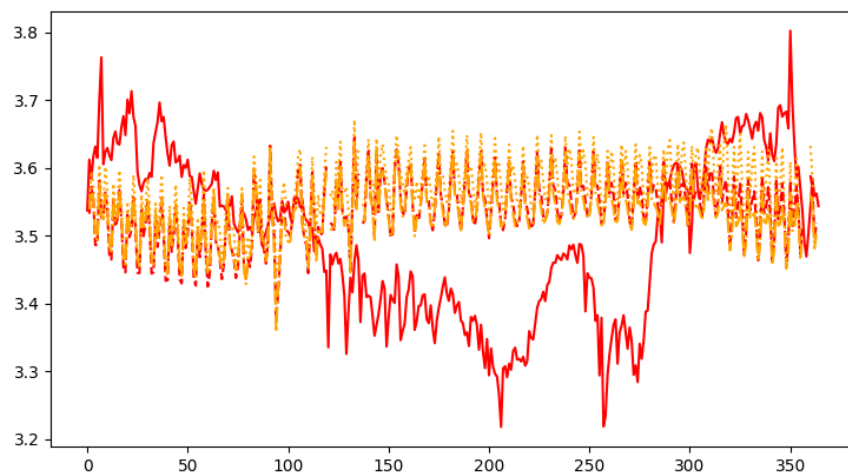


Variasjon i strømpriser

Le Quyen Vu

Våren 2023



Veileder: Torstein Kastberg Nilssen
Institutt for matematiske fag

Forord

Jeg avslutter min reise hos Universitetet i Agder med stolthet og glede ved å levere inn denne masteroppgaven. Å kombinere teoretisk matematikk med praktiske problemstillinger som er relevante for samfunnet har lenge vært et stort ønske for meg, og med denne oppgaven har jeg fått muligheten til å gjøre akkurat dette.

Denne oppgaven er blitt utarbeidet av min veileder, Torstein Kastberg Nilssen fra instituttet for matematiske fag. Jeg ønsker å gi han en stor takk for all veiledning og støtte gjennom hele året. Hans interesse for strømpriser og stokastiske differensiallikninger, som viser seg å være temaer som går hånd i hånd, har vært avgjørende for oppgaven min. Det har vært en ære for meg å ha jobbet med dette prosjektet under hans veiledning og ha fått muligheten til realisere hans visjon.

Jeg vil også takke min veileders bror, Trygve Kastberg Nilssen fra instituttet for økonomi, som også har vært en positiv innflytelse på oppgaven min. Vårt samarbeid har ført til en interessant sammenlikning mellom et matematisk og et økonomisk syn på strømpriser. Han er selv veileder for to masterstudenter i økonomi som skriver om samme tema som meg. Jeg ønsker å gi Robin Linaker Aalmen og Stian Tollefsen en stor takk for å ha bearbeidet dataene jeg trengte til oppgaven min. Deres innsats gjorde det mulig for meg å fokusere på selve matematikken og beregningene av disse dataene.

Jeg ønsker også å rette oppmerksomhet mot min fantastiske mor som har vært min største støttespiller siden jeg begynte å interessere meg for matematikk som seksåring. Hennes støtte og motivasjon har vært avgjørende for min suksess og utdanning. Dessverre gikk hun bort høsten 2022 og får ikke vært med på leveringen av masteroppgaven. Jeg vil dedikere denne oppgaven til henne og takke henne for all støtte og oppmuntring gjennom hele min skolegang og utdanning.

Til slutt vil jeg takke alle som har bidratt positivt til leveringen av min masteroppgave, og dette gjelder både faglig og personlig. Jeg leverer min siste oppgave ved Universitetet i Agder og jeg kommer til å savne det fine nettverket av studenter og forelesere. Universitetet er fullt av faglig dyktige mennesker som bryr seg om hverandre, og de har hjulpet meg med å levere mitt livs største oppgave under den tiden jeg trengte støtte mest.

Kristiansand, mai 2023

Le Quyen Vu

Abstrakt

Formålet med denne masteroppgaven er å beskrive strømprisene som en stokastisk differensiallikning med mål om å utvikle modeller som skal representere disse prisene. Disse modellene vil gi oss innsikt i strømprisenes struktur og utvikling over tid. Vi kommer til å bruke følgende stokastiske differensiallikning

$$dX_t = \theta(m_t - X_t)dt + \sigma dB_t$$

hvor θ , m_t og σ er parametere som skal estimeres slik at de passer best mulig til observerte data for strømpriser.

For å oppnå dette målet vil vi gi en innføring i grunnleggende begreper innenfor stokastiske prosesser, inkludert brownske bevegelser, martingaler og Ornstein-Uhlenbeck prosesser. Videre vil vi diskutere variasjon av stokastiske prosesser, Girsanovs teorem og maximum likelihood estimation for å utvikle metoder for å estimere parameterne i prosessen.

Abstract

The purpose of this master's thesis is to describe the electricity prices as a stochastic differential equation with the main goal to construct models that will represent these prices. These models will provide insight into the structure and development of electricity prices over time. We will use the following stochastic differential equation

$$dX_t = \theta(m_t - X_t)dt + \sigma dB_t$$

where θ , m_t and σ are parameters we are going to estimate to fit observed data for electricity prices.

To achieve this goal, we will provide an introduction to fundamental concepts in stochastic processes, including Brownian motion, martingales and Ornstein-Uhlenbeck processes. Additionally, we will discuss variation of stochastic processes, Girsanov's theorem and maximum likelihood estimation to develop methods for estimating the parameters in the process.

Innhold

Forord	i
Abstrakt	ii
Abstract	iii
1 Introduksjon	1
2 Itô kalkulus for stokastiske differensiallikninger	3
2.1 Brownske bevegelser	8
2.2 Martingaler	11
2.3 Den Radon-Nikodym deriverte	13
2.4 Girsanovs teorem	14
3 Ornstein-Uhlenbeck prosess	17
3.1 Volatiliteten: σ	20
3.1.1 Estimering av σ	23
3.2 Maximum likelihood estimation	25
3.3 Snittreverterings hastigheten: θ	29
3.4 Simulering av en Ornstein-Uhlenbeck prosess og estimering av θ	33
4 Modellering av strømprisene	39
4.1 Modell i én dimensjon	43
4.1.1 Estimere σ i én dimensjon	44
4.1.2 Funksjonen m_t : Gjennomsnittet	45
4.1.3 Estimere θ i én dimensjon	50
4.1.4 Uavhengige modeller for spotpriser og kontraktene	51
4.2 Modell i flere dimensjoner	60
4.2.1 Estimere σ i flere dimensjoner	63
4.2.2 Estimere θ i flere dimensjoner	69
4.2.3 Avhengig modell for spotpriser og kontraktene	82
4.2.4 Testing av den 4-dimensjonelle modellen	85
4.2.5 Monte Carlo-simulering	91
5 Oppsummering	95
Referanser	97

Figurer

1	Simuleringer av browniske bevegelser i ulike dimensjoner.	9
2	Ornstein-Uhlenbeck prosessen med $\sigma = 1$, $\mu = 0$ og ulike verdier for θ	17
3	Gjennomsnittlig spotpris.	18
4	Normalfordeling.	25
5	Visuell fremstilling av parameterne til funksjonen $m(t) = A + B \cos Ct$	33
6	Gjennomsnittlig spotpris plottet sammen med cosinusfunksjonen.	34
7	Simuleringer av Ornstein-Uhlenbeck prosessen med cosinusfunksjonen som driftsledd og ulike verdier for θ	36
8	Simulering av Ornstein-Uhlenbeck prosessen med 5 som driftsledd og $\theta = 2$	38
9	Gjennomsnittlig logaritmisk spotpris.	46
10	Gjennomsnittlige kontraktspriser.	48
11	Gjennomsnittlig spotpris sammen med gjennomsnittlige kontraktspriser.	49
12	Estimerte spotpriser sammenlignet med gjennomsnittlig spotpris.	52
13	1000 estimeringer av spotprisene.	53
14	Estimerte 1-årskontrakter sammenlignet med gjennomsnittlig 1-årskontrakt.	54
15	1000 estimeringer av 1-årskontraktene.	55
16	Estimerte 2-årskontrakter sammenlignet med gjennomsnittlig 2-årskontrakt.	56
17	1000 estimeringer av 2-årskontraktene.	57
18	Estimerte 3-årskontrakter sammenlignet med gjennomsnittlig 3-årskontrakt.	58
19	1000 estimeringer av 3-årskontraktene.	59
20	Estimeringer av strømprisene.	84
21	Faktiske strømpriser målt fra 2007 til 2019.	85
22	Estimerte strømpriser for 13 år.	86
23	Fordelinger av de syntetiske dataene for strømpriser.	92

1 Introduksjon

Strømpriser har stor påvirkning på samfunnet da de påvirker blant annet næringslivet, miljøet og husholdningers økonomi. Å estimere strømpriser kan være en utfordring da de avhenger av mange faktorer og viser stor variasjon over tid. Dette gjør det tilnærmet umulig å si sikkert hva de fremtidige prisene vil være. En indikator er at spotprisene viser typisk sesongvariasjon med lavere priser på sommeren og høyere priser på vinteren, men dette er ikke en fasit. Spotprisene kan likevel endre seg raskt og uforutsigbart.

Vi har også andre priser å ta hensyn til, som prisene for årskontrakt på strøm. Dette er en avtale mellom en strømleverandør og en kunde som garanterer en fast pris på strøm i en bestemt tidsperiode, vanligvis ett, to eller tre år. Årskontrakter for strømpriser kan gi forutsigbarhet og stabilitet for kunder som ønsker å unngå svingninger i spotprisene, men det kan også være en risiko hvis markedet plutselig faller og kunden blir sittende igjen med en høyere pris enn den som tilbys i markedet. Kontraktsprisene viser ikke samme sesongvariasjon som spotprisene, men de er likevel uforutsigbare og vi kan aldri garantere for at vi får en god pris på kontrakten.

Det er flere faktorer som påvirker strømprisene, blant annet tilbud, etterspørsel og værforhold, som gjør at de kan svinge betydelig over tid. Funksjonen som måler strømprisene vil derfor oppføre seg svært irregulært. En av årsakene til den irregulære og uforutsigbare strukturen som vi kan observere i strømprisene, er at prisene kan beskrives ved hjelp av en stokastisk differensiallikning som inneholder et ledd med en brownsk bevegelse.

Brownske bevegelser er en viktig kilde til opphavet til rammeverket som kalles Itô kalkulus. Itô kalkulus er en utvidelse av vanlig kalkulus som er spesielt utviklet for å håndtere stokastiske prosesser, som er prosesser der tilfeldighet spiller en rolle. Det er derfor en god idé å uttrykke strømpriser som en stokastisk prosess på grunn av strømprisenes tilfeldighet.

I denne masteroppgaven skal det utvikles en modell som skal estimere strømprisene. Denne modellen skal ha fire ukjente da vi ønsker å estimere spotpriser, 1-årskontrakter, 2-årskontrakter og 3-årskontrakter. Som nevnt vil det være tilnærmet umulig å forutse disse prisene på en presis måte, så istedenfor å fokusere på å gi nøyakte estimater av fremtidige strømpriser, vil oppgaven heller fokusere på å undersøke de matematiske egenskapene som ligger bak. Oppgaven vil være en kombinasjon av sannsynlighetsteori, Itô kalkulus for stokastiske differensiallikninger, statistikk, dataanalyse og simuleringer. Jeg vil studere spørsmål som hvor tilfeldige strømprisene egentlig er, hva slags prosess de kan beskrives som, hvor stor variasjonen er, og hvor raskt prisene endrer seg.

Vi skal bruke denne stokastiske differensiallikningen

$$dX_t = \theta(m_t - X_t)dt + \sigma dB_t \quad (1.1)$$

hvor m_t skal være en funksjon som gjør at vi lettere kan fange opp sesongvariasjonen til spotprisene. Likningen vil ha en deterministisk del som beskriver den gjennomsnittlige prisen, og en stokastisk del som beskriver usikkerheten i prisene. Basert på faktiske strømpriser i norden for 13 år, skal oppgaven estimere parametere som gjør at løsningen av (1.1) passer best mulig til observerte data.

2 Itô kalkulus for stokastiske differensiallikninger

Vi ønsker å beskrive strømprisene som en stokastisk differensiallikning. Slike likninger er beskrivelser av tilfeldige prosesser som endrer seg over tid og kan brukes til å modellere mange forskjellige systemer, inkludert strømpriser. For å kunne håndtere stokastiske differensiallikninger, trenger vi en spesiell type kalkulus som tar høyde for tilfeldighet.

Denne typen kalkulus kalles for Itô kalkulus og er en utvidelse av vanlig kalkulus som vi bruker til å beskrive deterministiske prosesser, og den gir oss verktøyene vi trenger for å håndtere tilfeldighet og usikkerhet. I det neste kapitlet vil jeg derfor gi en kort introduksjon til Itô kalkulus og hvordan vi kan bruke det til å håndtere stokastiske differensiallikninger. Definisjoner og teoremer som vil bli presentert i dette kapitlet er for det meste hentet fra Øksendal, B. (2003). *Stochastic Differential Equations: An Introduction with Applications* (6. utg.). Springer, samt fra Shreve, S. E. (2010). *Stochastic Calculus for Finance II: Continuous-Time Models*. Springer.

Som nevnt oppfører funksjonen for strømpriser seg veldig irregulært, som er et resultat av at strømprisene hopper opp og ned konstant. Den såkalte Ornstein-Uhlenbeck prosessen blir ofte brukt for å modellere strømprisene, som er en løsning av en stokastisk differensiallikning, nærmere bestemt likningen

$$dX_t = \theta(\mu - X_t)dt + \sigma dB_t \quad (2.1)$$

for gitte valg av θ , μ , og σ .

I motsetning til hvordan løsninger av differensiallikninger man støter på under utdanningskurs som er på lavere gradsnivå, ser vi at løsningen har den samme typen irregulære strukturen som vi kan observere i strømprisene. Det kommer av leddet som inneholder B_t som kalles en brownsk bevegelse. Noe av det som gjør den brownske bevegelsen så interessant er at den ikke er deriverbar. Siden (2.1) avhenger av den deriverte til den brownske bevegelsen, dB_t , må man først forklare hva denne likningen i det hele tatt skal bety før man kan snakke om å finne en løsning av den.

Rammeverket for å forklare dette ble funnet opp på 1950-tallet og kalles for Itô kalkulus. Itô kalkulus er oppkalt etter Kiyosi Itô og utvider metodene i kalkulus til å også gjelde for stokastiske prosesser, slik som brownske bevegelser. Itô kalkulus har viktige anvendelser i blant annet matematisk finans og stokastiske differensiallikninger.

Stokastiske differensiallikninger skal beskrive utviklingen av en tilfeldig variabel over tid ved å ta hensyn til endringer som skyldes både systematisk påvirkning og tilfeldige variabler. Dette gjør at slike likninger inneholder både deterministiske og stokastiske komponenter. Det stokastiske leddet i likningen modelleres vanligvis ved hjelp av brownske bevegelser eller andre typer stokastiske prosesser. Itô kalkulus er spesielt egnet til å beskrive og analysere brownske bevegelser og andre stokastiske prosesser som ligner på disse.

Den tilfeldige variabelen som den stokastiske differensiallikningen skal beskrive endringen til, kalles for en stokastisk prosess. En stokastisk prosess er en matematisk modell, altså en beskrivelse, av en tilfeldig variabel som endrer seg over tid. Så mens en stokastisk prosess er en beskrivelse av hvordan en tilfeldig variabel endrer seg over tid, er en stokastisk differensiallikning en matematisk ligning som beskriver denne endringen over tid og hvordan den påvirkes av tilfeldige faktorer. Vi definerer en stokastisk prosess tydeligere:

Definisjon 2.1 (Stokastisk prosess) *En stokastisk prosess er en familie av tilfeldige variabler som er parametrisert av T , med andre ord*

$$\{X_t\}_{t \in T} \tag{2.2}$$

og som er definert på et sannsynlighetsrom (Ω, \mathcal{F}, P) (Øksendal, 2003, s. 10).

Øksendal (2003, s. 7-8) definerer et sannsynlighetsrom, samt begrepene σ -algebra, målbart rom og sannsynlighetsmål, på følgende måte:

Definisjon 2.2 *La Ω være et gitt sett. Da er σ -algebraen \mathcal{F} på Ω en familie av delmengder av Ω med følgende egenskaper:*

- (i) $\emptyset \in \mathcal{F}$,
- (ii) $F \in \mathcal{F} \implies F^c \in \mathcal{F}$ hvor $F^c = \Omega \setminus F$ er komplementet av F i Ω ,
- (iii) $A_1, A_2, \dots \in \mathcal{F} \implies A := \bigcup_{i=1}^{\infty} A_i \in \mathcal{F}$.

Paret (Ω, \mathcal{F}) kalles et målbart rom. Et sannsynlighetsmål \mathbb{P} på et målbart rom (Ω, \mathcal{F}) er en funksjon $\mathbb{P} : \mathcal{F} \rightarrow [0, 1]$ slik at

(a) $\mathbb{P}(\emptyset) = 0$, $\mathbb{P}(\Omega) = 1$,

(b) hvis $A_1, A_2, \dots \in \mathcal{F}$ og $\{A_i\}_{i=1}^{\infty}$ er disjunkte (med andre ord $A_i \cap A_j = \emptyset$ hvis $i \neq j$), da gjelder $\mathbb{P}\left(\bigcup_{i=1}^{\infty} A_i\right) = \sum_{i=1}^{\infty} \mathbb{P}(A_i)$.

Trippelen $(\Omega, \mathcal{F}, \mathbb{P})$ kalles et sannsynlighetsrom.

En stokastisk differensiallikning er en matematisk beskrivelse av en stokastisk prosess som kan brukes til å simulere eller forutsi utviklingen av en tilfeldig variabel eller systemet det beskriver. For vår del, er dette systemet strømpriser som inkluderer spotpriser, 1-årskontrakter, 2-årskontrakter og 3-årskontrakter.

Stokastiske differensiallikninger har vanligvis formen

$$dX_t = b(t, X_t)dt + \sigma(t, X_t)dB_t, \quad b(t, x) \in \mathbb{R}, \sigma(t, x) \in \mathbb{R}, \quad (2.3)$$

hvor $b(t, X_t)$ er en deterministisk funksjon som beskriver den forventede endringen av X_t i løpet av en tidsenhet dt , $\sigma(t, X_t)$ er en deterministisk funksjon som beskriver hvor mye variasjonen i X_t vil endre seg i løpet av en tidsenhet dt og B_t er en brownisk bevegelse som representerer det tilfeldige støyet i systemet (Øksendal, 2003, s. 66).

En stokastisk prosess som kan beskrives ved hjelp av en stokastisk differensiallikning som inneholder et ledd med en brownisk bevegelse, kalles for Itô prosess. La X_t være en stokastisk prosess. X_t er også en Itô prosess hvis den er av denne formen

$$X_t = X_0 + \int_0^t u(s, w)ds + \int_0^t v(s, w)dB_s \quad (2.4)$$

(Øksendal, 2003, s. 44).

En Itô prosess er altså definert til å være en adaptert prosess (en prosess som ikke kan se i fremtiden) som kan uttrykkes som summen av et integral med hensyn til tid og et integral med hensyn til en brownisk bevegelse. Et viktig verktøy i stokastisk kalkulus for å håndtere stokastiske differensiallikninger som involverer browniske bevegelser, altså Itô prosesser, er Itôs formell. Itôs formell er en måte å derivere Itô prosesser på da det ikke holder med derivasjonsreglene fra vanlig kalkulus.

Teorem 2.1 (Itô's formell) La $g(t, x) \in C^2([0, \infty) \times \mathbb{R})$ (g er kontinuerlig og deriverbar to ganger) og

$$dX_t = udt + vdB_t$$

hvor X_t er en Itô prosess. Definer

$$Y_t = g(t, X_t)$$

hvor Y_t også er en Itô prosess. Da har vi at

$$dY_t = \frac{\partial g}{\partial t}(t, X_t)dt + \frac{\partial g}{\partial x}(t, X_t)dX_t + \frac{1}{2} \frac{\partial^2 g}{\partial x^2}(t, X_t)(dX_t)^2 \quad (2.5)$$

hvor $(dX_t)^2 = (dX_t) \times (dX_t)$ er beregnet etter følgende regler

\cdot	dt	dB_t
dt	0	0
dB_t	0	dt

(Øksendal, 2003, s. 44-45).

En vesentlig egenskap ved browniske bevegelser er at disse prosessene ikke kan deriveres på vanlig vis. Itô's formell gir oss en måte å beregne den totale endringen i en stokastisk prosess på, som da gjør det mulig for oss å også "derivere" browniske bevegelser.

Browniske bevegelser og andre stokastiske prosesser er preget av tilfeldighet og usikkerhet, og variasjon spiller derfor en vesentlig rolle i modelleringen av slike prosesser. En måte å måle variasjonene til stokastiske prosesser på er ved kvadratisk variasjon. Kvadratisk variasjon kan være nyttig for å analysere støy i data og å måle volatiliteten, som er noe vi skal se på senere i oppgaven. Vi definerer den kvadratiske variasjonen av en stokastisk prosess X_t på følgende måte

Definisjon 2.3 (Kvadratisk variasjon) Anta at vi har en stokastisk prosess $\{X_t\}_{t \in [0, T]}$. Den kvadratiske variasjonen av X_t er definert som

$$\langle X \rangle_t = \lim_{\|\pi\| \rightarrow 0} \sum_{i=0}^n (X_{t_{i+1}} - X_{t_i})^2 \quad (2.6)$$

over en partisjon $\pi = \{t_0, t_1, \dots, t_n\}$ hvor $\|\pi\| = \max_j |t_{j+1} - t_j| \rightarrow 0$ og $0 = t_0 \leq t_1 \leq \dots \leq t_n = T$ (Øksendal, 2003, s. 57).

Kvadratisk variasjon måler variasjonen av en enkelt stokastisk prosess over tid, men vi kan også måle hvordan to prosesser varierer sammen over tid ved å måle kovariasjonen mellom prosessene. Kovariasjonen mellom to tilfeldige variabler X_t og Y_t er et mål på hvor mye av variasjonen i X_t korrelerer med variasjonen i Y_t , og defineres slik

Definisjon 2.4 (Kovariasjon) *Anta at vi har to stokastiske prosesser $\{X_t\}_{t \in [0, T]}$ og $\{Y_t\}_{t \in [0, T]}$. Kovariasjonen mellom X_t og Y_t er definert som*

$$\langle X, Y \rangle_t = \lim_{\|\pi\| \rightarrow 0} \sum_{i=0}^n (X_{t_{i+1}} - X_{t_i}) (Y_{t_{i+1}} - Y_{t_i}) \quad (2.7)$$

over en partisjon $\pi = \{t_0, t_1, \dots, t_n\}$ hvor $\|\pi\| = \max_j |t_{j+1} - t_j| \rightarrow 0$ og $0 = t_0 \leq t_1 \dots \leq t_n = T$.

2.1 Brownske bevegelser

Brownske bevegelser spiller en svært viktig rolle i Itô kalkulus. Vi kan se på brownske bevegelser som stokastiske prosesser hvor bevegelsene er uregelmessige og tilfeldige. Itô kalkulus brukes ofte til å modellere priser på finansielle markeder hvor prisendringene er stokastiske og påvirkes av tilfeldige hendelser. Brownske bevegelser er en sentral ingrediens i slike modeller, og kan brukes til å beskrive hvordan priser beveger seg tilfeldig over tid.

En viktig egenskap ved brownske bevegelser er at de er kontinuerlige, men ikke differensierbare. Disse bevegelsene er uforutsigbare og tilfeldige som gjør at de ikke kan beskrives ved bruk av vanlig derivasjon. Det er her Itô kalkulus spiller inn for å håndtere disse ikke-differensierbare stokastiske prosessene. Brownske bevegelser kan vi definere slik:

Definisjon 2.5 (Brownsk bevegelse) *En stokastisk prosess $\{B_t\}_{t \in [0, \infty)}$ er en brownsk bevegelse hvis følgende er oppfylt:*

- $P(B_0 = x) = 1 \quad \forall x \in \mathbb{R}^d$,
- $B_t - B_s$ er uavhengig av $\mathcal{F}_s \quad \forall t \geq s$,
- $B_t - B_s \sim \mathcal{N}(0, (t - s)\mathcal{I})$ (Shreve, 2010, s. 94).

Vi kommer til å jobbe mye med brownske bevegelser og flere av simulasjonene vi skal lage kommer til å inneholde denne bevegelsen. Vi trenger derfor å kunne simulere en brownsk bevegelse i Python slik at vi kan bruke dette skriptet videre i andre simulasjoner.

```
1 import matplotlib.pyplot as plt
2 import numpy as np
3
4 def simulate_BM(d, n, T):
5     t = np.linspace(0, T, n)
6     BM = np.zeros((d,n))
7
8     for i in range(n-1):
9         BM[:,i+1] = BM[:,i] + np.sqrt(abs(t[i+1] - t[i])) *
10         np.random.randn(d)
11     return BM
```

Denne funksjonen vil simulere en d -dimensjonell brownsk bevegelse med n steg fra 0 til T . Nedenfor ser vi en 1-dimensjonell, 2-dimensjonell og 3-dimensjonell brownsk bevegelse hvor alle har $n = 500$ og $T = 10$.

