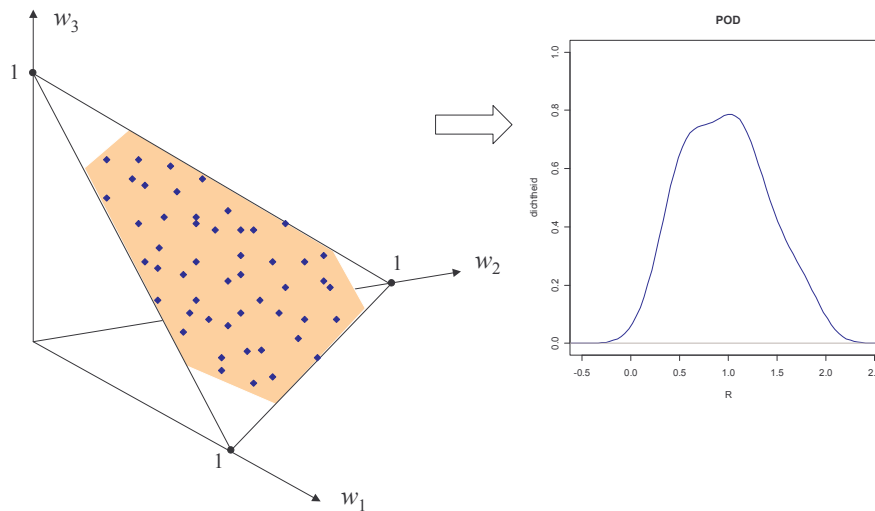


# Portfolio Opportunity Distributions

een simulatietechniek voor performance evaluatie van  
beleggingsmanagers



**Arno E. Weber**

Afstudeerscriptie Wiskunde

mei 2007



# Portfolio Opportunity Distributions

een simulatietechniek voor performance evaluatie van  
beleggingsmanagers

Arno E. Weber

Afstudeerscriptie Wiskunde

mei 2007

ORTEC bv  
Afdeling Performance and Risk Implementation  
Orly Plaza  
Orlyplein 145c  
1043 DV Amsterdam

Vrije Universiteit Amsterdam  
Faculteit der exacte wetenschappen  
Divisie Wiskunde en Informatica  
De Boelelaan 1083  
1081 HV Amsterdam

Begeleiding vanuit ORTEC:  
Begeleiding vanuit de Vrije Universiteit:

drs. J. Geenen  
dr. J.H. van Zanten  
prof. dr. A.C.M. Ran (tweede lezer)



## Voorwoord

Deze scriptie is voortgekomen uit mijn stage bij ORTEC, die ik van september 2006 tot mei 2007 heb gelopen. De stage is onderdeel van de studie Wiskunde aan de Vrije Universiteit Amsterdam en vormt tevens de afsluiting hiervan.

Tijdens de stage heb ik voor de afdeling Performance and Risk Implementation onderzoek gedaan naar een techniek die gebruikt kan worden bij performance evaluatie van beleggingsmanagers. Over deze techniek was aanvankelijk weinig bekend bij ORTEC en daarom was de doelstelling om dieper inzicht te verkrijgen in de precieze werking en de praktische bruikbaarheid van de methode. In deze scriptie zijn de beschrijving en de resultaten van het onderzoek uitgewerkt.

Mijn grote interesse ligt op het gebied van de toegepaste wiskunde. Het onderwerp van de scriptie past hier uitstekend bij. Het is een voorbeeld van hoe kennis van de klassieke wiskunde, gecombineerd met de rekenkracht van moderne computers, op een nuttige manier kan worden ingezet in het bedrijfsleven.

Omdat de wiskundige theorie veelvuldig wordt aangeropen is het noodzakelijk dat de lezer hiervan enige kennis en vaardigheden bezit, waarbij de nadruk ligt op de statistiek en de kanstheorie. Uitgebreide kennis van economische onderwerpen wordt niet verondersteld. Wel is het een uitgangspunt dat de lezer een globaal idee heeft van de werking van financiële markten en van het beleggen in financiële instrumenten.

Bij deze wil ik een aantal mensen hartelijk bedanken voor hun hulp.

- Jeroen Geenen. Hij heeft mij vanuit ORTEC begeleid en meegedacht over het probleem, de aanpak, en het schrijven van de scriptie. Daarnaast heeft hij mij geholpen bij het verkrijgen van onderzoeksdata.
- Harry van Zanten, mijn begeleider vanuit de Vrije Universiteit. Hij heeft ook meegedacht over het probleem en de aanpak. Verder heeft hij nuttige op- en aanmerkingen gegeven op de scriptie.
- André Ran, de tweede lezer van de scriptie.

Tevens wil ik een dankwoord richten aan de collega's die mij met bepaalde specifieke zaken hebben geholpen, en daarmee eveneens een belangrijke bijdrage hebben geleverd. Tot slot bedank ik mijn familie. Zij hebben mij de mogelijkheid gegeven om deze studie te doen.

Amsterdam, 18 mei 2007



## Samenvatting

Tijdens de stage heb ik mij verdiept in portfolio opportunity distributions (PODs). Dit is een techniek voor performance evaluatie van beleggingsmanagers, die gebruik maakt van computersimulatie, en momenteel nog weinig wordt toegepast. De standaardmethoden voor performance evaluatie, de benchmarks en de peer groups, hebben een aantal knelpunten die de resultaten in het algemeen onbetrouwbaar maken. PODs elimineren deze knelpunten en vormen een conceptueel betere methode, ook omdat, in tegenstelling tot de standaardmethodes, alle mandaatregels van de manager deel uitmaken van de invoergegevens.

Bij de POD techniek wordt het gerealiseerde rendement (over een bepaalde periode) statistisch vergeleken met alle rendementen die de manager had kunnen behalen. Daartoe wordt door middel van simulatie een groot aantal random portefeuilles gegenereerd die voldoen aan het mandaat. Dit stelt ons in staat om een kwalitatieve uitspraak te doen over de performance van de manager, inclusief statistische betrouwbaarheid.

Voor het genereren van de random portefeuilles zijn drie algoritmes ontwikkeld. Algoritme 1 genereert alle portefeuilles binnen een gediscretiseerde verzameling corresponderend met het mandaat. Algoritme 2 doet random trekkingen uit de volledige set portefeuilles en verworpt de trekkingen die niet voldoen aan de mandaatregels. Tenslotte doet algoritme 3 random trekkingen die direct binnen het mandaat vallen.

Voor allocatiemandaten met slechts enkele onderliggende segmenten adviseer ik om algoritme 1 of 2 te gebruiken. Voor selectiemandaten met veel stukken, maar zonder zeer restrictieve regels raad ik algoritme 2 aan. Is het mandaat wel restrictief, dan kan algoritme 3 het beste gebruikt worden, zij het dat dan mogelijk de rekentijd zeer groot is, of de geschatte POD qua vorm afwijkt van de ware POD. Wil men geen uniforme trekkingen doen maar kapitalisatie-gewogen trekkingen, dan is het advies om algoritme 3 altijd te gebruiken.

Een groot voordeel van PODs is dat deze niet onderhevig zijn aan de problemen inherent aan benchmarks en peer groups. Zo is het mogelijk om voor een zeer korte periode (bijvoorbeeld één maand) de performance te meten, zonder dat men jaren moet wachten op de beschikbaarheid van voldoende data. Een ander sterk punt is dat een POD de volledige set mandaatregels weerspiegelt, en niet alleen de markt, zoals bij benchmarks. Verder zijn PODs niet onderhevig aan survivorship bias, het bekendste knelpunt van peer groups.

Ondanks deze voordelen heeft de POD methode, zoals die in deze scriptie is uitgewerkt, ook enkele zwaktepunten. Eén daarvan is dat alleen de startportefeuille wordt gesimuleerd en géén transacties: de getrokken portefeuilles zijn buy-and-hold. De interpretatie is daarom dat het resultaat van de gekozen strategie wordt vergeleken met dat van alle “passieve” strategieën. Een uitbreiding mét random transacties is wenselijk omdat dit de methode vollediger en realistischer maakt. Het is echter onduidelijk hoe frequent deze transacties plaatsvinden en hoe groot deze dan moeten zijn. Aanvullend onderzoek over deze keuzeproblematiek acht ik noodzakelijk wanneer men de methode wil uitbreiden.





# Inhoudsopgave

1. Inleiding.....	1
1.1 Wat is performance evaluatie?.....	1
1.2 ORTEC en de afdeling Performance and Risk Implementation .....	1
1.3 De stageopdracht.....	2
2. Huidige technieken voor performance evaluatie .....	5
2.1 De rol van PE binnen het beleggingsproces .....	5
2.2 Rendement berekening .....	9
2.3 Benchmarks en peer groups.....	13
2.4 Risicometing en correctie .....	20
2.5 Performance attributie.....	27
2.6 Performance presentatie.....	31
3. Portfolio opportunity distributions .....	33
3.1 De alternatieve methode van PODs .....	33
3.2 Voordelen van PODs .....	36
3.3 Kritische kanttekeningen (en de antwoorden daarop).....	37
3.4 Verdelingsonderzoek .....	39
3.5 Statistische analyse van de performance.....	45
4. Algoritmes voor portefeuille trekkingen .....	53
4.1 Trekkingsprobleem en mandaten.....	53
4.2 Algoritme 1: Benadering van de POD via discretisatie .....	60
4.3 Algoritme 2: Een lineaire procedure voor uniforme trekkingen.....	66
4.4 Algoritme 3: Markov-keten Monte Carlo .....	73
4.5 Een vergelijking tussen de drie algoritmes .....	90
4.6 Uitbreidingen .....	92
5. Case studies .....	99
5.1 Allocatie: fixed income segment .....	99
5.2 Allocatie: equity segment .....	102
5.3 Allocatie: sectoren .....	105
5.4 Selectie: aandelen Verenigde Staten.....	107
5.5 Selectie: aandelen Telecom Services .....	110
5.6 Selectie: aandelen Japan .....	112
5.7 Simulatie met herbalancering .....	115
6. Conclusies en aanbevelingen.....	117
Literatuurlijst.....	121



# 1. Inleiding

## 1.1 Wat is performance evaluatie?

Veel instellingen in de financiële sector, waaronder pensioenfondsen en verzekeringsmaatschappijen, hanteren een beleggingsstrategie waarbij de premies, betaald door de deelnemers of cliënten, worden belegd in één of meerdere **portefeuilles** van marktinstrumenten. Dit is noodzakelijk om aan de toekomstige verplichtingen van uitbetaling te kunnen voldoen, en tegelijkertijd een buffer op peil te houden voor periodes waarin de economie in een onzekere toestand verkeert. Om dat te bereiken zijn rendementen nodig die naar verwachting hoger zijn dan het rendement van een risicovrije investering (zoals rente op de bank of op staatsobligaties).

Het is dan ook van belang dat beheerders van zulke portefeuilles, verder **managers** te noemen, kwaliteiten hebben in het voorspellen van (en anticiperen op) bewegingen in de markt. Zulke vaardigheden zijn nodig om de gewenste rendementen te kunnen behalen. Daarnaast is het vermogensbeheer een dynamische industrie, met veel concurrentie. Instellingen in deze branche selecteren veelal hun managers op grond van hun prestaties (ofwel **performance**) in het verleden. Verder is de performance van een opererende manager bepalend voor beslissingen omtrent het aanhouden of ontslaan van de manager.

Iedere manager krijgt door zijn opdrachtgever voorwaarden opgelegd met betrekking tot de te kiezen beleggingsportefeuille. Bijvoorbeeld, de portefeuille bestaat uit Nederlandse aandelen, waarvan minstens 10% van het kapitaal in de technologie sector dient te worden belegd. Deze voorwaarden of constructieregels tezamen (inclusief door de opdrachtgever bepaalde doelstellingen ten aanzien van de resultaten) wordt het **mandaat** van de manager genoemd. Hoewel het gegeven mandaat de manager beperkingen oplegt, heeft deze normaal gesproken de keuze uit een groot aantal mogelijke portefeuilles, en tevens bewegingsvrijheid om transacties uit te voeren, dat wil zeggen beleggingsobjecten te kopen of te verkopen. De totale waarde van de portefeuille gaat fluctueren door het resultaat op deze transacties en door het verloop van de koersen. De relatieve verandering in de portefeuillevaarde over een bepaalde periode heet het **rendement** (of **return**) over deze periode.

**Performance evaluatie (PE)** is het vakgebied dat zich bezighoudt met kwantitatieve methoden en technieken om beleggingsmanagers te beoordelen op de resultaten van de door hen beheerde portefeuilles. De performance wordt vastgesteld op basis van de gerealiseerde rendementen, samen met onder andere het mandaat, de ontwikkelingen in de markt, en in sommige gevallen de resultaten van andere managers.

## 1.2 ORTEC en de afdeling Performance and Risk Implementation

Deze paragraaf geeft een globale indruk van het bedrijf ORTEC. In het bijzonder komt de afdeling Performance and Risk Implementation aan bod, waar ik stage heb gelopen. Voor meer informatie verwijst ik naar de website van ORTEC: <http://nl.ortec.com/>.

ORTEC is op 1 april 1981 opgericht door vijf studenten econometrie, met als doel hun vakgebied in het bedrijfsleven toe te passen op consultancy basis. Wat aanvankelijk op kleine schaal begon, is uitgegroeid tot een internationale onderneming met meer dan 600 medewerkers (ten tijde 2007). De projecten waar ORTEC zich mee bezighoudt hebben veelal betrekking op planning, optimalisatie of simulatie. De kernbezigheden omvatten consultancy,

verkoop/implementatie van softwarepakketten, maatwerk en opleidingen. De doelgroep waar ORTEC voor werkt is zeer gevarieerd. Tot de klantenkring behoren ziekenhuizen, olieraffinaderijen, luchtvaartmaatschappijen, spoorwegen, post- en pakketdiensten, taxicentrales, overheid, banken, pensioenfondsen en verzekeringsmaatschappijen. Voor al deze klanten analyseert ORTEC complexe problemen en ontwikkelt het softwarepakketten die deze problemen zo goed mogelijk oplossen.

De afdeling **Performance and Risk Implementation**, onderdeel van de financiële tak van ORTEC, is verantwoordelijk voor onder andere de implementatie van modellen en methoden van performance evaluatie voor pensioenfondsen, en vermogensbeheerders. Sinds 1993 ontwikkelt de afdeling het softwarepakket **PEARL**, dat performance rapporten genereert op basis van data van de markt en de portefeuilles van de klant. Dit helpt de organisaties om hun beleggingsbeslissingen te analyseren en hun strategie op de juiste manier aan te passen met als doel om de totale performance te verbeteren.

### 1.3 De stageopdracht

Tijdens mijn stage heb ik een nieuwe methode voor performance evaluatie onderzocht. Daartoe zijn eerst de standaardmethoden geanalyseerd, die tegenwoordig het meest worden gebruikt. Twee van zulke methoden zijn:

- Vergelijking van de behaalde performance met die van een marktindex welke weergeeft hoe de markt zich heeft ontwikkeld over de evaluatieperiode. Zo'n index heet wel een **benchmark**.
- Vergelijking van de behaalde performance met die van een groep andere managers die een soortgelijke portefeuille hebben beheerd; een zogeheten **peer group**.

Deze standaardmethoden blijken echter onderhevig te zijn aan een aantal onzuiverheden. De alternatieve methode die ik heb onderzocht elimineert de onzuiverheden. Deze methode genereert zogeheten **portfolio opportunity distributions (PODs)**, en is halverwege de jaren '90 bedacht door Ronald Surz. De reden dat de methode toen pas tot stand is gekomen, is dat er gebruik wordt gemaakt van simulatie en dus van grote rekenkracht van moderne computers.

Tegenwoordig wordt de methode echter nog niet veel gebruikt. De benchmarks en peer groups zijn zo sterk ingeburgerd in de financiële wereld, dat vrijwel niemand twijfelt aan de correctheid van hun werking, en belangstelling heeft voor een alternatief. Dit is jammer, want zoals verderop wordt besproken, geven de standaardmethodes in veel gevallen onjuiste resultaten, met als gevolg dat managers onterecht worden ontslagen of aangehouden.

Ook bij ORTEC was voor aanvang van de stage weinig bekend over PODs. Het pakket PEARL is met name gebaseerd op benchmark methodes. Toch is het mogelijk dat de POD methode in de toekomst één van de standaarden wordt, zodat het belangrijk wordt om een product te ontwikkelen dat deze methode kan uitvoeren. Gebeurt dit niet op korte termijn, dan vormt een dergelijk product wellicht een goede aanvulling op de huidige dienstverlening. Mogelijk leveren de resultaten van de berekende PODs een toegevoegde waarde aan de performance rapporten van klanten.

De opdracht van de stage was het verkrijgen van dieper inzicht in de POD methode. Daarbij was de bedoeling in ieder geval niet om meteen een nieuw product of module te ontwikkelen. Uiteraard is dit in de toekomst wel mogelijk, maar de nadruk van de stage lag vooral op nieuwe inzichten. Aldus is in de komende hoofdstukken een uitgebreide beschrijving

opgenomen van de precieze werking van de methode, evenals een analyse met betrekking tot praktische bruikbaarheid ervan.

Hoofdstuk 2 geeft een overzicht van het vakgebied van PE in het algemeen. De belangrijkste begrippen en methoden komen aan de orde. De nadruk zal echter liggen op de gedeelten die voor het onderwerp van de scriptie het meest essentieel zijn. De standaardmethoden van benchmarks en peer groups worden in meer detail beschreven, samen met de onzuiverheden die hierboven genoemd zijn.

In hoofdstuk 3 wordt de alternatieve methode van PODs beschreven. We behandelen hoe de methode in grote lijnen werkt en hoe deze de tekortkomingen van de standaardtechnieken elimineert. Ook komen er enkele punten van kritiek aan de orde. De laatste twee paragrafen van dit hoofdstuk bespreken hoe de statistiek gebruikt kan worden om met behulp van PODs tot een performance uitspraak van een manager te komen.

Hoofdstuk 4 is gericht op de detaillering van de POD techniek. Er worden drie algoritmes geformuleerd, evenals de wiskundige theorie waarop deze zijn gebaseerd. Dit is het technische gedeelte van de scriptie. Vervolgens vergelijk ik de drie algoritmes met elkaar en geef ik advies betreffende het gebruik van ieder algoritme in verschillende situaties. Ook worden er in hoofdstuk 4 enkele uitbreidingen op de methode behandeld.

In hoofdstuk 5 pas ik de POD methode toe op een aantal voorbeelden die gebaseerd zijn op in werkelijkheid bestaande beleggingsportefeuilles. Deze voorbeelden illustreren de verschillende aspecten van PODs en laten tevens zien hoe de uitvoering van de methode in de praktijk werkt, en hoe de resultaten gepresenteerd kunnen worden.

Tot slot is hoofdstuk 6 gewijd aan conclusies en aanbevelingen die naar aanleiding van het onderzoek kunnen worden gedaan.

De tekst van de scriptie heb ik zoveel mogelijk geschreven in de wij-vorm. Hiermee neem ik de lezers bij de hand en poog ik de stof op een informele manier te behandelen. Echter, op plaatsen waar de situatie betrekking heeft op mij persoonlijk (bijvoorbeeld bij het geven van een advies, een conclusie of een persoonlijke mening) ga ik over op de ik-vorm.



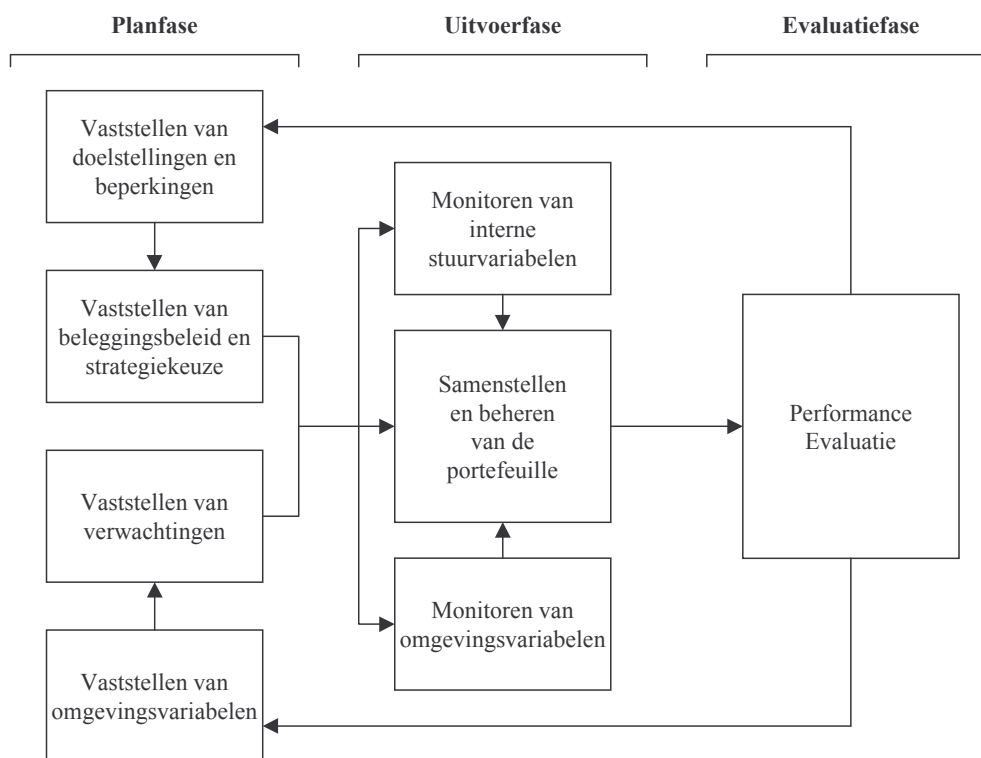
## 2. Huidige technieken voor performance evaluatie

In dit hoofdstuk komen beknopt de meest gebruikte methodes voor PE aan de orde. We bespreken onder meer hoe rendementen worden berekend en welke performance statistieken uit deze rendementen kunnen worden afgeleid. Daarbij worden onder andere de voor- en nadelen van benchmarks en peer groups uiteengezet. Het alternatief, PODs, is het onderwerp van hoofdstuk 3.

Bij de bespreking van deze algemene theorie beperken we ons tot datgene wat voor de scriptie het meest van belang is. Voor uitvoerige besprekingen van de theorie van PE verwijzen we naar [1], [2], [4] en [9] uit de literatuurlijst.

### 2.1 De rol van PE binnen het beleggingsproces

In het algemeen bestaat het beleggingsproces van een vermogensbeheerinstelling uit drie fasen: de planfase, de uitvoerfase en de evaluatiefase. Figuur 1 laat schematisch de processen binnen deze drie fasen zien. De pijlen in de figuur laten zien hoe de gegevensstroom verloopt: uitvoer van een proces is invoer voor een proces waar een uitgaande pijl naartoe leidt.

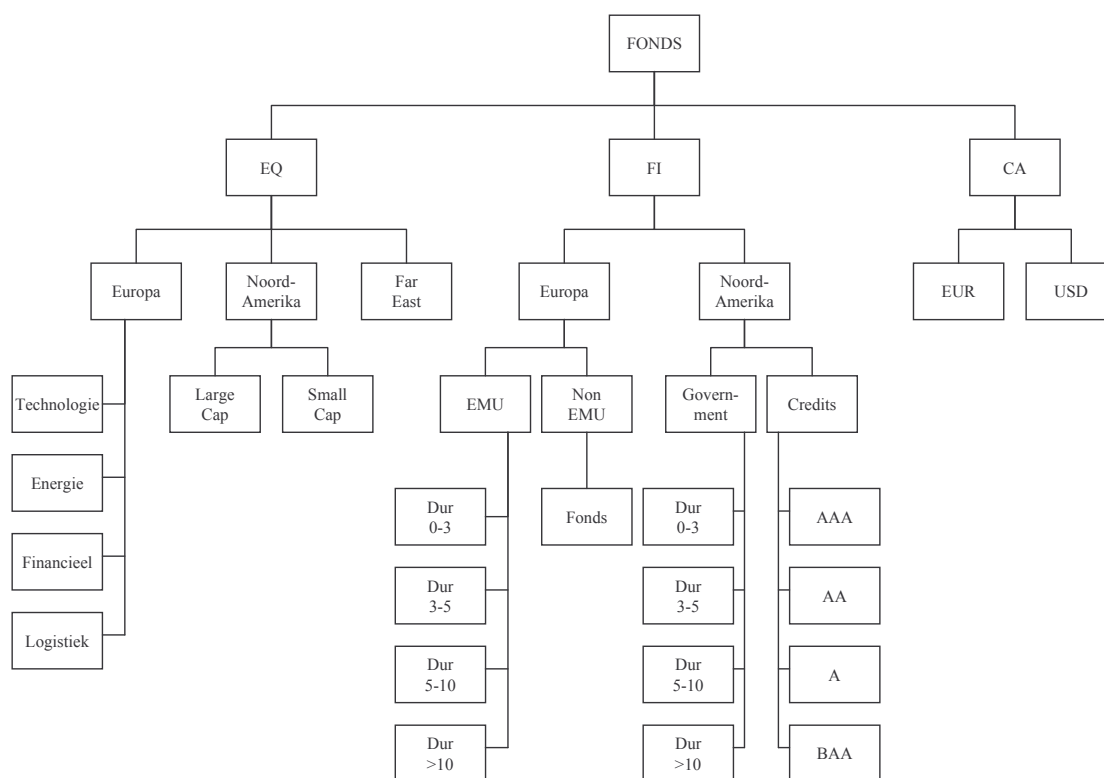


Figuur 1: Het beleggingsproces.

We zien dat performance evaluatie niet per definitie de laatste stap in het geheel is; de uitkomsten van PE leveren feedback aan de planfase voor een volgende periode. Dit maakt PE tot een volwaardig onderdeel van het gehele beleggingsproces.

## Investment Decision Process

Het gedeelte van de planfase en de uitvoerfase, waarbij de portefeuille wordt geconstrueerd en vergelijkingsmaatstaven voor de evaluatiefase worden vastgelegd, duidt men wel aan als het **Investment Decision Process (IDP)**. Het constructiegedeelte wordt vaak voorgesteld door een boomstructuur, zoals in figuur 2.



**Figuur 2: Voorbeeld IDP boom.**

Ieder knooppunt in deze boom stelt een **segment** van de totale portefeuille voor, dat afzonderlijk wordt beheerd door een manager of team van managers. De segmenten onderin de boom, die geen “kinderen” hebben, zijn uit marktinstrumenten opgebouwd. De overige segmenten kunnen worden opgevat als portefeuilles van deelsegmenten, ofwel deelmarkten. De beheerder van zo’n portefeuille kan schuiven met het kapitaal dat in iedere deelmarkt is belegd, op basis van zijn verwachting over het rendementverloop van de deelmarkten.

De allereerste vertakking in de IDP boom is een typische lange termijn beslissing. Een herbalancering vindt op dit niveau niet vaak plaats. Het is de verdeling van de portefeuille over de globale beleggingscategorieën. Grofweg zijn dat de volgende categorieën:

- aandelen (Equity, EQ);
- obligaties en andere vastrentende beleggingen (Fixed Income, FI);
- onroerend goed (Real Estate, RE, is achterwege gelaten in het voorbeeld hierboven);
- kas (Cash, CA).

Binnen deze categorieën wordt verder opgesplitst naar bijvoorbeeld:

- land of werelddeel;
- sector of stijl (marktkapitalisatie en koers/winst verhouding) voor EQ segmenten;
- kredietklasse of duration voor FI segmenten.



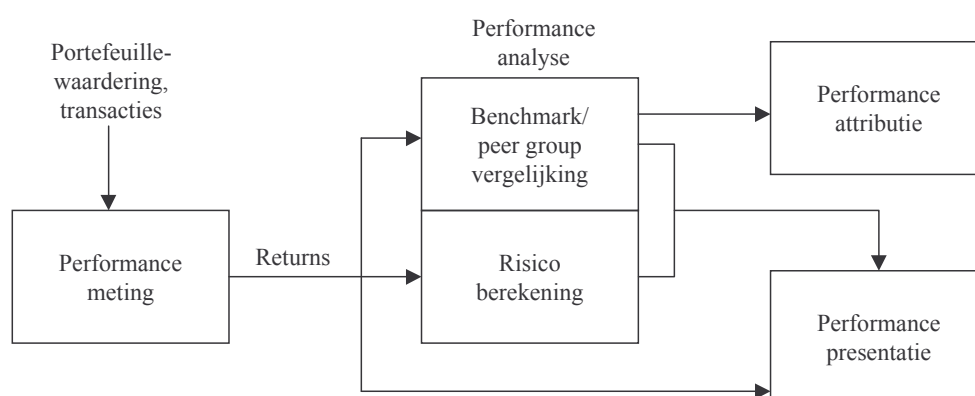
Het opsplitsen van de portefeuille in segmenten, zoals hierboven beschreven, en het verdelen van het kapitaal over deze segmenten, noemen we **asset allocatie** of **timing**. Dit proces vindt deels plaats op strategisch/tactisch niveau binnen de planfase, en deels op operationeel niveau binnen de uitvoerfase.

Het selecteren van marktinstrumenten binnen de segmenten onderin de IDP boom, en het beheren van deze segmenten, noemen we **security selectie**. Dit proces vindt geheel plaats op operationeel niveau binnen de uitvoerfase. Meestal heeft dit proces aanzienlijk minder invloed op het totale portefeuillerendement dan de asset allocatie.

In het algemeen kan gesteld worden dat hoe lager een segment zich in de boomstructuur bevindt, hoe frequenter de samenstelling van het segment wordt gewijzigd, terwijl de invloed op het rendement van de totale portefeuille juist kleiner is. Hoe tegenstrijdig het ook lijkt, de belangrijkste beleggingsbeslissingen blijken de minste aandacht te krijgen, en de onbelangrijkste beslissingen de meeste aandacht.

## Deelprocessen van PE

Het proces van performance evaluatie bestaat uit een aantal deelprocessen waarvan globaal de samenhang wordt weergegeven door onderstaande figuur 3.



Figuur 3: Stappen van Performance Evaluatie.

We onderscheiden de volgende deelprocessen.

- **Performance meting** – Dit is het berekenen van de rendementen van de totale portefeuille, de segmenten en de individuele marktinstrumenten. De verschillende formules en andere zaken gerelateerd hieraan worden in paragraaf 2.2 besproken.
- **Performance analyse**<sup>1</sup> – Het beantwoorden van de vraag: zijn de behaalde rendementen goed? Is de manager echt vaardig of heeft hij geluk gehad? Daartoe worden de rendementen ergens mee vergeleken, bijvoorbeeld een benchmark of peer group. Een beschrijving van deze maatstaven, alsmede de standaardtechnieken uit de statistiek waarmee deze vergelijkingen worden gedaan, komen aan de orde in paragraaf 2.3. Ook is het van belang om bij de beoordeling van de performance het genomen risico in ogenschouw te nemen. Immers, beleggers hebben in het algemeen bij vast rendement de voorkeur voor een belegging met zo weinig mogelijk risico. Bovendien zijn beleggers in

<sup>1</sup> Dit onderdeel afzonderlijk wordt ook wel **performance evaluatie** genoemd, waarmee wij echter het gehele proces aangeven. Helaas bestaat hiervoor geen eenduidige terminologie. Het gehele proces wordt in de literatuur ook met **performance meting** of **performance analyse** aangeduid.

hun mandaat vaak beperkt tot het nemen van een bepaald maximaal risico. Een aantal veelgebruikte risicocorrigerende statistieken worden besproken in paragraaf 2.4.

- **Performance attributie** – Na het analyseren van de performance wil de portefeuille-eigenaar weten hoe deze tot stand is gekomen. Bij attributie wordt de totale performance, gemeten door vergelijking met een benchmark, opgesplitst in termen (dit kan zowel additief als geometrisch), waarbij iedere term weergeeft wat de bijdrage is van een bepaalde managementbeslissing aan de totale performance. Paragraaf 2.5 is gewijd aan attributietechnieken.
- **Performance presentatie** – Dit is het presenteren van in het verleden behaalde performance resultaten door vermogensbeheerinstellingen aan huidige en potentiële klanten. Dit is van belang omdat klanten hun keuze voor een manager vaak baseren op diens resultaten. Om te voorkomen dat beleggers hun resultaten sterk kunnen manipuleren, zijn door instellingen als AIMR en VBA bepaalde regels, zogeheten performance presentatie standaards, opgesteld. Hierover meer in paragraaf 2.6.

## Criteria voor PE

Gegeven het feit dat de loopbaan van een portefeuillemanager in grote mate afhangt van zijn performance, is het zaak om evaluatiemethoden te ontwikkelen waarmee op een objectieve en eerlijke manier de beleggingsvaardigheden van een manager kunnen worden beoordeeld. Alvorens we de klassieke PE methoden behandelen in de volgende paragrafen, noemen we een aantal criteria waaraan goede PE methoden, en het gebruik ervan, dienen te voldoen:

1. **Geschiktheid** – De gekozen meetmethode moet passen bij de situatie van de belegger. Bijvoorbeeld, ingeval de belegger slechts enkele transacties per maand uitvoert heeft het weinig zin om dagelijkse rendementen te bepalen; dit geeft onregelmatige fluctuaties in de tijdreeks. Verder moet de analyse van de performance corresponderen met het mandaat van de manager. Is een manager bijvoorbeeld gebonden aan een maximaal risico, dan moet bij de beoordeling van zijn rendement ook het genomen risico meewegen.
2. **Zuiverheid** – De performance van een manager dient te worden afgeleid op basis van zijn eigen beleggingscriteria. Factoren waar de manager geen controle over heeft, mogen zijn performance niet beïnvloeden (dit zowel in positieve als negatieve zin). Gevolgen van beslissingen over de portefeuille die niet door de manager zijn genomen, dienen te worden geëlimineerd bij performance berekeningen.
3. **Consistentie** – De berekende maatstaven dienen consistent gebruikt te worden. Gebruik van steeds verschillende maatstaven over de tijd geeft een vertekend beeld van de totale performance, bijvoorbeeld wanneer steeds die maatstaven worden gebruikt die het gunstigst ogende resultaat laten zien met betrekking tot de performance.
4. **Significantie** – Bij gebruik van statistische technieken dient een voldoende groot aantal waarnemingen aanwezig te zijn, zodat de uitkomsten betrouwbaar zijn en corresponderen met de werkelijkheid. Tevens dient onderzocht te worden of de waarnemingen voldoen aan de voorwaarden van het gebruikte statistische model.
5. **Praktische uitvoerbaarheid** – Rapporten met performance berekeningen moeten kunnen worden gemaakt op basis van dagelijks beschikbare gegevens. Daarbij mag de rekentijd van de computer niet te groot zijn. En ingeval gebruik wordt gemaakt van benaderingen of schattingen, moet de benadering dicht genoeg bij de werkelijke waarde liggen, zodat de uiteindelijke conclusies niet bepaald worden door de benaderingsfout.
6. **Standaardisatie** – Er moet overeenstemming zijn in het gebruik van een gegeven meetmethode. Ingeval er meerdere varianten op een methode bestaan, moeten er universele afspraken zijn met betrekking tot welke variant in welke situatie wordt toegepast. Dit voorkomt dat de performance met verschillende maten wordt gemeten.

Aan de hand van deze criteria kan voor de PE methoden, die behandeld worden in de scriptie, een indruk worden verkregen in hoeverre zij geschikt zijn om in de praktijk te worden toegepast.

## 2.2 Rendement berekening

Deze paragraaf bespreekt hoe rendementen worden berekend. Dit zijn de basisstatistieken van performance evaluatie. De hierop volgende paragrafen behandelen hoe de gerealiseerde rendementen kunnen worden geanalyseerd.

### Rendement

Veronderstel dat  $V_b$  de marktwaarde is van een belegging aan het begin van een meetperiode en  $V_e$  de marktwaarde aan het einde van de periode. Het **rendement**  $R$  over de periode (in het Engels **rate of return** of kortweg **return**) is de relatieve aangroei van de waarde van de belegging:

$$R := \frac{V_e - V_b}{V_b} = \frac{V_e}{V_b} - 1.$$

De term  $V_e/V_b$  in de tweede formule is gelijk aan  $R+1$ , en heet de **groefactor** van het kapitaal. Groeifactoren kunnen worden vermenigvuldigd bij berekening van een rendement over meerdere (sub)periodes. Is  $R_i$  de return over subperiode  $[t_{i-1}, t_i]$  voor  $i=1, \dots, n$ , dan is de return  $R_{0,n}$  over de gehele periode  $[t_0, t_n]$  gegeven door

$$R_{0,n} = (1 + R_1) \cdots (1 + R_n) - 1.$$

Dit proces heet wel **geometrische samenstelling** (in het Engels **compounding** of **chain-linking**).

In veel gevallen komt rendement op een belegging niet alleen tot stand door verandering in de marktwaarde. Ook is het mogelijk dat gedurende de meetperiode **inkomen** wordt ontvangen, zoals dividend op een aandeel of couponrente op een obligatie. Een correcte returndefinitie houdt hier rekening mee. Is het totaal inkomen over de periode gelijk aan  $I$ , dan kan de formule voor het rendement worden aangepast door  $V_e$  te vervangen door  $V_e + I$ . Dit rendement heet een **total return**.

Soms wordt inkomen wel verdiend binnen de meetperiode maar nog niet uitbetaald. Dit wordt wel **accrued income** genoemd. In veel gevallen bestaat het totaal inkomen  $I$  uit zowel ontvangen als verdiend inkomen. Het is dan gebruikelijk om

- $V_e$  te corrigeren door het ontvangen inkomen en het verdiende inkomen aan het einde van de periode erbij op te tellen;
- $V_b$  te corrigeren door het verdiende inkomen aan het begin van de periode erbij op te tellen.

### Rendement op een portefeuille

De return op een portefeuille kan worden berekend uit de returns op alle marktinstrumenten waaruit de portefeuille bestaat. Stel, een portefeuille  $P$  is opgebouwd uit  $N$  verschillende instrumenten. Laat

- $V_b^j$  := de beginwaarde van het kapitaal belegd in instrument  $j$  (eventueel aangepast voor inkomen);  
 $V_e^j$  := de eindwaarde van het kapitaal belegd in instrument  $j$  (eventueel aangepast voor inkomen);  
 $R^j$  :=  $V_e^j / V_b^j - 1$ , het rendement op instrument  $j$ .

De fractie van het totale kapitaal belegd in  $j$  wordt gegeven door het **gewicht** van  $j$ :

$$w^j := \frac{V_b^j}{\sum_{j=1}^N V_b^j},$$

voor  $j=1, \dots, N$ . Het gewicht van een instrument wordt geïnterpreteerd als de gevoeligheid of **exposure** van de portefeuillewaarde voor dat instrument.

Het rendement op de portefeuille  $R^P$  wordt nu als volgt berekend:

$$R^P = \frac{\sum_{j=1}^N (V_e^j - V_b^j)}{\sum_{j=1}^N V_b^j} = \frac{\sum_{j=1}^N V_b^j R^j}{\sum_{j=1}^N V_b^j} = \sum_{j=1}^N w^j R^j.$$

Met andere woorden, de return op de portefeuille is een lineaire combinatie van de returns op de instrumenten, met als coëfficiënten de exposure gewichten. Deze lineaire combinatie wordt wel een **convexe combinatie** genoemd, omdat de coëfficiënten tot 1 sommeren.

De gegeven formule voor  $w^j$  is een basisformule voor het exposure gewicht. In werkelijkheid kan de gevoeligheid afwijkend zijn, bijvoorbeeld door toepassing van hedging of leverage (via gebruik van derivaten). Ook kunnen gewichten negatief zijn wanneer de portefeuille een short positie aanneemt in het corresponderende instrument. Op deze materie gaan wij verder niet in; we nemen steeds aan dat er gewichten  $w^1, \dots, w^N$  bestaan die tot 1 sommeren zodat aan bovenstaande formule voor het portefeuillerendement is voldaan.

## Kasstromen

Een **kasstroom** of **cash flow** is een transactie waarbij vanuit een externe bron een bedrag wordt toegevoegd aan de portefeuille (**inflow**) of juist wordt onttrokken van de portefeuille (**outflow**). Voorbeelden van cash flows zijn:

- stortingen of opnames door de eigenaar van het fonds;
- stortingen of opnames door aandeelhouders van een beleggingsfonds;
- uitbetaling van loon aan de manager met geld uit het fonds (**management fee**);
- betaling van andere lasten (bewaring, accountancy, belasting) met geld uit het fonds;
- overheveling van kapitaal van of naar een ander portefeuillesegment die door een andere manager wordt beheerd.

Geén cash flows zijn:

- koop- of verkoop transacties door de manager van het fonds;
- overheveling van kapitaal tussen segmenten die allebei onderverdeeld zijn in het segment dat de manager in beheer heeft.

Dit zijn geen cash flows omdat de geldstroom geheel binnen de portefeuille plaatsvindt. Er is geen externe bron waar het bedrag vandaan komt of waar het naartoe gaat.

Cash flows spelen een belangrijke rol bij returnberekening. Stel bijvoorbeeld dat een portefeuille in waarde stijgt tot halverwege de meetperiode. Op dat moment besluit de portefeuille eigenaar om een extra bedrag te storten. Er vindt dus een inflow plaats. Echter, gedurende de tweede helft van de periode daalt de portefeuillewaarde. Het behaalde rendement over de gehele periode (en over het volledige belegde kapitaal) valt nu lager uit dan ingeval de eigenaar geen storting had gedaan!

Rendementen zijn dus afhankelijk van cash flows, maar de manager heeft meestal geen invloed op de timing en grootte van deze cash flows. In het algemeen zijn het externe factoren! Het is dan ook oneerlijk om over de manager te oordelen op basis van rendementen die afhankelijk zijn van cash flows. Dit motiveert een definitie van twee soorten rendementen:

1. Een rendement van de *eigenaar*. Dit is het daadwerkelijke rendement van de portefeuille, inclusief de invloed van cash flows. Het geeft het verloop weer zoals de eigenaar dat ervaart. Een dergelijk rendement heet een **money-weighted return (MWR)**.
2. Een rendement van de *manager*. Dit is het rendement dat enkel en alleen tot stand komt door invloeden die de manager zelf onder controle heeft. Cash flows hebben geen invloed op dit rendement. Een dergelijk rendement heet een **time-weighted return (TWR)**.

Het onderscheiden van beide typen rendement werd voor 't eerst voorgesteld door P. Dietz in een artikel uit 1966. Naar aanleiding van dit artikel publiceerde de Bank Administration Institute (BAI) in 1968 een document met richtlijnen voor performance berekening. Eén van de nieuwe richtlijnen was het gebruik van TWR formules. We bespreken hoe MWRs en TWRs kunnen worden berekend.

### Money-weighted return

Stel, er vinden  $m$  cash flows plaats gedurende de evaluatieperiode. Definieer

$$\begin{aligned} CF_j &:= \text{bedrag van cash flow } j \text{ (positief bij een inflow, negatief bij een outflow);} \\ t_j &:= \text{tijdstip van cash flow } j. \end{aligned}$$

Definieer verder  $p_j \in [0,1]$  als de fractie tijd over de meetperiode waarin cash flow  $j$  aanwezig is in de portefeuille. Dus

$$p_j := \frac{t - t_j}{t},$$

voor  $j = 1, \dots, m$ , ingeval  $[0, t]$  de hele periode voorstelt.

Aan de hand van deze gegevens definiëren we de MWR volgens de **Internal Rate of Return (IRR)**. Dit is het rendement dat behaald zou zijn, wanneer alle componenten van het belegde kapitaal (beginwaarde plus cash flows) dezelfde return zouden ervaren (ofwel, wanneer de belegging continu volgens een constante factor zou stijgen of dalen). In wiskundige termen is de IRR gedefinieerd als het unieke reële getal  $R_{\text{MWR}}$  zodat

$$V_e = V_b(1 + R_{\text{MWR}}) + \sum_{j=1}^m CF_j(1 + R_{\text{MWR}})^{p_j}.$$

Het rechterlid is strikt stijgend of dalend in  $R_{\text{MWR}}$ , daarom is deze uniek.

Bovenstaande vergelijking kan tegenwoordig op een computer eenvoudig worden opgelost. Toch is er soms behoefte aan een expliciete formule voor de MWR. De **modified Dietz return** geeft een benadering voor de IRR en is gedefinieerd door

$$R_{\text{MWR}} = \frac{V_e - V_b - \sum_{j=1}^m CF_j}{V_b + \sum_{j=1}^m p_j CF_j}.$$

Deze benadering kan slecht zijn, met name als de waarden van de cash flows relatief groot zijn of de waarde van de portefeuille sterk fluctueert.

Soms worden de tijdstippen  $t_j$  niet bijgehouden en zijn de fracties  $p_j$  dus niet bekend. In dat geval wordt vaak als grove schatting de modified Dietz return genomen met alle  $p_j = 1/2$ . Dit heet wel de **simple Dietz return**.

