Mathias Smidsrød Fjelnseth

Porteføljeforvaltning med forsterkningslæring

Masteroppgave i Økonomi og administrasjon
Veileder: Denis Becker
Mai 2022

Masteroppgave

NTNU
Norges teknisk-naturvitenskapelige universitet
Fakultet for økonomi
NTNU Handelshøyskolen

NTNU
Kunnskap for en bedre verden

Mathias Smidsrød Fjelnseth

Porteføljeforvaltning med forsterkningslæring

Masteroppgave i Økonomi og administrasjon
Veileder: Denis Becker
Mai 2022

Norges teknisk-naturvitenskapelige universitet
Fakultet for økonomi
NTNU Handelshøyskolen

NTNU
Norwegian University of
Science and Technology

Sammendrag

Oppgaven tar for seg porteføljeforvaltning gjennom anvendelsen av forsterkningslæring. Denne maskinlæringsprosessen skal løse problemstillinger tilknyttet porteføljeforvaltning ved å benytte en handelsagent som lærer gjennom sekvensielle beslutningsprosesser. Agenten skal lære i et fremstilt miljø som representer finansmarkedet. Det benyttes en modellfri forsterkningslæring som tillater agenten å gjennomføre beslutninger fordelt i tidstrinn og samtidig lære optimal handlingsstrategi. Handelssystemet fremstilles gjennom kombinasjonen av finansielle matematiske formuleringer og forsterkningslæringsalgoritmer. Oppgaven anvender forsterkningslæringsalgoritmen Deep Deterministic Policy Gradient som en del av systemet. Tilnærmingen til læringsprosessen er at agenten skal "prøve og feile" for å skape problemløsning. For at dette skal være mulig er det etablert et minne i systemet. På denne måten kan agenten huske hvilke beslutninger som er gjennomført og lære av disse ved neste situasjon. For å undersøke effektene av algoritmen kontrolleres resultater gjennom back-testing på historisk data. Resultatene blir sammenlignet med markedsindeksen Dow Jones, tilhørende børsnoterte fond (DIA) og en mean-variance porteføljestrategi for å undersøke ytelsen av forsterkningslæringsagenten. Denne simuleringen av handel sjekker porteføljens robusthet, lønnsomhet og risikosensitivitet. Undersøkelsene viser lovende resultater som tyder på at forsterkningslæring har egenskapene til å skape vellykkede handelsstrategier.

Abstract

This thesis deals with portfolio management when applying reinforcement learning. This machine learning process will solve issues related to trading portfolio management by using an agent who learns through sequential decision-making processes. The learning process of the agent is manufactured in an environment that represents the financial market. Model-free reinforcement learning is used, and this allows the agent to implement actions in separate timeperiods and at the same time learn optimal action strategy. The trading system is produced through combinations of financial mathematical formulations and reinforcement learning algorithms. The thesis uses the reinforcement learning algorithm Deep Deterministic Policy Gradient. The approach to the learning process is for the agent to use "Trial-and-Error" technique to create optimal solutions. In order for this to be possible, a memory has been implemented in the system. With a memory the agent can remember what decisions have been made and learn from them in the next situation. To examine the results of the actions of the agent there will be executed back-testing on historical data. The results are then compared with the Dow Jones market index, DIA (Dow Jones ETF) and a Mean-Variance portfolio strategy to examine the performance of the reinforcement learning agent. This trading simulation checks the portfolio's robustness, profitability and risk sensitivity. The studies show promising results that suggest that reinforcement learning has the capabilities to create successful trading strategies.

Innholdsfortegnelse

1 Motivasjon

Ambisjonen med oppgaven er å skape en metode som kan anvendes for suksessfull handling i finansielle markeder. Flere studier viser til vellykkede resultater, men har en manglende fremlegging av forskningsdata. Det er også utfordrende å skaffe innsikt i de valgene og prosedyrene som gjennomføres, derfor er det også lite mulig å utnytte denne forskningen i praksis for andre. Ønsket er derfor å skape en brukervennlig funksjon som kan utnyttes av flere mennesker, uten at det forutsetter særlig høye akademiske krav til utnyttelse.

2 Introduksjon

Porteføljeforvaltning er beslutningsprosesser med kontinuerlig allokering av kapital og forvaltning av finansielle instrumenter som aksjer, obligasjoner og verdipapirer (Fabozzi et al., 2002). Der målsetningen er å maksimere avkastningen gitt en risikoposisjon (H. M. Markowitz, 1968). Porteføljer foretrekkes generelt fremfor enkeltstående posisjoner i finansielle instrumenter siden de generelt holder signifikant lavere risiko. Bevegelsene til finansielle instrumenter er varierende over tid og gjenspeiler blant annet forventninger i politiske og sosiale forhold (Black, 2013). Effekten av eksogene faktorer gjør at all offentlig og tilgjengelig informasjon vil påvirke aktivapriser (Fama, 1970). Markedenes kompleksitet gjenspeiles av volatilitet og korrelasjon som finnes mellom finansielle instrumenter (Bonanno et al., 2001). Utfordringen er å modellere påvirkningsfaktorer og videre benytte disse til predikering og forståelse av fremtidige hendelser (Bekaert & Hoerova, 2014)(Bollerslev et al., 2009).

Det er tydelig at forståelse av markedsutvikling krever stor behandlingsevne og kunnskap om overnevnte bevegelser, men også andre uforutsette forhold som kan innvirke på markedsverdier. Evnen til å identifisere komplekse mønster på en ressurseffektiv måte gjør dyplæringsteknikker attraktive for slike oppgaver (Soleymani & Paquet, 2021)(Goodfellow et al., 2016). Dette er maskinlæringsmetoder som anvender algoritmeløsninger med nevrale nettverk. Opprinnelig stammer dyplæring fra datavitenskap, men har siden blitt videreført til blant annet medisin, spill, fysikk og finans (Chai & Ngai, 2020)(Lillicrap et al., 2015)(Silver et al., 2016). Jiang et al. (2018) viser til bruken av logistisk metode og skjult Markov modell (HMM) i undersøkelser på systematiske mønster i tidsseriedata for finansiell prediksjon, men at disse metodene sliter på grunn av utfordringer med å identifisere ikke-lineære sammenhenger mellom komplekse finansmarkeder. For porteføljeforvaltning er det viktig å oppfatte disse sammenhengende, samtidig som en kan adressere problemet som ikke-lineært, stokastisk og tidsavhengig. Dette gjør at en kan fremstille problemet gjennom sekvensielle beslutningsprosesser (Zhang & Wang, 2017).

Flere studier har derfor forsøkt å introdusere maskinlæring gjennom forsterkningslæring til anvendelse på finansielle forhold (Mosavi et al., 2020)(Meng & Khushi, 2019)(Moody & Saffell, 2001)(Guo et al., 2018)(Z. Jiang et al., 2017)(Liang et al., 2018). Moody et al. (2001) benytter tilbakevendende forsterkningslæring som gir enklere og effektiv problemløsning, og unngår Bellman sin dimensjonalitetsforbannelse.[1] Som en videreføring av Moody et al. (2001) benyttet Deng et al. (2016) dype nevrale nettverk for å skape høyere avkastning. Jiang et al. (2017) produserte porteføljeforvaltning basert på "Deep Deterministic Policy Gradient" (DDPG) og utkonkurrerte tradisjonelle porteføljestrategier. Resul-

[1] Det er ikke klart om dette utrykket kommer fra Bellman, men er hvertfall nevnt i (Bellman, 1966).

tatet av disse studiene har vært oppløftende, men har noen mangler som gjøres gjeldende ved reel praksis. Guo et al. (2018) tillater handel av eiendeler på tvers av aktivaklasser, og Moody et al. (2001) anvender kun én aktivatype, men begge studiene har valgt å ikke innføre transaksjonskostnader. For Jiang et. al (2017) ble det innført transaksjonskostnader, men ikke tilstrekkelig tatt hensyn til mangelfull data i kryptomarkeder. Dette er noe som vil tas med i vurderinger og etablering av forsterkningslæringsmetoden som anvendes i denne oppgaven.

Prinsipielt drives forsterkningslæringen gjennom opplæring av en kontrollør som videre betegnes som agenten. Agenten gjennomfører handelsbeslutninger fra et mulig handlingsområde i et konstruert miljø, med intensjon om å maksimere porteføljeavkastning. Denne målsetningen kan utformes som et stokastisk optimeringsproblem. Det finnes flere forsterkningslæringsalgoritmer som kan benyttes til et slikt optimeringsproblem. "Q-learning" (Watkins & Dayan, 1992)(Neuneier, 1997) er en "off-policy" algoritme, som betyr at den lærer verdien av den optimale handelspolitikken uavhengig av den handelspolitikken som er tilknyttet agentens handlinger. "Deep Q Network (Mnih et al., 2013)(Mnih et al., 2015) er en videreføring av Watkins og Dayan (1992) og Neuneier (1997) som benytter et "Q-learning"-rammeverk med nevrale nettverk. Anvendelsen av dype nevrale nettverk skaper god representasjon av data som kan effektivisere forsterkningslæringsalgoritmer (Mnih et al., 2016). "Deterministic Policy Gradient" (DPG) er en "off-policy" aktør-kritiker algoritme som lærer en deterministisk handelspolitikk gjennom handlingsbeslutninger (Silver et al., 2014). Algoritmen har en bedre effekt på høydimensjonale handlingsområder sammenlignet med stokastiske metoder. Alle de overnevnte algoritmene er modellfrie, siden de ikke skaper en illustrasjon av miljøet. Motparten til dette ville vært modellbaserte algoritmer, eksempelvis SARSA (Zhao et al., 2016), men nøyaktig

konstruksjon og predikering av markedsutvikling er utfordrende og kan gi negativ innvirkning på agentens evne til problemløsing (Filos, 2019).

Denne oppgaven foreslår modellfri forsterkningslæring som tar hensyn til transaksjonskostnader for å lære handelsagenten å optimere porteføljeavkastning i henhold til agentens risikovilje. Derfor er det besluttet å anvende Deep Deterministic Policy Gradient (DDPG) (Lillicrap et al., 2015). Dette er en modellfri, "off-policy" tilnærming som lærer en deterministisk handelspolitikk i et kontinuerlig handlingsområde. Algoritmen er godt egnet til porteføljeforvaltning siden den har muligheten til å oppfatte handlingsområdet samtidig som metoden lærer en deterministisk handlingspolitikk.

Videre er oppgaven organisert gjennom en problemdefinisjon som forklarer de gjeldende forholdene tilknyttet porteføljeforvaltning, og matematiske formuleringer som benyttes til problemløsning. I datahåndteringen vises det hvilke korrigeringer som må gjennomføres slik at datasett er tilstrekkelig til anvendelse av forsterkningslæring. Dette etterfølges av porteføljeoptimeringsteori før en dypere forklaring av forsterkningslæring og nettverksoppbygging. Oppgaven avsluttes med resultatene fra eksperimenter og påfølgende konklusjon.

3 Definisjon av problem

Porteføljeforvaltning er som nevnt prosesser med kontinuerlig omdisponering av kapital til ulike finansielle eiendeler. Arbeidet skal undersøke effektiviteten av forsterkningslæring i porteføljeforvaltning, og om det er mulig å oppnå optimal problemløsning. De anvendte finansielle instrumentene begrenses til aksjemarkedet, med forhåndsutvalgte instrumenter som utgjør mulighetsområdet for porteføljen og et fastsatt budsjett tilgjengelig til handel ved etablering.

Alle transaksjoner gjennomføres med kontanter. Det vil ikke være mulig å kjøpe eiendeler uten kontanter eller selge uten å være i besittelse av eiendelene som selges. Dette betyr at porteføljen inneholder kun long-posisjoner, slik at investorer ikke kan tape kontanter utover investeringen. Porteføljeverdier vil oppdateres daglig, med endrede verdier for åpning og stenging basert på de individuelle kurssvingningene til hver eiendel. Ved slutten av en handelsdag vil det gjennomføres en vektfordeling i investeringer med fastsettelse av ny porteføljeverdi, som vil fremstå slik frem til åpning neste handelsdag. Det benyttes også forutsetninger om at én eiendel i porteføljen kun omsettes en gang hver handelsdag. Som gjør at det ikke er mulig å kjøpe og selge samme eiendel innen én handelsdag. For å fastsette intensjonen med oppgaven skal det videre i avsnittet fremstilles konkrete matematiske formuleringer av problemstillinger tilknyttet porteføljeforvaltning.

På neste side fremstilles en oversikt for de begrepene som benyttes til matematiske formuleringer i dette avsnittet.

Parametere	
a_t	Handlinger ved slutten av en tidsperiode.
s_t	Tilstand ved slutten av en tidsperiode.
m	Antall eiendeler i porteføljen.
v_t	Prisvektoren for en tidsperiode, som er sluttkursen for alle eiendeler.
	Åpningskursen for tidsperiode, $t+1$ i kontinuerlige markeder.
	Sluttkursen for tidsperiode, t i kontinuerlige markeder.
w_t	Porteføljevekten ved slutten av en tidsperiode.
$w_t^{'}$	Porteføljevekten ved slutten av en tidsperiode før påførte transaksjonskostnader.
y_t	Relativ prisvektor ved slutten av en tidsperiode.
p_t	Porteføljeverdien ved slutten av en tidsperiode.
$p_t^{'}$	Porteføljeverdien ved slutten av en tidsperiode før påførte transaksjonskostnader.
μ	Transaksjonskostnader ved handel av eiendeler.
ρ_t	Avkastningsrate ved slutten av en tidsperiode.
r_t	Logaritmisk avkastning ved slutten av en tidsperiode.
p_f	Endelig porteføljeverdi
$w_{i,t}$	Andel av eiendelen i i porteføljen ved slutten av en tidsperiode.
$v_{i,t}$	Sluttkursen for en eiendel i i en tidsperiode.
$v_t^{(hi)}$	Høyeste kurs for en tidsperiode.
$v_t^{(lo)}$	Laveste kurs for en tidsperiode.
$\odot$	Elementvis multiplikasjon.
$\oslash$	Elementvis divisjon.
C_o	Oppgang i kontanter.
C_n	Nedgang i kontanter.
c_s	Provisjonssats for salg av eiendeler.
c_k	Provisjonssats for kjøp av eiendeler.
$ReLU$	Elementvis rettet lineær enhet.
Indekser	
i	Indeks for individuell aktiva i porteføljen.
t	Indeks for tidsperiode.
T	Indeks som symboliserer hele tidsperspektivet.

Figure 1: Oversikt for notasjoner i problemdefinisjon

3.1 Matematiske formuleringer

De matematiske formuleringene og porteføljemodellen er inspirert av tidligere studier (Ormos & Urbán, 2013)(Z. Jiang et al., 2017)(Soleymani & Paquet, 2021). Metoden skal sikre en tilnærmet optimal vekstrate for investeringer (Soleymani & Paquet, 2021). Anvendelsen av Bellman-ligningen gjør det mulig å finne optimal løsning (Bellman, 1957a)(Bellman, 1957b).

Finansmarkedet består av et definert antall eiendeler, m. Porteføljen konstrueres av disse eiendelene og endres i henhold til handlinger begått av investorer gjennom en periode, t. I porteføljestyring er det ingen antagelser om vektfordeling av kapital. Som gjør at porteføljen starter med en posisjon utelukkende i kontanter. Vektfordelingen til kontanter er dermed lik 1, mens resten av vektene er lik 0. Grunnposisjonen for vektoren til porteføljevekten, w_0 blir valgt til å være den første basisvektoren i det euklidske rommet,

$$w_0 = (1, 0, ..., 0)^T \tag{1}$$

For andre posisjoner der investorer har gjennomført handlinger ved slutten av en periode t gjelder følgende porteføljevekt, w_t:

$$w_t = (w_{0,t}, w_{1,t}, w_{2,t}, ..., w_{i,t})^T \tag{2}$$

Sluttkursene for alle eiendeler i porteføljen i en periode utgjør prisvektoren, v_t samt mengden av tilgjengelig fri kapital.

$$v_t = (v_1, v_2, ..., v_m)^T \tag{3}$$

Det vil si at, $v_{i,t}$ er sluttkursen for en spesifikk eiendel, i for en periode t. I kontinuerlige markeder er prisvektoren, v_t sluttkursene for periode t, men også åpningskursene for periode $t+1$. Som en videre forutsetning tillates det ikke negativ vekting i prisvektoren, v_t og dermed er short-salg forbudt. Videre kan det formuleres en relativ prisvektor, y_t basert på elementvis divisjon ($\oslash$) av v_t på v_{t-1}:

$$y_t : v_t \oslash v_{t-1} = [1, \frac{v_{1,t}}{v_{1,t-1}}, \frac{v_{2,t}}{v_{2,t-1}}, \cdots, \frac{v_{m,t}}{v_{m,t-1}}]^T \tag{4}$$

Ligning (4) kan brukes til å beregne endringen i porteføljeverdi gjennom en periode, hvis porteføljeverdien er, p_{t-1} i begynnelsen av periode t og en ignorerer transaksjonskostnader:

$$p_t = p_{t-1} y_t \cdot w_{t-1} \tag{5}$$

w_{t-1} er vektoren for porteføljevekting i begynnelsen av periode t. En slik formulering av porteføljeverdien er derimot ikke forenlig med det denne oppgaven ønsker å gjennomføre. Dersom en ignorerer transaksjonskostnader distanseres metoden fra virkeligheten og reliabiliteten forsvinner (Ormos & Urbán, 2013). Realiteten i et virkelig scenario er at det påføres kostnader for kjøp og salg av eiendeler i aksjemarkedet. Innføringen av transaksjonskostnader gjør porteføljeforvaltning til et flertrinns beslutningsproblem (Neuneier, 1996). Slike beslutningsproblemer vektlegger alle tilstands- og handlingssekvenser, og fokuserer på de sekvensielle belønningene som følge av agentens handlinger.