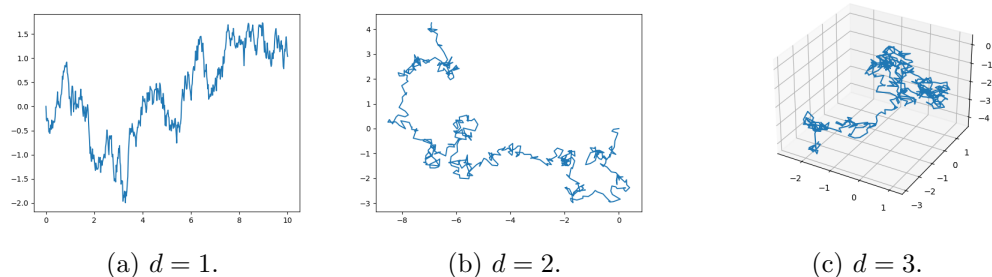


Figure 1: Simuleringer av browniske bevegelser i ulike dimensjoner.

Nå som vi har skriptet klart, kan det enkelt importeres og gjenbrukes til andre skript. Dette skriptet simulerer selve den browniske bevegelsen, men noe vi også kommer til å trenge, er inkrementene til bevegelsen. Ved definisjonen av en brownsk bevegelse ser vi at tredje punktet beskriver inkrementenes egenskaper.

Inkrementene har egenskapen at de er uavhengige, mens selve bevegelsen derimot er avhengig av tidligere punkter da den er definert ved hjelp av forrige skritt i bevegelsen. Inkrementene vil derfor være mer egnet til simuleringer av stokastiske prosesser hvor man ønsker å modellere uforutsigbare endringer over tid.

Vi kan se på inkrementene til en brownsk bevegelse som en tilfeldig prosess med gjennomsnitt lik 0 og ingen korrelasjon mellom verdier på ulike tidspunkter. Dette er også egenskapene til hvitt støy. Hvitt støy er formelt definert som den deriverte til den browniske bevegelsen (Øksendal, 2003, s. 65), altså dB_t , og vi kan se på det som inkrementene til den browniske bevegelsen.

Vi kommer til å se på flere stokastiske differensiallikninger som er drevet av hvitt støy, så vi trenger et skript som simulerer dette. Skriptet vil være svært lik skriptet som simulerer den browniske bevegelsen, men vi legger ikke til det forrige punktet i prosessen. Vi legger bare til inkrementene.

```
1 import matplotlib.pyplot as plt
2 import numpy as np
3
4 def simulate_WN(d, n, T):
5     t = np.linspace(0, T, n)
6     WN = np.zeros((d,n))
7
8     for i in range(n-1):
9         WN[:,i+1] = np.sqrt(abs(t[i+1] - t[i])) * np.random.
randn(d)
10     return WN
```

2.2 Martingaler

En annen viktig stokastisk prosess i Itô kalkulus er martingaler. En martingal er en sekvens av tilfeldige variabler, altså en stokastisk prosess, hvor på et bestemt tidspunkt, så er forventningsverdien gitt all tidligere informasjon lik den nåværende verdien.

Dette gir martingaler en viktig egenskap, nemlig at de er uforutsigbare. De er tilfeldige prosesser som ikke har noen systematisk trend eller forutsigbarhet i tiden. Dette gjør dem spesielt egnet til å modellere stokastiske prosesser som har en tilfeldig eller uforutsigbar karakter.

Vi definerer martingaler på følgende måte:

Definisjon 2.6 (Martingal) *Anta at*

- X_t er \mathcal{F}_t -målbar $\forall t$,
- $E[|X_t|] < \infty \quad \forall t$,
- $E[X_t | \mathcal{F}_s] = X_s \quad \forall s \geq t$.

Da er den stokastiske prosessen $\{X_t\}_{t \geq 0}$ en martingal med hensyn på filtrasjonen $\{\mathcal{F}_t\}_{t \geq 0}$ (Øksendal, 2003, s. 31).

Den siste egenskapen er spesielt viktig for å forutse fremtidige utfall, som er det vi skal prøve å gjøre med modellen for strømpriser. Martingaler brukes ofte til å modellere og analysere tilfeldige prosesser der tidligere informasjon kan gi informasjon om fremtiden. Tidligere informasjon av en stokastisk prosess kan brukes til å forutse fremtidige utfall, men bare til en viss grad. Vi skal bruke historisk data for strømpriser, som da er tidligere informasjon, til å prøve å si noe om fremtidige priser.

En brownsk bevegelse er en martingal med kontinuerlige stier, som vil si at prosessens bevegelser er en kontinuerlig funksjon som er avhengig av tid. Lévy's karakteriseringsteorem for en brownsk bevegelse gir kriteriene for å bestemme om en stokastisk prosess er en brownsk bevegelse basert på dens martingalegenskaper og kvadratiske variasjon.

Teorem 2.2 (Lévy's karakteriseringsteorem for én dimensjon)

Hvis en stokastisk prosess $\{M_t\}_{t \geq 0}$ er en martingal med hensyn på en filtrasjon $\{\mathcal{F}_t\}_{t \geq 0}$ og oppfyller følgende krav

- $M_0 = 0$,
- M_t har kontinuerte stier,
- $\langle M \rangle_t = t \quad \forall t \geq 0$,

da er M_t en 1-dimensjonell brownsk bevegelse (Shreve, 2010, s. 168).

Lévy's karakteriseringsteorem har også en utvidet versjon som gjør det mulig å identifisere to uavhengige brownske bevegelser. Nedenfor ser vi den 2-dimensjonelle versjonen av teoremet.

Teorem 2.3 (Lévy's karakteriseringsteorem for to dimensjoner)

Hvis to stokastiske prosesser $\{M_t^{(0)}\}_{t \geq 0}$ og $\{M_t^{(1)}\}_{t \geq 0}$ er martingaler med hensyn på en filtrasjon $\{\mathcal{F}_t\}_{t \geq 0}$ og oppfyller følgende krav for $i \in \{0, 1\}$

- $M_0^{(i)} = 0$,
- $M_t^{(i)}$ har kontinuerte stier,
- $\langle M^{(i)} \rangle_t = t \quad \forall t \geq 0$,

da er $M_t^{(0)}$ og $M_t^{(1)}$ uavhengige brownske bevegelser hvis i tillegg $\langle M^{(0)}, M^{(1)} \rangle_t = 0$ for alle $t \geq 0$ (Shreve, 2010, s. 169).

Disse teoremene kan brukes både for å identifisere en gitt stokastisk prosess som en brownsk bevegelse, og for å verifisere at en brownsk bevegelse oppfyller kravene i teoremene.

2.3 Den Radon-Nikodym deriverte

Når vi jobber med modellering av finansielle data som strømpriser, er det ofte nyttig å kunne forstå hvordan ulike sannsynlighetsmål forholder seg til hverandre. En måte å uttrykke denne sammenhengen på, er ved den Radon-Nikodym deriverte.

Den Radon-Nikodym deriverte uttrykker forholdet mellom to sannsynlighetsmål og gjør det mulig å sammenligne hvordan ulike hendelser inntreffer under ulike mål. Under visse vilkår kan et mål \mathbb{Q} uttrykkes med hensyn til et annet mål \mathbb{P} . Et av vilkårene er absolutt kontinuitet.

Definisjon 2.7 (Absolutt kontinuitet) *La \mathbb{Q} være et sannsynlighetsmål på (Ω, \mathcal{F}) . \mathbb{Q} er absolutt kontinuerlig med hensyn på \mathbb{P} hvis*

$$\mathbb{P}(A) = 0 \Rightarrow \mathbb{Q}(A) = 0 \quad \forall \text{ målbare mengder } A \in \mathcal{F}.$$

Vi bruker notasjonen $\mathbb{Q} \ll \mathbb{P}$ (Shreve, 2010, s. 45).

Hvis \mathbb{Q} er absolutt kontinuerlig med hensyn på et positivt mål \mathbb{P} , kan vi skrive \mathbb{Q} som

$$\mathbb{Q}(A) = \int_A M d\mathbb{P}. \quad (2.8)$$

Funksjonen M er den Radon-Nikodym deriverte av de to målene og uttrykkes som

$$M = \frac{d\mathbb{Q}}{d\mathbb{P}}. \quad (2.9)$$

Den Radon-Nikodym deriverte spiller en viktig rolle i Girsanovs teorem, som vi skal se på senere. Girsanovs teorem gir oss en måte å analysere hvordan stokastiske prosesser endrer seg under bytte av mål.

2.4 Girsanovs teorem

Girsanovs teorem bruker den Radon-Nikodym deriverte for å vise hvordan vi kan justere for endringer i sannsynlighetsmålet når vi går fra det ene målet til det andre. Dette teoremet spiller en viktig rolle i stokastisk analyse, men er også svært viktig i andre anvendelser, som for eksempel økonomi. Girsanovs teorem er et viktig resultat i blant annet finansiell modellering, som innebærer å lage en representasjon, eller modell, av en ekte finansiell situasjon. I vårt tilfelle bruker vi historisk data til å lage en modell som skal estimere strømpriser.

Teoremet beskriver hvordan fordelingen til stokastiske prosesser endrer seg under endring av mål. Et sannsynlighetsmål beskriver sannsynligheten for de ulike verdiene av en stokastisk prosess. Hvis vi endrer sannsynlighetsmålet, vil det endre måten vi vurderer sannsynligheten for de ulike verdiene av prosessen på. Dette teoremet har flere formuleringer og Øksendal gir tre forskjellige formuleringer av teoremet som tar for seg ulike aspekter. Vi kommer til å bruke to av de, og vi ser nedenfor Girsanovs teorem del I og del II formulert av Øksendal (2003, s. 165-168).

Teorem 2.4 (Girsanovs teorem (del I)) *La $Y_t \in \mathbb{R}^d$ være en Itô prosess av formen*

$$dY_t = a(t, \omega)dt + dB_t; \quad t \leq T, Y_0 = 0 \quad (2.10)$$

hvor $T \leq \infty$ er en gitt konstant og B_t er en d -dimensjonell brownsk bevegelse. Definer

$$M_t = \exp\left(-\int_0^t a(s, \omega)dB_s - \frac{1}{2}\int_0^t a^2(s, \omega)ds\right); \quad 0 \leq t \leq T. \quad (2.11)$$

Anta at M_t er en martingal med hensyn på $\mathcal{F}_t^{(d)}$ og \mathbb{P} . Definer målet \mathbb{Q} på \mathcal{F}_T ved

$$d\mathbb{Q}(\omega) = M_T(\omega)d\mathbb{P}(\omega). \quad (2.12)$$

Da er \mathbb{Q} et sannsynlighetsmål på $\mathcal{F}_T^{(n)}$ og Y_t er en d -dimensjonell brownsk bevegelse med hensyn på \mathbb{Q} for $0 \leq t \leq T$.

Denne delen av teoremet introduserer en Itô prosess Y_t som inkluderer et deterministisk ledd $a(t, \omega)dt$ og et stokastisk ledd dB_t hvor B_t er en d-dimensjonell brownsk bevegelse. Teoremet definerer også M_t som er en funksjon av a og B_t . M_t antas å være en martingal med hensyn på filtreringen $\mathcal{F}_t^{(d)}$ og sannsynlighetsmålet \mathbb{P} . Sannsynlighetsmålet \mathbb{Q} kan defineres med hensyn på \mathbb{P} og M_T .

Når man betrakter Y_t i henhold til det nye sannsynlighetsmålet \mathbb{Q} , så vil prosessen bli betraktet som en d-dimensjonell brownsk bevegelse for $0 \leq t \leq T$. Dette betyr at Y_t vil ha egenskaper som en brownsk bevegelse under dette nye sannsynlighetsmålet \mathbb{Q} .

Teorem 2.5 (Girsanovs teorem (del II)) *La $Y_t \in \mathbb{R}^n$ være en Itô prosess av denne formen*

$$dY_t = b(t; \theta)dt + \sigma(t)dB_t; \quad t \leq T \quad (2.13)$$

hvor $B_t \in \mathbb{R}^m$, $b(t; \theta) \in \mathbb{R}^n$ og $\sigma(t) \in \mathbb{R}^{n \times m}$. Anta at det finnes prosesser $u(t; \theta) \in \mathcal{W}_{\mathcal{H}}^m$ og $\alpha(t; \theta) \in \mathcal{W}_{\mathcal{H}}^n$ slik at

$$\sigma(t)u(t; \theta) = b(t; \theta) - \alpha(t; \theta). \quad (2.14)$$

Definer

$$M_t = \exp \left(- \int_0^t u(s; \theta)dB_s - \frac{1}{2} \int_0^t u^2(s; \theta)ds \right); \quad t \leq T \quad (2.15)$$

og

$$d\mathbb{Q} = M_T d\mathbb{P} \quad \text{på } \mathcal{F}_T^{(m)}. \quad (2.16)$$

Anta at M_t er en martingal med hensyn på $\mathcal{F}_t^{(n)}$ og \mathbb{P} . Da er \mathbb{Q} et sannsynlighetsmål på $\mathcal{F}_T^{(m)}$, prosessen

$$\hat{B}_t := \int_0^t u(s; \theta)ds + B_t; \quad t \leq T \quad (2.17)$$

er en brownsk bevegelse med hensyn på \mathbb{Q} og i form av \hat{B}_t har prosessen Y_t følgende stokastiske representasjon

$$dY_t = \alpha(t; \theta)dt + b(t; \theta)d\hat{B}_t. \quad (2.18)$$

Del II av teoremet starter også med en Itô prosess Y_t som er av en litt annerledes form enn prosessen i del I. Det antas at det eksisterer to stokastiske prosesser $u(t; \theta)$ og $\alpha(t; \theta)$ som oppfyller en spesiell betingelse. Produktet av $b(t; \theta)$ og $u(t; \theta)$ skal være lik differansen mellom $b(t; \theta)$ og $\alpha(t; \theta)$.

Vi definerer igjen M_t som antas å være en martingal med hensyn på filtreringen $\mathcal{F}_t^{(n)}$ og målet \mathbb{P} . Vi introduserer det nye målet \mathbb{Q} som er proporsjonalt med det opprinnelige målet \mathbb{P} ved hjelp av M_T .

Det defineres også en ny prosess \hat{B}_t som er en brownsk bevegelse med hensyn på det nye målet \mathbb{Q} . Prosessen Y_t kan uttrykkes med hensyn på den nye brownske bevegelsen med hensyn på \mathbb{Q} og følger den stokastiske representasjonen gitt i likning (2.18).

Begge delene av teoremet gir oss en måte å endre sannsynlighetsmålet til en stokastisk prosess på. Vi definerer martingalen M_t med hensyn på det opprinnelige målet \mathbb{P} slik at vi kan definere et nytt mål \mathbb{Q} . Vi legger merke til at M_t vil være lik den Radon-Nikodym deriverte av de to målene

$$\frac{d\mathbb{Q}}{d\mathbb{P}} = M_t. \quad (2.19)$$

Begge teoremene gir oss betingelsene for å konstruere et nytt sannsynlighetsmål \mathbb{Q} fra det opprinnelige målet \mathbb{P} , slik at den opprinnelige Itô-prosessen Y_t kan sees på som en brownsk bevegelse med hensyn på \mathbb{Q} . Forskjellen mellom del I og del II ligger i hvilke betingelser som gis og hvordan prosessene defineres.

3 Ornstein-Uhlenbeck prosess

Ornstein-Uhlenbeck prosessen blir ofte brukt for å modellere strømprisene. Denne prosessen er en løsning av følgende stokastiske differensiallikning

$$dX_t = \theta(\mu - X_t)dt + \sigma dB_t. \quad (2.1)$$

Prosesen er kjent som en snittreverterende prosess, som vil si at den har en tendens til å revertere tilbake til et punkt, som regel gjennomsnittet, over tid.

Ornstein-Uhlenbeck prosessen er en stokastisk prosess som kan beskrives som en tilfeldig vandring som har en tendens til å trekke seg tilbake mot et bestemt punkt, og med en større tiltrekning når prosessen er lenger unna dette punktet. Vi kan derfor se på prosessen som en modifikasjon av den browniske bevegelsen som er en tilfeldig vandring uten noen korrigeringer.

Det finnes flere typer Ornstein-Uhlenbeck prosesser, men for prosessen (2.1), så vil bevegelsen reverteres til punktet μ , som vi kaller for driftsleddet. θ er koeffisienten som bestemmer hvilken hastighet prosessen skal reverteres tilbake til μ . Nedenfor ser vi simuleringer av Ornstein-Uhlenbeck prosessen med forskjellige verdier for θ i blått sammen med den browniske bevegelsen i rødt. Her har vi valgt $\sigma = 1$ og $\mu = 0$.

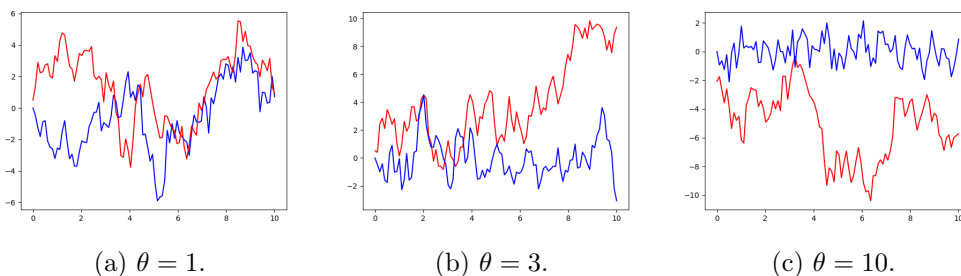


Figure 2: Ornstein-Uhlenbeck prosessen med $\sigma = 1$, $\mu = 0$ og ulike verdier for θ .

Vi ser at for $\theta = 1$ så klarer vi nesten ikke å avgjøre hvilken av prosessene som er den brownske bevegelsen. Dette gir mening, fordi jo lavere verdien for θ er, jo mer ligner prosessen en brownsk bevegelse. Hvis θ er lik 0, får vi prosessen $dX_t = \sigma dB_t$ som er en brownsk bevegelse når $\sigma = 1$. Ser vi på tilfellet hvor $\theta = 3$, ser vi at prosessen reverteres tilbake til 0. For $\theta = 10$ ser vi det enda tydeligere.

I denne oppgaven skal vi se på tilfellet hvor μ er en funksjon som avhenger av tid. Vi bruker da m_t som driftsledd

$$dX_t = \theta(m_t - X_t)dt + \sigma dB_t. \quad (1.1)$$

Det er denne prosessen vi vil bruke for å modellere strømprisene. Det er spesielt viktig for spotprisene at driftsleddet er en funksjon og ikke en konstant. Nedenfor ser vi plottet for gjennomsnittsprisen på spotpriser over ett år (x-aksen) i de nordiske landene for 13 år målt i øre per kWh (y-aksen).

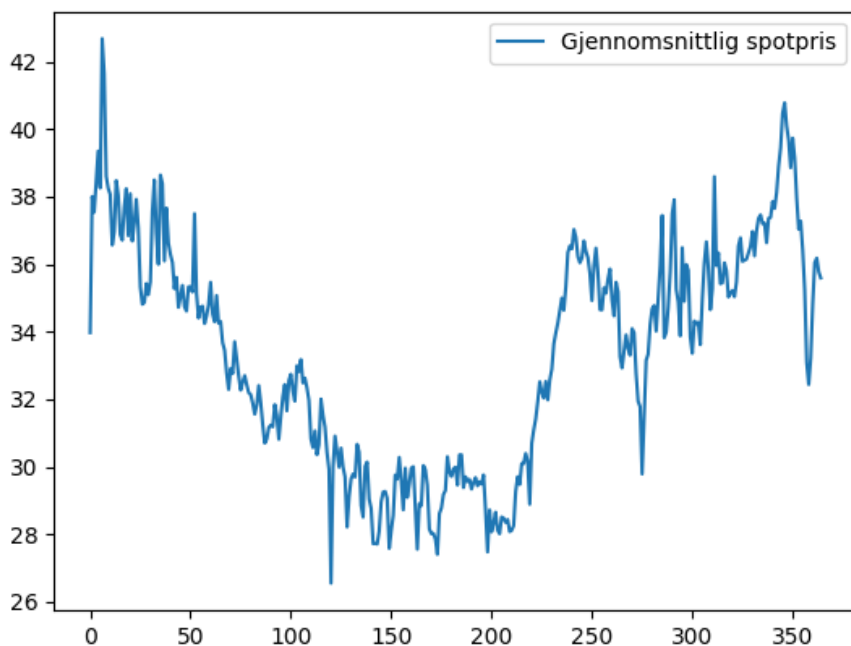


Figure 3: Gjennomsnittlig spotpris.

Vi ser at funksjonen oppfører seg svært irregulært, som er et resultat av at strømprisene svinger konstant. I tillegg ser vi at prisene ser ut til å være lavere på sommeren og høyere på vinteren, såkalt sesongvariasjon. Driftsleddet til prosessen må derfor være en funksjon slik at vi kan fange opp denne variasjonen.

Når det gjelder kontraktsprisene, så ser vi ingen sesongvariasjon i disse dataene. Langsiktige kontrakter på strøm er ikke like påvirket av årstid som spotprisene. Dette kan skyldes at langsiktige kontrakter er basert på forventet fremtidig etterspørsel og tilbud, og ikke bare på dagens markedssituasjon. Dette betyr at kontraktsprisene er mer stabile og mindre påvirket av kortsiktige svingninger i strømmarkedet.

I tillegg vil kontrakten begynne å løpe året etter du kjøper kontrakten. Det er altså ingen forskjell på kontraktsstart om du kjøper kontrakten sommeren 2012 eller vinteren 2012. Kontrakten starter likevel 1. januar 2013. Årstiden vil derfor ha lite innvirkning på kontraktsprisen da startdatoen for kontrakten forblir den samme.

Når vi lager modellen for spotpriser, tar vi utgangspunkt i sesongvariasjonen. Dette gjør vi ikke for kontraktsprisene, og driftsleddet for disse prisene vil være konstant. Felles for både spotpriser og kontraktspriser er at vi bruker en Ornstein-Uhlenbeck prosess til å modellere prisene. På denne måten vil de estimerte prisene følge en trend, men likevel inneholde støy og uregelmessigheter for å få med de forholdene som påvirker strømprisene som er umulig å forutse.

Nå som vi har en plan for hvordan vi ønsker at modellen vår skal oppføre oss, må vi ha verdier for parameterne i modellen, altså σ og θ . Disse beregner vi ved å bruke historisk data.

3.1 Volatiliteten: σ

Vi så tidligere at det finnes flere typer av Ornstein-Uhlenbeck prosessen, og vi må vurdere hvilken prosess vi ønsker å bruke etter hva slags struktur vi ønsker at modellen vår skal ha. Vi ønsker for eksempel å bruke Ornstein-Uhlenbeck prosessen med en funksjon som driftsledd (1.1) for å modellere spotprisene, mens Ornstein-Uhlenbeck prosessen hvor driftsleddet er konstant (2.1) passer bedre for kontraktsprisene. For å gjøre regningen videre mer generell slik at den kan passe for flere prosesser, bruker vi følgende stokastiske differensiallikning

$$dX_t = b(t, X_t; \theta)dt + \sigma dB_t, \quad (3.1)$$

eller på integralform

$$X_t = X_0 + \int_0^t b(s, X_s; \theta)ds + \sigma B_t. \quad (3.2)$$

I finans er volatiliteten graden av variasjon i prishistorikk og er en av parameterne vi ønsker å estimere for å lage modellen vår for strømpriser. Denne verdien betegner vi med σ og den vil være et mål på spredningen av data rundt gjennomsnittet over et gitt tidsintervall.

Estimering av σ innebærer å beregne variasjonen av dataene vi har over en gitt periode. Vi estimerer σ ved å lage et uttrykk for den kvadratiske variasjonen som inneholder σ . Før vi gjør dette, regner vi ut den kvadratiske variasjonen til en brownsk bevegelse, da dette er et resultat som vil være nyttig for oss videre.