### Time-weighted return

Om de TWR, de return zonder invloed van kasstromen, te kunnen berekenen hebben we de waardes nodig die de portefeuille aanneemt vlak voor een kasstroom. Zij  $V_{t_j}$  de portefeuillewaarde op tijdstip  $t_j$  vlak voordat cash flow  $j$  plaatsvindt. Noteer met  $R_j$  de return over subperiode tussen cash flow  $j-1$  en  $j$ :

$$R_j = \frac{V_{t_j} - V_{t_{j-1}} - CF_{j-1}}{V_{t_{j-1}} + CF_{j-1}},$$

voor  $j = 2, \dots, m$ , met in het bijzonder  $R_1 = (V_{t_1} - V_b) / V_b$  de return over  $[0, t_1]$  en  $R_{m+1} = (V_e - V_{t_m}) / V_{t_m}$  de return over  $[t_m, t]$ . De TWR is gedefinieerd als de geometrische samenstelling van de subperiodieke returns:

$$R_{\text{TWR}} := (1 + R_1) \cdots (1 + R_{m+1}) - 1.$$

Deze return is ongevoelig voor de timing en grootte van cash flows. Een nadeel is echter dat de portefeuillewaarden vlak vóór de cash flows bekend moeten zijn. Dit is niet altijd het geval. Daarom wordt in de praktijk vaak een benadering gebruikt bij TWR berekeningen. Stel bijvoorbeeld dat we van de portefeuille dagelijks de slotwaarden bijhouden. We weten op welke dagen de cash flows hebben opgetreden. Onder de aanname dat de cash flows aan het einde van de dag plaatsvinden, kunnen we de TWR benaderen door de dagelijkse returns geometrisch te linken, waarbij de dagelijkse returns benaderd zijn door een MWR (bijvoorbeeld IRR of modified Dietz). Deze approximatie is redelijk tenzij de cash flows groot zijn of de portefeuillewaarde sterk aan schommeling onderhevig is.

### Gemiddeld subperiodiek rendement

Wanneer we rendementen hebben gemeten over  $n$  opvolgende subperiodes (bijvoorbeeld per dag, maand of kwartaal), dan is het voor PE doeleinden van belang om een soort gemiddeld rendement te kunnen berekenen. Daarin kennen we twee smaken:

Het **rekenkundig (aritmetisch) gemiddelde** is

$$\bar{R}_n := \frac{1}{n} \sum_{i=1}^n R_i .$$

Het **meetkundig (geometrisch) gemiddelde** is

$$\bar{R}_n^{\text{geom}} := [(1 + R_1) \cdots (1 + R_n)]^{1/n} - 1 .$$

Welke van de twee gebruikt dient te worden hangt af van het doel waarvoor we het gemiddelde willen weten. Het meetkundig gemiddelde houdt rekening met het interactie-effect tussen opvolgende returns, dat ontstaat door de samenstelling. Het rekenkundig gemiddelde doet dat niet, maar kan dienen als schatter voor de verwachte return over een subperiode. Merk overigens op dat

$$\bar{R}_n \geq \bar{R}_n^{\text{geom}} ,$$

met gelijkheid als en alleen als de subperiodieke returns gelijk zijn.

### Rendement per jaar

Vaak worden rendementen op jaarbasis gerapporteerd. Dit maakt het (tot op zekere hoogte) mogelijk om rendementen onderling te vergelijken wanneer die over tijdsperiodes van verschillende lengte zijn verkregen. Een rendement  $R$  over een periode *langer* dan een jaar kan als volgt worden omgezet naar een rendement over een jaar - een zogeheten **annualized return**:

$$R^{\text{ann}} = (1 + R)^p - 1 ,$$

waarbij  $p$  de **periodiciteit**, dat wil zeggen het aantal subperiodes per jaar gedeeld door het totaal aantal subperiodes waaruit de meetperiode bestaat. Is  $R$  bijvoorbeeld berekend over 13 maanden, dan is  $p = 12/13$ . Het getal  $R^{\text{ann}}$  is dan de return die zou ontstaan, als een jaar lang per maand hetzelfde rendement behaald zou worden, namelijk het meetkundig gemiddelde over alle 13 maanden.

Bovenstaande truc kan rekentechnisch ook worden toegepast voor periodes *korter* dan een jaar, maar dat is enigszins bezwaarlijk, omdat prestaties voor kortere periodes meer uiteen kunnen lopen en daarom niet representatief zijn voor langere periodes.

## 2.3 Benchmarks en peer groups

Het rendement, zoals ingevoerd in de vorige paragraaf, is een belangrijke maatstaf voor performance evaluatie, maar geeft niet de volledige werkelijkheid weer. Stel bijvoorbeeld dat twee beleggers A en B over dezelfde periode dezelfde return hebben behaald, maar de algehele markt waarin A belegde was minder rendabel dan de markt waarin B belegde. Het is dan onterecht om te concluderen dat beide beleggers dezelfde performance hebben afgeleverd. Op grond van de gegevens ligt het voor de hand dat A beter heeft gepresteerd omdat de markt het hem moeilijker maakte.

Om een eerlijke performance meting te verrichten is het zaak om iets te hebben waarmee we het behaalde resultaat kunnen vergelijken. In deze paragraaf bespreken we de twee meest



gebruikte vergelijkingsmaatstaven: benchmarks en peer groups. Uit [11] blijkt dat 95% van de consultants en beleggingsanalisten gebruik maakt van benchmarks bij PE projecten. Gebruik van peer groups neemt de laatste tijd wat af, maar behoort nog steeds tot de standaarden. Doordat beide methoden zo standaard zijn wordt door slechts weinigen hun correctheid enigszins betwijfeld. Toch zijn beide onderhevig aan onzuiverheden, met als gevolg dat in veel situaties de resulterende statistieken niet weergeven welke performance daadwerkelijk is behaald.

## Benchmarks

Een **benchmark** is een (mogelijk fictieve) portefeuille van instrumenten uit de markt waarin een manager belegt. Het idee is dat het rendementverloop van een benchmark weergeeft hoe de markt als geheel rendeert over de te evalueren periode. We zeggen dat de manager **outperformance** levert wanneer zijn return hoger is dan die van de benchmark, en **underperformance** wanneer die lager is. Meestal is de keuze van de benchmark onderdeel van het IDP en vindt het plaats in de planfase.

De manager kan als doelstelling hebben om de markt te verslaan, dus een outperformance te behalen. We spreken dan van **actief beheer**. Een actieve manager zal de markt op de voet volgen en anticiperen op mogelijkheden om hoge winsten te behalen. Daarbij worden relatief veel transacties uitgevoerd en wordt eventueel gebruik gemaakt van derivaten.

Ook kan het mandaat stellen dat de benchmark slechts gevolgd dient te worden. Dit noemen we **passief beheer**. De manager houdt dan de exposure gewichten van de benchmark aan, en herbalanceert deze af en toe d.m.v. enkele transacties. In derivaten wordt bij dit type beheer niet veel gehandeld.

Er bestaan verschillende soorten benchmarks. Een **markt index** geeft het gezamenlijk verloop van een bepaalde collectie van marktinstrumenten weer. Standen van zulke indices worden veelal gepubliceerd in kranten en op internet. Voorbeelden zijn de AEX Index, een gewogen som van 25 Nederlandse large cap aandelen, en de Standards & Poor's 500 Index, die de Amerikaanse large cap aandelenmarkt representeert. Verder zijn er instellingen zoals MSCI en FTSE die specifieke indices beschikbaar stellen, op basis van eigenschappen zoals land, sector, kapitalisatie en koers/winst verhouding.

De stand van een index wordt in het algemeen niet in geld uitgedrukt maar in punten. Aanvankelijk wordt de index geïnitieerd op een basiswaarde van bijvoorbeeld 1000 punten. De indexwaarde aan het einde van een periode wordt als volgt berekend:

$$I_e = \frac{\sum_{j=1}^N a_e^j V_e^j}{\sum_{j=1}^N a_b^j V_b^j} I_b,$$

waarbij,

- $N$  := het aantal instrumenten in de index;
- $I_b$  := de indexwaarde aan het begin van de periode;
- $I_e$  := de indexwaarde aan het einde van de periode;
- $a_b^j$  := het aantal aanwezige stukken van instrument  $j$  aan het begin van de periode;
- $a_e^j$  := het aantal aanwezige stukken van instrument  $j$  aan het einde van de periode;
- $V_b^j$  := de marktwaarde van instrument  $j$  aan het begin van de periode;
- $V_e^j$  := de marktwaarde van instrument  $j$  aan het einde van de periode.



Het rendement op de index over dezelfde periode is dan gelijk aan

$$R = \frac{I_e}{I_b} - 1.$$

Het kan voorkomen dat de markt waarin een manager belegt zo specifiek is dat er geen index bestaat die deze markt correct representeert. Een index kan stukken bevatten waarin de manager niet mag beleggen, of er kunnen stukken ontbreken waarin de manager wel mag beleggen. Dit probleem wordt opgelost door gebruik te maken van een **custom benchmark** (ook genoemd **normal portfolio**). Een dergelijke benchmark wordt speciaal voor de manager samengesteld om overeen te komen met zijn individuele beleggingscriteria. Vaak is een custom benchmark een gewogen som van bestaande benchmarks, waarbij de gewichten worden afgeleid uit gegevens over het mandaat.

Aangenomen dat bij de te evalueren portefeuille een benchmark is gedefinieerd, is de vraag hoe deze benchmark gebruikt kan worden bij de beoordeling van de resultaten van een manager. Veronderstel dat gegeven is een reeks portefeuille returns  $R_1^P, \dots, R_n^P$  en benchmark returns  $R_1^M, \dots, R_n^M$ , over  $n$  subperiodes (bijvoorbeeld maanden, kwartalen of jaren). De **toegevoegde waarde** (ook genoemd **value added** of **excess return**)  $VA_i$  is een maat voor de hoeveelheid rendement dat boven het benchmark rendement uitkomt over subperiode  $i$ . Deze kan **aritmetisch** worden berekend:

$$VA_i := R_i^P - R_i^M,$$

maar ook **geometrisch**:

$$VA_i^{\text{geom}} := \frac{1 + R_i^P}{1 + R_i^M} - 1.$$

Welke van de twee beter is, is een oneindige discussie. De aritmetische value added heeft additieve eigenschappen. De geometrische value added kan worden samengesteld over meerdere periodes. In de praktijk worden alle twee de varianten gebruikt, waarbij geen van beide domineert.

We kunnen de reeks  $VA_1, \dots, VA_n$  statistisch onderzoeken om te bepalen of de manager vaardig is. Onder “vaardig” verstaan we dat de manager in staat is om de benchmark voldoende vaak en voldoende ruim te verslaan zodat zijn value added met een zekere betrouwbaarheid niet door puur geluk tot stand had kunnen komen. Dit kunnen we onderzoeken middels een **t-toets**. Laat de stochast  $VA$  een value added over een willekeurige subperiode voorstellen. We nemen eerst aan dat deze normaal verdeeld is en dat de waarnemingen  $VA_1, \dots, VA_n$  stochastisch onafhankelijk zijn. De hypothesen van de toets zijn:

$$H_0 : \mathbb{E}VA \leq 0;$$

$$H_1 : \mathbb{E}VA > 0.$$

De **t-statistiek** is gedefinieerd door

$$T_n := \sqrt{n} \frac{\overline{VA}_n}{\sigma_{VA,n}},$$

met,

$$\begin{aligned}\overline{VA}_n &:= \text{het gemiddelde van de } VA_1, \dots, VA_n; \\ \sigma_{VA,n} &:= \text{de standaarddeviatie van de } VA_1, \dots, VA_n.\end{aligned}$$

De nulhypothese wordt verworpen voor grote waarden van  $T_n$ . Bekend is dat  $T_n$ , in het grensgeval  $\mathbb{E}VA=0$  tussen beide hypothesen, de t-verdeling heeft met  $n-1$  vrijheidsgraden. Dus bij een onbetrouwbaarheidsdrempel  $\alpha$  wordt  $H_0$  verworpen als voor de realisatie  $t_n$  van  $T_n$  geldt dat

$$t_n > F_{t_{n-1}}^{-1}(1 - \alpha),$$

met  $F_{t_{n-1}}^{-1}$  de inverse verdelingsfunctie van de  $t_{n-1}$  verdeling. Dit is equivalent met de voorwaarde dat voor de overschrijdingskans geldt

$$\mathbb{P}_{\mathbb{E}VA=0}(T_n > t_n) \leq \alpha.$$

Ingeval de nulhypothese wordt verworpen heeft de manager genoeg outperformance geleverd, zodat we met betrouwbaarheid  $1 - \alpha$  kunnen zeggen dat de manager talent heeft om de markt te overtreffen met zijn beleggingen.

Merk echter op dat deze toets alleen betrouwbare resultaten geeft ingeval de aannames van onafhankelijkheid en normale verdeeldheid geoorloofd zijn. In veel gevallen vertoont een rendement (of value added) in een subperiode afhankelijkheid van voorgaande subperiodes. Deze afhankelijkheid is echter kleiner naarmate de subperiodes verder in het verleden liggen, waardoor voor grote  $n$  de waarnemingen wel bij benadering onafhankelijk kunnen worden verondersteld.

Een groter probleem is de normaliteitsaanname. Rendementen op zich zijn lang niet altijd normaal verdeeld. Met name bij beleggingsstrategieën waarbij derivaten voor een hefboomwerking zorgen, kunnen returnverdelingen met dikke staarten ontstaan. Dit probleem kan worden opgelost door gebruik te maken van verdelingsvrije methoden. Als  $n$  groot is, dan is de toetsingsgrootte bij benadering standaardnormaal verdeeld, vanwege de centrale limietstelling. We kunnen dan de kwantielen van de t-verdeling vervangen door die van de normale verdeling.

Er is echter een probleem dat ook niet wordt opgelost door verdelingsvrije methoden. Dit is het hoofdprobleem bij het gebruik van benchmark methodes:

*Statistische significantie moet over de tijd worden verkregen!*

Dat wil zeggen, we kunnen de performance pas evalueren zodra het aantal waarnemingen  $n$  groot genoeg is om betrouwbare conclusies uit de statistische toetsen te kunnen verkrijgen. Werken we met maandelijkse data, dan komt er elke maand één waarneming bij, en dus duurt het jaren voordat we op statistische gronden een betrouwbare performance uitspraak kunnen doen. In veel gevallen wijzigt de samenstelling van het management team over de tijd zodat de performance van het huidige team afhangt van de inbreng van personen die reeds vertrokken zijn. Nog erger wordt dit wanneer we de value added per kwartaal of per jaar meten!

Naast dit grote probleem zijn er nog enkele andere nadelen te noemen bij het gebruik van benchmarks:

- Veel benchmarks houden geen rekening met inkomen (dividend, coupon rente, etc.). Er bestaan echter zogeheten **total return indices** die dit wel doen. Dit probleem is dus in principe overkomelijk.
- Echte portefeuilles zijn onderhevig aan transactiekosten, management fees en belasting. Benchmarks verwaarlozen deze invloeden. Om benchmarks hiervoor te corrigeren moeten deze invloeden geschat kunnen worden.
- Benchmarks houden geen rekening met specifieke mandaateisen zoals grenzen op exposure gewichten, en risicomaten. Weliswaar voldoet een benchmark zelf aan deze eisen, maar dezelfde benchmark zou ook voldoen aan een vergelijkbaar mandaat met minder strikte eisen. Zo'n mandaat geeft de belegger meer mogelijkheden, terwijl dat geen effect heeft op het rendement van de benchmark.

## Peer groups

Een **peer group** of **referentie universum** is een groep van managers, die soortgelijke portefeuilles beheren en gedurende dezelfde tijdperiode opereren. Wat “soortgelijk” in dit verband precies betekent blijft vaag. In ieder geval is het de bedoeling dat de performance van een manager wordt geëvalueerd door deze te vergelijken met die van de andere managers uit een passende peer group. De beleggingsgegevens van dergelijke managers worden vaak verkregen uit historische databases, beheerd door instellingen zoals Morningstar, Wilshire en WM Company.

Peer group analyses zijn populair bij beleidsmakers, omdat er bekeken wordt hoe een manager presteert ten opzichte van zijn concurrenten. Dit idee botst enigszins met criterium 2 uit paragraaf 2.1, in de zin dat performance niet wordt geëvalueerd aan de hand van de mogelijkheden die de manager zelf heeft gehad; de performance is afhankelijk van hoe de concurrentie heeft gepresteerd. Desalniettemin bespreken we peer groups omdat veel ideeën worden overgenomen in de alternatieve methode van PODs.

Veronderstel dat gegeven is een portefeuille  $P$  met een gerealiseerd rendement  $R^P$ , en  $m$  peer group portefeuilles  $P_1, \dots, P_m$  met gerealiseerde rendementen  $R^1, \dots, R^m$ . Bij een peer group studie wordt er geanalyseerd hoe  $R^P$  uitvalt ten opzichte van de dataset  $R^1, \dots, R^m$ . Een mogelijke uitkomst is de positie van  $P$  binnen de peer group:

$$positie(P) := \#\{i = 1, \dots, m : R^i > R^P\}.$$

Deze positie wordt gebruikt in de definitie van de **percentiel-ranking**:

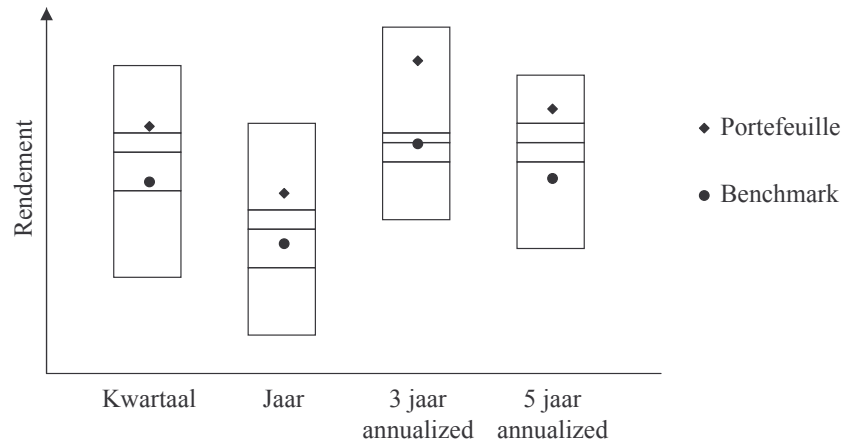
$$ranking_{100}(P) := \left\lceil \frac{100 \cdot positie(P)}{m + 1} \right\rceil,$$

waarbij  $\lceil \cdot \rceil$  staat voor afronding naar boven op gehelen. Dit getal ligt in de verzameling  $\{1, \dots, 100\}$ . Is de peer group verdeeld in 100 subgroepen op basis van posities van de peers, dan geeft de percentiel-ranking aan in welke subgroep  $P$  zich bevindt. Een ranking van 1 is het beste, een ranking van 100 is het slechtste.

Het verdelen van de peers in 100 groepen is alleen zinvol als het aantal observaties  $m$  minstens van dezelfde orde van grootte is. In plaats van 100 kan echter ieder aantal worden gebruikt. Een veelgebruikt aantal is 4; we spreken dan van een **kwartiel-ranking**:

$$ranking_4(P) := \left\lceil \frac{4 \cdot positie(P)}{m+1} \right\rceil.$$

Kwartielen worden vaak grafisch weergegeven in een zogeheten **floating bar chart**. Een voorbeeld is in figuur 4 afgebeeld.



Figuur 4: Floating bar chart.

Zoals we zien representeert een verticale “box” het bereik van de rendementen van de peer group, en er is aangegeven waar  $R^P$  ligt. Vijf horizontale lijnen bevinden zich op de hoogte van:

- de grootste return (of bijv. het 5% percentiel, afhankelijk van het aantal observaties);
- het 1-ste kwartiel;
- de mediaan (2-de kwartiel);
- het 3-de kwartiel;
- de kleinste return (of bijv. het 95% percentiel, afhankelijk van het aantal observaties).

De eerder gedefinieerde kwartiel-ranking geeft aan tussen welke lijnen  $R^P$  ligt. Veelal worden in een floating bar chart de gemiddelde resultaten over periodes van verschillende lengte gegeven, zoals hierboven. In het voorbeeld is tevens het gemiddelde van een benchmark opgenomen.

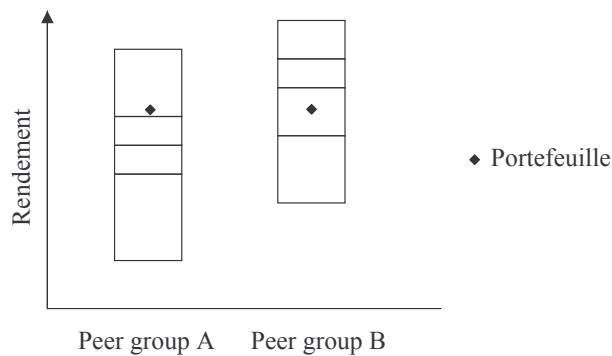
Overigens kunnen behalve rendementen ook andere performance statistieken in een peer group analyse worden vergeleken. Bijvoorbeeld risicogecorrigeerde statistieken die in de volgende paragraaf ter sprake komen.

Helaas zijn peer group analyses onderhevig aan een aantal onzuiverheden, die we nu zullen bespreken.

- **Classification bias** – Dit ontstaat doordat managers worden toegewezen aan één bepaalde beleggingsclassificatie. Veel managers zijn gebonden aan een unieke set van portefeuille constraints, zodat het misleidend is om ze in te delen in één specifieke categorie. Ook kunnen er, zelfs bij een redelijk vergelijkbaar mandaat, kleine verschillen zijn waardoor managers niet eerlijk te vergelijken zijn. Zo kan het voorkomen dat een bepaald aandeel niet in het mandaat van de manager valt, maar wel in dat van de meeste van zijn peers. Als toevallig dat aandeel een zeer hoog rendement heeft behaald over de periode, dan valt

mogelijk de ranking van de manager binnen zijn peer group erg laag uit, terwijl dat onterecht is.

- **Composition bias** – De oorzaak hiervan is dat voor bepaalde mandaten geen database van voldoende grootte bestaat. Het aantal vergelijkbare portefeuilles dat verkregen wordt is derhalve klein, hetgeen een betrouwbare statistische analyse onmogelijk maakt. Het gevolg is dat peer groups van verschillende data providers resulteren in een verschillende ranking en dus in verschillende performance (zie figuur 5 hieronder). Om toch statistische significantie te verkrijgen moeten (net als bij benchmarks) de resultaten gemeten worden over een groot aantal jaren, wat in de praktijk meestal onmogelijk is. Daarnaast is het een gegeven dat managers zelf mogen besluiten welke resultaten zij bijdragen aan een database. Niet zelden worden de slechte resultaten achtergehouden, zodat de database geen correcte weerspiegeling is van de werkelijkheid.



Figuur 5: Succes in peer group A, mislukking in peer group B.

Merk op dat classification bias en composition bias elkaar tegenwerken. Immers, maken we het universum groter, dan zijn de portefeuilles minder goed vergelijkbaar (classification bias). Selecteren we mandaten die beter te vergelijken zijn met die van de te evalueren manager, dan wordt het universum kleiner (composition bias).

- **Survivorship bias** – Dit is het belangrijkste en tevens bekendste probleem bij de peer group methode. Het treedt op wanneer portefeuilles die worden opgeheven (meestal door underperformance) verdwijnen uit de database en dus niet voorkomen in een peer group, terwijl dat wel zou moeten. Door dit verschijnsel wordt de data onzuiverder naarmate de evaluatieperiode langer wordt. Immers, hoe groter het tijdsinterval, hoe meer portefeuilles binnen dat tijdsinterval worden opgeheven. Dit resulteert in een overschatting van de peer group performance en dus in een onderschatting van de manager z'n performance. R. Surz vergelijkt het verschijnsel in [11] met een "marathon-analogie": als bij een marathon met 1000 atleten slechts 100 de finish halen, is de 100-ste dan de laatste? Of in de top 10%?

We zien dus dat peer group analyses door alle onzuiverheden vaak onbetrouwbare performance informatie leveren. Wel worden in vergelijking met benchmarks de nadelen van peer groups tegenwoordig meer erkend. Met name de gevolgen van survivorship bias zijn algemeen bekend bij PE professionals. Peer groups worden dan ook niet zo vaak meer gebruikt.

In het vervolg van dit hoofdstuk komen peer groups niet meer ter sprake. Benchmarks komen nog wel terug in de komende paragrafen. We zien hoe deze gebruikt worden bij risicometing en performance attributie.

## 2.4 Risicometing en correctie

**Risico** of **risk** wordt wel gedefinieerd als de mate van onzekerheid in de uitkomst van een experiment. Een voorbeeld is de (gedeeltelijk) onvoorspelbare variatie in rendementen op marktinstrumenten, waardoor ook rendementen van beleggingsportefeuilles onderhevig zijn aan risico. Hoe hoog dit risico is verschilt sterk per portefeuille. In het algemeen kunnen we het volgende aannemen:

- Beleggers zijn **risicomijdend (risk averse)**: gegeven een aantal portefeuilles met gelijk rendement wordt de voorkeur gegeven aan de portefeuille met het laagste risico.
- Hoe hoger het genomen risico, hoe hoger het rendement dat een manager kán behalen. Portefeuillemanagers worden daarom wel **risk managers** genoemd: ze worden betaald om risico te nemen en op de juiste manier met dit risico om te gaan zodat het behaalde rendement voldoet aan de gestelde doelen.

Risico wordt op twee manieren geëvalueerd. **Ex-ante risico** is een schatting of voorspelling van het toekomstige risico op een portefeuille, gegeven de huidige samenstelling. Managers zijn in hun mandaat vaak beperkt tot portefeuilles met ten hoogste een bepaald ex ante risico. Gedurende de meetperiode mag dit risico niet overschreden worden.

Bij performance evaluatie wordt echter gekeken naar prestaties in het verleden. **Ex-post risico**, ofwel **historisch risico**, is het risico waaraan de portefeuille, over een bepaalde periode in het verleden, onderhevig is geweest. We bespreken een aantal bekende ex post risicomaten.

### Risicomaten

Gegeven een reeks behaalde returns  $R_1, \dots, R_n$  over  $n$  subperiodes (bijvoorbeeld maandelijks of per kwartaal), wordt het ex post risico gemeten met de **variantie** van de returns<sup>2</sup>:

$$\sigma_{R,n}^2 := \frac{1}{n} \sum_{i=1}^n (R_i - \bar{R}_n)^2.$$

Is de stochast  $R$  een willekeurige subperiodieke return, dan is  $\sigma_{R,n}^2$  een schatter voor de variantie  $\sigma_R^2$  van  $R$ . Meestal werken we echter met niet-gekwadrateerde eenheden en dus met de **standaarddeviatie**, ook wel **volatiliteit** genoemd:

$$\sigma_{R,n} := \sqrt{\frac{1}{n} \sum_{i=1}^n (R_i - \bar{R}_n)^2}.$$

Het gemiddelde en de standaarddeviatie zijn twee basis kengetallen waarmee we analyseren hoe een portefeuille heeft geredeneerd ten opzichte van het genomen risico. Deze kengetallen kunnen ook worden geschat over een jaar met betrekking tot de schatting over een deel van een jaar. Dit geeft het **annualized gemiddelde**

$$\bar{R}_n^{\text{ann}} := p \cdot \bar{R}_n,$$

en het **annualized risico**

---

<sup>2</sup> Merk op dat door  $n$  wordt gedeeld in plaats van door  $n-1$ ; dit lijkt gewoonte te zijn voor de meerderheid van de performance analisten. Delen door  $n-1$  zou een zuivere schatter opleveren, maar voor grote  $n$  is het verschil tussen beide alternatieven klein.

$$\sigma_{R,n}^{\text{ann}} := \sqrt{p} \cdot \sigma_{R,n},$$

met  $p$  de periodiciteit (bijvoorbeeld  $p = 12$  bij maandelijksse returns).

Vaak hebben portefeuillemanagers als opdracht om een benchmark min of meer te volgen en door kleine afwijkingen een outperformance te behalen. De portefeuille wordt dan als meer risicovol beschouwd naarmate het rendement sterker fluctueert ten opzichte van de benchmark. Dit kan, gegeven een reeks portefeuille returns  $R_1^P, \dots, R_n^P$  en benchmark returns  $R_1^M, \dots, R_n^M$ , worden gemeten met de **correlatiecoëfficiënt**

$$\rho_n^{P,M} := \frac{\frac{1}{n} \sum_{i=1}^n (R_i^P - \bar{R}_n^P)(R_i^M - \bar{R}_n^M)}{\sigma_{R,n}^P \cdot \sigma_{R,n}^M}.$$

Deze grootheid ligt binnen het interval  $[-1,1]$  en kan als volgt worden geïnterpreteerd:

- $\rho_n^{P,M}$  in de buurt van 1: fonds en benchmark bewegen in gelijke richting;
- $\rho_n^{P,M}$  in de buurt van 0: fonds en benchmark zijn niet gerelateerd;
- $\rho_n^{P,M}$  in de buurt van -1: fonds en benchmark bewegen in omgekeerde richting.

Het kwadraat  $(\rho_n^{P,M})^2$  van de correlatie valt in  $[0,1]$ , en wordt wel de **determinatiecoëfficiënt** genoemd. Deze elimineert het effect dat fonds en benchmark zowel positief als negatief gecorreleerd kunnen zijn. We interpreteren de determinatiecoëfficiënt als de fractie variatie in de portefeuille returns dat verklaard kan worden door variatie in de benchmark returns.

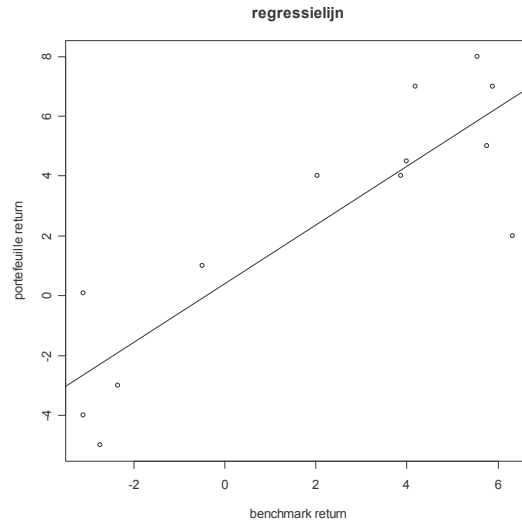
Wat de correlatie en de determinatiecoëfficiënt niet weergeven is de *graad* waarin de portefeuille en de benchmark samenlopen. Een fonds kan bijvoorbeeld tegelijk dalen en stijgen met de benchmark, maar grotere sprongen maken dan de benchmark. De graad van samenhang onderzoeken we aan de hand van de volgende **lineaire regressie** tussen fonds en benchmark return:

$$R^P = \alpha + \beta \cdot R^M + \varepsilon,$$

waarbij,

- $\alpha$  := x-as intercept van de regressielijn;
- $\beta$  := helling van de regressielijn;
- $\varepsilon$  := regressiefout, per aanname normaal verdeeld.

Onderstaande figuur 6 geeft een voorbeeld van een regressielijn die uit een reeks fonds en benchmark rendementen ontstaat.



**Figuur 6: Regressielijn.**

Passen we kleinste kwadraten regressie toe, dan is uit de statistiek bekend dat  $\beta$ , de **regressie bèta**, wordt geschat door

$$\beta_n := \frac{\sum_{i=1}^n (R_i^P - \bar{R}_n^P)(R_i^M - \bar{R}_n^M)}{\sum_{i=1}^n (R_i^M - \bar{R}_n^M)^2} = \rho_n^{P,M} \frac{\sigma_{R,n}^P}{\sigma_{R,n}^M}.$$

De interpretatie van de  $\beta$  is als volgt:

- $\beta > 1$ : fonds varieert evenredig met de benchmark, in grotere proportie;
- $\beta \approx 1$ : fonds varieert evenredig met de benchmark, in even grote proportie;
- $\beta \in (0,1)$ : fonds varieert evenredig met de benchmark, in kleinere proportie;
- $\beta \approx 0$ : variatie in het fonds is niet gerelateerd aan variatie in de benchmark;
- $\beta < 0$ : fonds varieert omgekeerd evenredig met de benchmark.

Is  $\beta_n$  berekend, dan kan de **regressie alfa** worden geschat door

$$\alpha_n := \bar{R}_n^P - \beta_n \cdot \bar{R}_n^M.$$

Deze alfa heeft als interpretatie hoeveel waarde er is toegevoegd ten opzichte van de benchmark, rekeninghoudend met de gevoeligheid van het fonds voor de benchmark.

## Het CAPM

Formules voor risico, zoals de standaarddeviatie en de regressie bèta, kunnen worden toegepast bij de berekening van **risicogecorrigeerde rendementen** (ofwel **risk-adjusted returns**). Dit zijn performance maatstaven die weergeven hoeveel rendement er is behaald ten opzichte van het genomen risico. In het onderstaande bespreken we een aantal voorbeelden van zulke maatstaven, die voortkomen uit het zogeheten **Capital Asset Pricing Model (CAPM)**.

Het CAPM is een theoretisch model voor het gedrag van de markt, dat ontwikkeld is omstreeks 1964 door W. Sharpe e.a.. Laatstgenoemde won voor zijn werk de Nobelprijs in

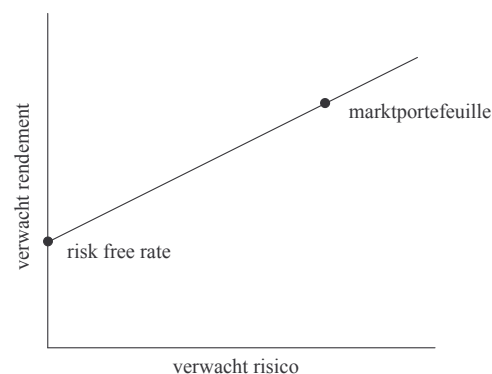


1990. Het CAPM is gebaseerd op de **moderne portefeuilletheorie**, ontwikkeld in 1956 door Markowitz, eveneens Nobelprijswinnaar. Eén van de resultaten van de moderne portefeuilletheorie is dat het risico kan worden verkleind door spreiding van het belegde kapitaal over meerdere instrumenten en beleggingscategorieën.

Het CAPM gaat uit van een aantal veronderstellingen, waaronder:

- Er bestaat een **risicovrij rendement** (ook genoemd **risk-free rate**). Iedere belegging zonder risico verdient dit rendement. (Hierbij is risico gedefinieerd door de volatiliteit.)
- Er bestaat een **marktportefeuille** die is opgebouwd uit alle op de markt aanwezige instrumenten, gewogen naar totale marktkapitalisatie. Dit kan bijvoorbeeld een benchmark zijn.
- Beleggers hebben, bij een gegeven verwacht rendement, voorkeur voor een portefeuille met een zo laag mogelijk verwacht risico. Een equivalente formulering is dat beleggers bij gegeven verwacht risico, een portefeuille zoeken met een zo hoog mogelijk verwacht rendement. Zulke portefeuilles heten **efficiënte portefeuilles**, en de verzameling van al deze portefeuilles heet de **efficient frontier**.
- De totale variatie in de returns wordt veroorzaakt door twee typen risico. Het eerste type is **systematisch risico** of **marktrisico**. Dit weerspiegelt de graad waarin de portefeuille varieert met de prijzen in de onderliggende markt. Het zijn dus de factoren die de markt als geheel beïnvloeden.
- Alle beleggingen binnen de markt (behalve de risicovrije belegging) zijn beïnvloed door systematisch risico, maar de graad van onderhevigheid verschilt per instrument.
- Het tweede type risico is **onsystematisch risico** of **uniek risico**. Dit is de volatiliteit die specifiek is voor de portefeuille.
- Onsystematisch risico kan worden geëlimineerd door portefeuillespreiding, zodat alleen het marktrisico overblijft, hetgeen door geen enkele belegger weg kan worden gespreid.
- De marktportefeuille is zodanig gespreid dat het alleen systematisch risico bevat. De marktportefeuille is derhalve efficiënt.
- Omdat onsystematisch risico geëlimineerd kan worden zal de markt dit (in evenwichtssituatie) niet belonen. De markt beloont alleen onderhevigheid aan *systematisch* risico en wel met een rendement; een zogeheten **risk premium**. Deze beloning is gelijk aan de risk-free rate plus een stijgende functie van het systematisch risico.

Uit het CAPM volgt dat de efficient frontier bestaat uit alle lineaire combinaties van de marktportefeuille en het risicovrije fonds. Immers, de marktportefeuille en het risicovrije fonds zijn beide efficiënt, en daarmee ook alle lineaire combinaties. Dit resultaat kan worden gevisualiseerd door een rechte lijn in het vlak dat de verwachte return van een portefeuille uitzet tegen het verwachte risico. Deze lijn heet de **Capital Market Line (CML)** en snijdt de twee punten die het risicovrije fonds en de marktportefeuille representeren. Zie figuur 7.



Figuur 7: Capital Market Line.

Ligt de portefeuille van een manager boven de CML, dan geeft dat een indicatie dat de manager goede voorspelkwaliteiten bezit. Om dit te meten bestaan er indicatoren die aangeven hoe ver een portefeuille boven of onder de CML ligt. De **Sharpe-ratio** van een portefeuille is de helling van de lijn door de portefeuille en het risicovrije fonds:

$$SR := \frac{\mathbb{E}R^P - R^F}{\sigma_R^P},$$

waarbij,

$\mathbb{E}R^P$  := de verwachte return van de portefeuille;

$R^F$  := het risicovrije rendement;

$\sigma_R^P$  := de standaarddeviatie van de return van de portefeuille.

Gegeven een reeks subperiodieke portefeuille returns  $R_1^P, \dots, R_n^P$  kunnen we de Sharpe-ratio schatten door

$$SR_n := \frac{\bar{R}_n^P - R^F}{\sigma_{R,n}^P},$$

of door

$$SR_n := \frac{\bar{R}_n^P - \bar{R}_n^F}{\sigma_{R,n}^P},$$

ingeval het risicovrije rendement (licht) varieert over de subperiodes en gegeven is door een reeks  $R_1^F, \dots, R_n^F$ . De interpretatie van  $SR_n$  is hoeveel rendement gemiddeld is behaald boven de risicovrije rente, per eenheid genomen risico.

Er bestaan nog meer risicocorrigerende performance maatstaven die in zekere zin varianten zijn van de Sharpe-ratio. Daartoe schatten we eerst het systematisch risico onder de aanname dat de marktportefeuille gerepresenteerd wordt door een benchmark. We breiden het regressiemodel als volgt uit<sup>3</sup>:

$$R^P - R^F = \alpha + \beta(R^M - R^F) + \varepsilon,$$

waarbij weer

$\alpha$  := x-as intercept van de regressielijn;

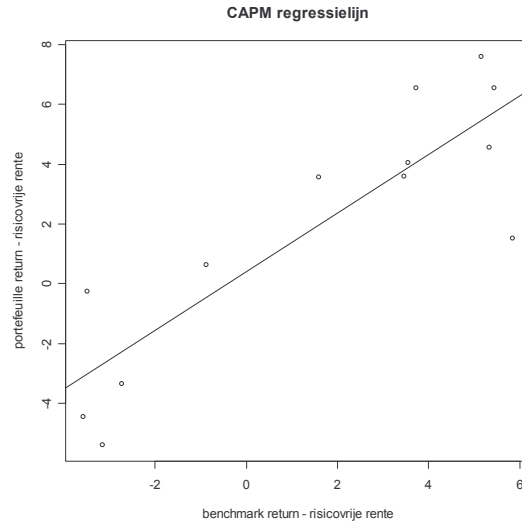
$\beta$  := helling van de regressielijn;

$\varepsilon$  := regressiefout, per aanname normaal verdeeld.

Zie figuur 8 voor een voorbeeld.

---

<sup>3</sup> Dit is alleen een uitbreiding als de risicovrije rente (licht) varieert over de subperiodes. Varieert deze niet dan is het CAPM regressiemodel equivalent met het hiervoor beschreven regressiemodel.



**Figuur 8: CAPM regressielijn.**

De  $\beta$ , die in dit verband de **CAPM bèta** wordt genoemd, representeert de marktgevoeligheid, dat wil zeggen de graad waarin de portefeuille onderhevig is aan systematisch risico. Analoog aan de formule voor de regressie bèta geldt:

$$\beta_n := \frac{\sum_{i=1}^n [(R_i^P - \bar{R}_n^P) - (R_i^F - \bar{R}_n^F)] [(R_i^M - \bar{R}_n^M) - (R_i^F - \bar{R}_n^F)]}{\sum_{i=1}^n [(R_i^M - \bar{R}_n^M) - (R_i^F - \bar{R}_n^F)]^2}.$$

Definiëren we

- $\sigma_{R,n}^{P-F}$  := spreiding van de verschillen tussen portefeuille return en risicovrije rente;
- $\sigma_{R,n}^{M-F}$  := spreiding van de verschillen tussen benchmark return en risicovrije rente;
- $\rho_n^{P-F, M-F}$  := correlatie van beide verschillen,

dan is de schatter voor bèta te schrijven als

$$\beta_n = \rho_n^{P-F, M-F} \frac{\sigma_{R,n}^{P-F}}{\sigma_{R,n}^{M-F}}.$$

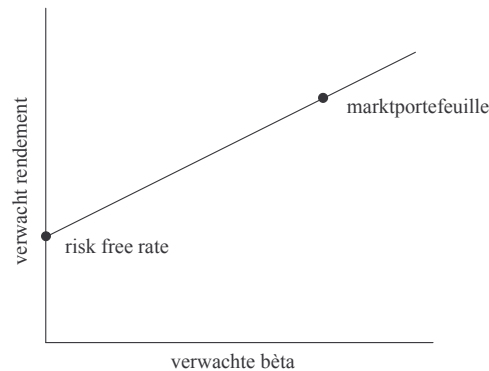
De interpretatie van de CAPM bèta is ongeveer hetzelfde als de regressie bèta (met als enige verschil de aanwezigheid van de risicovrije rente). Het geeft aan hoe sterk het verloop van het portefeuillerendement afhankelijk is van dat van de benchmark. Daarmee is bèta een maat voor de hoeveelheid systematisch risico.

Een beroemde toepassing van de CAPM bèta is de **Treynor ratio**:

$$TR_n := \frac{\bar{R}_n^P - \bar{R}_n^F}{\beta_n}.$$

Dit is de Sharpe ratio waarbij het totale risico is vervangen door het systematisch risico, oftewel de standaarddeviatie is vervangen door bèta. De **Security Market Line (SML)**

representeert die portefeuilles die, gegeven een verwachte hoeveelheid systematisch risico, het grootste verwachte rendement hebben. Zie figuur 9.



**Figuur 9: Security Market Line.**

De CML visualiseert de Sharpe ratio; de SML doet dit voor de Treynor ratio.

De alfa van het regressiemodel wordt wel **Jensen's alfa** genoemd. Deze kunnen we schatten door

$$\alpha_n := (\bar{R}_n^P - \bar{R}_n^F) - \beta_n (\bar{R}_n^M - \bar{R}_n^F).$$

Jensen's alfa is een belangrijke performance maatstaf. Een positieve alfa betekent dat het fonds betere performance heeft geleverd dan het CAPM zou voorspellen. Bij een negatieve alfa is dat juist slechter.

### Downside risico

De standaarddeviatie  $\sigma_{R,n}$  is in sommige situaties niet de ideale maat voor risico. De standaarddeviatie meet namelijk afwijkingen van rendementen beneden én boven het gemiddelde, terwijl vaak alleen *lage* rendementen als risicovol worden ervaren. Met "laag" wordt in zo'n geval bedoeld dat een rendement lager uitvalt dan een vooraf vastgestelde targetreturn, de **Minimum Acceptabele Return (MAR)**, genoteerd door  $R^{\text{MAR}}$ . Het **downside risico** meet de afwijking van de MAR en rekt daarbij alleen rendementen mee die onder de MAR liggen:

$$\sigma_{R,n}^{P,\text{MAR}} := \sqrt{\frac{1}{n} \sum_{\substack{i=1,\dots,n \\ R_i < R^{\text{MAR}}}} (R_i^P - R^{\text{MAR}})^2}.$$

Vervanging van de standaarddeviatie door het downside risico is met name gewenst wanneer de rendementverdeling niet normaal is, maar bijvoorbeeld naar beneden begrensd met een rechterstaart. De waarde van de standaarddeviatie is dan hoger dan het risico dat werkelijk gevoeld wordt.

Op de hierboven besproken Sharpe ratio bestaat een variant voor gevallen waarbij downside risico de relevante risicomaat is; de zogeheten **Sortino ratio**:

$$\text{Sort}R_n := \frac{\bar{R}_n^P - R^{\text{MAR}}}{\sigma_{R,n}^{P,\text{MAR}}}.$$

Dit is de Sharpe ratio met in de noemer de standaarddeviatie vervangen door het downside risico, en in de teller het verschil berekend met  $R^{\text{MAR}}$  in plaats van  $R^F$ .