Dersom det forutsettes en fast provisjonssats for transaksjonskostnader, vil porteføljeverdien i ligning (5) endres. Dette gjør at porteføljevektingen i slutten av en periode fremstår slik:

$$w_t^{'} = \frac{y_t \odot w_{t-1}}{y_t \cdot w_{t-1}} \tag{6}$$

der $\odot$ er den elementvise multiplikasjonen og w_{t-1} er vektoren for porteføljevekting i begynnelsen av periode t. Videre belastes gjennomførte transaksjoner med gebyrer $\mu \in (0, 1]$, som gir porteføljevekt, w_t og denne vil være gjeldende ved begynnelsen av neste periode.[2] Med denne porteføljevekten blir porteføljeverdien i ligningene (7) og (8), med p_{t-1} som porteføljeverdi i begynnelsen av periode t og $p_t^{'}$ på slutten før transaksjonskostnader:

$$p_t = \mu_t p_t^{'} \tag{7}$$

Den endelige porteføljeverdien ved slutten av perioden, altså p_t er dermed produkt av porteføljeverdien etter alle handlinger er gjennomført $p_t^{'}$ samt de kostnadene, μ_t som er påført i perioden.

$$p_t = \mu_t p_{t-1} \cdot y_t \cdot w_{t-1} \tag{8}$$

Porteføljevekten, w_t kan alltid summeres til 1 ved,

$$\sum_i w_{i,t} = 1 \forall t \tag{9}$$

[2]Figur vedlagt med illustrasjon av transaksjonskostnaders effekt på porteføljen i vedlegg A.1.

Innføringen av transaksjonskostnader leder til følgende avkastningsrate og logaritmisk avkastning:

$$\rho_t = \frac{p_t}{p_{t-1}} - 1 = \frac{\mu_t p_t}{p_{t-1}} - 1 = \mu_t y_t \cdot w_{t-1} - 1 \tag{10}$$

$$r_t = \ln \frac{p_t}{pt-1} = \ln(\mu_t y_t \cdot w_{t-1}) \tag{11}$$

og den endelige porteføljeverdien hvor p_0 er første investering blir:

$$p_f = p_0 \exp(\sum_{t=1}^{t_f+1} r_t) = p_0 \Pi_{t=1}^{t_f+1} \mu_t y_t \cdot w_{t-1} \tag{12}$$

For investorer er målsetningen å maksimere p_f over et definert tidsrom. De matematiske formuleringene vil løse problemstillingen så lenge vi forklarer den siste faktoren, μ_t. Løsningen for å formulere denne kommer gjennom kjøp og salg av eiendeler, som da medfører henholdsvis nedgang, C_n eller økning, C_o i kontanter.

$$C_o = ((1 - c_s))p_t^{'} \sum_{i=1}^{m} \cdot ReLU(w_{t,i}^{'} - \mu_t w_{t,i}) \tag{13}$$

$$C_n = (1-c_k)[w_{t,o}^{'} + (1-c_s) \sum_{i-1}^{m} ReLU(w_{t,i}^{'} - \mu_t w_{t,i}) - \mu_t w_{t,0}] = \sum_{i-1}^{m} ReLU(\mu_t w_{t,i} - w_{t,i}^{'}) \tag{14}$$

$0 \leq c_s$ ¡ 1 er provisjonssatsen for salg og $0 \leq c_k$ ¡ 1 er provisjonssatsen for kjøp. ReLU(x) = max(0, x) er en elementvis rettet lineær enhet. Med dette blir tilgjengelig kontanter endret fra $p_t^{'} w_{t,0}^{'}$til $\mu_t p_t^{'} w_{t,0}$. Da får en muligheten til

å løse μ_t ved å forenkle (14) på følgende måte:

$$\mu_t = \frac{1}{1 - c_k w_{t,0}} [1 - c_k w_{t,0}^{'} - (c_s + c_k - c_s c_k) \sum_{i-1}^{m} ReLU(w_{t,i}^{'} - \mu_t w_{t,i})] \quad (15)$$

$$\mu_t = \mu_t(w_{t-1}, w_t, y_t) \quad (16)$$

Tilstedeværelsen av μ_t inne i en lineær likerett betyr at μ_t ikke kan løses analytisk. Jiang et al. (2017) har derimot fremstilt en iterativ løsning.

3.2 Betingelser

Etter agentens opplæringsperiode gjennomføres back-testing på testdata, for å måle ytelsesevnen til metoden. Ved å introdusere noen markedsbetingelser kan back-testing enklere gjennomføres. Disse betingelsene er realistiske for aktive markeder.

Tilstrekkelig likviditet

Eiendeler betegnes som likvide dersom de enkelt kan konverteres til kontanter på kort tid. En av betingelsene er at alle finansielle instrumenter er likvide, og at enhver transaksjon kan gjennomføres under denne betingelsen.

Null glidning

Glidning viser til den forskjellen som oppstår mellom forventet pris ved en transaksjon og den faktiske prisen ved transaksjonens gjennomførsel. Med dette så kommer betingelsen om at markedet skal være tilstrekkelig likvid til at

transaksjoner skal gjennomføres det øyeblikket de blir iverksatt, og prisen ved det tidspunktet.

Ingen markedspåvirkning

Aktivapriser påvirkes av forhold i markedet, kombinasjonen av tilbud-etterspørsel gjennom markedssentiment og transaksjoner faller inn under denne påvirkningen. Denne betingelsene krever derimot at transaksjonene som gjennomføres av agenten ikke skal påvirke markedet, siden investeringene har en lav verdi som ikke vil påvirke markedet.

4 Datahåndtering

Kvaliteten av data i analytiske prosesser har stor påvirkning på resultatene. Dersom feilinformasjon komponeres i metoden, vil kvaliteten synke gjennom unøyaktige analyser og påvirke videre handlingsstrategier. For å unngå slike situasjoner er det essensielt å kvalitetssikre og evaluere datainformasjon. For denne oppgaven benyttes utelukkende datainformasjon fra aksjemarkedet, og videre i dette avsnittet vil det presenteres hvordan datahåndteringen er gjennomført. Det vil fremstilles informasjon om modifisert datasett, med de korrigeringer som er anvendt med hensikt å styrke metoden.

4.1 Datasett

Tilgjengeligheten på datakilder for finansielle markeder er god, og særlig tilknyttet aksjemarkeder. Problemet er at flere av disse tjenestene beskytter eller begrenser datatilgjengelighet gjennom avgifter, lisensiering, osv. Dette senker

mulighetene til å fremstille mer avansert og nyansert data. Heldigvis finnes det fortsatt offentlig tilgjengelig data med høy kvalitet, men noe mindre detaljert som potensielt krever modellering og bearbeidelse før anvendelse i modeller. Yahoo! Finance API har databaser for flere av de største aksjeindeksene, med datainformasjon som er basert på OHLC samt handelsvolum.[3] Innholdet dekker altså åpningskurs, høyeste, laveste og sluttkurs, i tillegg til handelsvolum for hver individuelle aktiva for hver handelsdag. For at det skal være mulig å gjennomføre forsterkningslæring med stort antall interaksjoner mellom agent og miljø krever det tilstrekkelig mengde data. Dette utelukker flere indekser, og gjennom undersøkelser er det kommet frem til at S&P 500 og Dow Jones fremstår som de mest komplette og robuste indeksene.

Videre brukes Dow Jones Industrial Average (DJIA), en aksjeindeks basert på 30 store børsnoterte selskaper fra det amerikanske aksjemarkedet. Datasettet hentes fra Yahoo! Finance API, og strekker seg fra 2002 til 2022. Eksperimentene er gjennomført basert på historisk data på forskjellige tidsperioder innen datasettet. Trenings- og testperioder har varierende lengde for å undersøke forsterkningslæringens ytelsesevne under forskjellige forhold. Mengden data må som nevnt være tilstrekkelig til å trene agenten. Indeksen inneholder kun 30 individuelle aksjer, men disse har til gjengjeld tilstrekkelig med data til å gjennomføre problemløsing. På denne måten unngås tilfeller hvor enkelte selskaper ikke oppfyller krav om tilgjengelighet gjennom både trenings- og testperiode. I tillegg har ikke disse selskaper store prisfall eller prisøkninger uten særlig årsakssammenheng, som ellers ville blitt fjernet fra data.

[3] "OHLC" står for "Open, High, Low, Close" og er standardformat til å illustrere prisutvikling i finansielle markeder.

Benyttelsen av disse 30 aksjene til å utgjøre eiendelene i porteføljen bidrar til å senke belastningen og ressurskravet til metoden. Dette vil være de eneste eiendelene i miljøet som agenten skal behandle. Ved å speile aksjene fra Dow Jones er det også mulig å få direkte sammenligningsgrunnlag i evaluering av resultatene. Selskapene er valgt fordi de vil være identiske med DIA, et børsnotert fond som følger Dow Jones. Det finnes muligheter for å implementere mer datainformasjon for å tydeliggjøre egenskaper og optimere data, men prinsippet med forsterkningslæring er at systemet skal lære gjennom miljøet uten perfekt informasjon. Oversikten over tilgjengelige aksjer kan sees i tabellen nedenfor.

Markedssektor	Selskaper
Teknologi	
	Apple Inc. (AAPL)
	Cisco Systems Inc. (CSCO)
	Intel Corp. (INTC)
	Microsoft Corporation (MSFT)
	International Business Machines Corp. (IBM)
	Verizon Communications Inc. (VZ)
	Honeywell International (HON)
	Salesforce.com Inc. (CRM)
Finans	
	American Express Co. (AXP)
	Goldman Sachs Group Inc. (GS)
	J.P. Morgan Chase & Co. (JPM)
	Travelers Companies Inc. (TRV)
	Visa Inc. (V)
Privat konsum	
	Coca-Cola Comp. (KO)
	Home Depot Inc. (HD)
	McDonald's Corp. (MCD)
	Nike Inc. (NKE)
	Procter & Gamble Co. (PG)
	Walgreens Boots Alliance Inc. (WBA)
	Walmart Inc. (WMT)
	The Walt Disney Company (DIS)
Industri	
	3M Co. (MMM)
	Boeing Co. (BA)
	Caterpillar Inc. (CAT)
	Dow Inc. (DOW)
Helse	
	Johnson & Johnson (JNJ)
	Merck & Co Inc. (MRK)
	Unitedhealth Group Inc. (UNH)
	Amgen Inc. (AMGN)
Energi	
	Chevron Corp. (CVX)

Figure 2: Utvalgte selskaper som kan utgjøre metodens porteføljе

Disse finansielle instrumentene representerer forskjellige bransjer i aksjemarkedet. Det er besluttet å velge teknologi, finans, privat konsum, industri, helse og energi som de forskjellige bransjene. Dette vil gi muligheter for å undersøke sammenhengene mellom instrumentene og muligens forklare hvilke relasjoner og forhold som oppstår mellom disse. Det at disse selskapene er representert på DIA gjør at dette børsnoterte fondet vil være en direkte sammenligning som kan benyttes i undersøkelser av resultater.[4]

Markedssektor	Prosentandel
Teknologi	26.67 %
Finans	16.67 %
Privat konsum	26.67 %
Industri	13.33 %
Helse	13.33 %
Energi	3.33 %

Table 1: Prosentfordeling i de forskjellige markedssektorene i porteføljen

4.2 Korrigering

Forsterkningslæringsalgoritmen er avhengig av kontinuerlig data, og er følsom for eventuelle feil ved data. Programpakken Yahoo! Finance er ikke feilfri, og kan derfor skape problemer for metoden. Manglende data er generelt de vanligste feilkildene som skaper problemer. Tidligere studier har introdusert en mulig løsning for problemet med manglende data for instrumenter under enkelte handelsdager der handel er stanset grunnet ekstraordinære omstendigheter. Tilfeller med manglende informasjon fremstår som "NaN", og kan ikke tolkes. For

[4]På engelsk omtales et børsnotert fond som "ETF"; "Exchange Traded Fund".

å korrigere for dette settes handelskursen til sluttkursen av nærmeste tilhørende handelsdag, samtidig som handelsvolumet settes til 0. Intensjonen med dette er at datasettet vil være anvendelig for metoden og kan skilles fra vanlige datapunkter (Kang et al., 2018). Tilsvarende prinsipp anvendes også for tilfeller ved andre datoer som ikke er handelsdager, og dermed kan agenten oppfatte at markedet er stengt.

4.3 Handelsperiode

Algoritmen som er benyttet til aktivahandel er tidsdrevet, med oppdelte tidsperioder med like lengder, t. I hver enkeltperiode omfordeles eiendeler i ulike aktiva av handlingsagenten. Det er besluttet hensiktsmessig å benytte tidsperiode på $t = 1$ handelsdag i alle eksperimenter, naturlig tatt i betraktning at handling av aksjer foregår med faste strukturer innenfor disse handelsdagene. Gjennom en periode vil aktivapriser variere, men fire viktige prispunkter karakteriserer periodens samlede bevegelse, nemlig åpnings-, høyeste-, laveste- og sluttkursene. Det er tidligere nevnt at for kontinuerlige markeder er åpningskursen på et finansielt instrument i en periode sluttkursen fra forrige periode. Det antas dermed at eiendeler i begynnelsen av en periode er tilgjengelig til åpningskursen i gitt periode.

4.4 Pristensor

Tensorer beskriver data eller fysiske enheter i n-dimensjoner. De beskriver et multilineært forhold mellom flere sett av algebraiske objekter relatert til et vektorrom (Pan, 2014). Anvendelse av tensorer gjøres for å skaffe naturlig representasjon av tidsseriedata. Den skal vise markedsfunksjoner og tekniske indikatorer for de utvalgte eiendelene. Andre metoder som transformerer data fra tensor til

vektor risikerer å miste informasjon relatert til tid og rom. Nevrale nettverk som CNN og RNN lærer tidsdata direkte som tensorer, og kan forklare årsakene til at disse metodene har vært suksessfulle ved tidligere studier (Tran et al., 2017).[5]

Prinsippet med pristensoren er at den produserer en portfeføljevektor gjennom å lede informasjon fra datasettet inn i et nevralt nettverk. Pristensoren, X_t med form (f, n, m) er altså da tilført data på slutten av en periode t. Der m er antall eiendeler i porteføljen, n er antall tilførselsperioder med data før periode t og f representerer antallet prisegenskaper. Eiendeler i porteføljen er tidligere fastsatt, $m = 30$. For å utvelge n vurderes effektene av tidligere priser, som synker med tiden. Det vil si at prisen fra ett år tilbake har mindre effekt en forrige ukes pris. Med dette settes $n = 60$, omtrentlig to måneder. Prisegenskapene for hver individuelle eiendel blir $f = 3$ og kommer av sluttkursen for en eiendel, $v_{i,t}$, samt henholdsvis høyeste og laveste kurs; $v_{i,t}^{(hi)}$og $v_{i,t}^{(lo)}$. Anvendelsen av 3 tekniske indikatorer komponerer effekter mer effektivt som vil hjelpe agentens læringsprosess, til forskjell fra mer ressurskrevende metoder (Park et al., 2020)(Huang et al., 2020). I avsnitt (3) har vi funnet at porteføljen varierer som en følge av prisendringer mellom perioder. Det gjør at prisegenskapene slik de er konstruert ikke vil gjøre nytte for det nevrale nettverket. Disse prisene er absolutte og må derfor normaliseres med hensyn på siste sluttkurs for å benyttes. Dette gjør også at individuelle priser for åpning ikke blir gjeldende for pristensoren og at vi kun opererer med; $v_{i,t}$ som en representasjon av prisen for slutten av periode t. Pristensoren er en stabling av de tre normaliserte prismatrisene, $V_t, V_t^{(hi)}$ og $V_t^{(lo)}$ hvor radene viser til hver spesifikke eiendel og kolonnene viser til de tekniske indikatorene i gitt tidsrom.