Fra Lévy's karakteriseringsteorem for 1-dimensjonelle brownske bevegelser (2.2), vet vi at den kvadratiske variasjonen til den brownske bevegelsen er definert på følgende måte

$$\langle B \rangle_t = \lim_{\|\pi\| \rightarrow 0} \sum_{i=0}^n (B_{t_{i+1}} - B_{t_i})^2 = t. \quad (3.3)$$

At den kvadratiske variasjonen til den brownske bevegelsen er lik t er en viktig egenskap ved brownske bevegelser. Dette tyder på at prosessens variasjonen øker uten begrensning over tid. Vi finner nå et annet uttrykk for den kvadratiske variasjonen av X_t .

$$\begin{aligned}
\langle X \rangle_t &= \lim_{\|\pi\| \rightarrow 0} \sum_{i=0}^n (X_{t_{i+1}} - X_{t_i})^2 \\
&= \lim_{\|\pi\| \rightarrow 0} \sum_{i=0}^n \left(\left(\cancel{X_0} + \int_0^{t_{i+1}} b(s, X_s; \theta) ds + \sigma dB_{t_{i+1}} \right) \right. \\
&\quad \left. - \left(\cancel{X_0} + \int_0^{t_i} b(s, X_s; \theta) ds + \sigma B_{t_i} \right) \right)^2 \\
&= \lim_{\|\pi\| \rightarrow 0} \sum_{i=0}^n \left(\int_0^{t_{i+1}} b(s, X_s; \theta) ds + \sigma dB_{t_{i+1}} - \int_0^{t_i} b(s, X_s; \theta) - \sigma dB_{t_i} \right)^2 \\
&= \lim_{\|\pi\| \rightarrow 0} \sum_{i=0}^n \left(\sigma(B_{t_{i+1}} - B_{t_i}) + \int_{t_i}^{t_{i+1}} b(s, X_s; \theta) ds \right)^2 \\
&= \lim_{\|\pi\| \rightarrow 0} \sum_{i=0}^n \left(\sigma(B_{t_{i+1}} - B_{t_i}) + b(t_i, X_{t_i}; \theta)(t_{i+1} - t_i) \right)^2 \\
&= \lim_{\|\pi\| \rightarrow 0} \sum_{i=0}^n \left(\sigma^2(B_{t_{i+1}} - B_{t_i})^2 + 2\sigma(B_{t_{i+1}} - B_{t_i})b(t_i, X_{t_i}; \theta)(t_{i+1} - t_i) \right. \\
&\quad \left. + b^2(t_i, X_{t_i}; \theta)(t_{i+1} - t_i)^2 \right) \\
&= \sigma^2 \lim_{\|\pi\| \rightarrow 0} \sum_{i=0}^n (B_{t_{i+1}} - B_{t_i})^2 + \text{I} + \text{II} \\
&= \sigma^2 \langle B \rangle_t \\
&= \sigma^2 t
\end{aligned}$$

Vi viser at I og II er lik 0:

I:

$$\begin{aligned}
&\lim_{\|\pi\| \rightarrow 0} \sum_{i=0}^n 2\sigma(B_{t_{i+1}} - B_{t_i})b(t_i, X_{t_i}; \theta)(t_{i+1} - t_i) \\
&\leq \lim_{\|\pi\| \rightarrow 0} \sum_{i=0}^n \underbrace{\max_i(t_{i+1} - t_i)}_{\|\pi\| \rightarrow 0} 2\sigma(B_{t_{i+1}} - B_{t_i})b(t_i, X_{t_i}; \theta) \\
&= 0
\end{aligned}$$

II:

$$\begin{aligned} & \lim_{\|\pi\| \rightarrow 0} \sum_{i=0}^n b^2(t_i, X_{t_i}; \theta) (t_{i+1} - t_i)^2 \\ & \leq \lim_{\|\pi\| \rightarrow 0} \sum_{i=0}^n \underbrace{\max_i (t_{i+1} - t_i)^2}_{\|\pi\|^2 \rightarrow 0} b^2(t_i, X_{t_i}; \theta) \\ & = 0 \end{aligned}$$

Maksimal avstand mellom partisjonene går mot 0, altså $(t_{i+1} - t_i)$ vil bli mindre og mindre. Summen i I og II vil begge gå mot 0 når avstanden mellom partisjonene går mot 0. For alle lignende stokastiske prosesser vil følgende gjelde $\langle X \rangle_t = \sigma^2 t$. Vi har nå et uttrykk for $\hat{\sigma}$

$$\hat{\sigma} = \sqrt{\frac{1}{t} \lim_{n \rightarrow \infty} \sum_{i=0}^n (X_{t_{i+1}} - X_{t_i})^2}. \quad (3.4)$$

Dette er for det 1-dimensjonelle tilfellet hvor $\hat{\sigma}$ er en skalar. Vi fokuserer først på én dimensjon, men vi skal senere beregne $\hat{\sigma}$ for flere dimensjoner også.

3.1.1 Estimering av σ

Vi har et uttrykk for $\hat{\sigma}$ som enkelt kan implementeres i Python. Før vi beregner $\hat{\sigma}$ for dataene våre for strømpriser, beregner vi $\hat{\sigma}$ for en simulering av en stokastisk prosess. For å definere denne prosessen, må vi velge en verdi for σ . Dette lar oss kjenne den faktiske verdien for σ , som gjør at vi kan sammenligne denne verdien med den estimerte verdien. På denne måten kan vi vurdere om metoden vi bruker gir et godt estimat for σ .

Vi ønsker å estimere σ for en standard brownsk bevegelse uten driftsledd

$$dX_t = \sigma dB_t. \quad (3.5)$$

Denne prosessen simuleres ved å gange σ med en brownsk bevegelse. Vi har tidligere laget en simulasjon av en brownsk bevegelse, så alt vi trenger å gjøre er å importere dette skriptet og gange bevegelsen med σ for å få prosessen X_t .

```
1 import brownian_motion
2
3 def prosess(n, T, sigma):
4     BM = brownian_motion.simulate_BM(1, n, T)
5     x = sigma * BM[0]
6     return x
```

Dette skriptet lager en funksjon som skal simulere prosessen (3.5) fra 0 til T med n antall tidssteg ved å ta inn en vilkårlig verdi for σ som skal ganges med en brownsk bevegelse.

Funksjonen som skal beregne $\hat{\sigma}$ tar inn to argumenter, prosessen x og tiden T . Funksjonen beregner den kvadratiske variasjonen til prosessen, og returnerer $\hat{\sigma}$ som er definert som kvadratroten av den kvadratiske variasjonen delt på tiden.

```
1 def est_sigma(x, T):
2     #Beregner kvadratisk variasjon av prosessen
3     kvad = 0
4     for i in range(len(x) - 1):
5         kvad = kvad + (x[i+1] - x[i])**2
6     #Volatiliteten er gitt ved kvadratroten av den
7     #kvadratiske variasjonen delt på tiden
8     estimert_sigma = np.sqrt(kvad/T)
9     return estimert_sigma
```

Vi velger $\sigma = 2$ og lar X_t være en prosess fra 0 til 10 med 1000 tidssteg. Etter å ha kjørt skriptet 10 ganger får vi følgende verdier for $\hat{\sigma}$:

- 2.0087654130376085
- 1.9878823458855244
- 1.9881278328532834
- 2.0200955339036186
- 1.9914451084498679
- 2.003266019993608
- 2.0009404329552742
- 2.0031712433110296
- 1.9997789185982948
- 2.0040039276102632

Denne metoden gir oss en gjennomsnittlig feil på 0.007300837, som er relativt lavt. Dette tyder på at metoden gir gode estimater, og vi kan dermed konkludere med at denne måten å beregne $\hat{\sigma}$ på i én dimensjon er pålitelig.

3.2 Maximum likelihood estimation

Vi ser på denne generelle stokastiske differensiallikningen

$$dX_t = b(X_t; \theta)dt + \sigma dB_t \quad (3.1)$$

med parametere σ og θ . Vi har funnet en god måte å estimere σ på ved hjelp av kvadratisk variasjon, og nå gjenstår det å finne et godt estimat for θ også.

For å estimere θ bruker vi en metode kalt maximum likelihood estimation. Denne metoden går ut på å finne den verdien av $\hat{\theta}$ som er mest sannsynlig for prosessen. Dette gjøres ved å maksimere en funksjon, som vi kaller likelihoodfunksjonen. Før vi ser nærmere på metoden, må vi forstå hva likelihood betyr og hva som skiller dette fra begrepet sannsynlighet.

Begrepene sannsynlighet og likelihood blir ofte brukt om hverandre i sannsynlighetsteori og statistiske analyser. Det finnes ingen direkte norsk oversettelse av likelihood, men begrepet er likevel i bruk innenfor fagfeltet. Jeg vil derfor bruke begrepet "likelihood" videre i oppgaven.

Sannsynlighet og likelihood har flere likhetstrekk, men det finnes en vesentlig forskjell. Sannsynlighet refererer til sannsynligheten for at et utfall skal skje, mens likelihood refererer til prosessen med å bestemme den beste sannsynlighetsfordelingen gitt bestemte utfall. Sannsynlighet er altså en verdi på hvor sannsynlig det er for at et utfall vil forekomme, og kan beskrives ved en fordeling.

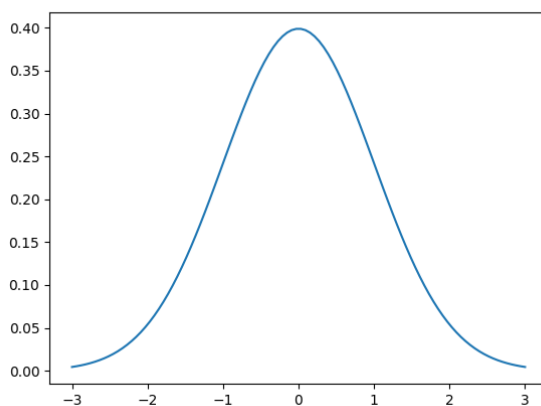


Figure 4: Normalfordeling.

Bildet viser en normalfordeling som beskriver hvordan ulike utfall x fordeler seg rundt gjennomsnittet. Observerte utfall deles altså inn i kjennetegn, og vi får da fordelingen som så gir oss sannsynligheten for de ulike hendelsene. Fordelingen er gitt ved observerte utfall, og sannsynligheten er gitt ved fordelingen.

Likelihood derimot refererer til prosessen med å bestemme den beste fordelingen gitt bestemte utfall. Når vi skal finne den fordelingen som passer best til bestemte utfall, forutsetter det at vi allerede har utfall tilgjengelig. Ved likelihood har utfallet dermed allerede skjedd, mens ved sannsynlighet bestemmer vi hvor sannsynlig det er for at et utfall skjer før det har skjedd. Vi kan oppsummere det slik med gjennomsnitt μ og standardavvik σ :

- Sannsynligheten for x gitt fordelingen: $P(x|\mu, \sigma)$,
- Likelihooden for fordelingen gitt x : $L(\mu, \sigma|x)$.

Ved sannsynlighet endrer vi x -verdiene når vi ønsker å finne sannsynligheten til ulike utfall, men parameterne μ og σ forblir det samme. Fordelingen er fiksert. Ved likelihood har utfallet vårt allerede skjedd og vi ønsker å finne ut hvor sannsynlig fordelingen er. Nå er det x -verdien som forblir det samme, men vi endrer på μ og σ for å finne likelihooden. Utfallene x er fiksert.

Vi starter med å se på likelihooden til en normalfordeling og vi ønsker å bruke maximum likelihood estimation til å estimere parameterne μ og σ gitt utfallet x . Vi ser her sannsynlighetstetthetsfunksjonen til normalfordelingen

$$f(x|\mu, \sigma) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}. \quad (3.6)$$

Denne funksjonen beskriver sannsynlighetsfordelingen til en normalfordeling gitt parameterne μ og σ . Funksjonen tar inn utfallet x som argument. Likelihoodfunksjonen vil også beskrive sannsynlighetsfordelingen, men nå er μ og σ variabler i funksjonen mens x er gitt på forhånd. Vi kan dermed definere likelihoodfunksjonen til å være lik sannsynlighetstetthetsfunksjonen, men med ulike argumenter

$$L(\mu, \sigma|x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \quad (3.7)$$

(Pavliotis, 2014, s. 159).

Hvis vi har flere uavhengige observasjoner $x = \{x_1, x_2, \dots, x_n\}$, vil likelihooden for parameterne μ og σ gitt disse utfallene være produktet av likelihoodfunksjonen for hvert utfall.

$$\begin{aligned}
& L(\mu, \sigma | x_1, \dots, x_n) \\
&= L(\mu, \sigma | x_1) \times \dots \times L(\mu, \sigma | x_n) \\
&= \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x_1-\mu)^2}{2\sigma^2}} \times \dots \times \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x_n-\mu)^2}{2\sigma^2}} \\
&= \prod_{i=1}^n \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x_i-\mu)^2}{2\sigma^2}}
\end{aligned} \tag{3.8}$$

Målet vårt er å finne optimale estimater for μ og σ , altså vi ønsker å finne maximum likelihood estimator for disse parameterne. Dette gjøres ved å maksimere likelihoodfunksjonen med hensyn på parameteren vi ønsker å estimere. Det vil si at vi deriverer likelihoodfunksjonen med hensyn på μ og på σ og setter disse uttrykkene lik 0.

Likelihoodfunksjonen er et produkt, som er vesentlig vanskeligere å derivere enn en sum. En metode for å gjøre derivasjonen enklere på er å ta logaritmen til likelihoodfunksjonen istedenfor. Dette kan vi gjøre uten å endre på resultatene siden logaritmen er en monoton økende funksjon. Det vil si at maksimering av logaritmen til en funksjon vil være ekvivalent med å maksimere selve funksjonen (Navidi, 2010, s. 282). Loglikelihoodfunksjonen og likelihoodfunksjonen vil altså ha samme maksimumsverdi.

$$\begin{aligned}
& \ln(L(\mu, \sigma | x_1, \dots, x_n)) \\
&= \ln(L(\mu, \sigma | x_1) \times \dots \times (L(\mu, \sigma | x_n))) \\
&= \ln(L(\mu, \sigma | x_1)) + \dots + \ln(L(\mu, \sigma | x_n)) \\
&= \ln\left(\frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x_1-\mu)^2}{2\sigma^2}}\right) + \dots + \ln\left(\frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x_n-\mu)^2}{2\sigma^2}}\right) \\
&= \frac{-1}{2} \ln(2\pi) - \ln(\sigma) - \frac{(x_1 - \mu)^2}{2\sigma^2} + \dots + \left(\frac{-1}{2}\right) \ln(2\pi) - \ln(\sigma) - \frac{(x_n - \mu)^2}{2\sigma^2} \\
&= \frac{-n}{2} \ln(2\pi) - n \ln(\sigma) - \frac{(x_1 - \mu)^2}{2\sigma^2} - \dots - \frac{(x_n - \mu)^2}{2\sigma^2}
\end{aligned}$$

Det er denne funksjonen vi ønsker å derivere med hensyn på μ og på σ .

1.

$$\begin{aligned} & \frac{\partial}{\partial \mu} (\ln(L(\mu, \sigma | x_1, \dots, x_n))) \\ &= \frac{\partial}{\partial \mu} \left(\frac{-n}{2} \ln(2\pi) - n \ln(\sigma) - \frac{(x_1 - \mu)^2}{2\sigma^2} - \dots - \frac{(x_n - \mu)^2}{2\sigma^2} \right) \\ &= 0 - 0 + \frac{x_1 - \mu}{\sigma^2} + \dots + \frac{x_n - \mu}{\sigma^2} \\ &= \frac{x_1 - \mu}{\sigma^2} + \dots + \frac{x_n - \mu}{\sigma^2} \\ &= \frac{1}{\sigma^2} ((x_1 + \dots + x_n) - n\mu) \end{aligned}$$

2.

$$\begin{aligned} & \frac{\partial}{\partial \sigma} (\ln(L(\mu, \sigma | x_1, \dots, x_n))) \\ &= \frac{\partial}{\partial \sigma} \left(\frac{-n}{2} \ln(2\pi) - n \ln(\sigma) - \frac{(x_1 - \mu)^2}{2\sigma^2} - \dots - \frac{(x_n - \mu)^2}{2\sigma^2} \right) \\ &= 0 - \frac{n}{\sigma} + \frac{(x_1 - \mu)^2}{\sigma^3} + \dots + \frac{(x_n - \mu)^2}{\sigma^3} \\ &= \frac{-n}{\sigma} + \frac{1}{\sigma^3} ((x_1 - \mu)^2 + \dots + (x_n - \mu)^2) \end{aligned}$$

Vi får at maximum likelihood estimator for μ og σ er følgende

$$\hat{\mu} = \frac{(x_1 + \dots + x_n)}{n}, \quad \hat{\sigma} = \sqrt{\frac{(x_1 - \mu)^2 + \dots + (x_n - \mu)^2}{n}}. \quad (3.9)$$

3.3 Snittreverteringshastigheten: θ

Vi har sett på en metode som fungerer bra for å estimere ukjente parametre. Vi ønsker nå å bruke samme metode til å estimere parameteren θ som representerer snittreverteringshastigheten til prosessen

$$dX_t = b(X_t; \theta)dt + \sigma dB_t, \quad (3.1)$$

og da trenger vi likelihoodfunksjonen til denne prosessen.

Girsanovs teorem er et resultat vi trenger når vi skal finne vår likelihoodfunksjon. Det finnes en sterk sammenheng mellom Girsanovs teorem, den Radon-Nikodym deriverte og likelihoodfunksjonen. Girsanovs teorem beskriver hvordan en stokastisk prosess endrer seg ved endring av mål. Et mål \mathbb{Q} har relasjon til et annet mål \mathbb{P} og gir et uttrykk for den Radon-Nikodym deriverte, som da blir vår likelihoodfunksjon (Pavliotis, 2014, s. 159-160).

Ved Girsanovs teorem del II (2.5) trenger vi en funksjon u som skal settes inn i M_t . Vi bruker $u = \frac{1}{\sigma}b(X_t; \theta)$, og dette gir oss følgende uttrykk

$$M_t = \exp \left(- \int_0^t \frac{1}{\sigma} b(X_s; \theta) dB_s - \frac{1}{2} \int_0^t \frac{1}{\sigma^2} b^2(X_s; \theta) ds \right).$$

Vi legger merke til at det første leddet integreres med hensyn på B_t . Det kan være lurt å lage et uttrykk for B_t som inneholder X_t slik at integralet avhenger av prosessen vår X_t istedenfor. Vi løser (3.1) med hensyn på dB_t og får følgende resultat

$$dB_t = \frac{1}{\sigma} dX_t - \frac{1}{\sigma} b(X_t; \theta) dt.$$

Vi husker at M_t er lik den Radon-Nikodym deriverte av målene \mathbb{P} og \mathbb{Q} , altså $M_t = \frac{d\mathbb{Q}}{d\mathbb{P}}$, som vil si at

$$d\mathbb{P} = M_t^{-1} d\mathbb{Q} \quad d\mathbb{Q} = M_t d\mathbb{P}. \quad (3.10)$$

Vi ønsker å finne funksjonen som maksimerer prosessen X_t , med andre ord så ønsker vi å finne $\max_{\theta} P(X_t \in A) = \max_{\theta} \int_A X_t d\mathbb{P}$. Den deriverte av målet \mathbb{P} kan uttrykkes som den inverse av M_t ganger den deriverte av målet \mathbb{Q} . Vi trenger et uttrykk for M_t^{-1} hvor integralene er integrert med hensyn på X_t .

$$\begin{aligned}
M_t^{-1} &= \exp \left(\int_0^t \frac{1}{\sigma} b(X_s; \theta) dB_s + \frac{1}{2} \int_0^t \frac{1}{\sigma^2} b^2(X_s; \theta) ds \right) \\
&= \exp \left(\int_0^t \frac{1}{\sigma} b(X_s; \theta) \left(\frac{1}{\sigma} dX_s - \frac{1}{\sigma} b(X_s; \theta) ds \right) + \frac{1}{2} \int_0^t \frac{1}{\sigma^2} b^2(X_s; \theta) ds \right) \\
&= \exp \left(\int_0^t \frac{1}{\sigma^2} b(X_s; \theta) dX_s - \frac{1}{\sigma^2} b^2(X_s; \theta) ds + \frac{1}{2} \int_0^t \frac{1}{\sigma^2} b^2(X_s; \theta) ds \right) \\
&= \exp \left(\int_0^t \frac{1}{\sigma^2} b(X_s; \theta) dX_s - \int_0^t \frac{1}{\sigma^2} b^2(X_s; \theta) ds + \frac{1}{2} \int_0^t \frac{1}{\sigma^2} b^2(X_s; \theta) ds \right) \\
&= \exp \left(\int_0^t \frac{1}{\sigma^2} b(X_s; \theta) dX_s - \frac{1}{2} \int_0^t \frac{1}{\sigma^2} b^2(X_s; \theta) ds \right)
\end{aligned}$$

Vi har nå et uttrykk for M_t^{-1} som vi kan bruke til å finne likelihood-funksjonen til X_t . X_t ligger i følgende sannsynlighetsrom $(\Omega, \{\mathcal{F}_t\}_{t \geq 0}, \mathbb{P})$. Vi skal bevege oss inn i dette sannsynlighetsrommet $(\Omega, \{\mathcal{F}_t\}_{t \geq 0}, \mathbb{Q})$. Girsanovs teorem forteller oss at en stokastisk prosess kan beskrives som en brownisk bevegelse hvis man bytter målet. Det betyr at under målet \mathbb{Q} , så kan X_t behandles som en brownisk bevegelse. Vi bruker dette til å finne den sannsynlighetsfunksjonen som maksimerer X_t .

$(\Omega, \mathcal{F}, \mathbb{P})$

$$\begin{aligned}
&\max_{\theta} P(X_t \in A) \\
&= \max_{\theta} \int_A X_t d\mathbb{P} \\
&= \max_{\theta} \int_A X_t M_t^{-1} d\mathbb{Q} \\
&= \max_{\theta} \int_A X_t \exp \left(\int_0^t \frac{1}{\sigma^2} b(X_s; \theta) dX_s - \frac{1}{2} \int_0^t \frac{1}{\sigma^2} b^2(X_s; \theta) ds \right) d\mathbb{Q}
\end{aligned}$$

I målet \mathbb{Q} kan vi behandle X_t som en brownisk bevegelse. Vi kan ikke maksimere browniske bevegelser, som vil si at vi kun maksimerer dette uttrykket

$$\exp \left(\int_0^t \frac{1}{\sigma^2} b(X_s; \theta) dX_s - \frac{1}{2} \int_0^t \frac{1}{\sigma^2} b^2(X_s; \theta) ds \right).$$

Funksjonen som maksimerer X_t er dermed lik

$$L(\theta) = \exp \left(\int_0^t \frac{1}{\sigma^2} b(X_s; \theta) dX_s - \frac{1}{2} \int_0^t \frac{1}{\sigma^2} b^2(X_s; \theta) ds \right). \quad (3.11)$$

Vi tar logaritmen til dette og får at loglikelihoodfunksjonen er lik

$$\ell(\theta) = \int_0^t \frac{1}{\sigma^2} b(X_s; \theta) dX_s - \frac{1}{2} \int_0^t \frac{1}{\sigma^2} b^2(X_s; \theta) ds. \quad (3.12)$$

Dette vil være loglikelihoodfunksjonen for alle prosesser av formen (3.1), så vil den estimerte av θ variere ved valg av b . Dette uttrykket deriveres med hensyn på θ og vi setter den deriverte lik 0 for å finne et uttrykk for $\hat{\theta}$.

Vi har tidligere nevnt at vi ønsker å bruke Ornstein-Uhlenbeck prosessen hvor driftsleddet er en funksjon (1.1) til å lage en modell for spotpriser. Vi velger derfor $b(X_t; \theta) = \theta(m_t - X_t)$.

$$\begin{aligned} \ell(\theta) &= \int_0^t \frac{1}{\sigma^2} \theta(m_s - X_s) dX_s - \frac{1}{2} \int_0^t \frac{1}{\sigma^2} \theta^2 (m_s - X_s)^2 ds \\ \frac{\partial \ell(\theta)}{\partial \theta} &= \frac{\partial}{\partial \theta} \left(\int_0^t \frac{1}{\sigma^2} \theta(m_s - X_s) dX_s - \frac{1}{2} \int_0^t \frac{1}{\sigma^2} \theta^2 (m_s - X_s)^2 ds \right) \\ &= \frac{\partial}{\partial \theta} \left(\theta \int_0^t \frac{1}{\sigma^2} (m_s - X_s) dX_s - \frac{1}{2} \theta^2 \int_0^t \frac{1}{\sigma^2} (m_s - X_s)^2 ds \right) \\ &= \int_0^t \frac{1}{\sigma^2} (m_s - X_s) dX_s - \theta \int_0^t \frac{1}{\sigma^2} (m_s - X_s)^2 ds \end{aligned}$$

Vi setter dette uttrykket lik 0 for å finne en verdi for $\hat{\theta}$

$$\int_0^t \frac{1}{\sigma^2} (m_s - X_s) dX_s - \theta \int_0^t \frac{1}{\sigma^2} (m_s - X_s)^2 ds = 0.$$

Dette medfører at

$$\begin{aligned} \hat{\theta} &= \frac{\int_0^t \frac{1}{\sigma^2} (m_s - X_s) dX_s}{\int_0^t \frac{1}{\sigma^2} (m_s - X_s)^2 ds} \\ &= \frac{\int_0^t (m_s - X_s) dX_s}{\int_0^t (m_s - X_s)^2 ds} \\ &\approx \frac{\sum_{i=0}^n (m_t - X_t)(X_{t_{i+1}} - X_{t_i})}{\sum_{i=0}^n (m_t - X_t)^2 (t_{i+1} - t_i)} \end{aligned} \quad (3.13)$$

Vi har også Ornstein-Uhlenbeck prosessen hvor driftsleddet er en konstant (2.1). Denne prosessen er mer passende for 1-årskontrakter, 2-årskontrakter og 3-årskontrakter hvor vi ikke har sesongvariasjon. Utregningen for å finne et estimat for θ med $b = \theta(\mu - X_t)$ er helt lik som for prosessen (1.1), og vi får følgende resultat

$$\begin{aligned}\hat{\theta} &= \frac{\int_0^t (\mu - X_s) dX_s}{\int_0^t (\mu - X_s)^2 ds} \\ &\approx \frac{\sum_{i=0}^n (\mu - X_{t_i})(X_{t_{i+1}} - X_{t_i})}{\sum_{i=0}^n (\mu - X_{t_i})^2 (t_{i+1} - t_i)}.\end{aligned}\tag{3.14}$$

Vi legger merke til at i uttrykkene for $\hat{\theta}$ så kan vi kansellere σ . Dette indikerer at σ ikke påvirker den optimale verdien av $\hat{\theta}$ som maksimerer log-likelihoodfunksjonen. Årsaken til dette kan være at volatiliteten påvirker størrelsen på variasjonene i den stokastiske prosessen, mens snittreverterings hastigheten påvirker tendensen til prosessen. De tilfeldige svingningene som er beskrevet av σ påvirker ikke den langsiktige tendensen til prosessen slik som θ gjør.

