste bila trgovina na osnovu „osjećaja". Značajan razvoj finansijskih tržišta implicira mnoštvo stvari, ali ono što je bitno za svrhe ovog rada jeste povećan broj dionica koje kotiraju na berzi, te povećana likvidnost tržišta, što znači da trgovanje na osnovu „osjećaja" više nije moglo biti efikasno. Neophodno je bilo da neko sistematski pristupi analizi i sintezi tržišnih varijabli kako bi se moglo optimalno upravljati vrijednosnim papirima.

Pionir ovih metoda, čije je nasljeđe savremeno shvatanje investicijskih strategija, Harry Markowitz je bio među prvima koji je uspješno formulisao i definisao osnovne pretpostavke optimizacije portfolija. Njemu se zajedno uz Black-Scholes-Mertona pripisuje razvoj moderne portfolio teorije (Kozarević, 2009, p. 11).

Markowitz svoj rad počinje sa onim što se danas smatra generalnim znanjem. Njegov rad otpočinje sa opisom nesigurnosti vrijednosnih papira. Ističe da je zbog asimetrije u podacima nemoguće da bilo koji analitičar sa 100% sigurnošću da procjenu o kretanju nekog vrijednosnog papira. Ističe da je bitno u poređenju vrijednosnog papira A i vrijednosnog papira B, uporediti sa kojim stepenom sigurnosti se može donijeti procjena o promjenama u cijeni vrijednosnog papira.

Nakon toga, ističe korelaciju, odnosno povezanost između vrijednosnih papira[1], kao varijablu koja bi se također trebala uzimati u obzir pri selekciji optimalnog portfolija vrijednosnih papira. Naime kao i sve ekonomske veličine, dva vrijednosna papira mogu imati direktnu ili inverznu povezanost. Njegov dokaz u to vrijeme, da je bitno posmatrati korelaciju, se svodi na sljedeću ideju: „Da vrijednosni papiri nisu povezani, diverzifikacija bi mogla eliminirati[2] sve rizike. Bilo bi kao bacanje mnogo novčića, mogli bi biti gotovo pa sigurni da će pola njih pasti na glavu".

U osnovi da korelacija među vrijednosnim papirima ne postoji, neovisno od svih varijabli očekivali bismo da ½ vrijednosti svih vrijednosnih papira poraste, a druga polovina da padne, u nekom datom trenutku.

Na kraju navodi odnos između rizika i prinosa, kao jednu od determinanti pri selekciji portfolija vrijednosnih papira. Navodi dva glavna cilja investitora:

1. Žele da prinos bude visok,

2. Žele da taj prinos bude siguran.

Odnosno da prosječni prinos ima veoma mala odstupanja, što predstavlja izvor sigurnosti.

[1] Korelacija ρ predstavlja mjeru povezanosti između dvije varijable, detaljnije će se pisati o njoj u nastavku rada.
[2] Diverzifikacija predstavlja smanjenje ukupnog rizika portfolija povećavanjem broja vrijednosnih papira. Npr. ako imamo 20 dionica u portfoliju i 10 njih izgubi svu svoju vrijednost izgubili smo ½ portfolija, dok ako držimo 200 dionica i 10 njih izgubi svoju vrijednost, izgubili smo samo 1/10 uloženog novca.

1.1. Elementarni pojmovi

Sve moderne metode, kao i metode koje su cilj ovog rada zasnovane su na Markowitzevoj metodi. Da bismo mogli opširnije pisati o njoj, definisat ćemo elementarne pojmove potrebne za dublje razumijevanje njegovog modela.

Za svaku vrijednost definišemo povrat, na jedan od 2 načina:

$$R_t = \frac{S_t}{S_{t-1}} - 1$$

, ili kao

$$R_t = \ln\left(\frac{S_t}{S_{t-1}}\right)$$

Gdje je S cijena vrijednosnog papira u vremenu t.

Druga definicija prinosa, ima određena matematička i statistička svojstva koja će nam biti korisna u daljoj razradi teme, te ako drugačije nije naznačeno, će se koristiti.

Kao što je u uvodu navedeno dvije veličine koje nas zanimaju jeste prosječni prinos, i njegova volatilnost, odnosno rizičnost.

Prosječna vrijednost se računa kao:

$$\mu = \frac{\sum_{i=1}^{n} x_i}{n}$$

Gdje je x_i oznaka za vrijednost opservacije, a n broj opservacija. Prosječni prinos $\bar{R}$ se dobije, kada je $x_i = R_i$, dok n predstavlja broj vrijednosnih papira u portfoliju. Odavdje možemo izvesti još jedan zaključak. Ako u skupu od n vrijednosnih papira, imamo n_1 vrijednosnog papira R_1, n_2 vrijednosnog papira R_2 itd. onda množimo povrate sa ponderima $p(S_i) = \frac{n_i}{n}$. Po definiciji ovo predstavlja relativnu frekvenciju javljanja, odnosno vjerovatnoću vrijednosnog papira R_i. To nam govori da možemo definisati prosječni povrat, kao očekivanu vrijednost, odnosno

$$E(\bar{R}) = \sum_{i=1}^{n} S_i \, p(S_i)$$

Generalno ovo se odnosi na bilo koju slučajnu varijablu koja ima mjerljivu distribuciju vjerovatnoće. Generalno za bilo koju slučajnu varijablu X vrijedi da

$$\mu = E(X) = \sum_{i=1}^{n} x_i \, p(x_i)$$

Gdje je $p(x_i)$ distribucija vjerovatnoće za x, odnosno odgovarajuća funkcija mase vjerovatnoće, u tačkama gdje postoji podrška za tu vrijednost.

Odavdje dalje možemo definisati varijansu. Varijansa je prosječno kvadratno odstupanje od aritmetičke sredine i data je u sljedećem obliku

$$\sigma^2 = Var(X) = \frac{\sum_{i=1}^{n}(x_i - \mu)^2}{n} = \sum_{i=1}^{n}(x_i - \mu)^2 p(x_i)$$

Standardna devijacija se definiše kao korijen varijanse shodno tome imamo

$$\sigma = \sqrt{\frac{\sum_{i=1}^{n}(x_i - \mu)^2}{n}} = \sqrt{\sum_{i=1}^{n}(x_i - \mu)^2 p(x_i)}$$

Nakon toga možemo definisati kovarijansu dvije varijable X i Y kao

$$Cov(X,Y) = \frac{1}{n}\sum_{i=1}^{n}(x_i - E(X))(y_i - E(Y))$$

Odavdje je očito da $Cov(X,X) = \sigma^2$, dok se korelacija između dvije varijable dobije kada $Cov(X,Y)$ pomnožimo sa $\frac{1}{\sigma_1\sigma_2}$. Za svrhe objašnjenja Markowitzeve teorije potrebno je još opisati i matricu varijanse-kovarijanse.

Neka je $\boldsymbol{X}$ n-dimenzionalni vektor, $\boldsymbol{X} = [X_1, X_2, X_3, \dots, X_n]^T$, za njega vrijedi sljedeće $E(\boldsymbol{X}) = [E(X_1), E(X_2), E(X_3), \dots, E(X_n)]^T$. Sada možemo pretpostaviti da postoji matrica $\mathbf{W}$ koja je faktički $mn * 1$ vektor shodno tome njeno očekivanje će biti $E(\boldsymbol{W}) = E[W_{ij}]$. Ovo nam omogućava da definišemo matricu varijanse-kovarijanse.

Neka je $\boldsymbol{X}$ kao i ranije, i neka je $\sigma_i^2 = Var(X_i)$. Prosjek vektora $\boldsymbol{X}$ je po definiciji $\mu = E(\boldsymbol{X})$, a matrica varijanse-kovarijanse po definiciji je

$$Cov(\boldsymbol{X}) = E[(\mathbf{X} - \boldsymbol{\mu})(\mathbf{X} - \boldsymbol{\mu})^T] = [\sigma_{ij}]$$

1.2 Formulacija Markowitzevog problema optimizacije portfolija

Sada kada imamo definisane varijable možemo predstaviti i model na kojim će se kasnija poglavlja zasnivati.

Kao što je ranije navedeno, ono što je od interesa investitoru su prosječni prinos, te njegova rizičnost. Racionalni investitor će nastojati na osnovu njegove lične sklonosti prema riziku da maksimizira prinos uz minimiziranje rizičnosti.

Radi primjera pretpostavit ćemo da u svom portfoliju investitor drži dvije dionice. Udio neke dionice u broju ukupnih dionica nekog portfolija ćemo označiti sa ω_i. Neka je prosječni relativni prinos prve dionice 0,15, druge dionice 0,04, te neka je standardna devijacija prve dionice 1, dok je standardna devijacija druge 0,30. Očekivani prinos na portfolio će iznositi $E[P] = \omega_1 * 0{,}15 + \omega_2 * 0{,}04$.

Generalno prema formuli 1.1.5, očekivana vrijednost portfolija iznosi $E(P) = \sum_{i=1}^{n} \omega_i R_i$. Volatilnost portfolija se i dalje izražava također putem kvadratnog korijena varijanse, ali se računa nešto drugačije. Prema formuli 1.1.6, imamo da $Var(X) = \sum_{i=1}^{n}(x_i - \mu)^2 p(x_i)$. Odakle možemo zaključiti da je $\sigma_p^2 = Var(P) = \sum_{i=1}^{n}(R_i - E(P))^2 \omega_i$. Radi pojašnjenja pretpostavimo da imamo portfolio od 2 vrijednosna papira R_1 i R_2. U tom slučaju $\sigma_p^2 = E[\omega_1 R_1 + \omega_2 R_2 - (\omega_1 \bar{R}_1 + \omega_2 \bar{R}_2)]^2 = E[\omega_1 (R_1 - \bar{R}_1) + \omega_2 (R_2 - \bar{R}_2)]^2$, onda imamo $\sigma_p^2 = E[\omega_1^2 (R_1 - \bar{R}_1)^2 + \omega_2^2 (R_2 - \bar{R}_2)^2 + 2\omega_1 (R_1 - \bar{R}_1)\omega_2 (R_2 - \bar{R}_2)]$, uzimajući u obzir da je očekivana vrijednost linearni operator, dolazimo na kraju do sljedećeg izraza: $\sigma_p^2 = \omega_1^2 E[(R_1 - \bar{R}_1)^2] + \omega_2^2 E[(R_2 - \bar{R}_2)^2] + 2\omega_1\omega_2 E[(R_1 - \bar{R}_1)\omega_2 (R_2 - \bar{R}_2)] = \omega_1^2\sigma_1^2 + \omega_2^2\sigma_2^2 + 2\omega_1\omega_2 Cov(R_1, R_2)$.

Odavdje možemo doći do zaključka da za portfolio sa n vrijednosnih papira varijansa se računa prema sljedećoj formuli

$$\sigma_p^2 = \sum_{i=1}^{n} \omega_i^2 \sigma_i^2 + \sum_{i=1}^{n}\sum_{j=1}^{n} \omega_i \omega_j Cov(r_i, r_j)$$

Obzirom da se rizik portfolija definiše kao njegova standardna devijacija, imamo sljedeće

$$\sigma_p = \sqrt{\sum_{i=1}^{n} \omega_i^2 \sigma_i^2 + \sum_{i=1}^{n}\sum_{j=1}^{n} \omega_i \omega_j Cov(r_i, r_j)}$$

Što nam govori da se rizičnost portfolija povećava ukoliko postoji povezanost između povrata vrijednosnih papira. Vraćajući se na pojam diverzifikacije iz prvog poglavlja, rekli smo da se s njom smanjuje rizičnost portfolija. Ukoliko je diverzifikacija potpuna $Cov(r_i, r_j) =$

$Corr(r_i, r_j) = 0$, što svodi formulu za rizik na $\sigma_p = \sqrt{\sum_{i=1}^{n} \omega_i^2 \sigma_i^2}$ što je u skladu sa Markowitzevom teorijom, jer je rizik manji.

Međutim, obzirom da rijetko koji investitor ima samo 2 vrijednosna papira u svom portfoliju, često se predstavlja u matričnom obliku, što ćemo i predstaviti kako bismo upotpunili teoriju.

Pretpostavimo da investitor u portfoliju ima n- vrijednosnih papira i da ih drži u nekom periodu T. Cijena vrijednosnog papira u trenutku 0 iznosiće S_0, dok na kraju perioda će iznositi S_1, njegova relativna promjena u vrijednosti će iznositi $\mathrm{P}_1 = \ln\left(\frac{S_1}{S_0}\right) = \ln(S_1) - \ln(S_0)$, odnosno kao prema formuli 1.1.2 imamo da $P_i = \ln(P_i) - \ln(P_{i-1})$. Neka x_i označava broj neke dionice koju investitor ima u svom portfoliju, onda x_i može imati sljedeće vrijednosti

$$\omega_i = \begin{cases} investitor\ zauzima\ dugu\ poziciju,\ \omega_i > 0 \\ investitor\ zauzima\ kratku\ poziciju,\ \omega_i < 0 \end{cases}$$

Za investitora kažemo da je zauzeo dugu poziciju, ako prodaje svoje vrijednosne papire, odnosno ako očekuje da će cijena vrijednosnog papira porasti u budućnosti i da će time ostvariti prihod.