[5]CNN står for "Convolutional Neural Network", mens RNN står for "Recurrent Neural Network".

$$V_t = [v_{t-n+1} \oslash v_t | v_{t-n+2} \oslash v_t | ... | v_{t-1} \oslash v_t | 1] \tag{17}$$

$$V_t^{(hi)} = [v_{t-n+1}^{(hi)} \oslash v_t | v_{t-n+2}^{(hi)} \oslash v_t | ... | v_{t-1}^{(hi)} \oslash v_t | v_t^{(hi)} \oslash v_t] \tag{18}$$

$$V_t^{(lo)} = [v_{t-n+1}^{(lo)} \oslash v_t | v_{t-n+2}^{(lo)} \oslash v_t | ... | v_{t-1}^{(lo)} \oslash v_t | v_t^{(lo)} \oslash v_t] \tag{19}$$

hvor $\oslash$ symboliserer elementvis divisjon. Porteføljevekten ved slutten av perioden skapes av pristensoren X_t, forrige periodes porteføljevekt w_{t-1} og handlingspolitikken π;

$$w_t = \pi(X_t, w_{t-1}) \tag{20}$$

Figur (3) viser oppbygningen av pristensoren.

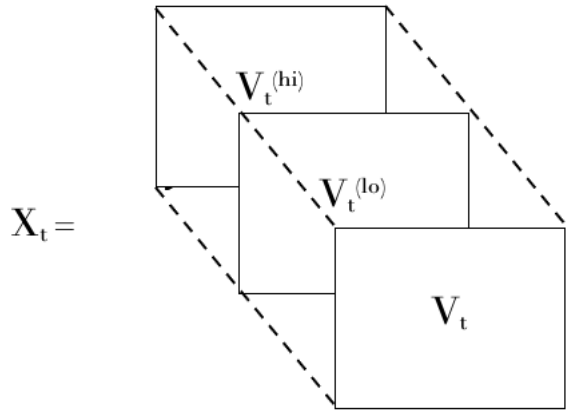


Figure 3: Pristensor med tre stablede tekniske indikatorer. Boksene som indikerer de tekniske indikatorene skal representere eiendeler i portefølje (horisontal kantlinje) og størrelsen på inndata (vertikal kantlinje).

5 Porteføljeoptimering

Porteføljeforvaltning er tidligere definert i avsnitt ovenfor, der det har vært fremlagt informasjon om hvordan de matematiske formuleringene danner utgangspunktet for de egenskapene og faktorene som inngår i porteføljer. Risiko er essensielt i vurderinger av finansielle posisjoner. Med porteføljeallokering kan en kontrollere og vurdere risiko i større grad sammenlignet med posisjoner i enkeltaktiva. Dette tillater investorer å kombinere egenskaper av enkeltaktiva for å fremheve de positive delene av markedet, samtidig som en senker de negative effektene. Investorer er ikke garantert suksess selv om de fordeler investeringer i forskjellige eiendeler, men en slik strategi kan bidra til å dempe store svingninger i avkastning (H. Markowitz, 1952)(Fabozzi et al., 2002). Det er viktig å påpeke at porteføljens oppbygning og effekten av aktivaallokering påvirker resultatene.

For å skape tilfredsstillende resultater kommer en til begrepet, porteføljeoptimering som ønsker å løse allokeringsproblemer gjennom systematisk metode. Optimalisert portefølje fremstilles gjennom en objektiv funksjon, som reflekterer preferanser hos investor med hensyn til porteføljevektoren. Dette avsnittet skal videre forklare porteføljeoptimering og hvordan en skal håndtere de tilknyttede problemstillingene.

5.1 Markowitz-modellen

Markowitz-modellen (1952) har vært ledende innen porteføljeteori og formulerer et matematisk porteføljefordelingsproblem. Moderne porteføljeteori er i hovedsak et investeringsrammeverk for utvelgelse og konstruksjon av investeringsporteføljer basert på maksimering av forventet avkastning av porteføljen og samtidig min-

imering av investeringsrisiko (Fabozzi et al., 2002). Prinsippet omhandler å finne en porteføljevektor, w blant m eiendeler, i henhold til bytteforholdet mellom avkastning og risiko. Formuleringen av modellen gir den optimale porteføljevektoren (porteføljebalansen), w_* som minimerer volatilitet for et gitt avkastningsnivå.

I prinsippet så fremstår dette som oversiktlig og grunnleggende for å finne optimal porteføljebalanse, men flere aspekter spiller en rolle. Eksemplifisert ved argumenter om at finansmarkedene vender tilbake til gjennomsnittet (mean-reversion), og at tidligere vinner dermed kan fremstå som tapere fremover og motsatt for tidligere tapere (Gilli et al., 2019). Dette vil være en av flere elementer som må vurderes i porteføljens utvalgsprosesser. For at metoden skal være relevant for virkelig handel er det også gunstig med begrensninger, som blant annet transaksjonskostnader og andelsbegrensning på eiendeler i porteføljen. Slike andelsbegrensninger vil senke risikoen av å være tungt investert i ett selskap eller én sektor. Involvering av begrensninger vil ha positiv effekt på styrken av metoden hvor empiriske tester kan vise relevansen for fremstilte begrensninger, og hvorvidt de er egnet til metoden. En metode som kan løses gjennom klassisk tilnærming er mean-variance optimering.

5.1.1 Mean-Variance optimalisering

Metoden har til hensikt å maksimere avkastningen for et gitt risikonivå. Forventet avkastning, η formuleres gjennom estimater på følgende måte: $\eta = [\eta_1, \eta_2, ..., \eta_M]^T \in \mathbb{R}^M$. Porteføljevariansen formuleres som $\sigma^2 = w^T \sum w$, og forventet avkastning er $\eta = w^T \eta$. Ved et fastsatt forventet avkastningsmål

$\eta_{mål}$, bestemmes porteføljevektoren $w \in \mathbb{R}^M$ ved:

$$minimere_w \qquad \frac{1}{2} w^T \sum w \tag{21}$$

$$forutsatt \qquad w^T \eta \geq \eta_{mål} \tag{22}$$

$$og \qquad e^T w = 1 \tag{23}$$

$$og \qquad w \succeq 0 \tag{24}$$

hvor e betegner kolonnevektoren av første orden, altså hver av komponentene av e er lik 1.[6] For å sørge for at short-salg forbys settes begrensningen $w \succeq 0$ inn, der $\succeq$ betegner en elementmessig ulikhetsoperator.

Ved å bruke Lagrange multiplikatorer ønsker vi å finne det punktet hvor funksjonens gradient peker i samme retning som begrensningenes gradient, samtidig som begrensningene tilfredsstilles. Med dette finner vi et ekstrempunkt. Gradienten er vektoren som samler partiell førstederiverte av alle funkjoner i et område. For Lagranges multiplikatorer $\gamma, \lambda \in \mathbb{R}$, vil Lagrange-funksjonen minimeres slik at:

$$\mathcal{L}(w, \gamma, \lambda) = \frac{1}{2} w^T \sum w - \gamma(w^T \eta - \eta_{mål}) - \lambda(e^T w - 1) \tag{25}$$

Foruten begrensning (24) kunne problemet blitt løst gjennom matriseinversjon, men denne begrensningen gjør at det ikke er mulig å løse gjennom et sett med lineære ligninger (Filos, 2019). Dette er nå en kvadratisk funksjon og kan løses numerisk ved hjelp av gradientbaserte algoritmer (Moré & Toraldo, 1989)(Gill et al., 2019).

[6] På engelsk anvendes begrepet "vector of ones" for å beskrive e.

5.1.2 Dynamisk porteføljeoptimering

Interaksjonen mellom agent og miljø skal gjennom forsterkningslæring resultere i optimalisert belønningsprosess. Dynamisk porteføljeoptimering har flere fellestrekk til disse prosessene. Investorer skal allokere $w_t \in \mathbb{R}$ på hver investerte eiendel for avkastning, r_t for å maksimere forventet nytte, U av fremtidig kapital:

$$max\mathbb{E}_{w_t}[U(\sum_{t=0}^{T-1} PnL_{t,t+1})] \tag{26}$$

gitt de begrensningene som fremstilles. PnL (Profit and Loss) viser forholdet mellom positiv og negativ avkastning over en periode t. På samme måte kan nyttemaksimering formuleres:

$$max\mathbb{E}_{w_t}[\sum_{t=0}^{T-1} w_{t+1}r_{t+1} - Tap(|w_{t+1} - w_t|) - Risiko(w_{t+1})] \tag{27}$$

Risikobegrepet kommer av nyttefunksjonen ovenfor og fungerer som en straff for risikable porteføljevekter, w_t. For å sammenkoble denne teorien med forsterkningslæring vil gjennomførte transaksjoner mellom perioder håndteres som handlinger, a_t slik at $a_t = w_{t+1} - w_t$ og derav $w_{t+1} = a_t + w_t$.

$$max\mathbb{E}_{w_t}[\sum_{t=0}^{T-1} w_{t+1}r_{t+1} - Tap(|a_t|) - Risiko(w_{t+1})] \tag{28}$$

Med dette er problemet omgjort fra dynamisk porteføljeoptimering til forsterkningslæring. På samme måte som ved avsnitt (5.1.1) vil dette kreve algoritmisk problemløsning.

6 Forsterkningslæring

Grunnprinsippene i forsterkningslæring ble forklart i introduksjonen, hvor intensjonen er å maksimere målsetningen. I dette tilfellet gjennom opplæringen av agenten i et stokastisk optimeringsproblem som skal sørge for maksimering av porteføljeavkastningen. Agenten iverksetter en handling, a_t fra et handlingsrom A basert på tilstanden i miljøet, s_t som tilhører et tilstandsrom S. Hver handling kan oppfattes som en interaksjon med miljøet, og som videre resulterer i en belønning r. I et kontinuerlig sekvensielt handlingsrom vil belønningen ankomme forsinket (Wiering & Van Otterlo, 2012). Det vil si at gjennom denne oppgaven vil det være ved neste tidsperiode og belønningen for en avgjørelse ved t vil komme ved $t + 1$. Funksjonen i forsterkningslæringen skal bestemme en optimal rute i handlingsrommet for å øke belønningen. Prosessen er sekvensiell og gjennomføres i alle tidsperioder, t. De gjentagende handlingsprosessene avledes av en handlingspolitikk (policy), som agenten skal lære.

I dette kapitlet introduserer vi elementene som utgjør forsterkningslæring, og forklarer hvordan algoritmer fungerer og skal benyttes til opplæring av metoden.

6.1 MDP

Markov beslutningsprosess (MDP) er en sekvensiell modell som behandler beslutningstaking i diskrete og stokastiske miljøer (Sutton & Barto, 2018). Modellen formulerer et miljø som endrer tilstand som respons på agentens handlinger, denne tilstandsovergangen påvirker den umiddelbare belønningen samt

sannsynligheten for fremtidige tilstandsoverganger. Agenten skal velge handlinger som skaper størst akkumulert verdi av langsiktig målsetning (Littman, 2001). Siden optimeringsproblemet i porteføljeforvaltning er stokastisk, vil MDP være diskret stokastisk hvor sekvensiell maksimering gjøres gjennom belønninger i en Bellman-ligning. Ved tilfeller med kjent belønningsfunksjon og en deterministisk MDP kan Bellman-ligningen løses ved dynamisk programmering (Sutton, Barto, et al., 1998). Dette er betingelser som ikke godtas i dette tilfellet, hvor fremtidige tilstander og avkastninger er ukjent og derfor må forsterkningslæring anvendes til problemløsning.

For at det skal være mulig å benytte MDP til finansiell anvendelse kreves det noen betingelser som forklarer begrensninger og overgangsfaser, disse er formulert gjennom ligninger (2, 9, 8, 6) ovenfor. Ligning (2); $w_t = (w_{0,t}, w_{1,t}, w_{2,t}, ..., w_{i,t})^T$ viser porteføljevekten etter agentens beslutninger ved slutten av periode t. Videre forklarer ligning (9); $\sum_i w_{i,t} = 1 \forall t$ at de samlede vektene, w_t ikke overskrider totalverdi på 1 i denne perioden. For ligning (8); $p_t = \mu_t p_{t-1} \cdot y_t \cdot w_{t-1}$ fremstilles den nye porteføljeverdien og ligning (6); $w_t^{'} = \frac{y_t \odot w_{t-1}}{y_t \cdot w_{t-1}}$ viser den nye porteføljevektingen som en følge av endringer i porteføljeverdi.

Forsterkningslæringsmetoden anvender MDP med kontinuerlige tilstandsrom og handlingsrom for å håndtere handelssituasjonene. Fastsettelsene av tilstandsrom og handlingsområde gjør det mulig å formulere MDP. Beslutningsprosessen er konstruert på følgende måte:

$$MDP = (S, A, P, r, \gamma, T)$$

- S er et tilstandsrom (state space).
- A er et handlingsområde (action space).
- $P : S \cdot A \cdot S \rightarrow R_+$ er en sannsynlighetsfunksjon av overgangen fra en tilstand til en annen.
- $r : S \rightarrow R$ er en avgrenset belønningsfunksjon.
- $\gamma \in (0, 1]$ er en rabattfaktor.
- T er en tidshorisont for alle beslutninger.

Tilstandsrommet utgjør en del av et flerdimensjonalt euklidsk rom, slik at $S \subseteq \mathbb{R}^{D_S}$ hvor $D_S \in N$ er dimensjonen til tilstandsrommet (Wiering & Van Otterlo, 2012). Overgangen fra en tilstand til en annen kan forklares gjennom;

$$s_{t+1} = P(s_t, a_t) + \omega_P(s_t, a_t) \tag{29}$$

der P er sannsynlighetsfunksjonen av overgangen fra en tilstand til en annen, mens ω_P er en støyvektor.

Handlingsområdet er kontinuerlig; $A \subseteq \mathbb{R}^{D_A}$ hvor $D_A \in N$ er dimensjonen til handlingsområdet. Dersom vi hadde benyttet et diskret handlingsrom ville det blitt slik; $A = \{a_1, a_2, a_3\} = \{\text{Selg, Hold, Kjøp}\}$. Belønningsfunksjonen gir forventet belønning etter handling a_t i overgangen mellom to tilstander, s_t til s_{t+1}. Med innføringen av støy vil funksjonen fremstå slik;

$$r_{t+1} = \mathbb{E}_r(s_t, a_t, s_{t+1}) + \omega_{\mathbb{E}_r}(s_t, a_t, s_{t+1}) \tag{30}$$

der $\mathbb{E}_r$ viser forventet belønningsfunksjon og $\omega_{\mathbb{E}_r}$er støyvektoren til den forventede belønningsfunksjonen. Dersom verken ω_p eller $\omega_{\mathbb{E}_r}$ er lik 0 ved alle tilfeller betyr dette at det finnes støy, og at MDP dermed må formuleres som et stokastisk problem. Derfor benyttes stokastisk MDP med kontinuerlige tilstandsrom og handlingsrom på ikke-lineære variabler. At variablene i data er ikke-lineære gir mindre garanti for suksess sammenlignet med MDP med lineære variabler, men kombinasjonen av forsterkningslæring og nevrale nettverk har gitt gode empiriske resultater i tidligere studier (Z. Jiang et al., 2017) (Tesauro, 1994) (Silver et al., 2016).

Handlingene til agenten defineres ofte av en policy $\pi_\theta : S \rightarrow A$ som er parametrisert av θ. En Q-verdi funksjon gir forventet samlet belønning ved utførelse av en handling a_t i tilstand s_t og følger policy π i fremtiden, som er

$$Q^\pi(s_t, a_t) = E_{(s_{t+1}, \sim\pi)}[\sum_{i=t}^{T} {}^i r(s_i, a_i)] \tag{31}$$

Belønningen skal reflektere nytten som agentens handling har på porteføljen. Dette er ikke nødvendigvis avhengig av positiv porteføljeavkastning, det skal også vurderes mot markedstrender. Situasjoner hvor handlingsområdet gir utelukkende positiv avkastning skal vurdere agentens beslutninger i henhold til handlingsområdet. Belønningen kommer av hvor god handlingen har vært i det handlingsområdet, men det er viktig å påpeke at belønninger også kan være negative uavhengig av markedstrender. Målsetningen til agenten er å lære en optimal policy:

$$\pi_{\theta*} = argmax_{\pi_\theta} E\pi_\theta[\sum_{i=0}^{T} {}^i r_{t+1} | s_t = s] \tag{32}$$

En viktig utfordring for porteføljeforvaltning er å balansere de forskjellige og noen ganger motstridende målene for ulike beslutningsprosesser. Porteføljeforvalternes hovedbekymringer handler om langsiktig fortjeneste, og oppfyllelse av egne avkastningskrav målt opp mot risikopreferanse. Som en del av dette kommer utførelsen av handlinger, der investorer ønsker å minimere handelskostnader i forhold til avkastning. Dette betyr blant annet at gjennomføringen av kjøp skal utføres på lave priser av forventet fremtidig verdi. Agenten skal lære optimal handelspolitikk, men vil slite med gjennomføring av læringsprosesser dersom oppgavene blir for komplisert samtidig som tidsperspektivet blir for langt. Til eksempel kan investorer anvende timer/dager/uker på bearbeidelse av informasjon før de gjennomfører handelsbeslutninger. Dette er en luksus som agenten ikke har til rådighet, og avgjørelser skal tas på noen sekunder. Dermed kan ikke datainformasjon fremstå for stort og komplekst, noe som gjør at agenten ikke får all informasjon tilgjengelig. Oppgaven blir da å anvende informasjonen som finnes og gjennom læring oppfatte underliggende sammenhenger og effekter som finnes i finansmarkedene.