Vi kommer til å se videre i oppgaven at dette kun gjelder for én dimensjon. I en flerdimensjonell prosess vil variasjonene i prosessen være avhengige av hverandre og kan ikke enkeltvis skilles ut som i det 1-dimensjonelle tilfellet. I tillegg vil den optimale verdien av $\hat{\theta}$ i det flerdimensjonelle tilfellet avhenge av både gjennomsnittet og variasjonen i prosessen. Vi kommer til å se senere at vi ikke kan kansellere σ i det flerdimensjonelle tilfellet.

3.4 Simulering av en Ornstein-Uhlenbeck prosess og estimering av θ

Vi skal simulere de ulike Ornstein-Uhlenbeck prosessene og estimere θ for hver av prosessene. Vi har sett at $\hat{\theta}$ ikke avhenger av σ , så vi setter $\sigma = 1$ for enkelthetens skyld.

Vi ønsker dermed å simulere prosessene

$$dX_t = \theta(m_t - X_t)dt + dB_t, \quad (3.15)$$

og

$$dX_t = \theta(\mu - X_t)dt + dB_t. \quad (3.16)$$

Vi skal bruke følgende cosinusfunksjon som m_t for prosessen (3.15)

$$m(t) = A + B \cos Ct = 32.5 + 5.5 \cos \frac{2\pi}{365}x \quad (3.17)$$

hvor A, B og C representerer parameterne

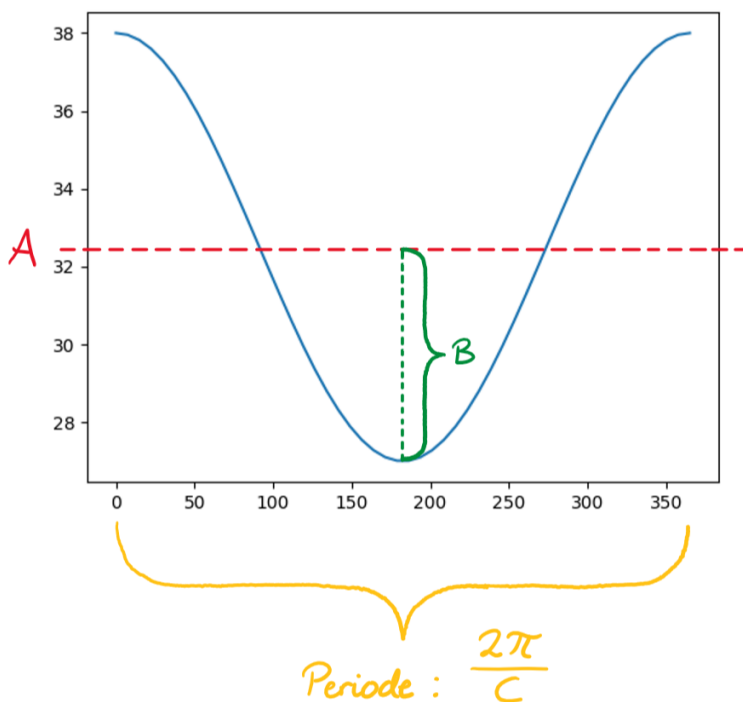


Figure 5: Visuell fremstilling av parameterne til funksjonen $m(t) = A + B \cos Ct$.

Disse parameterne skal tilpasses slik at funksjonen skal ligge tett på de faktiske dataene for gjennomsnittlig spotpris. Nedenfor ser vi cosinusfunksjonen plottet sammen med gjennomsnittlig spotpris.

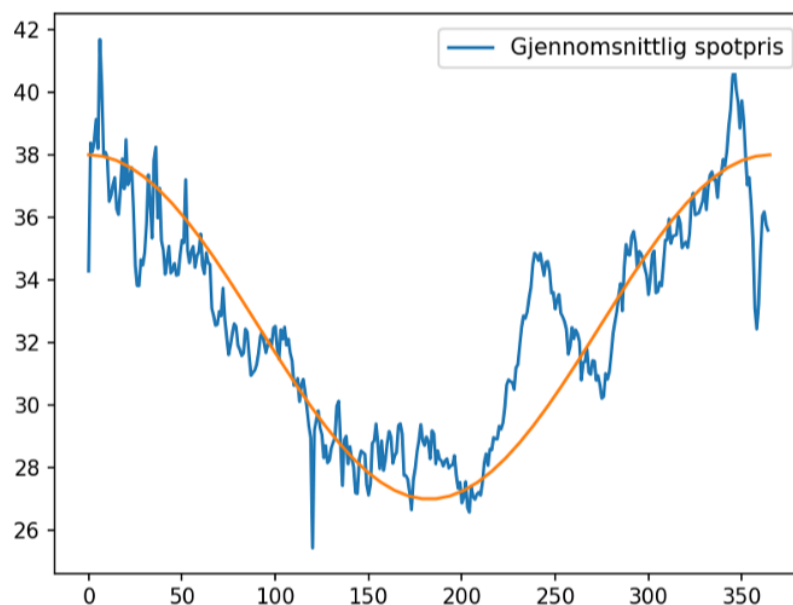


Figure 6: Gjennomsnittlig spotpris plottet sammen med cosinusfunksjonen.

Vi simulerer funksjonen m_t på følgende måte i Python

```
1 def m(t):  
2     return 32.5 + 5.5 * np.cos((2*math.pi/365) * t)
```

Når vi skal beregne verdien for $\hat{\theta}$, lager vi en funksjon som tar fire argumenter: tiden T , antall tidssteg n , driftsledd m og en prosess x . Vi bruker uttrykket for $\hat{\theta}$ som vi fant ved maximum likelihood estimation og skriptet vil til slutt returnere en verdi for $\hat{\theta}$.

```

1 def est_theta(n, T, m, x):
2     #Beregner snittreverteringshastigheten til prosessen
3     t = np.linspace(0, T, n)
4     teller = 0
5     nevner = 0
6
7     for i in range(len(x) - 1):
8         teller = teller + (m[i] - x[i]) * (x[i+1] - x[i])
9         nevner = nevner + (m[i] - x[i]) ** 2 \
10             * (t[i+1] - t[i])
11
12     estimated_theta = teller/nevner
13     return estimated_theta

```

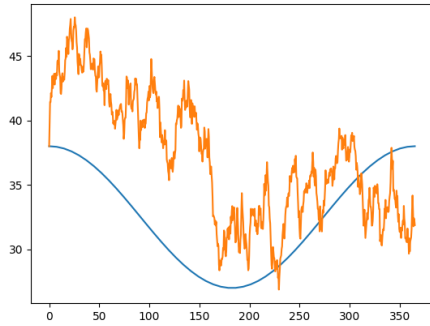
Vi bruker prosessen (3.15) som x .

```

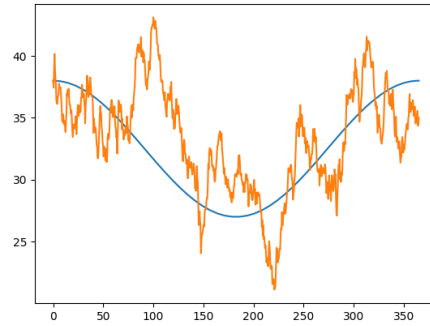
1 import numpy as np
2 import white_noise
3
4 def prosess(n, T, theta, m):
5     t = np.linspace(0, T, n)
6     dt = t[1] - t[0]
7
8     WN = white_noise.simulate_WN(1, n, T)
9     x = np.zeros(n)
10    x[0] = m(t[0])
11
12    for i in range(len(x) - 1):
13        x[i + 1] = x[i] + (theta * (m(t[i]) - x[i])) * dt +
14        WN[0, i]
15    return x

```

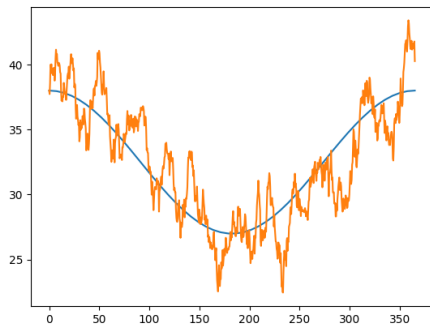
Her er m er cosinusfunksjonen som vi allerede har laget en simulasjon av. På neste side ser vi fire plot av denne prosessen med $n = 800$, $T = 365$ og ulike verdier for θ .



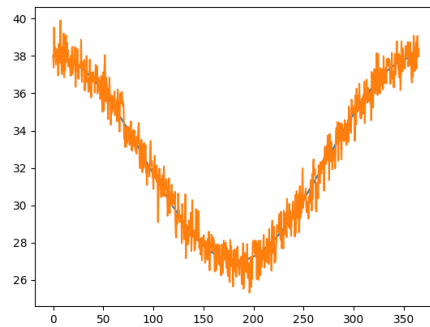
(a) $\theta = 0.006$



(b) $\theta = 0.04$



(c) $\theta = 0.1$



(d) $\theta = 2$

Figure 7: Simuleringer av Ornstein-Uhlenbeck prosessen med cosinusfunksjonen som driftsledd og ulike verdier for θ .

Basert på simuleringene ovenfor kan vi se at ulike verdier av θ kan føre til forskjeller i hvordan prosessene utvikler seg over tid. En høyere verdi av θ gjør at prosessen reverteres tilbake til gjennomsnittet med høyere hastighet. Vi kan se at bilde (d) har høyest verdi for θ , og denne prosessen holder seg mest stabilt rundt gjennomsnittet.

De andre plottene med lavere verdi for θ beveger seg langsommere tilbake mot gjennomsnittet og er mer irregulære. Dette stemmer bedre overens med hvordan spotprisene oppfører seg. Prosessen i bilde (d) er som sagt den mest stabile rundt gjennomsnittet, som kan være tegn på at modellen ikke tar høyde for all den variasjonen og usikkerheten som finnes i spotprisene. Vi kan derfor tenke oss at når vi beregner $\hat{\theta}$ for spotprisene, så vil vi få en verdi som ikke er like høy som 2. Realistiske verdier for $\hat{\theta}$ ser ut til å være under 0.1.

Etter å ha kjørt skriptet 10 ganger og funnet gjennomsnittlig verdi for $\hat{\theta}$, tar vi differansen mellom den faktiske verdien og den estimerte verdien for θ . Nedenfor ser vi verdien vi satt for θ sammen med differansen mellom θ og $\hat{\theta}$.

- $\theta = 0.006 : 0.005817298494726781$
- $\theta = 0.04 : 0.009008861117532511$
- $\theta = 0.1 : 0.003337047138557425$
- $\theta = 2 : 0.007348041082420487$

Vi ser at alle disse tallene har en desimalplassering som starter etter andre siffer. Differansen er lav, som betyr at det er gode estimater for θ . Vi kan dermed konkludere med at maximum likelihood estimation er en god metode for å estimere θ på.

Når vi ønsker å estimere θ for prosessen (3.16), gjør vi det på akkurat samme vis, men vi må bruke likelihoodfunksjonen for denne prosessen istedenfor. Skriptet for å finne en verdi for $\hat{\theta}$ blir som følgende

```

1 def est_theta(n, T, m, x):
2     #Beregner snittreverteringshastigheten til prosessen
3     t = np.linspace(0, T, n)
4     teller = 0
5     nevner = 0
6
7     for i in range(len(x) - 1):
8         teller = teller + (mu - x[i]) * (x[i+1] - x[i])
9         nevner = nevner + (mu - x[i]) ** 2 \
10             * (t[i+1] - t[i])
11
12     estimated_theta = teller/nevner
13     return estimated_theta

```

og vi bruker denne simulasjonen for prosessen x

```

1 import numpy as np
2 import white_noise
3
4 def proses(n, T, theta, mu):
5     t = np.linspace(0, T, n)
6     dt = t[1] - t[0]
7
8     WN = white_noise.simulate_WN(1, n, T)
9     x = np.zeros(n)
10    x[0] = m(t[0])
11
12    for i in range(len(x) - 1):
13        x[i + 1] = x[i] + (theta * (mu - x[i])) * dt \
14            + WN[0, i]
15    return x

```

Nedenfor ser vi plottet med $\mu = 5$, $\theta = 2$, $n = 500$ og $T = 100$.

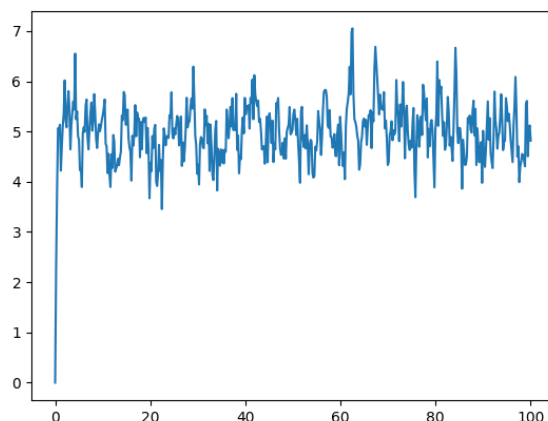


Figure 8: Simulering av Ornstein-Uhlenbeck prosessen med 5 som driftsledd og $\theta = 2$.

Vi ser at prosessen starter ved 0 og reverteres til 5 etterhvert som tiden går. Etter å ha kjørt skriptet 10 ganger og regnet ut gjennomsnittlig verdi for $\hat{\theta}$, får vi at differansen mellom estimert verdi og faktisk verdi er lik 0.0026158721029780896. Metoden fungerer bra for ulike prosesser også og man må bruke den likelihoodfunksjonen som passer for akkurat den prosessen man skal analysere.

4 Modellering av strømprisene

Vi har all teorien vi trenger og vi har testet metodene for å estimere ulike parametere i stokastiske differensiallikninger på. Vi beskriver strømprisene som stokastiske prosesser og vi trenger historisk data til å beregne to verdier: snittreverteringshastigheten θ , som representerer hastigheten prosessen reverterer tilbake til driftsleddet, og volatiliteten σ , som måler svingningene i prosessen for et tidsintervall.

Dataene som skal brukes i analysen er daglige verdier for spotpriser, 1-årskontrakter, 2-årskontrakter og 3-årskontrakter fra 2007 til 2019 i området NO2. Dataene for spotpriser er hentet fra NordPool, mens dataene for kontraktsprisene er fra NASDAQ. Dataene for kontraktspriser har blitt behandlet av økonomistudentene Robin Linaker Aalmen og Stian Tollefsen fra Handelshøyskolen ved UiA, så gjennom analysen forholder vi oss kun til ferdigbehandlet data.

Nedenfor ser vi dataene våre for spotpriser,

	Dato	Spot
0	2007-01-01	29.9854
1	2007-01-02	32.4075
2	2007-01-03	32.3279
3	2007-01-04	30.6196
4	2007-01-05	30.0246
...
4742	2019-12-27	37.9067
4743	2019-12-28	34.8221
4744	2019-12-29	33.4775
4745	2019-12-30	33.7767
4746	2019-12-31	32.8758

[4747 rows x 2 columns]

og for kontraktene

	Dato	1-årskontrakt	2-årskontrakt	3-årskontrakt
0	2007-01-02	40.25	42.00	42.40
1	2007-01-03	39.90	41.50	42.20
2	2007-01-04	39.90	41.25	42.00
3	2007-01-05	39.30	40.55	41.38
4	2007-01-08	37.90	38.90	40.00
...
3253	2019-12-19	36.35	34.01	33.06
3254	2019-12-20	36.25	34.15	33.21
3255	2019-12-23	35.85	34.05	33.17
3256	2019-12-27	35.06	33.90	33.13
3257	2019-12-30	34.56	33.15	32.95

[3258 rows x 4 columns]

Disse dataene er fra år 2007 til og med 2019. Vi har data fram til 2022, men vi ønsker ikke å ta med disse. I følge Thomas Aanensen (2021) fra Statistisk Sentralbyrå så var den gjennomsnittlige strømprisen for husholdningene i Norge 20.7 øre/kWh i 2020, som er det laveste som er registret siden 2002. Vinteren 2019/2020 var det rekordmye snø i fjellene og hydrologisk balanse, altså balansen mellom vanninngang og utgang fra et område eller et system. Denne balansen er en viktig faktor som påvirker vannressursene og var med på å bidra til rekordstor kraftproduksjon i 2020, som igjen førte til lave strømpriser.

Uregelmessige strømpriser endte ikke med dette. SSB forteller også at i 2021 så økte strømprisen kraftig. Strømprisen dette året var den høyeste som var målt i statistikken noen gang, og i 2022 var den enda høyere. Strømprisen for husholdninger i Norge var på 235.3 øre/kWh i 2022 (Holstad, 2023). Dette skyldes at i 2022 ble Ukraina invadert av Russland og det har hatt store konsekvenser for land i Europa. Mange europeiske land har pleid å kjøpe blant annet olje og gass fra Russland, men har valgt å avslutte handelen på grunn av Russlands angrep på Ukraina. Samtidig har Russland også begynt å sende mindre gass til Europa. Det gjør at prisen på gass stiger på grunn av etterspørsel og mangel, noe som er et av hovedfaktorene til økte strømpriser i 2022 (Fridtjof Nansens Institutt, 2022).

Alt dette fører til en svært uforutsigbar situasjon på markedet for strømpriser mellom 2020 og 2022. På grunn av dette velger vi å ikke inkludere data fra 2020 og senere i analysen. Dette vil bidra til å redusere den totale usikkerheten i vår analyse og gi et mer pålitelig grunnlag for å estimere parameterne i vår modell for strømpriser.

Dataene for strømpriser har en naturlig variasjon som skyldes flere faktorer. Det kan være utfordrende å arbeide med disse dataene, men et triks man pleier å bruke for uregjerlige data er å se på logaritmisk skala (Navidi, 2010, s. 259). Dette er en vanlig transformasjon i finansanalyse for å gjøre dataene mer normalfordelte. I noen sammenhenger er det rett og slett mer fornuftig å lage modeller for den logaritmiske prisen. Vi tar derfor logaritmen til dataene før vi begynner å estimere parameterne. Nedenfor ser vi de logaritmiske dataene for spotpriser og kontrakter.

	Dato	Spot			
0	2007-01-01	3.400710			
1	2007-01-02	3.478390			
2	2007-01-03	3.475931			
3	2007-01-04	3.421640			
4	2007-01-05	3.402017			
...			
4742	2019-12-27	3.635128			
4743	2019-12-28	3.550252			
4744	2019-12-29	3.510874			
4745	2019-12-30	3.519771			
4746	2019-12-31	3.492737			
[4747 rows x 2 columns]					
	Dato	1-årskontrakt	2-årskontrakt	3-årskontrakt	
0	2007-01-02	3.695110	3.737670	3.747148	
1	2007-01-03	3.686376	3.725693	3.742420	
2	2007-01-04	3.686376	3.719651	3.737670	
3	2007-01-05	3.671225	3.702536	3.722798	
4	2007-01-08	3.634951	3.660994	3.688879	
...	
3253	2019-12-19	3.593194	3.526655	3.498324	
3254	2019-12-20	3.590439	3.530763	3.502851	
3255	2019-12-23	3.579344	3.527830	3.501646	
3256	2019-12-27	3.557061	3.523415	3.500439	
3257	2019-12-30	3.542697	3.501043	3.494991	
[3258 rows x 4 columns]					

Videre vil det kun være disse dataene vi forholder oss til. Vi legger merke til at for kontraktene har vi bare data for handledagene. Dette betyr at vi bare har data for de dagene der strøm blir handlet på børsen. Dette inkluderer ikke helger og helligdager, da børsen er stengt og det ikke blir handlet strøm. Derfor vil det være hull i dataserien for kontraktene på disse dagene. Spotpriser derimot er tilgjengelige for alle dager i året, også i helger og på helligdager. Derfor har vi tilgang på et større antall datapunkter for spotprisene sammenlignet med kontraktprisene, som bare har datapunkter for hver handledag.

4.1 Modell i én dimensjon

Vi har sett på denne generelle stokastiske differensiallikningen

$$dX_t = b(X_t; \theta)dt + \sigma dB_t \quad (3.1)$$

med ulike funksjoner b utifra hvordan vi ønsker at modellen vår skal oppføre seg. Basert på historisk data for strømpriser, skal vi estimere parameterne σ og θ for denne prosessen. Dette vil gi oss en bedre forståelse av endringene og variasjonene i strømpriser over tid.

Vi lager fire separate 1-dimensjonelle modeller hvor hver modell skal representere spotpriser, 1-årskontrakter, 2-årskontrakter og 3-årskontrakter. Imidlertid er det viktig å merke seg at disse fire variablene er avhengige av hverandre, så å lage fire uavhengige modeller vil ikke fange opp det fulle bildet av deres gjensidige påvirkning og kan dermed gi et misvisende resultat.

Det kan likevel være nyttig å lage separate 1-dimensjonelle modeller for å utforske deres egenskaper og se hvordan dataene fungerer hver for seg. Det kan også være lurt å starte med et 1-dimensjonelt tilfelle før man går over i flere dimensjoner av flere grunner. Ved å begynne med én dimensjon, kan vi redusere kompleksiteten i problemet og dermed lettere evaluere ytelsen til forskjellige modeller og identifisere potensielle utfordringer.

4.1.1 Estimere σ i én dimensjon

Den enkleste parameteren å estimere er σ da vi kun trenger historisk data for å gjøre dette. Vi fant tidligere et uttrykk for $\hat{\sigma}$ ved hjelp av kvadratisk variasjon, og vi har allerede laget et skript som returnerer verdien av $\hat{\sigma}$ som tar inn argumentene x , som er prosessen, og T , som er tiden.

Vi bruker nå historisk data som x og antall kolonner, som da vil tilsvare antall dager, som T . Dataene våre gir oss følgende verdier for $\hat{\sigma}$:

- Spotpriser: $\hat{\sigma} = 0.11861578048443389$
- 1-årskontrakter: $\hat{\sigma} = 0.016220567259055685$
- 2-årskontrakter: $\hat{\sigma} = 0.012701539400773284$
- 3-årskontrakter: $\hat{\sigma} = 0.010713273490908214$

Basert på disse verdiene av $\hat{\sigma}$, kan vi si at spotprisene er mer volatile enn prisene på fremtidige kontrakter. Volatiliteten uttrykker gjerne hvor mye dataene svinger eller varierer over tid, som da betyr at spotprisene svinger mer enn kontraktene gjør.

Dette kan komme av at spotprisene påvirkes av mange kortsiktige faktorer, som tilbud og etterspørsel og svingninger i markedsstemningen, mens kontraktprisene reflekterer forventningene til markedet på et mer langsiktig tidsperspektiv. Kontraktene holder seg derfor mer stabile. Vi kan også se at volatiliteten reduseres jo lenger frem i tid kontraktene løper. 1-årskontraktene har litt høyere volatilitet enn 2-årskontraktene, som igjen har litt høyere volatilitet enn 3-årskontraktene. Dette skyldes sannsynligvis at usikkerheten og risikoen i markedet reduseres jo lengre frem i tid man ser.

En annen faktor som kan bidra til at spotprisene har høyere volatilitet enn kontraktene er sesongvariasjonen. Spotprisene kan påvirkes av endringer i tilbud og etterspørsel som kan oppstå på grunn av sesongmessige faktorer. Kontraktprisene derimot vil ikke påvirkes av slike umiddelbare endringer i like stor grad da disse prisene allerede vil ta hensyn til forventningene om endringer som vil skje i fremtiden.

Det er viktig å merke seg at disse konklusjonene er basert på volatiliteten alene, og det kan være andre faktorer som påvirker prisene og som ikke reflekteres i disse tallene. Vi kan si mer om dataene når vi har beregnet snittreverteringshastigheten også.

4.1.2 Funksjonen m_t : Gjennomsnittet

Vi ønsker at modellen vår for spotpriser skal fungere som en Ornstein-Uhlenbeck prosess hvor driftsleddet er en funksjon for å få med sesongvariasjonen til spotpriser. Før vi ser nærmere på θ , må vi bestemme hvilken funksjon eller verdi prosessen skal reverteres til. m_t er funksjonen som prosessen

$$dX_t = \theta(m_t - X_t)dt + \sigma dB_t \quad (1.1)$$

skal reverteres til. Denne funksjonen representerer gjennomsnittlig daglig spotpris for ett år.