Suprotno od duge pozicije je kratka pozicija, i ona se odnosi na situaciju gdje investitor prodaje „pozajmljene vrijednosne papire“, te će ih ponovo kupiti u nekoj tački vremena u budućnosti. Investitor se u toj situaciji nada da će cijena vrijednosnih papira opasti, kako bi pri ponovnoj kupovini vrijednosnih papira pri nižoj cijeni ostvario prihod.

Obzirom da je P_i povrat na dionicu, ukupni prinos iznosi $R_i = \sum_{i=1}^{n} P_i \omega_i$, odavdje možemo definisati vektor $\boldsymbol{\omega}$ čiji su elementi količina vrijednosnih papira, i neka je $\mathbf{P}$ vektor čiji su elementi relativni povrati na vrijednosne papire, onda imamo

$$\boldsymbol{\omega} = \begin{bmatrix} \omega_1 \\ \omega_2 \\ \vdots \\ \omega_n \end{bmatrix}, \boldsymbol{P} = \begin{bmatrix} p_1 \\ p_2 \\ \vdots \\ p_n \end{bmatrix}$$

Onda u matričnom obliku imamo da $\boldsymbol{R} = \boldsymbol{P^T \omega} = \boldsymbol{\omega^T P}$. Obzirom da moramo procjeniti prosječnu vrijednost potrebno je da definišemo nasumični vektor $\overline{\boldsymbol{P}}$, i njega definišemo kao $E[\boldsymbol{P}] = \overline{\boldsymbol{P}}$. Odavdje proizlazi pitanje koji je prosječni povrat odnosno $\boldsymbol{R}$. Obzirom da $\boldsymbol{R} = \boldsymbol{P}^T \boldsymbol{\omega} = \boldsymbol{\omega}^T \boldsymbol{P}$, imamo da $\overline{\boldsymbol{R}} = E[\boldsymbol{R}] = E[\boldsymbol{\omega^T P}]$ obzirom da je očekivana vrijednost konstante sama konstanta imamo da

$E[\boldsymbol{R}] = E[\boldsymbol{\omega^T P}] = \boldsymbol{\omega^T} E[\boldsymbol{P}] = \boldsymbol{\omega^T} \overline{\boldsymbol{P}}$.

Ova definicija nam je potrebna da bismo mogli izvesti volatilnost portfolija, što slijedi u nastavku.

Matrica varijanse-kovarijanse na povrate se u literaturi obično označava sa $\boldsymbol{\Sigma}$.[3] Ona se definiše prema formuli 1.1.9 kao $\boldsymbol{\Sigma} = E[(\mathbf{P} - \overline{\boldsymbol{P}})(\mathbf{P} - \overline{\boldsymbol{P}})^T]$.

Međutim, nas zanima varijansa povrata, i nju ćemo izvesti u nastavku, neka je $\boldsymbol{\sigma_r^2} = E[(\boldsymbol{R} - \overline{\boldsymbol{R}})^2]$ onda imamo da

$$\boldsymbol{\sigma_r^2} = E[(\boldsymbol{R} - \overline{\boldsymbol{R}})^2] =$$
$$E[(\boldsymbol{P}^T\boldsymbol{\omega} - \overline{\boldsymbol{P}}^T\boldsymbol{\omega})^2] =$$
$$E[((\boldsymbol{P} - \overline{\boldsymbol{P}})^T\boldsymbol{\omega})^2] =$$
$$E[((\boldsymbol{P} - \overline{\boldsymbol{P}})^T\boldsymbol{\omega})^T * ((\boldsymbol{P} - \overline{\boldsymbol{P}})^T\boldsymbol{\omega})] =$$
$$E[\boldsymbol{\omega}^T(\boldsymbol{P} - \overline{\boldsymbol{P}})(\boldsymbol{P} - \overline{\boldsymbol{P}})^T\boldsymbol{\omega}]$$

Što nas dovodi do

$$E[\boldsymbol{\omega}^T(\boldsymbol{P} - \overline{\boldsymbol{P}})(\boldsymbol{P} - \overline{\boldsymbol{P}})^T\boldsymbol{\omega}] =$$
$$\boldsymbol{\omega}^T E[(\boldsymbol{P} - \overline{\boldsymbol{P}})(\boldsymbol{P} - \overline{\boldsymbol{P}})^T]\boldsymbol{\omega}$$
$$\boldsymbol{\omega}^T\boldsymbol{\Sigma}\boldsymbol{\omega} = \boldsymbol{\sigma_r^2}$$

Time smo odredili varijansu povrata na prinose portfolija. Ovo predstavlja ciljnu funkciju u svim problemima optimizacije portfolija. U širem smislu ona predstavlja mjeru rizika. Postoje i drugi načini za mjerenje rizika, ali više o tome će se pisati u nastavku rada.

Ove formule predstavljaju suštinu moderne portfolio teorije i samim time suštinu misli iznesenih u poglavlju 1. Ako nastavimo u duhu tih misli, naveli smo da investitore zanima visok i relativno stabilan prinos.

Tačna visina prinosa određena je samom funkcijom korisnosti. Funkcija korisnosti određuje stepen investitorove averzije prema riziku. Iz tog razloga formula $\boldsymbol{\sigma_r^2}$ se nekada zapisuje i na sljedeći način

$$\boldsymbol{\sigma_r^2} = \boldsymbol{\omega}^T\boldsymbol{\Sigma}\boldsymbol{\omega} - q * \boldsymbol{R}^T\boldsymbol{\omega}$$

Gdje q predstavlja investitorov nivo tolerancije. Važno je još navesti da postoje sljedeće osnovne funkcije korisnosti: linearna, eksponencijalna i logaritamska korisnost.

[3] Prema ustanovljenoj notaciji, boldirane oznake predstavljaju matrice ili vektore, u ovom slučaju $\boldsymbol{\Sigma}$ označava matricu varijanse- kovarijanse, da ne bi došlo do zabune.

1.3. Efikasna granica

Na osnovu prethodnog vidjeli smo parametre koji određuju korisnost određenog portfolija nekom investitoru. Shodno tome postavlja se pitanje, koja je kombinacija parametara optimalna. Odnosno koja kombinacija vrijednosnih papira pruža najviši i najsigurniji prinos.

Krenut ćemo od jednostavnijeg primjera, kako bi generalizirali ka većem. Pretpostavit ćemo kao i u prethodnom poglavlju dvije dionice sa prosječnim prinosima $\mu_1 = E[R_1] = 0{,}15$ i $\mu_2 = E[R_2] = 0{,}04$. Također, pretpostavit ćemo iste standardne devijacije $\sigma_1 = 1$, te $\sigma_2 = 0{,}30$. Sa pretpostavkom da su relativni prinosi izračunati prema formuli 1.1.1. Zadnja pretpostavka koju ćemo napraviti jeste da su prinosi normalno distribuirani oko prosjeka, odnosno:

$$R_i \sim N(\mu_i, \sigma_i)$$

U slučaju da je investitor indiferentan u svojim sklonostima prema bilo kojoj od dvije dionice imat će jednake proporcije ω_i dionica, obzirom da imamo uslov da $\sum_{i=1}^{n} \omega_i = 1$ imamo da $\omega_1 = \omega_2 = 0{,}5$.

Ovaj slučaj nam daje situaciju da je očekivani relativni prinos portfolija jednak $E[P] = 0{,}5 * (0{,}15 + 0{,}04) = 0{,}095$. Da bi opširnije opisali situaciju iskoristit ćemo programski jezik R i simulirat ćemo 1000 nasumično generisanih relativnih prinosa sa pomenutom distribucijom za obje dionice. Tako ćemo dobiti sljedeću tabelu koja je skraćena radi kompaktnosti.

Tabela 1.3.1. Simulirani relativni prinosi za hipotetičke dionice

Dionica 1	***Dionica 2***
1.434905	0.008807
1.094779	0.069888
1.122227	-0.03442
1.066542	0.219337
0.166962	-0.31444
-0.68857	0.043414
1.225198	0.024802
-0.18843	-0.59192
1.259588	0.542457
0.201269	0.236868
-0.34605	0.441034
-0.3953	0.351577
-0.27216	0.230704
-0.01424	-0.53251
1.445922	0.187671
-1.53543	-0.54816

Izvor podataka: generisano putem rnorm komande

Puna tabela sa svim podacima o relativnim prinosima se može naći u prilogu. Prosječni prinos za Dionicu 1 iznosi $\mu_{sim1} = 0.1941409$, a std.dev $\sigma_{sim1} = 0.9775077$, za Dionicu 2 iznosi su $\mu_{sim2} = 0.03879672$, a std.dev $\sigma_{sim2} = 0.3087766$.

Izračunat ćemo rizičnost obje dionice putem formule σ_p^2 za slučaj dva vrijednosna papira. Izračunata kovarijansa iznosi $Cov(R_{sim1}, R_{sim2}) = -0.01042472$, dok korelacija iznosi $Corr(R_{sim1}, R_{sim2}) = -0.03453822$.

Ako ponovo uzmemo slučaj indiferentnosti varijansa takvog portfolija će iznositi $\sigma_{simp}^2 = \omega_1^2\sigma_1^2 + \omega_2^2\sigma_2^2 + 2\omega_1\omega_2 Cov(R_{sim1}, R_{sim2}) = 0{,}25 * (0{,}9775077 + 0{,}3087766) + 2 * 0{,}25 * -0{,}01042472 = 0{,}391358715$. Ova mjera sama po sebi nam ne govori puno, ne možemo konkretno ocjeniti da li je ovo najbolji mogući portfolio.

Odavdje proizlazi pitanje na koji način komparirati različite moguće kombinacije ω_1 i ω_2. Jedan prirodan način da se to postigne jeste da uporedimo sve moguće kombinacije uz ograničenje da $\omega_1 + \omega_2 = 1$ gdje ćemo pretpostaviti radi svrhe primjera da investitor uvijek zauzima dugu poziciju.

Pokretanjem sljedećeg koda u programskom jeziku R, možemo naći navedene kombinacije

Kod 1.3.1. R kod za izračun očekivane vrijednosti i volatilnosti portfolija

```
dionica1 <- rnorm(0.15, 1, 1000)
dionica2 <- rnorm(0.04, 0.30, 1000)

portmean <- seq(0, 1, by = 0.001)* mean(dionica1) + (seq(1, 0, by = -0.001
)) * mean(dionica2)

portvar <- seq(0, 1, by = 0.001)^2 * var(dionica1)^2 + (seq(1, 0, by = -0.
001))^2 * var(dionica2)^2 + 2 * seq(0, 1, by = 0.001) * (seq(1, 0, by = -0
.001)) * cov(dionica1, dionica2)
portrisk <- sqrt(portvar)
```

Izvor: djelo autora

Pokretanjem gore navedenog koda dobijemo sljedeću tabelu:

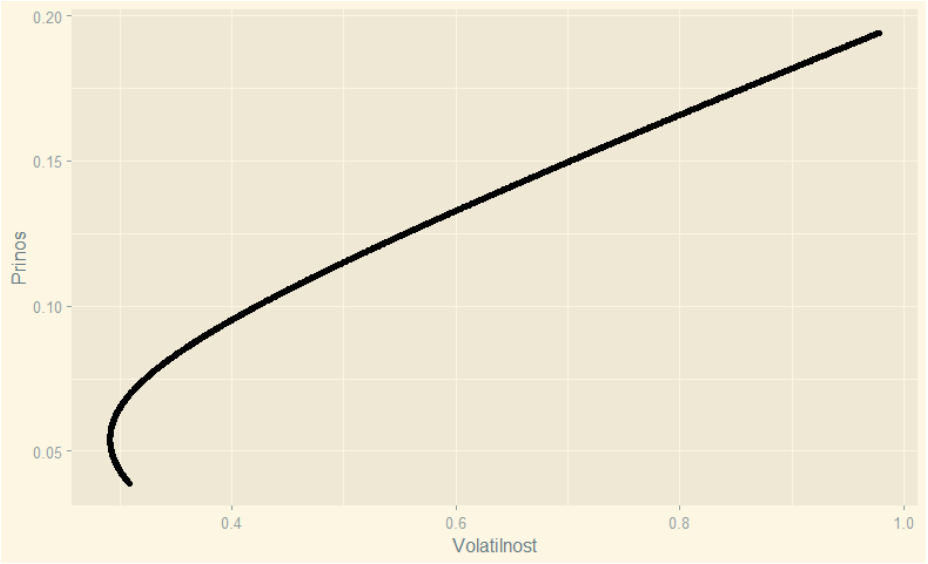
Tabela 1.3.2. Rizik i prinos simuliranog portfolija

Rizik portfolija	***Prinos portfolija***
0.308777	0.038797
0.308436	0.038952
0.308098	0.039107
0.307763	0.039263
0.307431	0.039418

Izvor: djelo autora

Cijela tabela se može naći u prilogu radi reproducibilnosti. Tu tabelu možemo iskoristiti da bismo napravili nešto što se naziva „Markowitzev metak“. Odnosno krivu koja po svom karakterističnom izgledu neke autore podsjeća na metak, gdje je na x-osi rizik određene kombinacije portfolija, a na y-osi prinos tog portfolija. Ovaj grafikon za navedeni primjer ima sljedeći izgled:

Slika 1.3.1. „Markowitzev metak“ za simulirani primjer



Izvor: djelo autora

Na grafikonu vidimo da pri manjem prinosu, manji je i nivo rizika. Međutim, također se može primjetiti da postoje dva portfolija sa istom volatilnošću, ali različitim prosječnim prinosima. Racionalni investitor kako bi maksimizirao svoju korisnost, kada je suočen sa izborom oko jednog od dva portfolija, koji pri istom nivou volatilnosti nude različite prinose, da bi se smatrao racionalnim investitorom, mora odabrati onaj koji mu pruža veći prinos.