## Relatief risico

**Relatief risico** is het risico dat een belegger loopt ten opzichte van zijn benchmark. Een portefeuille wordt in dit verband als risicovol gezien wanneer deze niet evenredig met de benchmark beweegt. Zoals we al gezien hebben kan dit risico worden gemeten met de correlatie, de determinatiecoëfficiënt en de bèta. Een andere veelgebruikte maat voor benchmark relatief risico is de zogeheten **tracking error** of **active risk**. Dit is de standaarddeviatie van de value added ten opzichte van de benchmark:

$$TE_n := \sqrt{\frac{1}{n} \sum_{i=1}^n (VA_i - \overline{VA_n})^2}.$$

Het is na te gaan dat

$$TE_n = \sigma_{R,n}^P \sqrt{1 - (\rho_n^{P,M})^2}.$$

In tegenstelling tot de correlatie, de determinatiecoëfficiënt en de bèta, stijgt deze grootte naarmate de rendementen van het fonds meer variatie vertonen ten opzichte van de benchmark. Deze eigenschap maakt de tracking error een geschikte maat voor relatief risico.

De **informatie ratio**, gedefinieerd door

$$IR_n := \frac{\overline{R}_n^P - \overline{R}_n^M}{TE_n},$$

geeft weer hoeveel value added is behaald ten opzichte van de benchmark, per eenheid relatief risico.

Ook de information ratio is qua vorm een variant op de Sharpe ratio. Ieder van deze ratio's vormt, voor een specifieke situatie, een maat voor het risicogecorrigeerde rendement.

## 2.5 Performance attributie

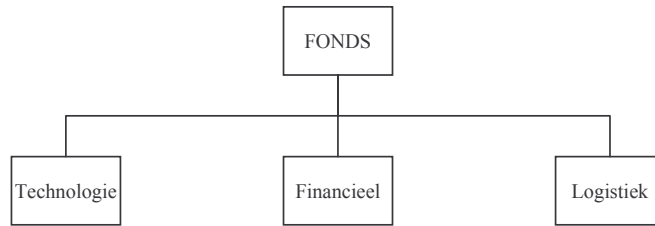
Onder **performance attributie** wordt verstaan een reeks decompositietechnieken waarbij de behaalde value added (ten opzichte van een benchmark) van een belegging wordt opgesplitst in termen. Iedere term representeert een bijdrage van een bepaalde managementbeslissing. Het doel van attributie is om te achterhalen welke beslissingen positief of negatief hebben bijgedragen aan de totale value added, en in welke mate.

Op het gebied van performance attributie is de laatste decennia veel onderzoek gedaan. Er bestaan dan ook veel verschillende attributie modellen. Welk model het beste gebruikt kan worden hangt af van de opzet van de beleggingsstrategie. Wij beperken ons in deze scriptie tot twee eenvoudige modellen. Deze modellen zijn vooral nuttig voor portefeuilles die in de aandelenmarkt beleggen. Attributie van bijvoorbeeld vastrentende beleggingen is een vakgebied op zich.

## Brinson en Fachler

Het attributiemodel van **Brinson en Fachler** (1985) is wellicht het bekendste en eenvoudigste model. Het is een **aritmetisch** (ofwel **additief**) attributiemodel, dat wil zeggen dat de som van de berekende termen gelijk is aan de aritmetische value added. Er bestaan ook **geometrische** attributiemodellen, waarbij door chain-linking van de termen de geometrische value added wordt verkregen. We laten deze modellen verder buiten beschouwing.

Veronderstel dat een portefeuillemanager een kapitaal belegt in  $N$  segmenten, bijvoorbeeld in drie sectoren, zoals in het voorbeeld hieronder in figuur 10. De afzonderlijke segmenten worden eventueel beheerd door andere managers.



Figuur 10: Voorbeeld allocatie.

De benchmark is op dezelfde manier opgebouwd. We gebruiken de volgende notaties:

- $w^{P,j}$  := het exposure gewicht van segment  $j$  van de portefeuille;
- $w^{M,j}$  := het exposure gewicht van segment  $j$  van de benchmark;
- $R^{P,j}$  := het rendement van segment  $j$  van de portefeuille;
- $R^{M,j}$  := het rendement van segment  $j$  van de benchmark,

voor  $j = 1, \dots, N$ .

De value added van de portefeuille kan als volgt worden ontbonden:

$$\begin{aligned}
 VA &= \sum_{j=1}^N w^{P,j} R^{P,j} - w^{M,j} R^{M,j} \\
 &= \sum_{j=1}^N w^{P,j} R^{M,j} + w^{M,j} R^{P,j} - 2w^{M,j} R^{M,j} + w^{P,j} (R^{P,j} - R^{M,j}) - w^{M,j} (R^{P,j} - R^{M,j}) \\
 &= \sum_{j=1}^N \underbrace{(w^{P,j} - w^{M,j}) R^{M,j}}_I + \underbrace{w^{M,j} (R^{P,j} - R^{M,j})}_II + \underbrace{(w^{P,j} - w^{M,j}) (R^{P,j} - R^{M,j})}_III.
 \end{aligned}$$

Het **allocatie-effect** of **timing-effect** van segment  $j$  is gegeven door term I. Deze term geeft aan hoe veel de afwijking van de benchmark in het gewicht van segment  $j$  heeft bijgedragen aan de totale value added.

Het **selectie-effect** van segment  $j$  is gegeven door term II. Deze term geeft aan hoe veel de afwijking in de selectie van de marktinstrumenten in segment  $j$  heeft bijgedragen aan de totale value added.

Ingeval de strategie van de manager een combinatie is van asset allocatie en security selectie, dan treedt er een interactie op tussen deze beslissingen. Het **interactie-effect** van segment  $j$

is gegeven door term III. Normaal gesproken is deze bijdrage klein ten opzichte van het allocatie- en selectie-effect. Grote bijdragen geven een indicatie dat de benchmark ongeschikt is bij de betreffende strategie. In de praktijk wordt het interactie-effect vaak verdeeld over het allocatie- en selectie-effect, zodat er per segment twee termen van de attributie zijn gedefinieerd.

Beschouw ter illustratie het volgende voorbeeld. Onderstaande tabel 1 geeft per segment de exposure gewichten en rendementen van zowel de portefeuille als de benchmark. De benchmark is verslagen met een totale value added van 1.04%.

Tabel 1: Gewichten en rendement per segment.

Sector	Gewicht (%)			Rendement (%)		
	Portefeuille	Benchmark	Vershil	Portefeuille	Benchmark	Vershil
Technologie	40	35	5	12.00	10.00	2.00
Financieel	30	35	-5	-3.00	-2.00	-1.00
Logistiek	30	30	0	-1.20	-1.00	-0.20
<i>Totaal</i>	<i>100</i>	<i>100</i>	<i>0</i>	<i>3.54</i>	<i>2.50</i>	<i>1.04</i>

Toepassing van de hierboven gegeven formules geeft de value added decompositie voor dit voorbeeld in tabel 2:

Tabel 2: Value added decompositie.

Sector	Allocatie (%)	Selectie (%)	Interactie (%)	Totaal (%)
Technologie	0.50	0.70	0.10	1.30
Financieel	0.10	-0.35	0.05	-0.20
Logistiek	0	-0.06	0	-0.06
<i>Totaal</i>	<i>0.60</i>	<i>0.29</i>	<i>0.15</i>	<i>1.04</i>

Aan de hand van de uitkomsten merken we het volgende op:

- Zowel aan de allocatie als aan de selectie heeft de technologie sector het meest bijgedragen. De overige twee sectoren hebben per saldo alleen negatief bijgedragen.
- De financiële sector heeft een negatief selectie-effect (de portefeuille behaalde een lager rendement dan de benchmark in deze sector), maar een positief allocatie-effect vanwege de onderweging ten opzichte van de benchmark (het rendement was negatief).
- De logistieke sector heeft geen allocatie effect, omdat het portefeuille gewicht niet afwijkt van het benchmark gewicht.
- Het totale allocatie effect is groter dan het totale selectie effect. We concluderen dat de allocatiebeslissing de grootste bijdrage heeft geleverd aan de value added.

In bovenstaand model bestaat de allocatieboom uit één laag; het kapitaal wordt in één keer opgedeeld in  $N$  segmenten. Er bestaan ook uitbreidingen op dit model voor beleggingen met meerdere lagen in de IDP boom (zogenoeten **multi-level modellen**). In plaats van twee effecten kent de wortel van de boom dan meerdere effecten, afhankelijk van het aantal lagen. Over de niveaus van de boom kent ieder knooppunt één effect minder dan zijn “parent”.

### Attributie van valuta

Sommige portefeuilles beleggen in verschillende valuta. Een rendement op een marktinstrument is dan afkomstig van twee factoren:

- winst/verlies in de lokale valuta (dat wil zeggen de valuta van het land waarin het instrument wordt verhandeld);
- winst/verlies op de wisselkoers tussen de lokale valuta en de valuta van de portefeuille.

Het is informatief om een idee te verkrijgen welk deel van de value added te danken is aan valuta schommeling en welk deel aan bewegingen in de lokale markt. Er bestaan attributiemodellen waarmee dit kan worden onderzocht. Iedere value added wordt dan opgesplitst in een valuta-effect, een allocatie-effect en een selectie-effect, waarbij de laatste twee onafhankelijk zijn van bewegingen in wisselkoersen. We behandelen deze modellen verder niet.

### Multi-periodieke attributie

Ingeval de performance wordt geëvalueerd over meerdere subperiodes, is het nuttig om te weten welke subperiodes wat hebben bijgedragen aan de totale performance. Ieder effect willen we dan schrijven als een som van corresponderende effecten over de subperiodes. Dit heet een **multi-periodieke attributie**. Hier hebben geometrische attributiemodellen een groot voordeel, omdat effecten over de tijd geometrisch kunnen worden samengesteld met als uitkomst het effect over de gehele periode. Een additief model is wel mogelijk, maar werkt ingewikkelder.

Veronderstel dat er  $n$  subperiodes zijn. We noteren:

$$\begin{aligned}
 w_i^{P,j} &:= \text{het gewicht over subperiode } i \text{ van segment } j \text{ van de portefeuille;} \\
 w_i^{M,j} &:= \text{het gewicht over subperiode } i \text{ van segment } j \text{ van de benchmark;} \\
 R_i^{P,j} &:= \text{het rendement over subperiode } i \text{ van segment } j \text{ van de portefeuille;} \\
 R_i^{M,j} &:= \text{het rendement over subperiode } i \text{ van segment } j \text{ van de benchmark;} \\
 R_i^P &:= \sum_{j=1}^N w_i^{P,j} R_i^{P,j}, \text{ het rendement over subperiode } i \text{ van de portefeuille;} \\
 R_i^M &:= \sum_{j=1}^N w_i^{M,j} R_i^{M,j}, \text{ het rendement over subperiode } i \text{ van de benchmark,}
 \end{aligned}$$

voor  $i=1, \dots, n$  en  $j=1, \dots, N$ .

De aritmetische value added  $VA$  kan middels de volgende stappen worden opgesplitst in een som met in iedere term een subperiodieke value added  $VA_i := R_i^P - R_i^M$ .

1. Neem een subperiode  $i$  en bereken de subperiodieke value added  $VA_i$ .
2. Link dit getal met de portefeuille rendementen voor alle voorgaande subperiodes (voor zover die er zijn), d.w.z. vermenigvuldig met  $\prod_{k=1}^{i-1} (1 + R_k^P)$ .
3. Link het resultaat met de benchmark rendementen voor alle opvolgende subperiodes (voor zover die er zijn), d.w.z. vermenigvuldig met  $\prod_{k=i+1}^n (1 + R_k^M)$ .

Het resultaat is de bijdrage van subperiode  $i$  aan de value added  $VA$ . De som van deze bijdragen is gelijk aan  $VA$ .

Het bewijs van deze multi-periodieke decompositie is de volgende berekening. (Definieer voor het gemak een leeg product gelijk aan 1.)



$$\begin{aligned}
& \sum_{i=1}^n \left[ \prod_{k=1}^{i-1} (1 + R_k^P) \right] (R_i^P - R_i^M) \left[ \prod_{k=i+1}^n (1 + R_k^M) \right] \\
&= \sum_{i=1}^n \left[ \prod_{k=1}^{i-1} (1 + R_k^P) \right] (1 + R_i^P) \left[ \prod_{k=i+1}^n (1 + R_k^M) \right] - \left[ \prod_{k=1}^{i-1} (1 + R_k^P) \right] (1 + R_i^M) \left[ \prod_{k=i+1}^n (1 + R_k^M) \right] \\
&= \prod_{i=1}^n (1 + R_i^P) - \prod_{i=1}^n (1 + R_i^M) \\
&= VA.
\end{aligned}$$

De tweede gelijkheid volgt uit het feit dat de reeks een telescoopreeks van verschillen vormt, waarbij slechts twee termen niet tegen een andere term wegvallen.

De behandelde decompositie van value added over de tijd kan ook worden toegepast op de value added van afzonderlijke segmenten en op effecten zoals allocatie-effect, selectie-effect en interactie-effect. Dit geeft een additieve attributiemethode over meerdere subperiodes.

## 2.6 Performance presentatie

Het onderwerp performance presentatie speelt niet zo'n belangrijke rol in deze scriptie. Daarom is voor de volledigheid slechts een beknopte beschrijving opgenomen in deze paragraaf.

Vermogensbeheerders presenteren regelmatig hun behaalde resultaten aan huidige en potentiële klanten. Klanten die op zoek zijn naar een portefeuillebeheerder laten zich bij hun keuze vaak in grote mate leiden door resultaten van de kandidaat-managers uit het verleden. Veel instellingen proberen dan ook de historische prestaties van hun managers zodanig te manipuleren dat deze in een zo gunstig mogelijk daglicht worden gesteld. Enkele voorbeelden van keuzes die een vermogensbeheerder zou kunnen maken zijn:

- van welke portefeuilles de performance wordt getoond en van welke niet;
- over welke tijdsperiode de performance wordt gepresenteerd;
- hoe rendementen worden berekend;
- welke performance statistieken worden gebruikt.

Om tegen te gaan dat performance resultaten zodanig worden beïnvloed dat de gepresenteerde resultaten een incorrect beeld geven van de werkelijke resultaten, heeft de **Association for Investment Management and Research (AIMR)** in 1993 een set regels en richtlijnen gepubliceerd met als doel dat er eerlijke, nauwkeurige en volledige performance informatie wordt getoond. Deze regels en richtlijnen staan bekend als de **AIMR Performance Presentation Standards (AIMR-PPS)**.

Het hanteren van deze standaards is geheel vrijwillig, al is het in de Verenigde Staten door de jaren heen steeds moeilijker geworden om nieuwe klanten aan te trekken, ingeval een asset management instelling niet kan claimen te voldoen aan de AIMR-PPS. Wanneer een instelling wel claimt te voldoen aan de standaards, wijst een accountantscontrole uit of de PPS-claim terecht is. Het ontbreken van de claim is strafbaar en bovendien schadelijk voor de goede naam van de instelling.

We behandelen een aantal voorbeelden van door de AIMR gestelde regels:

- De TWR is het relevante return-begrip. Omdat de waardering van de portefeuille op het moment van een cash flow niet altijd bekend is, staat AIMR benaderingen toe waarbij per maand een MWR berekening is toegestaan.
- Rendementen moeten total returns zijn: naast marktwaardering moeten inkomen en uitgaven (zowel gerealiseerd als ongerealiseerd) worden betrokken bij de berekening.
- Performance moet worden berekend vóór aftrek van management fee. De reden is dat de management fee in het algemeen onderhandelbaar is en dus verschillend zal zijn voor iedere potentiële klant.
- De periode waarover gepresenteerd wordt is minstens de afgelopen 10 jaar, tenzij de instelling nog geen 10 jaar bestaat; dan wordt de performance vanaf het moment van oprichting gepresenteerd.
- Voor ieder jaar moet het rendement, behaald over dat jaar (**jaarrendement**), worden gepresenteerd.
- Iedere portefeuille (die tegen betaling wordt beheerd) dient te worden opgenomen in minstens één **composite**. Dit is een samenvoeging van alle portefeuilles die de instelling beheert volgens een soortgelijke strategie (bijv. een composite ‘Nederlandse aandelen’ of een composite ‘Amerikaanse government bonds’). Vaak worden resultaten van een composite gepresenteerd aan potentiële klanten die voor een bepaald type portefeuille een manager zoeken.
- De verdeling van de portefeuilles over de composites moet over de tijd worden gehandhaafd, tenzij er significante veranderingen in een mandaat of composite-definitie optreden.
- Van portefeuilles die zijn opgeheven moet de volledige historische performance beschikbaar blijven binnen de composite waartoe deze heeft behoord.
- Eigenschappen van composites zoals aanvangsdatum, omschrijving, gebruikte valuta, en specificaties ten aanzien van rendementberekeningen dienen op aanvraag beschikbaar te zijn.

Een volledige lijst van standaards is te vinden in de handboeken van AIMR en op de internetsite van CFA: <http://www.cfainstitute.org>. In de appendix van [4] is een uitgebreide samenvatting van de standaards opgenomen.

### **PPS buiten de VS**

In de loop van de tijd zijn de AIMR-PPS ook in landen buiten de Verenigde Staten in gebruik gekomen. Echter, door afwijkende lokale omstandigheden en wetgevingen ten opzichte van de VS, zijn er per land aangepaste standaarden ontwikkeld. Zo zijn sinds 1998 in Nederland de **VBA Performance Presentatie Standaarden** in gebruik.

Sinds 2000 bestaat er ook een wereldwijde set standaards, de **Global Investment Performance Standards (GIPS)**. De GIPS zijn bedoeld als een minimum set van standaards voor performance presentatie. Ieder land vult deze standaards aan met regels en richtlijnen die toegespitst zijn op de lokale situatie.

### 3. Portfolio opportunity distributions

Dit hoofdstuk behandelt in hoofdlijnen de methode van PODs, het onderwerp van de scriptie. We gaan in op hoe de methode globaal werkt en hoe deze de problemen van benchmarks en peer groups elimineert. Tevens bespreken we een aantal punten van kritiek op het gebruik van PODs en wat eventueel als antwoord op deze kritiek gegeven kan worden. In de laatste twee paragrafen komt de statistiek, die bij de analyse van PODs een rol speelt, aan de orde.

#### 3.1 De alternatieve methode van PODs

In het vorige hoofdstuk zijn een aantal problemen besproken bij gebruik van benchmarks en peer groups. Ieder probleem heeft als gevolg dat de gebruikte methode niet voldoet aan één of meer van de in paragraaf 2.1 genoemde criteria voor PE. We geven hiervan een overzicht. De knelpunten van benchmark methodes staan in tabel 3:

Tabel 3: Knelpunten van benchmarks.

Knelpunt	Geschonden criteria
Statistische significantie is alleen te verkrijgen over de tijd.	4 (significantie)
Een benchmark is niet volledig representatief voor het mandaat (dit in mindere mate bij custom benchmarks).	1 (geschiktheid)
Er wordt geen rekening gehouden met inkomen (behalve bij total return benchmarks).	1 (geschiktheid)
Er vindt meestal geen correctie plaats voor transactiekosten, fees en belasting.	1 (geschiktheid)

De nadelen van peer groups staan in tabel 4:

Tabel 4: Knelpunten van peer groups.

Knelpunt	Geschonden criteria
Classification bias: peers komen qua mandaat niet overeen met de manager (te veel en/of te weinig criteria).	1 (geschiktheid); 2 (zuiverheid)
Composition bias: er is te weinig data beschikbaar.	4 (significantie)
Survivorship bias: portefeuilles worden opgeheven.	2 (zuiverheid)

Al deze problemen bij gebruik van de standaard benchmarks en peer groups motiveren het onderzoek naar de alternatieve PE techniek, die hieronder wordt beschreven.

#### Een alternatief: PODs

Deze nieuwe methode is halverwege de jaren '90 ontwikkeld door Ronald Surz. Hierbij wordt de klassieke statistiek gecombineerd met de huidige rekenkracht van computers. In zijn artikelen [10] - [15] uit de literatuurlijst beschrijft Surz hoe performance berekening met PODs werkt. Andere artikelen en boeken waarin PODs aan de orde komen zijn [1], [3] en [6].

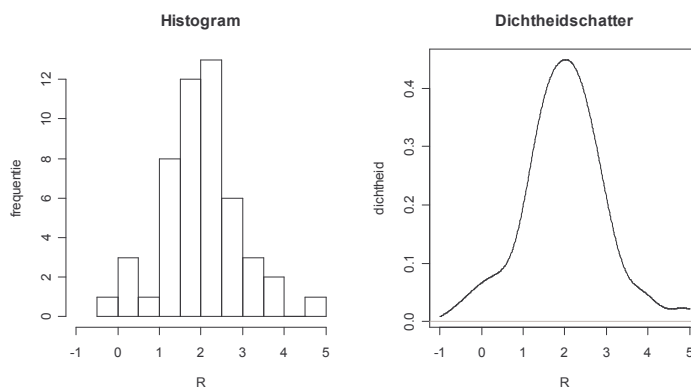
De methode is gebaseerd op het volgende principe. Om te kunnen beoordelen of een manager goed of slecht heeft gepresteerd dient zijn behaalde resultaat vergeleken te worden met alle resultaten die hij *had kunnen* behalen. Daartoe relateren we de door de manager gekozen portefeuille aan de verzameling van alle portefeuilles die de manager had kunnen samenstellen, gegeven zijn portefeuille mandaat. Concreter, we relateren het gerealiseerde rendement aan alle rendementen die behaald konden worden. Voor een gegeven portefeuille kan het rendement eenvoudig berekend worden, omdat we achteraf weten wat het verloop van de markt is geweest. Het mandaat (ofwel de set constructieregels) van de manager staat dus centraal in de methode, en het is dus van essentieel belang dat deze helder omschreven is.

De vraag is hoe we “alle mogelijke rendementen” kunnen bepalen. Dit zijn er in het algemeen oneindig veel. Daarom zullen we het concept van **random portefeuille trekking** introduceren. Stel, we trekken willekeurig een portefeuille uit de verzameling van alle toegelaten portefeuilles, volgens een bepaalde kansverdeling. Het rendement dat met deze portefeuille verkregen wordt voldoet ook aan een kansverdeling; deze verdeling noemen we de **portfolio opportunity distribution (POD)**. De verzameling van alle toegelaten portefeuilles noemen we de **POD universe**. Gegeven de POD kunnen we bepalen hoe hoog het gerealiseerde rendement is ten opzichte van de POD.

Dit idee sluit aan op een wedstrijd die The Wall Street Journal in 1988 organiseerde, waarbij beleggers het opnamen tegen een fictieve aap, die door het werpen van dartspijltjes een willekeurige portefeuille construeerde. Verrassend of niet, het bleek niet geheel eenvoudig om de aap te verslaan in termen van rendement. In de nieuwe performance methode kunnen we de POD universe vergelijken met het dartboard en de POD met de returnverdeling van de aap.

De POD, zoals hierboven geïntroduceerd, is een theoretische kansverdeling die we niet kennen. We kunnen deze echter benaderen. Eén manier om dat te doen is de verzameling van toegelaten portefeuilles te **discretiseren** en alle rendementen van de portefeuilles uit de (eindige) gediscretiseerde verzameling uit te rekenen. Een andere techniek is **Monte Carlo simulatie**. We trekken dan heel vaak, aan de hand van de gegeven constructieregels, een willekeurige toegelaten portefeuille, en berekenen de return die deze portefeuille over de meetperiode behaald zou hebben. Dit levert een reeks getallen op waarvan de empirische kansverdeling de POD benadert.

In veel gevallen heeft de POD ongeveer de vorm van een berg, zoals in figuur 11 hieronder, waarin een histogram en een dichtheidschatter van een POD is getekend.<sup>4</sup>



Figuur 11: Histogram en dichtheidschatter op basis van 50 trekkingen.

<sup>4</sup> In het vervolg zullen we met de term ‘POD’ zowel de exacte POD als de via Monte Carlo benaderde POD aanduiden.

Een curve van deze vorm doet denken aan de normale verdeling. Men kan daarom vermoeden dat PODs vaak binnen de klasse van normale verdelingen vallen. Of een aanname van normaliteit inderdaad geoorloofd is kan worden onderzocht via statistische technieken. Maar ongeacht de vorm van de POD kunnen we analyses uitvoeren die aanleiding geven tot een op statistische gronden geoorloofde performance uitspraak. Deze statistische analyses bespreken we in de paragrafen 3.4 en 3.5.

## **Twee keuzes voor de kansverdeling**

Tot besluit van deze paragraaf snijden we een punt van discussie aan. Veel PE methoden (inclusief PODs) zijn gebaseerd op het vergelijken van een prestatie met andere prestaties. Twee mogelijkheden hiervoor zijn:

- Vergelijking met prestaties van concurrenten, ofwel van de markt als geheel. De macro-economische effecten worden in dit geval dus meegenomen. Beleggingsobjecten waarin door alle beleggers gezamenlijk een groter kapitaal is belegd zullen zwaarder meewegen bij de performance vergelijking.
- Vergelijking met prestaties die de manager zelf kon behalen, gegeven zijn mandaat. Ieder individueel beleggingsobject waarin de manager kan beleggen weegt dan even zwaar mee bij de vergelijking.

Een groot deel van de bestaande methoden neigt naar de eerste variant; het relateren van de performance aan de markt. Peer group methodes relateren de behaalde performance direct aan dat van concurrentmanagers. Benchmark methodes vergelijken een reeks portefeuille rendementen met een reeks rendementen van een index, die geacht wordt representatief te zijn voor de markt waarin belegd wordt. In zekere zin lijkt dit soort vergelijking zinvol omdat iedere manager opereert in een markt en probeert de concurrentie te verslaan binnen die markt. Men kan echter als bezwaar opwerpen dat de performance afhankelijk wordt van hoe de tegenspelers presteren. De manager heeft zelf geen invloed op deze factoren en dat maakt de methode in strijd met criterium 2 (zuiverheid) uit paragraaf 2.1. Om die reden ben ik zelf geen voorstander van dit type vergelijking.

Een argument voor het tweede type (zonder macro-economische effecten) is het feit dat de manager zelf beslist hoe hij zijn portefeuille samenstelt. Bepaalde instrumenten worden door meer beleggers in portefeuille gehouden dan andere instrumenten, maar dat dwingt een individuele manager niet om ook in deze instrumenten te beleggen. In een liquide markt kan een manager transacties uitvoeren in elk aanwezig instrument. Daarom hoeft er niet meer gewicht te worden toegekend aan instrumenten die voor een hoger bedrag uitstaan in de markt, hetgeen wel gebeurt bij de eerste soort vergelijking.

Welke van de twee wordt toegepast is in feite een keuze. Deze keuze heeft een analogie in de keuze hoe een topsporter wordt beoordeeld: op basis van wedstrijduitslagen (afhankelijk van tegenstanders) of op basis van zijn getoonde spel (onafhankelijk van tegenstanders). Het is zeer belangrijk dat er onderscheid wordt gemaakt tussen de twee varianten. Dat ik persoonlijk een voorkeur heb voor de tweede variant is bij deze medegedeeld.

In deze scriptie wordt desalniettemin een poging gedaan om de POD methode beide vergelijkingen te laten ondersteunen. De keuze van de vergelijkingsmethode is in essentie de keuze van de kansverdeling volgens welke de random portefeuilles worden getrokken uit de toegelaten verzameling:

- **Uniforme verdeling** – de kansdichtheid is gelijk voor iedere toegelaten portefeuille. Alle portefeuilles uit de toegelaten verzameling kunnen even waarschijnlijk door de “aap” worden getrokken. Deze verdeling correspondeert met vergelijking van de tweede soort.
- **Kapitalisatie-gewogen verdeling**, ook genoemd **marktverdeling** – de dichtheid in iedere toegelaten portefeuille is groter naarmate de gewogen som van de marktkapitalisaties van alle instrumenten groter is. De **marktkapitalisatie**  $a^j$  (ook genoemd **market cap**) van een instrument  $j$  is het totale bedrag dat voor dat instrument uitstaat in de markt. Bijvoorbeeld, de kapitalisatie van een aandeel is het aantal uitstaande aandelen vermenigvuldigd met de huidige koers van het aandeel. Deze kansverdeling weerspiegelt de markt waarin de manager belegt. Instrumenten met een grotere kapitalisatie krijgen een groter gewicht in de verdeling. Daarmee correspondeert de verdeling met vergelijking van de eerste soort, waarbij macro-economische effecten meewegen.

Verder is het een wenselijke eigenschap van een kapitalisatie-gewogen verdeling dat het gemiddelde (of de mediaan) een goed passende benchmark portefeuille vormt. Immers, bij benchmarks zijn de exposure gewichten normaal gesproken evenredig met de marktkapitalisaties gekozen. Het gemiddelde van de resulterende POD kan dan worden gezien als een benchmark-rendement, maar wel van een benchmark die rekening houdt met alle portefeuille constructieregels die de manager opgelegd heeft gekregen. Ingeval dit rendement significant afwijkt van het rendement van een klassieke benchmark, dan geeft dat een aanwijzing dat de gekozen benchmark ongeschikt is voor het gegeven mandaat. Dit komt dan waarschijnlijk doordat bepaalde specifieke constraints niet door de benchmark worden weerspiegeld. Overigens zullen voorstanders van de uniforme verdeling beweren dat het gemiddelde van de uniforme POD de ideale benchmark portefeuille vormt.

In hoofdstuk 4, waarin we trekkingsalgoritmes beschrijven, gaan we specifiek in op hoe deze twee kansverdelingen eruit zien en hoe uit beide kansverdelingen getrokken zou kunnen worden. Het zal blijken dat uniforme trekkingen vaak makkelijker zijn uit te voeren dan kapitalisatie-gewogen trekkingen. Dit geeft, naast de conceptuele motivering, ook een praktische motivering voor gebruik van de uniforme verdeling.

### 3.2 Voordelen van PODs

Gebruik van PODs voor performance evaluatie combineert in zekere zin de goede aspecten van de klassieke technieken met peer groups en benchmarks. Alle zwakke punten van deze technieken worden door de POD methode geëlimineerd, zoals in het onderstaande wordt toegelicht.

Het gemiddelde (of eventueel de mediaan) van de POD kan worden opgevat als een perfecte benchmark: het is een soort middelste waarde van alle mogelijkheden die de belegger heeft, gegeven zijn mandaat. Men kan dus alle benchmark methodes, die in hoofdstuk 2 zijn beschreven, toepassen op het POD gemiddelde. Dit heeft aanvullend de volgende voordelen:

- Per periode hebben we niet één getal, maar een hele (geschatte) kansverdeling tot onze beschikking. Dit geeft meer informatie, zodat statistische significantie niet over de tijd verkregen hoeft te worden. Dit was het voornaamste probleem bij benchmarks.
- PODs zijn, in tegenstelling tot benchmarks, volledig representatief voor het mandaat van een manager. De behaalde performance wordt immers vergeleken met dat van alle, door het mandaat toegelaten portefeuilles.
- Afhankelijk van de gekozen rendementberekening kan er bij PODs rekening worden gehouden met inkomen, management fee en belasting. Deze gegevens zijn immers achteraf bekend.



We zien dus dat de nadelen van benchmarks worden geëlimineerd door PODs.

Tevens kan de POD universe gebruikt worden als peer group. Op de vraag welke portefeuilles deze peer group bevat luidt het antwoord: alle voor de belegger relevante portefeuilles. In zekere zin kan de POD universe worden gezien als een perfecte peer group omdat het ook alle onzuiverheden, inherent aan traditionele peer groups, elimineert:

- Een POD universe kan een zeer groot aantal (mogelijk duizenden) portefeuilles bevatten. Daarmee is de methode niet onderhevig aan composition bias.
- Door dit grote aantal portefeuilles, die letterlijk uit het mandaat zijn getrokken, wordt de situatie van de manager zeer dicht benaderd. De POD representeert de beleggingscriteria bijna exact. Dus classification bias wordt opgeheven.
- Ook survivorship bias wordt opgeheven, omdat beëindiging van portefeuilles geen rol speelt bij gesimuleerde portefeuilles.

We zien dus dat de POD methode heel wat voordelen heeft boven de standaardmethoden, en zelfs de nadelen van deze standards elimineert. Dit komt onder andere doordat niet de markt of de concurrentie centraal staat, maar het beleggingsmandaat! Alle portefeuille constructieregels, die gezamenlijk het mandaat vormen, maken deel uit van de input gegevens. Eigenlijk is dit een minimumeis voor een goede performance meettechniek, maar benchmarks noch peer groups voldoen aan dit kenmerk.

Paragraaf 3.3 bespreekt de mogelijke punten van kritiek op PODs.

### **3.3 Kritische kanttekeningen (en de antwoorden daarop)**

Ondanks alle voordelen, genoemd in de vorige paragraaf, worden PODs in de praktijk nog weinig gebruikt. Een groot gedeelte van de PE professionals weet zelfs van het bestaan niet af, en dat terwijl de methode meer dan tien jaar geleden is ontdekt. Een mogelijke verklaring hiervoor is gelegen in de knelpunten van PODs die we hieronder bespreken.

- Het mandaat van de te evalueren manager moet precies gedefinieerd zijn. Maar in de praktijk is niet altijd voor iedere portefeuille exact na te gaan of deze wel of niet onder het mandaat valt. De opdrachtgever kan bijvoorbeeld stellen dat “het risico niet te groot mag zijn”. Wat betekent “het risico” en wat betekent “niet te groot”? In zo’n geval is het mandaat niet in exacte termen gegeven. Voor de trekkingsprocedure wordt het dan noodzakelijk dat er extra aannames worden gedaan. Zo kunnen we van de portefeuille eisen dat de geschatte standaarddeviatie (of bijv. de geschatte tracking error) beneden een bepaald percentage blijft. Door te experimenteren kan de invloed van dit percentage op de performance inzichtelijk worden gemaakt.
- Een even zo belangrijk punt van kritiek heeft te maken met de zienswijze dat de POD universe een perfecte peer group voorstelt. We merken namelijk op dat de portefeuilles in deze peer group zijn beheerd door een fictieve “aap” (die via een toevalsexperiment de portefeuille samenstelt), terwijl de portefeuilles in een traditionele peer group door echte managers, en dus (hopelijk) met verstand, zijn beheerd. Dit betekent dat, ervan uitgaande dat intelligentie vaak positief bijdraagt aan de resultaten, de ranking in de POD universe mogelijk hoger uitvalt dan in een klassieke peer group, eventueel zelfs ver in de rechterstaart. Een POD universe geeft in dit opzicht dus niet aan wat veel klanten willen weten, namelijk hoe goed de performance is ten opzichte van die van gelijksoortige beheerders. Klanten willen de “wedstrijd” tussen de beleggers kunnen volgen. Men moet echter bedenken dat een traditionele peer group analyse leidt tot goede performance als de concurrentie slecht is en tot slechte performance als de concurrentie goed is. Met name bij kleine peer groups kan het algehele niveau van de concurrentie sterk aan toeval onderhevig

zijn. De te beoordelen manager heeft geen invloed op hoe dit uitpakt en mag niet op dit resultaat worden afgerekend! Dit zou anders zijn wanneer er een groot aantal perfect overeenkomende tegenspelers te vinden zijn, maar het is een feit dat zo'n groot aantal niet bestaat (composition bias). Kortom, performance vergelijken met dat van andere managers is van belang uit concurrentie overwegingen, maar dit is geen correcte wijze van performance evaluatie zoals dat in hoofdstuk 2 is gedefinieerd.

- Een ander nadeel van PODs is dat deze door middel van simulatie worden berekend, wat veel rekentijd vergt, en dus een probleem zou kunnen vormen ten aanzien van criterium 5 (praktische uitvoerbaarheid). Verderop in de scriptie zullen we zien wanneer dit probleem prominent optreedt en wanneer niet. Er worden suggesties gedaan voor alternatieve rekenwijzen ingeval de simulatie te lang duurt.
- Portefeuilles die via Monte Carlo simulatie getrokken zijn hebben dezelfde samenstelling over de gehele meetperiode (buy-and-hold). Maar bij echte portefeuilles verandert zo nu en dan de samenstelling doordat er transacties worden uitgevoerd. Daarom zouden bij het trekkingsproces ook transacties gesimuleerd moeten worden. Het is echter niet duidelijk hoe frequent deze transacties plaatsvinden en hoe radicaal de samenstelling van de portefeuille dan verandert. Een portefeuille kan meerdere keren per jaar “over de kop”, maar kan ook één of enkele keren per jaar aan een kleine gewichtscorrectie onderhevig zijn. Bovendien is het niet eenvoudig om te bepalen hoe het rendement verandert door de transactiekosten. Op dit probleem wordt verderop in de scriptie nog nader ingegaan, maar een bevredigende oplossing is er op heden niet. Wel merken we op dat vergelijking met een POD zonder transacties daarom niet volledig zinloos is. Het gerealiseerde rendement wordt vergeleken met alle mogelijke rendementen die via een “passieve strategie” behaald konden worden.
- Verder is er het nadeel dat POD methodes lastiger te begrijpen zijn dan klassieke benchmark en peer group methodes. Kennis en inzicht op het gebied van wiskundige statistiek en computersimulatie zijn noodzakelijk om te begrijpen hoe de POD techniek werkt en hoe de uitkomsten geïnterpreteerd moeten worden. Mogelijk is dit één van de belangrijkste redenen waarom van PODs thans niet veel gebruik wordt gemaakt. Van de klassieke methoden zijn vooral benchmarks welbekend en zijn de uitkomsten begrijpbaar voor wiskunde-leken. Bovendien is er veel literatuur over dit onderwerp verkrijgbaar. Waarom zou PE dan op een andere manier moeten worden uitgevoerd? Uiteraard vanwege de problemen die benchmarks met zich meebrengen! Zolang deze problemen niet bekend zijn bij de PE professionals (of wel bekend zijn, maar genegeerd worden), zal er geen behoefte aan een alternatief ontstaan.
- Tot slot, er is momenteel nog geen standaardisatie in de wijze waarop de Monte Carlo simulatie wordt uitgevoerd. Dit is nadelig ten aanzien van criterium 6 (standaardisatie). In deze scriptie wordt daarom het Monte Carlo gedeelte goed geanalyseerd, zodat een beargumenteerd voorstel voor een standaard rekenwijze kan worden gedaan.

We kunnen dus stellen dat er een aantal problemen verbonden zijn aan PODs. Voor sommige van deze problemen bestaat een redelijk bevredigende oplossing. Met name de problemen van gecompliceerdheid en het ontbreken van een standaardgebruik zorgen ervoor dat de klassieke methoden blijven domineren en de POD methode niet snel van de grond komt. Inmiddels bestaan PODs meer dan tien jaar en staat de methode nog altijd in de kinderschoenen. Op zich is dit niet vreemd: de Moderne Portefeuilletheorie heeft ook 30 jaar nodig gehad om standaard te worden. De geschiedenis leert dat vele nieuwe ontdekkingen tijd nodig hebben om in gebruik te komen. PODs vormen hierop geen uitzondering. Mogelijk worden POD analyses in de toekomst vaker opgenomen in performance rapporten, naast analyses met benchmarks en peer groups. Of PODs de klassieke methoden helemaal zullen gaan vervangen blijft voor nu onzeker, maar als aanvulling kunnen zij zeker goede diensten bewijzen.



### 3.4 Verdelingsonderzoek

Hoe het simulatiegedeelte van de POD methode (het random trekken van portefeuilles) precies werkt komt aan de orde in hoofdstuk 4. Maar als we eenmaal een reeks gesimuleerde portefeuilles hebben, hoe kunnen we deze dan onderzoeken? Aan deze vraag zijn de huidige en de volgende paragraaf gewijd.

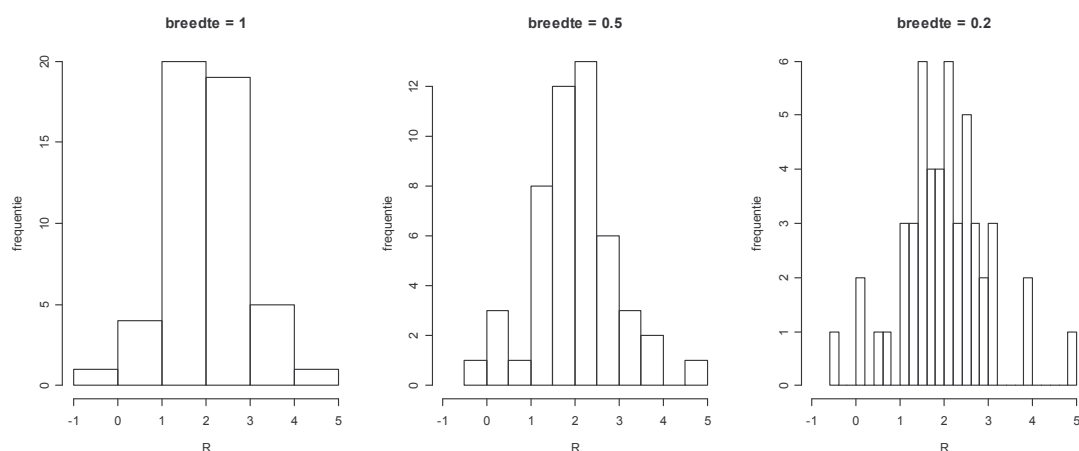
Er worden statistische technieken behandeld waarmee de gesimuleerde portefeuille-rendementen kunnen worden geanalyseerd om zo de performance van de manager te beoordelen. Het uitgangspunt is een dataset  $R_1, \dots, R_n$  van  $n$  rendementen van getrokken portefeuilles. Deze worden gezien als onafhankelijke realisaties van de stochast  $R$ , die de exacte POD als kansverdeling heeft.

Om statistische analyses op een wiskundig verantwoorde manier te kunnen toepassen is het raadzaam om de dataset eerst aan een verdelingsonderzoek te onderwerpen. Op deze manier kan een idee van de aard van de data worden verkregen (denk aan locatie, spreiding, vorm van de verdeling, aanwezigheid van staarten, etc.) en kan bovendien worden onderzocht of de aannames van een statistisch model geoorloofd zijn of juist niet. Deze paragraaf behandelt de voor ons meest bruikbare methoden voor verdelingsonderzoek, welke beschreven staan in [17] en [18].

#### Histogram en dichtheidschatter

Een eerste indruk van de onderliggende verdeling kan worden verkregen door het tekenen van een histogram of een dichtheidschatter. Tevens kan een dergelijk plaatje worden opgenomen in een performance rapport wanneer gebruik wordt gemaakt van PODs.

De betekenis van het histogram spreekt voor zich: voor intervallen van gelijke breedte wordt geteld hoeveel waarnemingen binnen het interval liggen. In figuur 12 zien we drie histogrammen met verschillende intervalbreedtes, alle toegepast op een dataset met 50 waarnemingen. Om een goede indruk van de verdeling te verkrijgen moeten de intervallen niet te breed zijn, en ook niet te smal, zoals de figuur duidelijk maakt. Een te breed interval zorgt er immers voor dat het histogram weinig informatie over de verdeling geeft. En een te smal interval leidt tot pieken in het histogram, terwijl we die wellicht niet verwachten in de onderliggende verdeling.

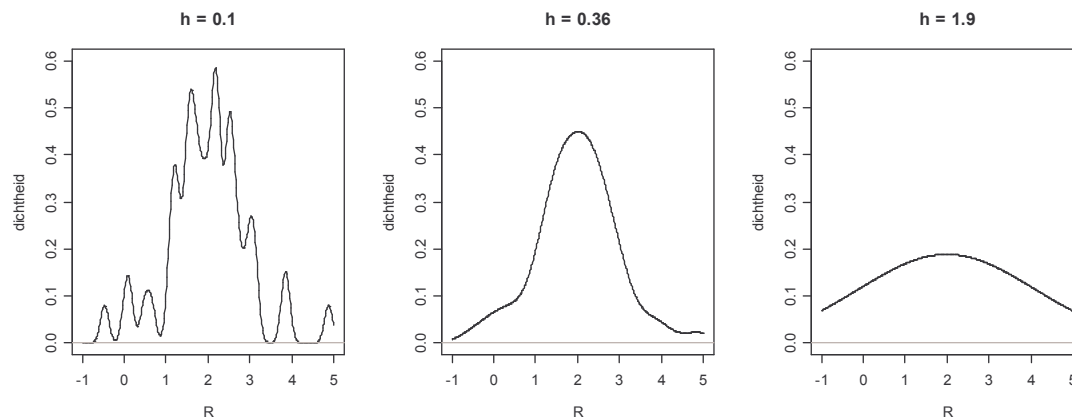


Figuur 12: Histogram voor verschillende intervalbreedte.

Dichtheidschatters verdienen wat meer toelichting. We gebruiken de methode van kernschatters om dichtheidschatters uit te rekenen. De **kernschatter** voor de onbekende dichtheid  $f^R$  van  $R$  met **bandbreedte**  $h > 0$ , op basis van de dataset  $R_1, \dots, R_n$ , is de stochastische functie

$$\hat{f}_{n,h}^R(t) := \frac{1}{nh} \sum_{i=1}^n \varphi\left(\frac{t - R_i}{h}\right).$$

Hierbij is  $\varphi$  een geschikt gekozen kansdichtheid met mediaan 0, meestal de standaard normale dichtheid. We zien de kernschatter  $\hat{f}_{n,h}^R(t)$  als een som van  $n$  “bergjes” gecentreerd rond de waarnemingen  $R_1, \dots, R_n$ , elk met een oppervlakte  $1/n$  onder zich, zodat het resultaat een dichtheid vormt met hoge waarden in gebieden waar veel data liggen, en lage waarden in gebieden waar weinig data liggen. Naarmate de bandbreedte  $h$  kleiner wordt gekozen, vertoont de schatter meer pieken, terwijl zeer grote waarden van  $h$  leiden tot een te glad resultaat. Dit is te zien in onderstaande figuur 13.