	Dag	Måned	Spot
0	1	1	3.537510
1	2	1	3.612353
2	3	1	3.584388
3	4	1	3.616396
4	5	1	3.631566
...
360	27	12	3.541774
361	28	12	3.584527
362	29	12	3.558917
363	30	12	3.562377
364	31	12	3.543819

[365 rows x 3 columns]

Verdiene i kolonnen "Spot" er verdiene for funksjonen m_t hvor $t \in \{0, 364\}$. Vi får følgende plot for m_t for spotpriser:

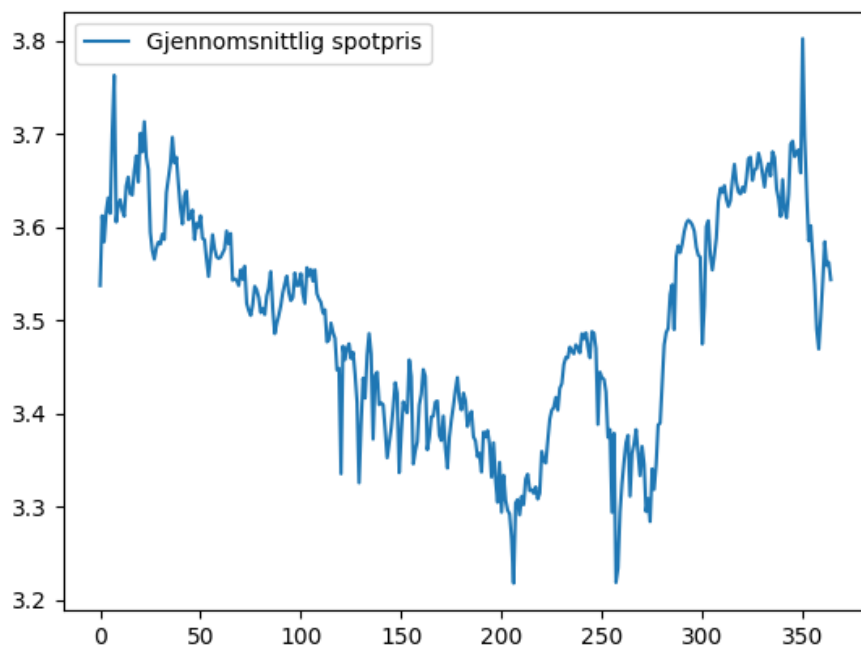


Figure 9: Gjennomsnittlig logaritmisk spotpris.

Plottet viser gjennomsnittlig spotpris for hver dag i året og gir oss en god oversikt over sesongvariasjonen i prisene. Vi ser også en del svingninger langs kurven, noe som er forventet da vi vet at spotprisene varierer veldig og at sesongvariasjonen kun er en liten indikator på hvordan prisene fungerer.

Når det kommer til kontraktsprisene, så kunne vi utifra verdiene vi fikk for $\hat{\sigma}$ konkludere med at kontraktsprisene er mer stabile og svinger mindre enn spotprisene. Vi vurderer derfor å bruke en konstant μ som driftsledd når vi skal modellere kontraktsprisene. Vi ønsker likevel å analysere svingningene i kontraktsprisene for å bekrefte at dette er et godt valg.

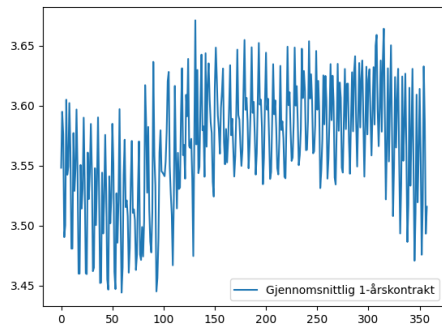
Vi definerer derfor funksjoner m_t som skal representere gjennomsnittlig daglig kontraktspris for ett år for hver av kontraktene. Ved å plote disse funksjonene for hver kontrakt, kan vi observere hvor mye prisene faktisk varierer. Hvis det er lite variasjon i kontraktsprisene, vil det styrke antagelsen vår om å velge en konstant μ .

Nedenfor ser vi en tabell med gjennomsnittlig kontraktspris for hver dag i året. Hver kolonne med kontraktspris representerer funksjonen m_t for den typen kontrakt. Her har vi $t \in \{0, 357\}$.

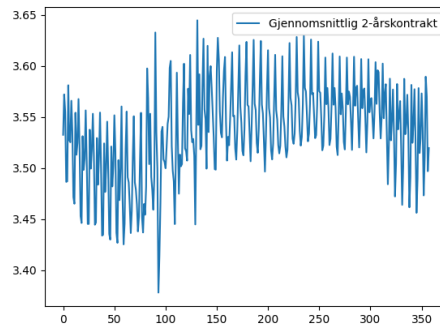
	Dag	Måned	1-årskontrakt	2-årskontrakt	3-årskontrakt
0	2	1	3.548156	3.532450	3.538931
1	3	1	3.594832	3.572020	3.570215
2	4	1	3.576129	3.554793	3.555520
3	5	1	3.490316	3.486364	3.500582
4	6	1	3.500165	3.486935	3.493985
...
353	23	12	3.517866	3.510384	3.497025
354	27	12	3.632620	3.589364	3.560278
355	28	12	3.586238	3.569912	3.543832
356	29	12	3.493198	3.496811	3.481506
357	30	12	3.515697	3.519481	3.502067

[358 rows x 5 columns]

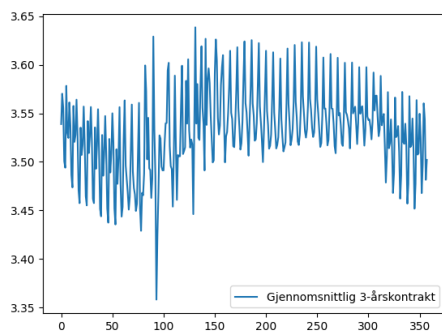
Vi plotter hver av kolonnene i tabellen som vil representere gjennomsnittlig kontraktspris for hver dag i året for hver av kontraktene. Det siste bildet viser alle funksjonene m_t i samme plot.



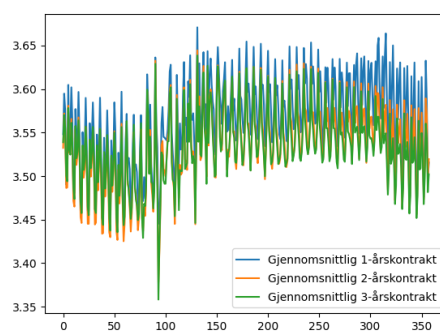
(a) Gjennomsnittlig 1-årskontrakt



(b) Gjennomsnittlig 2-årskontrakt



(c) Gjennomsnittlig 3-årskontrakt



(d) Gjennomsnittlig kontraktspris for ett år, to år og tre år.

Figure 10: Gjennomsnittlige kontraktspriser.

Vi ser ingen tydelig sesongvariasjon med lavere priser på sommeren enn på vinteren. Det kan se ut som prisene varierer en del, men hvis vi ser nøyere på y-aksen, så ser vi at funksjonene holder seg mellom 3.45 og 3.65. Det er ikke like voldsomt som spotprisene som svinger mellom 3.2 og 3.8. Vi plotter spotprisene sammen med plottet for kontraktene for å tydeliggjøre dette.

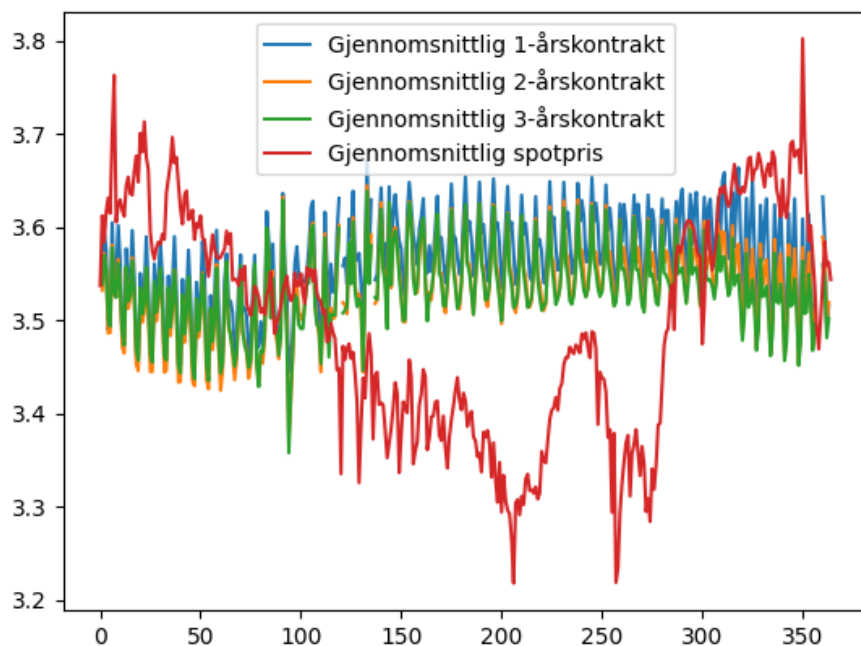


Figure 11: Gjennomsnittlig spotpris sammen med gjennomsnittlige kontraktspriser.

Her ser vi at funksjonene for gjennomsnittlig kontraktspris holder seg nokså stabilt mellom 3.4 og 3.6. For kontraktsprisene velger vi derfor en konstant μ som driftsledd. Vi kan enkelt finne μ for kontraktene ved å beregne gjennomsnittet av alle dataene.

- 1-årskontrakter: $\mu = 3.568469293376401$
- 2-årskontrakter: $\mu = 3.5380112638971624$
- 3-årskontrakter: $\mu = 3.5333447299835687$

For å gjøre det enklere videre i analysen, kan vi behandle μ som en funksjon som avhenger av tid. Vi definerer $m_t = \mu$ slik at m_t er den samme verdien for hver t . Ved å definere driftsleddet som en funksjon for både spotprisene og kontraktene, vil de ha samme form og det vil gjøre det enklere å analysere dem i et felles skript i Python.

4.1.3 Estimere θ i én dimensjon

Vi har tilgjengelig historisk data og definert et driftsledd m_t som vil si at vi kan beregne en verdi for $\hat{\theta}$ som beskriver dataenes reverteringshastighet mot m_t . Vi har allerede et uttrykk for $\hat{\theta}$ som vi fant ved maximum likelihood estimation og et skript som returnerer denne verdien ved å ta inn argumentene x , som er prosessen, T som er tiden, n som er antall tidssteg og m som er driftsleddet.

Vi bruker dataene vi har for strømprisene som argumentet x , gjennomsnittlig strømpris for m og lengden til dataserien som n og T . Dette gir oss følgende verdier for $\hat{\theta}$:

- Spotpriser: $\hat{\theta} = 0.04291111970866316$
- 1-årskontrakter: $\hat{\theta} = 0.00175722020106258$
- 2-årskontrakter: $\hat{\theta} = 0.0010720609306570249$
- 3-årskontrakter: $\hat{\theta} = 0.0007471569834810026$

Det kan se ut som om spotprisene reverteres tilbake til gjennomsnittet med en høyere hastighet enn kontraktsprisene over tid. Dette kan forklares ved at spotpriser er priser på øyeblikkelig levering, og er derfor mer påvirket av kortsiktige tilbud- og etterspørselsfaktorer enn kontraktspriser som er avtalt på forhånd for en bestemt tidsperiode. Kontraktspriser vil dermed trolig ha en mer jevn utvikling over tid, og det kan derfor ta lenger tid før de reverteres tilbake til gjennomsnittet.

4.1.4 Uavhengige modeller for spotpriser og kontraktene

Vi har definert driftsledet m_t og vi har verdier for estimatene $\hat{\sigma}$ og $\hat{\theta}$ for spotpriser, 1-årskontrakter, 2-årskontrakter og 3-årskontrakter. Vi kan nå definere prosessen

$$dX_t = \theta(m_t - X_t)dt + \sigma dB_t \quad (1.1)$$

med de parameterne som passer best til våre data. Skriptet under er generell og skal passe til alle fire variablene vi har.

```
1 import numpy as np
2 import pandas as pd
3 import matplotlib.pyplot as plt
4 import theta1d
5 import sigma1d
6 import white_noise
7
8 def model(n, T, sigma, theta, m):
9     t = np.linspace(0, T, n)
10    dt = t[1] - t[0]
11
12    WN = white_noise.simulate_WN(1, n, T)
13
14    x = np.zeros(T)
15    x[0] = m[0]
16
17    for i in range(0, T-1):
18        x[i+1] = x[i] + theta * (m[i] - x[i]) * dt \
19            + sigma * WN[0, i]
20
21    return x
```

Funksjonen tar inn fem argumenter, n og T som er lengden av dataene, samt parameterne σ , θ og m som representerer volatiliteten, snittreverterings hastigheten og driftsledet. Disse tre er definert for hver av dataene vi har. Vi trenger nå bare å importere skriptene som beregner $\hat{\sigma}$ og $\hat{\theta}$ og de nye datafilene vi lagde for m_t som er gjennomsnittlig verdi av strømprisene. Nedenfor ser vi modeller for estimerte spotpriser plottet sammen med det faktiske gjennomsnittet av spotprisene.



Figure 12: Estimerte spotpriser sammenlignet med gjennomsnittlig spotpris.

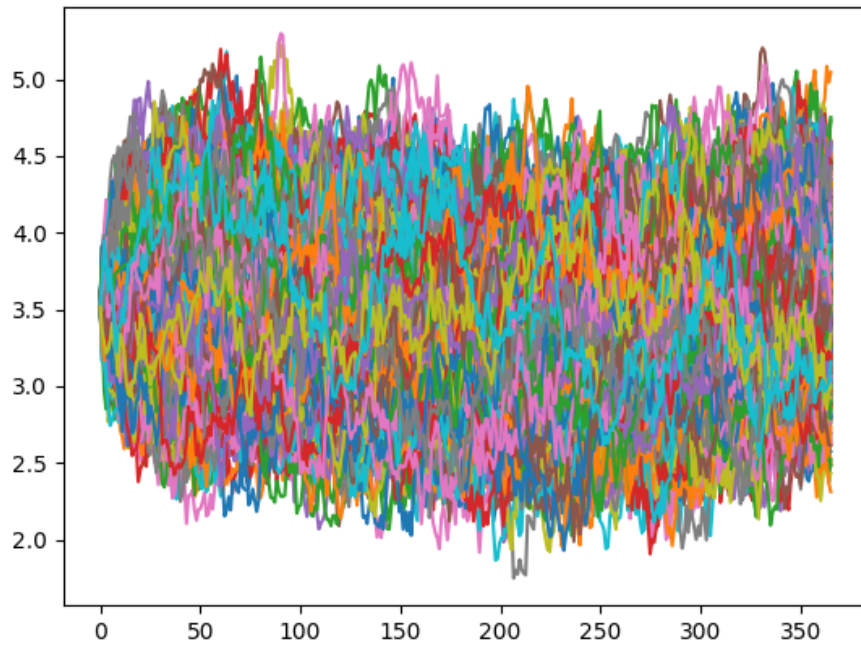


Figure 13: 1000 estimeringer av spotprisene.

Vi har kjørt skriptet 1000 ganger og ovenfor ser vi alle plottene i samme bilde. Prosessen vår inneholder inkrementene av en brownsk bevegelse og snittreverteringshastigheten er ikke særlig høy. Prosessen vil derfor variere veldig og det er ganske tilfeldig hvordan den kommer til å se ut. Slik er det for spotpriser også, så vi kan forestille oss at modellen klarer å representere spotprisenes bilde på en nøyaktig måte. Det er likevel en grense på hvor mye prisene kan variere, og vi ser at prosessen holder seg omtrent mellom verdiene 2 og 5.

Vi gjør dette for kontraktsprisene også, og får følgende modeller for 1-årskontrakter



Figure 14: Estimerte 1-årskontrakter sammenlignet med gjennomsnittlig 1-årskontrakt.

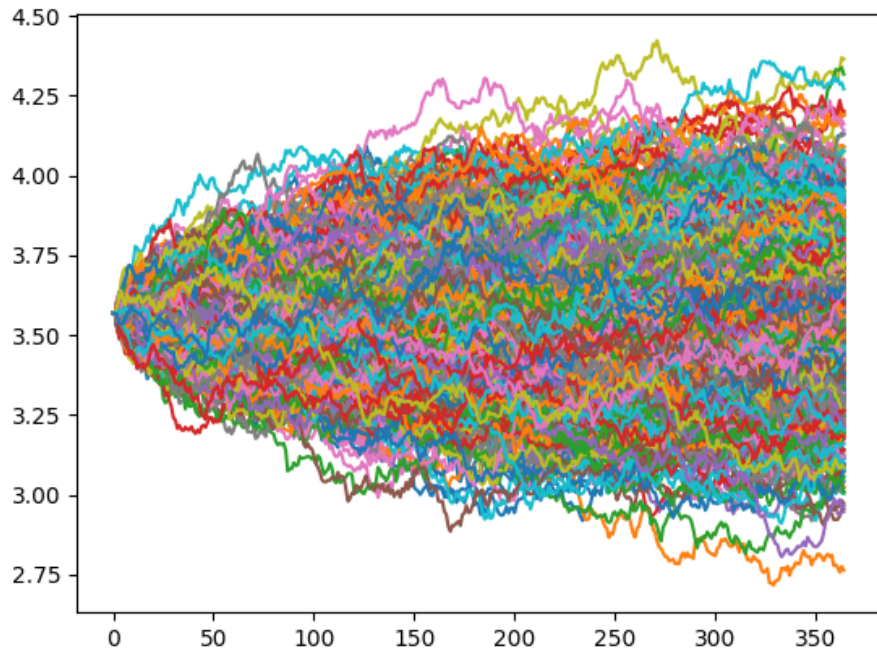


Figure 15: 1000 estimeringer av 1-årskontraktene.

for 2-årskontrakter

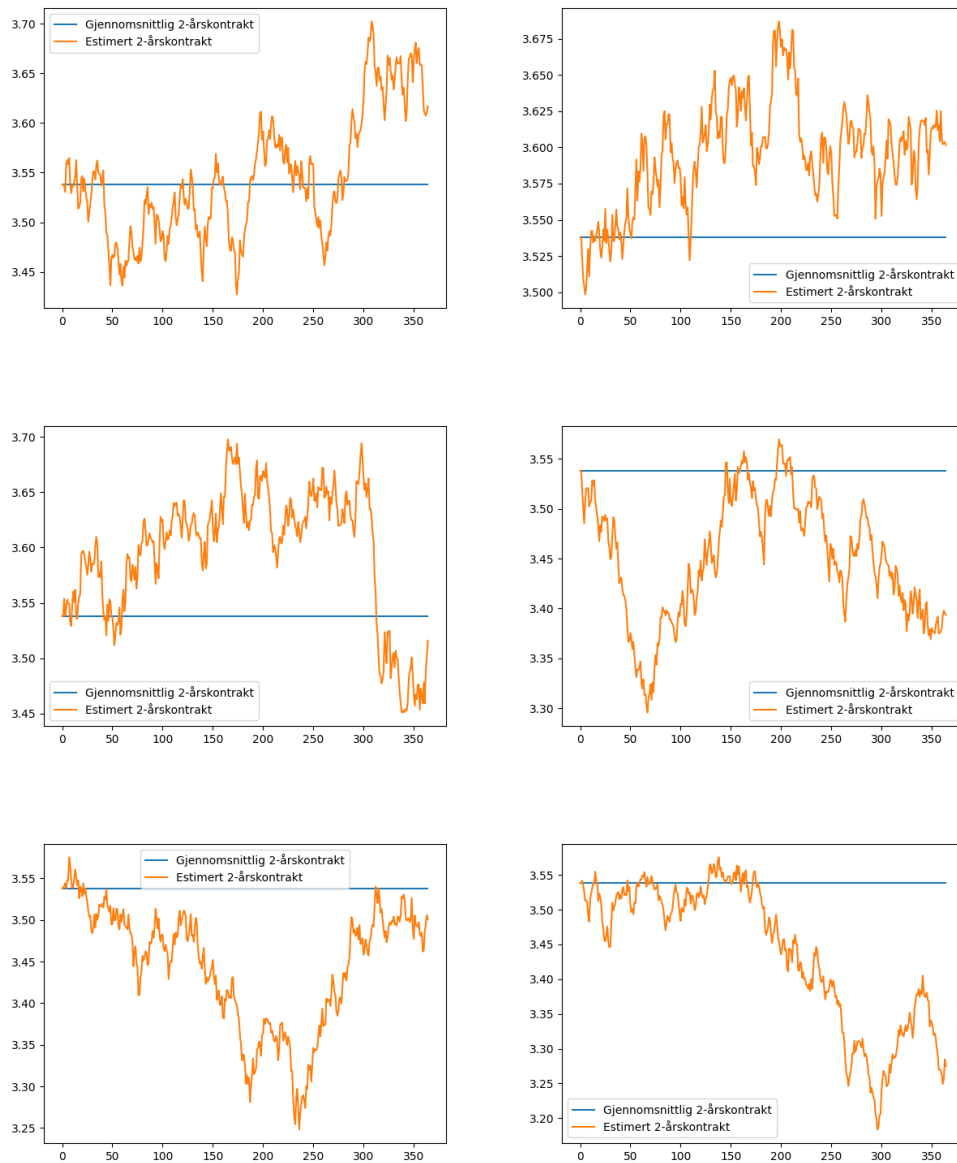


Figure 16: Estimerte 2-årskontrakter sammenlignet med gjennomsnittlig 2-årskontrakt.

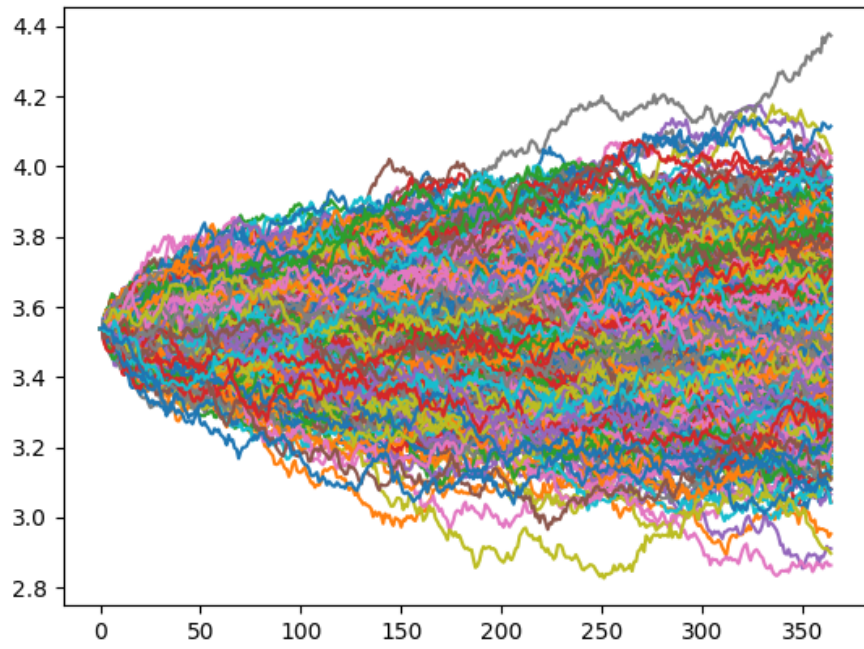


Figure 17: 1000 estimeringer av 2-årskontraktene.

og for 3-årskontrakter



Figure 18: Estimerte 3-årskontrakter sammenlignet med gjennomsnittlig 3-årskontrakt.

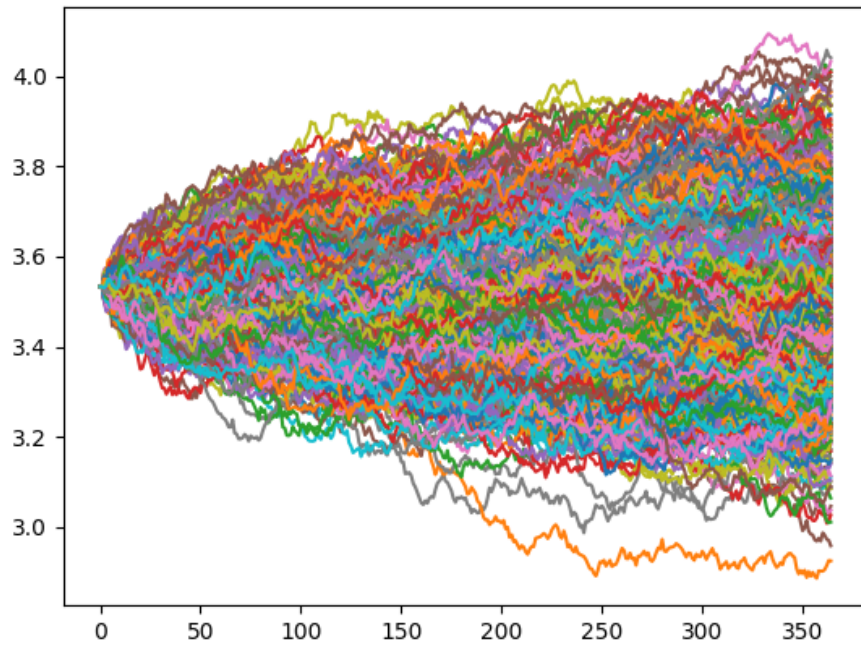


Figure 19: 1000 estimeringer av 3-årskontraktene.

Generelt sett er markedene for strøm svært volatile, og prisene kan svinge mye over tid. Vi vet at dette skyldes flere faktorer som er vanskelig å forutsi. En modell som skal representere strømpriser bør derfor ta hensyn til denne tilfeldigheten og usikkerheten, slik som disse modellene gjør.