Konverzno, može se reći također da i ako ima dva portfolija koji pri različitim nivoima volatilnosti nude isti prinos, investitor se smatra racionalnim ukoliko odabere onaj sa nižom volatilnošću.

Obzirom da performans portfolija mjerimo jedinicom prinos po jedinicu volatilnosti, William F. Sharpe je razvio svoj poznati Sharpe racio koji je određen sljedećom formulom

$$Sharpe\ racio = \frac{R_p - R_f}{\sigma_p}$$

Gdje je R_p prinos na portfolio, R_f bezrizična kamatna stopa, a σ_p volatilnost portfolija. Ovaj racio mjeri višak prinosa na portfolio po jedinici rizika. dakle kada se od njega oduzme bezrizična kamatna stopa.

Bezrizična kamatna stopa predstavlja onu stopu prinosa pri kojoj je investitorov očekivani gubitak jednak nuli. Međutim, u stvarnom svijetu nekada je teško odrediti tu stopu, nekada je nemoguće. Obično se uzima da je bezrizična kamatna stopa, ona stopa koju možemo naći na sigurnim vrijednosnim papirima kao npr. državnim dionicama.

Neki autori navode da se za potrebe analize može koristiti i redukovani Sharpe racio, odnosno onaj gdje je $R_f = 0$. Tako da imamo $Sharpe\ racio = \frac{R_p}{\sigma_p}$. Za gore simulirani primjer najviši Sharpe racio iznosi 0,478303, što govori da uzmemo onaj portfolio koji ima 0,48 jedinica prinosa po jedinici rizika. Očekivana vrijednost tog portfolija se može zapisati na sljedeći način $0{,}056 * 0{,}19741409 + (1 - 0{,}056) * 0.03879672 = 0{,}048638904$. Dok volatilnost ovog portfolija iznosi $0{,}1016894417$. Time smo odredili prema određenom kriteriju optimalni portfolio prema pretpostavkama racionalnosti, te teoretskim pretpostavkama koje smo naveli u samom uvodu.

1.4. Rješenje za matrični oblik

U poglavlju 1.2. izvedena je ciljna funkcija za problem optimizacije, i ona ima sljedeći oblik

$$\boldsymbol{\omega}^T \boldsymbol{\Sigma} \boldsymbol{\omega} = \boldsymbol{\sigma}_r^2$$

Obzirom da ovo predstavlja mjeru nesigurnosti, prema pretpostavkama moderne portfolio teorije, ovo je funkcija koju treba minimizirati. Također, pri minimiziranju trebaju se uvesti određena ograničenja. Prema poglavlju 1.1. investitor želi što viši mogući prinos uz što nižu volatilnost, što se može zapisati u sljedećem obliku

$$\min_{\boldsymbol{\omega}} \boldsymbol{\omega}^T \boldsymbol{\Sigma} \boldsymbol{\omega} = \boldsymbol{\sigma}_r^2$$

$$uz\ ograničenja: \overline{\boldsymbol{P}}^T \boldsymbol{\omega} \geq \boldsymbol{R}_{min}$$

$$\sum_{i=1}^{n} \omega_i = 1$$

S tim da smo dodali ograničenje da suma svih pondera udjela određenog vrijednosnog papira mora iznositi 1.

Analitičko rješenje u ovoj vrsti problema često je neefikasno, s tim da može postojati veliki broj ograničenja, te broj vrijednosnih papira koje investitor posjeduje u svom portofliju je obično > 100. Iz tog razloga često se koriste numeričke metode kako bi se došlo do rješenja, što će se koristiti i u ovom radu. Princip za rješavanje je sličan kao i u navedenom simuliranom primjeru za dvije dionice u poglavlju 1.3.

Ipak ćemo navesti neke od metoda koje se mogu koristiti za rješavanje, a to su sljedeće: algoritam kritične linije, lagrandžov metod i kvadratno programiranje.

1.5. Nedostaci moderne portfolio teorije

Pretpostavke na kojima se zasniva moderna portfolio teorija su poprilično racionalne, same metode su empirijski dokazane. Međutim, postoji niz kritika usmjerenih ka modernoj portfolio teoriji, koji su zasnovani na određenim teorijskim pojednostavljenjima koje je Markowitz uveo kako bi proračuni bili jednostavniji.

Neki od tih argumenata su sljedeći:

- Izbor optimalnog portfolija, MPT ne posmatra kao kontinuirani proces praćenja promjena i prilagođavanja portfolija kroz vrijeme, već kao jednokratnu odluku.

- Pri izboru optimalnog portfolija, MPT ne uzima u obzir transakcione troškove povezane sa trgovinskim transakcijama, kao što su: provizije plaćene kao kompenzacija usluga brokera (kao agenata), razmjenske takse i sl.

- Primjenu MPT dovode u pitanje faktori koji su prisutni u zemljama u tranziciji, kao što su: relativno niska i varijabilna likvidnost finansijskih instrumenata, relativna nestabilnost koeficijenata korelacije između finansijskih instrumenata, varijablinost rizika finansijskih instrumenata tokom vremena, plitko tržište finansijskih instrumenata, problem pouzdanosti informacija i finansijskih izvještaja i sl.

Osvrnut ćemo se na prvu od ove 3 kritike. Doista, ako pogledamo i u simuliranom primjeru naša izmjerena rizičnost postoji samo u zadnjem trenutku T kada smo je izmjerili. Međutim, mi pretpostavljamo da je kretanje vrijednosti dionica neprekidno, što je razumna pretpostavka, pogotovo ako se radi o visoko likvidnim dionicama.

Nadalje, gore navedeno se odnosi samo na jednu dionicu. Možemo zamisliti portfolio koji sadrži 1000 dionica. Do trenutka kada se izračunaju svi potrebni parametri i izvrši optimizacija, moguće je da je dosta finansijskih instrumenata promjenilo svoju vrijednost, a samim tim i rizičnost.

Prosjek većine finansijskih instrumenata nije stacionaran, odnosno prosjek vrijednosti u trenutku t i u trenutku t+n mogu znatno odstupati.

Na osnovu navedenog mnogi finansijski inžinjeri su pokušali pronaći prikladan model koji će otkloniti sve postojeće nedostatke. Krajem 1980-ih, razvoj kompjuterskih nauka i tehnologije je izrodio razvojem koncepta neuralnih mreža i mašinskog učenja.

Finansijski inžinjeri su ubrzo uvidjeli mogućnost primjene novih alata, te moderna portfolio teorija doživjela „preporod“, jer više nije morala ovisiti o nekim rigoroznim statističkim pretpostavkama kako bi bila sprovodiva.

1.6. Uvod u mašinsko učenje i neuralne mreže

Pojam mašinsko učenje odnosi se na automatizovano otkrivanje logički konzistentnih obrazaca u podacima (Shalev-Shwartz & David, 2014, p. XV). Često se pri objašnjavanju mašinskog učenja navodi sljedeći primjer.

Pretpostavimo da ste dobili spam (neželjeni) mail, i želite prestati dobivati takve mailove u budućnosti. „Ručni" način da postignete taj cilj jeste da prema sadržaju maila identifikujete neke obrasce na osnovu kojih možete doći do zaključka da pošta ima neželjen sadržaj (npr. e-mail adresa, naziv pošiljaoca, zahtijevanje ličnih informacija u sadržaju poruke). Naravno, ubrzo bi bilo neefikasno da svaki dan pregledate potencijalne mailove.

Vaš sljedeći korak bi mogao biti da probate napraviti kompjuterski program koji će na osnovu neke predprogramirane logike ustanoviti da je sadržaj pošte neželjen. To bi bio validan pristup, samo ukoliko pošiljaoc pošte neće mijenjati sadržaj tokom vremena. Naravno svjesni smo da interes pošiljaoca zahtjeva da promjeni svoje metode kako bi izbjegavao detekciju.

U ovoj situaciji mašinsko učenje dobija svoj smisao, jer ono na osnovu prethodno učitanih podataka pokušava naći obrasce prema kojim će sa određenim stepenom sigurnosti identifikovati koji mail je spam, te doći do zaključka o generalnoj situaciji.

Naravno, da bi mogao sigurno identifikovati koja vrsta mail-a je spam potrebno je puno primjeraka sadržaja neželjene pošte, naime dovoljno da bi mogao diferencirati između mail-a koji želite pročitati, i onog koji ne želite.

Mašinsko učenje je korisno u sljedećim situacijama (Shalev-Shwartz & David, 2014, pp. 3-4):

- Zadaci koje izvode ljudi/životinje: Postoje brojni zadaci koje ljudi rade na dnevnoj bazi, ali naše razumijevanje toga kako ih radimo nije dovoljno da bi napravili dobro definisan program. Neki od primjera su: vožnja, prepoznavanje govora i prepoznavanje slika.

- Zadaci koji su van ljudskih mogućnosti: postoji još jedan širok spektar zadataka koji se mogu efikasno riješiti putem mašinskog učenja, a odnosi se na tehnike vezane za velikih i kompleksnih skupova podataka (npr. astronomski podaci, procjena vremenske prognoze, itd.). Kako su sve dostupniji digitalno zapisani podaci, sve je jasnije da postoje značajne informacije zakopane u arhivama koje su prevelike i prekompleksne da bi ljudi mogli otkriti bilo šta iz njih.

Ova podjela ujedno i ukazuje na to kada je prikladno koristiti mašinsko učenje. Naime, ako je problem dovoljno jednostavan da se može riješiti koristeći neki deterministički algoritam, nije potrebno da se njegovo rješenje traži putem nekih kompleksnijih načina, što je generalni princip Okamove oštrice.[4]

Nastavljajući u duhu analogije učenja, čovjek je ovdje u principu mašinin učitelj. Odnosno on mora pokazati mašini kojim putem da ide, kako bi odredila sa određenim stepenom sigurnosti da li je identifikovala dobar obrazac, i da li je na osnovu toga indukcija validna.

[4] Okamova oštrica označava princip u kojem su najbolja rješenja obično najjednostavnija.

1.7. Vrste mašinskog učenja

Na osnovu toga u kakvoj interakciji su osoba i program mašinskog učenja postoje različite klasifikacije mašinskog učenja, imamo dakle sljedeće (Pratap, 2017, pp. 50-51):

- Nadgledano učenje: u nadgledanom učenju uloga čovjeka jeste da mašini kaže koji je tačan odgovor, time će mašina dobiti signal da je na pravom putu sa pronalaženjem obrasca. Tako će nastojati da replicira tačne odgovore.
- Nenadgledano učenje: u nenadgledanom učenju čovjek daje mašini podatke, i ona na osnovu podataka pokušava naći neku strukturu.
- Reinforsirano učenje: reinforsirano učenje je vrsta nenadgledanog učenja, ali se razlikuje u tome što ako dobije dobar odgovor dobije neku nagradu, ukoliko ne oduzima se. Time pokušava pronaći optimalno rješenje.

Svaki od ovih modela jer napravljen da simulira neki način učenja, ili želi pronaći generalni proces koji stvara određeni output, na osnovu datih inputa.

Oznake za inpute $\boldsymbol{x} = [x_1, x_2, x_3, \dots, x_n]$ se nazivaju osobinama modela, dok su outputi modela $\boldsymbol{y} = [y_1, y_2, y_3, \dots, y_n]$ oznake modela. Sa ovakvim oznakama možemo nešto konkretnije opisati radi boljeg raspoznavanja.

Nadgledano učenje podrazumijeva da su nam dati parovi osobina i oznaka $(x_1, y_1), (x_2, y_2), \dots (x_n, y_n)$ gdje su $x_1, x_2, \dots, x_n \in X$, $y_1, y_2, \dots, y_n \in Y$, odnosno realizacije slučajnih varijabli, i želimo saznati funkcionalni odnos između varijabli X i Y.

Nenadgledno učenje, nema oznake, nego samo osobine dakle $\boldsymbol{x}$, ali je cilj pronaći kao što je ranije navedeno neku strukturu, ili možda grupaciju prema skrivenim obrascima.

Treća vrsta će biti od posebnog interesa u kasnijim poglavljima, ali za sada ne možemo reći da ima neka nagrada R u trenutku koja je realizacija slučajne varijable sa određenom distribucijom, te ovaj algoritam želi maksimizirati nagradu (Dixon, et al., 2020, p. 8).

Ako se vratimo na primjer neželjenog maila, vidimo da to pripada skupu problema nadgledanog učenja. X će predstavljati neki mail, a Y predstavlja da li smo mi označili mail sa spam ili ne.

Generalno da li se neki problem može naučiti izražava se sa vjerovatno aproksimativno korektnom mogućnosti učenja problema koji je preopširan za potrebe seminarskog rada ali se može pronaći u (Dixon, et al., 2020).