Figuur 13: Kernschatter voor verschillende bandbreedte.

Het gaat er dus om een optimale waarde voor de bandbreedte te vinden, die leidt tot een zo dicht mogelijk benadering van de ware dichtheid. Twee veelgebruikte technieken om een geschikte  $h$  te bepalen zijn:

- Minimalisatie van de **mean integrated square error (MISE)**.
- **Kruisvalidatie**.

De mean integrated square error is de functie

$$h \mapsto MISE(\hat{f}_{n,h}^R) := \mathbb{E} \int_{-\infty}^{\infty} (\hat{f}_{n,h}^R(t) - f^R(t))^2 dt.$$

Het punt van minimalisatie van deze functie wordt benaderd door de MISE uit te schrijven in een variantieterm en een gekwadraterde onzuiverheidsterm:

$$MISE(\hat{f}_{n,h}^R) = \int_{-\infty}^{\infty} \text{var} \hat{f}_{n,h}^R(t) dt + \int_{-\infty}^{\infty} (\mathbb{E} \hat{f}_{n,h}^R(t) - f^R(t))^2 dt.$$

Via Taylor ontwikkelingen kan op basis van bovenstaande decompositie de optimale bandbreedte  $h_{\text{opt}}$  worden benaderd, maar de uitdrukking daarvoor blijkt af te hangen van de onbekende dichtheid  $f^R$ . Daarom wordt in deze stap wel veronderstelt dat  $R$  uit de familie van normale verdelingen afkomstig is. De optimale bandbreedte wordt dan benaderd door

$$h_{\text{opt}} \approx 1.06 \cdot \hat{\sigma}_{R,n} n^{-1/5},$$

waarbij  $\hat{\sigma}_{R,n}$  de standaarddeviatie, geschat uit de data, voorstelt. Deze methode blijkt redelijk te werken, ook ingeval  $f^R$  niet normaal is.

De tweede techniek, kruisvalidatie, berekent een geschikte bandbreedte zonder verdere aannamen. De **integrated square error (ISE)** kan als volgt worden ontbonden:

$$ISE(\hat{f}_{n,h}^R) := \int_{-\infty}^{\infty} (\hat{f}_{n,h}^R(t) - f^R(t))^2 dt = \int_{-\infty}^{\infty} \hat{f}_{n,h}^R(t)^2 dt - 2 \int_{-\infty}^{\infty} \hat{f}_{n,h}^R(t) f^R(t) dt + \int_{-\infty}^{\infty} f^R(t)^2 dt.$$

De laatste term hangt niet van de dichtheidschatter af, zodat het volstaat om de eerste twee termen te minimaliseren. Het idee is om de stochastische variabele

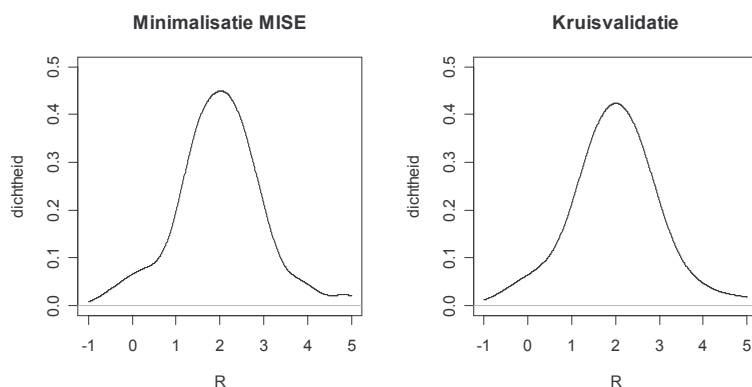
$$T(\hat{f}_{n,h}^R) := \int_{-\infty}^{\infty} \hat{f}_{n,h}^R(t)^2 dt - 2 \int_{-\infty}^{\infty} \hat{f}_{n,h}^R(t) f^R(t) dt.$$

te vervangen door een schatter  $\hat{T}(\hat{f}_{n,h}^R)$  en deze te minimaliseren naar  $h$ . De tweede integraal in de uitdrukking is gelijk aan de verwachting  $\mathbb{E}(\hat{f}_{n,h}^R(R))$ , vanwege de substitutieregels voor verwachtingen. We schatten deze verwachting uit de gegeven data door een gemiddelde te nemen, maar steeds voor de  $i$ -de term de dichtheidschatter  $\hat{f}_{n,h,-i}^R$ , gebaseerd op alle waarnemingen behalve de  $i$ -de waarneming, in te vullen. De gezochte schatter is derhalve

$$\hat{T}(\hat{f}_{n,h}^R) := \int_{-\infty}^{\infty} \hat{f}_{n,h}^R(t)^2 dt - 2 \frac{1}{n} \sum_{i=1}^n \hat{f}_{n,h,-i}^R(R_i).$$

Deze uitdrukking is slechts afhankelijk van de data en van  $h$  en kan worden geminimaliseerd naar  $h$ , wat een optimale bandbreedte  $h_{\text{opt}}$  oplevert.

Voor beide methoden bestaan scripts die in een statistisch pakket zoals R kunnen worden uitgevoerd. Hieronder is voor het voorbeeld in figuur 14 de kernschatter getekend, met optimale bandbreedte bepaald via zowel minimalisatie van de MISE als via kruisvalidatie.



Figuur 14: Kernschatter met optimale bandbreedte berekend volgens twee criteria.

## QQ-plot

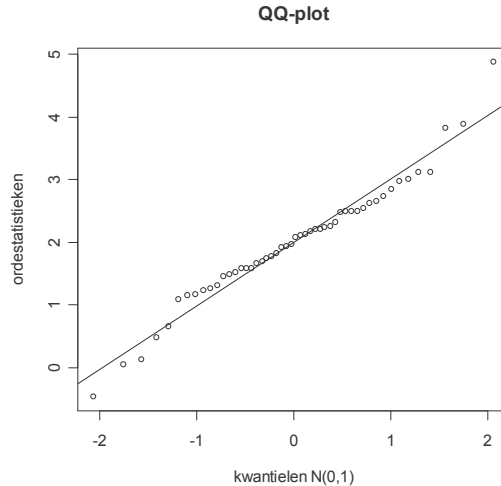
Een histogram of dichtheidschatter kan een idee geven tot welke **locatie-schaal-familie** van verdelingen de POD behoort. Dit is een verzameling van verdelingen  $\mathcal{F} = \{a + bX : a \in \mathbb{R}; b > 0\}$ , met de stochast  $X$  een **voortbrenger**, die de vorm van de lidverdelingen bepaalt, en getallen  $a, b$  die de locatie en de schaal bepalen. Een voorbeeld van zo'n familie is de normale familie  $\{\mu + \sigma N(0,1) : \mu \in \mathbb{R}; \sigma > 0\}$ , bestaande uit alle  $N(\mu, \sigma^2)$  verdelingen. De voortbrenger is in dit geval de standaardnormale verdeling.

Een eenvoudige manier om te bepalen of het terecht is om aan te nemen dat de onderliggende verdeling van  $R_1, \dots, R_n$  lid is van een bepaalde locatie-schaal-familie, is het tekenen van een **QQ-plot**. Dit is een scatterplot in het vlak van de volgende punten:

$$\left\{ \left( F^{-1}\left(\frac{i}{n+1}\right), R_{(i)} \right) : i = 1, \dots, n \right\},$$

met  $F$  de verdelingsfunctie van een lid van de familie (bijvoorbeeld de voortbrenger), en  $R_{(i)}$  de  $i$ -de ordestatistiek van de  $R_1, \dots, R_n$ . Liggen de punten in de scatterplot nagenoeg op één lijn, dan geeft dat een indicatie dat de onderliggende verdeling er één uit de locatie-schaal-familie van  $F$  is. Eventueel kunnen de getallen  $a, b$  worden geschat met behulp van schatters voor verwachting en variantie. Vormen de punten een gebogen kromme dan geeft dat een aanwijzing over de relatieve zwaarte van de staarten van de onderliggende verdeling ten opzichte van  $F$ .

In het figuur 15 hieronder zijn datapunten uitgezet tegen normale kwantielen. Op het oog vormen de datapunten redelijkerwijs een rechte lijn, hoewel de uitschieters aan beide kanten iets afwijken van de lijn. Dit komt doordat in het voorbeeld waarden van de data begrensd zijn, terwijl de normale verdeling staarten heeft. De  $R_1, \dots, R_n$  zouden dus wellicht gezien kunnen worden als realisaties uit een normale verdeling, maar zeer overtuigend is dat niet.



Figuur 15: QQ-plot.

### Goodness of fit toetsen

Hoewel een QQ-plot vaak een redelijke impressie geeft, bestaan er meer formele methoden om een dataset te fitten met een bepaalde verdeling of familie van verdelingen. Dit zijn de statistische toetsen van de vorm

$$\begin{aligned} H_0 &: F^R \in \mathcal{F}; \\ H_1 &: F^R \notin \mathcal{F}, \end{aligned}$$

met  $\mathcal{F}$  een gegeven familie van kansverdelingen. We behandelen drie van zulke **goodness of fit toetsen**:

- Kolmogorov-Smirnov:  $\mathcal{F} = \{F\}$ ;
- Shapiro-Wilk:  $\mathcal{F} = \text{'normale familie'}$ ;
- Jarque-Bera:  $\mathcal{F} = \text{'normale familie'}$ .

De eerste van de drie, de **Kolmogorov-Smirnov-toets** is wellicht het bekendst. Dit is een enkelvoudige toets, dat wil zeggen  $\mathcal{F}$  is van de vorm  $\mathcal{F} = \{F\}$ , zodat de nulhypothese geschreven kan worden als  $F^R = F$ . De **empirische verdelingsfunctie** van de dataset  $R_1, \dots, R_n$  is:

$$\hat{F}_n^R(t) := \frac{\#\{i : R_i \leq t\}}{n} = \frac{1}{n} \sum_{i=1}^n 1_{\{R_i \leq t\}}.$$

De **Kolmogorov-Smirnov-statistiek** is gedefinieerd door

$$D_n := \sup_{t \in \mathbb{R}} \left| \hat{F}_n^R(t) - F(t) \right| = \max_{i=1, \dots, n} \max \left\{ \left| F(R_{(i)}) - \frac{i}{n} \right|, \left| F(R_{(i)}) - \frac{i-1}{n} \right| \right\}.$$

Deze grootte convergeert bijna zeker naar 0, voor  $n \rightarrow \infty$  (dit is de stelling van **Glivenko-Cantelli**). We verwerpen de nulhypothese voor grote waarden van  $D_n$ . Ingeval  $F$  een continue verdeling is, hangt de verdeling van  $D_n$  onder de  $H_0$  niet af van de ware verdeling  $F^R$  van de rendementen. Met andere woorden, de Kolmogorov-Smirnov-statistiek is

verdelingsvrij over de klasse van continue verdelingsfuncties. Voor een gegeven onbetrouwbaarheidsdrempel  $\alpha$  kunnen kritieke waarden of overschrijdingskansen worden uitgerekend met behulp van een statistisch pakket.

Merk op dat de Kolmogorov-Smirnov-toets in bovenstaande vorm alleen voor een enkelvoudige nulhypothese toepasbaar is. Het is in deze vorm niet mogelijk om te toetsen of de onderliggende verdeling tot een bepaalde locatie-schaal-familie, zoals de normale familie, behoort. Wanneer we de data standaardiseren, met als doel deze te fitten met de  $N(0,1)$  verdeling, geldt niet meer dat  $D_n$  verdelingsvrij is onder  $H_0$ !

Om te toetsen of de rendementen  $R_1, \dots, R_n$  afkomstig zijn uit een normale verdeling kunnen we de **toets van Shapiro-Wilk** toepassen. De nulhypothese van deze toets zegt dat de onderliggende verdeling  $F^R$  lid is van de normale familie. Om een passende toetsingsgrootte te vinden voeren we wat notatie in. Laat  $Z_1, \dots, Z_n$  een steekproef uit de  $N(0,1)$  verdeling voorstellen. We definiëren de vector  $c$  door de verwachting van de vector  $(Z_{(1)}, \dots, Z_{(n)})$  van ordestatistieken:

$$c := (\mathbb{E}Z_{(1)}, \dots, \mathbb{E}Z_{(n)}),$$

de matrix  $\Sigma$  door de covariantiematrix van dezelfde vector:

$$\Sigma := \text{cov}(Z_{(1)}, \dots, Z_{(n)}),$$

en de vector  $a$  door

$$a := \frac{\Sigma^{-1}c}{\sqrt{c^T \Sigma^{-1} \Sigma^{-1} c}}.$$

De toetsingsgrootte is nu gegeven door

$$W_n := \frac{(\sum_{i=1}^n a_i R_{(i)})^2}{\sum_{i=1}^n (R_i - \bar{R}_n)^2} \in (0, 1].$$

De teller van deze grootte schat het getal  $n \cdot \text{var} R$  wanneer de aanname van normaliteit klopt (dit volgt uit een lineair regressiemodel met gewogen kleinste kwadraten). De noemer schat  $n \cdot \text{var} R$  altijd! Simulatiestudies laten zien dat de verdeling van  $W_n$  onder een niet normale verdeling doorgaans naar links verschoven is, zodat we normaliteit verwerpen voor kleine waarden van  $W_n$ . De toets kan voor gegeven data worden uitgevoerd in een statistisch pakket. De getallen  $a_i$  en de kritieke waarden van  $W_n$  zijn daarin beschikbaar.

Een andere veelgebruikte toets, die speciaal ontworpen is om normaliteit te verwerpen, is de **Jarque-Bera-toets**. De bijbehorende toetsingsgrootte maakt gebruik van twee kengetallen van de dataset, te weten de scheefheid en de kurtosis. De **scheefheid**  $\gamma_n$  van de waarnemingen  $R_1, \dots, R_n$  is gedefinieerd door het derde gestandaardiseerde moment:

$$\gamma_n := \frac{1}{n} \sum_{i=1}^n \frac{(R_i - \bar{R}_n)^3}{\sigma_{R,n}^3}.$$

Het is een schatter voor de scheefheid  $\gamma$  van  $R$ :

$$\gamma := \frac{\mathbb{E}(R - \mathbb{E}R)^3}{(\text{sd } R)^3}.$$

Dit is een maat voor de asymmetrie in de kansverdeling van  $R$  om z'n verwachting. Voor een normale verdeling is de scheefheid gelijk aan 0. De **kurtosis**  $k_n$  van  $R_1, \dots, R_n$  is het vierde gestandaardiseerde moment:

$$k_n := \frac{1}{n} \sum_{i=1}^n \frac{(R_i - \bar{R}_n)^4}{\sigma_{R,n}^4}.$$

Het is een schatter voor de kurtosis  $k$  van  $R$ :

$$k := \frac{\mathbb{E}(R - \mathbb{E}R)^4}{(\text{sd } R)^4}.$$

Dit is een maat die aangeeft hoe gepiekt de kansdichtheidsfunctie van  $R$  is. Voor een normale verdeling is de kurtosis gelijk aan 3. Ten behoeve van vergelijking met de normale verdeling wordt wel de **excess kurtosis** van de waarnemingen gedefinieerd door

$$k_n^{\text{excess}} := k_n - 3.$$

De **Jarque-Bera-statistiek** is een stijgende functie van de scheefheid en de excess kurtosis:

$$JB_n := \frac{n}{6} \left( \gamma_n^2 + \frac{(k_n^{\text{excess}})^2}{4} \right).$$

Onder de nulhypothese van normaliteit van  $R$  blijkt deze statistiek asymptotisch de chikwadraat-verdeling met 2 vrijheidsgraden te bezitten. De toets verwerpt  $H_0$  voor grote waarden van  $JB_n$ , dus bij onbetrouwbaarheid  $\alpha$  ongeval voor de realisatie  $jb_n$  geldt

$$jb_n > F_{\chi_2^2}^{-1}(1 - \alpha),$$

met  $F_{\chi_2^2}^{-1}$  de inverse verdelingsfunctie van de chikwadraat-verdeling met 2 vrijheidsgraden. Voor  $\alpha = 0.05$  is dit ongeveer 6. Dus normaliteit wordt verworpen als de Jarque-Bera statistiek boven de 6 uitkomt.

### 3.5 Statistische analyse van de performance

Wanneer we een indruk hebben gekregen hoe de POD eruit ziet, kunnen we onderzoeken hoe het door de manager gerealiseerde rendement zich verhoudt tot de POD. Deze onderzoeken zijn het onderwerp van de huidige paragraaf. Gegeven is een dataset  $R_1, \dots, R_n$  van rendementen van getrokken portefeuilles, en een gerealiseerd rendement  $R^P$  van de te beoordelen manager.

In eerste instantie kunnen de evaluatiemethodes van peer groups worden toegepast. Immers, de setting van het huidige probleem is analoog aan dat van een peer group analyse. De dataset  $R_1, \dots, R_n$  is een reeks van onafhankelijke rendementen, hetgeen de peer group rendementen ook zijn. We kunnen dus informatie verkrijgen uit bijvoorbeeld een percentiel-ranking of een floating bar chart (zie paragraaf 2.3). Naast deze analyses zijn er nog andere statistische technieken mogelijk, die echter vaak wel een groot aantal waarnemingen vereisen.

## POD ranking

De **POD ranking**  $\theta$  is gedefinieerd door

$$\theta := \mathbb{P}(R > R^p),$$

ofwel de kans dat de ‘‘aap’’ een hoger rendement behaalt dan de belegger. Gegeven de dataset  $R_1, \dots, R_n$  (die we interpreteren als een reeks van onafhankelijke realisaties van  $R$ ), kunnen we  $\theta$  schatten door het gemiddelde

$$\hat{\theta}_n := \bar{Y}_n = \frac{1}{n} \sum_{i=1}^n Y_i,$$

van

$$Y_i := \begin{cases} 1 & \text{als } R_i > R^p; \\ 0 & \text{als } R_i \leq R^p, \end{cases}$$

voor  $i = 1, \dots, n$ . Hoe dicht deze schatter in de buurt van de werkelijke  $\theta$  ligt, is te beoordelen aan de hand van een betrouwbaarheidsinterval. Een asymptotisch betrouwbaarheidsinterval met een onbetrouwbaarheid  $\alpha$  is te vinden met behulp van een uitbreiding van de centrale limietstelling. Deze zegt dat bij benadering

$$\sqrt{n} \frac{\bar{Y}_n - \theta}{\sigma_{Y,n}} \sim N(0, 1),$$

waarbij de standaarddeviatie  $\sigma_{Y,n}$  gegeven is door de formule

$$\sigma_{Y,n} = \sqrt{\frac{n}{n-1} \bar{Y}_n (1 - \bar{Y}_n)}.$$

Deze eenvoudige formule geldt omdat  $n\bar{Y}_n$  binomiaal verdeeld is met parameters  $n$  en  $\theta$ . Een benaderd betrouwbaarheidsinterval voor  $\theta$  is derhalve gegeven door

$$\theta = \bar{Y}_n \pm \frac{\sigma_{Y,n}}{\sqrt{n}} \Phi^{-1}(1 - \alpha/2),$$

ofwel door

$$\theta = \bar{Y}_n \pm \sqrt{\frac{\bar{Y}_n (1 - \bar{Y}_n)}{n-1}} \Phi^{-1}(1 - \alpha/2),$$



met  $\Phi^{-1}$  de inverse verdelingsfunctie van de standaardnormale verdeling. De breedte van het betrouwbaarheidsinterval hangt sterk af van het aantal waarnemingen. Willen we voor  $\alpha = 0.05$  een breedte van minder dan 0.01 (dat wil zeggen, de schatter  $\bar{Y}_n$  schat  $\theta$  op ongeveer 2 decimalen achter de komma nauwkeurig), dan kunnen we erop rekenen dat we in de orde van 40000 waarnemingen nodig zullen hebben. Voor een breedte van 0.1 hebben we in de orde van 400 waarnemingen nodig. Dit volgt uit het feit dat voor  $\alpha = 0.05$  geldt  $\Phi^{-1}(1 - \alpha/2) \approx 1.96$ . Verder kan de afschatting  $\bar{Y}_n(1 - \bar{Y}_n) \leq 1/4$  worden gemaakt. Invullen in de ongelijkheid

$$2 \cdot 1.96 \cdot \sqrt{\frac{1/4}{n-1}} \leq \text{gewenste breedte},$$

geeft een benedengrens op  $n$ .

Voor de POD ranking  $\theta$  is ook een exact betrouwbaarheidsinterval mogelijk (zonder gebruik van de centrale limietstelling). Deze wordt gevonden door de toets met nulhypothese van de vorm  $\theta = \theta_0$  te “inverteren”. Dit levert bij realisaties  $y_i$  van  $Y_i$  het volgende interval:

$$\left\{ \theta_0 : \mathbb{P}_{\theta=\theta_0} \left( \sum_{i=1}^n Y_i \geq \sum_{i=1}^n y_i \right) > \alpha/2 \quad \text{en} \quad \mathbb{P}_{\theta=\theta_0} \left( \sum_{i=1}^n Y_i \leq \sum_{i=1}^n y_i \right) > \alpha/2 \right\}.$$

Om de grenzen van dit interval te vinden dienen we de vergelijkingen

$$\mathbb{P}_{\theta=\theta_0} \left( \sum_{i=1}^n Y_i \geq \sum_{i=1}^n y_i \right) = \alpha/2$$

$$\mathbb{P}_{\theta=\theta_0} \left( \sum_{i=1}^n Y_i \leq \sum_{i=1}^n y_i \right) = \alpha/2$$

op te lossen naar  $\theta_0$ . Door te gebruiken dat  $\sum_{i=1}^n Y_i$  onder  $\theta = \theta_0$  de  $\text{Bin}(n, \theta_0)$  verdeling heeft, kunnen deze vergelijkingen op de computer worden opgelost. Voor verdere details verwijzen we naar [17].

## Toetsen voor locatie

Een schatter voor  $\theta$  geeft aan of de “aap” is verslagen. Met behulp van een toets kan worden bepaald of de “aap” *significant* is verslagen. De toets die we willen uitvoeren zal van de volgende vorm zijn:

$$H_0 : \text{med}(R) \geq R^P;$$

$$H_1 : \text{med}(R) < R^P,$$

waarbij  $\text{med}(R)$  de mediaan van de stochast  $R$  voorstelt. We moeten daartoe de aanname maken dat de verdelingsfunctie van  $R$  de waarde 1/2 aanneemt, met andere woorden dat de mediaan bestaat. Hier is zeker aan voldaan indien  $R$  een dichtheid heeft. Wordt de nulhypothese verworpen, dan is met grote zekerheid te concluderen dat het gerealiseerde rendement boven de meeste POD rendementen uitkomt, en dus dat de manager significant beter belegt dan de “aap”.

Hoe de bovenstaande toets verder vorm krijgt hangt ervan af of we  $R$  normaal verdeeld kunnen veronderstellen. Is dit zo, dan is de mediaan gelijk aan de verwachting, en volstaat een **t-toets**, waarvan eerder een variant is besproken in paragraaf 2.3. In de huidige context is de bijbehorende **t-statistiek** gelijk aan

$$T_n := \sqrt{n} \frac{\bar{R}_n - R^P}{\sigma_{R,n}},$$

met

$$\begin{aligned} \bar{R}_n &:= \text{het gemiddelde van de } R_1, \dots, R_n; \\ \sigma_{R,n} &:= \text{de standaarddeviatie van de } R_1, \dots, R_n. \end{aligned}$$

De nulhypothese wordt verworpen voor kleine waarden van  $T_n$ . Ingeval  $\mathbb{E}R = R^P$  heeft de toetsingsgrootheid  $T_n$  de t-verdeling met  $n-1$  vrijheidsgraden. Dus bij een onbetrouwbaarheidsdrempel  $\alpha$  wordt  $H_0$  verworpen als voor de realisatie  $t_n$  van  $T_n$  geldt dat

$$t_n < F_{t_{n-1}}^{-1}(\alpha),$$

met  $F_{t_{n-1}}^{-1}$  de inverse verdelingsfunctie van de  $t_{n-1}$  verdeling. Dit gebeurt ook als voor de overschrijdingskans geldt

$$\mathbb{P}_{\mathbb{E}R=R^P}(T_n < t_n) \leq \alpha.$$

Is de POD verdeling niet normaal, dan zullen in het algemeen verdelingsvrije toetsen gebruikt moeten worden. Zulke toetsen hebben minder onderscheidend vermogen dan de t-toets, omdat er minder bekend wordt verondersteld over de onderliggende verdeling. Is de nulhypothese in dat geval ongeldig, dan zullen er meer waarnemingen nodig zijn opdat deze daadwerkelijk wordt verworpen. Een mogelijke toets voor dit geval is de **tekentoets**. De toetsingsgrootheid is

$$T_n := \#\{i = 1, \dots, n : R_i > R^P\} = \sum_{i=1}^n Y_i,$$

Voor  $Y_i$  zoals eerder gedefinieerd. We verwerpen  $H_0$  voor kleine waarden van  $T_n$ . Onder  $\text{med}(R) = R^P$  heeft  $T_n$  een binomiale verdeling met parameters  $n$  en  $1/2$ . Dus  $H_0$  wordt verworpen bij een drempel  $\alpha$  indien

$$t_n < F_{\text{Bin}(n, 1/2)}^{-1}(\alpha),$$

of indien voor de overschrijdingskans geldt

$$\mathbb{P}_{\text{med}(R)=R^P}(T_n < t_n) \leq \alpha.$$

### Gevallen met een kleine dataset

Er kunnen zich situaties voordoen waarbij het praktisch niet mogelijk om een (zeer) groot aantal portefeuilles te trekken, en we genoeg moeten nemen met bijvoorbeeld een 10-tal of 100-tal waarnemingen. Betrouwbaarheidsintervallen zullen dan wellicht breder zijn en toetsen zullen minder onderscheidend zijn. Bovendien kan de ware verdeling van een gebruikte statistiek afwijken van de gebruikte verdeling, omdat die gebaseerd is op een schatter met een

groot aantal waarnemingen. Om toch een poging te doen zo veel mogelijk informatie uit de kleine dataset te halen, kunnen we een **bootstrap** toepassen.

Bij een bootstrap genereren we heel veel trekkingen uit een schatter van de onderliggende verdelingsfunctie  $F^R$ . Dit kan in principe op twee manieren:

- **Parametrisch:** we hebben reden om aan te nemen dat  $F^R$  lid is van een bepaalde locatie-schaal-familie, bijvoorbeeld de normale familie. In dit laatste geval trekken we uit de verdeling, die het meest voor de hand ligt als schatter van  $F^R$ , namelijk de  $N(\bar{R}_n, \sigma_{R,n}^2)$  verdeling.
- **Empirisch:** we weten niets over de verdeling  $F^R$ . In dat geval trekken we uit de empirische verdelingsfunctie  $\hat{F}_n^R$ , omdat die dan het meest voor de hand ligt als schatter van  $F^R$ . Dit is equivalent met het trekken met teruglegging uit de oorspronkelijke dataset  $R_1, \dots, R_n$ , wat gemakkelijk op de computer kan worden uitgevoerd.

We bespreken twee mogelijke toepassingen van de bootstrap. We beginnen met een alternatieve methode om een betrouwbaarheidsgebied voor de POD ranking  $\theta$  te verkrijgen ingeval het aantal waarnemingen  $n$  niet zo groot is. Daartoe zijn we geïnteresseerd in de verdeling van de grootheid

$$T_n := \sqrt{n} \frac{\bar{Y}_n - \theta}{\sigma_{Y,n}},$$

met nog steeds de  $Y_i$  gedefinieerd door

$$Y_i := \begin{cases} 1 & \text{als } R_i > R^P; \\ 0 & \text{als } R_i \leq R^P, \end{cases}$$

voor  $i=1, \dots, n$ . Deze is nu niet noodzakelijk normaal verdeeld, omdat de centrale limietstelling uitgaat van een grote  $n$ . Het idee van de bootstrap is dat we de verdeling  $G$  van  $T_n$  schatten door een groot aantal  $B$  realisaties van  $T_n$  te genereren met behulp van trekkingen uit een schatter van  $F^R$ . Dit gaat in een aantal stappen:

- Genereer  $B$  datasets van  $n$  observaties  $R_{1,b}, \dots, R_{n,b}$ ,  $b=1, \dots, B$ , door onafhankelijke trekkingen te doen uit een schatter van  $F^R$ . Dit doen we parametrisch ingeval we sterke aanwijzingen hebben voor normaliteit van  $F^R$ , en empirisch ingeval we die niet hebben.
- Bereken de bijbehorende  $B$  realisaties  $T_{n,1}, \dots, T_{n,B}$  van  $T_n$ .
- De schatter  $\hat{G}$  voor de verdeling  $G$  van  $T_n$  is de empirische verdelingsfunctie bepaald door de waarnemingen  $T_{n,1}, \dots, T_{n,B}$ .

Gegeven de  $T_{n,1}, \dots, T_{n,B}$  en een onbetrouwbaarheid  $\alpha$ , kunnen de kwantielen voor een betrouwbaarheidsinterval als volgt gevonden worden met behulp van de ordestatistieken  $T_{n,(1)}, \dots, T_{n,(B)}$ :

$$\begin{aligned} G^{-1}(1-\alpha/2) &\approx \hat{G}^{-1}(1-\alpha/2) = T_{n,(\lceil(1-\alpha/2)B\rceil)}; \\ G^{-1}(\alpha/2) &\approx \hat{G}^{-1}(\alpha/2) = T_{n,(\lceil(\alpha/2)B\rceil)}, \end{aligned}$$

waarbij  $\lceil \cdot \rceil$  staat voor afronding naar boven. Een betrouwbaarheidsinterval van niveau ongeveer  $\alpha$  is nu gegeven door

$$\left\{ \theta : G^{-1}(\alpha/2) \leq T_n \leq G^{-1}(1-\alpha/2) \right\} \\ \approx \left[ \bar{Y}_n - \frac{\sigma_{Y,n}}{\sqrt{n}} \hat{G}^{-1}(1-\alpha/2), \bar{Y}_n - \frac{\sigma_{Y,n}}{\sqrt{n}} \hat{G}^{-1}(\alpha/2) \right].$$

Vervangen we de standaarddeviatie, analoog aan eerder, door de formule

$$\sigma_{Y,n} = \sqrt{\frac{n}{n-1} \bar{Y}_n (1 - \bar{Y}_n)},$$

dan ziet het interval er als volgt uit:

$$\left[ \bar{Y}_n - \sqrt{\frac{\bar{Y}_n (1 - \bar{Y}_n)}{n-1}} \hat{G}^{-1}(1-\alpha/2), \bar{Y}_n - \sqrt{\frac{\bar{Y}_n (1 - \bar{Y}_n)}{n-1}} \hat{G}^{-1}(\alpha/2) \right].$$

Hoewel  $n$  mogelijk klein is, kiezen we zelf hoe groot  $B$  zal zijn. Vandaar dat  $\hat{G}$  in veel gevallen de verdeling  $G$  van  $T_n$  redelijk goed benadert. Toch is voorzichtigheid geboden bij gebruik van bootstrap methodes. Wanneer bijvoorbeeld de te schatten verdeling relatief dikke staarten heeft, kunnen verschillende toepassingen van de bootstrap aanzienlijk verschillende resultaten opleveren, ongeacht de grootte van  $B$ . Aangeraden wordt dan ook om naast een bootstrap-betrouwbaarheidsinterval altijd op een andere manier een betrouwbaarheidsinterval te berekenen (hetzij exact, hetzij asymptotisch) en na te gaan of er opvallende verschillen zijn.

De tweede toepassing van de bootstrap techniek die we bespreken, is het uitvoeren van een toets ingeval voor de toetsingsgrootte  $T_n$  alleen een asymptotisch resultaat bekend is. Dat wil zeggen, ingeval de verdeling  $G$  van  $T_n$  onder de nulhypothese bekend is voor  $n \rightarrow \infty$ . Bij een kleine dataset kan de werkelijke verdeling dan afwijken van de asymptotische verdeling.

Bij de meeste toetsen die wij hebben behandeld is dit probleem echter niet van toepassing. De verdeling van de toetsingsgrootte onder  $H_0$  is meestal bekend voor iedere  $n$ , zodat bootstrappen niet nodig is. Desondanks bespreken we voor de volledigheid de bootstrap techniek voor toetsen.

De onbekende verdeling  $G$  kunnen we schatten door een groot aantal  $B$  realisaties  $T_{n,1}, \dots, T_{n,B}$  van  $T_n$  te berekenen, waarbij elke realisatie gebaseerd is op trekkingen uit een schatter van de verdeling  $F^R$  van de data. Dit kan weer ofwel parametrisch, ofwel empirisch. Veronderstel dat we de nulhypothese verwerpen voor grote waarden van  $T_n$ . Dit gebeurt dan met onbetrouwbaarheid van ongeveer  $\alpha$ , indien voor de realisatie  $t_n$  geldt

$$t_n > \hat{G}^{-1}(1-\alpha).$$

Dit gebeurt ook indien

$$\frac{\#\{b=1, \dots, B : T_{n,b} > t_n\}}{B} \leq \alpha.$$

Hierbij is de fractie  $\#\{b=1,\dots,B:T_{n,b} > t_n\}/B$  de bootstrap schatter voor de overschrijdingskans  $\mathbb{P}_{H_0}(T_n > t_n)$ .

Ingeval we  $H_0$  verwerpen voor kleine waarden van  $T_n$ , of voor grote én kleine waarden van  $T_n$ , zijn er analoge voorwaarden waarop er verworpen wordt.

Ook voor deze toepassing wordt aangeraden om naast het uitvoeren van een bootstrap toets eveneens op een andere manier te toetsen, om te kunnen beoordelen in hoeverre de bootstrap methode betrouwbare resultaten levert.



## 4. Algoritmes voor portefeuille trekkingen

In dit hoofdstuk komt aan de orde hoe random portefeuilles uit de toegelaten verzameling, bepaald door een gegeven mandaat, kunnen worden getrokken. Allereerst stellen we een model op voor dit probleem en geven we een kort overzicht van de typen constructieregels die in een mandaat kunnen voorkomen.

Vervolgens komen algoritmes voor random trekking aan bod, samen met mogelijke uitbreidingen. In totaal worden drie algoritmes behandeld. Alle drie de algoritmes zijn uitgewerkt in een klein DOS programma dat bij de scriptie hoort, en geschreven is in de taal C++.

### 4.1 Trekkingsprobleem en mandaten

De algemene opzet van het portefeuille trekkingsprobleem is als volgt. Gegeven zijn  $N$  **beleggingsobjecten** waarover de belegger zijn exposure kan verdelen. Deze objecten kunnen marktinstrumenten (ingeval van een *selectiemanager*) of portefeuillesegmenten (ingeval van een *allocatiemanager*) zijn.

We noteren:

- $w^j$  := het gekozen gewicht van object  $j$ ;
- $w$  :=  $(w^1, \dots, w^N)$ , de  $N$ -dimensionale vector van de gewichten  $w^j$ ;
- $R^j$  := een rendement gekoppeld aan object  $j$ .

Het gerealiseerde “rendement” van de portefeuille is dan gelijk aan de convexe combinatie van de rendementen, met de gewichten als coëfficiënten:

$$R = \sum_{j=1}^N w^j R^j .$$

De vector  $w$  karakteriseert gekozen de portefeuille. De  $R^j$  zijn gegeven getallen.

#### Rendementen

Ingeval het beschouwde mandaat een selectiemandaat is, nemen we de  $R^j$  gelijk aan het behaalde rendement van het  $j$ -de instrument. In dat geval is  $R$  gelijk aan het gerealiseerde rendement van de portefeuille.

Voor allocatiemandaten ligt dit iets ingewikkelder. Wanneer we voor  $R^j$  het behaalde rendement van segment  $j$  nemen, is  $R$  ook gelijk aan het gerealiseerde rendement van de portefeuille, maar wordt de performance van de manager afhankelijk van hoe de onderliggende managers hebben gepresteerd. Dit is een ongewenste situatie. De taak van een allocatiemanager is in feite het verdelen van het kapitaal over een aantal deelmarkten, en dus dient hij afgerekend te worden op zijn vaardigheden in het voorspellen van deze markten. Daarom nemen we voor  $R^j$  niet het gerealiseerde rendement van segment  $j$ , maar het rendement van een benchmark behorende bij segment  $j$ . (Voor deze benchmark kan een index worden genomen of een POD gemiddelde, dit is in feite een keuze.) Nu is  $R$  weliswaar

niet gelijk aan het gerealiseerde rendement van de portefeuille, maar kan dit getal wel geïnterpreteerd worden als een gerealiseerde performancemaat van de manager, waarbij invloed van de onderliggende managers is geëlimineerd. Samenvattend kunnen we zeggen:

- voor allocatiemandaten is  $R^j$  het behaalde rendement van de benchmark van segment  $j$ ;
- voor selectiemandaten is  $R^j$  het behaalde rendement van instrument  $j$ .

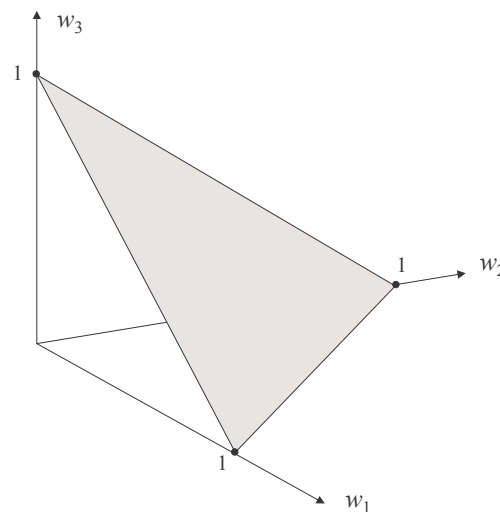
In beide gevallen zijn de  $R^j$  in de formule  $R = \sum_{j=1}^N w^j R^j$  achteraf gegeven getallen.

## Gewichten

De vector  $w = (w^1, \dots, w^N)$  is datgene wat we random willen trekken. In feiten is  $w$  dan stochastisch, dus noteren we deze met hoofdletters:  $W = (W^1, \dots, W^N)$ . Aangezien  $W$  de totale exposure van 100% verdeelt over zijn coördinaten, geldt dat  $W$  getrokken dient te worden uit de **eenheidssimplex** in  $\mathbb{R}^N$ . Dit is de verzameling

$$S_N := \left\{ w \in \mathbb{R}^N : \sum_{j=1}^N w^j = 1; w^j \geq 0, j = 1, \dots, N \right\},$$

zie figuur 16.



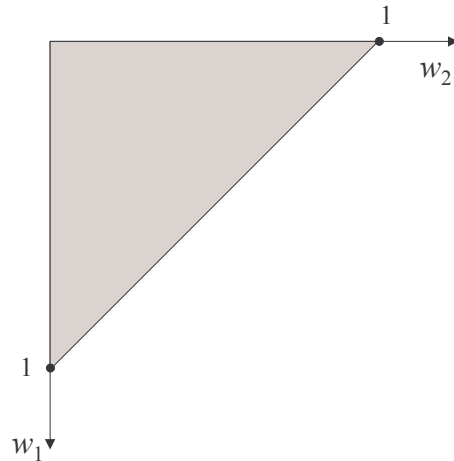
Figuur 16: Eenheidssimplex in de ruimte met dimensie 3.

Merk op dat  $w^N = 1 - \sum_{j=1}^{N-1} w^j$  voor alle  $w \in S_N$ . De verzameling  $S_N$  is daarom volledig bepaald door zijn projectie

$$S_N^{\text{Proj}} := \left\{ (w^1, \dots, w^{N-1}) \in \mathbb{R}^{N-1} : \sum_{j=1}^{N-1} w^j \leq 1; w^j \geq 0, j = 1, \dots, N-1 \right\}$$

op het hypervlak met coördinaten  $(w^1, \dots, w^{N-1}) \in \mathbb{R}^{N-1}$ , zie figuur 17.

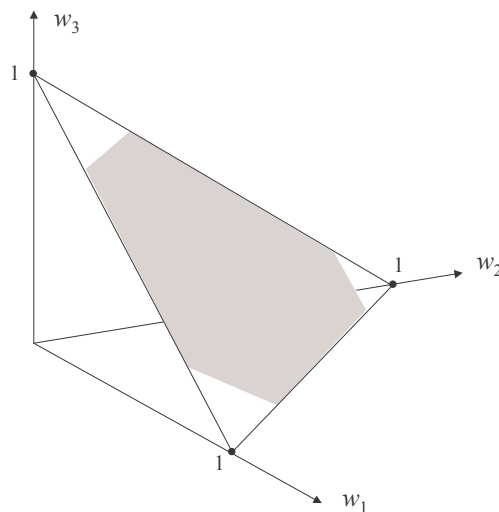




Figuur 17: Projectie van de eenheidssimplex.

## Mandaten

Hoewel iedere portefeuille gerepresenteerd wordt door een punt in de simplex, behoort niet ieder punt in de simplex tot de toegelaten verzameling. Dit komt doordat de manager is gebonden aan een zeker mandaat. Een welgedefinieerd mandaat kan worden gezien als een deelverzameling  $S_N^{\text{Mdt}} \subset S_N$  van de simplex. De portefeuille constructieregels die gezamenlijk het mandaat vormen, zorgen ervoor dat bepaalde delen van de simplex niet tot de toegelaten verzameling behoren. Dit effect wordt geïllustreerd in figuur 18.



Figuur 18: Mandaat gezien als deel van de eenheidssimplex.

We bespreken nu belangrijkste constructieregels die het mandaat specificeren. Vaak is het doel van een constructieregel om een beleggingstijl te specificeren, of het risico te beperken. De volgende constraints (bijvoorwaarden) op de gekozen gewichtsvector  $w$  geven de meest voorkomende regels:

- Ondergrenzen  $w^{j,\min}$  en bovengrenzen  $w^{j,\max}$  op de te kiezen gewichten  $w^j$ . Het verschil  $w^{j,\max} - w^{j,\min}$  wordt wel de **bandbreedte** van object  $j$  genoemd.
- Een minimum aantal objecten  $N^{\min}$  die positief gewicht moeten hebben, en/of een maximum aantal objecten  $N^{\max}$  die positief gewicht mogen hebben.

- Een minimum of maximum gewicht waarmee belegd wordt in een bepaalde deelverzameling van de objecten. Ingeval het bijvoorbeeld gaat om een mandaat uit Amerikaanse aandelen, kan er worden geëist dat er een minimum exposure in de technologie sector wordt aangenomen. Dergelijke regels komen met name voor bij zogeheten **bottom-up** strategieën, waarbij allocatie- en selectiebeslissingen tegelijk worden genomen.
- Een bovengrens op een bepaalde ex-ante risicomaat. Dit zijn schattingen van het toekomstig risico op basis van gegeven rendementen uit het verleden. Voorbeelden van zulke maten zijn de standaarddeviatie en de tracking error ten opzichte van een benchmark. Voor een gegeven vector  $w$  kan (met behulp van data uit het verleden) bepaald worden of de corresponderende portefeuille al dan niet (bij benadering) voldoet aan een dergelijk constraint.

Bovenstaande constructieregels vormen niet altijd een lineaire ongelijkheid of een anderszins “mooie” constraint op de toegelaten verzameling. Regels van type 4 leiden tot niet-lineaire constraints en regels van type 2 zullen de dimensie van de toegelaten verzameling verkleinen (dat wil zeggen, de verzameling wordt beperkt tot een deel van de rand van de simplex). Het is daarom in het algemeen niet te zeggen of de toegelaten verzameling bepaalde meetkundige eigenschappen (zoals samenhang of convexiteit) heeft.

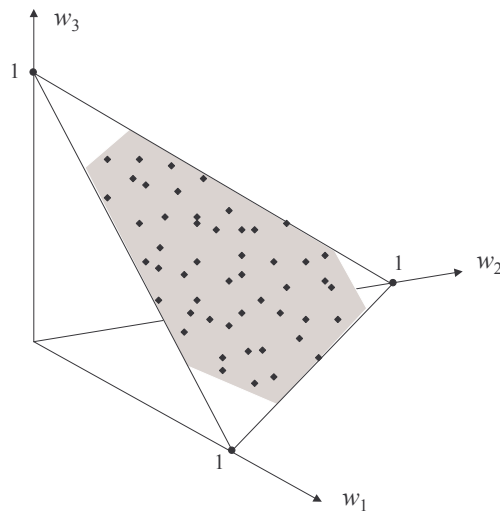
Naast de vier hierboven genoemde constraints zijn er enkele typen constraints die we zullen verwaarlozen:

- Regels ten aanzien van long en short posities. De gewichten die wij gaan trekken zijn niet-negatief. Dat wil zeggen, posities zijn ofwel long, ofwel zodanig gehedged dat de resulterende exposure niet-negatief is.
- Constraints met betrekking tot het gebruik van derivaten. We gaan er steeds vanuit dat de belegger een exposure  $w^j$  in een object  $j$  kiest die ligt tussen 0% en 100%. Deze exposure kan zowel met als zonder derivaten worden bereikt, maar we behandelen  $w^j$  alsof het gelijk is aan de fractie beginkapitaal dat in  $j$  is belegd.
- Grenzen op de hoeveelheid transacties die gedurende de evaluatieperiode worden uitgevoerd. Alle portefeuilles die we trekken zijn “beheerd” volgens een buy-and-hold strategie, dat wil zeggen dat de gewichtsverdeling aan het begin vaststaat en gedurende de meetperiode verandert door het verloop van koersen. We zullen verderop kort ingaan op de nadelen hiervan en op de mogelijkheden voor random trekkingen mét transacties.