6.2 POMDP

I MDP finnes det en antagelse om at agenten har full oversikt over miljøets tilstand. Dette fremstår noe urealistisk og begrenser mulighetene til å anvende modellen. Derfor finnes en delvis observerbar MDP (POMDP) som tillater prinsippene i MDP selv under tilfeller der agenten ikke har full informasjon om miljøtilstand (Kaelbling et al., 1998). De samme algoritmene som anvendes i MDP kan også anvendes i POMDP, og dermed vil formuleringer ovenfor fortsatt være gyldig. Endringen i POMDP fører til følgende modell:

$$POMDP = (S, A, P, r, , T, \Omega, Z)$$

- S er et tilstandsrom (state space).
- A er et handlingsområde (action space).
- $P : S \cdot A \cdot S ß R_+$ er en sannsynlighetsfunksjon av overgangen fra en tilstand til en annen.
- $r : S ß R$ er en avgrenset belønningsfunksjon.
- $(0, 1]$ er en rabattfaktor.
- T er en tidshorisont for alle beslutninger.
- Ω er et observasjonsområde.
- Z en en observasjonsmodell.

Med begrenset observerbarhet må agenten selv ta avgjørelser for hvilken informasjon som skal undersøkes. Forskjellen fra POMDP til MDP er at agenten nå oppfatter en observasjon $O \in \Omega$ fra det tilgjengelige observasjonsområdet; $\Omega = \{o_1, o_2, ..., o_n\}$. Tiden er diskretisert i periode t.[7] Tilgjengelige observasjoner avhenger av neste tilstand s_{t+1}, men også basert på handlinger, a_t. Observasjonsfunksjonen blir $O : S \cdot A \cdot \Omega \rightarrow [0, 1]$. Treningsprosesser vil bli mer ressurskrevende og handelsalgoritmen vil kreve at agenten tillegges hukommelse/minne for å kompensere for manglende observerbarhet (Wierstra et al., 2007). Handlingsprosesser uten minne vil med større sannsynlighet gi lavere akkumulert belønning, sammenlignet med prosesser hvor agenten kan huske gjennomførte handlinger. Ved å innføre minne kan agenten skape en handlingspolitikk som vil være nært optimal verdi i MDP (Singh et al., 1994). Vi benytter derfor POMDP med kontinuerlige tilstandsrom og handlingsrom for å håndtere handelssituasjonene.

[7] Samme som formuleringer fra avsnitt (3).

6.3 Miljø og agent

Forsterkningslæring er drevet av algoritmiske funksjoner og er egnet til å optimalisere styringen av dynamiske systemer gjennom interaksjonen mellom agent og miljø (Busoniu et al., 2017). Agenten fremstår som lederen av porteføljen som utfører handelsbeslutninger i aksjemarkedet som et modifisert miljø. Miljøet består av de utvalgte finansielle instrumentene, med generelle betingelser som ville vært tilstede i en realistisk finansiell situasjon.[8] Forbudet mot short-salg er nevnt og benyttes for å skape en metode som blant annet vil være enklere anvendbar for en større målgruppe av potensielle investorer.

Samhandlingen mellom agent og miljø baseres på signaler som sendes mellom enhetene. For en agent er det komplisert å skaffe komplett informasjon om miljøet, grunnet kompleksiteten og størrelsen av miljøet. Dette gjør tilstandsvurderinger vanskelige, som videre kompliserer agentens beslutningsprosess. Utfordringene kommer av at store deler av agentens læring kommer gjennom tolkning av informasjon. For å videre komplisere porteføljeforvaltningen vil prisutviklingen i instrumenter påvirkes av all informasjon som finnes i miljøet (Z. Jiang et al., 2017). Dette vil si at miljøtilstander påvirkes og representeres av alle tidligere investeringstransaksjoner for individuelle aksjer frem til tilstanden hvor agenten befinner seg. Datasettet dekker handelshistorikken, men det vil være utfordrende for agenten å behandle all informasjon. Ved å fordele handelshistorikk i periodevise utvalg vil agenten enklere identifisere relevant informasjon for å kartlegge tilstanden for miljøet. Det at utvalgte instrumenter utgjør miljøet bidrar også til å senke informasjonsmengden. Agenten skal dermed fokusere på høyeste, laveste og sluttkurs i de utvalgte tidsperiodene for å vurdere prisene på instrumentene i porteføljen.

[8] Finansielle instrumenter er illustrert i figur (2), avsnitt (4). Betingelsene er forklart under avsnitt (3.2)

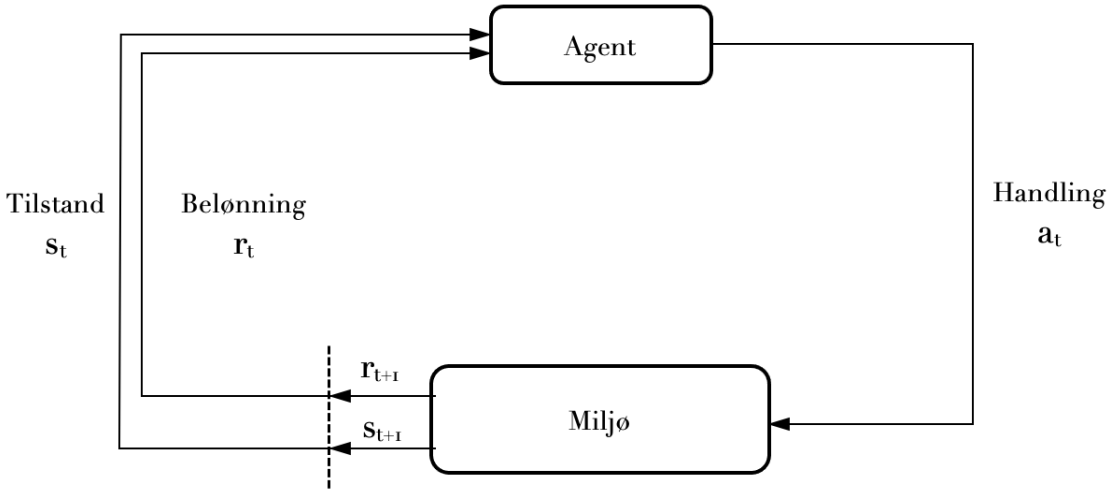


Figure 4: Enkel illustrasjon av interaksjonen mellom agent og miljø i forsterkningslæring.

6.4 Handling

Handlingen $a_t \in$ A fungerer som et signal til systemet om gjennomførte aksjoner på tidspunkt t. Handlingsområdet A viser til de potensielle beslutningene som agenten er tillatt å gjennomføre. Dette er agentens eneste mulighet til å påvirke miljøtilstand, som leder videre til forskjellige belønningssignaler fra systemet. For én eiendel har agenten beslutninger om å kjøpe, holde eller selge med henholdsvis vektorform (-1, 0, 1) gitt at begrensninger og krav tilfredsstilles.

Forutsetninger for metoden tilsier at agentens handelsbeslutning, a_t ikke påvirker priser på finansielle instrumenter, der prisene representeres av X_t. Agentens handling vil derimot ha en innvirkning på neste periodes belønning, r_{t+1}. Handlinger i en periode vil dermed påvirke belønningen i neste periode, men også

fremtidige tilstander. Transaksjoner for neste periode avgjøres av forskjellen mellom porteføljevektorene $w_t^{'}$ og w_t. I ligning (6) er $w_t^{'}$ definert med hensyn på w_{t1}, og spiller en rolle i forrige periodes handlinger. Siden w_{t1} allerede er bestemt i forrige periode, kan agentens handling ved tidspunkt t alene representeres av porteføljevektoren w_t,

$$a_t = w_t \tag{33}$$

Derfor har en tidligere handling innflytelse på beslutningen til den nåværende gjennom avhengigheten av r_{t+1} og μ_{t+1} på w_t. Forutsetningen for dette er at handlinger i periode t fullføres innen den tidsperioden.

6.5 Tilstand og observasjon

Tilstanden $s_t \in S$ indikerer situasjonen til både agenten og miljøet. Agentens manglende evne til å kartlegge komplett tilstand er forklart i avsnitt (6.2). Det finnes tilgjengelig informasjon for agenten som fremstilles hver periode t som observasjoner, $o_t \in O$. Tilstanden kan videre fordeles som et sett av, $s_t = [v_t, w_t, p_t]$ (Xiong et al., 2018). Settet inneholder tilgjengelig prisinformasjon $v_t \in \mathbb{R}_+^m$, vektbalansen i porteføljen $w_t \in \mathbb{Z}_+^m$ og porteføljeverdi $p_t \in \mathbb{R}_+$, basert på totalverdien av kontanter og aksjer. m er antallet eiendeler som vurderes for porteføljen, og $\mathbb{Z}_+$ angir ikke-negative heltall.

Miljøets tilstand

Tilstanden i miljøet s_t^e fungerer som den interne representasjonen av systemet, og benyttes til å bestemme neste observasjon o_{t+1} og belønning r_{t+1}. Miljøets tilstand er vanligvis usynlig for agenten, og selv om den er synlig, kan den in-

neholde irrelevant informasjon (Sutton, Barto, et al., 1998).

Agentens tilstand

Historikken $\overrightarrow{g_t}$ ved tidspunkt t er en sekvens av observasjoner, handlinger og tilstander opp til det tidspunktet t, slik:

$$\overrightarrow{g_t} = (o_1, a_1, r_1, o_2, a_2, r_2, ..., o_t, a_t, r_t) \tag{34}$$

Agentens tilstand s_t^a er den interne representasjonen om agentens posisjon i miljøet, for å bestemme neste handling a_{t+1} som kan være en funksjon av historikken (Sutton, Barto, et al., 1998):

$$s_t^a = F(\overrightarrow{g_t}) \tag{35}$$

Observerbarhet

Ved fullt observerbare miljøer tillates agenten å direkte observere miljøets tilstand, slik at:

$$o_t = s_t^e = s_t^a \tag{36}$$

Delvis observerbare miljøer gir indirekte tilgang til miljøets tilstand, derfor må agenten konstruere sin egen representasjon s_t^a gjennom anvendelse av historikken $\overrightarrow{g_t}$ eller et tilbakevendene nettverk:

$$s_t^a = F(s_{t-1}^a, o_t; \theta) \tag{37}$$

6.5.1 Tilstandsområde

Tilstandsområdet S viser de mulige tilstandene som agenten kan observere eller konstruere. I dagens rammeverk er innflytelsen av en tidligere handling på nåværende beslutning vist ved å vurdere w_{t1} som en del av miljøet og er lagt inn i agentens handlingspolitikk, slik at tilstanden ved t er representert ved w_{t1} og pristensoren, X_t,

$$s_t = (X_t, w_{t-1}) \tag{38}$$

der w_0 er forhåndsbestemt i ligning (1). Tilstanden s_t består av to deler, den eksterne tilstanden representert ved pristensor, X_t og den interne tilstanden representert av porteføljevektor fra forrige periode, w_{t1}. Betingelsene i avsnitt (3) viser at agentens transaksjoner har ingen markedspåvirkning, og gjør at p_t ikke er inkludert i den interne tilstanden.

6.6 Belønning

Belønningen $r_t \in \mathbb{B} \subseteq \mathbb{R}$ er et responssignal som indikerer kvaliteten av agentens beslutninger ved tidsperiode t. Den umiddelbare belønningen ved $t-1$ blir:

$$r_t(s_{t-1}, a_{t-1}) = ln(a_{t-1}y_{t-1} - \mu_{t-1}\sum_{i-1}^{m}|a_{i,t.1} - w_{i,t-1}|) \tag{39}$$

Agentens målsetning er å maksimere den samlede belønningen fra alle gjennomførte sekvensielle prosesser. På denne måten skal agenten være i stand til å ofre umiddelbar belønning for høyere langsiktig belønning, med andre ord prioritere ligning (40, 41) fremfor ligning (39). Ved finansiell anvendelse skal dermed agenten maksimere den endelige porteføljeverdien p_f for ligning (12)

ved slutten av t_{f+1}. En fullverdig belønningsfunksjon vil ikke tilfredsstilles av kun den avkastningen som oppstår fra $t = 0$ til t_{f+1}, og derfor må en involvere transaksjonskostnader og risikoen som kommer av eiendelene i porteføljen. Konstruksjonen av belønningsfunksjonen, R vil være å maksimere gjennomsnittlig logaritmisk samlet avkastning (Z. Jiang et al., 2017), siden agenten ikke har kontroll over valgene til første investering, p_0 og lengden på hele porteføljestyringsprosessen, t_f.

$$R(s_1, a_1, ..., s_{t_f}, a_{t_f}, s_{t_f+1}) := \frac{1}{t_f} \ln \frac{p_f}{p_0} = \frac{1}{t_f} \sum_{t=1}^{t_f+1} \ln(\mu_t y_t \cdot w_{t-1}) \tag{40}$$

$$= \frac{1}{t_f} \sum_{t=1}^{t_f+1} r_t \tag{41}$$

Dette gjør metoden svært attraktiv for finansiell anvendelse, særlig med tanke på de forskjellige tidshorisontene som kommer med finansielle investeringer. Metoden skal kunne gjøre det mulig å gjennomføre læring på kortere perioder over noen dager, men også gjennom uker, måneder og år.

Utfordringen ligger i det å justere systemets responssignal for belønninger. Dersom en klarer å konstruere dette vil potensialet være høyt for vellykket forsterkningslæring. Responssignalet vil påvirke læringen hos agenten i henhold til målsetningen for metoden.

6.7 Modellfri forsterkningslæring

Modellfri forsterkningslæring antar ikke en modell av miljøet, men gjennomfører avgjørelser basert på informasjonen om belønninger som mottas ved hver periode. Dette gjør at selv om agenten ikke har kunnskap og informasjon om miljøet

kan det utvikles en handelspolitikk ved prøving og feiling i gjentagende prosesser. Metoden hindrer derimot ikke agenten i å få informasjon om miljøet, men har muligheten til å drives uten informasjon. Årsaken til at det er mulig å løse problemet på denne måten er egenskapene til Bellman-ligningen, som kan løses med liten kunnskap om underliggende dynamikk så lenge det finnes tilstrekkelig datainformasjon. Denne løsningsmetoden er dermed godt egnet til miljøer som er delvis observerbar.

Mnih et al. (2015) innførte suksessfullt modellfri forsterkningslæring til anvendelse på videospill gjennom "Deep Q Network" (DQN). De nevrale nettverkene benyttes til å estimere verdifunksjonen i høydimensjonale tilstandsrom. Problemet med denne tilnærmingen er at de kun kan håndtere diskrete og lavdimensjonale handlingsrom. Dermed er ikke denne metoden egnet til finansielt arbeid hvor handlingsrommene er kontinuerlige og høydimensjonale. Diskrete handlingsrom ville ført til den tidligere nevnte dimensjonalitetsforbannelsen. Ideene fra DQN er videreført til en annen metode som kan håndtere betingelsene for finansmarkedene, DDPG.

6.8 Deep Deterministic Policy Gradient

Deep Deterministic Policy Gradient (DDPG) er en modellfri, "off-policy" forsterkningslæringsalgoritme som lærer en deterministisk handlingspolitikk i et kontinuerlig handlingsrom med anvendelse av nevrale nettverk (Lillicrap et al., 2015).[9] Agenten kjenner opprinnelig ikke til dynamikken i handelspolitikken, men skal lære denne gjennom interaksjon med miljøet. Styrken til metoden er at den kun krever enkel aktør-kritiker arkitektur og en læringsalgoritme med svært få

[9] Under vedlegg finnes det en pseudokode for DDPG i vedlegg A.3.

"bevegelige deler". Dette gjør at metoden er godt egnet til vanskeligere problemstillinger og større nettverk (Lillicrap et al., 2015).

Algoritmen har gode evner til å håndtere dimensjonalitet i kontinuerlige rom. Disse egenskapene gjør at DDPG skal anvendes til problemløsning for porteføljeforvaltning. Algoritmen kombinerer Q-Learning og Policy Gradient, der de nevrale nettverkene bygger på Policy Gradient algoritmer. En policy fungerer som databehandling (mapping) fra tilstandsområdet til handlingsområdet, $\pi : S \rightarrow A$, og kan enkelt forklares som funksjoner for agentens handelspolitikk. Ved tradisjonelle Policy Gradient algoritmer representeres policyen med en parametrisk sannsynlighetsfordeling, $\pi_\theta(a_t|s_t) = \mathbb{P}[a_t|s_t;\theta]$ som stokastisk velger handling, a_t i tilstand, s_t i henhold til parametervektor, θ. Den stokastiske policyen samples og θ justeres for å skape større belønning. De samme prinsippene gjelder for DDPG, men omhandler en annerledes deterministisk policy; $a_t = \pi_\theta(s_t)$. Fordelen med algoritmen er at agenten kan vurdere mulige handlinger mens en lærer policyen, og blir mer effektiv enn Policy Gradient (S. Gu et al., 2016). Til gjengjeld er metoden tidkrevende og krever et stort antall treningsfaser.