1.8. Neuralne mreže

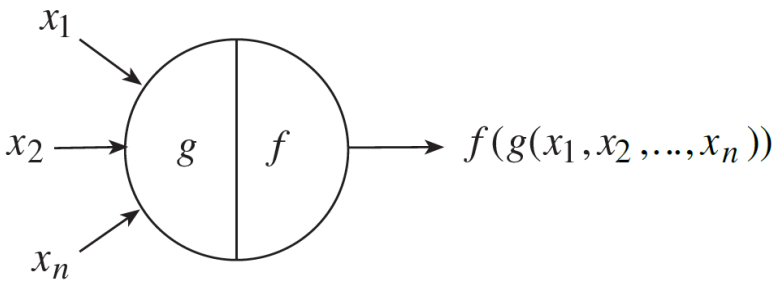
Mašinsko učenje opisuje generalni proces učenja, međutim alat koji se koristi konkretno u svrhu učenja jesu neuralne mreže. Neuralne mreže se definišu kao skup neurona koji između sebe imaju neku funkcionalnu povezanost (Dixon, et al., 2020). Analogija koja se inače predstavlja jeste da su umjetne neuralne mreže apstrakcija prirodnih neuralnih mreža.

Npr. prema (Aggarwal, 2018) „Ljudski nervni sistem sadrži ćelije, koje nazivamo neuronima. Neuroni su sastavljeni jedan sa drugim putem aksona i dendrita, a regije spajanja između aksona i dendrita se nazivaju sinapse. Snaga sinaptičkih veza se mijenja vanjskim podražajima. Ovo je način na koji se učenje odvija“.

Ranije je navedeno, za primjer nadgledanog učenja, da postoji funkcionalna veza između oznaka i osobina. Ova funkcionalna veza se može posmatrati kao sinaptička veza, odnosno neka povezanost koja svoj output $\boldsymbol{y}$ prilagođava na osnovu nekog inputa $\boldsymbol{x}$.

Jedno od polaznih shvatanja jeste da neuralne mreže posmatramo kao „black-box“ algoritme, odnosno određeni input bi trebao proizvesti određeni output. Prema tome neuralnu mrežu možemo predstaviti sljedećom slikom:

Slika 1.8.2: Primjer neurona



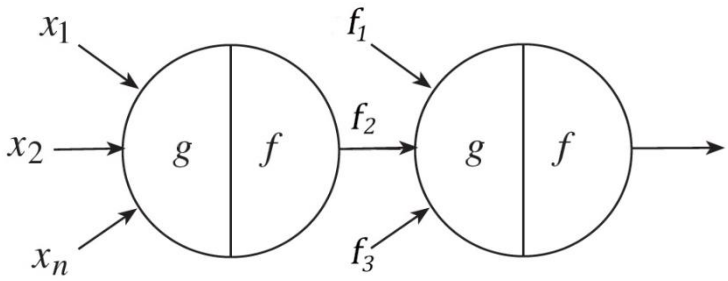
Izvor: (Rojas, 1996)

Kao i prije imamo n inputa, ali radi jednostavnosti proračuna, definiše se neka funkcija g nad inputima, ovo je obično aditivna funkcija ($g(x_1, \ldots, x_n) = \sum_{i=1}^{n} x_i$). To smanjuje broj inputa funkcije f na samo jedan, umjesto opet n inputa. Svakom inputu možemo dodati neku težinu w koja određuje koliko je relevantan input za output pojedinog neurona. Kao što prema slici vidimo output svakog neurona je kompozicija nekih, za sada nama nepoznatih funkcija f i g.

Funkcija f se obično u literaturi vezanoj za neuralne mreže naziva aktivacijskom funkcijom (Rojas, 1996). Aktivacijske funkcije predstavljaju srž mašinskog učenja, a razlog tome će biti nešto kasnije. Za sada nam je bitno da aktivacijska funkcija f definisana nad g će biti input za sljedeći neuron.

Što je prikazano na slici 1.8.3. na sljedećoj stranici.

Slika 1.8.3: Povezanost između dva i više neurona



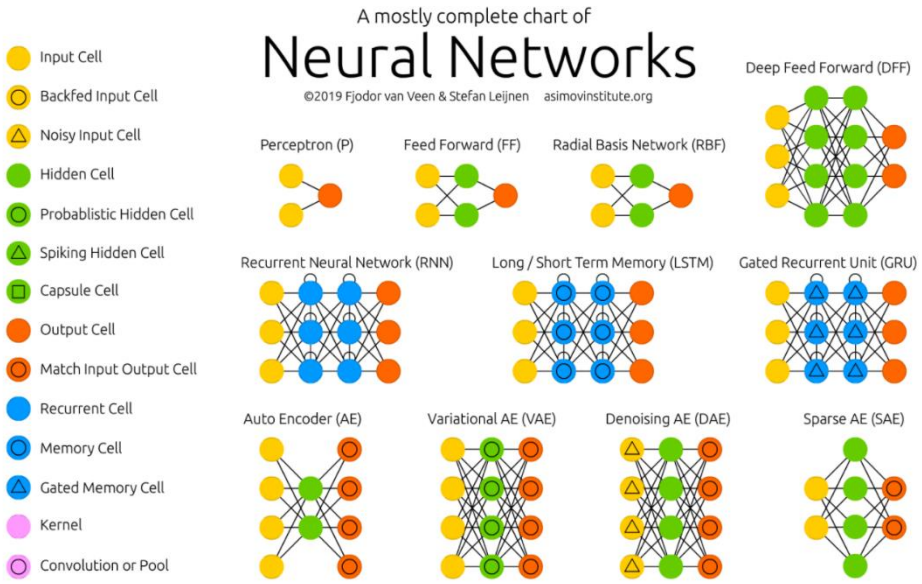
Izvor: modifikovano na osnovu (Rojas, 1996)

Ako nastavimo dalje sa dodavanjem neurona uvijek u principu imamo isto shemu, tj. da je u sljedećem neuronu ulaz, output iz prethodnog neurona, tj. $f(g(f_n(g(x_1, x_2, x_3, \dots, x_n))))$. Eventualno ako nastavimo sa dodavanjem slojeva neurona dobit ćemo neuralnu mrežu.

Postoje razne topologije za neuralne mreže, neke od njih dozvoljavaju rekurziju, neke su sinhronizovane, neke asihnronizovane (ne računaju sve outpute u isto vrijeme), i imamo vagane i nevagane mreže.

Postoji veliki broj topologija sa različitim nivoima složenosti. Ovdje ćemo predstaviti najjednostavniji model radi kompaktnosti, međutim više topologija se može naći u izvoru slike.

Slika 1.8.4: Primjeri topologija neuralnim mreža



Izvor: van Veen, F. & Leijnen, S., 2020. The Neural Network Zoo. [Na mreži] *https://www.asimovinstitute.org/neural-network-zoo/ [Poslednji pristup 18 7 2021].*

Da bismo upotpunili sliku neuralnih mreža definisat ćemo i vagane topologije. Ako se vratimo na sliku 1.8.3 prirodno je pretpostaviti da svaki input nije jednako relevantan za output. Shodno tome definišemo neki ponder $w \in R$, te neku aktivacijsku funkciju kao $\sigma(w * x + b)$ gdje je b parametar pristrasnosti.[5] Neuralna mreža, kao što možemo vidjeti na istoj slici, u najosnovnijem obliku može imati 2 inputa i biti bez skrivenih slojeva. Skriveni slojevi predstavljaju dodatne slojeve neurona koji na osnovu outputa iz prethodnog neurona mogu odrediti dodatno koliko osobina utiče na određivanje oznake. Određivanje optimalnog broja skrivenih slojeva je preopširno za potrebe ovog rada, obzirom da većina paketa za mašinsko učenje ima u sebi ugrađene potrebne algoritme.

Na osnovu gore izvedenog možemo zapisati generalni oblik neuralne mreže

$$\widehat{\boldsymbol{Y}} = \sigma\left(\begin{bmatrix} w_{0,0} & w_{0,1} & \cdots & w_{0,n} \\ \vdots & \vdots & \ddots & \vdots \\ w_{n,0} & w_{1,0} & \cdots & w_{n,k} \end{bmatrix}\begin{bmatrix} x_0^L \\ x_1^L \\ x_n^L \end{bmatrix} + \begin{bmatrix} b_0 \\ b_1 \\ b_n \end{bmatrix}\right)$$

Gdje L predstavlja broj skrivenih mreža, a za $\hat{Y}$ ova formula se može zapisati u sljedećem obliku

$$\widehat{\boldsymbol{Y}} = \sigma(\boldsymbol{w}\boldsymbol{x}^T + \boldsymbol{b})$$

Ako $\boldsymbol{Y}$ označava pravilan output za neko dato $\boldsymbol{x}$, možemo nasumično inicijalizirati naše tegove i dobiti neku procjenu $\widehat{\boldsymbol{Y}}$. Naša procjena u zavisnosti od strukture mreže će vjerovatno odstupati od stvarne vrijednosti. Iz tog razloga definišemo neku funkciju gubitka $\boldsymbol{L} = \left(\boldsymbol{Y} - \widehat{\boldsymbol{Y}}\right)^2$. (Mohri, et al., 2018, p. 59)

Postoji niz različitih načina kako da se postigne ovaj uslov. U jednostavnom slučaju kada se radi o konveksnoj funkciji lokalni minimum je jednak globalnom minimumu, u tom slučaju prvi izvod funkcije će nam dati optimalne rezultate.

Obzirom da dosta problema u stvarnom svijetu nije konveksno, popularno se koristi algoritam gradientnog spuštanja. Ovaj algoritam, generalno govoreći, pokušava naći u nekom ograničenom domenu smjer, gdje je nagib linije tangentne funkciji najniži, te na osnovu unaprijed određenog parametra „ide“ ka tom smjeru pri nekom unaprijed određenom parametru u kojoj „dužini“ će se kretati. Ove metode su preopširne za potrebe ovog rada, ali više se može o njima pročitati u (Aggarwal, 2018, p. 121). Ali ćemo navesti princip na kojem se zasniva učenje na osnovu skupa podataka. Iz skupa podataka uzmemo neki procenat za testiranje, npr. 50%. Pustimo algoritam da minimizira kvadratnu grešku u procjeni na 25% podataka. Onda vršimo nešto što se zove unakrsna validacija sa preostalih 25% da bismo potvrdili tačnost modela, te tako dobijemo istreniranu mrežu.

[5] b- može biti pozitivan i negativan, ako stavimo negativan predznak, znači da zahtjevamo da $w * a$ bude najmanje b, kako bi se aktiviralo. Ukoliko je b pozitivno onda želimo reći da znatno pridodaje funkciji.

1.9. Uvod u reinforsirano učenje

Reinforsirano učenje smo već ranije definisali kao dio skupa generalnih metoda mašinskog učenja. Suštinski predstavlja oblik nenadgledanog učenja, a razlike između klasičnog učenja i reinforsiranog će postati jasnije u nastavku.

Problem reinforsiranog učenja se može svesti na 5 osnovnih varijabli, a to su (Mohri, et al., 2018):

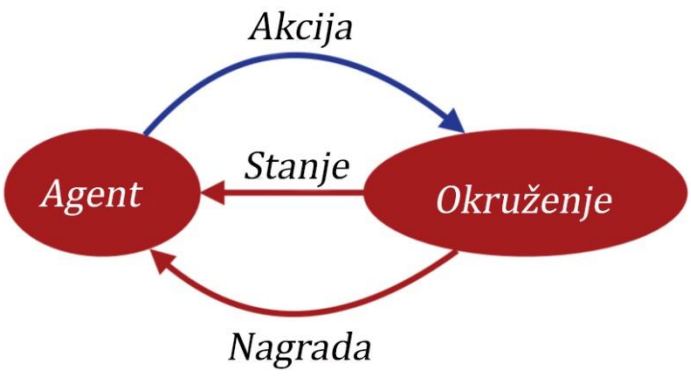
- Agent
- Okruženje
- Stanje
- Akcija
- Nagrada

Ovo predstavlja veoma jednostavnu paradigmu, gdje je neki agent u interakciji sa okruženjem. Čest primjer koji se navodi kako bi se objasnilo reinforsirano učenje jeste način na koji se treniraju psi.

Ako želimo naučiti psa da sjedne na komandu, prvo mu damo znak da treba sjesti, onda ukoliko to izvrši dadne mu se neka nagrada. U ovom primjeru možemo identificirati varijable a to su agent (pas), okruženje (okruženje ovdje može biti proizvoljno), stanje (stojanje, s pretpostavkom da ne sjedi već), akcija (čin sjedanja) i nagrada (slatkiš ili nešto što će mu označiti da je dobro obavio posao).

Ovo je veoma pojednostavljen primjer, međutim nešto generalniji zaključak se može izvesti o tome kako bi se bilo koji agent mogao ponašati u bilo kojem datom okruženju. Iz sljedeće slike možemo vidjeti generalni princip interakcije agenta sa okolinom.

Slika 1.9.4: Model interakcije agenta sa okruženjem



Izvor: (Mohri, et al., 2018, p. 380)

1.10. Markovljev proces odlučivanja

Vidićemo nešto kasnije u nastavku da Markovljev proces odlučivanja predstavlja osnovu reinforsiranog učenja.