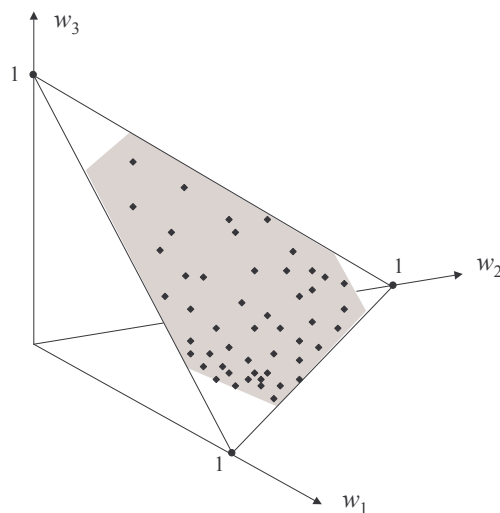
Merk op dat deze drie typen constraints wel degelijk relevant zijn voor performance berekening. Is bijvoorbeeld een belegger niet gebonden aan niet-negatieve exposures, dan geeft dat meer mogelijkheden dan hetzelfde mandaat mét die beperking. Een bijbehorende POD zou er dan ook anders uit moeten zien. Desalniettemin verwaarlozen we de mandaatregels die in deze categorieën vallen, omdat zij lastiger te hanteren zijn voor de trekkingsalgoritmes, die in de volgende paragrafen worden besproken.

## Kansverdelingen

Zoals in paragraaf 3.1 is aangegeven zijn er in feite twee keuzes voor de kansverdeling op de toegelaten verzameling. Wanneer we volgens de uniforme verdeling trekken is de dichtheid constant op dit gebied. De trekkingen worden dan gelijkmatig verspreid, zoals in figuur 19. Wanneer uit de marktverdeling wordt getrokken hebben portefeuilles een grotere “kans” om getrokken te worden naarmate zij een grotere exposure aannemen in objecten met een grotere marktkapitalisatie. Dit zien we in figuur 20, waarbij de kapitalisatie van object 1 het grootst is, gevolgd door object 2 en object 3.



Figuur 19: Uniforme trekking.



Figuur 20: Kapitalisatie-gewogen trekking.

We zullen van beide kansverdelingen een formulevorm geven. Ingeval we de uniforme verdeling prefereren is de dichtheid in ieder punt  $w \in S_N^{\text{Mdt}}$  gelijk aan

$$f_w(w) = 1 / \int \dots \int_{\bar{w} \in S_N^{\text{Mdt}}} d\bar{w}^1 \dots d\bar{w}^{N-1}.$$

We gebruiken in de formule het symbool  $\bar{w}$  als dummy variabele in de integraal. De integratievariabelen in de noemer gaan slechts tot  $\bar{w}^{N-1}$  omdat  $\bar{w}^N$  volledig bepaald is door de overige coördinaten:  $\bar{w}^N = 1 - \bar{w}^1 - \dots - \bar{w}^{N-1}$ . We integreren dus over de projectie van  $S_N^{\text{Mdt}}$  op  $S_N^{\text{Proj}}$  en de noemer is gelijk aan het volume hiervan.

Ingeval de voorkeur uitgaat naar een kapitalisatie-gewogen verdeling, verwachten we van de dichtheid van deze verdeling dat die stijgend is in de **marktkapitalisatie van de portefeuille**. Dit is het getal  $\sum_{j=1}^N w^j a^j$  indien de portefeuille gekarakteriseerd is door de vector  $w \in S_N^{\text{Mdt}}$ . Hierbij is  $a^j$  de marktkapitalisatie van beleggingsobject  $j$ . Door deze eis te stellen zullen portefeuilles met veel exposure in objecten met grote kapitalisatie eerder getrokken worden.

Een voor de hand liggende dichtheid die aan bovenstaande eis voldoet is de lineaire dichtheid, gegeven in een punt  $w \in S_N^{\text{Mdt}}$  door de volgende formule:

$$f_w(w) = \frac{\sum_{j=1}^N w^j a^j}{\int \dots \int_{\bar{w} \in S_N^{\text{Mdt}}} \sum_{j=1}^N \bar{w}^j a^j d\bar{w}^1 \dots d\bar{w}^{N-1}} \sim \sum_{j=1}^N w^j a^j .$$

Deze verdeling is dus in ieder punt evenredig met de kapitalisatie van de portefeuille bepaald door dit punt (genoteerd met het symbool  $\sim$ ). In het bijzonder verhouden de “extreme punten” zich volgens de kapitalisaties: de dichtheid in een punt corresponderend met een 100% exposure in object  $j$  is evenredig met  $a^j$ . Dit is een gewenste eigenschap, gezien de interpretatie van de getallen  $a^j$ .

Echter, de lineaire dichtheid hierboven is één van de mogelijke keuzes. Er zijn veel meer verdelingen denkbaar die meer gewicht toekennen aan portefeuilles met een grotere kapitalisatie en die zich in de randpunten verhouden volgens de kapitalisaties  $a^j$ .

Daar komt bij dat nog een andere eigenschap wenselijk is voor een marktgewogen verdeling. Het zou namelijk prettig zijn als het gemiddelde (of mediaan) van de getrokken portefeuille gelijk is aan de benchmark portefeuille (mits er geen verdere restricties zijn, en dus de toegelaten verzameling gevormd wordt door de gehele simplex). De benchmark portefeuille heeft normaal gewichten die zich onderling verhouden volgens de kapitalisaties  $a^j$ . Het verwachte rendement van de POD is dan ook gelijk aan het rendement van de benchmark (immers het rendement is een lineaire transformatie op de gewichtsvector). We kunnen dan met recht zeggen dat de POD in feite de benchmark is, met “daaromheen” een kansverdeling.

Helaas echter voldoet de hierboven gegeven lineaire verdeling niet aan deze wens. Om dat te zien beschouwen we het geval  $N=2$ . In dit geval is  $W^2 = 1 - W^1$  en is de kapitalisatie-gewogen dichtheid van  $W^1$  gegeven door

$$f_{W^1}(w^1) = \frac{a^2 + (a^1 - a^2)w^1}{\int_0^1 a^2 + (a^1 - a^2)\bar{w}^1 d\bar{w}^1} = 2 \frac{a^2 + (a^1 - a^2)w^1}{a^1 + a^2} .$$

We zouden wensen dat  $\mathbb{E}W^1 = a^1 / (a^1 + a^2)$ , zodat de verhouding van de verwachtingen  $\mathbb{E}W^1 / \mathbb{E}W^2 = a^1 / a^2$  is. Dit komt niet precies zo uit, immers:

$$\mathbb{E}W^1 = \frac{2}{a^1 + a^2} \int_0^1 a^2 w^1 + (a^1 - a^2)(w^1)^2 dw^1 = \frac{2}{a^1 + a^2} \left( \frac{1}{2} a^2 + \frac{1}{3} (a^1 - a^2) \right) = \frac{2a^1 + a^2}{3(a^1 + a^2)} .$$

Een lineaire transformatie op de vector  $(a^1, a^2)$  loslaten werkt ook niet, want dan kan een dichtheid met negatieve waarden ontstaan.

We lopen hier dus tegen een definitieprobleem aan. Welke verdeling voldoet het beste als kapitalisatie-gewogen verdeling? Wat we graag willen is een verdeling waarvoor

- de kans op trekken (in een continue setting de dichtheid) groter is naarmate de kapitalisatie van de portefeuille groter is;
- de kansen in de extreme punten, met 100% exposure in  $j$ , zich verhouden volgens  $a^j$ ;
- het gemiddelde gelijk is aan de vector waarvan de gewichten zich verhouden volgens  $a^j$ .

Helaas is er tot nu toe geen verdeling gevonden die aan al deze voorwaarden voldoet. De lineaire verdeling voldoet slechts aan de eerste twee. En als er een dichtheid zou bestaan die aan alle drie voldoet, dan is dit zeker geen lineaire dichtheid. Welke interpretatie kunnen we dan aan deze dichtheid verbinden? Dit definitieprobleem geeft een argument om helemaal niet uit een marktgewogen verdeling te trekken, maar altijd de uniforme verdeling toe te passen.

Desalniettemin zullen we in het vervolg van de scriptie enkele keren de POD methode toepassen met marktweigingen. We gebruiken dan de lineaire verdeling. Zoals we zullen zien is dit mogelijk met tenminste één van de te behandelen trekkingsalgoritmes.

### Effecten van mandaatregels

We besluiten deze paragraaf met enkele opmerkingen over de effecten van individuele regels van het mandaat. Door gebruik te maken van POD's is het mogelijk om inzicht te verkrijgen in de invloed die bepaalde constraints hebben op de behaalde performance. Dit kan men doen door een POD te simuleren voor zowel het volledige mandaat, als een mandaat waarbij één of meerdere constraints zijn weggelaten (een **gerelaxeerd mandaat**). Door voor beide mandaten bepaalde kengetallen, zoals gemiddelde, variantie, POD-ranking en overschrijdingskansen van toetsen te berekenen, en deze met elkaar te vergelijken, kan bekeken worden of de constraints voor de manager een grote beperking vormden, of juist niet. Dit geeft extra inzichten in hoe de manager heeft gepresteerd.

Een belangrijke vraag is wat er in het algemeen gebeurt met de performance wanneer er aan het mandaat een extra constraint wordt toegevoegd. Concreter, stel dat twee managers A en B over dezelfde periode hetzelfde mandaat opgelegd hebben gekregen (onafhankelijk van elkaar), met één verschil. Het mandaat van manager A bevat één extra constraint die het mandaat van B niet heeft. Veronderstel ook dat beide managers over de periode hetzelfde rendement van  $x\%$  hebben behaald. Wie van de twee heeft de beste performance geleverd? Op het eerste gezicht kan men opmerken dat manager B meer mogelijkheden heeft gehad dan A, omdat hij minder beperkingen had. Om die reden zou men van B een hoger rendement verwachten dan van A. Dit heeft hij niet gehaald, en dus heeft A een betere performance geleverd.

Helaas gaat de redenering hierboven niet altijd op! Intuïtief kan dit worden verklaard door het feit dat manager B met zijn extra mogelijkheden zowel slimme als domme dingen kan doen. Tot de extra mogelijkheden behoren dus zowel hele hoge als hele lage rendementen.

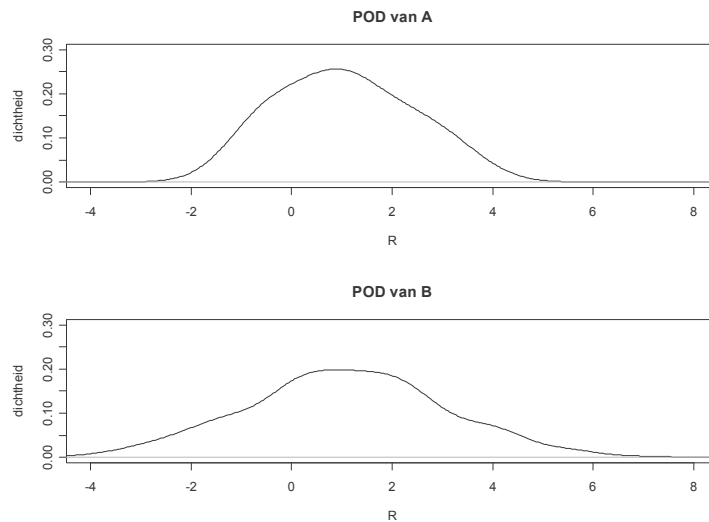
Wie van de twee managers uiteindelijk het beste heeft gepresteerd hangt af van het effect van de extra constraint op het toegelaten gebied, en ook van de exacte waarde van het gezamenlijke rendement  $x$ . POD's maken dit inzichtelijk. We onderscheiden de volgende gevallen (waarvan uiteraard ook combinaties kunnen voorkomen):

- Het gemiddelde van de POD verandert;
- De spreiding van de POD verandert.

Wanneer door toevoeging van een extra constraint het POD gemiddelde lager wordt, en de spreiding (en de vorm van de verdeling) nagenoeg gelijk blijft, zullen performance maten zoals de POD ranking  $\theta$  (i.e. de kans dat de aap de manager verslaat) lager uitvallen voor

manager A en dus heeft hij de beste performance geleverd. Het omgekeerde gebeurt wanneer het gemiddelde hoger wordt. Merk op dat beide scenario's in de praktijk kunnen voorkomen. Immers, de extra constraint kan een stuk van het toegelaten gebied afsnijden waar de rendementen relatief hoog of juist laag zijn.

Wanneer de spreiding van de POD door de extra constraint wordt verlaagd, en het gemiddelde (en de vorm van de verdeling) nagenoeg gelijk blijft, dan hangt het van  $x$  af welke manager het beste heeft gepresteerd. Vergelijk de twee dichtheidschatters in figuur 21 met elkaar.



Figuur 21: PODs van beide managers.

Het gemiddelde is in beide gevallen ongeveer 1%, en de standaarddeviatie is ongeveer 1% voor A, en 2% voor B. Ingeval  $x$  relatief groot is, bijvoorbeeld 4%, dan valt de ranking  $\theta$  (de oppervlakte onder de grafiek rechts van het punt  $x$ ) voor A lager uit en dus is zijn performance beter. Is  $x$  relatief klein, bijvoorbeeld -2%, dan valt  $\theta$  voor B lager uit en dus is zijn performance beter. De omgekeerde uitkomsten krijgen we wanneer de constraint de spreiding verhoogt in plaats van verlaagt. In de praktijk kunnen beide scenario's zich voordoen. Een constraint zoals een bovengrens op exposure-gewichten zorgt ervoor dat portefeuilles geconcentreerd in slechts enkele beleggingsobjecten niet voor kunnen komen. Dit leidt tot een gedwongen spreiding over de objecten, zodat de standaarddeviatie van het rendement naar verwachting verlaagt. Een constraint zoals een bovengrens op het aantal positieve gewichten zorgt er daarentegen voor dat juist portefeuilles die sterk gespreid zijn worden uitgesloten, zodat de standaarddeviatie verhoogt.

We zien dus dat het van verschillende factoren afhangt wat het effect is van een extra constraint op de behaalde performance. In hoofdstuk 5 zullen we voor een concreet voorbeeld zien hoe de performance verandert bij toevoeging van een regel aan het mandaat.

## 4.2 Algoritme 1: Benadering van de POD via discretisatie

Deze paragraaf behandelt de eerste van de drie algoritmes die de POD benaderen. De meest voor de hand liggende manier om dit te doen is door eenvoudigweg alle toegelaten portefeuilles, ofwel gewichtsvectoren  $w = (w^1, \dots, w^N)$  in de eenheidssimplex  $S_N$ , langs te lopen (**enumeratie**) en de bijbehorende rendementen uit te rekenen. Maar dit zullen er – bij realistische mandaten – oneindig veel zijn.

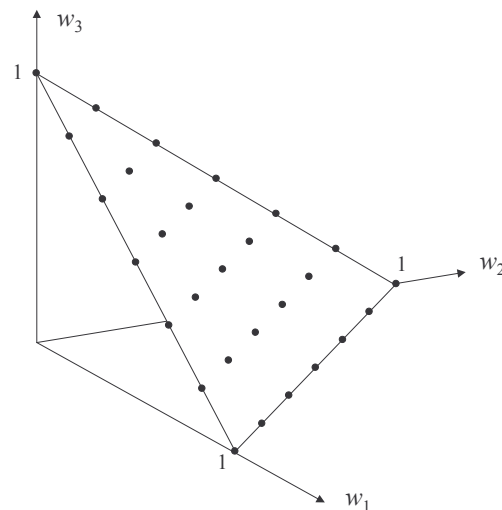
Het idee is dan om een **discretisatie** toe te passen. We nemen een klein getal  $d$ , de **discretisatiestap**, en we lopen alleen die portefeuilles langs waarvan alle gewichten een veelvoud zijn van  $d$ . Deze punten vormen een grid (rooster) in de simplex, om precies te zijn de verzameling

$$S_N^{\text{Discr}} := \left\{ w \in \mathbb{R}^N : \sum_{j=1}^N w^j = 1; w^j \geq 0; w^j / d \in \mathbb{Z}, j = 1, \dots, N \right\},$$

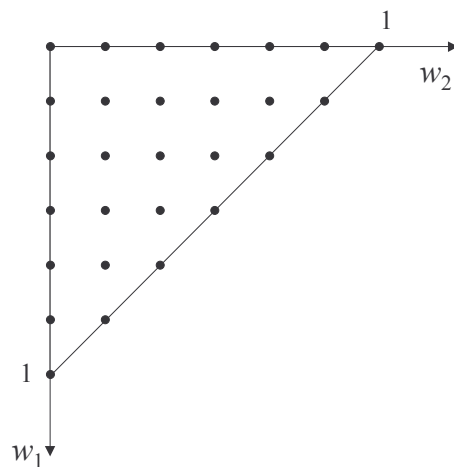
zie figuur 22. Ook deze verzameling wordt volledig bepaald door zijn projectie

$$S_N^{\text{Discr, Proj}} := \left\{ (w^1, \dots, w^{N-1}) \in \mathbb{R}^{N-1} : \sum_{j=1}^{N-1} w^j \leq 1; w^j \geq 0; w^j / d \in \mathbb{Z}, j = 1, \dots, N-1 \right\}$$

op  $\mathbb{R}^{N-1}$ , zie figuur 23. De verdeling, bepaald door de rendementen van de punten in  $S_N^{\text{Discr}}$ , levert een goede benadering van de POD, mits  $d$  niet te groot is. Uiteraard moet  $d$  zo gekozen zijn dat het totaal van 100% deelbaar is door  $d$  (dus  $1/d \in \mathbb{N}$ ).



Figuur 22: Gediscrètiseerde eenheidssimplex.



Figuur 23: Projectie van de gediscrètiseerde eenheidssimplex.

In de praktijk blijkt dat exposures van beneden de 0.01% zelden voorkomen. Het lijkt daarom redelijk om als doelstelling het getal  $d = 0.01\%$  (ofwel  $d = 10^{-4}$ ) te kiezen. In veel gevallen zullen de praktische exposure gewichten dan (ongeveer) een veelvoud zijn van  $d$ .

We bespreken nu een methode om alle punten uit de gediscretiseerde eenheidssimplex langs te lopen. Dit lijkt in eerste instantie lastig omdat de coördinaten van ieder punt  $w$  tot 1 moeten sommeren. Het idee is om via een zoekprocedure deze punten te vinden. We zullen daarbij gebruik maken van **recursie**, dat wil zeggen het aanroepen van een procedure binnen de definitie van dezelfde procedure. Eerst formuleren we het algoritme, vervolgens maken we enkele opmerkingen over de werking. Definiëren we de volgende hulpvariabelen:

$w$  := huidige gewichtsvector;  
 $j$  := huidige coördinaat die gekozen wordt;  
 $som$  := som van de nog te kiezen coördinaten,

dan ziet de recursieve procedure er in pseudo-programmeertaal als volgt uit:

```

LoopOverSimplex( j, som )
{
  if j = N
  {
     $w^j := som$  ;
    'Verwerk portefeuille w verder' ;
  }
  else for i:=0 to som/d
  {
     $w^j := i \cdot d$  ;
    LoopOverSimplex( j+1, som - i \cdot d ) ;
  }
}
  
```

De aanroep hieronder genereert alle portefeuilles in de gediscretiseerde eenheidssimplex:

```
LoopOverSimplex(1,1) ;
```

We maken de opmerking dat de procedure in de praktijk stabielier blijkt te werken ingeval we deze als volgt aanpassen:

```

LoopOverSimplex( j, som )
{
  if j = N
  {
     $w^j := som \cdot d$  ;
    'Verwerk portefeuille w verder' ;
  }
  else for i:=0 to som
  {
     $w^j := i \cdot d$  ;
    LoopOverSimplex( j+1, som - i ) ;
  }
}
  
```

De aanroep wordt in dit geval:

```
LoopOverSimplex(1,1/d) ;
```

In deze aanpassing is het tweede argument  $som$  altijd geheel en geeft dit aan hoe vaak de discretisatiestap  $d$  binnen de 100% valt. Er vindt nu een deling in de aanroep plaats, maar



niet meer in het for-statement, met als gevolg dat per gewichtsvector de som van de gewichten slechts verwaarloosbaar afwijkt van de 100%. Bij deling in het for-statement blijken (door afrondfouten) significante afwijkingen te ontstaan.

Het statement ‘Verwerk portefeuille  $w$  verder’ dient ervoor te zorgen dat we, telkens wanneer we een vector  $w$  uit de simplex hebben gegenereerd, alle informatie met betrekking tot  $w$  die we nodig hebben wordt berekend en bewaard. Er is echter een set constructieregels gegeven en dus zal niet van iedere portefeuille de return in de dataset moeten worden opgenomen. We hanteren daarom een zogeheten **acceptatie/rejectie-methode**: voor iedere bezochte portefeuille controleren we of deze voldoet aan alle constraints van het mandaat. Is dit niet zo, dan gaan we meteen verder met de volgende portefeuille. Is dit wel zo, dan berekenen we het rendement van de portefeuille en nemen we deze op in de dataset. Het uitgangspunt is dat er een procedure beschikbaar is waarmee kan worden nagegaan of een gegeven  $w$  toegelaten is of niet.

Het hierboven beschreven algoritme zullen we in het vervolg aanduiden met **algoritme 1**. Dit algoritme benadert de POD ingeval de ‘‘aap’’ uniforme trekkingen uit de simplex doet. Het acceptatie/rejectie principe verwijdert de data die corresponderen met portefeuilles buiten het mandaat. De resterende data vormen een benadering van de POD ingeval de ‘‘aap’’ uniforme trekkingen uit de toegelaten verzameling doet.

## Complexiteit

Het aantal portefeuilles dat het algoritme langsloopt is, bij gegeven  $N$  en  $d$ , gelijk aan het aantal punten in de gediscretiseerde eenheidssimplex van dimensie  $N$  met discretisatiestap  $d$ . Uit de combinatoriek volgt dat, voor  $k := 1/d$ , dit aantal gelijk is aan het binomiaalcoëfficiënt

$$\binom{N+k-1}{k} = \frac{(N+k-1)!}{(N-1)!k!} = \frac{(k+1) \cdots (N+k-1)}{(N-1)!},$$

ofwel het aantal  $k$ -herhalingscombinaties uit  $N$  elementen. Met behulp van de Stirling formule kan dit getal ook als volgt worden geschreven:

$$\binom{N+k-1}{k} \sim \frac{1}{\sqrt{2\pi}} \frac{(N+k-1)^{N+k-1/2}}{(N-1)^{N-3/2} k^{k-1/2}}.$$

Tabel 5 hieronder geeft een aantal waarden hiervan voor verschillende  $N$  en  $d$ .

**Tabel 5: Aantal elementen van de gediscretiseerde simplex.**

$N \setminus d$	0.01%	0.1%	1%
2	10001	1001	101
3	50015001	501501	5151
4	$5.0031 \cdot 10^{11}$	167668501	176851
5		$4.2085 \cdot 10^{10}$	4598126
6			96560646
7			1705904746
8			$2.6076 \cdot 10^{10}$

Deze waarden zijn niet alleen met de formule uitgerekend, maar ook geteld door het uitvoeren van algoritme 1 met behulp van het DOS programma dat aan het begin van dit hoofdstuk is genoemd. Dit programma is ongeveer 5 minuten bezig met rekenen ingeval het aantal portefeuilles gelijk is aan  $10^6$ . Merk echter op dat het wegschrijven van rendementen aanzienlijk meer tijd kost dan het rekenwerk. Bijgevolg doet het programma iets meer dan 1 minuut over het langslipen van  $10^6$  vectoren indien we het schrijven van uitvoer achterwege laten. Verder is het evident dat de rekentijd ongeveer lineair afhangt van het aantal portefeuilles dat langselopen moet worden.

We zien in de tabel dat het aantal portefeuilles exponentieel stijgt in de dimensie  $N$ . In de praktijk komen echter waarden van  $N$  boven de 500 voor! Dat betekent dat het hierboven beschreven algoritme praktisch onuitvoerbaar wordt wanneer  $N$  groot is – niet alleen vanwege de rekentijd, maar ook omdat de geheugenruimte, gebruikt door de recursie, boven elke praktische grens stijgt. Zoals we in de tabel kunnen zien helpt het niet veel als we  $d$  groter kiezen, bijvoorbeeld 0.1% of 1%.

Bovendien schuilt in het nemen van een grote  $d$  het gevaar dat punten op de rand van de simplex te zwaar meewegen in de benaderde POD. Dit komt doordat deze punten zeer hoge of lage rendementen kunnen geven vanwege de grote exposure in een beperkt aantal objecten. Het gevolg is dat de spreiding van de verdeling te groot uitvalt. Dit effect treedt het sterkst op wanneer bijv.  $d = 10\%$ . Het valt nog mee indien  $d$  gelijk is aan 0.1% of 1%.

Om algoritme 1 praktisch uit te kunnen voeren zal  $N$  klein moeten zijn (zeg hooguit 5) en zal  $d$  als volgt gekozen moeten worden:

- $N = 2$ :  $d = 0.01\%$ ;
- $N = 3$ :  $d = 0.1\%$ ;
- $N = 4,5$ :  $d = 1\%$ .

De rekentijd zal een aantal minuten bedragen voor  $N = 5$  en minder dan een minuut voor kleinere dimensie.

## Testen

Iedere implementatie van algoritme 1 dient getest te worden op correcte werking, voordat deze op echte mandaten wordt toegepast. Voor de huidige implementatie (het DOS programma) zijn er verschillende tests uitgevoerd. Er is een testrapport waarin niet alleen de resultaten beschreven staan, maar ook hoe de tests algemeen kunnen worden uitgevoerd (voor een willekeurige implementatie). Door ruimtegebrek is echter besloten dit testrapport niet op te nemen in de scriptie. Eventueel is het rapport op aanvraag beschikbaar.

Overigens gelden deze opmerkingen ook voor de andere twee algoritmes die in de volgende paragrafen ter sprake komen.

## Uitbreiding met grenzen op gewichten

Aan het probleem van een te groot aantal vectoren in de simplex valt wel iets te doen ingeval het mandaat stelt dat de exposure-gewichten  $w^j$  dienen te liggen tussen een benedengrens  $w^{j, LB}$  en een bovengrens  $w^{j, UB}$ . We kunnen dan in het script het for-statement

```
for  $i := 0$  to  $som/d$ 
```

vervangen door

$$\text{for } i := \max(w^{j,\text{LB}}, \text{som} - \sum_{k=j+1}^N w^{k,\text{UB}}) / d \quad \text{to } \min(w^{j,\text{UB}}, \text{som} - \sum_{k=j+1}^N w^{k,\text{LB}}) / d.$$

Er worden dan alleen punten langsgelopen die voldoen aan de onder- en bovengrenzen. Dit zijn er mogelijk aanzienlijk minder dan alle punten in de simplex, waardoor het algoritme sneller werkt.

### **Mogelijke uitbreiding naar marktgewogen verdeling**

Omdat algoritme 1 alle punten in het rooster binnen de simplex langsloopt, en deze punten gelijkmatig verspreid binnen de simplex liggen, is algoritme 1 bij uitstek geschikt om de POD te benaderen ingeval we volgens de uniforme verdeling over het toegelaten gebied willen trekken. In feite doet algoritme 1 geen random trekkingen, maar genereert het alle portefeuilles die onderling op een gelijke afstand van hun buurpunten liggen. De empirische verdelingsfunctie van de bijbehorende rendementen benadert derhalve de (onbekende) verdelingsfunctie van de POD, mits deze gedefinieerd is via uniforme trekkingen.

De vraag is of (en zo ja hoe) algoritme 1 kan worden uitgebreid naar het geval waarin een kapitalisatie-gewogen verdeling (de lineaire variant) gewenst is. Dit lijkt niet eenvoudig omdat dan de portefeuilles niet gelijkmatig over het gebied worden gegenereerd, maar dichter bij elkaar liggen op plaatsen waar de dichtheid van de gewogen verdeling een grotere waarde heeft.

De enige mogelijkheid die voor de hand lijkt te liggen is door portefeuilles meer dan één keer op te nemen in de resulterende dataset. Daartoe wordt eerst de getallenlijn van alle voorkomende marktkapitalisaties gediscrètiseerd, zeg met een stap  $d_{\text{kap}}$ . Van iedere gegenereerde portefeuille wordt het rendement  $m$  keer toegevoegd aan de dataset indien de marktkapitalisatie van de portefeuille benaderd wordt door  $md_{\text{kap}}$ . Om de POD op deze manier goed te kunnen benaderen moet echter  $d_{\text{kap}}$  relatief klein worden gekozen zodat de dataset al snel erg groot wordt. We hebben gezien dat er voor grote dimensie er veel portefeuilles langsgelopen worden. Als we dan ook nog bepaalde rendementen heel vaak (wellicht 100 of 1000 keer) toevoegen neemt de dataset een ontoelaatbaar formaat aan.

Om deze redenen werken we het idee niet verder uit. In paragraaf 4.4 komt een algoritme aan bod waarmee kapitalisatie-gewogen trekkingen verricht kunnen worden.

### **Opmerkingen ten aanzien van de bruikbaarheid**

We concluderen dat algoritme 1 mogelijk bruikbaar is ingeval de uniforme verdeling gewenst is, en voor mandaten waarbij  $N$  hooguit 5 is, en waarbij de keuzes voor gewichten aanzienlijk beperkt zijn door onder- en bovengrenzen. Een voorbeeld van zo'n mandaat is de tactische asset allocatie (de top van de IDP boom). In andere gevallen echter lijkt algoritme 1 praktisch niet erg bruikbaar. We zullen derhalve in de komende paragrafen een andere techniek introduceren: Monte Carlo simulatie. In plaats van de POD analytisch te benaderen, wordt deze benaderd door het trekken van random portefeuilles uit de toegelaten verzameling. Op die manier wegen alle portefeuilles mee en worden geen gebieden uitgesloten, mits we een voldoende groot aantal trekkingen uitvoeren.

### 4.3 Algoritme 2: Een lineaire procedure voor uniforme trekkingen

In deze en de volgende paragraaf bespreken we de mogelijkheden om stochastisch een groot aantal portefeuilles uit een gegeven mandaat te trekken. Ingeval de keuze voor de kansverdeling de uniforme is, bestaat er een eenvoudige (en snelle) procedure hiervoor. Deze procedure is gebaseerd op de volgende stelling.

**Stelling.** Gegeven zijn  $N$  onafhankelijke trekkingen  $X^1, \dots, X^N$  uit de exponentiële verdeling met parameter 1. Laat  $S := \sum_{j=1}^N X^j$ , en  $W$  de stochastische vector met coördinaten  $W^j$  gelijk aan  $X^j / S$ . Dan vormt  $W$  een uniforme trekking uit de eenheidssimplex  $S_N$ .

**Bewijs.** Het is evident dat  $W$  in de eenheidssimplex ligt, aangezien alle  $W^j > 0$  en de coördinaten van  $W$  worden genormeerd door hun som. Omdat  $W^N = 1 - \sum_{j=1}^{N-1} W^j$ , is de bewering dat  $W$  uniform verdeeld is op de simplex, equivalent met de bewering dat de vector  $(W^1, \dots, W^{N-1})$  uniform verdeeld is op de geprojecteerde verzameling

$$S_N^{\text{Proj}} := \left\{ (w^1, \dots, w^{N-1}) \in \mathbb{R}^{N-1} : \sum_{j=1}^{N-1} w^j \leq 1; w^j \geq 0, j=1, \dots, N-1 \right\}.$$

Neem een punt  $(w^1, \dots, w^{N-1}) \in S_N^{\text{Proj}}$ . We tonen aan dat de simultane verdelingsfunctie in dit punt evenredig is met het product  $w^1 \cdots w^{N-1}$ , de oppervlakte van het rechthoek  $[0, w^1] \times \cdots \times [0, w^{N-1}]$ . Bijgevolg is  $(W^1, \dots, W^{N-1})$  uniform verdeeld op  $S_N^{\text{Proj}}$ . We maken gebruik van de volgende lineaire transformatie, waarvan de Jacobiaan (d.w.z. de determinant van de afgeleidenmatrix) gelijk is aan 1:

$$\begin{pmatrix} z_1 \\ \vdots \\ z_{N-1} \\ z_N \end{pmatrix} \mapsto \begin{pmatrix} x_1 \\ \vdots \\ x_{N-1} \\ x_1 + \dots + x_N \end{pmatrix}.$$

De berekening is als volgt:

$$\begin{aligned} \mathbb{P}(W_1 \leq w_1, \dots, W_{N-1} \leq w_{N-1}) &= \mathbb{P}(X_1 / \sum_{j=1}^N X_j \leq w_1, \dots, X_{N-1} / \sum_{j=1}^N X_j \leq w_{N-1}) \\ &= \int_{x_1, \dots, x_N} \dots \int e^{-(x_1 + \dots + x_N)} dx_1 \cdots dx_N = \int_{z_1, \dots, z_N} \dots \int e^{-z_N} dz_1 \cdots dz_N \\ &\quad \left( \begin{matrix} x_1 / \sum_{j=1}^N x_j \\ \vdots \\ x_{N-1} / \sum_{j=1}^N x_j \end{matrix} \leq \begin{pmatrix} w_1 \\ \vdots \\ w_{N-1} \end{pmatrix} \right) \quad \left( \begin{matrix} z_1 / z_N \\ \vdots \\ z_{N-1} / z_N \end{matrix} \leq \begin{pmatrix} w_1 \\ \vdots \\ w_{N-1} \end{pmatrix} \right) \\ &= \int_0^\infty \int_0^{w_{N-1} z_N} \dots \int_0^{w_1 z_N} e^{-z_N} dz_1 \cdots dz_N = \int_0^\infty (w_{N-1} z_N) \cdots (w_1 z_N) e^{-z_N} dz_N \\ &= w_1 \cdots w_{N-1} \int_0^\infty z_N^{N-1} e^{-z_N} dz_N = w_1 \cdots w_{N-1} \Gamma(N) = w_1 \cdots w_{N-1} (N-1)! \end{aligned}$$

De verdelingsfunctie is dus evenredig met het product  $w^1 \cdots w^{N-1}$ , met  $(N-1)!$  als evenredigheidsconstante. Hiermee is het bewijs voltooid.  $\square$

De stelling hierboven geeft een algoritme om één random portefeuille  $W$  uit de eenheidssimplex te genereren. Dit algoritme ziet er in pseudo-programmeertaal als volgt uit. De procedure gebruikt een array  $Exp$ , gevuld met exponentiële trekkingen.

```
TrekPortefeuille()
{
    S:=0;
    for j=1 to N
    {
        E ~ Exponentieel(1);
        Expj := E;
        S := S + E;
    }
    for j=1 to N
    {
        Wj := Expj / S;
    }
    'Verwerk portefeuille W verder';
}
```

Het statement 'Verwerk portefeuille  $w$  verder' zorgt er weer voor dat alle informatie over  $W$  die we nodig hebben wordt berekend en bewaard. Om met behulp van dit algoritme een uniforme trekking uit de *toegelaten* portefeuilleverzameling te verkrijgen, gebruiken we, net zoals in de vorige paragraaf, een acceptatie/rejectie methode: voor iedere getrokken portefeuille controleren we binnen de verwerkingsstap of deze voldoet aan alle constraints. Zo ja, dan accepteren we de trekking, zo nee, dan trekken we opnieuw. Het algoritme dat dan verkregen is zullen we voortaan **algoritme 2** noemen.

Algoritme 2 zal op zich goed werken, tenzij de regels van het mandaat zo beperkend zijn, dat slechts een klein deel van de eenheidssimplex toegelaten is. Met name constraints die restricties leggen op het aantal positieve gewichten zorgen ervoor dat een aanzienlijk deel van de simplex afvalt. Immers, regels van dit type verlagen de dimensie van het toegelaten gebied. In dat geval zou er bij de acceptatie/rejectie met een grote kans niet worden geaccepteerd, zodat het veel tijd kan kosten om een voldoende groot aantal trekkingen uit het mandaat te doen.

Zoals we echter hieronder zullen zien, bestaat er op dit algoritme een uitbreiding waarmee automatisch trekkingen worden gegenereerd die voldoen aan de restrictie op het aantal positieve gewichten, vóórdat de acceptatie-rejectie in werking treedt. Desondanks kunnen er nog andere constraints zijn die het toegelaten gebied aanzienlijk verkleinen, bijvoorbeeld restricties op de ex-ante tracking error, of op exposure gewichten.

## Complexiteit

Het grote voordeel van bovenstaand algoritme is dat de benodigde rekentijd lineair afhangt van de dimensie  $N$ , en niet exponentieel, zoals bij algoritme 1 uit de vorige paragraaf. Uit tests blijkt dat, voor  $N$  in de orde van 500 of 1000, het enkele minuten duurt om 100000 trekkingen te doen. Deze tests zijn uitgevoerd in het eerder genoemde DOS programma.

Het is dus mogelijk om binnen afzienbare tijd miljoenen trekkingen te verrichten. Problemen met geheugenruimte treden, zoals bij algoritme 1, niet op omdat er geen informatie gedurende het hele trekkingsproces bewaard moet blijven. De gebruikte variabelen krijgen telkens nieuwe waarden wanneer er een nieuwe trekking wordt gedaan.

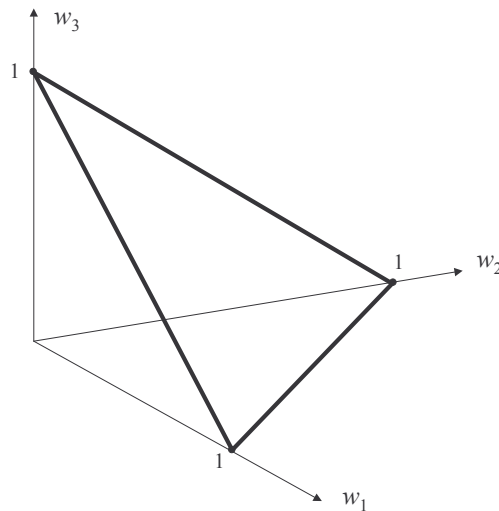
## Testen

Zoals hierboven al opgemerkt is ook de huidige implementatie van algoritme 2 aan een aantal tests onderworpen. Behalve het meten van de rekentijd is de methode ook getest om na te gaan of de werking juist is. Deze tests staan in het testrapport dat niet in deze scriptie is opgenomen.

## Uitbreiding met maximum aantal positieve gewichten

We weten inmiddels dat algoritme 2 niet erg bruikbaar is wanneer het mandaat zo restrictief is dat slechts een zeer klein deel van de simplex toegelaten is. Toch valt er wel iets te verbeteren voor specifieke constraints.

Stel dat  $N$  de dimensie van het mandaat is en dat er een bovengrens  $N^{\max} < N$  op het aantal positieve gewichten gegeven is. Wanneer we de simplex in  $N$  dimensies verminderen met alle vectoren met meer dan  $N^{\max}$  positieve gewichten, houden we een deel van de rand over. Deze rand bestaat uit een aantal even grote simplices van dimensie  $N^{\max}$  die zijdelings met elkaar verbonden zijn. Figuur 24 hieronder laat de rand (dikke lijn) zien ingeval  $N = 3$  en  $N^{\max} = 2$ .



Figuur 24: Toegelaten gebied als rand van de simplex.

Het uniform trekken uit het deel van de rand is dus equivalent met eerst uniform kiezen van één zo'n deelsimplex, en vervolgens uniform trekken uit deze deelsimplex. Het trekken van een deelsimplex komt overeen met het trekken, zonder teruglegging, van  $N^{\max}$  getallen uit de verzameling  $\{1, \dots, N\}$ , waarvoor er

$$\binom{N}{N^{\max}} = \frac{N!}{N^{\max}!(N - N^{\max})!}$$

mogelijkheden zijn. Deze getallen stellen de gewichten voor die positief kunnen zijn.

We vatten algoritme 2, voor mandaten met een bovengrens  $N^{\max}$  op het aantal positieve gewichten, als volgt samen:

1. Trek uniform, zonder teruglegging,  $N^{\max}$  elementen uit  $\{1, \dots, N\}$ .
2. Genereer een uniforme trekking uit de  $N^{\max}$ -dimensionale eenheidssimplex.
3. Maak een gewichtsvector  $w$  zodat voor alle in stap 1 getrokken getallen  $j$ , de coördinaat  $w^j$  van  $w$  gelijk is aan een coördinaat van de trekking uit stap 2, en de overige 0. Met behulp van een teller kan bepaald worden wanneer welke coördinaat van de trekking uit 2 wordt toegekend.
4. Pas acceptatie/rejectie toe op  $w$  en ga terug naar 1 totdat er voldoende trekkingen zijn verricht.

### Mogelijke uitbreiding naar marktgewogen verdeling

Net als voor algoritme 1 geldt voor algoritme 2 dat deze niet eenvoudig kan worden uitgebreid naar een soortgelijk algoritme, met als resulterende verdeling de kapitalisatie-gewogen verdeling. Er zijn wel enkele voor de hand liggende manieren, maar deze leveren niet precies de lineaire kansdichtheid

$$f(w) \sim \sum_{j=1}^N w^j a^j,$$

die in paragraaf 4.1 is gedefinieerd (herinner dat  $a^j$  de kapitalisatie van object  $j$  voorstelt). We noemen er twee:

- Normeer met  $S := \sum_{j=1}^N a^j X^j$  in plaats van  $S := \sum_{j=1}^N X^j$  en neem  $W^j := a^j X^j / S$ . (Dit is equivalent met  $X^j$  Exponentieel( $1/a^j$ ) trekken in plaats van Exponentieel(1), en normeren met de som  $S := \sum_{j=1}^N X^j$ .)
- Trek  $X^j$  uit de Gamma verdeling met schaal 1 en vorm  $a^j$ , en normeer met  $S := \sum_{j=1}^N X^j$ .

Al deze methoden zorgen ervoor dat objecten met een grotere marktkapitalisatie zwaarder meewegen bij het trekken uit de simplex. Maar de onderlinge verhoudingen waarin de objecten meewegen zijn niet precies evenredig met de kapitalisaties  $a^j$ . Bij de tweede methode is de resulterende verdeling een zogeheten Dirichlet verdeling. Deze verdeling heeft een dichtheid die 0 is op de rand van de simplex, en een “berg” vormt op het inwendige van de simplex. Dit is dus allesbehalve lineair in de gewichten.