Algoritmen har en "aktør-kritiker" utforming, hvor aktøren estimerer en deterministisk policy, mens kritikeren evaluerer aktørens handling ved å estimere Q-funksjonen. Metoden fungerer gjennom de dype nevrale nettverkene for aktøren, θ^μ og kritikeren, θ^Q som samhandler med hverandre. Aktørens handlinger utgjør porteføljevektingen, mens kritikeren kalkulerer avkastningen som vist i figur (5). "Aktør-kritiker"-algoritmen har evnen til å skape kontinuerlige handlinger samtidig som en unngår store variasjoner i policyen.

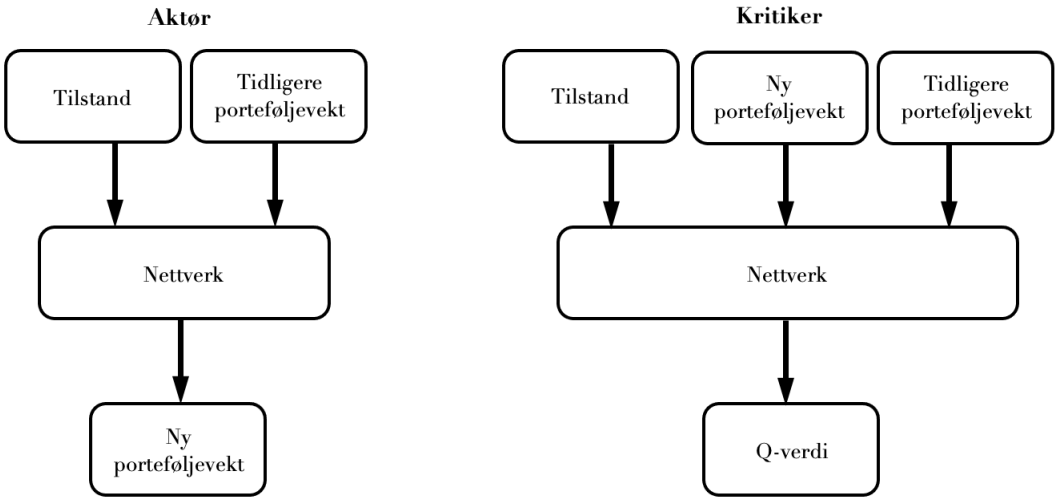


Figure 5: Aktørnettverk og Kritikernettverk

Figur (5) viser at aktør- og kritikernettverkene tar inn både nåværende tilstand og tidligere porteføljevekter. Årsaken til dette er at nettverkene benytter minnet til å lære seg at de ikke kan avvike mye fra forrige porteføljevekter, for å forhindre høye og unødvendige transaksjonskostnader. For de fleste andre forsterkningslæringsalgoritmer antas det uavhengighet mellom observasjoner og tilstander i forskjellige tidstrinn, siden de ikke anvender dette minnet (Lillicrap et al., 2015).

Treningen foregår off-policy og benytter Bellman-ligningen for å lære miljøet (Silver et al., 2014). I Q-learning benyttes Q-verdifunksjonen som gir forventet samlet belønning ved utførelse av en handling a_t i tilstand s_t og følger policy π i fremtiden, som er

$$Q_\pi(s_t, a_t) = \mathbb{E}_{(s_{t+1}, \sim\pi)}[\sum_{i=t}^{T} \gamma^i r(s_i, a_i)] \tag{42}$$

der $\mathbb{E}$ symboliserer en forventning. Bellmann-ligningen gjør det mulig å dekomponere funksjonen i flere komponenter. Ligningen viser at den forventede belønningen av en handling, a_t beregnes ved å ta forventning om belønning, $r(s_t, a_t, s_{t+1})$ samt forventet belønning i neste tilstand s_{t+1}. Basert på antakelsen om at avkastningen er diskontert med faktoren γ.

$$Q_\pi(s_t, a_t) = \mathbb{E}_{s_{t+1}}[r(s_t, a_t, s_{t+1}) + \gamma \mathbb{E}_{a_{t+1} \sim \pi(s_{t+1})}[Q_\pi(s_{t+1}, a_{t+1})]] \tag{43}$$

Ved deterministisk policy kan policyen skrives som; $\pi_d = S \leftarrow A$, og dermed unngås den indre forventningen slik at:

$$Q^\mu(s_t, a_t) = \mathbb{E}_{s_{t+1}}[r(s_t, a_t, s_{t+1}) + \gamma Q^\mu(s_{t+1}, \pi_d(s_{t+1})] \tag{44}$$

Dette viser at forventningen $\mathbb{E}$ er kun avhengig av miljøet. På denne måten er det mulig å lære Q^μ "off-policy". "Q-learning" er "off-policy" og benytter en "greedy policy", $\pi_d(s) = argmax_a Q(s, a)$ (Watkins & Dayan, 1992).[10] Videre er det en optimering av Q ved å minimere tap fra kritikernettverket θ^Q der y_t har en avhengighet av θ^Q.;

$$L(\theta^Q) = \mathbb{E}_{s_t \sim \rho^\beta, a_t \sim \beta, r \sim E}[(Q(s_t, a_t|\theta^Q) - y_t)^2] \tag{45}$$

For å lære aktøren benyttes Policy Gradient (Silver et al., 2014):

$$\nabla_{\theta\mu} J = \mathbb{E}_{s_t \sim \rho^\beta}[\nabla_a Q(s, a|\theta^Q)|_{s=s_t, a=\pi_d(s_t)} \nabla_{\theta_{\pi_d}} \pi_d(s|\theta^\mu)|_{s=s_t}] \tag{46}$$

På denne måten er det konstruert formuleringer for aktør- og kritikernettverket

[10] "Greedy policy" betyr at agenten gjennomfører den handlingen, a_t som er antatt å gi den høyeste belønningen.

som kan anvendes til dynamiske systemer.

7 Nettverksoppbygging

Nettverksoppbyggingen skal sørge for konstruering av handelspolitikken π_θ. De ulike nettverkene skal konsentreres omkring selve optimeringsproblemet av porteføljen, metodens evne til å huske og anvende tidligere porteføljevektingen og et mindre læringsnettverk for gjentatt trening av metoden.

Anvendt data er utformet som et euklidsk rom, og dette tillater konvolusjon. Detter gjør at forsterkningslæringen kan benyttes med konvolusjonelle nevrale nettverk (CNN).[11] Denne typen nettverk har vært anvendt til mønstergjenkjenning, lokalisering og parameterreduksjon (J. Gu et al., 2018). CNN benyttes i denne oppgaven til å hente prisinformasjon fra data.

7.1 Handelsnettverk

Handelsnettverket som benyttes er satt sammen av tre deler; databehandler, område for handelsalgoritmen og markedsimulator. Inndata til nettverket er pristensoren, X_t hvor pristensorens dimensjoner benyttes. Etter prosesser i nettverket skapes porteføljevekten, w_t. Databehandlingen er basert på avsnitt (4), og formulerer markedsdata til å bli tilgjengelig for nettverket. Det skal være kontinuerlig tilførsel av data for at handelsalgoritmen skal kunne operere. Området for handelsalgoritmen samler alle funksjoner, som beskrevet i avsnitt (6). Informasjon om agent-miljø og andre egenskaper ved forsterkningslæringsal-

[11] Vedlagt finnes det en illustrasjon av hvordan CNN fungerer i porteføljeforvaltning. Finnes i vedlegg A.2.

goritmen samles i denne nettverksdelen. Markedsimulatoren baseres på handlinger, og gjennomfører transaksjoner og signaliserer tilbakemeldinger på utførte handlinger til agenten i handelsalgoritmen. På denne måten kan handelsalgoritmen vurdere belønningsnivået.

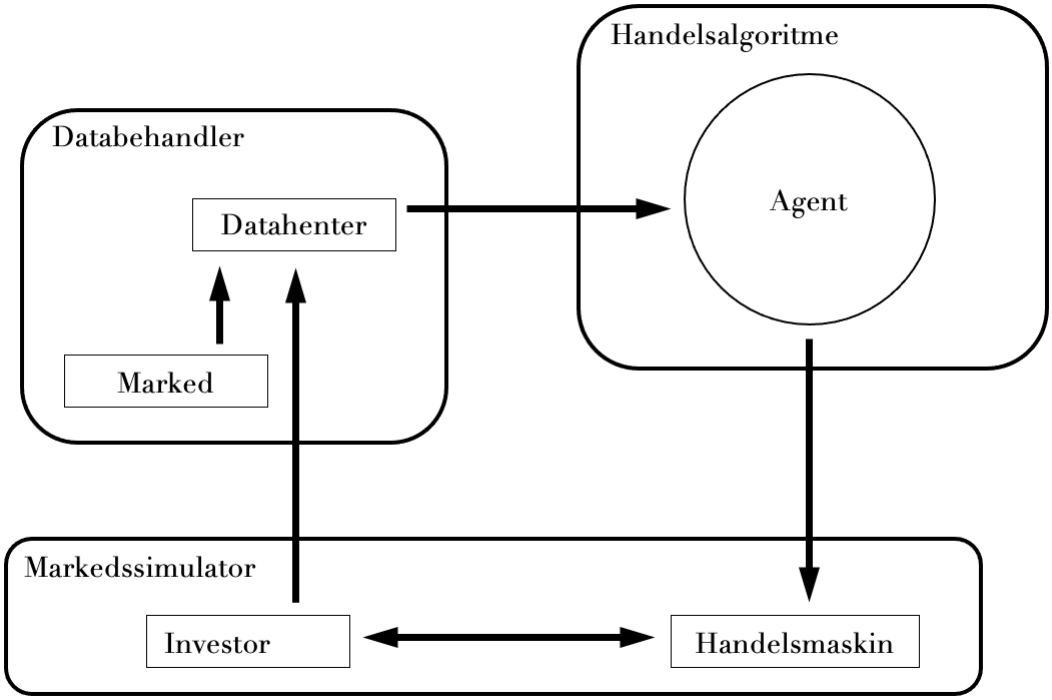


Figure 6: Oppbyggingen av handelsnettverket.

7.2 Agentnettverk

Agenten trenes og testes med utvalgt og prosessert datainformasjon. For hvert uavhengige eksperiment velges et tilfeldig tidsperspektiv t_f for både trening og test. Råmaterialet av data behandles i henhold til avsnitt (4), slik at metodens inndata er egnet for læring av agenten. For hver periode t vil da markedsdata

hentes til en datahenter som videresender data til agenten. Med dette blir agentnettverket oppdatert med informasjon som lagres og kombineres med tidligere informasjon fra nettverkets hukommelse.

Med den anvendte nettverksstrukturen i systemet vil ikke agentens observasjoner identifisere alle miljøets egenskaper og tilstander. Videre får dette konsekvenser for belønningene som er direkte knyttet til miljøets tilstander og overgangsfaser. Noe som gjør at individuelle observasjoner som overføres som signaler fra datahenteren ikke kan direkte anvendes for optimalisering. Dette gjør at agenten har behov for minne. Denne hukommelsesdelen brukes også til å vurdere transaksjonskostnader ved å vurdere forrige periodes porteføljevekt, w_{t-1} før neste handelsbeslutning. Nettverkets hukommelse er også kjent som "Replay Buffer".

Da informasjon er samlet og det er kommet en oppdatering i systemet vil agentnettverket være klart for neste handling a_{t+1}. Dette vil da være en ny fordeling av porteføljevekter w_{t+1} som skjer gjennom aktørnettverket. Disse handlingene påvirkes av støy som kommer fra markedssimulatoren, som viser at den har flere roller i nettverk som både behandler og signalelement. Støyet som blir påført av markedsimulatoren observeres av agenten som da går videre til neste tilstandsperiode s_{t+1}. Informasjon om tilstander, handlinger, belønninger og forventninger om neste tilstandsperiode lagres i nettverkets hukommelse. Prosessene som er beskrevet vil fortsette videre for hver periode inntil en når slutten t_f. Gjennom testperioden har agentnettverket en litt annen dynamikk, og treningsfunksjonene er fjernet. Agenten er avhengig av å benytte kunnskap fra treningsperioden og anvende denne for testperioden. Handlingene som gjennomføres kommer fra aktørnettverket, hvor agenten da ikke er i stand til å

observere støyen på samme måte som ved trening.

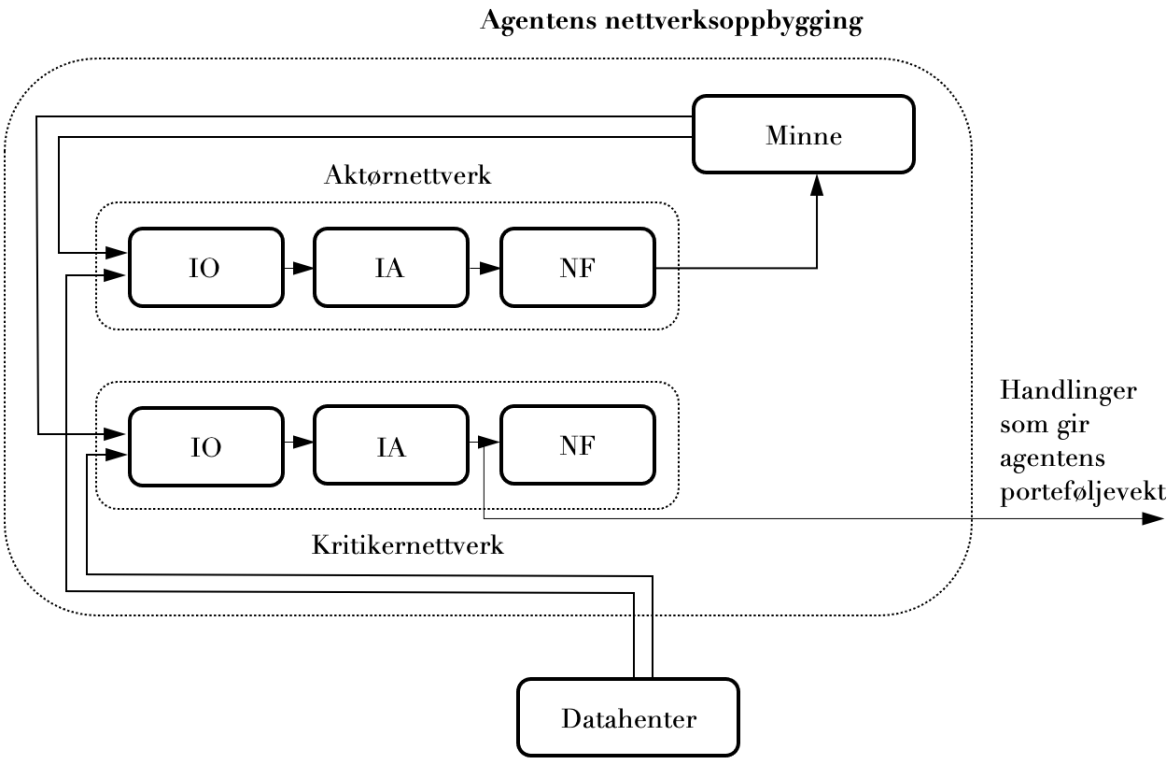


Figure 7: Agentens nettverksoppbygging

Databehandlingen tar inndata gjennom en designert enhet, datahenter som videre leder informasjonen til agenten. Siden DDPG blir benyttet er det alltid minst to nettverk som behandles, altså aktør og kritiker. Hver av disse er videre inndelt i tre enheter; informasjonsoppfatter (IO), informasjonsanalytiker (IA) og nettverksforbedrer (NF). Første enhet IO har som oppgave å oppfatte og hente informasjon fra pristensoren X_t. Som tidligere nevnt er dimensjonen på pristensoren, $X_t = (f, n, m) = (3, 60, 30)$. Informasjonen som produseres i IO skal videresendes til IA på vektorform. Denne informasjonen gir innsikt i ele-

menter ved markedsdata, blant annet om hvordan nåværende situasjon fremstår i henhold til prisnivå. Informasjonen sendes fra IO som en vektor og bearbeides gjennom et sett med nevroner (IA). Nettverket som foreslås i Jiang et al. (2017) inneholder ikke enheter som minner om IA som benyttes i denne metoden. Det kan gjøre at ikke-lineære forbindelser mellom de finansielle instrumentene ikke avdekkes. Nettverksforbedreren (NF) viser til hvordan nettverkene oppdateres og opererer for videre utvikling. NF-enhetene i aktør- og kritikernettverkene samhandler under læringsprosessen slik at aktøren fungerer med lavere intensitet enn kritikeren for å skape stabilitet. Minnet tilfører også informasjon til NF i aktørnettverket for å sørge for en lavere volatilitet i porteføljen.