Tipični primjer za uvođenje ovog problema jeste sljedeći: (Sutton & Barto, 2018, pp. 25-27) pretpostavite da ste na igraćoj mašini sa k brojem dugmića. Pritiskom bilo kojih od tih dugmića dobijete neku novčanu nagradu. Međutim, vama ta nagrada nije poznata, postoji jedino način da je procijenite. Akcija predstavlja postupak gdje pristisnete neko dugme i dobijete nagradu što se označava sa $q_*(a) = E[R_t|A_t = a]$, gdje $A_t = a$ predstavlja akciju u nekom trenutku t. Da su nagrade poznate racionalno bi bilo uvijek pritistikati dugme sa najvišom nagradom. Međutim, kako nisu, možemo procjeniti nagradu, recimo ako pritisnemo neko dugme 10 puta nalazimo se u trenutku t = 10, možemo sumirati sve do tada primljene nagrade i podijeliti sa t – 1 kako bi procjenili nagradu, to se može zapisati na sljedeći način $Q_t(a) = \frac{\sum_{i=1}^{t-1} R_t * 1_{A_i=a}}{\sum_{i=1}^{t-1} 1_{A_i=a}}$[6] .

Suština ovog problema se svodi na to da optimalna situacija u kojoj bi se igrač tada nalazio je onda kada je $q_*(a) - Q_t(a) \approx 0$. Tada bi nagrade u suštini bile deterministčke, mogli bismo procjeniti koja je najbolja vrijednost u bilo kojem trenutku.

Iz ovog principa definišu se dva osnovna modusa ponašanja (Sutton & Barto, 2018, p. 28): pohlepno i ε- pohlepno ponašanje. Pohlepno ponašanje predstavlja ranije naveden princip da u svakom trenutku t, odabiremo onu akciju koja će nam osigurati najveću očekivanu nagradu. Takav pristup ima svoje prednosti, recimo kada su u pitanju nestacionarni vremenski nizovi. U nestacionarnim nizovima prosjek se s vremenom mijenja, tako da optimalan oblik ponašanja je definisan prema trenutku u kojem se nalaze.

Međutim moguć je i drugi način ponašanja koji se naziva ε- pohlepno ponašanje. Ovaj princip je sličan pohlepnom, ali on kaže da se većinu vremena algoritam ponaša tako, ali da ponekad sa nekom malom vjerovatnoćom ε poduzme neku drugu akciju. Ukoliko smo definisali algoritam da uzima samo najveću nagradu u datom trenutku, onda kažemo da je algoritam eksploatoran, u suprotnom ukoliko algoritam istražuje, onda se shodno tome algoritam naziva eksplorativnim.

Jedna prednost takve metode jeste ta što sa velikim brojem koraka, radi svrhe objašnjenja recimo beskonačnim brojem koraka ova metoda osigurava da će $Q_t(a)$ konvergirati u $q_*(a)$. Također, kao što smo rekli, nagrade u svakom trenutku nisu jednako bitne. Odavdje možemo zaključiti da sljedeće varijable određuju Markovljev proces odlučivanja (Mohri, et al., 2018, p. 381):

• neko početno stanje $s_0 \in$ S.

• skup akcija koje se mogu uzeti A, gdje broj akcija može biti beskonačan

• vjerovatnoća tranizicije $P[S_0|s, a]$: distribucija definisana na budućim stanjima $s_0 = \delta(\mathrm{s}, \mathrm{a})$

• vjerovatnoća nagrade $P[r_0|s, a]$: distribucija vjerovatnoće definisana na $r_0 = r(s, a)$.

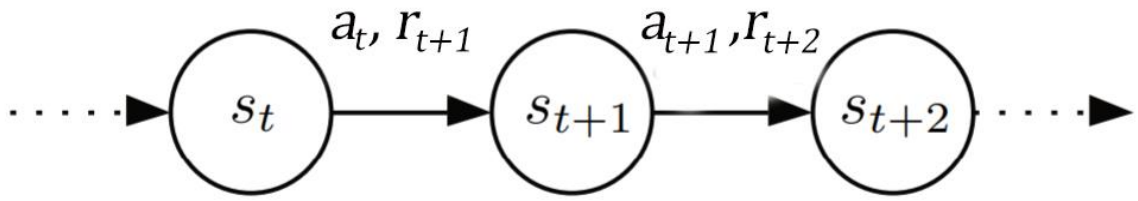
[6] $1_{A_i=a}$ je specifična oznaka. Ona govori da uzmemo vrijednost 1 ukoliko je akcija u vremenu i = a, u suprotnom 0, za drugi slučaj definiše se neka standardna vrijendost kojoj će jednačini biti jednaka.

Razlog zašto se ovi modeli nazivaju Markovljevim se svodi na dva razloga. Prvi je taj što ukoliko bi postojala samo jedna akcija i samo jedno stanje, ova metoda bi bila prosti Markovljev lanac, ali čak i tada bilo bi potrebno da su nagrade deterministične u datom trenutku, ali kao što smo vidjeli po definiciji ovog problema one to nisu.

Za diskretan MPO (Markovljev proces odlučivanja) do odluke se dolazi u odlučivačkim epohama. Epoha u govoru mašinskog učenja se odnosi na situaciju gdje prvo prođe „naprijed" kroz uzorke za treniranje, a onda unazad sa nekom funkcijom definisanom kao npr. u poglavlju 1.8 (Dixon, et al., 2020, p. 145).

Možemo primjetiti da MPO može biti i neprekidan, u smislu vremena, ako se odluke donose u trenucima $t \in \mathrm{T}$, gdje je $\mathrm{T} \in \mathbb{R}$. Ako, je $\mathrm{T} < \infty$, onda se za MPO kaže da je konačnog vidika, međutim to ne znači da je sam MPO konačan. Ranije smo naveli da S i A mogu biti beskonačni, što nije teško zamisliti u stvarnom svijetu. Recimo da imamo robota koji se može rotirati oko svoje ose, on ima beskonačno potencijalnih akcija i stanja u intervalu $[0, 2\pi]$. Shodno tome, za MPO kažemo da je konačan ukoliko je broj stanja i akcija konačan skup. Možemo zamisliti robota koji se uči hodati uz stepenice, ima relativno mali i konačan broj stanja akcija (npr. podigni nogu, spusti nogu), te ima konačan skup stanja (prva stepenica, druga stepenica, ... , n-ta stepenica) (Sutton & Barto, 2018, p. 70).

Slika 1.10.4: Ilustrativna shema MPO



Izvor: (Mohri, et al., 2018)

Ovdje proces postaje malo jasniji u nekom trenutku t, agent poduzima akciju $\mathrm{a_t}$, time prelazi u stanje $\mathrm{s_t}$. Prelaskom iz jednog stanja u drugo agent dobije nagradu $\mathrm{r_t}$. Kao što smo ranije naveli, u kojem se stanju agent trenutno nalazi, zavisi od prethodnih nagrada i stanja, zato se vjerovatnoća da se agent nalazi u trenutnom stanju može izraziti kao $\mathrm{P}[\mathrm{s_t}|\mathrm{s_{t-1}}, \mathrm{a_{t-1}}]$, dok za nagradu se može izraziti kao $\mathrm{P}[\mathrm{r_t}|\mathrm{s_{t-1}}, \mathrm{a}_{t-1}]$. Radi razjašnjenja možemo povezati sa igrom šaha, poredak figura na tabli u trenutno s_t zavisi od poteza koji su prethodili tom stanju.

Za igru šaha mora postojati neka strategija kojom se igrač koristi kako bi pobjedio igru. Tako i u primjeru MPO, moramo definisati neki način ponašanja odnosno politiku kojom će se agent voditi kako bi dostigao optimalnu strategiju.

Mogu se definisati dvije vrste strategija (Mohri, et al., 2018, p. 382):

- stacionarne strategije
- nestacionarne strategije

Prije svega, politika (strategija) koju će agent koristiti predstavlja mapiranje iz stanja u vjerovatnoće da će odabrati neku akciju a. Formalno se može zapisati kao $\pi : S \rightarrow \Delta(A)$, gdje $\Delta(A)$ je skup distribucija vjerovatnoće definisanih na A.

Za politiku se kaže da je stacionarna ukoliko je i deterministična, a determinstična je ukoliko je vjerovatnoća prelaska u tu politiku = 1. Formalnije se može zapisati na sljedeći način: politika π je deterministična ako za bilo koje $s \in S$, postoji unikatno $a \in A$, takvo da je $\pi(s, a) = 1$.

Obzirom da je ovo sigurna politika ona ne ovisi o vremenu, prema tome definiše se kao stacionarna strategija. Ukoliko bi ovisila o vremenu onda bi bila nestacionarna strategija, koja ima svoje prednosti u nekim situacijama.

Ovdje ćemo dodatno eksplicitno definisati nagradu koju agent može primiti u trenutku t, neka je (Aggarwal, 2018, p. 383)

$$R_t = \gamma^0 r_t + \gamma^1 r_{t+1} + \gamma^2 r_{t+2} + \cdots \gamma^i r_{t+i} = \sum_{i=0}^{\infty} \gamma^i r_{t+i}$$

Kao što možemo vidjeti u ovom slučaju nagrada se odnosi na MPO beskonačnog vidika, a koeficijent $\gamma \in [0, 1)$ predstavlja diskontni faktor, slično kao i u sadašnjoj vrijednosti novca. On određuje koliko su nam bitne nagrade u budućnosti. Možemo vidjeti ako u gore navedenu jednačinu uvrstimo da $\gamma = 0$, model uzima samo r_t u obzir. Ako je model konačnog vidika onda vrijedi da $\gamma = 1$. Kumulativna nagrada u datom trenutku zavisi od stanja u kojem se agent nalazi u tom trenutku i strategijom kojom se vodi. Obzirom na to možemo naći da se formule zapisuju i kao $R_t = \sum_{i=0}^{\infty} \gamma^t r(s_t, \pi(s_t))$

Korisno je da određenoj politici dodjelimo neku brojčanu vrijednost, kako bi algoritam mogao maksimizirati tu vrijednost. Ako se prisjetimo u uvodu (poglavlje 1.9) rekli smo da je agent u interakciji sa okruženjem kako bi maksimizirao svoju kumulativnu nagradu.

Prema tome agent da bi morao maksimizirati kumulativnu nagradu krećući se iz stanja $s \in S$, tražeći optimalnu politiku sa najvećom vrijednošću odnosno $V_\pi(s)$.

Postoje dvije vrste optimalne politike u odnosu da li je stacionirana ili nije, a to su (Mohri, et al., 2018):

- za konačni vidik $V_\pi(s) = E_{a_t \sim \pi(s_t)}[\sum_{t=0}^{T} r(s_t, a_t) | s_0 = s]$
- za beskonačni vidik $V_\pi(s) = E_{a_t \sim \pi(s_t)}[\sum_{t=0}^{T} \gamma^t r(s_t, a_t) | s_0 = s]$

Gdje se stanje $a_t$eksplicitno definiše kao stanje u kojem smo završili povodom distribucije $\pi(s_t)$ definisane nad a_t.

Uzimajući prethodno u obzir optimalna politika se definiše na sljedeći način, neka postoji neka optimalna politika π^*ako je $V_{\pi^*}(s) > V_\pi(s)$. Odavdje možemo optimalnu politiku izraziti na sljedeći način (Dixon, et al., 2020, p. 297)

$$V_{t^*} \coloneqq V_{\pi^*} = \max_\pi V_\pi(s) \; \forall s \in S$$

Stanju s pripisujemo neku funkciju akcije-vrijednosti koja se definiše kao (Sutton & Barto, 2018, p. 78)

$$Q_\pi(s,a) = E[r(s,a)] + E_{a_t \sim \pi(s_t)}[\sum_{t=0}^{T} \gamma^t r(s_t, a_t) | s_0 = s, a_0 = a]$$
$$= E[r(s,a) + \gamma V_\pi(s_1) | s_0 = s, a_0 = a]$$

Prethodna jednačina nam daje osnovu da analiziramo optimalnost našeg algoritma, kako bi procjenili da li je „na dobrom putu". Da bi utvrdili to navodimo Bellmanov uslov optimalnosti koji nalaže da: politika je optimalna ako, i samo ako za svaki par $(s,a) \in S\ x\ A$ gdje $\pi(s,a) > 0$ važi sljedeće:

$$a \in \underset{a' \in A}{\operatorname{argmax}}\, Q_\pi(s,a')$$

Što u suštini govori da ukoliko smo došli u stanje nekom politikom π putem koja nas je navodila putem nekih akcija a koje nisu element gore navedenog uvjeta, možemo pronaći bolju politiku π^*, koja će osigurati veći kumulativni povrat.