4.2 Modell i flere dimensjoner

Spotprisene og langsiktige kontraktspriser er variabler som avhenger av hverandre og det finnes derfor en korrelasjon mellom disse prisene. Endringer i spotprisene kan ha en innvirkning på hva som anses som en rimelig pris for en langsiktig kontrakt, og det er derfor viktig å ta hensyn til denne korrelasjonen når man modellerer priser i strømmarkedet. Det finnes også andre faktorer som spiller inn på strømprisene, men modellen vi skal lage skal kun fokusere på historisk data og hvordan disse fungerer i sammenheng.

Vi skal utvikle en 4-dimensjonell modell som tar hensyn til korrelasjonen mellom de fire strømprisene. Vi vil fortsatt bruke Ornstein-Uhlenbeck prosessen med m_t som driftsledd

$$dX_t = \theta(m_t - X_t)dt + \sigma dB_t. \quad (1.1)$$

Forskjellen fra da vi lagde modellen for én dimensjon er at denne prosessen vil nå være 4-dimensjonell. Vi har altså

$$\underbrace{dX_t}_{\in \mathbb{R}^4} = \underbrace{\theta}_{\in \mathbb{R}^{4 \times 4}} \left(\underbrace{m_t}_{\in \mathbb{R}^4} - \underbrace{X_t}_{\in \mathbb{R}^4} \right) dt + \underbrace{\sigma}_{\in \mathbb{R}^{4 \times 4}} \underbrace{dB_t}_{\in \mathbb{R}^4} \quad (4.1)$$

hvor $X_t = \begin{pmatrix} X_t^{(0)} \\ X_t^{(1)} \\ X_t^{(2)} \\ X_t^{(3)} \end{pmatrix}$ og komponentene til X_t representerer følgende verdier

$$\begin{aligned} X_t^{(0)}: & \quad \text{Spotpriser,} \\ X_t^{(1)}: & \quad \text{1-årskontrakter,} \\ X_t^{(2)}: & \quad \text{2-årskontrakter,} \\ X_t^{(3)}: & \quad \text{3-årskontrakter.} \end{aligned}$$

Vi ser at nå er σ og θ matriser og m_t er en vektor. Vi må derfor beregne $\hat{\sigma}$ og $\hat{\theta}$ på nytt med nye formler som passer for det flerdimensjonelle tilfellet.

Når vi beveger oss fra én dimensjon til flere dimensjoner, så øker kompleksiteten til modellen. I det 1-dimensjonelle trengte vi kun å forholde oss til én og én variabel, altså vi regnet ut parameterne for spotpriser, 1-årskontrakter, 2-årskontrakter og 3-årskontrakter hver for seg. Disse variablene må vi håndtere sammen i det flerdimensjonelle tilfellet.

Hvis vi klarer å utvikle en god og effektiv metode for å beregne $\hat{\sigma}$ og $\hat{\theta}$ i for eksempel to dimensjoner, så vil denne metoden også kunne brukes i andre dimensjoner. Dette betyr at i praksis har vi egentlig bare to tilfeller når vi jobber med disse modellene: én for 1-dimensjonelle modeller og én for flerdimensjonelle modeller.

Vi har data for spotpriser, 1-årskontrakter, 2-årskontrakter og 3-årskontrakter. For å kunne sammenligne parameterne for disse fire variablene, trenger vi at dataene er i samme tidsperiode og har samme antall tidssteg. Dataene er allerede i samme tidsperiode, altså fra 2007 til og med 2019.

Dataseriene har foreløpig ikke samme antall tidssteg. Spotprisene er tilgjengelige for hver dag i året, mens kontraktene bare er tilgjengelige for de dagene som er handledager. Kontraksprisene vil derfor ha færre tidssteg enn spotprisene. For å løse dette problemet, begrenser vi spotprisene til kun de dagene som er handledager. Dette vil sørge for at både spotprisene og kontraktene har samme antall tidssteg, og kan dermed analyseres sammen.

Vi kombinerer disse dataene i en felles dataserie. På den måten blir det enklere å jobbe med dataene i Python ved at vi kan bruke de samme funksjonene og metodene for å undersøke både spot- og kontraktpriser.

	Dato	1-årskontrakt	2-årskontrakt \
0	2007-01-02	3.695110	3.737670
1	2007-01-03	3.686376	3.725693
2	2007-01-04	3.686376	3.719651
3	2007-01-05	3.671225	3.702536
4	2007-01-08	3.634951	3.660994
...
3253	2019-12-19	3.593194	3.526655
3254	2019-12-20	3.590439	3.530763
3255	2019-12-23	3.579344	3.527830
3256	2019-12-27	3.557061	3.523415
3257	2019-12-30	3.542697	3.501043

	3-årskontrakt	Spot
0	3.747148	3.478390
1	3.742420	3.475931
2	3.737670	3.421640
3	3.722798	3.402017
4	3.688879	3.348208
...
3253	3.498324	3.624085
3254	3.502851	3.607637
3255	3.501646	3.607659
3256	3.500439	3.635128
3257	3.494991	3.519771

[3258 rows x 5 columns]

Nå som vi har sikret at alle dataseriene har samme tidsperiode og samme antall tidssteg, kan vi beregne estimatene.

4.2.1 Estimere σ i flere dimensjoner

Det finnes en korrelasjon mellom strømprisene som vil si at hvordan en pris endrer seg vil ha effekt på hvordan de andre prisene oppfører seg. I det flerdimensjonelle tilfellet er σ en $d \times d$ matrise for å få med denne avhengigheten mellom de ulike prisene.

Vi har tidligere beregnet $\hat{\sigma}$ for det 1-dimensjonelle tilfellet. I bunn og grunn er det ingen konseptuelle forskjeller mellom å beregne $\hat{\sigma}$ i én dimensjon og å beregne $\hat{\sigma}$ i flere dimensjoner. Forskjellen er at nå er $\hat{\sigma}$ en $d \times d$ matrise og vi bruker kovariasjonsmatrisen istedenfor kvadratisk variasjon.

Kovariasjonsmatrisen er matrisen som inneholder alle kryssvariasjoner. Vi innfører denne notasjonen

$$\langle\langle X \rangle\rangle_t = \begin{pmatrix} \langle X^{(0)} \rangle_t & \langle X^0, X^1 \rangle_t & \dots & \langle X^{(0)}, X^{(d-1)} \rangle_t \\ \langle X^{(1)}, X^{(0)} \rangle_t & \langle X^1 \rangle_t & \dots & \langle X^{(1)}, X^{(d-1)} \rangle_t \\ \vdots & \vdots & \ddots & \vdots \\ \langle X^{(d-1)}, X^{(0)} \rangle_t & \langle X^{(d-1)}, X^{(1)} \rangle_t & \dots & \langle X^{(d-1)} \rangle_t \end{pmatrix}. \quad (4.2)$$

Generelt gjelder det at hver komponent (i, j) i kovariasjonsmatrisen kan skrives som

$$(\langle\langle X \rangle\rangle_t)^{(i,j)} = \langle X^{(i)}, X^{(j)} \rangle_t \quad (4.3)$$

(Øksendal, 2003, s. 315-316).

Vi legger merke til at langs diagonalen, altså når $i = j$, så er denne komponenten lik den kvadratiske variasjonen av $X_t^{(i)}$. Den kvadratiske variasjonen er positiv, som betyr at alle komponentene langs diagonalen til kovariasjonsmatrisen er positiv. I tillegg er kovariasjonen symmetrisk, altså

$$\begin{aligned} \langle X, Y \rangle_t &= \lim_{n \rightarrow \infty} \sum_{i=0}^n (X_{t_{i+1}} - X_{t_i}) (Y_{t_{i+1}} - Y_{t_i}) \\ &= \lim_{n \rightarrow \infty} \sum_{i=0}^n (Y_{t_{i+1}} - Y_{t_i}) (X_{t_{i+1}} - X_{t_i}) \\ &= \langle Y, X \rangle_t. \end{aligned} \quad (4.4)$$

Dette betyr at komponentene over og under diagonalen er like hverandre. Disse egenskapene gjør at kovariasjonsmatrisen er en symmetrisk og positiv definit matrise.

Kovariasjonsmatrisen $\langle\langle X \rangle\rangle_t$ vil inneholde kovariasjonen mellom alle komponentene, og vil være definert som $\sigma^2 t$ delt på tiden

$$\langle\langle X \rangle\rangle_t = \sigma^2 t = \sigma \times \sigma t \quad (4.5)$$

hvor

$$\sigma = \begin{pmatrix} \sigma_{00} & \cdots & \sigma_{0(d-1)} \\ \vdots & \ddots & \vdots \\ \sigma_{(d-1)0} & \cdots & \sigma_{(d-1)(d-1)} \end{pmatrix}.$$

Vi ønsker å finne et uttrykk for alle komponentene til $\langle\langle X \rangle\rangle_t$.

Før vi generaliserer metoden, ser vi først på det 2-dimensjonelle tilfellet. La X_t være prosessen (3.5) i to dimensjoner, altså

$$\begin{aligned} dX_t &= \sigma dB_t \\ d \begin{pmatrix} X_t^{(0)} \\ X_t^{(1)} \end{pmatrix} &= \begin{pmatrix} \sigma_{00} & \sigma_{01} \\ \sigma_{10} & \sigma_{11} \end{pmatrix} d \begin{pmatrix} B_t^{(0)} \\ B_t^{(1)} \end{pmatrix} \\ &= \begin{pmatrix} \sigma_{00} dB_t^{(0)} + \sigma_{01} dB_t^{(1)} \\ \sigma_{10} dB_t^{(0)} + \sigma_{11} dB_t^{(1)} \end{pmatrix}. \end{aligned}$$

Hver komponent av X_t kan skrives slik

$$\begin{aligned} X_t^{(0)} &= \sigma_{00} B_t^{(0)} + \sigma_{01} B_t^{(1)}, \\ X_t^{(1)} &= \sigma_{10} B_t^{(0)} + \sigma_{11} B_t^{(1)}, \end{aligned}$$

og kovariasjonsmatrisen for denne prosessen er lik

$$\langle\langle X \rangle\rangle_t = \begin{pmatrix} \langle X^{(0)} \rangle_t & \langle X^{(0)}, X^{(1)} \rangle_t \\ \langle X^{(1)}, X^{(0)} \rangle_t & \langle X^{(1)} \rangle_t \end{pmatrix}. \quad (4.6)$$

Vi ønsker å finne et uttrykk for hver av komponentene i matrisen, altså $\langle X^{(0)} \rangle_t$, $\langle X^{(0)}, X^{(1)} \rangle_t = \langle X^{(1)}, X^{(0)} \rangle_t$ og $\langle X^{(1)} \rangle_t$. Vi husker at den kvadratiske variasjonen til en brownsk bevegelse er lik t , som er et resultat vi kommer til å trenge. Vi trenger også kovariasjonen til to uavhengige brownske bevegelser, som vi vet fra Lévy's karakteriseringsteorem for to dimensjoner (2.3) er lik 0

$$\underbrace{\langle B^{(0)}, B^{(1)} \rangle_t}_{\text{Uavhengige brownske bevegelser}} = 0. \quad (4.7)$$

Det er også viktig å merke seg at kovariasjonen er bilinear

$$\langle \alpha X + \beta Z, Y \rangle_t = \alpha \langle X, Y \rangle_t + \beta \langle Z, Y \rangle_t \quad (4.8)$$

(Kessler et al., 2012, s. 128). Vi bruker dette til å vise at kvadratsetningene også gjelder for kovariasjonen

$$\begin{aligned} \langle \alpha X + \beta Y \rangle_t &= \langle \alpha X + \beta Y, \alpha X + \beta Y \rangle_t \\ &= \alpha^2 \langle X \rangle_t + 2\alpha\beta \langle X, Y \rangle_t + \beta^2 \langle Y \rangle_t. \end{aligned} \quad (4.9)$$

Vi kan nå regne ut de ulike komponentene for kovariasjonsmatrisen i to dimensjoner.

1.

$$\begin{aligned} \langle X^{(0)} \rangle_t &= \langle \sigma_{00} B^{(0)} + \sigma_{01} B^{(1)} \rangle_t \\ &= \sigma_{00}^2 \underbrace{\langle B^{(0)} \rangle_t}_t + 2\sigma_{00}\sigma_{01} \underbrace{\langle B^{(0)}, B^{(1)} \rangle_t}_0 + \sigma_{01}^2 \underbrace{\langle B^{(1)} \rangle_t}_t \\ &= \sigma_{00}^2 t + \sigma_{01}^2 t \end{aligned}$$

2.

$$\begin{aligned} \langle X^{(1)} \rangle_t &= \langle \sigma_{10} B^{(0)} + \sigma_{11} B^{(1)} \rangle_t \\ &= \sigma_{10}^2 \underbrace{\langle B^{(0)} \rangle_t}_t + 2\sigma_{10}\sigma_{11} \underbrace{\langle B^{(0)}, B^{(1)} \rangle_t}_0 + \sigma_{11}^2 \underbrace{\langle B^{(1)} \rangle_t}_t \\ &= \sigma_{10}^2 t + \sigma_{11}^2 t. \end{aligned}$$

3.

$$\begin{aligned} \langle X^{(0)}, X^{(1)} \rangle_t &= \langle \sigma_{00} B^{(0)} + \sigma_{01} B^{(1)}, \sigma_{10} B^{(0)} + \sigma_{11} B^{(1)} \rangle_t \\ &= \sigma_{00}\sigma_{10} \underbrace{\langle B^{(0)} \rangle_t}_t + \sigma_{00}\sigma_{11} \underbrace{\langle B^{(0)}, B^{(1)} \rangle_t}_0 \\ &\quad + \sigma_{01}\sigma_{10} \underbrace{\langle B^{(1)}, B^{(0)} \rangle_t}_0 + \sigma_{01}\sigma_{11} \underbrace{\langle B^{(1)} \rangle_t}_t \\ &= \sigma_{00}\sigma_{10} t + \sigma_{01}\sigma_{11} t \end{aligned}$$

Vi ser et mønster her og ser at vi kan skrive hver komponent (i, j) i matrisen slik

$$(\langle\langle X \rangle\rangle_t)^{(i,j)} = \langle X^{(i)}, X^{(j)} \rangle_t = \left\langle \sum_{k=0}^{d-1} \sigma_{ik} B^{(k)}, \sum_{l=0}^{d-1} \sigma_{jl} B^{(l)} \right\rangle_t.$$

Vi vet at

$$\begin{cases} \langle B^{(k)}, B^{(l)} \rangle_t = 0 \text{ når } k \neq l \\ \langle B^{(k)}, B^{(l)} \rangle_t = t \text{ når } k = l \end{cases}, \quad (4.10)$$

som betyr at vi bare vil ha igjen de leddene hvor $k = l$ da de andre vil være lik 0. Vi kan dermed forenkle uttrykket for komponent (i, j)

$$(\langle\langle X \rangle\rangle_t)^{(i,j)} = \langle X^{(i)}, X^{(j)} \rangle_t = \sum_{k=0}^{d-1} \sigma_{ik} \sigma_{jk} t. \quad (4.11)$$

$\langle\langle X \rangle\rangle_t$ kan lett implementeres på en datamaskin, som gjør at vi enkelt kan beregne $\hat{\sigma}$. Det eneste som gjenstår for å finne $\hat{\sigma}$ er å ta kvadratroten av $\langle\langle X \rangle\rangle_t$ delt på t .

```

1 import numpy as np
2 import scipy
3
4 def est_sigma(d, T, x):
5     #Beregner kovariasjonsmatrisen av prosessen
6     var = np.zeros((d,d))
7     for i in range(0, d):
8         for j in range(0, d):
9             var[i][j] = np.sum(np.diff(x[i]) * np.diff(x[j]))
10    #Volatiliteten er gitt ved kvadratroten av
11    #kovariasjonsmatrisen delt på tiden
12    estimated_sigma = scipy.linalg.sqrtm(var/T)
13    return estimated_sigma

```

Dette skriptet starter med en nullmatrise som etterhvert skal fylles opp med kovariasjonen mellom alle mulige par av komponentene i x . x kan være en stokastisk prosess, eller det kan være data med flere kolonner. d tilsvarer dimensjonen og T er tiden.

Før vi bruker skriptet på faktisk data for strømprisene, tester vi det først ut på en stokastisk prosess som vi selv konstruerer. Vi bruker samme prosessen som vi brukte for å teste ut det 1-dimensjonelle tilfellet, altså

$$dX_t = \sigma dB_t \quad (3.5)$$

der B_t er en flerdimensjonell brownsk bevegelse og σ er en vilkårlig $d \times d$ matrise. Ved å vite den faktiske verdien for σ , kan vi enklere sammenligne resultatene vi får fra skriptet for $\hat{\sigma}$.

```

1 import numpy as np
2 import brownian_motion
3
4 def prosess(d, n, T, sigma):
5     BM = brownian_motion.simulate_BM(d, n, T)
6     x = sigma@BM
7     return x

```

Vi lar X_t være en 3-dimensjonell prosess med 50000 tidssteg fra 0 til 10 og vi setter $\sigma = \begin{pmatrix} 2 & 3 & 4 \\ 3 & 5 & 6 \\ 4 & 6 & 10 \end{pmatrix}$. Med denne prosessen får vi følgende estimat

$$\hat{\sigma} = \begin{pmatrix} 2.002944 & 3.00455185 & 4.00868902 \\ 3.00455185 & 5.00592756 & 6.01213786 \\ 4.00868902 & 6.01213786 & 10.02593723 \end{pmatrix}.$$

Vi ser at denne verdien ligner på den faktiske verdien for σ . For å tydeliggjøre dette, beregner vi normen til differansen mellom disse to matrisene. Vi får at differansen er 0.03469344235877567, som er en nokså lav verdi. Vi kan dermed stole på at skriptet gir oss et godt estimat for σ .

Nå som vi har testet ut dette, kan vi bruke dataene for strømpriser i skriptet. Dataserien vi har inneholder fem kolonner hvor den ene er dato-kolonnen og de andre fire er data for strømpriser. Hver kolonne med data for strømpriser vil tilsvare hver komponent i vektoren X_t

$$X_t = \begin{pmatrix} X_t^{(0)} \\ X_t^{(1)} \\ X_t^{(2)} \\ X_t^{(3)} \end{pmatrix} = \begin{pmatrix} 3.478390 & 3.475931 & \dots & 3.635128 & 3.519771 \\ 3.69511 & 3.686376 & \dots & 3.557061 & 3.542697 \\ 3.737670 & 3.725693 & \dots & 3.523415 & 3.501043 \\ 3.747148 & 3.742420 & \dots & 3.500439 & 3.494991 \end{pmatrix}.$$

Vi bruker T lik tidsstegene, som er 3258 og $d = 4$. Med x definert som X_t , som er vektoren som inneholder alle dataene for strømpriser, får vi følgende verdi for $\hat{\sigma}$

$$\hat{\sigma} = \begin{pmatrix} 0.11223847 & -0.00015466 & -0.0002181 & -0.00022423 \\ -0.00015466 & 0.01409231 & 0.00646873 & 0.00475871 \\ -0.0002181 & 0.00646873 & 0.00946879 & 0.00545703 \\ -0.00022423 & 0.00475871 & 0.00545703 & 0.007893 \end{pmatrix},$$

og kovariasjonsmatrise lik

$$\langle\langle X \rangle\rangle_t = \begin{pmatrix} 41.04296765 & -0.07172793 & -0.09372714 & -0.09403635 \\ -0.07172793 & 0.85720196 & 0.58126776 & 0.45597803 \\ -0.09372714 & 0.58126776 & 0.52561022 & 0.40912495 \\ -0.09403635 & 0.45597803 & 0.40912495 & 0.37393444 \end{pmatrix}.$$

Vi definerer kovariasjonsmatrisen i fire dimensjoner på følgende måte:

$$\langle\langle X \rangle\rangle_t = \begin{pmatrix} \langle X^{(0)} \rangle_t & \langle X^{(0)}, X^{(1)} \rangle_t & \langle X^{(0)}, X^{(2)} \rangle_t & \langle X^{(0)}, X^{(3)} \rangle_t \\ \langle X^{(1)}, X^{(0)} \rangle_t & \langle X^{(1)} \rangle_t & \langle X^{(1)}, X^{(2)} \rangle_t & \langle X^{(1)}, X^{(3)} \rangle_t \\ \langle X^{(2)}, X^{(0)} \rangle_t & \langle X^{(2)}, X^{(1)} \rangle_t & \langle X^{(2)} \rangle_t & \langle X^{(2)}, X^{(3)} \rangle_t \\ \langle X^{(3)}, X^{(0)} \rangle_t & \langle X^{(3)}, X^{(1)} \rangle_t & \langle X^{(3)}, X^{(2)} \rangle_t & \langle X^{(3)} \rangle_t \end{pmatrix}. \quad (4.12)$$

Vi ser at alle kovariasjonene mellom spotpriser og kontraktspriser er negative, mens resten av kovariasjonene er positive. Kovariasjonen er et mål på hvordan to tilfeldige variabler endrer seg i forhold til hverandre. Hvis kovariasjonen mellom to variabler er negativ, betyr det at de to variablene har en tendens til å bevege seg i motsatt retning av hverandre. Det vil så at når en av de øker, så vil den andre minke, og omvendt. Positiv kovariasjon betyr at variablene har en tendens til å bevege seg i samme retning, altså at de øker eller minker sammen. Det kan se ut til at når spotprisene øker, så minker kontraktene, og omvendt.

4.2.2 Estimere θ i flere dimensjoner

Når vi skal estimere θ i det flerdimensjonelle tilfellet, bruker vi maximum likelihood estimation slik som vi gjorde for det 1-dimensjonelle tilfellet. Vi trenger en ny likelihoodfunksjon som passer til vår 4-dimensjonelle prosess.

Prosessens vår

$$dX_t = \theta(m_t - X_t)dt + \sigma dB_t \quad (1.1)$$

er av samme form som prosessen som nevnes i Girsanovs teorem del II

$$dY_t = b(t; \theta)dt + \sigma(t)dB_t. \quad (2.13)$$

Dette vil si at

$$\begin{aligned} dY_t &= dX_t \in \mathbb{R}^d, \\ b(t; \theta)dt &= \theta(m_t - X_t)dt \in \mathbb{R}^d, \\ \sigma(t) &= \sigma \in \mathbb{R}^{d \times d}. \end{aligned}$$

For at dette skal gi mening, må X_t , m_t og B_t være $d \times 1$ vektorer, og θ og σ må være $d \times d$ matriser

$$\underbrace{dX_t}_{d \times 1} = \underbrace{\theta}_{d \times d} \underbrace{(m_t - X_t)}_{d \times 1} dt + \underbrace{\sigma}_{d \times d} \underbrace{dB_t}_{d \times 1}.$$

Vi definerer hver av elementene i prosessen slik

$$\begin{aligned} X_t &= \begin{pmatrix} X_t^{(0)} \\ \vdots \\ X_t^{(d-1)} \end{pmatrix}, m_t = \begin{pmatrix} m_t^{(0)} \\ \vdots \\ m_t^{(d-1)} \end{pmatrix}, B_t = \begin{pmatrix} B_t^{(0)} \\ \vdots \\ B_t^{(d-1)} \end{pmatrix}, \\ \theta &= \begin{pmatrix} \theta_0 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & \theta_{d-1} \end{pmatrix}, \sigma = \begin{pmatrix} \sigma_{00} & \dots & \sigma_{0(d-1)} \\ \vdots & \ddots & \vdots \\ \sigma_{(d-1)0} & \dots & \sigma_{(d-1)(d-1)} \end{pmatrix}. \end{aligned}$$

σ er en symmetrisk, positiv definit matrise, og θ er en diagonal matrise med nuller overalt utenom diagonalen.

Vi må finne en u som tilfredsstiller (2.14), altså $\sigma u = \theta(m_t - X_t)$. Vi velger

$$u = \underbrace{\underbrace{\sigma^{-1}}_{d \times d} \underbrace{\theta(m_t - X_t)}_{d \times 1}}_{d \times 1} \in \mathbb{R}^d.$$

Vi repeterer definisjonen av M_t

$$M_t = \exp \left(- \int_0^t u(s; \theta) dB_s - \frac{1}{2} \int_0^t u^2(s; \theta) ds \right); \quad t \leq T. \quad (2.15)$$

I første leddet har vi $u(s; \theta) dB_s$. Siden både u og B_t er $d \times 1$ vektorer, må u være transponert for at dette uttrykket skal gi mening. Ser vi på andre leddet, så har vi $u^2(s; \theta)$. u er en vektor, og det gir ikke mening å opphøye en vektor i andre. Det vi kan gjøre derimot er å ta prikkproduktet mellom u og dens transponerte.