Dat ook de eerste methode niet helemaal doet wat we willen laten we zien aan de hand van het voorbeeld met  $N = 2$ . In dit geval is  $W^2 = 1 - W^1$  en wensen we dat de dichtheid van  $W^1$  evenredig is met de volgende functie:

$$f_{W^1}(w^1) \sim a^2 + (a^1 - a^2)w^1.$$

Dus de dichtheid is lineair en de eindpunten verhouden zich als  $f_{W^1}(1)/f_{W^1}(0) = a^1/a^2$ . We laten zien dat dit in het algemeen niet zo uitkomt. Aangezien

$$W^1 = \frac{a^1 X^1}{a^1 X^1 + a^2 X^2},$$

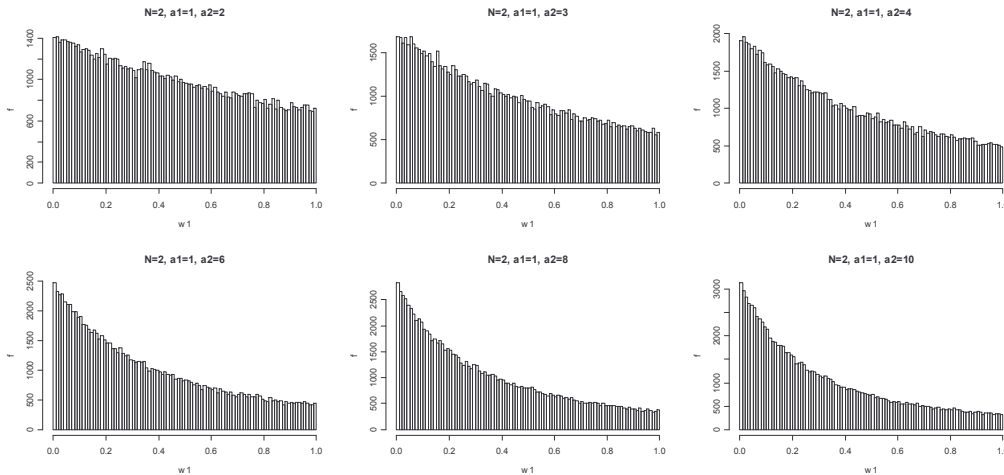
kan men berekenen dat

$$\begin{aligned}
\mathbb{P}(W^1 \leq w^1) &= \mathbb{P}\left(X^1 \leq \frac{a^2 w^1}{a^1(1-w^1)} X^2\right) = \int_0^{\frac{a^2 w^1}{a^1(1-w^1)} x^2} \int_0^\infty e^{-x^1 - x^2} dx^1 dx^2 \\
&= \int_0^\infty \left(1 - e^{-\frac{a^2 w^1}{a^1(1-w^1)} x^2}\right) e^{-x^2} dx^2 = \int_0^\infty (e^{-x^2} - e^{-\left(1 + \frac{a^2 w^1}{a^1(1-w^1)}\right) x^2}) dx^2 \\
&= 1 - \frac{1}{1 + \frac{a^2 w^1}{a^1(1-w^1)}} = \frac{a^2 w^1}{a^1(1-w^1) + a^2 w^1}.
\end{aligned}$$

De dichtheid  $f_{W^1}(w^1)$  is de afgeleide van deze functie naar  $w^1$ :

$$f_{W^1}(w^1) = \frac{d}{dw^1} \frac{a^2 w^1}{a^1(1-w^1) + a^2 w^1} = \frac{a^1 a^2}{(a^1(1-w^1) + a^2 w^1)^2},$$

hetgeen niet evenredig is met  $a^2 + (a^1 - a^2)w^1$ . En de verhouding van de eindpunten is  $f_{W^1}(1)/f_{W^1}(0) = (a^1)^2/(a^2)^2$ . Vervangen we echter  $a^j$  door  $\sqrt{a^j}$ , dan is deze verhouding wel  $a^1/a^2$ . Aldus ontstaat het idee dat het algoritme wellicht wel goed werkt wanneer we  $W^j \sim \sqrt{a^j} X^j$  nemen in plaats van  $W^j \sim a^j X^j$ . De verhouding van de eindpunten klopt dan in ieder geval. Of de dichtheid ook lineair is onderzoeken we empirisch. De histogrammen in figuur 25 geven een indruk van de verkregen dichtheid voor verschillende  $a^1, a^2$ .



Figuur 25: Histogram bij verschillende kapitalisatie-verhoudingen.

We concluderen voorzichtig dat de dichtheid een lineaire functie benadert mits de verhouding van  $a^1, a^2$  niet te groot is. Is deze verhouding wel groot, dan is de functie niet lineair, maar hebben we in ieder geval bereikt dat het object met de grootste kapitalisatie het zwaarst weegt (zij het niet in extreme proporties).

De vraag is hoe deze methode kan worden uitgebreid naar hogere  $N$ . Voor de hand ligt wederom om als schaalfactoren de  $a^j$  tot een bepaalde macht te nemen. Welke macht dat is hangt natuurlijk af van  $N$ . Empirisch is onderzocht dat de machten in tabel 6 redelijk werken:



Tabel 6: Gekozen machten.

$N$	macht
2	1/2
3	5/14
4	2/7
$\geq 5$	1/4

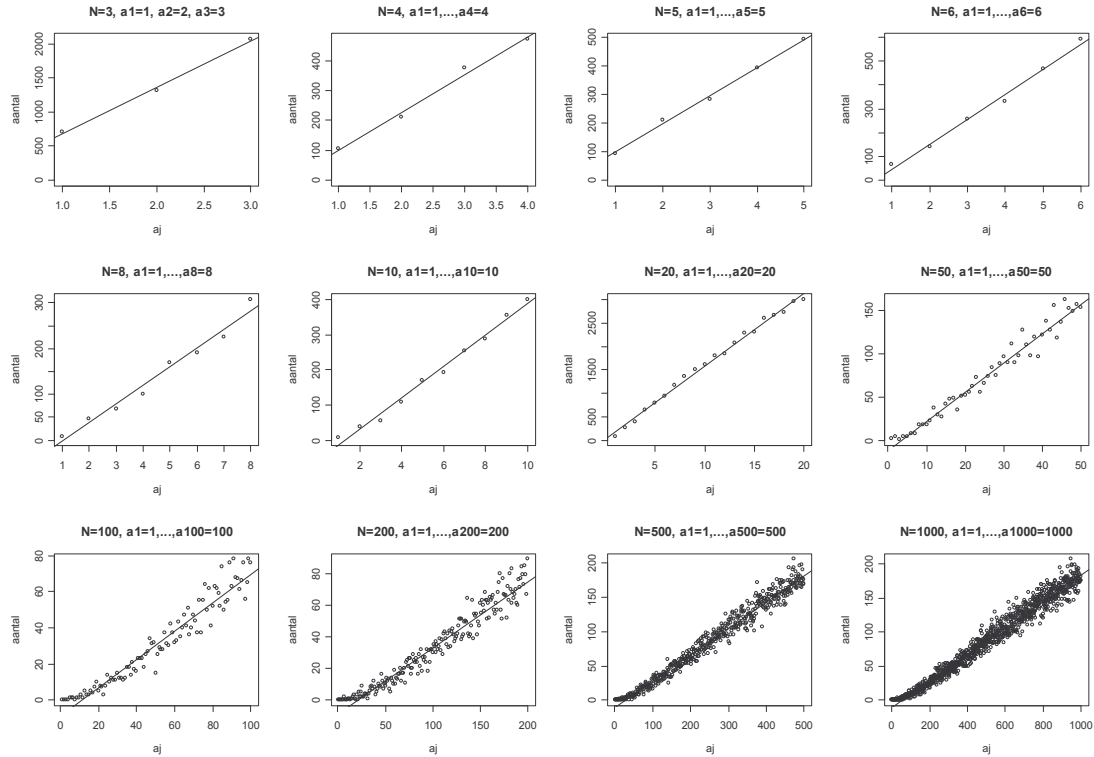
Dit is gedaan door de  $N$  marktkapitalisaties gelijk te nemen aan resp.  $1, \dots, N$ , en te kijken of de dichtheid in de hoekpunten van de simplex (die corresponderen met portefeuilles geconcentreerd in één object), zich inderdaad volgens de kapitalisaties  $1, \dots, N$  tot elkaar verhouden. Het schatten van de dichtheid in een hoekpunt  $i$  doen we door het aantal trekkingen te berekenen waarvoor het gewicht  $w^i$  groter is dan een bepaald percentage. Dit percentage nemen we kleiner naarmate  $N$  groter is. De berekende aantallen verhouden zich dan bij benadering zoals de dichtheid in de hoekpunten.

Tabel 7 toont de waarden van  $N$  waarvoor dit experiment is uitgevoerd, samen met de gekozen macht, de gebruikte grens op het gewicht, en het aantal gedane trekkingen.

Tabel 7: Gegevens per experiment.

$N$	macht	grens	trekkingen
3	-5/14	95%	500000
4	-2/7	92%	500000
5	-1/4	85%	500000
6	-1/4	78%	500000
8	-1/4	70%	500000
10	-1/4	60%	500000
20	-1/4	20%	100000
50	-1/4	10%	10000
100	-1/4	6%	10000
200	-1/4	3%	10000
500	-1/4	1%	10000
1000	-1/4	0.5%	10000

In figuur 26 hieronder staan de resultaten in de vorm van plots van de aantallen trekkingen die de gegeven gewichtsgrens in het betreffende hoekpunt hebben overschreden. Ook is steeds een kleinste kwadratenlijn toegevoegd.



Figuur 26: Aantal trekkingen die de gewichtsgrens in een hoekpunt hebben overschreden.

In de plots zien we dat de aantallen min of meer lineair stijgen, en zich dus per coördinaat bij benadering verhouden tot  $1, \dots, N$ , zodat de gekozen macht inderdaad als een goede keuze kan worden opgevat. Het enige wat niet uit deze resultaten volgt is wanneer de resulterende dichtheidsfunctie ook lineair is en dus lijkt op de gewenste dichtheid. Analooq aan het geval  $N = 2$  verwachten we dat dit bij benadering het geval zal zijn mits de kapitalisaties  $a^j$  niet extreem veel van elkaar afwijken. We hebben dus ook voor grotere  $N$  bereikt dat objecten met een grotere kapitalisatie zwaarder wegen bij het trekken, maar dat deze relatieve zwaarten niet in extreme proporties optreden.

We concluderen dat we een aardige variant op algoritme 2 hebben om uit een marktverdeling te kunnen trekken. Omdat echter de verkregen dichtheid niet precies die dichtheid is die we wensen, is de methode helaas niet helemaal volmaakt en zullen we deze niet gebruiken in het vervolg van de scriptie.

### Opmerkingen ten aanzien van de bruikbaarheid

Zoals we hebben gezien heeft algoritme 2 het nadeel dat er geen bevredigende variant is om uit de kapitalisatie-gewogen verdeling te trekken. Desalniettemin is de methode nuttig en vaak goed bruikbaar wanneer we uniform willen trekken. De rekentijd is lineair in de dimensie  $N$ , en voor honderdduizenden trekkingen kan dit in minuten gemeten worden, zelfs als  $N = 500$  of  $1000$ .

In de praktijk kan men echter tegen het probleem aanlopen dat de acceptatie/rejectie procedure zelden accepteert, zodat wellicht één trekking al veel tijd kost. Dit is niet te wijten aan een aanwezige bovengrens  $N^{\max}$  op de exposures groter dan nul (daarvoor hebben we een eenvoudige oplossing uitgewerkt), maar wel aan andere constraints zoals restricties op de tracking error of op exposure gewichten.

We concluderen dat algoritme 2 geschikt is voor mandaten met zowel kleine als grote  $N$ , mits deze die niet te beperkend zijn door mandaatregels naast de bovengrens  $N^{\max}$ . Veel mandaten zijn gebonden aan restricties op de ex-ante tracking error of aan een bovengrens  $w^{\text{UB}}$  die de  $N$  exposure gewichten niet mogen overschrijden. Wat het effect is van een tracking error restrictie is niet zo eenvoudig in kaart te brengen. Wel kunnen we de invloed van  $w^{\text{UB}}$  op de grootte van het toegelaten gebied onderzoeken. Naarmate dit gewicht en  $N$  lager worden neemt de kans op verwerpen (niet accepteren) van een trekking toe. De volgende tabel 8 geeft enkele waarden die uit tests voorkomen. Deze tabel geeft voor een aantal waarden van  $N$  en  $w^{\text{UB}}$  de verwerpingskans, ofwel de kans dat minstens één van de gewichten boven  $w^{\text{UB}}$  uitkomt. De tests zijn uitgevoerd in het DOS programma. De corresponderende mandaten kennen naast de bovengrens  $w^{\text{UB}}$  geen verdere restricties.

Tabel 8: Rejectie-kansen voor verschillende dimensies en bovengrenzen.

$N$	$w^{\text{UB}}$	rejectie-kans	$N$	$w^{\text{UB}}$	rejectie-kans
50	5%	99.89%	200	1.6%	99.998%
	4.5%	99.995%		1.5%	99.9998%
	4%	100.000%		1%	100.000%
100	3%	99.92%	500	1%	97.38%
	2.8%	99.991%		0.9%	99.82%
	2.7%	99.9975%		0.8%	99.9988%
	2.6%	99.9991%		0.7%	100.000%
	2.5%	100.000%		1000	0.5%
200	2%	98.7%	1000	0.45%	99.9996%
	1.7%	99.986%		0.4%	100.000%

In gevallen waarbij de rejectiekans boven de 99.999% uitkomt, wordt naar verwachting minder dan 1 op de 100000 portefeuilles geaccepteerd, zodat het minuten kan duren voordat er één trekking wordt gerealiseerd. Dit is bijvoorbeeld het geval voor  $N=100$  en  $w^{\text{UB}} \leq 2.6\%$ , en ook voor  $N=1000$  en  $w^{\text{UB}} \leq 0.45\%$ . Het is echter de vraag of in de praktijk mandaten voorkomen met zulke lage bovengrenzen  $w^{\text{UB}}$  (ten opzichte van  $N$ ). Mocht dit toch voorkomen, dan zal een andere trekkingsmethode gebruikt moeten worden, bijvoorbeeld het algoritme dat in de volgende paragraaf ter sprake komt. Uiteraard geldt dit ook wanneer een ander soort constraint (of een combinatie van meerdere constraints) de rejectiekans te groot maakt.

#### 4.4 Algoritme 3: Markov-keten Monte Carlo

Het eenvoudige algoritme, behandeld in paragraaf 4.3, is in veel gevallen bruikbaar maar heeft een aantal beperkingen waarvan de voornaamste zijn:

- geen bevredigende uitbreiding voor kapitalisatie-gewogen kansverdeling;
- te veel rekentijd bij mandaten die corresponderen met een zeer klein toegelaten gebied.

Deze beperkingen geven aanleiding om onderzoek te doen naar een methode die direct portefeuilles uit de toegelaten verzameling trekt, zonder gebruik van acceptatie/rejectie, en tevens de mogelijkheid biedt om uit de kapitalisatie-gewogen verdeling te trekken. Deze omvangrijke paragraaf is gewijd aan een dergelijke methode. We behandelen we de simulatietechniek van **Markov-keten Monte Carlo** (in het Engels: **Markov Chain Monte Carlo**, **MCMC**), zie onder andere [16]. Dit gebeurt eerst in algemene termen, vervolgens in termen van onze toepassing, namelijk portefeuilletrekking.

## Wat is MCMC?

Veronderstel dat er  $m$  uitkomsten van een zeker experiment gegeven zijn, evenals positieve getallen  $b_1, \dots, b_m$  met som  $B$ . Hoe kunnen we een trekking uit deze uitkomsten realiseren met kansverdeling gegeven door  $\pi_j = b_j / B$  voor alle  $j$ ? Vaak is  $B$  niet bekend of lastig te berekenen, bijvoorbeeld ingeval de uitkomsten meerdimensionale vectoren zijn en het aantal uitkomsten  $m$  niet eenvoudig te bepalen is.

Bovenstaande vraag kan worden beantwoord via het begrip ‘reversibele Markov-keten’. We roepen daartoe kort wat theorie over Markov-ketens in herinnering.

Een **Markov-keten** is een rij stochastische variabelen  $(X_n)_n = X_0, X_1, X_2, \dots$  met waarden in een zekere (discrete) **toestandsruimte**  $I$ , zodat de **Markov-eigenschap** geldt:

$$\mathbb{P}(X_{n+1} = k \mid X_n = j, X_{n-1} = j_{n-1}, \dots, X_0 = j_0) = \mathbb{P}(X_{n+1} = k \mid X_n = j),$$

voor iedere tijdsparameter  $n$  en toestanden  $j_0, \dots, j_{n+1} \in I$ . In woorden: de huidige toestand geeft voldoende informatie om de kansverdeling van de volgende toestand te bepalen. Verder veronderstellen we dat de keten **tijd-homogeen** is: de kansen  $\mathbb{P}(X_{n+1} = k \mid X_n = j)$  zijn hetzelfde voor iedere  $n$ . Bijgevolg is het stochastisch gedrag van de keten bepaald door de verdeling van  $X_0$ , en de **overgangskansen**

$$p_{jk} := \mathbb{P}(X_{n+1} = k \mid X_n = j),$$

$j, k \in I$ . De **overgangsmatrix**  $P := (p_{jk})_{jk}$  is de matrix met alle overgangskansen als elementen. We zijn met name geïnteresseerd in het gedrag van de keten wanneer we  $n$  naar oneindig sturen. Daarvoor hebben we enkele begrippen nodig.

- Een verzameling toestanden  $I' \subset I$  heet een **fuik** als  $p_{jk} = 0$  voor alle  $j \in I'$  en  $k \in I \setminus I'$ . M.a.w., bevindt de keten zich eenmaal binnen  $I'$ , dan zal de keten geen waarden buiten  $I'$  meer aannemen.
- Een reeks toestanden  $j_1, \dots, j_n \in I$  heet een **pad** als  $p_{j_t, j_{t+1}} > 0$  voor  $t = 1, \dots, n-1$ . De **lengte** van een pad  $j_1, \dots, j_n$  is gedefinieerd door het aantal overgangen  $n$ .
- Een keten heet **aperiodiek** als er een toestand  $j$  bestaat, zodat de grootste gemene deler van de lengtes van alle mogelijke paden van  $j$  naar zichzelf gelijk aan 1 is.

Voor onze doeleinden is het voldoende om het geval te beschouwen dat de toestandsruimte  $I$  eindig is. Maken we in dat geval de volgende aannames:

- er zijn geen twee disjuncte fuiken;
- de keten is aperiodiek,

dan geldt dat de limiet

$$\pi_k := \lim_{n \rightarrow \infty} \mathbb{P}(X_n = k \mid X_0 = j)$$

bestaat en onafhankelijk is van  $j$ . De vector  $\pi = (\pi_k)_k$  is een kansverdeling, de **limietverdeling**, en is de unieke oplossing van het stelsel vergelijkingen

$$\pi_j = \sum_{k \in I} \pi_k P_{kj},$$

$j \in I$ , in combinatie met  $\sum_{j \in I} \pi_j = 1$ . Voor een bewijs van dit basisresultaat verwijzen we naar [7]. Het bovenstaande stelsel vergelijkingen noemt men wel de **evenwichtsvergelijkingen**, en kan in vector-matrix notatie geschreven worden als

$$\pi = \pi P.$$

Nu behandelen we een tweetal stellingen die nuttig zijn in verband met het trekkingsprobleem.

**Stelling.** Zij  $(X_n)_n$  een aperiodieke Markov-keten met eindige toestandsruimte  $I$ ,  $m := \#I$ , overgangsmatrix  $P = (p_{jk})_{jk}$ , en geen twee disjuncte fuiken, zodat de limietverdeling  $\pi$  bestaat. Stel, er zijn getallen  $g_1, \dots, g_m$ , niet alle gelijk aan nul, zodat

$$g_j P_{jk} = g_k P_{kj},$$

voor alle  $j, k \in I$ . Dan bestaat er een  $c \neq 0$  zodat

$$g_j = c\pi_j,$$

voor alle  $j \in I$ .

**Bewijs.** Voor iedere  $j$  geldt

$$g_j = \sum_{k=1}^m g_j P_{jk} = \sum_{k=1}^m g_k P_{kj}.$$

Deze vergelijkingen vormen precies de evenwichtsvergelijkingen van de Markov-keten, en deze bepalen de  $g_j$  op een multiplicatieve constante na: er is een  $c \neq 0$  zodat

$$g_j = c\pi_j. \quad \square$$

Een onmiddellijk gevolg is dat, onder de aannames van de stelling,

$$\pi_j P_{jk} = \pi_k P_{kj}.$$

De keten  $(X_n)_n$  is dus **reversibel**: voor ieder tweetal toestanden  $j, k$  is op de lange duur het gemiddeld aantal overgangen van  $j$  naar  $k$  per tijdseenheid gelijk aan het gemiddeld aantal overgangen van  $k$  naar  $j$  per tijdseenheid.

Zij vervolgens  $(M_n)_n$  een nader te specificeren aperiodieke Markov-keten met toestandsruimte  $I = \{1, \dots, m\}$ , overgangskansen  $m_{jk}$ , voor  $j, k \in I$ , en geen twee disjuncte fuiken. We nemen aan dat  $m_{jk} > 0$  als en alleen als  $m_{kj} > 0$ , voor alle  $j, k \in I$  (de toestandsovergangen zijn dus omkeerbaar). Uit de keten  $(M_n)_n$  construeren we een **afgeleide**

**Markov-keten**  $(X_n)_n$  met dezelfde toestandsruimte door de overgangskansen  $p_{jk}$  als volgt te definiëren:

$$p_{jk} := \begin{cases} m_{jk}\alpha_{jk}, & j \neq k; \\ m_{jj} + \sum_{\ell \neq j} m_{j\ell}(1 - \alpha_{j\ell}), & j = k, \end{cases}$$

voor geschikt gekozen getallen  $\alpha_{jk} \in [0,1]$  met  $\alpha_{jj} = 1$  voor alle  $j$ .

**Stelling.** Kiezen we

$$\alpha_{jk} := \begin{cases} 0, & m_{jk} = 0 = m_{kj}; \\ \min \left\{ \frac{b_k m_{kj}}{b_j m_{jk}}, 1 \right\}, & m_{jk} \neq 0 \neq m_{kj}, \end{cases}$$

dan heeft  $(X_n)_n$  een limietverdeling  $\pi$  die gegeven is door

$$\pi_j = \frac{b_j}{B},$$

voor  $j \in I$ .

**Bewijs.** Omdat  $(M_n)_n$  geen twee disjuncte fuiken heeft en aperiodiek is, geldt dit ook voor  $(X_n)_n$ . De limietverdeling van  $(X_n)_n$  bestaat derhalve. Voordat we deze uitrekenen tonen we aan dat  $b_j p_{jk} = b_k p_{kj}$  voor alle  $j, k$ .

Ingeval  $j = k$  is dit triviaal, dus veronderstel  $j \neq k$ . Ingeval  $m_{jk} = 0 = m_{kj}$ , is  $\alpha_{jk} = 0$ , dus

$$p_{jk} = m_{jk} \cdot 0 = 0 = m_{kj} \cdot 0 = p_{kj},$$

dus ook

$$b_j p_{jk} = 0 = b_k p_{kj}.$$

Nu bewijzen we de bewering voor het geval  $m_{jk} \neq 0 \neq m_{kj}$ . Neem aan, zonder beperking der algemeenheid, dat  $b_k m_{kj} / b_j m_{jk} < 1$ . Dan is  $\alpha_{jk} = b_k m_{kj} / b_j m_{jk}$  en  $\alpha_{kj} = 1$ . Er geldt:

$$p_{jk} = m_{jk} \alpha_{jk} = m_{jk} \frac{b_k m_{kj}}{b_j m_{jk}} = \frac{b_k m_{kj}}{b_j},$$

en

$$p_{kj} = m_{kj} \alpha_{kj} = m_{kj} \cdot 1.$$

Er volgt

$$b_j p_{jk} = b_k m_{kj} = b_k p_{kj}.$$

Nu is aangetoond dat  $b_j p_{jk} = b_k p_{kj}$  voor alle  $j, k$ . Daarmee voldoet de keten  $(X_n)_n$  aan de voorwaarden van de voorgaande stelling met  $g_j = b_j$ , en dus is er een  $c \neq 0$  zodat  $b_j = c\pi_j$ . Aangezien  $\pi$  een kansverdeling is en dus de coördinaten tot 1 sommeren, concluderen we dat

$$\pi_j = \frac{b_j}{B},$$

voor  $j \in I$ . □

Deze stelling geeft een methode om een trekking te kunnen doen uit  $m$  uitkomsten met kansverdeling  $\pi_j = b_j / B$  voor alle  $j$ , het probleem waarmee we deze paragraaf begonnen! Het idee is om een Markov-keten te construeren met  $(\pi_j)_j$  als limietverdeling, en deze keten een bepaalde tijd te laten lopen zodat bij benadering uit de gewenste kansverdeling wordt getrokken. Het precieze algoritme is als volgt:

### Hastings-Metropolis algoritme:

0. Specificeer een geschikte gekozen Markov-keten  $(M_n)_n$  op de toestandruimte  $I = \{1, \dots, m\}$ , met overgangskansen  $m_{jk}$ , waarvoor  $m_{jk} > 0 \Leftrightarrow m_{kj} > 0$ ,  $j, k \in I$  en zodat de keten aperiodiek is en geen twee disjuncte fuiken heeft.

Neem een starttoestand  $j \in I$ .

Laat  $n := 0$  en  $X_0 := j$ .

Kies de looptijd  $n^{\text{end}}$  van de keten.

1. Genereer een kandidaat-toestand  $K$  volgens het mechanisme  $P(K = k) = m_{X_n, k}$ .  
Genereer een uniform(0,1) verdeeld getal  $U$ .
2. Als

$$U < \frac{b_K m_{K, X_n}}{b_{X_n} m_{X_n, K}},$$

dan  $H := K$ ;

en anders  $H := X_n$ .

3. Laat  $n := n + 1$ ;  
 $X_n := H$ ,

en ga naar 1, of stop als  $n = n^{\text{end}}$ .

$X_{n^{\text{end}}}$  is bij benadering een trekking uit de verdeling gegeven door  $\pi_j = b_j / B$ .

### Toepassing op portefeuilletrekking

We beschrijven nu hoe het bovenstaande Hastings-Metropolis algoritme wordt vertaald naar onze situatie, waarin we een willekeurige portefeuille uit een mandaat willen trekken. Om te beginnen zullen we, net als in paragraaf 4.2, de eenheidssimplex discretiseren met een bepaalde stap  $d$ . Als doelstelling kiezen we  $d = 0.01\%$ , hetgeen goede benaderingen oplevert. De toestandruimte nemen we gelijk aan de verzameling van alle punten uit de gediscretiseerde simplex, waarvan de corresponderende portefeuilles voldoen aan alle regels van het mandaat. We gaan er vanuit dat er een procedure `CheckMandaat(W)` beschikbaar is, die voor een gegeven gewichtsvector  $W$  bepaalt of deze voldoet of niet.

Welke limietverdeling  $\pi_j = b_j / B$  wij wensen hangt af van de kansverdeling volgens welke we uit het mandaat willen trekken:

- kapitalisatie-gewogen verdeling;
- uniforme verdeling.

Bij de kapitalisatie-gewogen verdeling (de lineaire variant) is voor iedere toestand  $j$  de verdelingsfactor  $b_j$  gelijk aan de marktkapitalisatie van de portefeuille corresponderend met  $j$ . Ervan uitgaande dat de kapitalisaties van de beleggingsobjecten bekend zijn, is voor gegeven  $j$  de kapitalisatie van de portefeuille uit te rekenen als de gewogen som hiervan, gewogen met de gewichten van  $j$ . Om precies te zijn: is  $j$  de gewichtsvector  $(w^1, \dots, w^N)$ , en  $a^k$  de kapitalisatie van instrument  $k$ , dan is

$$b_j = \sum_{k=1}^N w^k a^k .$$

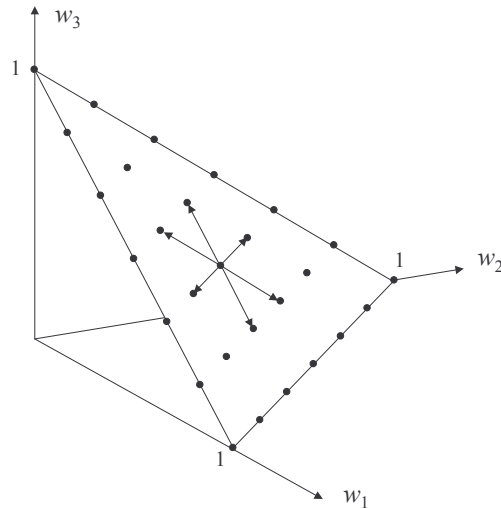
De dan verkregen *discrete* kansverdeling  $\pi_j = b_j / B$  benadert de *continue* marktverdeling die gegeven is in paragraaf 4.1.

Bij de uniforme verdeling is voor iedere toestand  $j$  de verdelingsfactor  $b_j = 1$ : de kansmassa is gelijkmatig verspreid over alle portefeuilles binnen het mandaat. Deze *discreet*-uniforme verdeling benadert de gewenste *continu*-uniforme verdeling.

We nemen aan dat er een goed werkende procedure `VerdFactor(W)` bestaat, die voor een gegeven gewichtsvector  $W$  de verdelingsfactor retourneert. Ingeval de verdeling uniform is komt hier steeds hetzelfde getal uit. Ingeval de marktverdeling gebruikt wordt retourneert de functie een getal dat evenredig is met de som van de kapitalisaties, gewogen volgens de vector  $W$ . Hiermee zien we dat het voor de correcte werking van het algoritme niet uitmaakt of we de uniforme of kapitalisatie-gewogen verdeling prefereren. Dit is het grote voordeel van het Hastings-Metropolis algoritme!

Een belangrijke vraag is wat voor Markov-keten  $(M_n)_n$  we op de toestandsruimte simuleren. Het idee is om een zogeheten **naaste-buren-keten** te gebruiken. Voor een gewichtsvector  $w$  in de toegelaten verzameling beschouwen we alle gewichtsvectoren, verschillend van  $w$ , die ook binnen de toegelaten verzameling liggen, en die uit  $w$  gevormd kunnen worden door  $d$  bij één coördinaat op te tellen en van één coördinaat af te trekken. Deze gewichtsvectoren noemen we de **naaste burens** van  $w$ , zie figuur 27.





**Figuur 27: Naaste buren van een portefeuille.**

In de Markov-keten  $(M_n)_n$  bestaan de toestandsovergangen uit overgangen van een gewichtsvector  $W$  naar één van zijn naaste buren, waarbij iedere naaste buur met gelijke kans de nieuwe toestand wordt. Indien de toegelaten verzameling samenhangend is, bestaat er altijd tenminste één naaste buur. Tevens voldoet de keten  $(M_n)_n$  aan alle voorwaarden die nodig zijn om voor MCMC gebruikt te kunnen worden:

- Voor alle  $j, k \in I$  is  $m_{jk} > 0$  als en alleen als  $m_{kj} > 0$ .
- Er zijn geen twee disjuncte fuiken, mits de toegelaten verzameling samenhangend is. Op deze voorwaarde gaan we verderop nog nader in.
- De keten  $(M_n)_n$  is aperiodiek. Dit wordt bewerkstelligd indien de dimensie van het toegelaten gebied groot genoeg is zodat de keten vanuit een zekere toestand via verschillende toestanden kan terugkeren.
- Bij een gegeven overgang  $j \rightarrow k$  zijn de kansen  $m_{jk}$  en  $m_{kj}$  eenvoudig uit te rekenen.

De laatste voorwaarde is belangrijk omdat de kansen  $m_{jk}$  en  $m_{kj}$  nodig zijn in stap 2 van het Hastings-Metropolis algoritme. De afgeleide keten  $(X_n)_n$  kan niet gesimuleerd worden zonder kennis van  $m_{jk}$  en  $m_{kj}$ . Wanneer we het aantal naaste buren van toestand  $j$  hebben geteld, weten we dat

$$m_{jk} = \frac{1}{\#\text{naaste buren van } j},$$

want toestand  $k$  is één van de naaste buren van toestand  $j$ . Ook weten we dat

$$m_{kj} = \frac{1}{\#\text{naaste buren van } k},$$

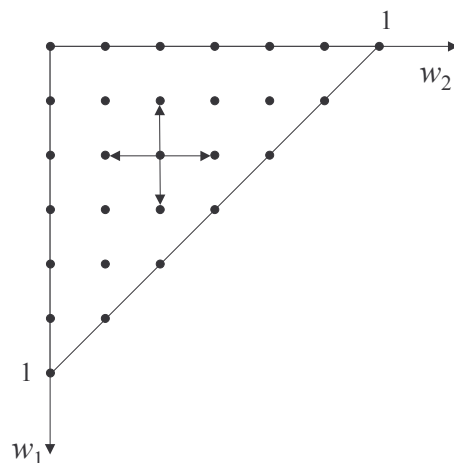
want toestand  $j$  is één van de naaste buren van toestand  $k$ .

## Verkleining van het aantal naaste buren

We merken op dat het niet noodzakelijk is om steeds alle naaste buren te beschouwen. Doen we dit echter wel (zoals in figuur 27), dan is het aantal buren dat in stap 1 langsgelopen moet worden kwadratisch in  $N$ . Er zijn immers  $N$  objecten die met  $d$  verhoogd kunnen worden en ook  $N$  objecten die met  $d$  verlaagd kunnen worden. Om te bepalen welke van deze buren toegelaten zijn en het aantal daarvan te tellen, moeten al deze buren langsgelopen worden. Dit kost evident te veel rekentijd als  $N$  groot is. Daarom kiezen we voor een andere aanpak.

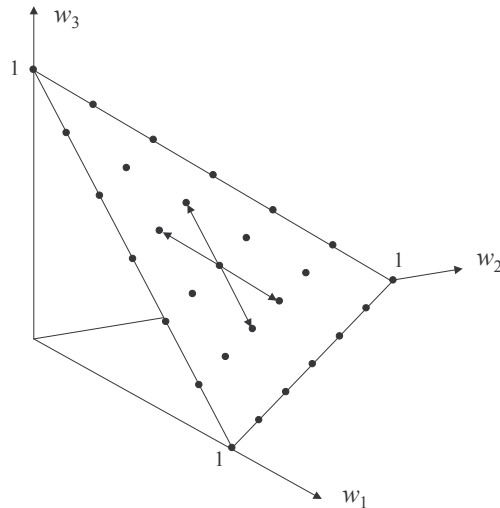
Het idee is dat de dimensie van de simplex  $S_N$  waarin we werken gelijk is aan  $N-1$ . Het moet dan mogelijk zijn binnen om  $S_N$  een coördinatenstelsel van  $N-1$  componenten in te voeren. In iedere  $N-1$  richtingen kunnen we dan een stap  $d$  vooruit of achteruit, zodat het totaal aantal buren gelijk wordt aan  $2(N-1)$ , hetgeen niet kwadratisch maar lineair in  $N$  is.

Om dit te realiseren gaan we werken met punten in de projectie van de eenheidssimplex  $S_N^{\text{Proj}}$ . Dit is een ruimte van dimensie  $N-1$  waarin van iedere vector de buren worden verkregen door bij één van de  $N-1$  coördinaten  $d$  op te tellen of af te trekken. Figuur 28 toont de mogelijke buren ingeval  $N=3$ .



Figuur 28: Naaste buren in geprojecteerde simplex.

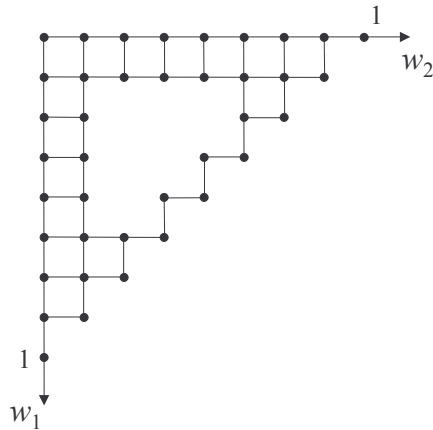
Om ervoor te zorgen dat de corresponderende vector zich in de simplex  $S_N$  bevindt, zal bij de  $N$ -de coördinaat het getal  $d$  worden afgetrokken of opgeteld al naar gelang deze bij een andere coördinaat is opgeteld of afgetrokken. Figuur 29 toont de mogelijke buren binnen  $S_N$  voor  $N=3$ . Dit aantal is inderdaad kleiner dan in figuur 27.



**Figuur 29: Mogelijke buren in de eenheidssimplex.**

We maken nog een opmerking. Stel dat  $N = 3$ ,  $d = 1\%$ , en dat er een constraint is opgelegd die zegt dat het aantal positieve gewichten maximaal 2 is. Zoals in het voorafgaande simuleren we de Markov-keten op de projectie naar coördinaat  $N = 3$ . Wanneer nu de keten zich in de vector  $(50\%, 50\%, 0\%)$  bevindt, dan heeft de toestand geen buren! Immers, de coördinaat waarnaar we projecteren verandert altijd en in dit geval is deze 0%. Bij een toestandsovergang wordt deze 1%, en één van de overige twee coördinaten 49%, zodat er 3 gewichten positief zijn. Er zijn verschillende manieren om dit probleem tegen te gaan:

- We projecteren naar een coördinaat die in de huidige toestand ongelijk aan 0 is. Nu is de projectiecoördinaat dus niet altijd  $N$ , maar wordt deze afhankelijk van de toestand. Dit lost inderdaad het probleem op dat sommige punten geen buren hebben. Maar bij deze methode ontstaat wel een nieuw probleem. Stel bijvoorbeeld dat  $(0\%, 99\%, 1\%)$  de huidige toestand is en dat we projecteren naar de eerste coördinaat ongelijk nul (gerekend vanaf 1). In dit geval is dat de tweede coördinaat. Nu is er géén overgang naar  $(1\%, 99\%, 0\%)$  mogelijk, omdat dan de tweede coördinaat niet verandert. De omgekeerde overgang, van  $(1\%, 99\%, 0\%)$  naar  $(0\%, 99\%, 1\%)$ , is echter wel mogelijk, omdat de eerste coördinaat verandert! We zien dus dat dit model niet altijd voldoet aan de eis dat  $m_{jk} > 0$  als en alleen als  $m_{kj} > 0$ .
- We nemen de buren in de projectie naar coördinaat  $N$ , als in de huidige toestand  $w^N \neq 0$ . Het aantal buren is dan lineair in  $N$ . Maar als  $w^N = 0$ , dan nemen we alle naaste buren. Dit aantal is kwadratisch in  $N$ . Nu hebben alle punten tenminste twee buren, en van alle mogelijke toestandsovergangen is ook de omgekeerde overgang mogelijk. Het nadeel van deze methode is natuurlijk dat het langslopen van de buren in sommige punten een kwadratisch proces is, hetgeen nadelig is voor de rekentijd. Daar staat tegenover dat de kwadratische berekening in slechts een klein deel van de simplex wordt toegepast. In de overige delen wordt de lineaire berekening toegepast.
- We nemen de buren altijd in de projectie naar coördinaat  $N$ , maar nemen het toegelaten gebied iets ruimer. Dit doen we door niet te eisen dat maximaal  $N^{\max}$  gewichten groter dan 0 zijn, maar te eisen dat minimaal  $N - N^{\max} - 1$  gewichten gelijk zijn aan 0 en minimaal  $N - N^{\max}$  gewichten ten hoogste  $d$  zijn. Er mag dus één gewicht  $d$  zijn in plaats van 0. Omdat  $d$  in het algemeen klein is, zal de resulterende POD niet veel afwijken van de werkelijke POD. Bij deze methode hebben alle punten tenminste twee buren, en is van alle mogelijke toestandsovergangen ook de omgekeerde overgang mogelijk. Onderstaande figuur 30 laat voor  $N = 3$  en  $N^{\max} = 2$  de mogelijke overgangen (in beide richtingen) zien:



Figuur 30: Mogelijke overgangen bij dimensie 3 en maximum van 2.

Van deze drie methoden gaat de voorkeur uit naar de derde, omdat deze het gunstigst is voor de rekentijd. Deze zullen we dan ook gebruiken. De benodigde wijzigingen kunnen worden aangebracht in de eerder genoemde functie `CheckMandaat`.

### Formulering van het algoritme

Hieronder volgt een concrete en volledige beschrijving van het trekkingsalgoritme in pseudo-programmeertaal. We maken gebruik van een aantal (globale) variabelen en arrays:

- $d$  := discretisatiestap (bijv.  $d = 0.01\%$ );
- $W$  := huidige gewichtsvector (huidige toestand van de keten  $(X_n)_n$ );
- $W_{\text{cand}}$  := kandidaat-gewichtsvector (kandidaat-toestand van  $(X_n)_n$ );
- $\#B$  := aantal naaste burens van de huidige toestand;
- $\#B_{\text{cand}}$  := aantal naaste burens van de kandidaat-toestand;
- $B_u$  := array die voor iedere naaste buur aangeeft welke coördinaat met  $d$  wordt verhoogd;
- $B_d$  := array die voor iedere naaste buur aangeeft welke coördinaat met  $d$  wordt verlaagd;
- $B_{u,\text{cand}}$  := zelfde als  $B_u$ , maar voor kandidaat-toestand i.p.v. huidige toestand;
- $B_{d,\text{cand}}$  := zelfde als  $B_d$ , maar voor kandidaat-toestand i.p.v. huidige toestand;
- $\gamma$  := boolean variabele; geeft aan of de kandidaat-toestand geaccepteerd is of niet;
- $m_{jk}$  := overgangskans van huidige naar kandidaat-toestand;
- $m_{kj}$  := overgangskans van kandidaat- naar huidige toestand;
- $b_j$  := verdelingsfactor van huidige toestand;
- $b_k$  := verdelingsfactor van kandidaat-toestand.

Merk op dat we er vanuit gaan dat er een startportefeuille  $W$  is gegeven. Met andere woorden, de vector  $W$  heeft aanvankelijk een toegelaten waarde. In de praktijk kan deze, gegeven het mandaat, handmatig worden geconstrueerd, of men kan een “middelste” waarde nemen zoals  $W = (1/N, \dots, 1/N)$  (mits deze toegelaten is).

Allereerst formuleren we een procedure die voor een meegegeven kandidaat-toestand  $W_{\text{cand}}$  alle naaste burens opslaat in de arrays  $B_{u,\text{cand}}$  en  $B_{d,\text{cand}}$ , telt hoeveel naaste burens er zijn, en het aantal naaste burens opslaat in de variabele  $\#B_{\text{cand}}$ .

```

NaasteBuren( $W_{\text{cand}}$ )
{
    # $B_{\text{cand}}$  := 0 ;
     $j := 1$  ;
    for  $k := 1$  to  $N - 1$ 
    {
         $Y := W_{\text{cand}}$  ;
         $Y^k := Y^k + d$  ;
         $Y^N := Y^N - d$  ;
        if (CheckMandaat( $Y$ ))
        {
            # $B_{\text{cand}}$  ++ ;
             $B_{\text{u,cand}}^j := k$  ;
             $B_{\text{d,cand}}^j := N$  ;
             $j ++$  ;
        }
         $Y := W_{\text{cand}}$  ;
         $Y^k := Y^k - d$  ;
         $Y^N := Y^N + d$  ;
        if (CheckMandaat( $Y$ ))
        {
            # $B_{\text{cand}}$  ++ ;
             $B_{\text{u,cand}}^j := N$  ;
             $B_{\text{d,cand}}^j := k$  ;
             $j ++$  ;
        }
    }
}

```

De procedure die één iteratie van het Hastings-Metropolis algoritme uitvoert, maakt gebruik van  $\text{NaasteBuren}(W_{\text{cand}})$ , en ziet er als volgt uit.

```

HM_Iteratie( $\gamma$ )
{
    'Bepaal aantal naaste burenen van  $W$ '
    'Trek  $W_{\text{cand}}$ '
    'Bepaal aantal naaste burenen van  $W_{\text{cand}}$ ': NaasteBuren( $W_{\text{cand}}$ ) ;
    'Bereken  $b_j$  en  $b_k$ ':  $b_j := \text{VerdFactor}(W)$  ;
     $b_k := \text{VerdFactor}(W_{\text{cand}})$  ;
    'Bereken  $m_{kj}$  en  $m_{jk}$ ':  $m_{kj} := 1/\#B_{\text{cand}}$  ;
     $m_{jk} := 1/\#B$  ;
    'Bepaal volgende toestand  $W$ '
}

```

Het gebruik van de parameter  $\gamma$  wordt verderop nader toegelicht. Van drie stappen is gegeven (achter de dubbele punt) welke statements daarvoor moeten worden uitgevoerd. De overige drie stappen zijn minder eenvoudig; we werken ze hieronder verder uit.