Ovo nam daje osnovu za evaluaciju bilo koje arbitrarne politike. Prema (Sutton & Barto, 2018, p. 74) i (Mohri, et al., 2018, p. 385) možemo vidjeti da sljedeća jednačina zadovoljava Bellmanov uslov optimalnosti.

$$\forall s\ e\ S, V_\pi(s) = E_{a_1 \sim \pi(s)}[r(s,a_1)] + \gamma \sum_{s'} P[s'|s,\pi(s)] V_\pi(s')$$

Na osnovu toga i nazivaju se Bellmanove jednačine. One nam omogućavaju da MPO zapišemo i u matričnom obliku na sljedeći način:

$$\boldsymbol{V} = \boldsymbol{R} + \gamma \boldsymbol{P}\boldsymbol{V}$$

Gdje $\boldsymbol{V}$ označava vektor sa vrijednostima polisa, $\boldsymbol{P}$ označava matricu vjerovatnoće tranzicije stanja i **R** označava vektor sa nagradama u datom stanju s.

Ovi svi uslovi važe i kada su nam nepoznata stanja, ili nepoznat model okruženja. Postoji niz različitih algoritama koji mogu riješiti gore navedene jednačine. Kada je poznat model okruženja, neke od njih su: algoritam iteracije vrijednosti, algoritam iteracije politike i problem formulisan u obliku linearnog programiranja. Više o tim specifičnim algoritmima se može naći u (Mohri, et al., 2018).

1.11. Proširenje MPO u problem reinforsiranog učenja

U prethodnom dijelu iznesene su osnove MPO, međutim implicitno su važile neke pretpostavke koje nemaju osnovu u stvarnom svijetu. Implicitno smo pretpostavili da je okruženje uvijek potpuno poznato i deterministično, samo smo pretpostavljali da je optimalan prelazak stanja nepoznat.

Kada ove pretpostavke važe, rješenje je egzaktno. Da bi se napravio dobar model stvarnog svijeta, obično je potreban velik broj varijabli koje će ga egzaktno opisati. Ako želimo bolji model, potrebno nam je sve više faktora, a time raste i dimenzionalnost modela. Što ne znači da se problem i sa velikim brojem varijabli ne može riješiti putem MPO, ali kako raste kompleksnost problema, tehnološki zahtjevi rastu.

Obično ono što nam jeste poznato iz stvarnog svijeta su prostor stanja, akcija i diskontni faktor. Za procjenu ostaje politika kojom će se naš algoritam voditi, da bi to postigli, moraju odrediti politiku na osnovu podataka koji im se daju, za razliku od slučaja primjera na početku poglavlja 1.10 sa igraćom mašinom.

U tom slučaju postoje iterativne metode koje pronalaze aproksimativno optimalnu politiku, koja konvergira u optimalnu politiku sa dovoljno podataka i dovoljno testiranja (Lagoudakis & Parr, 2003).

Postavlja se pitanje kako u kontekstu pojednostavljenog modela svijeta dati agentu upute za pronalazak optimalne politike. Slično kao u slučaju nadgledanog učenja, potrebno je da agentu damo podatke i da ga istreniramo na osnovu nekog manjeg skupa, gdje će svoju politiku odrediti na osnovu nagrada iz tog skupa (Mohri, et al., 2018, p. 393).

Postoje dva osnovna načina kako implementirati reinforsirano učenje (Sutton & Barto, 2018, p. 159):

- reinforsirano učenje bez modela – zasnovano je na učenju
- reinforsirano učenje zasnovano na modelu – zasnovano je na planiranju

Pod modelom okruženja se podrazumijeva bilo šta što model može iskoristiti za predviđanje budućeg stanja okruženja. Odnosno u toj situaciji model na osnovu stanja i akcija može dolazi do sljedećeg stanja, ako je model stohastičan, postoji više mogućih stanja sa različitim stanjima.

Ako model može u potpunosti za svako moguće stanje generisati vjerovatnoću prelaska u to stanje, onda se taj model naziva model distribucije. Međutim ponekad model može proizvesti vjerovatnoću za samo određeni podskup svih mogućih stanja, ako imamo taj model onda ga nazivamo modelom uzorka (Sutton & Barto, 2018, p. 159).

Planiranje u kontekstu reinforsiranog učenja odnosi na model koji za input ima neki model stanja prirode, a za output ima poboljšanu politiku (strategiju) sa datim stanjem prirode.

Postoje dvije vrste ovakvog planiranja, ono koje se najčešće koristi u mašinskom učenju jeste model koji istražuje stanja, kako bi odredio optimalnu politiku, i ovo planiranje se naziva planiranje stanja. Drugi model se zove planiranje planova, njegova suština se svodi na to da prolazi kroz sve moguće planove, dok ne nađe optimalni (Sutton & Barto, 2018, p. 159).

1.12 Q-učenje

Postoji veliki broj metoda koje dovoljno dobro aproksimiraju vrijednost Q funkcije. Međutim, za potrebe ovog seminarskog rada koristiti ćemo aproksimaciju same Q funkcijie na sljedeći način (Sutton & Barto, 2018, p. 131):

$$Q(S_t, A_t) \leftarrow Q(S_t, A_t) + \alpha[R_{t+1} + \gamma \max_a(S_{t+1}, a) - Q(S_t, A_t)]$$

Kao što smo ranije naveli $Q(S_t, A_t)$ predstavlja funkciju stanja-akcije, prema navedenoj formuli, ne istražujemo politiku, iz tog razloga ovaj tip metoda u reinforsiranom učenju se naziva metod učenja bez politike.

Prema ovoj formuli Q funkcija u trenutku t se ažurira sa nagradom iz trenutka $t+1$. Te procijenjenom budućom vrijednosti maksimalne Q funkcije iz trenutka $t+1$. α predstavlja parametar veličine koraka iz trenutka t u trenutak $t+1$. Postoje različiti načini da se definiše α. Može se koristiti konstantna α, može se definisati kao $\alpha = \frac{1}{n(a)}$ gdje $n(a)$ predstavlja broj puta koliko je neka akcija preduzeta. Nešto generalnije α se zapisuje kao $\alpha(a)$, tj. obično se definiše kao neka funkcija akcije za dati problem (Sutton & Barto, 2018).

Za ovaj postupak je dokazano da konvergira sa vjerovatnoćom 1 (Watkins & Dayan, 1992, pp. 282-286), što znači da algoritam doista putem gore navedene formule za $Q(S_t, A_t)$ sa dovoljnim brojem koraka i dovoljno treniranja aproksimira ciljnu funkciju q^*.

Q-učenje i dalje zahtjeva određivanje neke politike, iz razloga što Q funkcija u datom trenutku je određena sljedećom akcijom, međutim optimalna politika se ne može odrediti odabirom samo jedne akcije. Da bi ovaj algoritam konvergirao potrebno je da se Q funkcije za sve akcije konstantno ažuriraju (Sutton & Barto, 2018).

Neki autori (Dixon, et al., 2020, pp. 315-316) predlažu da se to može postići tako da se napravi tabelarna reprezentacija funkcija akcije stanja, za sve prethodno odabrane parove $(s, a) \in S\ x\ A$ da bi se mogla procjeniti vrijednost $\max_{a'} Q_{t+1}(s', a')$ koristeći prethodne podatke i predstojeće tranzicije.

Algoritam za postizanje ove aproksimacije se može zapisati na sljedeći način:

- inicijaliziramo parametre algoritma: $\alpha(a) \in (0.1]$, i dovoljno malo $\varepsilon > 0$
- inicijaliziramo $Q(s, a)$, za svako $s \epsilon S^+, a \epsilon A(s)$, arbitrarno osim da u terminalnom koraku $Q(terminalno, \ .) = 0$

Za svaku epizodu se se onda:

- inicijalizira S
- bira A iz skupa S koristeći politiku izvedenu iz Q
- odabire A, i posmatra se rezultirajuća nagrada R, kao i stanje S'
- $Q(S, A) \leftarrow Q(S, A) + \alpha[R + \gamma \max_a[Q(S', a) - Q(S, A)]$
- $S' \leftarrow S$, do terminalnog stanja.

2. Metodologija

Za primjenu reinforsiranog učenja, tj. Q-učenja koristit ćemo programski jezik Python, na skupu podataka od 15 dionica, a to su: Apple (AAPL), NVIDIA (NVDA), Tesla (TSLA), Amazon.com (AMZN), Moderna (MRNA), Microsoft (MSFT), Boeing (BA), Facebook (FB), AMC Entertainment (AMC), Alphabet C (GOOG), Alibaba ADR (BABA), JPMorgan (JPM), Bank of America (BAC), Nio A ADR (ADR), Visa (V). Ove dionice su odabrane zbog visoke likvidnosti na tržištu.

Koristit ćemo podatke, tj. dnevne cijene vrijednosnih papira za period od 14.1.2020 do 9.7.2021 što ukupno iznosi 376 podataka u uzorku,

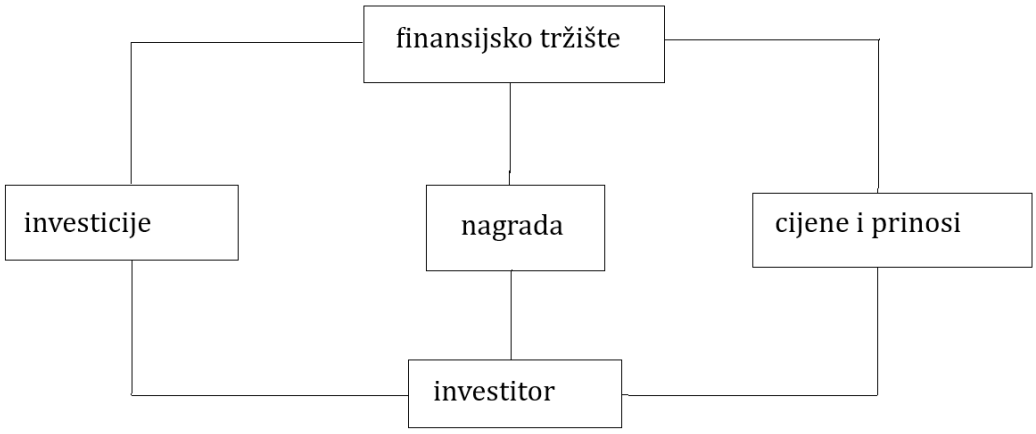
Za obučavanje modela koristi se skup od 180 podataka, gdje agent pokušava maksimizirati svoju nagradu koja je definisana kao pozitivan prinos, onda će se Q-funkcija koju je agent naučio iskoristiti na skupu ukupnih podataka, što znači da će agent morati primjeniti ono što je naučio na 196 preostalih podataka.

Također, za usporedbu će se koristiti Markowitzev model gdje minimiziramo volatilnost portfolija uz maksimiziranje prinosa.

Dozvolit ćemo negativne težine, što znači da je agentu dozvoljeno da uči i kratke pozicije, sukladno tome dozvoljena je i kratka pozicija u Markowitzevom portfoliju.

Ovdje ćemo objasniti na kojem principu su definisane akcije, stanja i nagrade za agenta putem sljedeće slike.

Slika 2.1: Model stanja-akcije-nagrade za finansijsko tržište



Izvor: (Neuneier, 1998, p. 1)

Prema slici akter je investitor i on u trenutku t se može naći u vektoru stanja $\boldsymbol{x_t} = [S_t, K_t]$, gdje S_t predstavlja elemente koji opisuju finansijsko tržište (prinos, cijene finansijskih instrumenata), dok K_t predstavlja koliko je kapitala uloženo u određeni finansijski instrument.

Pretpostavlja se također da akter u ovom modelu ne može uticati na tržište na bilo koji način, i da nema transakcijskih troškova. Investitor može donijeti, kao što je navedeno ranije, odluke (poduzeti akcije) a da zauzme dugu ili kratku poziciju, drži sredstva, i/ili ih ne drži nikako.

Svakim donešenjem odluke u trenutku t on mijenja svoj portfolio (sa veoma malom vjerovatnoćom potencijalno ostaje isti) u neki portfolio K_{t+1} i suočava se sa novim stanjem na tržištu S_{t+1}, dakle dolazi novo stanje $\boldsymbol{x}_{t+1}$. Ova odluka će se desiti sa nekom vjerovatnoćom prelaska u naredno stanje $p(x_{t+1}|x_t, a_t)$. Prelaskom u naredno stanje akter povećava svoju kumulativnu nagradu R_{t+1}, koju nastoji maksimizirati.

Optimalna Q-funkcija Q^* uvažavajući Bellmanov uvjet optimalnosti je sljedeća:

$$Q^*(x_t,\ a_t) \coloneqq \sum_{x_0}^{x_{t+1}} p(x_{t+1}|x_t, a) r_t + \ \gamma \sum_{x_0}^{x_{t+1}} p(x_{t+1}|x_t, a) r_t V^*(x_{t+1})$$

Obzirom na obimnost potpunog koda predstavit ćemo algoritam koji će se koristiti za procjenu Q-vrijednosti funkcije. Ovaj algoritam je zasnovan na izloženom u poglavlju 1.12. sa pojedinim izmjenama koje će biti objašnjene u daljem tekstu.

Klasični oblik Q-učenja za potrebe ovog rada je neefikasan iz razloga što bi morali napraviti tabelu sa svim vrijednostima Q-funkcije, za svako moguće stanje i akciju, kako postoji veliki broj potencijalnih stanja na finasijskom tržištu, potreban nam je drugi pristup.

Shodno tome koristit ćemo drugi pristup koji se naziva duboko Q-učenje (engl. deep Q-learning), i ono predstavlja spoj dubokog mašinskog učenja (neuralne mreže sa više „skrivenih slojeva“) i reinforsiranog učenja.