Vi skriver om definisjonen av M_t slik at den skal gjelde for dimensjoner høyere enn én

$$M_t = \exp \left(- \int_0^t u^T dB_s - \frac{1}{2} \int_0^t (u^T u) ds \right); \quad t \leq T. \quad (4.13)$$

Som vi så tidligere, så er det M_t^{-1} som blir vår likelihoodfunksjon

$$M_t^{-1} = \exp \left(\int_0^t u^T dB_s + \frac{1}{2} \int_0^t (u^T u) ds \right).$$

Vi ønsker et uttrykk som er integrert med hensyn på dX_s og ikke dB_s . Det vi kan gjøre er å bruke prosessen vår dX_t og skrive det om slik at vi får et uttrykk for dB_t .

$$\begin{aligned} \theta(m_t - X_t)dt + \sigma dB_t &= dX_t \\ \sigma dB_t &= dX_t - \theta(m_t - X_t)dt \\ dB_t &= \sigma^{-1} dX_t - \sigma^{-1} \theta(m_t - X_t)dt \end{aligned}$$

Vi har nå et uttrykk for dB_t som vi kan sette inn i M_t^{-1} .

$$\begin{aligned}
M_t^{-1} &= \exp \left(\int_0^t u^T dB_s + \frac{1}{2} \int_0^t (u^T u) ds \right) \\
&= \exp \left(\int_0^t u^T (\sigma^{-1} dX_s - \sigma^{-1} \theta(m_s - X_s) ds) + \frac{1}{2} \int_0^t (u^T u) ds \right) \\
&= \exp \left(\int_0^t u^T \sigma^{-1} dX_s - \int_0^t u^T \sigma^{-1} (\theta(m_s - X_s) ds) + \frac{1}{2} \int_0^t (u^T u) ds \right) \\
&= \exp \left(\int_0^t u^T \sigma^{-1} dX_s - \int_0^t \underbrace{u^T \sigma^{-1} (\theta(m_s - X_s) ds)}_u + \frac{1}{2} \int_0^t (u^T u) ds \right) \\
&= \exp \left(\int_0^t u^T \sigma^{-1} dX_s - \int_0^t (u^T u) ds + \frac{1}{2} \int_0^t (u^T u) ds \right) \\
&= \exp \left(\int_0^t u^T \sigma^{-1} dX_s - \frac{1}{2} \int_0^t (u^T u) ds \right)
\end{aligned}$$

Vi får at loglikelihoodfunksjonen vår er lik

$$\ell(\theta) = \int_0^t u^T \sigma^{-1} dX_s - \frac{1}{2} \int_0^t (u^T u) ds. \quad (4.14)$$

Vi ønsker å omformulere denne likningen slik at det blir enklere å derivere med hensyn på θ . Vi vet at $u = \sigma^{-1} \theta(m_t - X_t)$ er en funksjon med hensyn på θ som avhenger av tiden t , så vi skriver $u_t(\theta)$. σ^{-1} er en matrise, og det er ikke viktig nøyaktig hva komponentene er, men vi vet at vi skal bruke komponenter av den inverse av σ . Vi velger å bruke $\sigma^{-1} = A$, så ser vi kun på komponentene i matrisen A . Vi får følgende omskriving av loglikelihoodfunksjonen

$$\ell(\theta) = \int_0^t u_t^T(\theta) A dX_s - \frac{1}{2} \int_0^t (u_t(\theta)^T u_t(\theta)) ds.$$

Vi har dette uttrykket

$$u_t(\theta) = A\theta(m_t - X_t)$$

hvor

$$A = \begin{pmatrix} a_{00} & \dots & a_{0(d-1)} \\ \vdots & \ddots & \vdots \\ a_{(d-1)0} & \dots & a_{(d-1)(d-1)} \end{pmatrix}.$$

Dette er en vektor, som vil si at vi kan definere $u_t(\theta)$ på denne måten også

$$u_t(\theta) = \begin{pmatrix} u_t^{(0)}(\theta) \\ \vdots \\ u_t^{(d-1)}(\theta) \end{pmatrix}.$$

Vi ønsker å finne et uttrykk for hver komponent $u_t^{(j)}(\theta)$ for $j \in \{0, 1, \dots, d-1\}$. Vi vet at $u_t(\theta) = A\theta(m_t - X_t)$, så vi starter med å se på $A\theta$, som er to $d \times d$ matriser multiplisert med hverandre. Produktet av A og θ vil være en ny $d \times d$ matrise. Komponentene c_{ij} i matrisen $A\theta$ er gitt ved følgende formel:

$$c_{ij} = a_{i1} \times \theta_{1j} + a_{i2} \times \theta_{2j} + \dots + a_{in} \times b_{nj}. \quad (4.15)$$

θ er en diagonal matrise, som vil si at komponent c_{ij} kan reduseres slik:

$$c_{ij} = a_{ij} \times b_{jj}. \quad (4.16)$$

Vi kan nå regne ut $A\theta$

$$\begin{aligned} A\theta &= \begin{pmatrix} a_{00} & \dots & a_{0(d-1)} \\ \vdots & \ddots & \vdots \\ a_{(d-1)0} & \dots & a_{(d-1)(d-1)} \end{pmatrix} \begin{pmatrix} \theta_0 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & \theta_{d-1} \end{pmatrix} \\ &= \begin{pmatrix} a_{00}\theta_0 & a_{01}\theta_1 & \dots & a_{0(d-1)}\theta_{(d-1)} \\ \vdots & \vdots & \ddots & \vdots \\ a_{(d-1)0}\theta_0 & a_{(d-1)1}\theta_1 & \dots & a_{(d-1)(d-1)}\theta_{(d-1)} \end{pmatrix}. \end{aligned}$$

$A\theta$ er en $d \times d$ matrise, som vi videre skal multiplisere med $(m_t - X_t)$, som er en $d \times 1$ vektor. En $d \times d$ matrise med en $d \times 1$ vektor gir oss en $d \times 1$ vektor. Komponent c_i i den nye vektoren kan skrives på følgende måte

$$c_i = a_{i1} \times x_1 + a_{i2} \times x_2 + \dots + a_{in} \times x_n. \quad (4.17)$$

Vi kan nå finne et uttrykk for $u_t(\theta) = A\theta(m_t - X_t)$.

$$\begin{aligned}
& A\theta(m_t - X_t) \\
&= \begin{pmatrix} a_{00}\theta_0 & \dots & a_{0(d-1)}\theta_{(d-1)} \\ \vdots & \ddots & \vdots \\ a_{(d-1)0}\theta_0 & \dots & a_{(d-1)(d-1)}\theta_{(d-1)} \end{pmatrix} \begin{pmatrix} m_t^{(0)} - X_t^{(0)} \\ \vdots \\ m_t^{(d-1)} - X_t^{(d-1)} \end{pmatrix} \\
&= \begin{pmatrix} a_{00}\theta_0(m_t^{(0)} - X_t^{(0)}) + \dots + a_{0(d-1)}\theta_{(d-1)}(m_t^{(d-1)} - X_t^{(d-1)}) \\ \vdots \\ a_{(d-1)0}\theta_0(m_t^{(0)} - X_t^{(0)}) + \dots + a_{(d-1)(d-1)}\theta_{(d-1)}(m_t^{(d-1)} - X_t^{(d-1)}) \end{pmatrix}.
\end{aligned}$$

Vi ser et mønster her og det er fargekodet for å se hvilke komponentverdier som varierer likt. Vi kan nå lage et uttrykk for hver komponent j av vektoren u

$$u_t^{(j)} = \sum_{i=0}^{d-1} a_{ji}\theta_i(m_t^{(i)} - X_t^{(i)}). \quad (4.18)$$

Det vil si at vi kan skrive

$$u_t^T(\theta) = \left(\sum_{i=0}^{(d-1)} a_{0i}\theta_i(m_t^{(i)} - X_t^{(i)}), \dots, \sum_{i=0}^{(d-1)} a_{(d-1)i}\theta_i(m_t^{(i)} - X_t^{(i)}) \right). \quad (4.19)$$

Videre multipliserer vi dette uttrykket med matrisen A . Resultatet er en vektor med samme dimensjon som $u_t^T(\theta)$

$$u_t^T(\theta)A = \left(\sum_{j,i=0}^{(d-1)} a_{ji}\theta_i(m_t^{(i)} - X_t^{(i)})a_{j0}, \dots, \sum_{j,i=0}^{(d-1)} a_{ji}\theta_i(m_t^{(i)} - X_t^{(i)})a_{j(d-1)} \right).$$

Dette skal igjen multipliseres med vektoren dX_t . Dette vil resultere i en skalar.

$$\begin{aligned}
u_t^T(\theta)AdX_t &= \sum_{j,i=0}^{(d-1)} a_{ji}\theta_i(m_t^{(i)} - X_t^{(i)})a_{j0}dX_t^{(0)} \\
&\quad + \dots + \sum_{j,i=0}^{(d-1)} a_{ji}\theta_i(m_t^{(i)} - X_t^{(i)})a_{j(d-1)}dX_t^{(d-1)} \\
&= \sum_{k,j,i=0}^{d-1} a_{ji}\theta_i(m_t^{(i)} - X_t^{(i)})a_{jk}dX_t^{(k)}
\end{aligned}$$

Vi har funnet et uttrykk for første leddet av loglikelihoodfunksjonen. Nå gjenstår det å finne et uttrykk for $u_t^T(\theta)u_t(\theta)$, som er prikkproduktet av to vektorer i \mathbb{R}^d . Dette vil gi oss en skalar.

$$\begin{aligned}
& u_t^T(\theta)u_t(\theta) \\
&= \sum_{i=0}^{(d-1)} a_{0i}\theta_i(m_t^{(i)} - X_t^{(i)}) \sum_{i=0}^{(d-1)} a_{0i}\theta_i(m_t^{(i)} - X_t^{(i)}) \\
&\quad + \dots + \sum_{i=0}^{(d-1)} a_{(d-1)i}\theta_i(m_t^{(i)} - X_t^{(i)}) \sum_{i=0}^{(d-1)} a_{(d-1)i}\theta_i(m_t^{(i)} - X_t^{(i)}) \\
&= \left(\sum_{i=0}^{(d-1)} a_{0i}\theta_i(m_t^{(i)} - X_t^{(i)}) \right)^2 + \dots + \left(\sum_{i=0}^{(d-1)} a_{(d-1)i}\theta_i(m_t^{(i)} - X_t^{(i)}) \right)^2 \\
&= \sum_{i,j=0}^{(d-1)} a_{0i}\theta_i(m_t^{(i)} - X_t^{(i)})a_{0j}\theta_j(m_t^{(j)} - X_t^{(j)}) \\
&\quad + \dots + \sum_{i,j=0}^{(d-1)} a_{(d-1)i}\theta_i(m_t^{(i)} - X_t^{(i)})a_{(d-1)j}\theta_j(m_t^{(j)} - X_t^{(j)}) \\
&= \sum_{k,i,j=0}^{(d-1)} a_{ki}\theta_i(m_t^{(i)} - X_t^{(i)})a_{kj}\theta_j(m_t^{(j)} - X_t^{(j)})
\end{aligned}$$

Nå har vi et uttrykk for begge leddene i $\ell(\theta)$ som vi kan bruke i funksjonen

$$\begin{aligned}
\ell(\theta) &= \int_0^t \sum_{k,j,i=0}^{d-1} a_{ji}\theta_i(m_s^{(i)} - X_s^{(i)})a_{jk}dX_s^{(k)} \\
&\quad - \frac{1}{2} \int_0^t \sum_{k,j,i=0}^{d-1} a_{ki}\theta_i(m_s^{(i)} - X_s^{(i)})a_{kj}\theta_j(m_s^{(j)} - X_s^{(j)})ds.
\end{aligned} \tag{4.20}$$

Nå ønsker vi å finne $\frac{\partial \ell(\theta)}{\partial \theta_l}$ for hver $l \in \{0, 1, 2, \dots, d-1\}$ og sette uttrykket lik 0 for å estimere hver komponent l av θ . Siden vi deriverer med hensyn på θ_l , vil alle leddene som ikke inneholder θ_l bli lik null når vi deriverer. Den deriverte av θ_l er lik 1, så leddene som inneholder θ_l forblir det samme, men med $\theta_l = 1$.

For hver komponent θ_i har vi følgende:

$$\theta_i = \begin{cases} 1, & \text{hvis } i = j \\ 0, & \text{hvis } i \neq j \end{cases} \quad (4.21)$$

Dette betegnes som Kronecker-Delta, $\delta_{i=l}$ og $\delta_{i \neq l}$. Vi deler opp uttrykket for å få bedre oversikt:

1.

$$\begin{aligned} & \frac{\partial}{\partial \theta_l} \sum_{k,j,i=0}^{d-1} a_{ji} \theta_i (m_s^{(i)} - X_s^{(i)}) a_{jk} dX_s^{(k)} \\ &= \frac{\partial}{\partial \theta_l} \sum_{k,j,i=0}^{d-1} \underbrace{a_{ji} (m_s^{(i)} - X_s^{(i)}) a_{jk} dX_s^{(k)}}_{c_{kji}} \theta_i \\ &= \frac{\partial}{\partial \theta_l} \sum_{k,j,i=0}^{d-1} c_{kji} \theta_i \\ &= \sum_{k,j=0}^{d-1} c_{kjl} \delta_{i=l} + \sum_{k,j=0}^{d-1} c_{kjl} \delta_{i \neq l} \\ &= \sum_{k,j=0}^{d-1} c_{kjl} \\ &= \sum_{k,j=0}^{d-1} a_{jl} (m_s^{(l)} - X_s^{(l)}) a_{jk} dX_s^{(k)} \end{aligned}$$

2.

$$\begin{aligned}
& \frac{\partial}{\partial \theta_l} \sum_{k,j,i=0}^{d-1} a_{ki} \theta_i (m_s^{(i)} - X_s^{(i)}) a_{kj} \theta_j (m_s^{(j)} - X_s^{(j)}) \\
&= \frac{\partial}{\partial \theta_l} \sum_{k,j,i=0}^{d-1} \underbrace{a_{ki} (m_s^{(i)} - X_s^{(i)}) a_{kj} (m_s^{(j)} - X_s^{(j)})}_{c_{kji}} \theta_i \theta_j \\
&= \sum_{k,j,i=0}^{d-1} c_{kji} \frac{\partial}{\partial \theta_l} \theta_i \theta_j \\
&= \sum_{k,j,i=0}^{d-1} c_{kji} (\delta_{i=l} \theta_j + \delta_{k=l} \theta_i) \\
&= \sum_{k,j=0}^{d-1} c_{kji} (\theta_j + \theta_i) \\
&= \sum_{k,j,i=0}^{d-1} c_{kji} \theta_j + \sum_{k,j,i=0}^{d-1} c_{kji} \theta_i \\
&= 2 \sum_{k,j=0}^{d-1} c_{kji} \theta_j \quad (\text{På grunn av symmetri}) \\
&= 2 \sum_{k,j=0}^{d-1} a_{kl} (m_s^{(l)} - X_s^{(l)}) a_{kj} (m_s^{(j)} - X_s^{(j)}) \theta_j
\end{aligned}$$

Vi kombinerer begge uttrykkene og får at den deriverte av loglikelihood-funksjonen med hensyn på θ_l er lik

$$\begin{aligned}
\frac{\partial \ell(\theta)}{\partial \theta_l} &= \int_t^0 \sum_{k,j=0}^{d-1} a_{jl} (m_s^{(l)} - X_s^{(l)}) a_{jk} dX_s^{(k)} \\
&\quad - \int_0^t \sum_{k,j=0}^{d-1} a_{kl} (m_s^{(l)} - X_s^{(l)}) a_{kj} (m_s^{(j)} - X_s^{(j)}) \theta_j ds.
\end{aligned} \tag{4.22}$$

Vi setter dette uttrykket lik 0 og vi får at

$$\begin{aligned} & \int_0^t \sum_{k,j=0}^{d-1} a_{jl}(m_s^{(l)} - X_s^{(l)})a_{jk}dX_s^{(k)} \\ &= \int_0^t \sum_{k,j=0}^{d-1} a_{kl}(m_s^{(l)} - X_s^{(l)})a_{kj}(m_s^{(j)} - X_s^{(j)})\theta_j ds. \end{aligned} \quad (4.23)$$

Vi legger merke til at vi deriverte med hensyn på θ_l , og det ga oss et uttrykk for θ_j . Vi gjør dette for alle komponentene $l \in \{0, 1, \dots, d-1\}$, og får et likningssett med d likninger og d ukjente. Dette likningssettet kan vi skrive på formen $\theta = M^{-1}b$ hvor θ er en vektor med alle løsningene av θ_l .

$$\theta = (\theta_0, \theta_1, \dots, \theta_{d-1})$$

$$b = (b_0, b_1, \dots, b_{d-1})$$

Vi ser på uttrykket vårt (4.23) og ser at vi kan skrive det på denne formen

$$\underbrace{\theta}_{d \times 1} = \underbrace{M^{-1}}_{d \times d} \underbrace{b}_{d \times 1}$$

med

$$\begin{aligned} b_l &= \int_0^t \sum_{k,j=0}^{d-1} a_{jl}(m_s^{(l)} - X_s^{(l)})a_{jk}dX_s^{(k)} \\ &\approx \sum_{i=0}^n \sum_{k,j=0}^{d-1} a_{jl}(m_s^{(l)} - X_s^{(l)})a_{jk}dX_s^{(k)}(X_{t_{i+1}}^{(k)} - X_{t_i}^{(k)}) \end{aligned} \quad (4.24)$$

og

$$\begin{aligned} M_{lj} &= \int_0^t \sum_{k,j=0}^{d-1} a_{kl}(m_s^{(l)} - X_s^{(l)})a_{kj}(m_s^{(j)} - X_s^{(j)})\theta_j ds \\ &\approx \sum_{i=0}^n \sum_{k,j=0}^{d-1} a_{kl}(m_s^{(l)} - X_s^{(l)})a_{kj}(m_s^{(j)} - X_s^{(j)})\theta_j(t_{i+1} - t_i). \end{aligned} \quad (4.25)$$

```

1 import numpy as np
2 import pandas as pd
3 import sigma_multidimensional
4 import white_noise
5
6 def est_theta(d, n, T, sigma, x, m):
7     t = np.linspace(0, T, n)
8
9     A = np.linalg.inv(sigma)
10    b = np.zeros((d))
11    M = np.zeros((d, d))
12
13    for l in range(0, d):
14        for i in range(0, len(x[0]) - 1):
15            for k in range(0, d):
16                for j in range(0, d):
17                    b[l] = b[l] + A[j,l] \
18                        * (m[l,i] - x[l,i]) * A[j,k] \
19                        * (x[k,i+1] - x[k,i])
20                    M[l,j] = M[l,j] + A[k,l] \
21                        * (m[l,i] - x[l,i]) \
22                        * A[k,j] * (m[j,i] - x[j,i]) \
23                        * (t[i+1] - t[i])
24
25    estimated_theta = (np.linalg.inv(M)) @ b
26    estimated_theta = np.diag(estimated_theta)
27    return estimated_theta

```

Dette skriptet definerer et intervall med n steg fra 0 til T . Vi har definert θ til å være lik $M^{-1}b$, så vi må først regne ut M og b for å finne θ . Både M og b er definert med komponenter av σ^{-1} , så vi definerer $A = \sigma^{-1}$.

Videre definerer vi b til å være en nullvektor og M til å være en nullmatrise. Deretter utføres en serie med beregninger for hver komponent i b og i M . θ beregnes til slutt ved å løse likningssystemet, og resultatet returneres som en diagonal matrise.

Vi vil igjen teste ut skriptet på en stokastisk prosess før vi bruker det på faktiske data for strømpriser. Vi simulerer samme prosessen som vi skal bruke for strømpriser, altså $dX_t = \theta(m_t - X_t)dt + \sigma dB_t$ (1.1), men med vilkårlige verdier for σ og θ , og en vilkårlig funksjon m_t .


```

1 import numpy
2 import white_noise
3
4 def prosess(d, n, T, sigma, theta, m):
5     t = np.linspace(0, T, n)
6     dt = t[1] - t[0]
7
8     WN = white_noise.simulate_WN(d, n, T)
9
10    x = np.zeros((d, n))
11    x[:, 0] = m[:, 0]
12
13    for i in range(0, n - 1):
14        x[:, i + 1] = x[:, i] + theta @ (m[:, i] - x[:, i]) *
            dt + sigma @ WN[:, i]

```

Vi simulerer en 3-dimensjonell prosess med $T = 10$ og $n = 10000$. Vi bruker samme σ som vi testa med tidligere, altså $\sigma = \begin{pmatrix} 2 & 3 & 4 \\ 3 & 5 & 6 \\ 4 & 6 & 10 \end{pmatrix}$, og vi

setter $\theta = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 3 \end{pmatrix}$. Siden dette er en prosess vi simulerer selv, så er ikke valget av m så viktig. m representerer driftsleddet som prosessen reverterer til og vil ikke være avgjørende for å få et godt estimat. Vi velger derfor å la komponentene til m være konstante for enkelthetens skyld. Vi bruker

$$m = \underbrace{\begin{pmatrix} 3 & 3 & \dots & 3 & 3 \\ 2 & 2 & \dots & 2 & 2 \\ 8 & 8 & \dots & 8 & 8 \end{pmatrix}}_{n \text{ ganger}}.$$

Vi setter inn alle disse verdiene i skriptet og får følgende verdi

$$\hat{\theta} = \begin{pmatrix} 0.99777287 & 0. & 0. \\ 0. & 2.00033758 & 0. \\ 0. & 0. & 3.00344406 \end{pmatrix}.$$

Differansen vi får mellom θ og $\hat{\theta}$ er lik 0.004115287246371323. Dette er ganske lovende, så vi kan stole på at dette skriptet gir oss et godt estimat for θ .

Nå som vi har testet ut skriptet for en stokastisk prosess vi selv har konstruert, kan vi bruke dataene vi har for strømpriser. Vi definerer x på samme måte som da vi estimerte σ , hvor hver av kolonnene med data tilsvarer en komponent i vektoren x .

Vi viser til dataene for gjennomsnittlig spotpris for hver dag i året:

	Dag	Måned	Spot
0	1	1	3.537510
1	2	1	3.612353
2	3	1	3.584388
3	4	1	3.616396
4	5	1	3.631566
...
360	27	12	3.541774
361	28	12	3.584527
362	29	12	3.558917
363	30	12	3.562377
364	31	12	3.543819

[365 rows x 3 columns]

Vi slår sammen disse dataene med dataene for strømpriser, slik at vi har en kolonne som viser hva spotprisen faktisk var på en bestemt dato, og hva gjennomsnittet for den datoen er. Dette er ikke nødvendig å gjøre med kontraktene, da gjennomsnittet for kontraktene er konstant.