8 Eksperimenter

Forsterkningslæringssystemet som er konstruert og forklart i avsnittene ovenfor skal gjennomføres og vurderes basert på syv uavhengige back-test eksperimenter.[12] Resultatene fra disse vil redegjøres, forklares og sammenlignes mot en annen porteføljestrategi, aksjeindeks og et børsnotert fond tilknyttet aksjeindeksen. Sammenligningene gjøres gjennom forskjellige målinger, hvor hovedvekten vil fokusere på den genererte porteføljeavkastningen. For å videre forstå resultatene vil andre ytelsesindikatorer komplementer undersøkelsene, som i dette tilfellet vil være volatilitet, Sharpe Ratio, "Maximal Drawdown" (MDD) og "Conditional Value at Risk" (CVaR).

[12] "Back-test" betyr at historisk data anvendes slik at en undersøker hvor godt systemet ville fungert på tidligere hendelser.

Tidsperiodene som er benyttet til trenings- og testsett er presentert i tabell (2). Fra datasettet benyttes åpningskurs, høyeste, laveste og sluttkurs for alle eksperimenter. Det er valgt å benytte varierende tidsperioder for treningssettene slik at agenten opplever ulike læringsforhold i tillegg til at en får vurdert agentens prestasjoner med ulik lengde på læringsperioder. I forhold til pristensorens dimensjon n strekkes mindre treningsperioder over kortere perioder innad i selve treningssettet. Porteføljen består kun av kontanter ved begynnelsen av første aktive handelsdag, og det er agentens handlinger fra første periode som videre tilsier porteføljevektingen.

Table 2: Oversikt over datasett i eksperimenter

Eksperiment	Treningssett	Testsett
1	01.01.2003 - 01.09.2019	01.03.2021 - 01.09.2021
2	01.01.2003 - 27.11.2017	01.06.2018 - 01.02.2019
3	01.01.2003 - 14.03.2015	01.05.2015 - 27.01.2016
4	01.01.2003 - 01.07.2011	21.11.2012 - 21.05.2013
5	01.01.2003 - 12.10.2009	14.10.2009 - 07.03.2010
6	01.01.2003 - 27.11.2017	24.02.2018 - 07.06.2018
7	24.01.2012 - 23.02.2019	27.02.2019 - 07.04.2021

Læringsraten til metoden har stor innvirkning på treningsprosessene i de nevrale nettverkene. Dersom læringsraten ikke er riktig justert vil det være umulig å oppnå optimaliserte løsninger. Ved for høy læringsrate beveger treningen seg nært optimal løsning, men vil ikke være kapabel til å komme nærmere enn de løsningene som oppnås tidlig i treningsprosessen. Ved for lav læringsrate vil det kreve flere treningsepisoder for å nærme seg optimal løsning, og vil dermed skape ressurskrevende prosesser. Med gjentatte læringsprosesser er læringsraten

justert til det mest vellykkede nivået, med en rate på 0.001.

Det er valgt å sette transaksjonskostnadene til 0.002 ($\mu = 0.2\%$) av handelssum for både kjøp og salg. Kostnadsnivået som settes er tilstrekkelig til å dekke både provisjonsgebyr og andre tilknyttede kostnader. I tillegg er det for disse eksperimentene satt en kvantumsbegrensning på individuelle aksjer i porteføljen. Det er ikke tillatt å eie mer enn 200 aksjer fra hvert selskap. Ved oppstarten er agentens budsjettet på 1 million amerikanske dollar.

8.1 Kvalitetsmåling

For å løse problemstillingen ønsker vi at agenten skal maksimere porteføljeavkastningen, samtidig som risikoen minimeres. Den klareste måten å bedømme prestasjonen av en porteføljestrategi vil være gjennom avkastningen.

$$CR = \frac{p_{t_f} - p_0}{p_0} \tag{47}$$

Dette vil derimot være noe primitivt da investorenes risikovilje og risikoposisjon ikke vurderes. Derfor er det benyttet andre metoder for å vurdere risikoen av porteføljene fra back-test eksperimentene. Sharpe Ratio måler volatiliteten av investeringen som et forhold mellom forventet avkastning og risiko (Sharpe, 1964)(Sharpe, 1994).

$$S_r = \frac{E[r_f - r_b]}{\sigma_d} \tag{48}$$

σ_d viser til standardavviket for meravkastningen, altså volatiliteten. $E[r_f - r_b]$ er forventet avkastning utover risikofri rente. Dette fungerer som et hjelpemiddel

for investorer til å vurdere handlinger gjennom avkastning på risikonivåer. En svakhet med metoden er avhengigheten av standardavvik, og økning/nedgang i avkastning. Positive endringer gir høyere avkastning, mens negative endringer gir lavere avkastning, men det er ikke sikkert at Sharpe klarer å tydeliggjøre dette i sin representasjon.

Sharpe har som nevnt svakhet i vurdering av nedgangsperioder, derfor benytter vi også maksimal nedtrekking (MDD).

$$MDD_t = \frac{Topp - Bunn}{Topp} \tag{49}$$

Metoden viser forskjellen i porteføljeverdiens topp- og bunnpunkt gjennom en tidsperiode før en når et nytt topppunkt. Metoden viser i realiteten den prosentvise størrelsen av tapet.

Videre anvendes "Conditional Value at Risk" (CVaR) som er en forlengelse av "Value at Risk" (VaR). VaR estimerer potensialet av en tapssituasjon som metoden anser som verste scenario og den tilknyttede sannsynligheten. Metoden er lett anvendelig, men viser derimot ikke høyeste potensielle tap, kun sannsynligheten. Porteføljerisikoen kan faktisk være høyere enn estimatet og misvisende ved ustabile omstendigheter. Derfor benyttes CVaR som en risikoestimator, som estimerer haleverdien av VaR. Det vil si at metoden gir forventet tap gitt at en krysser det scenario som er satt av VaR.

8.2 Resultater

I denne delen fremvises resultatene av handelsstrategiene fra eksperimentene i tabellene (3, 4, 5). Det blir vist at strategien i metoden kan utkonkurrere markedsindeksen og annen porteføljestrategi. Handelssimuleringene gjøres ved 7 eksperimenter med ulik trenings- og testlengde for å vurdere systemet i forskjellige tilstander og perioder. Handelssimuleringene viser varierte resultater som vil undersøkes nærmere.

Det viser seg at de fleste transaksjoner foregår i de begynnende fasene av porteføljens tidsperiode. Grunnen til dette er at agenten har lært effekten av transaksjonskostnader. En handelsstrategi som har flere transaksjoner og dermed høy frekvens i endring av porteføljevekt vil være urimelig i bytteforholdet mellom avkastning og kostnad. Dette vil da forhindre hensikten med porteføljeforvaltningen, derfor vil det ikke være gunstig og agenten har oppfattet denne dynamikken.

Den rikelige tilgangen på data gjør at agenten skal være i stand til å vurdere risikoen. Periodene som dekkes av treningsdata er også i alle eksperimentene dekket av både Bull- og Bear-situasjoner. Dette gir flere erfaringer for agenten under treningen og bedre grunnlag for testperioden. Mangel på data ville gjort metoden mye mer sårbar, og risikert å gå utover ytelsen. For forsterkningslæring er det også viktig å benytte mye data, og med mindre data er det ikke sikkert at metoden ville vært egnet til dette formålet. Det å benytte de kvalitetsmålingene som benyttes gjør at vi kan undersøke agentens prestasjon og med dette viste til risikoposisjon i porteføljen.

Høyeste kurs, laveste kurs og sluttkurs er de egenskapene som er utvalgt til den endelige pristensoren som anvendes i eksperimentene. Kompleksiteten i porteføljeforvaltning gjør at flere egenskaper kan involveres. I tillegg til disse egenskapene som beskriver pris, kunne andre finansielle indikatorer som beskriver markedsbevegelser vært inkludert. Begynnende forsøksperioder involverte flere egenskaper, men disse ble irrelevant for prosessen og tilførte støy. Forsøksperiodene viste at de egenskapene som skaper høyeste ytelse var de som utgjør anvendt pristensor.

Resultatene av eksperimentene og de innledende undersøkelsene viser ingen relevante antydninger på sammenheng mellom agentens prestasjon og lengde på treningssett. Dette kan tyde på at læringsraten som er bestemt til eksperimentene er robust overfor varierende treningslengde.

Table 3: Risikoevaluering for forsterkningslæringsmetoden

Eksperiment	Avkastning	Sharpe Ratio	CVaR	MDD	Volatilitet
1	12.229 %	2.18	-1.246%	-4.337%	10.617%
2	6.791 %	0.72	-1.898%	-14.232%	15.410%
3	-9.421 %	-0.69	-2.212 %	-14.440%	17.185%
4	24.099 %	4.69	-1.023%	-2.891%	9.526%
5	7.721%	1.32	-1.854%	-7.723%	15.357%
6	-2.381 %	-0.36	-1.944%	-7.460%	15.255%
7	37.532%	0.73	-3.088%	-31.687%	25.085%

Kun basert på tabell (3) kan det være vanskelig å trekke konkrete konklusjoner angående prestasjonen i de forskjellige eksperimentene. Flere av eksperimentene har positiv avkastning, men noen har gjennom testperioden opplevd negativ

utvikling. Effekten av avkastningene gjenspeiles også på Sharpe Ratio som er negative ved negativ avkastning. Slik det fremstår er det noen eksperimenter som leverer sterke resultater, mens andre skuffer. Det som tabellen ikke tar hensyn til er de overordnede forholdene i markedet gjennom testperiodene. For å skape god oversikt vil det også produseres tabeller for Dow Jones og DIA (ETF). Ved å gjøre dette får en satt resultatene i perspektiv, og det blir mulig å virkelig evaluere ytelsen til forsterkningslæringens handelsstrategi.

8.2.1 Dow Jones og DIA

Table 4: Risikoevaluering for Dow Jones

Eksperiment	Avkastning	Sharpe Ratio	CVaR	MDD	Volatilitet
1	12.254 %	2.12	-1.308%	-4.278%	11.127%
2	1.541 %	0.22	-2.173%	-18.772%	17.371%
3	-11.867 %	-0.95	-2.179 %	-14.449%	16.794%
4	19.612%	3.90	-1.070%	-3.092%	9.685%
5	4.276 %	0.82	-1.812%	-7.618%	14.760%
6	-5.969 %	-1.03	-2.042%	-8.464%	15.694%
7	29.024 %	0.59	-3.325%	-37.086%	26.884%

Table 5: Risikoevaluering for DIA

Eksperiment	Avkastning	Sharpe Ratio	CVaR	MDD	Volatilitet
1	13.123 %	2.28	-1.288%	-4.155%	11.010%
2	2.911 %	0.34	-2.167%	-18.140%	17.387%
3	-10.221 %	-0.80	-2.172 %	-13.867%	16.818%
4	21.016 %	4.23	-1.039%	-3.054%	9.513%
5	5.374 %	1.03	-1.756%	-7.418%	14.406%
6	-5.310 %	-0.89	-2.058%	-8.314%	15.891%
7	34.903 %	0.67	-3.296%	-36.696%	26.724%

Avkastning

Ved å utelukkende se på avkastningen kommer det frem at strategien har gode forutsetninger for å utkonkurrere markedet. Det er kun første eksperiment som leverer svakere avkastning enn markedsindeksen, og til gjengjeld er denne forskjellen marginal på 0.025 %. Det største positive avviket kommer ved eksperiment 7, der agenten opplever 8.508 % høyere avkastning over drøye 2 år. Den største relative differansen kommer i andre eksperiment på 5.250 % som strekker seg over 8 måneder. Ved sammenligning med DIA kommer det også frem at forsterkningslæringen leverer svakere resultater i første eksperiment. Det er også det eneste tilfelle hvor agenten utkonkurreres, og viser at kun tatt i betraktning avkastning er ytelsen til agenten tilfredsstillende i 6 av 7 eksperimenter.

Sharpe Ratio

Som forventet har eksperimentenes avkastning og Sharpe Ratio tilsvarende likheter. Dette betyr at risikonivået mellom porteføljen og aksjeindeksen ikke er så ulike, og det som skaper Sharpe Ratioen er rett og slett den oppnådde avkastningen. Dette er meget positivt for forsterkningslæringen, og viser at den klarer å

forholde seg til et risikoområde og derfra vurdere allokeringsmulighetene. Ønsket med metoden var å skape en programvare som kan være bærekraftig over perioder og samtidig vurdere risiko på en god måte. Dersom det hadde vært enormt store forskjeller mellom metode og indeks ville det vært grunn til å stille spørmål ved agentens risikoposisjon.

De negative Sharpe Ratioene viser at handelsstrategien ikke alltid klarer å håndtere risikoen i forhold til avkastning, men dette er ikke overraskende. DIA er ikke ideelt for en investor som ønsker bred eksponering mot amerikanske aksjer, og anvendt portefølje speiler det børsnoterte fondet. Dette er på grunn av det relativt lave antallet individuelle eiendeler i porteføljen som kan skape skjevheter mot bransjer og sektorer i forhold til et bredere marked. Siden porteføljen har tilsvarende eiendeler som Dow Jones og DIA er det relative forholdet til disse sammenligningene bedre representant for ytelse sammenlignet med en gitt verdi for Sharpe Ratio.[13] Selv om dette er tilfellet er særlig ytelsen i eksperiment 3 og 6 mye svakere enn hva som ville vært foretrukket.

Volatilitet

Ved å se på volatiliteten kommer det fram at agenten gjør det bra sammenlignet med markedet. Det er positivt. Dersom volatiliteten hadde vært markant høyere enn indeksen og børsnotert fond ville det vært bekymringer for eventuell "over-fitting" til treningssett. Hvor agenten da ved testing kan gjennomføre beslutninger som fører til hyppigere tap som videre øker volatiliteten. Det at agenten har klart og holde volatiliteten nede gjør at risikograden senkes, noe som var en av ønskene med metoden. Ønsket var å utfordre markedet, men ikke bare på avkastning. Målsetningen var å skape en gjennomgående solid

[13]Sharpe Ratio over 1 er sett på som godt resultat, mens verdier over 2 er veldig gode.

forsterkningslæringsmetode som kan optimere porteføljeforvaltningen, da med blant annet bedre risikohåndtering.

Conditional Value at Risk

Forskjellene mellom agentens handelsstrategi og de andre sammenligningene gir små forskjeller i daglig CVaR. De tilfellene hvor volatiliteten er høyest gir også høyeste CVaR, og gir grunnlag for å dele vurderingene fra delen om volatilitet ovenfor.

Grafisk illustrasjon

Figur (8) illustrerer avkastningsutviklingen under første eksperiment. Prestasjonen er lavere enn alle sammenligninger. Utviklingen er tilsvarende for alle metoder, med lignende svingninger og dermed mindre variasjoner mellom dem. I denne handelsperioden har DIA høyeste avkastning ved tidsslutt.

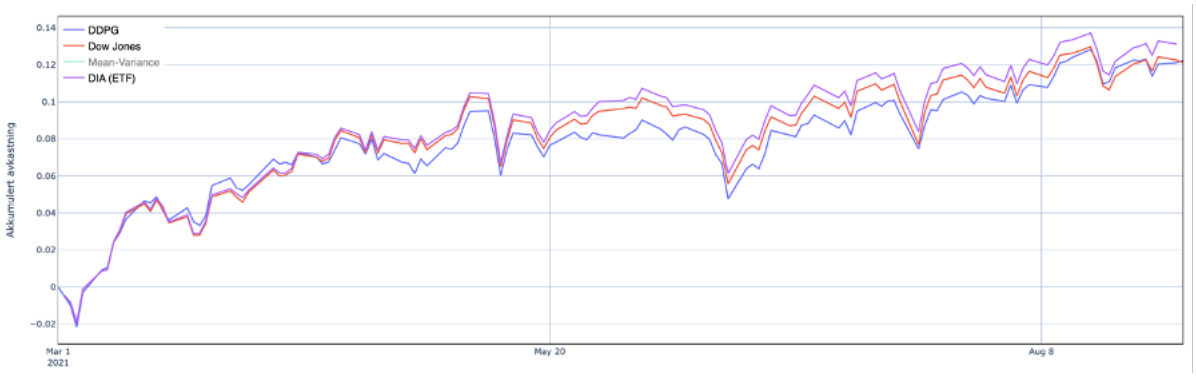


Figure 8: Avkastning for Dow Jones, DIA og agentens handlingsstrategi.

Eksperiment 2 opplevde den største forskjellen i avkastning mellom handlingsstrategien, Dow Jones og DIA. I dette eksperimentet viser det seg at agenten klarer å kapitalisere bedre på vekstperioder, men klarer ikke å utnytte nedgangsperioder

på bedre måte. Figur (9) viser utviklingen gjennom testperioden.