Za razliku od klasičnog Q-učenja, zbog preopširnosti svih mogućih kombinacija stanja i akcija, umjesto tabele formira se neuralna mreža. Input ove mreže jesu samo stanja iz posmatranog okruženja, te se pomoću nje procjenjuju Q-vrijednosti za svaku moguću akciju.

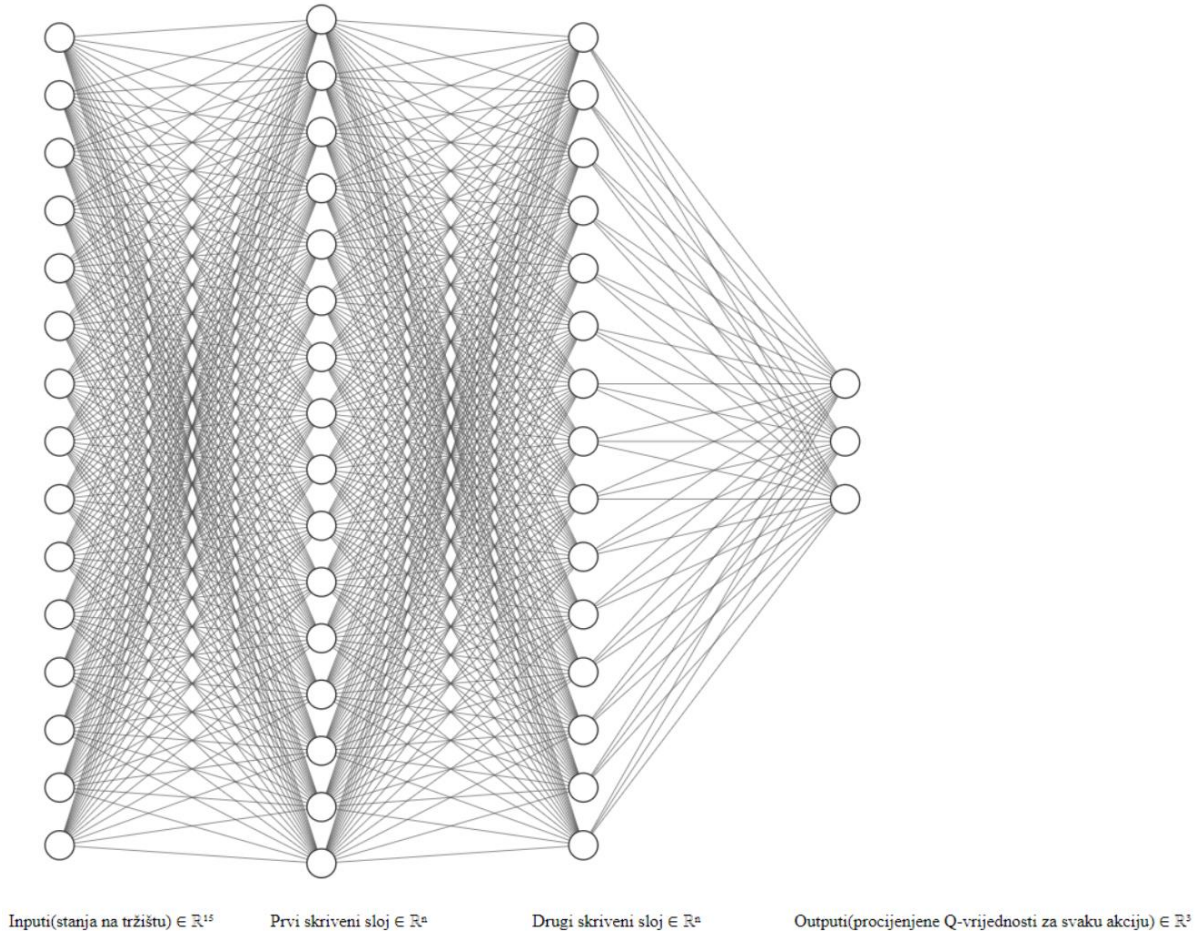
Dok metode reinforsiranog učenja, pokušavaju pronaći maksimum date funkcije, metode mašinskog učenja su korisne jer putem njih možemo pronaći minimum kvadratnog odstupanja ciljane od procijenjene vrijednosti (predstavljenu funkcijom gubitka, engl. „loss function“).

U slučaju ovog modela koristiti će se sljedeća funkcija gubitka:

$$loss(Q^*) = \left(\sum_{x_0}^{x_{t+1}} p(x_{t+1}|x_t, a) r_t + \ \gamma \sum_{x_0}^{x_{t+1}} p(x_{t+1}|x_t, a) r_t V^*(x_{t+1}) - \sum_{x_0}^{x_{t+1}} p(x_t|x_t, a) r_t V^*(x_t) \right)^2$$

Radi lakšeg razumijevanja slikovni model dubokog Q-učenja primjenjenog na optimizaciju portfolija vrijednosnih papira se može predstaviti na sljedeći način:

Slika 2.2: Model dubokog Q-učenja primjenjenog na portfolio vrijednosnih papira



Izvor: djelo autora

U gore navedenoj slici svaki skriveni sloj kao što je ranije navedeno predstavlja niz aktivacijskih funkcija.

Osim toga u ovoj primjeni koristiti ćemo opdajuću epsilon politiku, gdje epsilonu prvenstveno dodijelimo neku poprilično visoku vrijednost $\varepsilon = p_{start}$ gdje će opadati pri određenoj stopi sve do nekog epsilon kada smo sve bliže optimalnoj strategiji $\varepsilon = p_{kraj}$. Ovo opadanje se kreće po sljedećoj formuli:

$$r = \max\left(\frac{BrojEpizoda}{BrojEpizoda} - \frac{trenutnaEpizoda}{Broj\ epizoda}, 0\right), \varepsilon \leftarrow r\left(p_{start} - p_{kraj}\right) + p_{kraj}$$

3. Rezultati primjene mašinskog učenja u optimizaciji portfolija vrijednosnih papira

Prema ranije navedenoj metodologiji napravljen je kod u Python-u, te su postavljeni sljedeći inicijalni parametri:

- $\alpha(a) = 0.9\ (konstanta)$,
- $\gamma = 0{,}5\ (konstanta)$ i,
- $\varepsilon = 0{,}9$

Inicijalizirana je neuralna mreža sa dva sloja, gdje prvi sloj ima 100 neurona, a drugi 50, te se koristi ELU(engl. Exponential Linear Unit) aktivacijska funkcija, i ona je definisana na sljedeći način:

$$y = \begin{cases} \alpha(e^x - 1), & x > 0 \\ x, & x < 0 \end{cases}$$

Nagrada za model je definisana prema ranije navedenoj formuli u vektorskom obliku, te nastoji maksimizirati nagradu koju ostvari kada je prinos veći od nule. Udio svake dionice u ukupnom portfoliju se inicijalizira nasumično, onda će se na osnovu ukupne nagrade pokušavati odrediti optimalni udjeli dionica, uz minimalnu volatilnost. Model je pokrenut u 500 epizoda, i veličinom serije od 250, te ćemo u nastavku predstaviti performanse modela. Za usporedbu ćemo koristiti portfolio optimiziran prema Markowitz-evom modelu.

Epsilon se smanjuje prema ranije navedenoj formuli pri stopi od 0,99 te su u sljedećoj tabeli navedeni koraci u kojim je epsilon smanjen.

Tabela 3.1: Vrijednosti ε za epizode u kojima je došlo do promjene

BROJ EPIZODE	**EPSILON(ε)**
1	0,9
32	0.891
63	0,88209
94	0.8732691
125	0.8645364
157	0.85589104491
188	0.8473321344609
219	0.8388588131162911
250	0.8304702249851281
282	0.8221655227352769
313	0.8139438675079241
344	0.8058044288328449
375	0.7977463845445164
407	0.7897689206990712
438	0.7818712314920805
469	0.7740525191771597

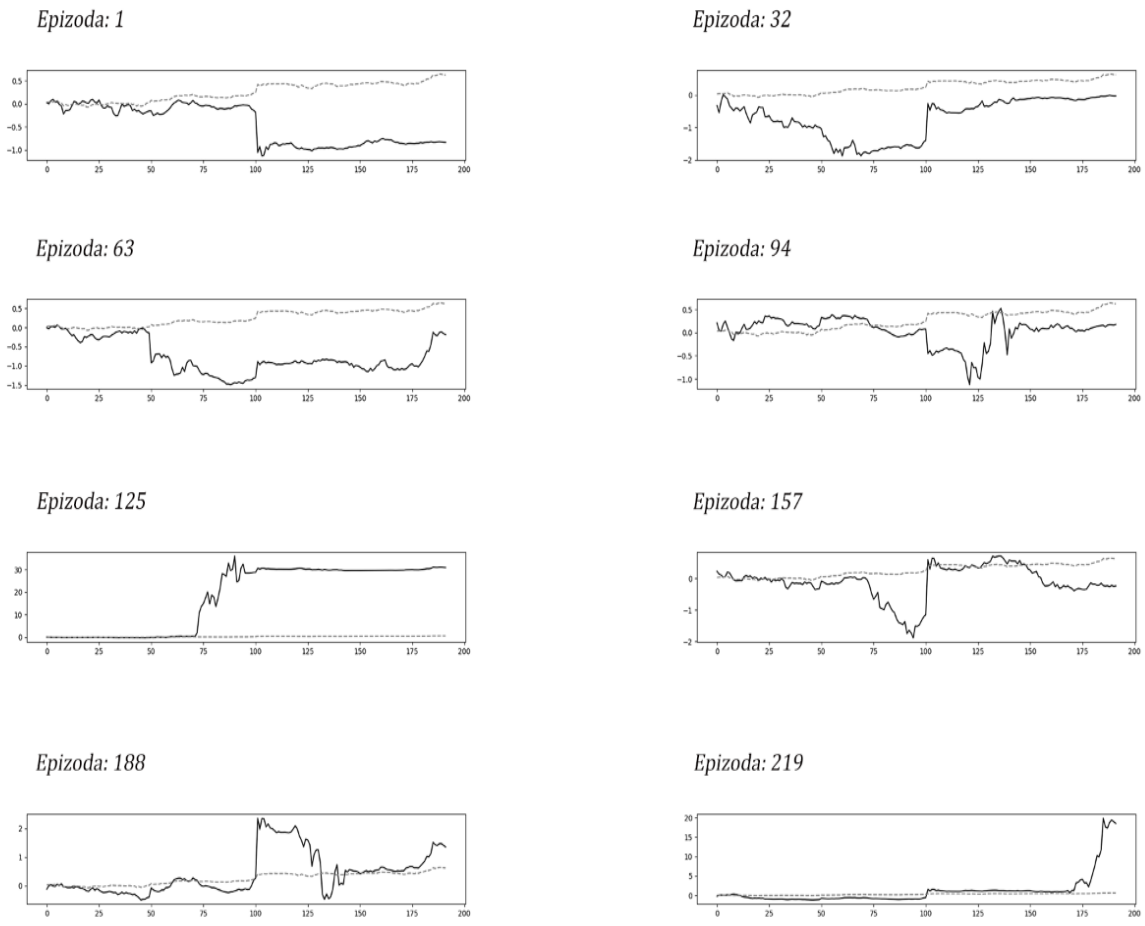
Izvor: djelo autora

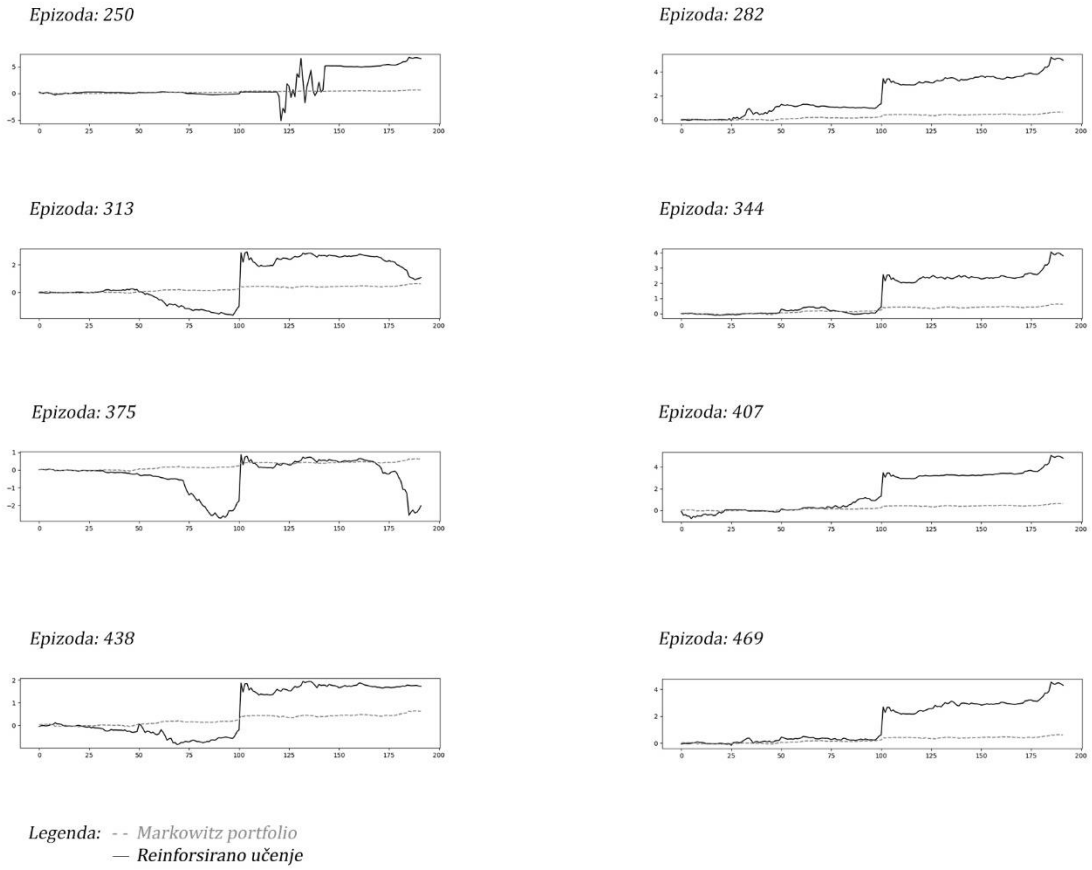
Obzirom da je okruženje s kojim je naš agent u interakciji stohastično, bilo bi poželjno da model obavlja više istraživačkih akcija, obzirom da ne možemo računati na dugoročnu konzistentnost jedne strategije. Prema tome, ova vrijednost epsilona je zadovoljavajuća, te vidimo da je u epizodi 469 epsilon i dalje imao poprilično visoku vrijednost odnosno 0.76, što znači da će od 100 akcija 76 puta izabrati neku nasumičnu istraživačku akciju, a 24 puta će eksploatisati ono što je već naučio iz cijena vrijednosnih papira za preduzeća navedena u metodologiji.