Om het aantal naaste burenen van  $W$  te bepalen zouden we een soortgelijke procedure  $\text{NaasteBuren}$  voor  $W$  als voor  $W_{\text{cand}}$  kunnen gebruiken. In dat geval wordt echter per iteratiestap twee keer een for-loop doorlopen, hetgeen niet nodig is omdat de  $W$  van de

huidige iteratie altijd gelijk is aan de  $W$  of de  $W_{\text{cand}}$  van de vorige iteratie. Om die reden wordt er ook een boolean variabele  $\gamma$  bijgehouden die aangeeft of in de vorige iteratie de kandidaat-toestand geaccepteerd is als nieuwe toestand. Is dit zo, dan zijn de gegevens met betrekking tot de naaste burenen van  $W$  gelijk aan die van  $W_{\text{cand}}$ . Is dit niet zo, dan heeft  $W$  dezelfde waarde als in de vorige iteratie en hoeft er niets berekend te worden, aangezien de gegevens al eerder berekend zijn. De procedure voor het bepalen van de naaste burenen van  $W$  is dus als volgt.

```

`Bepaal aantal naaste burenen van W`
{
  if  $\gamma = true$ 
  {
     $\#B := \#B_{\text{cand}}$  ;
     $B_u := B_{u,\text{cand}}$  ;
     $B_d := B_{d,\text{cand}}$  ;
  }
}

```

Om de kandidaat-toestand  $W_{\text{cand}}$  te genereren wordt gebruik gemaakt van de arrays  $B_u$  en  $B_d$ , die in de vorige stap zijn gevuld en die samen alle naaste burenen van  $W$  weergeven. Per buur geeft een element van  $B_u$  aan welke coördinaat van  $W$  met  $d$  wordt verhoogd, en geeft het element met dezelfde index van  $B_d$  aan welke coördinaat met  $d$  wordt verlaagd. Door een (discreet-)uniforme trekking uit de verzameling  $\{1, \dots, \#B\}$  te doen en uit beide arrays  $B_u$ ,  $B_d$  de elementen met de getrokken index te nemen, wordt bepaald hoe  $W_{\text{cand}}$  uit  $W$  wordt verkregen via verhoging en verlaging van coördinaten met  $d$ .

```

`Trek  $W_{\text{cand}}$ `
{
   $Z := \text{Uniform}(\{1, \dots, \#B\})$  ;
   $k_u := B_u^Z$  ;
   $k_d := B_d^Z$  ;
   $W_{\text{cand}} := W$  ;
   $W_{\text{cand}}^{k_u} := W_{\text{cand}}^{k_u} + d$  ;
   $W_{\text{cand}}^{k_d} := W_{\text{cand}}^{k_d} - d$  ;
}

```

Tenslotte moet worden bepaald of de kandidaat-toestand als nieuwe toestand wordt geaccepteerd of niet. Daarvoor zijn de berekende getallen  $b_j, b_k, m_{jk}, m_{kj}$  nodig. Tevens retourneert de totale procedure een boolean, met de waarde *true* als de kandidaat-toestand is geaccepteerd, en de waarde *false* anders. Dit is nodig om de waarde van  $\gamma$  voor de volgende iteratie te bepalen.

```

`Bepaal volgende toestand W`
{
   $U := \text{Uniform}(0,1)$  ;
  if  $U < b_k m_{kj} / b_j m_{jk}$ 
  {
     $W := W_{\text{cand}}$  ;
    return true ;
  }
  else return false ;
}

```

Nu we de procedure voor één Hastings-Metropolis-iteratie gespecificeerd hebben, zou de procedure voor het trekken van een random portefeuille via MCMC, met  $n^{\text{end}}$  iteraties, er als volgt uit kunnen zien. Het verkregen algoritme duiden we verder aan met **algoritme 3**.

```
TrekPortefeuilleMCMC()
{
   $W := \text{'Startportefeuille'}$ ;
  'Initialisatie':
   $\gamma := \text{true}$ ;
   $W_{\text{cand}} := W$ ;
  TelNaasteBuren( $W_{\text{cand}}$ );

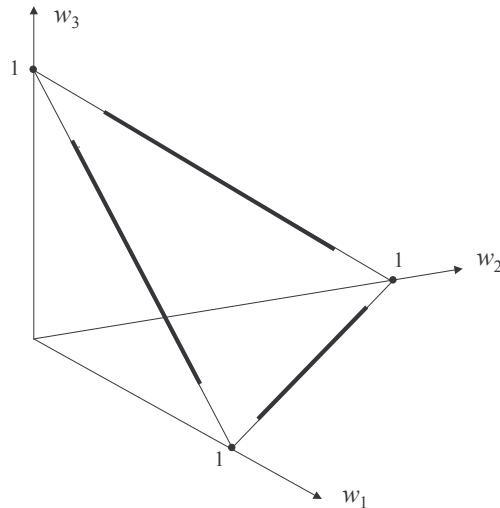
  for  $n=1$  to  $n^{\text{end}}$ 
  {
     $\gamma := \text{HM\_Iteratie}(\gamma)$ ;
  }
  'Verwerk portefeuille  $W$  verder';
}
```

Het statement 'Verwerk portefeuille  $w$  verder' zorgt ervoor dat alle informatie met betrekking tot  $W$  die we nodig hebben (bijv. het rendement) wordt berekend en bewaard.

### Samenhang van het toegelaten gebied

We hebben reeds opgemerkt dat MCMC alleen werkt voor portefeuilletrekking indien het toegelaten gebied samenhangend is. Het toegelaten gebied moet zelfs zo geconstrueerd zijn dat het rooster op de gediscretiseerde simplex, met verbindingen bepaald door de Markov-keten, een samenhangend netwerk vormt. Immers, is dit niet zo, dan heeft de keten twee of meer disjuncte fuiken en bestaat de limietverdeling niet.

We hebben echter in paragraaf 4.1 gezien dat de gegeven mandaatregels niet noodzakelijk leiden tot een samenhangend gebied. Het type regel dat het meest voor problemen zorgt is een maximum  $N^{\text{max}}$  op het aantal objecten dat positief gewicht heeft. Een dergelijke regel verlaagt de dimensie van het toegelaten gebied indien  $N^{\text{max}} < N$ , aangezien het toegelaten gebied dan geheel op de rand van de simplex ligt. Andere constraints, zoals grenzen op gewichten of risicomaten, zorgen er dan voor dat het toegelaten gebied uit meerdere componenten bestaat die onderling niet verbonden zijn. Figuur 31 laat een voorbeeld zien voor  $N=3$  en  $N^{\text{max}}=2$ . De rand wordt in stukken verdeeld door andere constraints zoals bovengrenzen op exposure-gewichten. De MCMC methode is niet toepasbaar op zo'n mandaat omdat de keten altijd binnen één component blijft (namelijk de component waarbinnen de startportefeuille ligt).

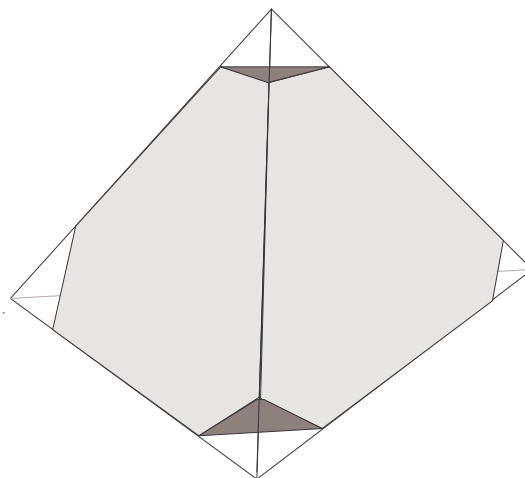


**Figuur 31: Toegelaten gebied bestaat uit onsamenhangende stukken.**

We maken over deze problematiek de volgende opmerkingen.

Voor allocatiemanagers zijn er geen constraints met  $N^{\max} < N$ . Allocatiemanagers verdelen hun kapitaal over een relatief klein aantal portefeuillesegmenten, die ieder weer door een andere manager worden beheerd. Het is niet te verwachten dat sommige van die segmenten niet “mee mogen doen”. Dus: allocatiemandaat vormen een samenhangend toegelaten gebied.

Constraints met  $N^{\max} < N$  treden derhalve alleen op voor selectiemanagers, welke direct in marktinstrumenten beleggen. In zulke gevallen is  $N$  vaak groot en  $N^{\max}$  een stuk kleiner. Er ontstaan dan meerdere componenten indien bijvoorbeeld  $N^{\max} = 2$  en er maximumwaarden op de gewichten ( $< 100\%$ ) zijn opgelegd. Dit is precies wat er gebeurt in het voorbeeld hierboven. Dit voorbeeld is echter een randgeval. In de praktijk is voor een selectiemanager bijvoorbeeld  $N = 500$  en  $N^{\max} = 60$ . Het toegelaten gedeelte van de rand bestaat dan niet slechts uit een aantal lijnstukken, maar uit zijvlakken die met elkaar in verbinding staan via de zijvlakken van lagere dimensies. Voor  $N = 4$  en  $N^{\max} = 3$  ziet het plaatje eruit volgens figuur 32.



**Figuur 32: Samenhangend toegelaten gebied met gaten.**



Dit is het analogon van de combinatie  $N=3$ ,  $N^{\max}=2$  met dimensie 1 opgehoogd. Nu ziet het plaatje er veel gunstiger uit: het toegelaten gebied bevat gaten, maar is wel samenhangend.

We concluderen dat in praktijksituaties voor zowel allocatie- als selectiemanagers de toegelaten verzameling zeer waarschijnlijk wel samenhangend is. Is dit desondanks niet het geval, dan luidt het advies om één of enkele constraints te verwaarlozen bij de Monte Carlo simulatie, zodat de toegelaten verzameling enigszins wordt uitgebreid om deze samenhangend te maken.

## Complexiteit; Testen

Net als voor de eerste twee algoritmes, maken we voor algoritme 3 een aantal opmerkingen met betrekking tot de complexiteit. Het mag duidelijk zijn dat per iteratiestap de procedure die alle burens langsloopt de meeste tijd vergt. We hebben afgeleid dat deze procedure lineair in de dimensie  $N$ . het aantal rekenstappen is dan van de orde  $2(N-1)$ .

Er is echter bij algoritme 3 nog een belangrijke kwestie: hoeveel iteratiestappen dienen er per trekking te worden uitgevoerd? Ofwel: hoe groot is  $n^{\text{end}}$ ? Als er te weinig iteraties worden uitgevoerd zullen de getrokken portefeuilles met grote kans dicht bij de startportefeuille blijven liggen, zodat de gewenste limietverdeling niet bereikt wordt. Dit probleem treedt prominenter op naarmate  $N$  groter wordt. Immers, per iteratiestap veranderen maar 2 van de  $N$  gewichten. Om een uniforme of marktverdeling te verkrijgen zal de keten heel vaak in alle richtingen moeten lopen. Voor grote  $N$  is dit aantal richtingen groot, dus zal het aantal iteraties opgeschroefd moeten worden.

De discretisatiestap  $d$  hebben we als doelstelling gelijk aan 0.01% gekozen. Dit getal is erg klein, helemaal wanneer  $N$  groot is: per iteratiestap veranderen er slechts 2 gewichten en deze gewichten worden resp. vergroot en verkleind met slechts 0.01%. Het is daarom aannemelijk dat het proces sneller verloopt (dus sneller naar een limietverdeling convergeert) indien we  $d$  groter kiezen. We delen daarom het proces op in vier **fases**. In iedere fase wordt een nader te bepalen aantal iteraties uitgevoerd, met een bepaalde discretisatiestap  $d$ . De vier fases zijn:

- Fase 1:  $d = 10\%$ ;
- Fase 2:  $d = 1\%$ ;
- Fase 3:  $d = 0.1\%$ ;
- Fase 4:  $d = 0.01\%$ ;

Hierbij maken we een belangrijke opmerking. Ingeval er een bovengrens  $N^{\max}$  op het aantal positieve gewichten aanwezig is, hebben we het algoritme zo geconstrueerd dat steeds minimaal  $N - N^{\max} - 1$  gewichten gelijk zijn aan 0 en hooguit  $N^{\max}$  boven  $d$  uitkomen. Wanneer we overgaan naar een volgende fase wordt  $d$  kleiner, en dus kan een toestand die eerst wel toegelaten was opeens niet meer toegelaten zijn. Dit probleem lossen we op door de toestand op te schuiven naar een buurpunt dat wel toegelaten is. We voeren dus telkens bij een overgang naar een nieuwe fase de volgende stappen uit:

- Tel hoeveel gewichten 0 zijn. Zijn dit er minstens  $N - N^{\max}$ , dan kunnen we stoppen; de huidige toestand is ook toegelaten in de nieuwe fase.
- Zijn dit er  $N - N^{\max} - 1$ , zoek een gewicht gelijk aan  $d$  en maak deze 0.
- Tel  $d$  op bij één van de overige gewichten (ongelijk 0) zodat de verkregen portefeuille voldoet aan alle constraints.

Hiermee is het opdelen van het proces in vier fases ook mogelijk voor Markov-ketens op de rand van de simplex.

Een voor de hand liggende vraag is nu hoeveel iteraties er per fase nodig zijn, gegeven de dimensie  $N$ . Merk eerst op dat het voor bepaalde  $N$  niet zinvol is, of zelfs onmogelijk is, om in iedere fase iteraties uit te voeren. Iteraties in fase 1 hebben geen zin als  $N > 10$ . Neem bijvoorbeeld  $(1/N, \dots, 1/N)$  als startportefeuille. Er kunnen dan geen gewichten met 10% worden verhoogd en verlaagd zonder op negatieve waarden uit te komen. De portefeuille heeft dan 0 naaste burens en het algoritme loopt vast. Om soortgelijke redenen doen we geen iteraties in fase 2 als  $N > 100$ .

Algoritme 3 is getest (in het DOS programma) voor verschillende  $N$  op correcte werking. Dit wordt gedaan door de resulterende POD te vergelijken met die POD die ontstaat wanneer algoritme 2 gebruikt wordt. Details hiervan staan in het eerder genoemde testrapport. In de gevallen waarbij we hebben geconcludeerd dat de resultaten nagenoeg hetzelfde zijn, weten we niet alleen dat algoritme 3 correct werkt, maar ook dat het gekozen aantal iteraties voldoende is om de limietverdeling goed te benaderen. Helaas hebben we slechts voor  $N \leq 100$  kunnen concluderen dat algoritme 3 goed werkt. Waarschijnlijk is dit omdat voor grotere  $N$  het aantal benodigde iteraties groter is dan het aantal waarvoor we hebben kunnen testen. Om die reden wordt het afgeraden om algoritme 3 te gebruiken wanneer  $N > 100$ . Verderop bespreken we een oplossing voor gevallen waarbij de dimensie wel zo groot is.

In de volgende tabel 9 staat het gebruikte aantal iteraties voor alle  $N$  waarvoor de test geslaagd is.

Tabel 9: Gebruikte aantallen iteraties.

$N$	Gebruikte # iteraties	mogelijke verdeling over de fasen
2	300	(0, 200, 50, 50)
3	500	(200, 100, 100, 100)
4	500	(200, 100, 100, 100)
5	500	(200, 100, 100, 100)
6	500	(200, 100, 100, 100)
8	800	(500, 100, 100, 100)
10	2000	(0, 1500, 500, 0)
25	2500	(0, 2000, 500, 0)
50	3500	(0, 3000, 500, 0)
100	3500	(0, 1000, 2500, 0)

Aangeraden wordt om de iteraties te verdelen over de verschillende fasen, met de meeste iteraties in de fase met de grootste (zinvolle) discretisatiestap  $d$ , want deze fase bepaalt voor het grootste deel de resulterende verdeling. De vector met de verdeling over de fasen geeft het aantal iteraties weer in respectievelijk de fasen 1, 2, 3 en 4.

Ook is er door middel van tests bepaald hoeveel tijd de iteraties kosten, als functie van  $N$ . In tabel 10 hieronder staan de tijden. Ook deze tests zijn uitgevoerd in het DOS programma. De exacte tijden kunnen afwijkend zijn in een andere implementatie, maar de onderlinge verhoudingen zullen wel ongeveer kloppen. Verder zien we, zoals te verwachten, dat de tijden ongeveer lineair zijn in  $N$ , maar niet helemaal.

**Tabel 10: Rekeningtijden per iteratie voor verschillende dimensies.**

$N$	1 iteratie	100 iteraties	10000 iteraties
2	$1.11 \cdot 10^{-4}$ sec.	0.0111 sec.	1.11 sec.
3	$2.18 \cdot 10^{-4}$ sec.	0.0218 sec.	2.18 sec.
5	$4.44 \cdot 10^{-4}$ sec.	0.0444 sec.	4.44 sec.
10	$9.96 \cdot 10^{-4}$ sec.	0.0996 sec.	9.96 sec.
25	$2.67 \cdot 10^{-3}$ sec.	0.267 sec.	26.7 sec.
50	$5.41 \cdot 10^{-3}$ sec.	0.541 sec.	54.1 sec.
100	0.0111 sec.	1.11 sec.	111 sec.
250	0.0296 sec.	2.96 sec.	296 sec.
500	0.0674 sec.	6.74 sec.	674 sec.
1000	0.162 sec.	16.2 sec.	1620 sec.

Met behulp van de twee tabellen hierboven kan berekend worden hoeveel tijd één trekking met algoritme 3 kost als functie van  $N$ . Dit is ongeveer het aantal benodigde iteraties maal de tijd die één iteratie kost. Tabel 11 geeft de resultaten.

**Tabel 11: Rekeningtijden per trekking voor verschillende dimensies.**

$N$	rekeningtijd per trekking
2	0.0333 sec.
3	0.109 sec.
5	0.222 sec.
10	1.494 sec.
25	5.34 sec.
50	16.23 sec.
100	33.3 sec.

We zien dat de rekeningtijd snel oploopt als functie van  $N$  en dat het ingeval  $N = 100$  ongeveer een halve minuut duurt om één trekking te verkrijgen. Voor het verkrijgen van een voldoende grote dataset zal de rekeningtijd dus behoorlijk groot uitvallen.

### Groepenindeling bij hoge dimensie

Wanneer algoritme 3 toch gebruikt moet worden voor grote  $N$  kan de volgende methode soms uitkomst bieden. We verdelen de verzameling van alle mogelijke instrumenten in groepen van (ongeveer) gelijke grootte. Verstandig is om groepen te maken met stukken die veel overeenkomsten hebben (bijvoorbeeld ten aanzien van sector, regio, land, etc.), want deze vertonen vaak een positieve correlatie in de rendementen. Een andere mogelijkheid is de stukken te sorteren op rendement en op basis daarvan in groepen te verdelen.

Vervolgens wordt er niet gesimuleerd over de instrumenten, maar over de groepen. Die zijn namelijk veel kleiner in aantal. Is bijvoorbeeld  $N = 500$ , dan kunnen groepen van 50 stukken gemaakt worden, zodat uiteindelijk over 10 objecten wordt gesimuleerd. Een exposure van 1% in een groep komt overeen met een exposure van 0.02% in ieder stuk van de groep. Op deze manier kan de werkelijke POD worden benaderd.

Bovenstaande methode werkt ook wanneer de marktverdeling gebruikt wordt: voor de marktkapitalisatie van een groep nemen we de som van de kapitalisaties van de stukken in de groep.

Helaas blijkt uit tests echter dat de POD er iets anders uit kan zien wanneer groepsindeling gebruikt wordt. Met name de spreiding is vaak groter dan wanneer direct over de instrumenten wordt gesimuleerd.

### **Opmerkingen ten aanzien van de bruikbaarheid**

Zoals we hebben gezien werkt algoritme 3 in theorie voor bijna alle denkbare mandaten en kansverdelingen. Helaas echter is de rekentijd een zeer beperkende factor, waardoor algoritme 3 praktisch alleen bruikbaar is voor  $N \leq 100$ . En zelfs dan duurt het voor  $N = 100$  nog een halve minuut om slechts één trekking te genereren. Het gebruiken van een groepsindeling is een aardige methode voor hogere dimensies, maar de resulterende POD kan afwijkend uitvallen, zodat deze methode niet helemaal een bevredigende oplossing vormt voor het hoge dimensie probleem.

Een sterk punt van algoritme 3 is dat het gebruikt kan worden wanneer de (lineaire) marktverdeling gewenst is. Het verschil zit 'm slechts in de coëfficiënten  $b_j$  die een rol spelen bij de bepaling of een kandidaat-toestand wordt geaccepteerd of niet. Hierdoor is nauwelijks extra rekentijd nodig om uit een gewogen verdeling te trekken. Aangezien algoritme 1 en 2 geen bevredigende uitbreiding hebben voor de marktverdeling, is algoritme 3 dus dé methode hiervoor.

Een ander sterk punt is dat geen acceptatie/rejectie wordt toegepast. Algoritme 3 kan daarom ook gebruikt worden waar algoritme 1 en 2 falen, namelijk in de gevallen met een zeer klein toegelaten gebied. Een voorwaarde is echter wel dat er een startportefeuille gespecificeerd moet worden. Deze portefeuille vormt de starttoestand van de simulatie, en moet derhalve voldoen aan alle mandaatregels. Het vinden van zo'n startportefeuille is verre van triviaal, zeker wanneer het toegelaten gebied klein is.

Ondanks deze sterke punten blijft de rekentijd een nare beperking vormen. We daarom raden aan om algoritme 3 in de praktijk zo weinig mogelijk te gebruiken. Alleen als er geen alternatieven zijn, bijvoorbeeld ingeval van een klein toegelaten gebied of bij gebruik van de marktverdeling, is het verstandig om de trekkingen via algoritme 3 te genereren.

## **4.5 Een vergelijking tussen de drie algoritmes**

In deze paragraaf worden de drie algoritmes voor POD berekening naast elkaar gezet en wordt een overzicht gegeven van de voor- en nadelen die in de paragrafen hiervoor besproken zijn. Tevens wordt bepaald naar welk algoritme in welke situatie de voorkeur uitgaat. Allereerst zetten we de belangrijkste kenmerken van de drie methodes op een rij.

Algoritme 1:

- Genereert géén random trekkingen, maar loopt alle portefeuilles uit de gediscretiseerde simplex langs, en combineert dit met het acceptatie/rejectie principe.
- Is niet echt bruikbaar voor mandaten waarvoor slechts een klein deel van de simplex toegelaten is. Dit omdat acceptatie/rejectie gebruikt wordt. Wel kan met gewichtsgrenzen automatisch rekening worden gehouden, voordat acceptatie/rejectie in werking treedt.
- Is waarschijnlijk niet bruikbaar ingeval de kapitalisatie-gewogen verdeling gewenst is in plaats van de uniforme verdeling.
- Heeft een rekentijd die exponentieel stijgt in de dimensie  $N$ . Daardoor is de methode slechts bruikbaar voor  $N \leq 4$  met  $d = 0.01\%$ , of  $N = 5$  met  $d = 0.1\%$ . Voor hogere  $N$  is

ofwel de rekentijd ofwel het formaat van de dataset te groot. Wanneer we  $d$  groter kiezen valt de standaarddeviatie van de POD te hoog uit.

Algoritme 2:

- Genereert uniforme trekkingen uit de simplex, en combineert dit met het acceptatie/rejectie principe.
- Is niet echt bruikbaar voor mandaten, waarvoor slechts een zeer klein deel van de simplex toegelaten is. Dit omdat acceptatie/rejectie gebruikt wordt. Wel kan met een bovengrens  $N^{\max}$  op het aantal positieve gewichten automatisch rekening worden gehouden, voordat acceptatie/rejectie in werking treedt.
- Is waarschijnlijk niet bruikbaar ingeval de kapitalisatie-gewogen verdeling gewenst is in plaats van de uniforme verdeling, hoewel er een benaderende methode bestaat.
- Heeft een rekentijd die lineair afhangt van de dimensie  $N$ . Daardoor kunnen, zelfs bij dimensies van 500 of 1000, in enkele minuten tijd honderdduizenden trekkingen worden verricht.

Algoritme 3:

- Genereert trekkingen uit de toegelaten verzameling via Markov-keten Monte Carlo simulatie.
- Is ook bruikbaar voor mandaten met een zeer klein toegelaten gebied, omdat iedere trekking automatisch binnen het toegelaten gebied valt. Daar staat tegenover dat er wel een startportefeuille gegeven moet zijn, die voldoet aan alle mandaatregels.
- Kan zowel uit de uniforme als de kapitalisatie-gewogen verdeling trekken. Het trekken uit de kapitalisatie-gewogen verdeling (de lineaire variant) vergt nauwelijks extra inspanning.
- Heeft per iteratie een rekentijd die lineair afhangt van  $N$ . Maar het aantal iteraties dat nodig is voor één trekking kan soms heel groot zijn, met name bij grote  $N$ . Daardoor kan het op een computer soms etmalen rekentijd kosten om bijvoorbeeld 100 trekkingen te verrichten. We hebben geconcludeerd dat algoritme 3 praktisch uitvoerbaar is voor  $N \leq 100$ . Bij grotere dimensie moet deze kleiner gemaakt worden door het indelen van de beschikbare stukken in groepen, maar dan wijkt de vorm van de verdeling iets af.

Aan de hand van dit overzicht van de voor- en nadelen kan een idee worden verkregen voor welk type mandaat welk algoritme het meest geschikt is. Onderstaande tabel 12 geeft een voor de hand liggende indeling hiervan.

Tabel 12: Geschikte algoritmes per type mandaat.

Type mandaat	Aanbevolen algoritmes
Allocatiemandaat met $N \leq 5$ , strakke onder- en bovengrenzen op de gewichten, en uniforme verdeling	Algoritme 1 of 2
Allocatiemandaat met $N > 5$ , zonder strakke gewichtsgrenzen, en uniforme verdeling	Algoritme 2 of 3
Selectiemandaat met niet zo grote grote $N$ (bijvoorbeeld $N \leq 50$ ), en uniforme verdeling	Algoritme 2 of 3
Selectiemandaat met grote $N$ , veel vrijheid in de keuze van de portefeuille (“veel uit veel”) en uniforme verdeling	Algoritme 2
Selectiemandaat met grote $N$ , weinig vrijheid in de keuze van de portefeuille (“weinig uit veel”) en uniforme verdeling	Algoritme 3
Mandaat met marktgewogen verdeling	Algoritme 3

Merk op dat wat “veel” of “weinig” vrijheid precies betekent mede afhankelijk is van de dimensie. Een maximale exposure van 5% per object legt veel beperking op als  $N = 30$  of 40, maar weinig beperking als  $N = 500$  of 1000 (meer voorbeelden staan in tabel 8 in paragraaf 4.3).

## 4.6 Uitbreidingen

In deze slotparagraaf van hoofdstuk 4 gaan we in op een aantal manieren waarop de POD methode en/of één van de behandelde algoritmes kan worden uitgebreid. Deze uitbreidingen zijn soms nuttig voor de toepassing, of geven een mogelijke verbetering op een bestaand algoritme, of leiden tot een manier om rekening te houden met een specifiek type constraint.

Een aantal van deze uitbreidingen zijn volledig uitgewerkt; andere geven slechts een eerste aanzet tot een eventueel vervolgonderzoek.

### Ex-ante tracking error

Een veel voorkomend constraint in een beleggingsmandaat is een bovengrens (soms ook een ondergrens) op een bepaalde risicomaat. Dit is meestal de tracking error ten opzichte van een benchmark, waarbij bijvoorbeeld maanden als subperiodes worden gekozen.

Een belangrijke opmerking hierbij is dat deze maat zowel ex-ante als ex-post berekend kan worden. Ex-ante maten schatten het toekomstige risico, en ex-post maten schatten het risico over een gegeven periode in het verleden. In paragraaf 2.4 hebben we gezien hoe de ex-post tracking error kan worden berekend. De formule is als volgt:

$$TE_n := \sqrt{\frac{1}{n} \sum_{i=1}^n (VA_i - \overline{VA_n})^2}.$$

Ex-ante risicomaten worden uitgerekend via een gecompliceerde simulatietechniek. Hierbij worden portefeuillewaardes in de toekomst gesimuleerd, op basis van het stochastisch gedrag in het verleden. Op basis van een groot aantal simulaties wordt vervolgens de tracking-error berekend.

De genoemde boven- en ondergrenzen in een portefeuillemandaat hebben normaal gesproken betrekking op een ex-ante tracking error. Dat betekent dat telkens wanneer van een gegeven portefeuille bepaald moet worden of deze toegelaten is of niet, de ex-ante tracking error van deze portefeuille gesimuleerd dient te worden, hetgeen veel te veel computertijd kost. We hebben gezien dat het voor alle besproken algoritmes noodzakelijk is om portefeuilles snel te kunnen evalueren op toelaatbaarheid. Hier ontstaat dus een probleem wanneer het mandaat eisen stelt op een ex-ante tracking error.

Om aan dit probleem iets te kunnen doen zoeken we een snellere manier om ex-ante tracking errors te schatten. Een heel gemakkelijk te berekenen – maar ook heel grove – schatter is de ex-post tracking error, berekend over een periode in het verleden. Dus, wanneer het tijdsinterval  $[t_0, t_1]$  de meetperiode is, worden voor een gegeven samenstelling van de portefeuille de (bijvoorbeeld) maandelijkse rendementen berekend over een (niet te) lange periode die voorafgaat aan  $t_0$ . Naast de portefeuille rendementen worden ook de benchmark rendementen over een periode vóór  $t_0$  bepaald. Op basis van deze gegevens wordt de formule voor de ex-post tracking error toegepast.

In het vervolg van deze scriptie zullen we deze grove schattingsmethode gebruiken.

### Andere rendementen

Tot nu toe is van de getrokken portefeuilles steeds als “basis performancemaat” het rendement berekend. Door het gerealiseerde rendement te vergelijken met de getrokken rendementen wordt de behaalde performance vastgesteld.

Er zijn echter situaties denkbaar waarin we andere grootheden dan rendementen willen vergelijken. Wanneer bijvoorbeeld de manager sterk wordt beoordeeld op zijn genomen risico, is het wellicht zinvol om in plaats van het “gewone” rendement een risicocorrigerend rendement te berekenen. Voorbeelden hiervan zijn de Sharpe-ratio, Treynor-ratio, Sortino-ratio en de Information-ratio, die in paragraaf 2.4 zijn behandeld. In dat geval worden per getrokken portefeuille de maandelijkse rendementen over de meetperiode berekend. In combinatie met de gerealiseerde maandelijkse benchmark rendementen, of de risicovrije rente, kan de gewenste ratio worden berekend. Uiteraard kan ook een andere subperiode dan een maand worden gebruikt.

### Short posities en derivaten

In het voorafgaande zijn we er steeds min of meer vanuit gegaan dat een manager belegt in “primitieve” beleggingsobjecten. Ingeval van een allocatiemanager zijn dit onderliggende segmenten (of markten). Ingeval van een selectiemanager zijn dit bijvoorbeeld aandelen, obligaties of valuta.

In de praktijk echter beleggen selectie managers vaak naast deze instrumenten ook een deel van het kapitaal in afgeleide instrumenten, de zogeheten **derivaten**. Voorbeelden van derivaten zijn future contracten, en opties op aandelen of indices. Het effect van zo’n derivaat is dat de exposure in een bepaald instrument (de gevoeligheid van de portefeuillewaarde voor de marktwaarde van het instrument) kleiner dan 0% of groter dan 100% kan worden. We spreken dan resp. van een **short positie** of van een **hefboomwerking**.

In het raamwerk van PODs worden de exposure gewichten random getrokken, zodat de som 100% is. We kunnen dit uitbreiden door gewichten kleiner dan 0% of groter dan 100% toe te laten, maar wel te blijven eisen dat de totale som van de gewichten 100% is. De toegelaten verzameling valt dan gedeeltelijk buiten de eenheidssimplex, maar ligt wel in het vlak van de



eenheidssimplex. Op deze manier wordt handelen in derivaten meegenomen in de Monte Carlo simulatie.

Merk echter op dat algoritme 1 en 2 voor dit doel niet zomaar bruikbaar zijn. Immers, beide algoritmes genereren alleen trekkingen binnen de simplex, terwijl het toegelaten gebied nu ook buiten de simplex kan liggen. Een mogelijkheid om dit op te lossen is een transformatie toepassen op een trekking uit de simplex, zodat de trekking uit een groter gebied gedaan kan worden. Een alternatief is om algoritme 3 te gebruiken.

## Transacties

Tot nu toe hebben we bij de simulatie van PODs geen rekening gehouden met transacties (en de daaruit voortkomende kosten). We hebben het rendement van een manager beoordeeld ten opzichte van een “passief alternatief”: het kiezen van een portefeuille en deze aanhouden gedurende de meetperiode. Op zich kan een dergelijke analyse nuttige informatie verschaffen, maar er kunnen redenen zijn om toch het effect van tussentijdse transacties te laten meewegen:

- De mogelijkheid om transacties te doen geeft meer mogelijkheden voor het uiteindelijke rendement. Dit komt doordat marktinstrumenten (of portefeuillesegmenten) in waarde fluctueren, zodat het schuiven van kapitaal van invloed is op het eindresultaat.
- Meestal krijgen managers aan het begin van de meetperiode een startportefeuille toegewezen. Zij kunnen dan via transacties de samenstelling naar eigen inzicht wijzigen en proberen een goede performance te leveren. In dit geval echter heeft de manager de startportefeuille niet zelf gekozen (of wel gekozen maar in de vorige periode) en kan hij deze (vanwege transactiekosten) ook niet radicaal wijzigen. Maar in de huidige POD methode zijn het precies de startportefeuilles die getrokken worden. Zo'n POD is derhalve niet helemaal representatief voor de beleggingsmogelijkheden van de manager.
- Sommige mandaten kunnen een manager dwingen om zo nu en dan transacties uit te voeren. Dit is bijvoorbeeld het geval bij mandaten waarbij een benchmark wordt gevolgd, en door een kleine afwijking (tracking error) een klein overrendement behaald kan worden. De huidige POD methode loopt dan vast doordat de toegelaten verzameling leeg is. Men moet dan de mandaatregels afzwakken (dit is bij de voorbeelden in hoofdstuk 5 een paar keer gedaan) of de meetperiode korter maken.
- Portefeuilles zonder transacties zijn niet “reëel”; in de praktijk worden de meeste portefeuilles regelmatig gewijzigd via transacties. Daarom worden POD methodes zonder transacties mogelijk niet geaccepteerd door de PE professionals.

Dat de huidige POD methode geen transacties genereert betekent niet dat de uitkomst zinloos is. Het behaalde resultaat wordt vergeleken met wat er zoal mogelijk is geweest via passieve strategieën. Wanneer de meetperiode relatief kort is (zodat er in de praktijk weinig transacties worden gedaan), geeft een POD een aardig beeld van de rendementen die haalbaar zijn geweest. Alleen zou het beeld completer zijn wanneer ook transacties worden gesimuleerd.

Het simuleren van een portefeuille met transacties zou in feite als volgt moeten verlopen.

- Ga uit van de startportefeuille. Dit is de samenstelling van de portefeuille aan het begin van de meetperiode.
- Voer periodiek op bepaalde momenten enkele transacties uit en bereken de transactiekosten.
- Houd bij het berekenen van het rendement rekening met de gemaakte transactiekosten.

Hier speelt echter een keuzeprobleem met betrekking tot de frequentie waarin de herbalancerings plaatsvinden, en de mate waarin de portefeuille dan “over de kop” gaat. Hoe

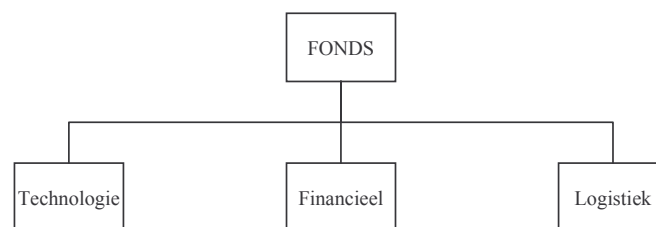


groter de transactie frequentie, hoe meer transactiekosten de simulaties maken, en dus hoe lager de kansverdeling van de POD uitvalt. Vaak bevatten mandaten geen duidelijke regels over de transacties, en zijn ook de transactiekosten lastig te schatten. Om de POD methode op dit punt te verbeteren zal vervolgonderzoek nodig zijn.

### Performance attributie met PODs

Zoals al in hoofdstuk 3 is opgemerkt kan de gemiddelde portefeuille van de POD worden opgevat als een ideale benchmark portefeuille. Dit betekent dat alle performance evaluatietechnieken die gebruik maken van benchmarks ook kunnen worden benut wanneer een POD is berekend.

In het bijzonder kan performance attributie (besproken in paragraaf 2.5) worden toegepast op het POD gemiddelde. Als voorbeeld nemen we het eenvoudige attributiemodel van Brinson en Fachler. We gaan weer uit van een allocatie over  $N$  portefeuillesegmenten, zoals in figuur 33 hieronder, waarin naar drie sectoren wordt opgesplitst. De onderliggende segmenten zijn selectieportefeuilles.



Figuur 33: Voorbeeld allocatie.

Definieer

- $w^{P,j}$  := het exposure gewicht van segment  $j$  van de portefeuille;
- $w^{M,j}$  := het exposure gewicht van segment  $j$  van de POD gemiddelde portefeuille;
- $R^{P,j}$  := het gerealiseerde rendement van segment  $j$  van de portefeuille;
- $R^{M,j}$  := het POD gemiddelde rendement van segment  $j$ ,

voor  $j=1, \dots, N$ . We nemen dus in dit geval het rendement van segment  $j$ , zoals de allocatiemanager die ervaart, gelijk aan het POD gemiddelde van  $j$ . Dit rendement vormt het benchmark rendement, die het verloop van segment  $j$  representeert. Nu is het gerealiseerde rendement van de portefeuille gelijk aan

$$\sum_{j=1}^N w^{P,j} R^{P,j} ,$$

en is het gemiddelde rendement van de POD van de portefeuille, hetgeen we opvatten als het benchmark rendement, gelijk aan

$$\sum_{j=1}^N w^{M,j} R^{M,j} .$$

Deze formules komen overeen met de formules voor klassieke benchmarks. De value added (in dit geval het verschil tussen portefeuille rendement en POD gemiddelde rendement) kan ook op dezelfde wijze worden ontbonden in verschillende effecten:

$$VA = \sum_{j=1}^N \underbrace{(w^{P,j} - w^{M,j})R^{M,j}}_I + \underbrace{w^{M,j}(R^{P,j} - R^{M,j})}_{II} + \underbrace{(w^{P,j} - w^{M,j})(R^{P,j} - R^{M,j})}_{III}$$

Net zoals in paragraaf 2.5 vatten we term I op als het allocatie-effect van segment  $j$ . Dit effect is onafhankelijk van het selectiebeleid van de manager van segment  $j$ , maar wordt bepaald door de afwijking van de portefeuille ten opzichte van de POD gemiddelde portefeuille. De term met nummer II representeert het selectie-effect van segment  $j$ . Dit wordt bepaald door de afwijking van de selectiemanager ten opzichte van zijn POD gemiddelde. Tot slot is er een interactie-effect dat gegeven is door term III.

### Variant op MCMC zonder discretisatie

Zoals bekend heeft de Markov-keten Monte Carlo methode vaak veel iteraties nodig om de limietverdeling goed te kunnen benaderen. Dit is vooral te wijten aan het feit dat per iteratiestap de portefeuille niet zo veel verandert. Er worden twee random gewichten gekozen; de één wordt verhoogd met  $d$  en de ander wordt verlaagd met  $d$ . Helemaal wanneer  $d$  klein is verandert er per iteratie niet veel.

Mogelijk convergeert het proces sneller naar de limietverdeling als er per iteratie meer verandert aan de portefeuille. We geven een eerste aanzet om dit te verbeteren. Het idee van de verbetering is dat we de toestandsruimte niet langer discretiseren. In plaats daarvan passen we een variant van MCMC op een continue toestandsruimte toe, met als limietverdeling de gewenste dichtheid  $\pi(w)$  op het toegelaten gebied. Dit is dus de uniforme dichtheid of de kapitalisatie-gewogen dichtheid, en we veronderstellen dat deze bekend is op een multiplicatieve constante na. Een getrokken kandidaat-toestand kan dan mogelijk iedere toestand in het toegelaten gebied zijn, zodat de convergentie naar de limietverdeling mogelijk sneller verloopt.

Het Hastings-Metropolis algoritme kent verschillende varianten voor continue toestandsruimte. Hieronder geven we een voorbeeld van zo'n variant.

#### Hastings-Metropolis algoritme zonder discretisatie:

0. Specificeer een reeks kansdichtheden  $m(w,z)$ , voor vaste  $w$  en variabele  $z$ , waarvoor  $m(w,z) > 0$ , voor alle  $w,z$  in het toegelaten gebied.

Neem een starttoestand  $w$ . Laat  $n := 0$  en  $X_0 := w$ .

Kies de looptijd  $n^{\text{end}}$  van de keten.

1. Genereer een kandidaat-toestand  $K$  uit de dichtheid  $f_K(k) := m(X_n, k)$ .
2. Neem  $H := K$  met kans

$$\min \left\{ \frac{\pi(k)m(k, X_n)}{\pi(X_n)m(X_n, k)}, 1 \right\}$$

en anders  $H := X_n$ .

3. Laat  $n := n + 1$ ;  $X_n := H$ , en ga naar 1, of stop als  $n = n^{\text{end}}$ .

Deze variant staat ook beschreven in [5]. Het is echter niet eenvoudig om een geschikte kandidaat dichtheid  $m(w,z)$  te vinden. Voor de hand ligt bijvoorbeeld een dichtheid die symmetrisch is om het punt  $w$ , maar ook moet  $z$  in het toegelaten gebied liggen, indien

$m(w,z) > 0$ . Verder is het van belang dat, gegeven een toestand  $w$  en een getrokken kandidaat-toestand  $z$ , de verhouding van de dichtheden  $m(z,w)/m(w,z)$  berekend kan worden. Een vervolgonderzoek op dit gebied kan wellicht uitkomst bieden. Het boek [8] uit de literatuurlijst bevat veel theorie over Monte Carlo simulatie op grote toestandsruimten, en vormt dus mogelijk een goed startpunt voor vervolgonderzoek.



## 5. Case studies

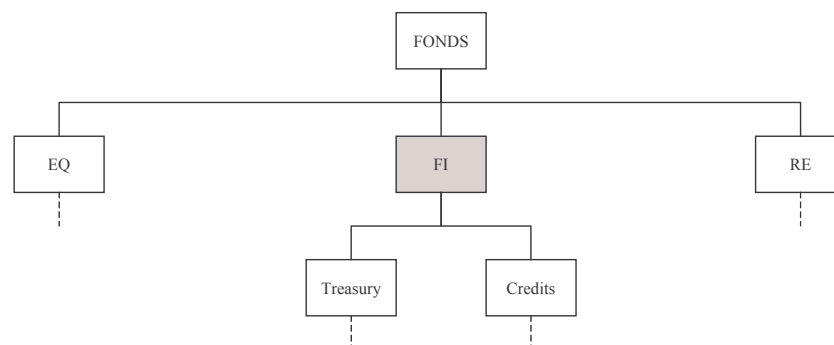
Dit hoofdstuk illustreert de POD performance evaluatietechniek aan de hand van een aantal concrete voorbeelden. De mandaten die we analyseren zijn gebaseerd op mandaten die in werkelijkheid (bij een Nederlandse vermogensbeheerinstelling) voorkomen.

Hier en daar zijn enkele aanpassingen gedaan (of zijn bepaalde regels weggelaten) om het mandaat passend te maken met het model. Omdat de random portefeuilles geen transacties uitvoeren, bleek het soms onmogelijk om te voldoen aan regels zoals een tracking error bovengrens. Het toegelaten gebied was in dergelijke gevallen leeg. Om dit tegen te gaan zijn sommige van deze regels enigszins afgezwakt.

De simulaties zijn uitgevoerd met behulp van het DOS programma. Voor de analyses van de gesimuleerde PODs is gebruik gemaakt van het statistisch pakket R. De paragrafen 5.1 t/m 5.3 bevatten analyses van drie allocatiemandaten; paragrafen 5.4 t/m 5.6 analyseren drie selectiemandaten.

### 5.1 Allocatie: fixed income segment

We beginnen met een eenvoudig mandaat, dat slechts twee onderliggende objecten kent. Het betreft het **fixed income segment** van een vermogensbeheerinstelling, waarbij het kapitaal wordt opgesplitst in een segment voor staatsobligaties (**treasuries**) en bedrijfsobligaties (**credits**). Dit is een belangrijke allocatiebeslissing, omdat treasuries in het algemeen minder volatiel zijn (en dus minder risico met zich meebrengen) dan credits. In figuur 34 is weergegeven welke plaats het beschouwde segment in een IDP boom zou kunnen hebben.



Figuur 34: Positie in IDP structuur.

Ook de mandaatregels zijn voor dit probleem betrekkelijk eenvoudig. Naast het gegeven universum zijn er alleen voor beide onderliggende segmenten een boven- en ondergrens op het te kiezen gewicht gelegd. We bepalen de performance van dit mandaat over een meetperiode van drie jaar: 1 januari 2004 t/m 31 december 2006. Een dergelijk lange periode is zinvol, omdat het indelen van het fixed income kapitaal in treasuries en credits een typische lange termijn beslissing is. In tabel 13 staat een overzicht van de gegevens van dit onderzoek.

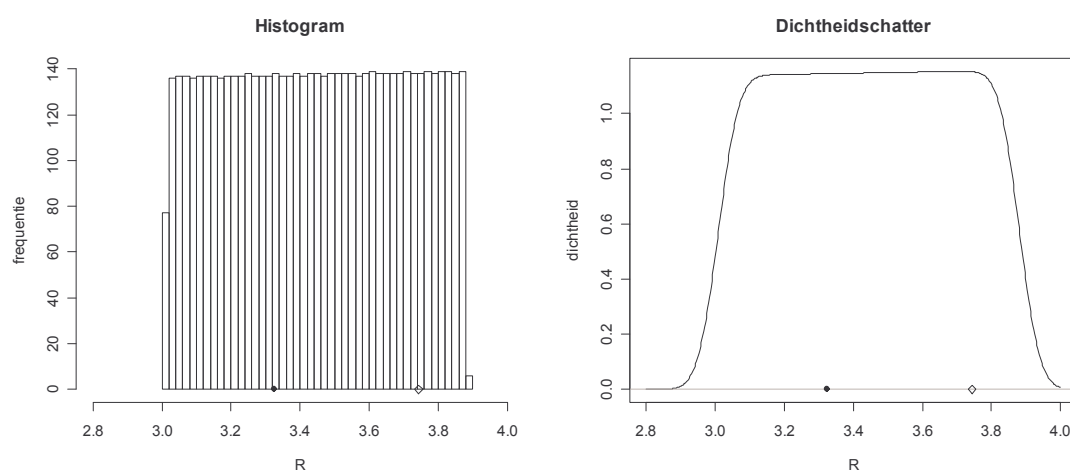
Tabel 13: Karakteristieken van het fixed income segment.