Figure 9: Avkastning for Dow Jones, DIA og agentens handlingsstrategi.

8.2.2 Mean-Variance

Figur (10) viser grafisk utvikling i avkastning fra eksperiment 2.[14] Dette viser at selv om agenten utkonkurrerer Dow Jones og DIA er det ikke kommet frem til optimal løsning, siden mean-variance strategien markant utkonkurrerer avkastningsnivået til de andre.

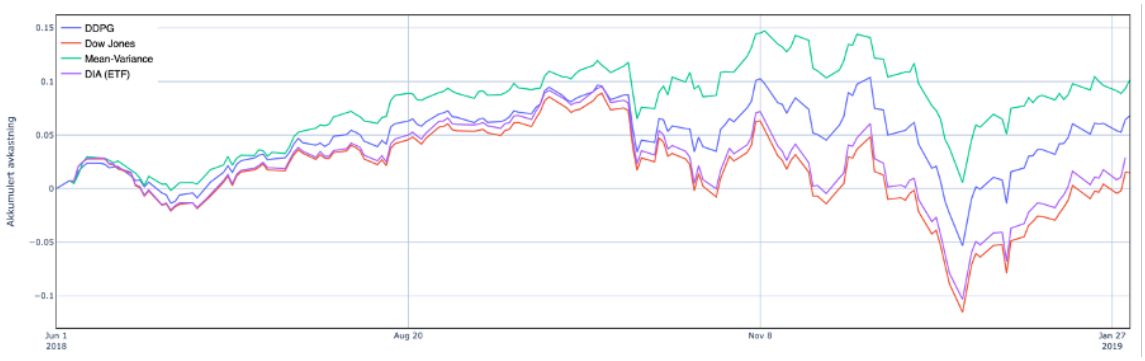


Figure 10: Avkastning for Dow Jones, DIA og agentens handlingsstrategi.

[14] Samme illustrasjon som figur (9), men inkluderer også Mean-Variance

Ved implementeringen av mean-variance kommer det tydeligere frem hvor sterk prestasjonen til agenten har vært. Kun ved eksperiment 4, 5 og 7 er det agentens handelsstrategi som utkonkurrer alle de andre.[15] Eksperiment 6 gir tilnærmet lik utvikling mellom agenten og mean-variance.

9 Konklusjon

Forsterkningslæringens struktur har vært meget solid og god. Konseptet har vært å ikke forsøke å komplisere nettverkene. Ved å unngå for komplekse nettverk har vi skapt en metode som ikke risikerer høy grad av "over-fitting". Det viser seg at det å komplisere metoden kan føre til nettopp "over-fitting" (Yu et al., 2019)(Liang et al., 2018). Evnen til å diversifisere data under datahåndteringen har gjort det mulig å forhindre "over-fitting". Det å gjøre nettverkene mer kompleks kan fungere som en "ond sirkel" hvor en videre må kompensere for effekter ved å involvere flere enheter og strukturer som kanskje virker mot sin hensikt. Anvendelsen av modellfri forsterkningslæring har bidratt til å unngå disse kompliserte forholdene.

Resultatene fra eksperimentene har vært oppløftende og viser at handlingsstrategien leverer gode resultater sammenlignet med markedsindeksen og DIA. Metoden sliter med kontinuerlig utkonkurrering av mean-variance tilnærmingen, som tyder på at metoden ikke har klart å etablere optimal handlingsstrategi ved alle tilfeller. Resultatene viser at det finnes potensial i arbeidet, men at det nok er behov for justeringer av parametere, nettverksroller og lignende for å skape optimale problemløsninger. Noe av utfordringen er dynamikkene i finansielle

[15] Illustrasjoner av de grafiske utviklingene i avkastning er vedlagt for eksperiment 4, 5 og 7, samt andre eksperimenter som ikke er illustrert i teksten. Finnes i vedlegg A.4.

marked og de kontinuerlige handlings- og tilstandsrommene, hvor suksessfaktorene endres over tid og dermed krever at forsterkningslæringen modifiseres for å håndtere disse forandringene.

Potensialet til å anvende suksessfulle forsterkningslæringsalgoritmer i finansielle markeder er definitivt tilstede. Ved videre utvikling av DDPG eller tilsvarende metoder som bygger på de samme konseptene bør det være mulig å skape nye handlingsstrategier som kan kontinuerlig utkonkurrere andre handlingsstrategier.

9.1 Forslag til videre arbeid

Sammenligninggrunnlaget for agentens handelsstrategi og andre handelsstrategier kan med forbehold utvides. Ved videre arbeid ville det vært hensiktsmessig å inkludere flere metoder til sammenligning. Det ville også vært interessant å testet agenten mot andre markedindekser, både med samme og annen aktivasammensetning i porteføljen. Undersøkelsene som er gjennomført er også utelukkende basert på aksjemarkeder og muligheten for å involvere andre aktivaklasser (obligasjoner, kryptovaluta osv.) er noe som kan utforskes videre. Det å endre systemet slik at det tillatter short-posisjoner er også interessant for videre arbeid. Det hadde vært fordelaktig med lengre tid til å bearbeide systemet og dermed kommet med dypere evalueringer av resultater

Beslutningsprosessene i forsterkningslæringen er basert på ett-nivå MDP, men kunne vært transformert til hierarkisk utforming. En slik tilnærming gjør at agenten skal gjennomføre beslutninger gjennom en hierarkisk handelspolitikk på to nivåer (høy og lav). Porteføljestyringen skal fremstå som høynivå og avgjør de

målsetningene som skal oppnås, mens handelsbeslutninger vil utgjøre lavnivået der mindre oppgaver skal gjennomføres for å oppnå porteføljestyringens målsetninger Le et al., 2018. På denne måten ville det vært enklere å skape belønninger som fokuserte mer på kombinasjonen av avkastning og risiko enn det som er tilfelle i nåværende system.

Handelsnettverket er konstruert slik at agenten skal lære miljøet uten særlig inngående kunnskap. Med lengre bearbeidingstid kunne det vært mulig å legge inn flere modelleringer i nettverket for å undersøke de effektene som ville vært på systemet som en følge av disse endringene. Da hadde det vært mulig å se på hvordan systemet tilpasser seg data.

Referanseliste

Bekaert, G., & Hoerova, M. (2014). The vix, the variance premium and stock market volatility. *Journal of econometrics*, *183*(2), 181–192.

Bellman, R. (1957a). A markovian decision process. *Journal of mathematics and mechanics*, 679–684.

Bellman, R. (1966). Dynamic programming. *Science*, *153*(3731), 34–37.

Bellman, R. (1957b). Dynamic programming princeton university press princeton. *New Jersey Google Scholar*.

Black, J. (2013). Reconceiving financial markets—from the economic to the social. *Journal of corporate law studies*, *13*(2), 401–442.

Bollerslev, T., Tauchen, G., & Zhou, H. (2009). Expected stock returns and variance risk premia. *The Review of Financial Studies*, *22*(11), 4463–4492.

Bonanno, G., Lillo, F., & Mantegna, R. N. (2001). High-frequency cross-correlation in a set of stocks.

Busoniu, L., Babuska, R., De Schutter, B., & Ernst, D. (2017). *Reinforcement learning and dynamic programming using function approximators*. CRC press.

Chai, J., & Ngai, E. W. (2020). Decision-making techniques in supplier selection: Recent accomplishments and what lies ahead. *Expert Systems with Applications*, *140*, 112903.

Deng, Y., Bao, F., Kong, Y., Ren, Z., & Dai, Q. (2016). Deep direct reinforcement learning for financial signal representation and trading. *IEEE transactions on neural networks and learning systems*, *28*(3), 653–664.

Fabozzi, F., Gupta, F., & Markowitz, H. (2002). The legacy of modern portfolio theory. *The Journal of Investing*, *11*, 7–22. https://doi.org/10.3905/joi.2002.319510

Fama, E. F. (1970). Efficient capital markets: A review of theory and empirical work. *The journal of Finance*, *25*(2), 383–417.

Filos, A. (2019). Reinforcement learning for portfolio management. *arXiv preprint arXiv:1909.09571*.

Gill, P. E., Murray, W., & Wright, M. H. (2019). *Practical optimization*. SIAM.

Gilli, M., Maringer, D., & Schumann, E. (2019). *Numerical methods and optimization in finance*. Academic Press.

Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. MIT press.

Gu, J., Wang, Z., Kuen, J., Ma, L., Shahroudy, A., Shuai, B., Liu, T., Wang, X., Wang, G., Cai, J., et al. (2018). Recent advances in convolutional neural networks. *Pattern Recognition*, *77*, 354–377.

Gu, S., Lillicrap, T., Ghahramani, Z., Turner, R. E., & Levine, S. (2016). Q-prop: Sample-efficient policy gradient with an off-policy critic. *arXiv preprint arXiv:1611.02247*.

Guo, Y., Fu, X., Shi, Y., & Liu, M. (2018). Robust log-optimal strategy with reinforcement learning. *arXiv preprint arXiv:1805.00205*.

Huang, G., Zhou, X., & Song, Q. (2020). Deep reinforcement learning for portfolio management based on the empirical study of chinese stock market. *arXiv preprint arXiv:2012.13773*.

Jiang, X., Pan, S., Jiang, J., & Long, G. (2018). Cross-domain deep learning approach for multiple financial market prediction. *2018 international joint conference on neural networks (IJCNN)*, 1–8.

Jiang, Z., Xu, D., & Liang, J. (2017). A deep reinforcement learning framework for the financial portfolio management problem. *arXiv preprint arXiv:1706.10059*.

Kaelbling, L. P., Littman, M. L., & Cassandra, A. R. (1998). Planning and acting in partially observable stochastic domains. *Artificial intelligence*, *101*(1-2), 99–134.

Kang, Q., Zhou, H., & Kang, Y. (2018). An asynchronous advantage actor-critic reinforcement learning method for stock selection and portfolio management. *Proceedings of the 2nd International Conference on Big Data Research*, 141–145.

Le, T. P., Vien, N. A., & Chung, T. (2018). A deep hierarchical reinforcement learning algorithm in partially observable markov decision processes. *IEEE Access*, *6*, 49089–49102. https://doi.org/10.1109/ACCESS.2018.2854283

Liang, Z., Chen, H., Zhu, J., Jiang, K., & Li, Y. (2018). Adversarial deep reinforcement learning in portfolio management. *arXiv preprint arXiv:1808.09940*.

Lillicrap, T. P., Hunt, J. J., Pritzel, A., Heess, N., Erez, T., Tassa, Y., Silver, D., & Wierstra, D. (2015). Continuous control with deep reinforcement learning. *arXiv preprint arXiv:1509.02971*.

Littman, M. (2001). Markov decision processes. In N. J. Smelser & P. B. Baltes (Eds.), *International encyclopedia of the social behavioral sciences* (pp. 9240–9242). Pergamon. https://doi.org/https://doi.org/10.1016/B0-08-043076-7/00614-8

Markowitz, H. (1952). Portfolio selection. *The Journal of Finance*, *7*(1), 77–91. Retrieved May 5, 2022, from http://www.jstor.org/stable/2975974

Markowitz, H. M. (1968). Portfolio selection. *Portfolio selection*. Yale university press.

Meng, T. L., & Khushi, M. (2019). Reinforcement learning in financial markets. *Data*, *4*(3), 110.

Mnih, V., Badia, A. P., Mirza, M., Graves, A., Lillicrap, T., Harley, T., Silver, D., & Kavukcuoglu, K. (2016). Asynchronous methods for deep reinforcement learning. *International conference on machine learning*, 1928–1937.

Mnih, V., Kavukcuoglu, K., Silver, D., Graves, A., Antonoglou, I., Wierstra, D., & Riedmiller, M. (2013). Playing atari with deep reinforcement learning. *arXiv preprint arXiv:1312.5602*.

Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., Graves, A., Riedmiller, M., Fidjeland, A. K., Ostrovski, G., et al. (2015). Human-level control through deep reinforcement learning. *nature*, *518*(7540), 529–533.

Moody, J., & Saffell, M. (2001). Learning to trade via direct reinforcement. *IEEE transactions on neural Networks*, *12*(4), 875–889.

Moré, J. J., & Toraldo, G. (1989). Algorithms for bound constrained quadratic programming problems. *Numerische Mathematik*, *55*(4), 377–400.

Mosavi, A., Faghan, Y., Ghamisi, P., Duan, P., Ardabili, S. F., Salwana, E., & Band, S. S. (2020). Comprehensive review of deep reinforcement learning methods and applications in economics. *Mathematics*, *8*(10), 1640.

Neuneier, R. (1997). Enhancing q-learning for optimal asset allocation. *Advances in neural information processing systems*, *10*.

Ormos, M., & Urbán, A. (2013). Performance analysis of log-optimal portfolio strategies with transaction costs. *Quantitative Finance*, *13*(10), 1587–1597.

Pan, R. (2014). Tensor transpose and its properties. *arXiv preprint arXiv:1411.1503*.

Park, H., Sim, M. K., & Choi, D. G. (2020). An intelligent financial portfolio trading strategy using deep q-learning. *Expert Systems with Applications*, *158*, 113573.

Sharpe, W. F. (1964). Capital asset prices: A theory of market equilibrium under conditions of risk. *The journal of finance*, *19*(3), 425–442.

Sharpe, W. F. (1994). The sharpe ratio. *The Journal of Portfolio Management*, *21*(1), 49–58. https://doi.org/10.3905/jpm.1994.409501

Silver, D., Huang, A., Maddison, C. J., Guez, A., Sifre, L., Van Den Driessche, G., Schrittwieser, J., Antonoglou, I., Panneershelvam, V., Lanctot, M., et al. (2016). Mastering the game of go with deep neural networks and tree search. *nature*, *529*(7587), 484–489.

Silver, D., Lever, G., Heess, N., Degris, T., Wierstra, D., & Riedmiller, M. (2014). Deterministic policy gradient algorithms, I–387–I–395.

Singh, S. P., Jaakkola, T., & Jordan, M. I. (1994). Learning without state-estimation in partially observable markovian decision processes. *Machine learning proceedings 1994* (pp. 284–292). Elsevier.

Soleymani, F., & Paquet, E. (2021). Deep graph convolutional reinforcement learning for financial portfolio management–deeppocket. *Expert Systems with Applications*, *182*, 115127.

Sutton, R. S., Barto, A. G. et al. (1998). Introduction to reinforcement learning.

Sutton, R. S., & Barto, A. G. (2018). *Reinforcement learning: An introduction*. MIT press.

Tesauro, G. (1994). Td-gammon, a self-teaching backgammon program, achieves master-level play. *Neural computation*, *6*(2), 215–219.

Tran, D. T., Magris, M., Kanniainen, J., Gabbouj, M., & Iosifidis, A. (2017). Tensor representation in high-frequency financial data for price change prediction. *2017 IEEE Symposium Series on Computational Intelligence (SSCI)*, 1–7.

Watkins, C. J., & Dayan, P. (1992). Q-learning. *Machine learning*, *8*(3), 279–292.

Wiering, M. A., & Van Otterlo, M. (2012). Reinforcement learning. *Adaptation, learning, and optimization*, *12*(3), 729.

Wierstra, D., Foerster, A., Peters, J., & Schmidhuber, J. (2007). Solving deep memory pomdps with recurrent policy gradients. *International conference on artificial neural networks*, 697–706.

Xiong, Z., Liu, X.-Y., Zhong, S., Yang, H., & Walid, A. (2018). Practical deep reinforcement learning approach for stock trading. *arXiv preprint arXiv:1811.07522*.

Yu, P., Lee, J. S., Kulyatin, I., Shi, Z., & Dasgupta, S. (2019). Model-based deep reinforcement learning for dynamic portfolio optimization. *arXiv preprint arXiv:1901.08740*.

Zhang, W., & Wang, J. (2017). Nonlinear stochastic exclusion financial dynamics modeling and time-dependent intrinsic detrended cross-correlation. *Physica A: Statistical Mechanics and its Applications*, *482*, 29–41.

Zhao, D., Wang, H., Shao, K., & Zhu, Y. (2016). Deep reinforcement learning with experience replay based on sarsa. *2016 IEEE Symposium Series on Computational Intelligence (SSCI)*, 1–6.

Figurliste

A Vedlegg

A.1 Effekten av transaksjonskostnader

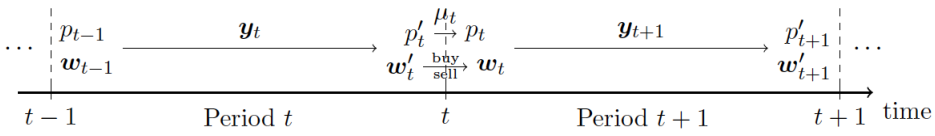


Figure 11: Figuren viser effekten av transaksjonskostnader, μ_t. Gjennomførte handlinger i en periode gir avsluttende porteføljeverdi, $p_t^{'}$ og porteføljevekt, $w_t^{'}$ før transaksjonskostnadene er medberegnet. Figuren viser videre at transaksjonskostnadene fører til nye verdier som blir henholdsvis; p_t og w_t. Dette er da avsluttende verdier etter en periode, men også begynnende verdier for neste periode.