Sljedeće što ćemo pogledati jesu prinosi na portfolio ostvareni u datim epizodama za portfolio koji koristi metod Markowitz, sa ciljem maksimiziranja prinosa, te primjene reinforsiranog učenja.

Na slici 3.1. možemo primjetiti postepeno poboljšanje u ostvarenim prinosima za portfolio koji koristi reinforsirano učenje. Do 157. epizode ima znatno lošije performanse od portfolija optimiziranog putem Markowitz-evog modela.

Slika 3.1: Model dubokog Q-učenja primjenjenog na portfolio vrijednosnih papira





Izvor: djelo autora

Također je vidljivo da je u okrugu vremena t=100 identificirao određeni obrazac na kojem je ostvaruje kumulativni prinos nešto malo veći od 4, na istreniranom modelu ćemo vidjeti da li će taj obrazac ostati, obzirom da se u fazi učenja ispituje „prozor“ od samo 196 zabilježenih cijena vrijednosnih papira, dok poslije 157. epizode model ostvaruje konzistentno bolje performanse.

Obzirom da je vidljivo da već pri 500 epizoda model je pronašao neku stabilnu strategiju, možemo politiku koju je otkrio primjeniti na ukupnom skupu podataka od 196 podataka za cijene od 15 dionica. Nakon što je to obavljeno dobiveni su sljedeći rezultati

Tabela 3.2: Rezultati za dva posmatrana portfolia

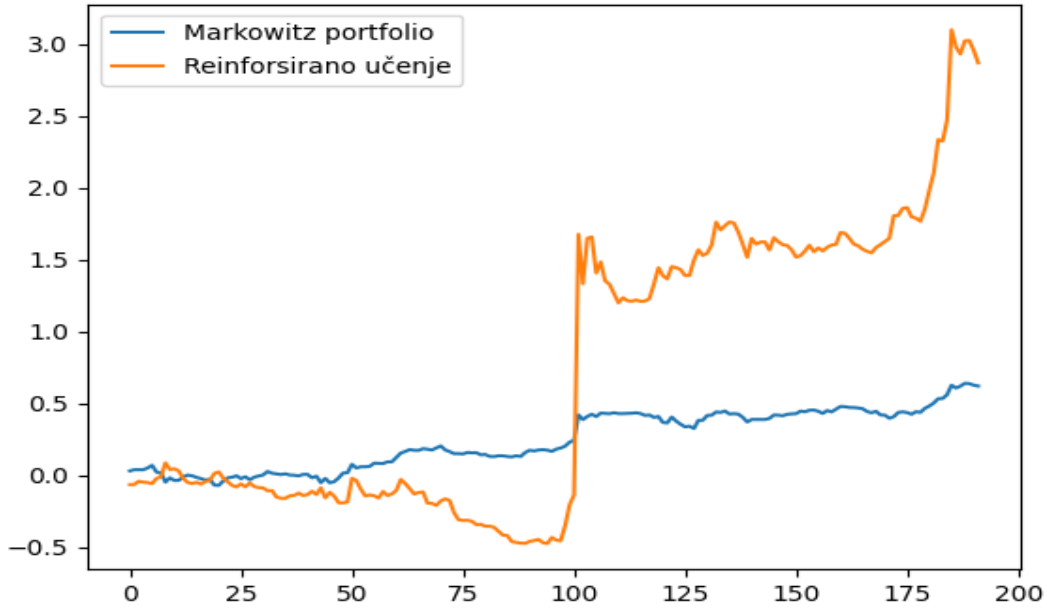
Vrsta portfolija	*Prosječni prinosi*	*Varijansa*	*Std. devijacija*	*Sharpe racio*
Reinforsirano učenje	0.018	0.1764	0,42	0,042856
Markowitz portfolio	0,0032	0,0221	0,14866	0,0215256

Izvor: djelo autora

Kao što možemo vidjeti iz date tabele, portfolio koji koristi reinforsirano učenje ima znatno više prosječne prinose u odnosu na drugi portfolio, naravno viši prinos prema pravilu dolazi i sa većom rizičnošću portfolija. Zato ćemo komparirarti ova dva portfolija na osnovu Sharpe racia sa pretpostavljenom bezrizičnom stopom = 0, te kao što možemo vidjeti u tabeli Sharpe racio za portfolio reinforsiranog učenja iznosi 0,042856; dok za Markowitz-ev iznosi 0,0215256, što znači da se ostvaruje veći dodatni prinos za dodatnu jedinicu rizika korištenjem portfolija sa dubokim Q-učenjem.

Iznimna volatilnost posebno je uočljiva ako se istreniran model pokrene ponovo na skupu od 196 podataka:

Slika 3.2: Rezultati primjene istreniranog modela dubokog Q-učenja



Izvor: djelo autora

Iz predstavljenog grafikona vidljivo je da oba modela imaju donekle inverzna kretanja, odnosno periode ostvarivanja pozitivnih i negativnih prinosa, ali uočljiva je i znatno veća stabilnost Markowitz-evog portfolija. Ponovo je vidljiv i obrazac oko vremena t=100 gdje model ostvari znatan prinos. Razlog tome jeste što se u januaru 2020. godine dogodila Gamestop kontraverza koja je povezana sa industrijom video igara, u koju je znatno uključena kompanija Nvidia čije su dionice dio našeg portfolija. Obzirom da model generiše rezultate za svakih od 196 trenutaka u vremenu, zbog preopširnosti nije moguće predstaviti sve optimalne pondere koje je model odredio.

Nakon toga vidimo ponovo znatno volatilniju strategiju od uslovno rečeno benchmark portfolija koji koristi Markowitz-evu metodu, ali sa znatno izraženim i dobitcima i gubitcima. Na kraju model je ponovo ostvario znatan prinos trgovajući sa dionicama MRNA.

Zaključak

Suština primjene dubokog Q-učenja u optimizaciji portfolija vrijednosnih papira jeste da nadoknadi nedostatke moderne portfolio teorije. Naveli smo da je glavni nedostatak moderne portfolio teorije to što „izbor optimalnog portfolija, MPT ne posmatra kao kontinuirani proces praćenja promjena i prilagođavanja portfolija kroz vrijeme, već kao jednokratnu odluku“. Dakle portfolio optimiziran putem Markowitz-eve metode, ne uzima u obzir prethodne događaje unutar cijena dionica. U trenutku izračuna odbacuje sve prethodne događaje koji znatno utiču na kretanje cijena dionica u budućnosti. Agent istreniran na historijskim podacima je uspio u podacima sa vremenskim rasponom od godinu dana uspio razviti dvije strategije, mada je jedna razvijena na osnovu testnih podataka, vidjeli smo da je nakon toga uspio samostalno ostvariti doista velik prinos trgovanjem sa MRNA dionicama.

Oba portfolija su također imala dodatni izazov to što su morala trgovati u vremenskom rasponu kada se desio prvenstveno pad berze, a također i zanimljiv i kontraverzan događaj u januaru 2020. godine. Oba portfolija su se na veoma zanimljiv način suočila sa oba događaja, vidimo da Markowitz portfolio je održao relativno stabilnu strategiju, mada nije uspio ostvariti znatne prinose na tom događaju. Kod portfolija sa reinforsiranim učenjem vidimo nešto drugačiju priču gdje je uspio ostvariti znatno visoke prinose na osnovu ovog događaja. Zanimljivo je da ovaj model, uopšte nije imao bilo kakve eksterne informacije potrebne da dođe do zaključka o tom događaju, obzirom da je njegov jedini input bio skup od 15 dionica koji samo uključuje dionice Nvidie, odnosno preduzeća koja je usko povezana sa industrijom gdje Gamestop trguje. Uspio je ponoviti svoju strategiju izvan uzorka sa dionicama MRNA, čije su vrijednosti znatno skočile početkom masovnog vakcinisanja povodom pandemije COVID-19.

Međutim, vidimo da je portfolijo reinforsiranog učenja znatno osjetljiviji na promjene u tržšitu. Na slici 3.2. možemo vidjeti da gdje postoje mali skokovi uzrokovani raznim fluktuacijama na tržištu sa portfolijom izračunatog putem metode MPT, ti skokovi su znatno izraženiji kod portfolija sa reinforsiranim učenjem. Ta pojava se odražava u povećanoj volatilnosti takvog portfolija što je vidljivo iz tabele 3.2. Zaključak koji možemo izvesti iz predočenog jeste ujedno prednost i nedostatak portfolija optimiziranog putem mašinskog učenja.

Naime, vidjeli smo sposobnost portfolija da oktrije strategije unutar skupa podataka, ali njegova znatno povećana volatilnost znači da neće biti privlačan korisnicima sa nešto većom averzijom prema riziku. Osim toga, strategije koje otkrio moguće je da postoje samo u određenom vremenskom intervalu. Cijene dionica su prema svojoj prirodi veoma dinamično okruženje, te je za očekivati da se odnosi između pojedinih vrijednosnih papira promijene tokom vremena, skladno tome, model bi bilo potrebno konstantno revidirati i reevaluirati u funkciji pronalaska vremenskog perioda koji najbolje odražava razvojne tendencije vrijednosnih papira sadržanih u investitorovom portfoliju.

Još jedan nedostatak proizlazi iz same prirode modela. Kako model sa vjerovatnoćom epsilon čini nasumične postupke, nije moguće znati tačno koju će politiku agent iskoristiti, te često ta strategija nije jasna a priori, sasvim je moguće da model uoči neke obrasce u cijenama koje investitoru nisu odmah jasne, što stvara dodatni stepen nesigurnosti u investicionoj strategiji, a predstavlja svojstveni „trade-off“ za ovaj model, koji može ponuditi zanimljive i visokoprofitabilne strategije, pri cijeni od dodatnog stepena neizvjesnosti.

Popis slika i tabela

Literatura

Aggarwal, C. C., 2018. *Neural Networks and Deep Learning.* Cham: Springer.

Dixon, M. F., Halperin, I. & Bilokon, P., 2020. *Machine Learning in Finance From Theory to Practice.* Cham: Springer.

Kozarević, E., 2009. *Analiza i upravljanje finansijskim ricizima.* Tuzla: CPA.

Lagoudakis, M. G. & Parr, R., 2003. Least-Squares Policy Iteration. *Journal of Machine Learning Research 4,* p. 1113.

Markowitz, H., 1959. *Portfolio Selection: Efficient Diversification of Investments.* New York: John Wiley & Sons.

Mohri, M., Rostamizadeh, A. & Talwalkar, A., 2018. *Foundations of Machine Learning.* Cambridge: The MIT Press.

Neuneier, R., 1998. Enhancing Q-learning for Optimal Asset Allocation. *Advances in Neural Information Processing Systems,* Tom. 8, p. 1.

Pratap, D., 2017. *Statistics for machine learning.* Birmingham: Packt Publishing.

Rojas, R., 1996. *Neural Networks A Systematic Introduction.* Berlin: Springer.

Shalev-Shwartz, S. & David, S. B., 2014. *Understanding Machine Learning From Theory to Algorithms.* New York: Cambridge University Press.

Sutton, R. S. & Barto, A. G., 2018. *Reinforcement Learning An Introduction.* Second ur. Cambridge: The MIT Press.

van Veen, F. & Leijnen, S., 2020. *The Neural Network Zoo.* [Na mreži] Available at: https://www.asimovinstitute.org/neural-network-zoo/ [Poslednji pristup 18 7 2021].

Watkins, C. & Dayan, P., 1992. Q-learning. *Machine Learning, 8,* pp. 282-286.