	Dato	1-årskontrakt	2-årskontrakt	3-årskontrakt \
0	2007-01-02	3.695110	3.737670	3.747148
1	2007-01-03	3.686376	3.725693	3.742420
2	2007-01-04	3.686376	3.719651	3.737670
3	2007-01-05	3.671225	3.702536	3.722798
4	2007-01-08	3.634951	3.660994	3.688879
...
3253	2019-12-19	3.593194	3.526655	3.498324
3254	2019-12-20	3.590439	3.530763	3.502851
3255	2019-12-23	3.579344	3.527830	3.501646
3256	2019-12-27	3.557061	3.523415	3.500439
3257	2019-12-30	3.542697	3.501043	3.494991

	Spot	Spot (gjennomsnitt)
0	3.478390	3.612353
1	3.475931	3.584388
2	3.421640	3.616396
3	3.402017	3.631566
4	3.343805	3.614912
...
3253	3.624085	3.637583
3254	3.607637	3.585597
3255	3.607659	3.537906
3256	3.635128	3.541774
3257	3.519771	3.562377

[3258 rows x 6 columns]

Den første komponenten av vektoren m_t , som er driftsleddet, vil være kolonnen med gjennomsnittlig spotpris. Resten av komponentene til m_t vil inneholde gjennomsnittlige kontraktspriser, og denne vil holdes konstant. m_t vil altså se slik ut:

$$m_t = \underbrace{\begin{pmatrix} 3.61235304 & 3.58438784 & \dots & 3.54177433 & 3.56237745 \\ 3.56846929 & 3.56846929 & \dots & 3.56846929 & 3.56846929 \\ 3.53801126 & 3.53801126 & \dots & 3.53801126 & 3.53801126 \\ 3.53334473 & 3.53334473 & \dots & 3.53334473 & 3.53334473 \end{pmatrix}}_{3258 \text{ ganger}}.$$

Vi bruker disse verdiene for x og m , verdien vi fikk for $\hat{\sigma}$ fra dataene, $d = 4$ og n og T lik lengden av dataene. Dette gir oss følgende verdi for $\hat{\theta}$:

$$\hat{\theta} = \begin{pmatrix} 0.03824143 & 0. & 0. & 0. \\ 0. & 0.0037009 & 0. & 0. \\ 0. & 0. & 0.00263384 & 0. \\ 0. & 0. & 0. & 0.00202072 \end{pmatrix}$$

4.2.3 Avhengig modell for spotpriser og kontraktene

```
1 import numpy as np
2 import pandas as pd
3 import matplotlib.pyplot as plt
4 import scipy.linalg
5 import white_noise
6 import sigma_multidimensional
7 import theta_multidimensional
8
9 def model(n, T, sigma, theta, m):
10     t = np.linspace(0, T, n)
11     dt = t[1] - t[0]
12
13     WN = white_noise.simulate_WN(4, n, T)
14
15     x = np.zeros((4, T))
16     x[:, 0] = m[:, 0]
17
18     for i in range(0, len(x[0]) - 1):
19         x[:, i+1] = x[:, i] + theta @ (m[:, i] - x[:, i]) \
20             * dt + sigma @ WN[:, i]
21
22     return x
```

Dette er modellen vår for strømpriser. Funksjonen tar inn fem argumenter, $\sigma \in \mathbb{R}^{4 \times 4}$, $\theta \in \mathbb{R}^{4 \times 4}$, $m \in \mathbb{R}^4$, tidssteg n og tiden T . I dette skriptet vil funksjonen som simulerer hvitt støy importeres, og sammen med parametrene σ og θ , og vektoren m , konstrueres den stokastiske prosessen

$$X_t = \theta(m_t - X_t)dt + \sigma B_t \quad (1.1)$$

som skal beskrive strømprisene. Vi bruker

$$\hat{\sigma} = \begin{pmatrix} 0.11223847 & -0.00015466 & -0.0002181 & -0.00022423 \\ -0.00015466 & 0.01409231 & 0.00646873 & 0.00475871 \\ -0.0002181 & 0.00646873 & 0.00946879 & 0.00545703 \\ -0.00022423 & 0.00475871 & 0.00545703 & 0.007893 \end{pmatrix}$$

og

$$\hat{\theta} = \begin{pmatrix} 0.03824143 & 0. & 0. & 0. \\ 0. & 0.0037009 & 0. & 0. \\ 0. & 0. & 0.00263384 & 0. \\ 0. & 0. & 0. & 0.00202072 \end{pmatrix}$$

som parametere i modellen, og for driftsleddet m bruker vi

$$m_t = \underbrace{\begin{pmatrix} 3.61235304 & 3.58438784 & \dots & 3.54177433 & 3.56237745 \\ 3.56846929 & 3.56846929 & \dots & 3.56846929 & 3.56846929 \\ 3.53801126 & 3.53801126 & \dots & 3.53801126 & 3.53801126 \\ 3.53334473 & 3.53334473 & \dots & 3.53334473 & 3.53334473 \end{pmatrix}}_{358 \text{ ganger}}.$$

Vi lager en modell som skal estimere strømprisene i løpet av ett år med 358 handledager. Vi setter n og T til å være lengden av dataene og vi har modellen vår klar. Vi får følgende plott for estimerte strømpriser:

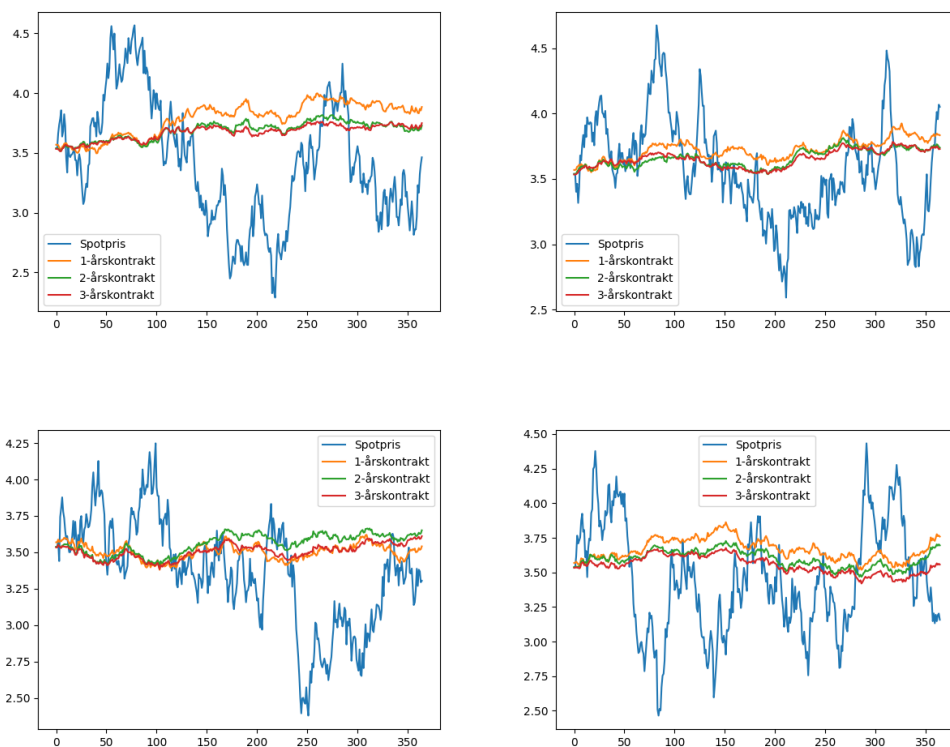




Figure 20: Estimeringer av strømprisene.

4.2.4 Testing av den 4-dimensjonelle modellen

Vi har utviklet en modell for strømpriser ved å beskrive disse prisene som en stokastisk differensiallikning. Parameterne i likningen estimerte vi ved å bruke historisk data for strømpriser. Nå ønsker vi å teste hvor nøyaktig parameterne er estimert ved å bruke modellen til å generere syntetisk data for strømpriser. Vi vil deretter estimere de samme parameterne fra de syntetiske dataene slik at vi kan sammenligne de med parameterne fra de faktiske dataene. Hvis parameterne fra de syntetiske dataene er sammenlignbare med parameterne fra de faktiske dataene, kan vi anta at modellen vår for strømpriser er akseptabel og at den kan brukes til å generere prognoser på lignende data i fremtiden. Nedenfor ser vi plottet for faktiske strømpriser gjennom 13 år.

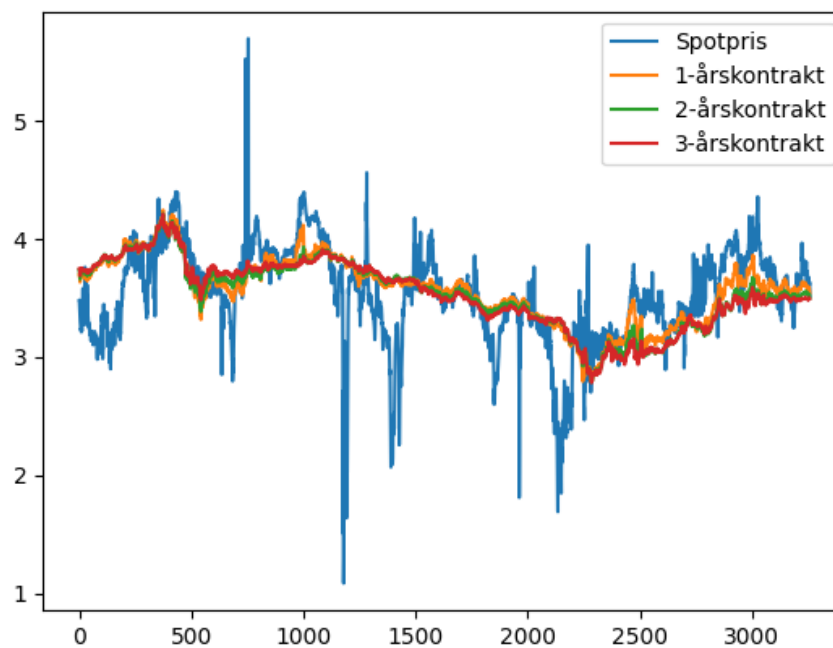
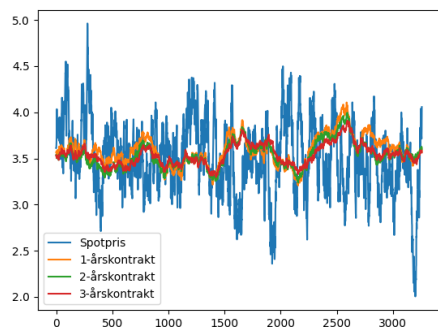
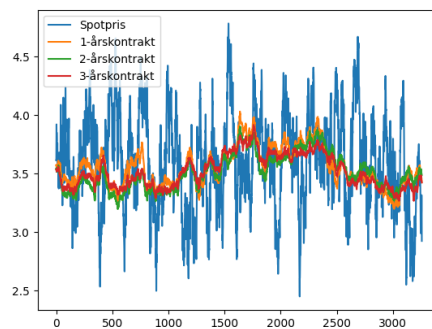


Figure 21: Faktiske strømpriser målt fra 2007 til 2019.

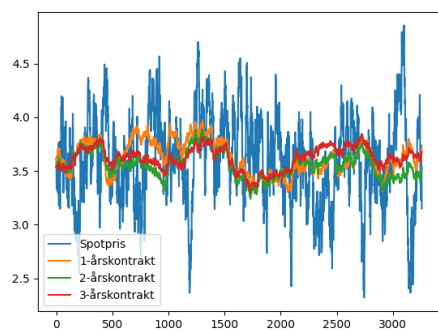
Vi generer følgende fire modeller:



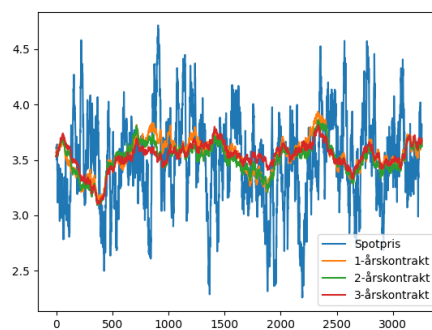
(a)



(b)



(c)



(d)

Figure 22: Estimerte strømpriser for 13 år.

og får følgende syntetisk data:

	Spot	1-årskontrakt	2-årskontrakt	3-årskontrakt
0	3.612353	3.568469	3.538011	3.533345
1	3.612353	3.568469	3.538011	3.533345
2	3.656857	3.568241	3.536745	3.533483
3	3.682752	3.556695	3.523782	3.520904
(a) 4	3.795359	3.542007	3.503527	3.506074
...
3253	3.757752	3.602463	3.607691	3.570884
3254	3.734844	3.605470	3.625253	3.583097
3255	3.713915	3.616142	3.628099	3.588068
3256	3.809338	3.595534	3.621377	3.584214
3257	4.057767	3.570366	3.599928	3.564456

[3258 rows x 4 columns]

	Spot	1-årskontrakt	2-årskontrakt	3-årskontrakt
0	3.612353	3.568469	3.538011	3.533345
1	3.612353	3.568469	3.538011	3.533345
2	3.729474	3.575143	3.546891	3.538088
3	3.919807	3.552288	3.524740	3.521400
(b) 4	3.775280	3.544154	3.514349	3.516051
...
3253	3.126762	3.440352	3.486345	3.432590
3254	3.183227	3.461297	3.503213	3.441228
3255	3.314943	3.455008	3.496249	3.438986
3256	3.194518	3.429743	3.491838	3.431558
3257	2.923195	3.423460	3.487346	3.422412

[3258 rows x 4 columns]

	Spot	1-årskontrakt	2-årskontrakt	3-årskontrakt
0	3.612353	3.568469	3.538011	3.533345
1	3.612353	3.568469	3.538011	3.533345
2	3.592305	3.580575	3.540338	3.531400
3	3.394434	3.607843	3.555565	3.546721
4	3.332740	3.612079	3.549283	3.548249
...
3253	3.437254	3.670877	3.508455	3.660511
3254	3.516471	3.698586	3.520735	3.665578
3255	3.524621	3.687920	3.509302	3.662305
3256	3.256969	3.691897	3.511966	3.662407
3257	3.149710	3.718096	3.532241	3.678212

(c) [3258 rows x 4 columns]

	Spot	1-årskontrakt	2-årskontrakt	3-årskontrakt
0	3.612353	3.568469	3.538011	3.533345
1	3.612353	3.568469	3.538011	3.533345
2	3.593159	3.543601	3.525227	3.526555
3	3.537660	3.535942	3.525886	3.535924
4	3.525374	3.562186	3.542722	3.549645
...
3253	3.760533	3.653823	3.645549	3.675280
3254	3.743368	3.648778	3.639781	3.677994
3255	3.824515	3.677507	3.655631	3.695246
3256	3.636847	3.666733	3.646510	3.687774
3257	3.622714	3.640587	3.620526	3.669196

(d) [3258 rows x 4 columns]

Disse syntetiske dataene gir oss følgende estimater for σ og θ :

(a)

$$\hat{\sigma} = \begin{pmatrix} 0.11294927 & -0.00039905 & -0.00034293 & -0.00040317 \\ -0.00039905 & 0.01410025 & 0.00642739 & 0.00482133 \\ -0.00034293 & 0.00642739 & 0.00948216 & 0.00542122 \\ -0.00040317 & 0.00482133 & 0.00542122 & 0.00785978 \end{pmatrix}$$

$$\hat{\theta} = \begin{pmatrix} 0.04172137 & 0. & 0. & 0. \\ 0. & 0.00699743 & 0. & 0. \\ 0. & 0. & 0.00582531 & 0. \\ 0. & 0. & 0. & 0.00427146 \end{pmatrix}$$

(b)

$$\hat{\sigma} = \begin{pmatrix} 0.11399462 & -0.00050094 & -0.00043417 & -0.00037164 \\ -0.00050094 & 0.01414503 & 0.00658842 & 0.00487355 \\ -0.00043417 & 0.00658842 & 0.00956808 & 0.00554826 \\ -0.00037164 & 0.00487355 & 0.00554826 & 0.00794291 \end{pmatrix}$$

$$\hat{\theta} = \begin{pmatrix} 0.04164213 & 0. & 0. & 0. \\ 0. & 0.00467398 & 0. & 0. \\ 0. & 0. & 0.00389992 & 0. \\ 0. & 0. & 0. & 0.00308568 \end{pmatrix}$$

(c)

$$\hat{\sigma} = \begin{pmatrix} 0.11328317 & -0.00027369 & -0.00029117 & -0.00034539 \\ -0.00027369 & 0.01401326 & 0.00633643 & 0.00462024 \\ -0.00029117 & 0.00633643 & 0.00947534 & 0.00545678 \\ -0.00034539 & 0.00462024 & 0.00545678 & 0.00789218 \end{pmatrix}$$

$$\hat{\theta} = \begin{pmatrix} 0.0354454 & 0. & 0. & 0. \\ 0. & 0.00244894 & 0. & 0. \\ 0. & 0. & 0.00250677 & 0. \\ 0. & 0. & 0. & 0.00129065 \end{pmatrix}$$

(d)

$$\hat{\sigma} = \begin{pmatrix} 1.130273e^{-01} & 3.214479e^{-04} & 1.553643e^{-04} & 8.615375e^{-05} \\ 3.214479e^{-04} & 1.394626e^{-02} & 6.396571e^{-03} & 4.613221e^{-03} \\ 1.553643e^{-04} & 6.396571e^{-03} & 9.537789e^{-03} & 5.476320e^{-03} \\ 8.615375e^{-05} & 4.613221e^{-03} & 5.476320e^{-03} & 7.859814e^{-03} \end{pmatrix}$$
$$\hat{\theta} = \begin{pmatrix} 0.03833443 & 0. & 0. & 0. \\ 0. & 0.00565484 & 0. & 0. \\ 0. & 0. & 0.00436478 & 0. \\ 0. & 0. & 0. & 0.00315719 \end{pmatrix}$$

Dette gir oss at differansen mellom parameterne fra faktiske data og fra syntetiske data er lik

(a) $\hat{\sigma} : 0.000857357762544865$, $\hat{\theta} : 0.006190777486006116$

(b) $\hat{\sigma} : 0.001883487680713856$, $\hat{\theta} : 0.003915903886161882$

(c) $\hat{\sigma} : 0.0011132941561777438$, $\hat{\theta} : 0.0031408422789097933$

(d) $\hat{\sigma} : 0.0012759230495338306$, $\hat{\theta} : 0.0028504819280665995$

Vi ser at differansene angis på minst to desimaler. Dette gjør at vi kan konkludere med at parameterne fra de syntetiske dataene er sammenlignbare med parameterne fra de faktiske dataene, noe som indikerer at modellen vår for strømpriser er lovende.

4.2.5 Monte Carlo-simulering

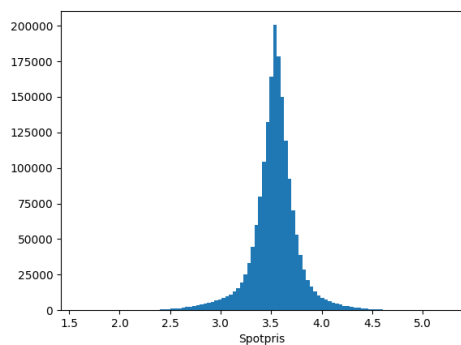
Vi har utviklet en modell som skal estimere strømpriser basert på historisk data. Som en del av modelleringen har vi også analysert hvor godt modellen estimerer parametere til de faktiske dataene ved å bruke modellen til å utvikle syntetisk data. For å analysere modellen videre, vil vi undersøke fordelingen av de syntetiske dataene.

En måte å gjøre dette på er ved å bruke Monte Carlo-simulering. En typisk Monte Carlo-simulering innebærer å beregne modellen for eksempel 1000 ganger slik at vi får 1000 forskjellige dataserier for estimerte strømpriser. Når simuleringen er fullført, vil vi ha et stort antall av resultater som vi har fått fra modellen. Disse resultatene vil gi oss informasjon om fordelingen av de estimerte strømprisene (Thete, 2022).

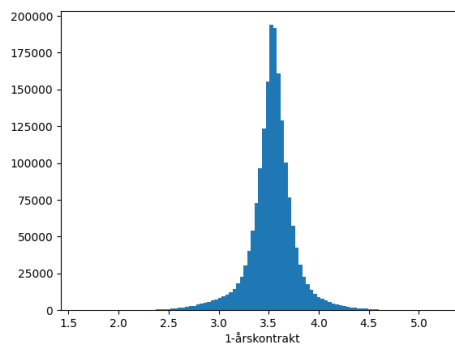
Vi legger til disse linjene i skriptet hvor modellen som estimerer strømprisene blir konstruert:

```
1 results = []
2 #Kjører modellen 5000 ganger og lagrer resultatene
3 #i "results".
4 for i in range(5000):
5     x = model(len(m[0]), sigma, theta, m)
6     results.append(x)
7 #Omformer "results" til en array for å enklere håndtere
8     dataene.
9 results = np.array(results)
10 results = results.reshape(-1, 4)
11
12 fig_list = []
13 #Lager fire histogrammer. Ett for hver kolonne i "results"
14 #som da tilsvarer hver av strømprisene.
15 for i in range(4):
16     fig, ax = plt.subplots()
17     ax.hist(results[:, i], bins=100)
18     fig_list.append(fig)
19 plt.show()
20
21 #Beregner og printer gjennomsnittet og 95%-konfidensintervall
22 #for alle prisene.
23 for i in range(0, 4):
24     print(np.mean(results[i]))
25     print(stats.t.interval(confidence=0.95,
26                             df=len(results[i]) - 1,
27                             loc=np.mean(results[i]),
28                             scale=stats.sem(results[i])))
```

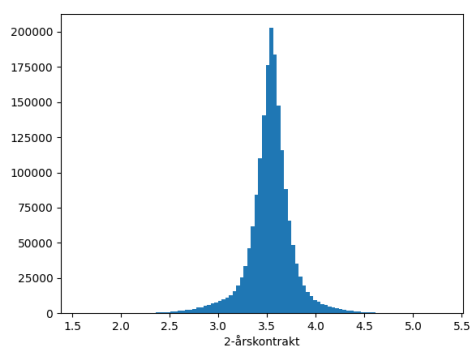
Her kjører vi modellen 5000 ganger for å få fordelingene av dataene. Til slutt printes gjennomsnitt og konfidensintervall ut for alle strømprisene. Skriptet gir oss følgende fordelinger av data:



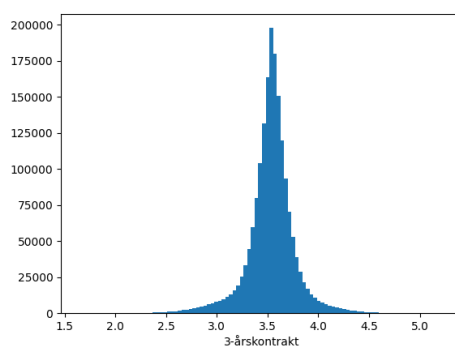
(a) Spotpris.



(b) 1-årskontrakt.



(c) 2-årskontrakt.



(d) 3-årskontrakt

Figure 23: Fordelinger av de syntetiske dataene for strømpriser.

Nedenfor ser vi gjennomsnittet og 95%-konfidensintervall for alle prisene.

- Spotpris:
 - Gjennomsnitt: 3.5202279873969458
 - Konfidensintervall: (3.3988199705433644, 3.641636004250527)
- 1-årskontrakt:
 - Gjennomsnitt: 3.564942865545931
 - Konfidensintervall: (3.501955343080609, 3.6279303880112534)
- 2-årskontrakt:
 - Gjennomsnitt: 3.5761367237080397
 - Konfidensintervall: (3.434564732950826, 3.717708714465253)
- 3-årskontrakt:
 - Gjennomsnitt: 3.571683100254339
 - Konfidensintervall: (3.457371670848761, 3.6859945296599173)

Vi kan være 95% sikre på at strømprisene vil holde seg innenfor disse intervallene. Vi beregner gjennomsnittet av de faktiske strømprisene

- Spotpriser: $\mu = 3.525087784370532$,
- 1-årskontrakter: $\mu = 3.568469293376401$,
- 2-årskontrakter: $\mu = 3.5380112638971624$,
- 3-årskontrakter: $\mu = 3.5333447299835687$,

og ser at gjennomsnittsverdiene for faktiske data og for syntetiske data er nokså like. Dette kan tyde på at modellen vår har klart å fange opp strømprisenes bilde på en nøyaktig måte.

5 Oppsummering

I denne oppgaven har vi undersøkt de matematiske egenskapene til strømpriser og utviklet en modell for å estimere disse prisene. Vi har sett at strømpriser kan beskrives som en stokastisk prosess og at Itô kalkulus er et viktig verktøy for å analysere slike stokastiske prosesser.

En stokastisk prosess beskriver utviklingen av en tilfeldig variabel, som for oss er strømprisene, over tid. Stokastiske differensiallikninger er en matematisk beskrivelse av endringen til denne tilfeldige variabelen over tid og vi har klart å presentere en 4-dimensjonell modell for strømpriser i form av en slik likning. Ved å bruke den stokastiske prosessen

$$dX_t = \theta(m_t - X_t)dt + \sigma dB_t \quad (1.1)$$

hvor $X_t, m_t, B_t \in \mathbb{R}^d$ og $\theta, \sigma \in \mathbb{R}^{d \times d}$, har vi klart å utvikle en modell som tar hensyn til både spotpriser, 1-årskontrakter, 2-årskontrakter og 3-årskontrakter. Vi har brukt data for faktiske strømpriser for å estimere de nødvendige parameterne i denne prosessen slik at de passer best mulig til de observerte dataene.

Selv om det er tilnærmet umulig å gi nøyaktige estimater av fremtidige strømpriser på grunn av deres uforutsigbare natur, har vår modell gitt oss verdifull innsikt i prisenes struktur og utvikling. Våre funn indikerer at modellen kan være nyttig for å forutsi strømpriser i fremtiden og bidrar til bedre forståelse av strømpriser og matematikken rundt dette området.

Vi har også undersøkt spørsmål som hvor tilfeldige strømprisene egentlig er, hva slags prosess de kan beskrives som, hvor stor variasjonen er, og hvor raskt prisene endrer seg. Selvom strømprisene følger visse trender og mønstre, er de likevel svært uforutsigbare og de kan endre seg raskt og uventet.

Vi håper denne analysen vil være et viktig bidrag til feltet innenfor energimarkedet og at den fører til videre arbeid som vil styrke forståelsen av strømpriser. Vi har lært mye om anvendelsen av matematikk innen økonomi og energisektoren gjennom arbeidet med denne oppgaven, og vi håper at disse resultatene kan bidra til å forbedre forståelsen av strømpriser i fremtiden.

Referanser

- [1] Aanensen, T. (2021, 15. februar). *Veldig lav strømpris i 2020*. Statistisk sentralbyrå. <https://www.ssb.no/energi-og-industri/artikler-og-publikasjoner/veldig-lav-strompris-i-2020>
- [2] Fridtjof Nansens Institutt. (2022, 3. september). *Hvorfor gjør krigen i Ukraina at strømmen blir dyr i Norge?* <https://forskning.no/energi-fridtjof-nansens-institutt-krig-og-fred/hvorfor-gjor-krigen-i-ukraina-at-strommen-blir-dyr-i-norge/2072345>
- [3] Holstad, M. (2023, 14. februar). *Rekordhøy strømpris i 2022: dempet av strømstøtte*. Statistisk Sentralbyrå. <https://www.ssb.no/energi-og-industri/energi/statistikk/elektrisitetspriser/artikler/rekordhoy-strompris-i-2022--dempet-av-stromstotte>
- [4] Kessler, M., Lindner, A. & Sørensen, M. (2012). *Statistical Methods for Stochastic Differential Equations*. CRC Press.
- [5] Navidi, W. (2011). *Statistics for Engineers and Scientists* (3.utg.). McGraw-Hill.
- [6] Pavliotis, G. A. (2014). *Stochastic Processes and Applications: Diffusion Processes, the Fokker-Planck and Langevin Equations* (2.utg.). Springer.
- [7] Shreve, S. E. (2010). *Stochastic Calculus for Finance II: Continuous-Time Models*. Springer.
- [8] Thete, J. (2022, 12. februar). *Understanding Monte Carlo Simulation and its implementation with Python*. Medium. https://medium.com/@juee_thete/understanding-monte-carlo-simulation-and-its-implementation-with-python-3ecacb958cd4
- [9] Øksendal, B. (2003). *Stochastic Differential Equations: An Introduction with Applications* (6.utg.). Springer.