A.2 Convolutional Neual Network

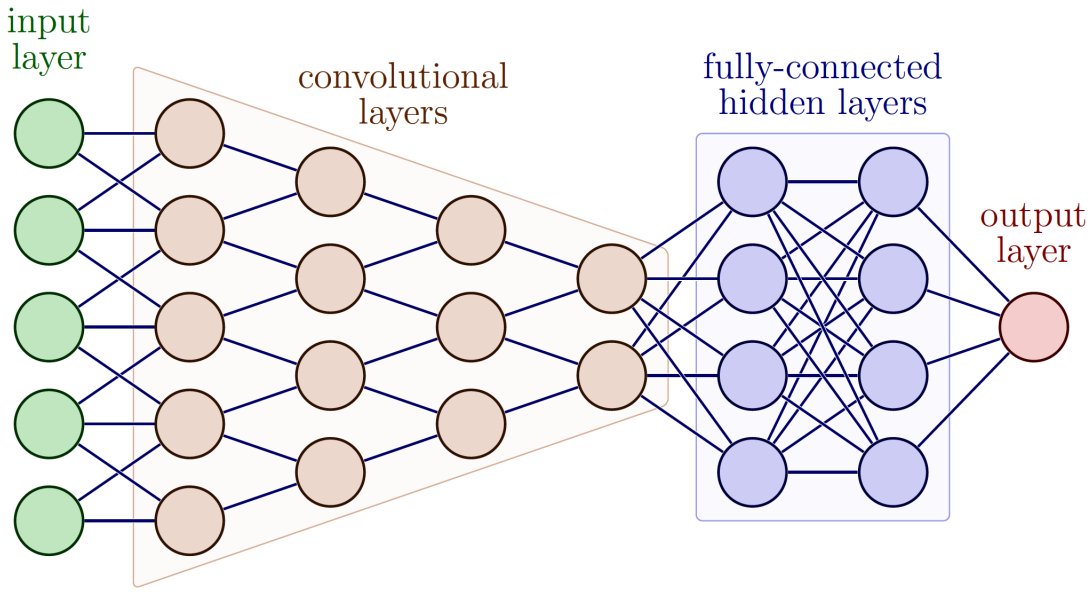


Figure 12: Illustrasjon av CNN som viser hvordan nettverket behandler inndata, og transformerer det til utdata som i dette tilfellet er porteføljevekt, w_t.

A.3 Pseudokode for DDPG

Algorithm 1 Deep Deterministic Policy Gradient

Input: initial policy parameters θ, Q-function parameters ϕ, empty replay buffer $\mathcal{D}$
Set target parameters equal to main parameters $\theta_{\text{targ}} \leftarrow \theta$, $\phi_{\text{targ}} \leftarrow \phi$
repeat
  Observe state s and select action $a = \text{clip}(\mu_\theta(s) + \epsilon, a_{Low}, a_{High})$, where $\epsilon \sim \mathcal{N}$
  Execute a in the environment
  Observe next state s', reward r, and done signal d to indicate whether s' is terminal
  Store (s, a, r, s', d) in replay buffer $\mathcal{D}$
  If s' is terminal, reset environment state.
  if it's time to update **then**
    for however many updates **do**
      Randomly sample a batch of transitions, $B = \{(s, a, r, s', d)\}$ from $\mathcal{D}$
      Compute targets

$$y(r, s', d) = r + \gamma(1 - d)Q_{\phi_{\text{targ}}}(s', \mu_{\theta_{\text{targ}}}(s'))$$

      Update Q-function by one step of gradient descent using

$$\nabla_\phi \frac{1}{|B|} \sum_{(s,a,r,s',d) \in B} \left(Q_\phi(s, a) - y(r, s', d)\right)^2$$

      Update policy by one step of gradient ascent using

$$\nabla_\theta \frac{1}{|B|} \sum_{s \in B} Q_\phi(s, \mu_\theta(s))$$

      Update target networks with

$$\phi_{\text{targ}} \leftarrow \rho\phi_{\text{targ}} + (1 - \rho)\phi$$
$$\theta_{\text{targ}} \leftarrow \rho\theta_{\text{targ}} + (1 - \rho)\theta$$

    end for
  end if
until convergence

Figure 13: Pseudokode som forklarer dynamikken i Deep Deterministic Policy Gradient. Hentet fra ***https://spinningup.openai.com/en/latest/algorithms/ddpg.html***

A.4 Grafer fra eksperimenter

A.4.1 Eksperiment 1

Figure 14: Avkastning for Dow Jones, DIA og agentens handlingsstrategi i eksperiment 1.

A.4.2 Eksperiment 2

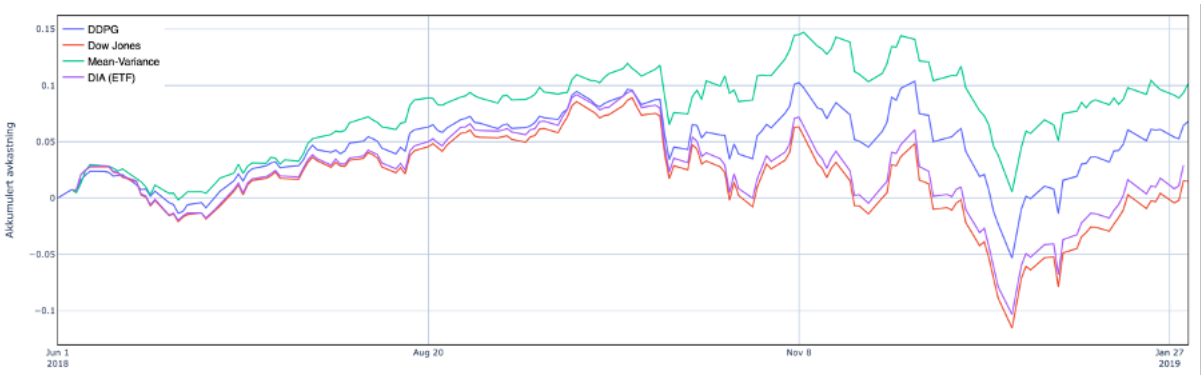


Figure 15: Avkastning for Dow Jones, DIA og agentens handlingsstrategi i eksperiment 2.

A.4.3 Eksperiment 3

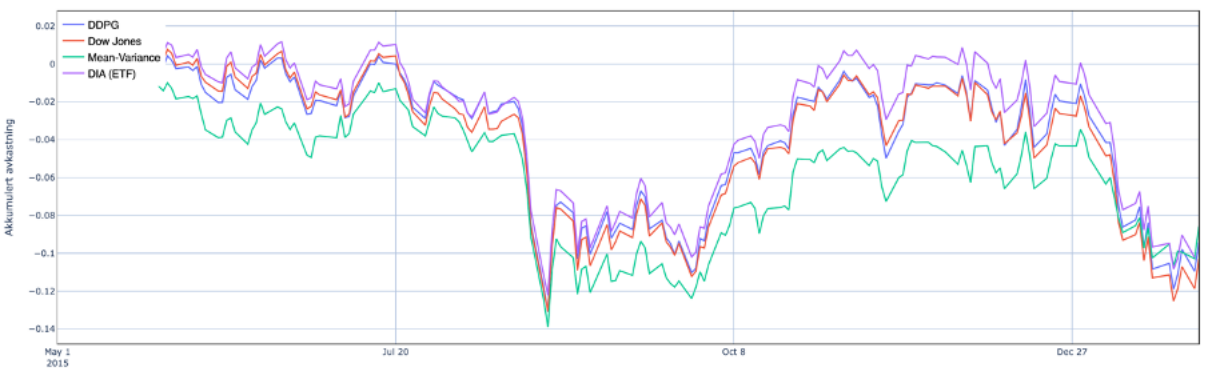


Figure 16: Avkastning for Dow Jones, DIA og agentens handlingsstrategi i eksperiment 3.

A.4.4 Eksperiment 4

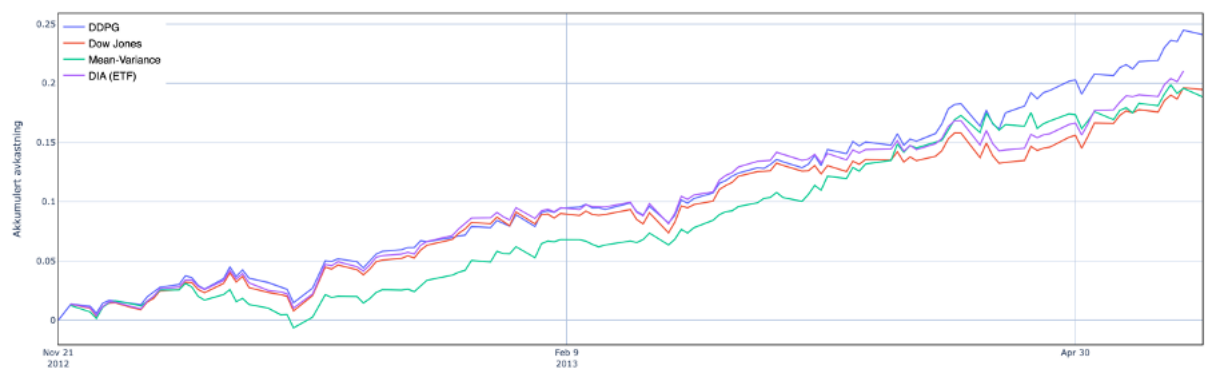


Figure 17: Avkastning for Dow Jones, DIA og agentens handlingsstrategi i eksperiment 4.

A.4.5 Eksperiment 5

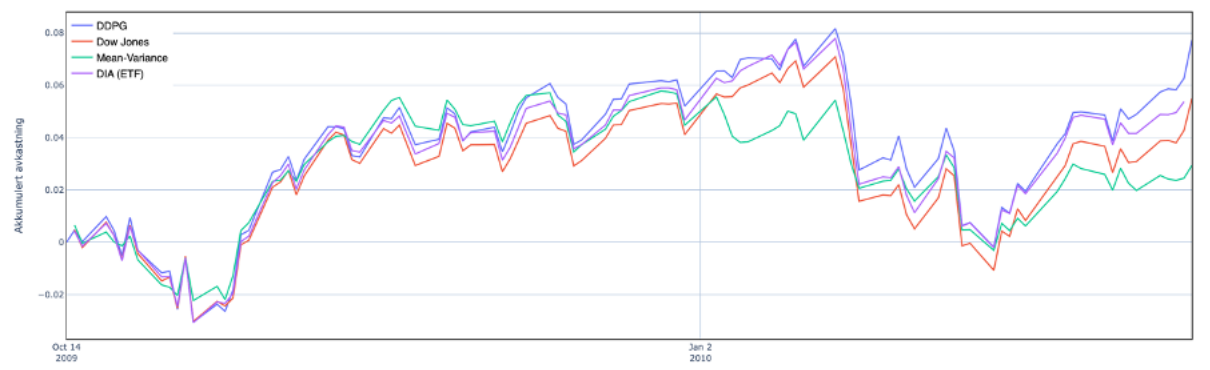


Figure 18: Avkastning for Dow Jones, DIA og agentens handlingsstrategi i eksperiment 5.

A.4.6 Eksperiment 6

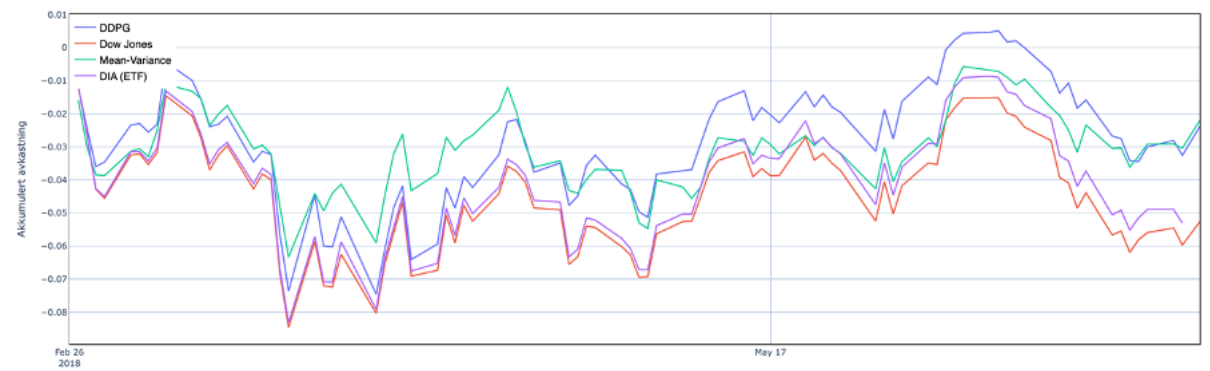
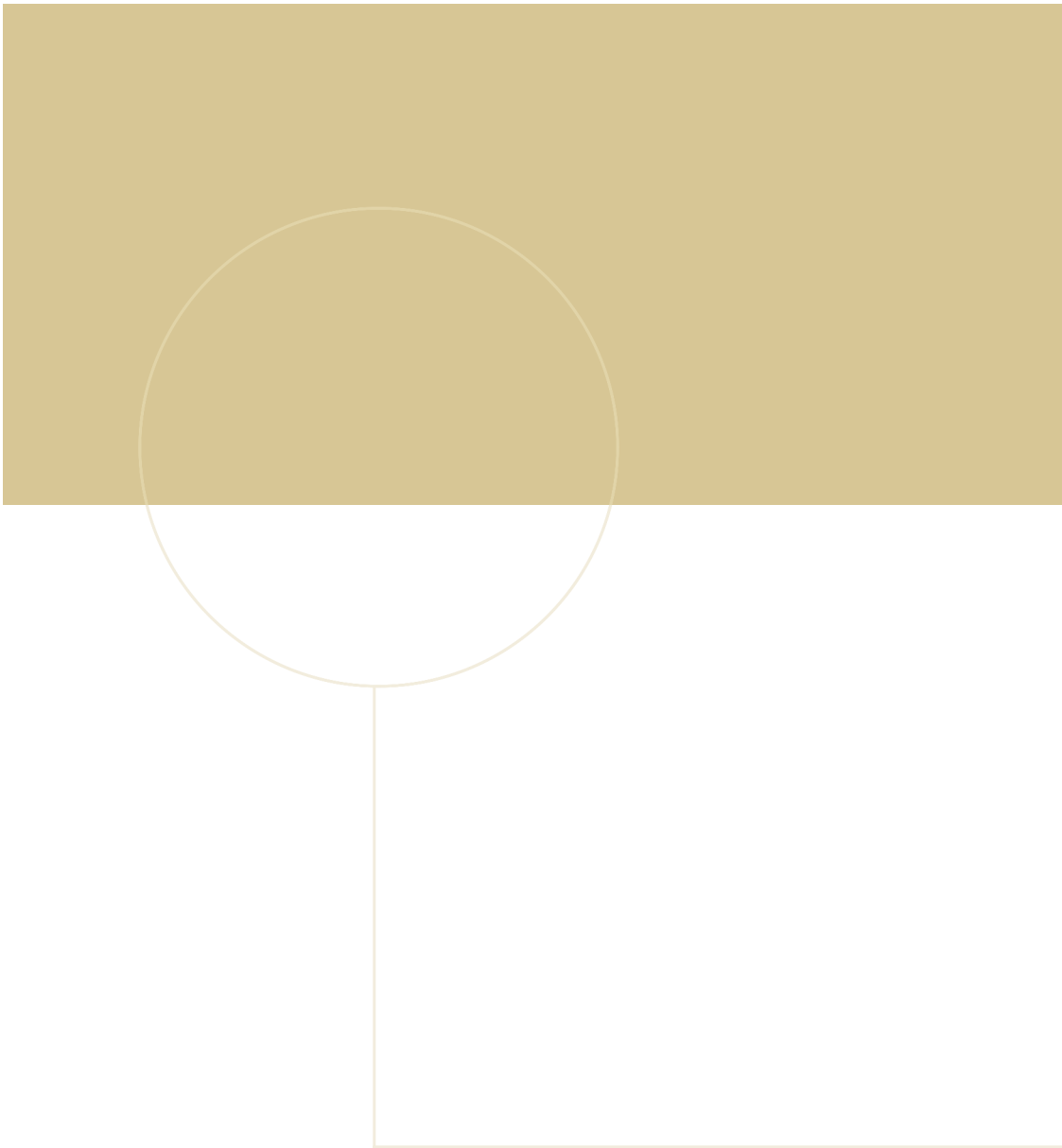


Figure 19: Avkastning for Dow Jones, DIA og agentens handlingsstrategi i eksperiment 6.

A.4.7 Eksperiment 7

Figure 20: Avkastning for Dow Jones, DIA og agentens handlingsstrategi i eksperiment 7.



NTNU

Kunnskap for en bedre verden