Portefeuille:	<b>Fixed income segment</b>				
Omschrijving:	Allocatie over 2 fixed income segmenten.				
Meetperiode:	Januari 2004 t/m december 2006.				
Mandaatregels:	<ul style="list-style-type: none"> <li>• Universum bestaat uit de onderliggende segmenten Treasury en Credits.</li> <li>• Exposure gewichten liggen binnen de volgende intervallen: <table style="margin-left: 20px;"> <tr> <td>Treasury:</td> <td>[5%, 65%];</td> </tr> <tr> <td>Credits:</td> <td>[35%, 95%].</td> </tr> </table> </li> </ul>	Treasury:	[5%, 65%];	Credits:	[35%, 95%].
Treasury:	[5%, 65%];				
Credits:	[35%, 95%].				

## POD analyse met algoritme 1

Achteraf is bekend dat het treasury segment over 2004-2006 een rendement van 4.383% (geannualiseerd) heeft behaald, en dat het credits segment over de meetperiode een rendement van 2.936% (geannualiseerd) heeft behaald.<sup>5</sup> Gegeven dat het gerealiseerde rendement van het gehele FI segment gelijk is aan 3.744% (geannualiseerd), vragen we ons af of dit een goede prestatie is ten opzichte van wat het mandaat mogelijk maakte.

Om dit te onderzoeken zijn eerst met algoritme 1 alle vectoren uit de 2-dimensionale eenheidssimplex, met discretisatiestap 0.01%, gegenereerd. Binnen enkele seconden waren deze 10001 vectoren langsgelopen. Vervolgens werden door de acceptatie-rejectie procedure 6001 portefeuilles geaccepteerd als zijnde toegelaten met betrekking tot de mandaatregels, en 4000 verworpen. Dit stemt overeen met het feit dat 4/10 van de vectoren buiten de gewichtsgrenzen vallen. Figuur 35 toont een histogram en een dichtheidschatter van de POD, op basis van de rendementen van de 6001 geaccepteerde portefeuilles.<sup>6</sup>



Figuur 35: Histogram en dichtheidschatter via algoritme 1.

De “diamond” op de horizontale as stelt het gerealiseerde rendement voor, en het punt links ervan het benchmark rendement (dat overigens gelijk was aan 3.324% geannualiseerd). We zien dus dat het behaalde rendement in ieder geval in de “rechterhelft” van de POD valt.

<sup>5</sup> We zullen in dit hoofdstuk telkens met geannualiseerde rendementen werken, mits de meetperiode een jaar of langer is.

<sup>6</sup> Bij de dichtheidsschatters in dit hoofdstuk wordt de optimale bandbreedte gekozen volgens het MISE criterium (zie paragraaf 3.4).

De statistieken uit paragraaf 3.5 geven meer kwantitatieve performance resultaten. Allereerst schatten we de POD ranking  $\theta$ , ofwel de kans dat de “aap” de manager verslaat. Deze wordt berekend door de fractie rendementen die boven het gerealiseerde rendement uitkomen:

$$\hat{\theta}_n = 0.158.$$

Paragraaf 3.5 geeft ook een standaard benaderingsmethode voor een betrouwbaarheidsinterval voor  $\theta$ . Voor een betrouwbaarheid van 95% is het resultaat als volgt:

$$\theta \in [0.149, 0.167].$$

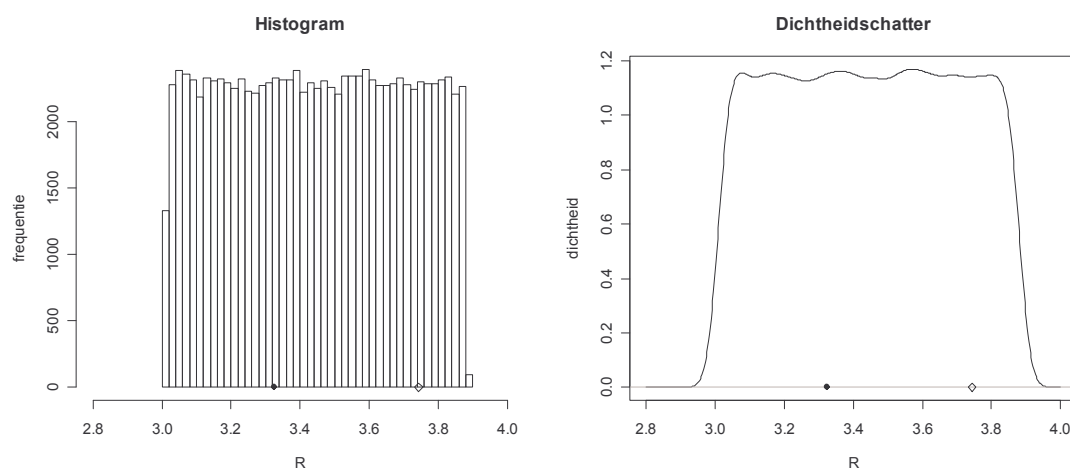
Naast het schatten van de POD ranking is ook een statistische toets uitgevoerd met als nulhypothese dat het gerealiseerde rendement niet boven de mediaan van de POD uitkomt. Aangezien een normale verdeling zeker niet in aanmerking komt als POD, is de variant van de tekentoets uitgevoerd. De resulterende p-waarde blijkt verwaarloosbaar klein te zijn:

$$p = 0.$$

We kunnen dus met grote betrouwbaarheid zeggen dat de strategie van de manager significant beter heeft uitgepakt dan een gemiddelde random strategie.

## POD analyse met algoritme 2

Ter vergelijking is de POD voor dit mandaat ook berekend met algoritme 2. In totaal zijn 100000 trekkingen gedaan, in minder dan een minuut tijd. De acceptatie-rejectie weigerde ongeveer 40% van de getrokken portefeuilles. Dit klopt want de grenzen op de gewichten zorgen ervoor dat precies 60% van de simplex (een lijn in het 2-dimensionale vlak) is toegelaten. In figuur 36 staan het histogram en de dichtheidschatter van de POD die door algoritme 2 is gegenereerd. Zoals verwacht lijkt het resultaat sterk op dat van algoritme 1. Alleen zijn de grafieken iets onregelmatiger omdat de portefeuilles random zijn getrokken en niet regelmatig verspreid liggen over het toegelaten gebied.



Figuur 36: Histogram en dichtheidschatter via algoritme 2.

Ook de bijbehorende schatting van de POD ranking verschilt niet veel met het resultaat van algoritme 1:

$$\hat{\theta}_n = 0.156, \text{ met } 95\% \text{ b.t.i. } \theta \in [0.154, 0.158].$$

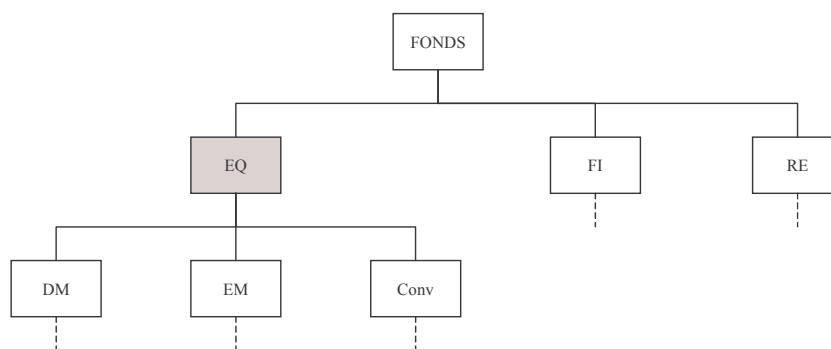
Het betrouwbaarheidsinterval is wel een stuk smaller dan bij algoritme 1. Dit komt uiteraard doordat er meer waarnemingen zijn bij algoritme 2 (namelijk 100000 trekkingen tegen 10001 stuks). De p-waarde van de mediaantoets is ook bij algoritme 2 verwaarloosbaar klein:

$$p = 0.$$

Dus de belegger komt qua rendement significant boven de gemiddelde simulatie uit.

## 5.2 Allocatie: equity segment

Het tweede allocatiemandaat dat we bekijken is een **equity (aandelen) segment**, waarbij het kapitaal wordt verdeeld over drie onderliggende segmenten. Eén segment belegt in de ontwikkelde markt (**developed market, DM**), één in de opkomende markt (**emerging market, EM**) en één in converteerbare obligaties (**convertible bonds, Conv**). Figuur 37 toont een mogelijk positie binnen de IDP structuur.



Figuur 37: Positie in IDP structuur.

De mandaatregels zijn voor dit segment iets gecompliceerder dan het fixed income segment, in de zin dat er naast constraints op exposure gewichten ook een constraint op de ex-ante tracking error voorkomt. De lengte van een subperiode van de tracking error is (net als in de rest van dit hoofdstuk) een maand. De benchmark ten opzichte waarvan de tracking error wordt berekend is een gewogen som van de MSCI Developed World index, de MSCI EMF index, en de UBS Convertible Global index. In tabel 14 staat een overzicht van de mandaatgegevens. De meetperiode bestaat in dit geval uit twee jaar, namelijk 2005 en 2006.



Tabel 14: Karakteristieken van het equity segment.

Portefeuille:	<b>Equity segment</b>
Omschrijving:	Allocatie over 3 fixed equity segmenten.
Meetperiode:	Januari 2005 t/m december 2006.
Mandaatregels:	<ul style="list-style-type: none"> <li>• Universum bestaat uit de onderliggende segmenten DM, EM en Conv.</li> <li>• Exposure gewichten liggen binnen de volgende intervallen: <ul style="list-style-type: none"> <li>DM: [77.8%, 91.6%];</li> <li>EM: [5.5%, 13.9%];</li> <li>Conv: [2.8%, 8.4%].</li> </ul> </li> <li>• Ex-ante mnd. tracking error is ten hoogste 5.5%.</li> </ul>

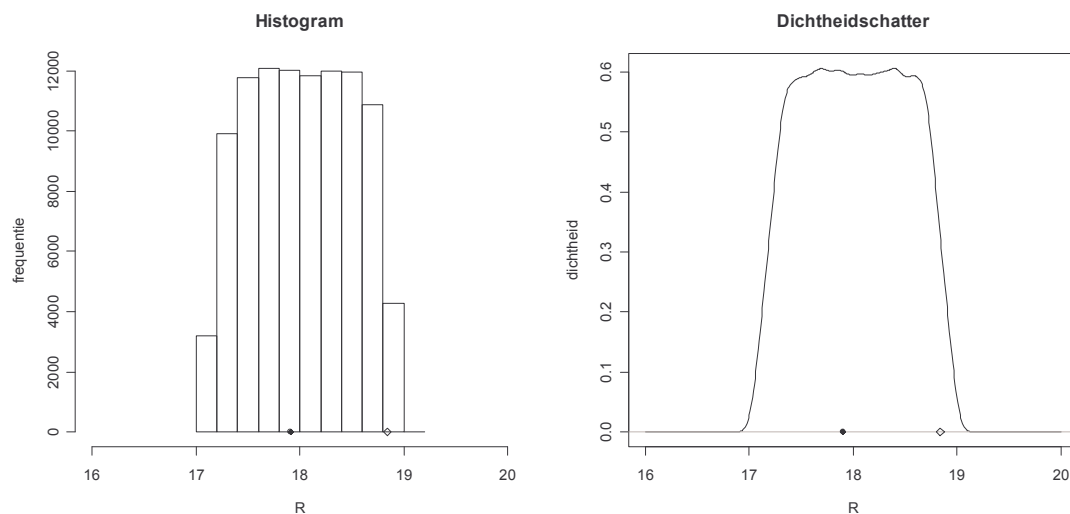
Merk op dat de grens op de ex-ante tracking error in het echte mandaat 1.01% is. Dit leidt echter tot een leeg toegelaten gebied. Om die reden is tracking-error voorwaarde enigszins afgezwakt.

### POD analyse

Het gerealiseerde rendement over 2005-2006 is 18.839% (ann.). Het benchmark rendement is 17.904% (ann.). De rendementen van de onderliggende segmenten zijn 16.40% (ann.) voor DM, 35.11% (ann.) voor EM, en 10.29% (ann.) voor Conv. Er viel dus relatief veel winst te behalen in de opkomende markt, hoewel het mandaat slechts een kleine exposure in deze markt toestaat.

Voor deze portefeuille is de POD berekend via algoritme 2. De verwerpingskans is redelijk groot, ongeveer 99.98%. Desondanks zijn in enkele minuten tijd 100000 trekkingen verricht. Zoals eerder vermeld is steeds de ex-ante tracking error geschat door middel van de eenvoudige procedure die in paragraaf 4.6 beschreven staat. Hierbij wordt de tracking error berekend op basis van 15 maanden vóór aanvang van de meetperiode; dit zijn de maanden oktober 2003 tot en met december 2004.

In figuur 38 staat het histogram en de dichtheidschatter van de gegenereerde POD.



Figuur 38: Histogram en dichtheidschatter van de POD.

De ligging van het gerealiseerde rendement suggereert een betrekkelijk goede performance over 2005-2006. De POD ranking bevestigt deze observatie:

$$\hat{\theta}_n = 0.0291, \text{ met } 95\% \text{ b.t.i. } \theta \in [0.0281, 0.0302].$$

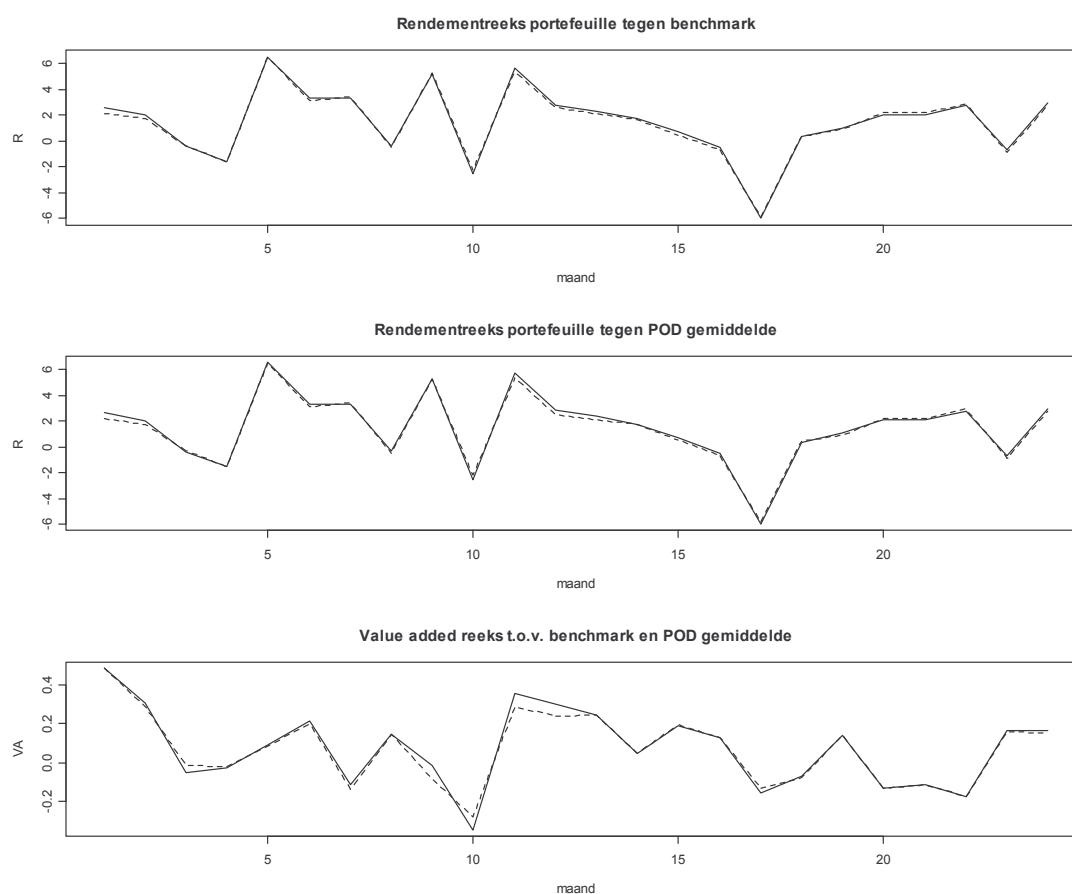
De p-waarde van de mediaantoets geeft aan dat de performance van de manager significant beter is dan dat van een gemiddelde simulatie:

$$p = 0.$$

## Vergelijking met de benchmark

Het is wellicht interessant om te zien hoe de resultaten van een POD analyse eruit zien in vergelijking met een standaard benchmark analyse. Voor dit mandaat is zo'n vergelijking uitgevoerd, waarbij de gemiddelde POD portefeuille wordt opgevat als een custom benchmark.

Figuur 39 toont drie tijdreeksen over de 24 maanden van de meetperiode. De bovenste grafiek bevat de gerealiseerde rendementreeks (ongebroken lijn) en de benchmarkreeks (gebroken zijn). De middelste grafiek vergelijkt de gerealiseerde rendementreeks (ongebroken lijn) met de reeks van het POD gemiddelde (gebroken lijn). De onderste grafiek toont de value added reeksen, dat wil zeggen de verschillen in rendement tussen de portefeuille en de benchmark (gebroken lijn), resp. het POD gemiddelde (ongebroken lijn). We zien dat de verschillen redelijk gelijk opgaan.



Figuur 39: Rendementreeksen.

Tevens voeren we voor beide benchmarks de benchmarktoets uit paragraaf 2.3 uit. Omdat de rendementen niet normaal verdeeld lijken te zijn gebruiken we de variant waarbij een benadering met de centrale limietstelling wordt toegepast. Wanneer deze toets een kleine p-waarde heeft, is de benchmark (met een bepaalde betrouwbaarheid) significant door de belegger verslagen over de meetperiode. Deze p-waarde met betrekking tot de benchmark is bij benadering gelijk aan

$$p^M = 0.0359 .$$

Met betrekking tot het POD gemiddelde is de p-waarde bij benadering gelijk aan

$$p^{POD} = 0.0338 .$$

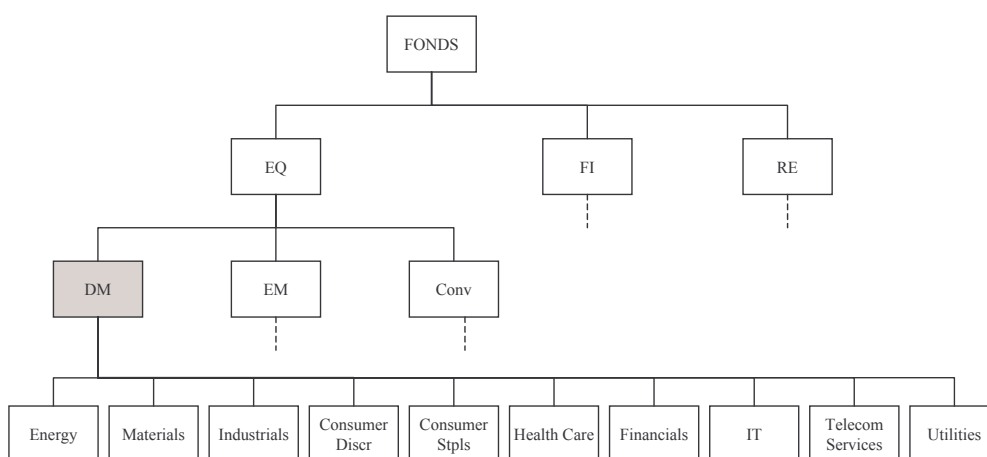
We concluderen dus dat voor dit mandaat de benchmark methode en de POD methode tot nagenoeg dezelfde performance leidt. Dit komt vermoedelijk door de aanwezigheid van een tracking error constraint die de portefeuille dwingt in de buurt van de benchmark te blijven.

### 5.3 Allocatie: sectoren

Het volgende mandaat is een allocatie binnen het aandelensegment – en binnen de ontwikkelde markt – naar de volgende 10 sectoren<sup>7</sup>

- Energy
- Materials
- Industrials
- Consumer Discretionary
- Consumer Staples
- Health Care
- Financials
- Information Technology (IT)
- Telecom Services
- Utilities

Figuur 40 geeft een mogelijke plaats binnen het IDP.



Figuur 40: Positie in IDP structuur.

De meetperiode en de mandaatregels staan in tabel 15. Opvallend is dat gewichtsgrenzen op de tien segmenten ontbreken in de mandaatregels. Merk verder op dat de tracking error wordt gemeten ten opzichte van de MSCI Developed World, de benchmark. De schattingsperiode voor de tracking error duurt van april 2002 tot en met december 2003 (21 maanden).

<sup>7</sup> We gebruiken de naamgevingen zoals die door de MSCI worden gehanteerd.

Tabel 15: Karakteristieken van het sector allocatie segment.

Portefeuille:	<b>Sectorallocatie segment</b>
Omschrijving:	Allocatie over 10 sectoren.
Meetperiode:	Januari 2004 t/m december 2006.
Mandaatregels:	<ul style="list-style-type: none"> <li>• Universum bestaat uit de genoemde 10 onderliggende segmenten.</li> <li>• Ex-ante mnd. tracking error is ten hoogste 7%.</li> </ul>

De echte tracking error grens is 1.73%. Dit is in het voorbeeld vergroot omdat anders een leeg toegelaten gebied ontstaat.

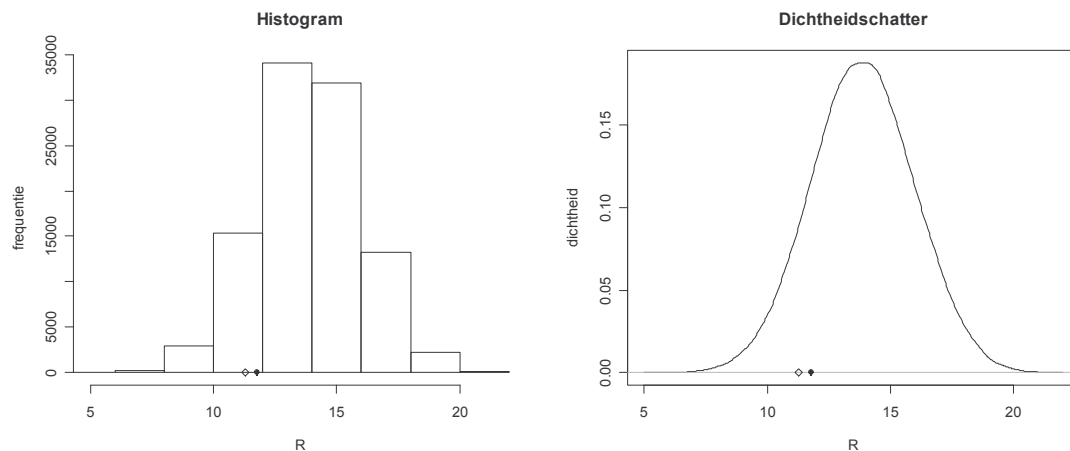
## POD analyse

Het gerealiseerde rendement over de drie gemeten jaren is 11.283% (ann.). Het benchmark rendement is 11.791% (ann.). De rendementen van de segmenten staan hieronder in tabel 16.

Tabel 16 : Rendementen van onderliggende segmenten.

Segment	Ann. rendement (%)
Energy	22.456
Materials	20.365
Industrials	14.893
Consumer Discr.	8.999
Consumer Stpls.	10.741
Health Care	6.993
Financials	15.411
IT	0.697
Telecom Services	10.060
Utilities	24.105

De POD is bepaald via algoritme 2. De verwerpingskans was zeer klein: ongeveer 3%. Derhalve kon in zeer korte tijd 100000 trekkingen verricht worden. Figuur 41 bevat een histogram en een dichtheidschatter.



Figuur 41: Histogram en dichtheidschatter van de POD.

Er is in dit geval dus een onderperformance geleverd: het gerealiseerde rendement ligt links van de “bult” in de POD. De geschatte POD ranking heeft dan ook een redelijk hoge waarde:

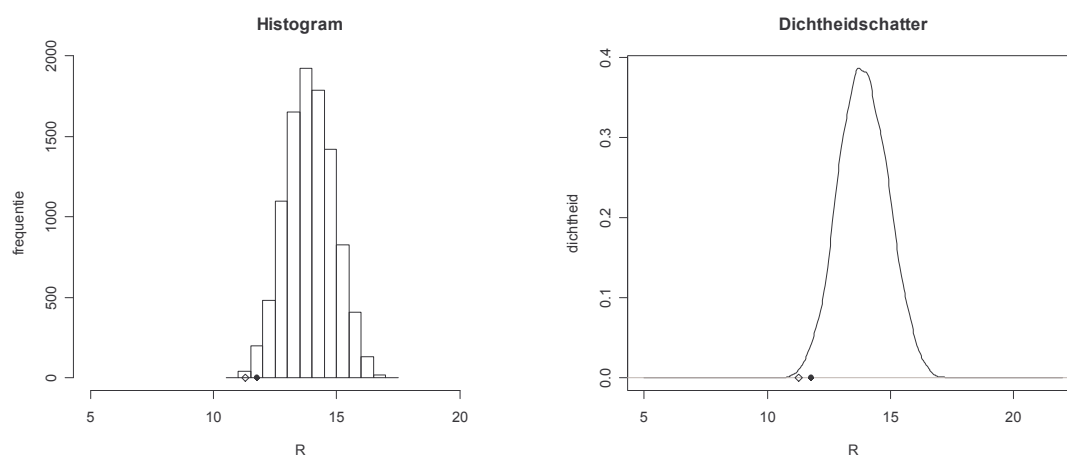
$$\hat{\theta}_n = 0.8930, \text{ met } 95\% \text{ b.t.i. } \theta \in [0.8911, 0.8949].$$

Dit bevestigt dat de performance van dit mandaat over 2004-2006 tegenvalt.

### POD analyse met extra mandaatregel

Zoals opgemerkt bevat het mandaat van deze paragraaf geen grenzen op de gewichten van de 10 sectoren. Als test is ook onderzocht wat de performance geweest zou zijn als zulke grenzen op het mandaat er wel waren geweest, bij hetzelfde gerealiseerde rendement. De ondergrenzen nemen we 5% en de bovengrenzen 25% voor alle onderliggende segmenten.

Ook nu is de POD bepaald via algoritme 2. De verwerpskans is nu echter heel wat groter, namelijk 99.8%. Voor het mandaat zonder gewichtsgrenzen was dit 3%. De extra mandaatregel verkleint het toegelaten gebied dus aanzienlijk. Figuur 42 toont het histogram en de dichtheidschatter.



Figuur 42: Histogram en dichtheidschatter met extra mandaatregel.

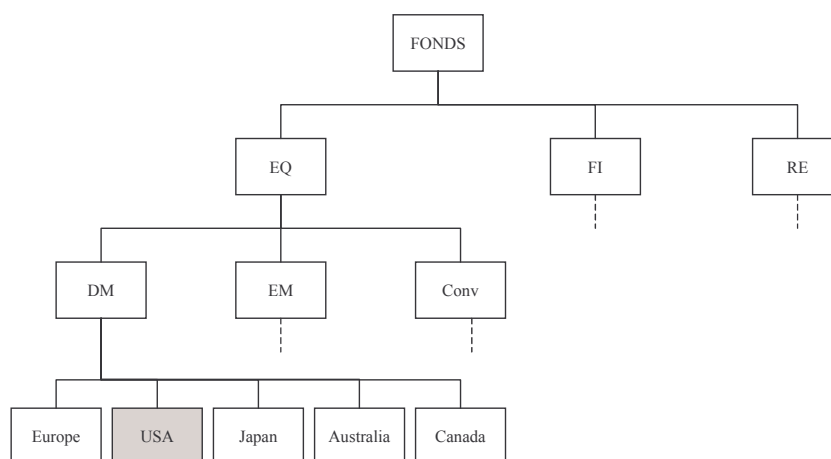
De grafieken suggereren een performance die nog slechter is dan zonder de extra mandaatregel. Inderdaad is de POD ranking nog groter:

$$\hat{\theta}_n = 0.9982, \text{ met } 95\% \text{ b.t.i. } \theta \in [0.9974, 0.9990].$$

Eerder in de scriptie is opgemerkt dat een extra mandaatregel niet noodzakelijk tot een betere performance leidt, gegeven eenzelfde rendement. In bovenstaand voorbeeld maakt de extra regel de performance juist minder goed.

## 5.4 Selectie: aandelen Verenigde Staten

In deze paragraaf analyseren we het eerste selectiemandaat. Dit mandaat vereist dat er wordt belegd in de Amerikaanse aandelenmarkt. Als benadering zijn alle Amerikaanse aandelen genomen die in de meetperiode, het jaar 2006, genoteerd hebben gestaan bij de MSCI. Dit zijn de grootste en belangrijkste aandelen in de VS. Figuur 43 toont de positie binnen het IDP.



**Figuur 43: Positie in IDP structuur.**

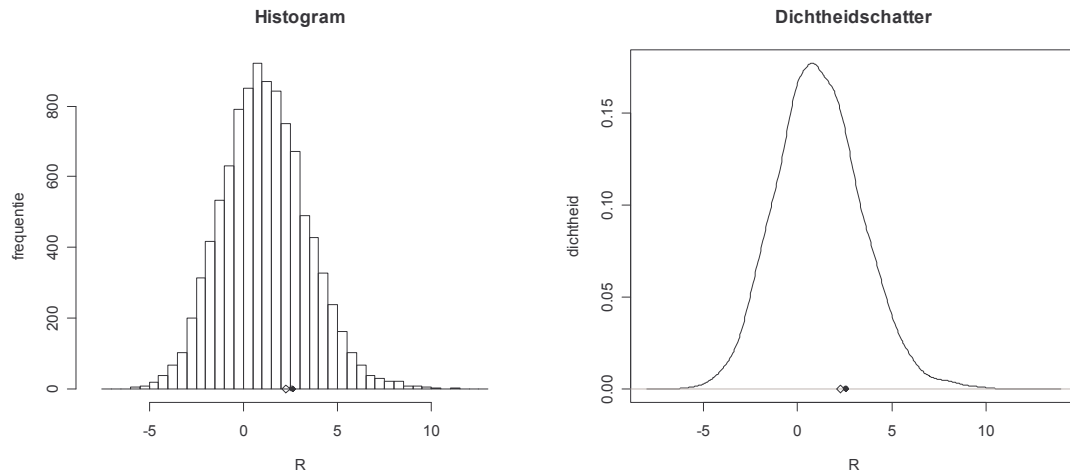
De mandaatregels en andere karakteristieken staan in tabel 17. De beschouwde case is de eerste waarbij er grenzen zijn gesteld op het aantal stukken in de portefeuille.

**Tabel 17: Karakteristieken van de USA aandelenportefeuille.**

Portefeuille:	<b>Aandelen USA</b>
Omschrijving:	Selectie op universum van 651 aandelen.
Meetperiode:	Het jaar 2006.
Mandaatregels:	<ul style="list-style-type: none"> <li>• Universum bestaat uit 651 stukken: de Amerikaanse aandelen die genoteerd staan bij de MSCI.</li> <li>• Exposure gewichten liggen voor alle aandelen binnen het interval [0%, 5%].</li> <li>• Het aantal stukken met een positief gewicht ligt tussen 100 en 200.</li> <li>• Ex-ante mnd. tracking error ligt tussen 1.3% en 1.7%.</li> </ul>

In het echte mandaat zijn de tracking error grenzen 0.43% en 0.58%. Deze zijn echter vergroot om een toegelaten gebied te verkrijgen dat niet leeg is.

Met behulp van algoritme 2 is de POD berekend. Zoals we in hoofdstuk 4 hebben gezien houdt algoritme 2 automatisch (vóórdat de acceptatie-rejectie procedure wordt aangeroepen) rekening met de bovengrens op het aantal stukken. De overige mandaatregels worden door de acceptatie-rejectie gegarandeerd. De verwerpingskans daarbij is ongeveer 93.63, zodat binnen een minuut 10000 trekkingen zijn gedaan. Figuur 44 toont het resulterende histogram en de dichtheidschatter. Het gerealiseerde rendement over 2006 is gelijk aan 2.274% en het benchmark rendement is 2.582%.

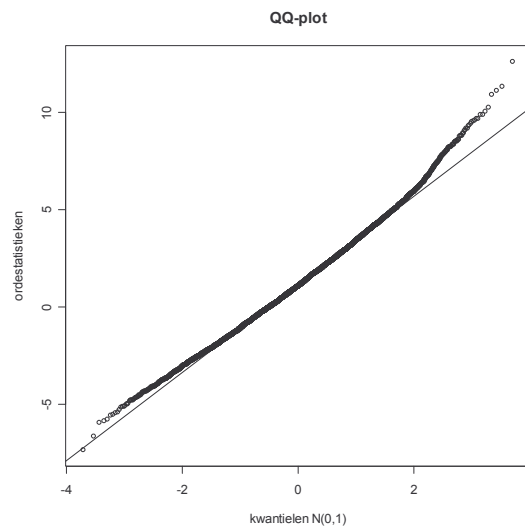


Figuur 44: Histogram en dichtheidschatter van de POD.

De POD-ranking is gelijk aan

$$\hat{\theta}_n = 0.2966, \text{ met } 95\% \text{ b.t.i. } \theta \in [0.2876, 0.3056].$$

Er is dus een outperformance geleverd ten opzichte van de middelste waarde van de POD. Middels een toets kunnen we onderzoeken of het rendement significant boven de middelste waarde ligt. Merk hierbij op dat de klokvormige curve in het histogram en dichtheidschatter doet denken aan een normale verdeling. Mocht de onderliggende verdeling inderdaad normaal zijn, dan kunnen we een t-toets gebruiken in plaats van de mediaantoets. Echter, het QQ-plot in figuur 45 tegen normale kwantielen laat zien dat de vorm van de verdeling vooral aan de randen niet overeenkomt met die van de normale verdeling.



Figuur 45: QQ-plot van de POD t.o.v. de normale verdeling.

Voor de zekerheid voeren we ook de drie in paragraaf 3.4 genoemde toetsen voor normaliteit uit. De Kolmogorov-Smirnov toets verwierpt normaliteit met een p-waarde van 0.0028, en de Shapiro-Wilk toets doet dit met een p-waarde van  $3.95 \cdot 10^{-16}$ . Tot slot berekenen we dat de

Jarque-Bera statistiek gelijk is aan 196.2. Dit is ruim groter dan 6 zodat ook de Jarque-Bera toets het normaal verdeeld zijn van de POD verwerpt met een betrouwbaarheid van 95%.

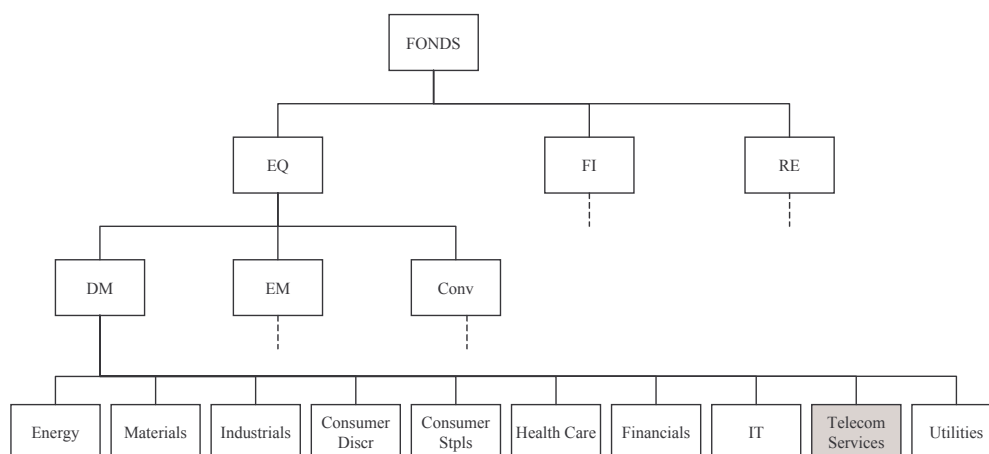
Kortom, we gebruiken geen t-toets om de significantie van de outperformance te testen, maar (net als in de voorgaande paragrafen) de mediaantoets. De resulterende p-waarde blijkt verwaarloosbaar klein te zijn,

$$p = 0,$$

zodat de outperformance inderdaad significant is.

## 5.5 Selectie: aandelen Telecom Services

Het volgende selectiemandaat belegt in MSCI Developed World aandelen binnen de sector **Telecom Services**. Dit segment valt onder het sectorallocatie segment die we eerder in paragraaf 5.3 bestudeerden, zie figuur 46.



**Figuur 46: Positie in IDP structuur.**

De meetperiode is wederom het jaar 2006. In het jaar 2006 bestonden er 51 MSCI stukken die vielen onder de ontwikkelde markt en tevens onder de telecom sector. De dimensie van het probleem is dus 51 (dit is relatief klein voor een selectiemandaat). Verdere gegevens, waaronder de mandaatregels, staan in onderstaande tabel 18.

**Tabel 18: Karakteristieken van de telecom services portefeuille.**

Portefeuille:	<b>Aandelen Telecom Services</b>
Omschrijving:	Selectie op universum van 51 aandelen.
Meetperiode:	Het jaar 2006.
Mandaatregels:	<ul style="list-style-type: none"> <li>• Universum bestaat uit 51 stukken: de MSCI Dev. World aandelen binnen de telecom sector.</li> <li>• Exposure gewichten liggen voor alle aandelen binnen het interval [0%, 10%].</li> <li>• Ex-ante mnd. tracking error is ten hoogste 1.7%.</li> </ul>

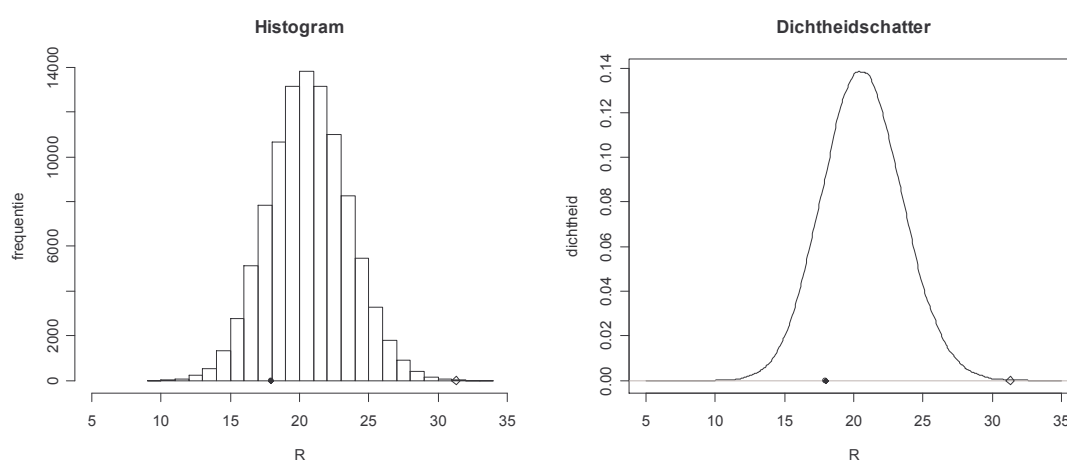


Omdat het aantal stukken in het universum relatief klein is, zijn er bij dit selectiemandaat geen grenzen op het aantal stukken met positieve exposure. De ex-ante tracking error wordt geschat over de 12 maanden vóór de meetperiode. Dit zijn die van 2005.

Gerealiseerd is een rendement van 31.284%. Dit is een flinke outperformance ten opzichte van de benchmark, die over 2006 een rendement van 17.974% behaalde. We zullen zien of de POD analyse ook tot een (zeer) goede performance leidt.

## POD analyse met algoritme 2

Eerst is algoritme 2 gebruikt voor de POD berekening. De verwerpingskans is 98.30%. In ongeveer 15 minuten tijd zijn 100000 portefeuilles getrokken. Het histogram en de dichtheidschatter van de POD staan in figuur 47.



Figuur 47: Histogram en dichtheidschatter via algoritme 2.

We zien dat het gerealiseerde rendement ver in de rechterstaart valt. De POD ranking is dan ook zeer klein:

$$\hat{\theta}_n = 1.4 \cdot 10^{-4}, \text{ met } 95\% \text{ b.t.i. } \theta \in [6.7 \cdot 10^{-5}, 2.1 \cdot 10^{-4}].$$

Zoals te verwachten is er ook een significante outperformance ten opzichte van het gemiddelde. De p-waarde van de mediaantoets is verwaarloosbaar klein:

$$p = 0.$$

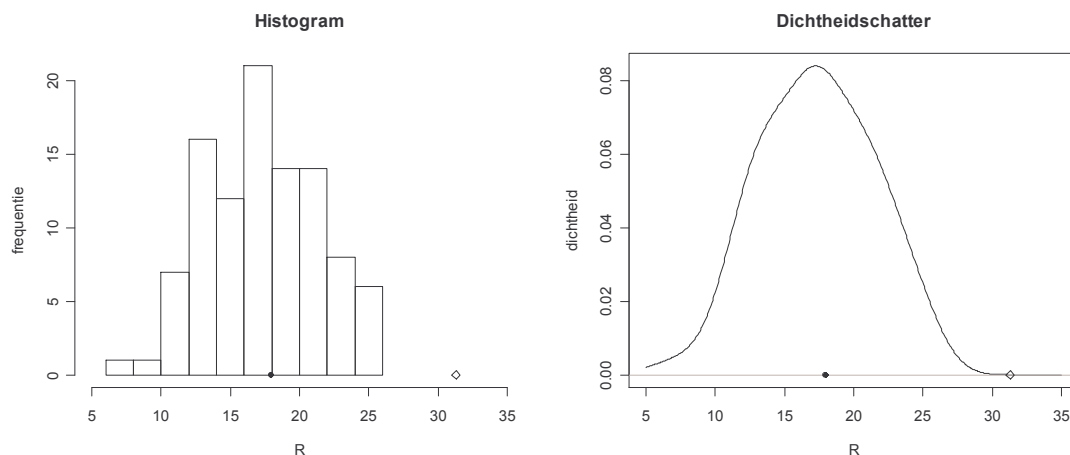
We concluderen dat de gekozen strategie bijna optimaal is ten opzichte van de POD. Hierbij dient echter wel opgemerkt te worden dat de echte portefeuille transacties uitvoert, terwijl de gesimuleerde portefeuilles allemaal buy-and-hold zijn.

## POD analyse met algoritme 3

Voor deze case is de simulatie ter vergelijking ook met algoritme 3 uitgevoerd. Om de simulatie in redelijke tijd te kunnen uitvoeren zijn de stukken in groepen ingedeeld. De maximale grootte van een groep is 4, zodat de dimensie is gereduceerd tot 13. Voor een goede benadering zijn de stukken gegroepeerd op rendement. Dus de 4 laagste rendementen komen in één groep, de 4 opvolgend hogere rendementen in één groep, enzovoorts. Met deze indeling is de simulatie uitgevoerd met 4000 iteraties in de fase met discretisatiestap 1% en

1000 iteraties in de fase met stap 0.1%. In ongeveer een uur tijd zijn er 100 trekkingen gegenereerd.

In figuur 48 zien we het resulterende histogram en de dichtheidschatter.



Figuur 48: Histogram en dichtheidschatter via algoritme 3.

In vergelijking met algoritme 2 is de verdeling meer gespreid en iets naar links verschoven. Vermoedelijk komen deze verschillen door de groepering, en wellicht door het kleine aantal trekkingen (100 stuks) door algoritme 3.

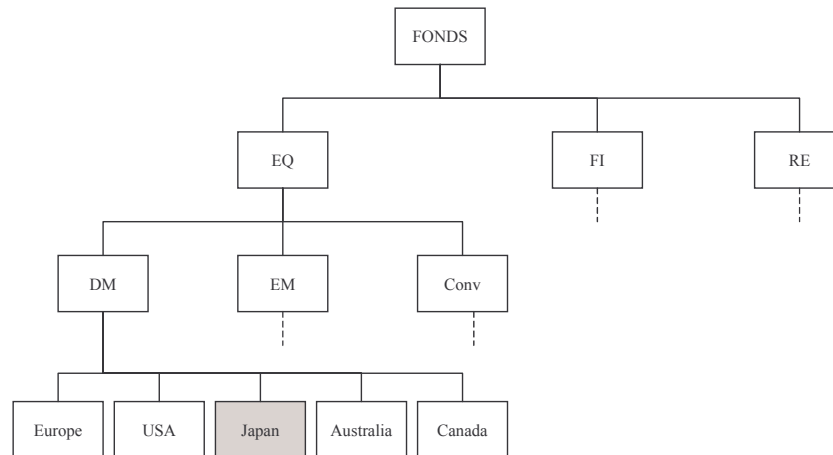
Het gerealiseerde rendement is niet bereikt door de simulatie. Daardoor nemen de POD-ranking en de p-waarde van de mediaantoets gedegeneerde waarden aan:

$$\hat{\theta}_n = 0, \text{ en } p = 0.$$

In ieder geval is de eindconclusie voor beide algoritmes hetzelfde, namelijk dat de performance zeer goed is ten opzichte van de POD!

## 5.6 Selectie: aandelen Japan

Tot besluit van dit hoofdstuk analyseren we een selectiemandaat waarbij het universum bestaat uit alle Japanse aandelen. We benaderen dit universum door alle aandelen die tijdens de meetperiode, het jaar 2006, genoteerd stonden in de MSCI Japan index. Dit zijn in totaal 314 stukken. Figuur 49 geeft de positie van het mandaat in het IDP.



Figuur 49: Positie in IDP structuur.

De karakteristieken van de portefeuille, inclusief de mandaatregels staan in tabel 19.

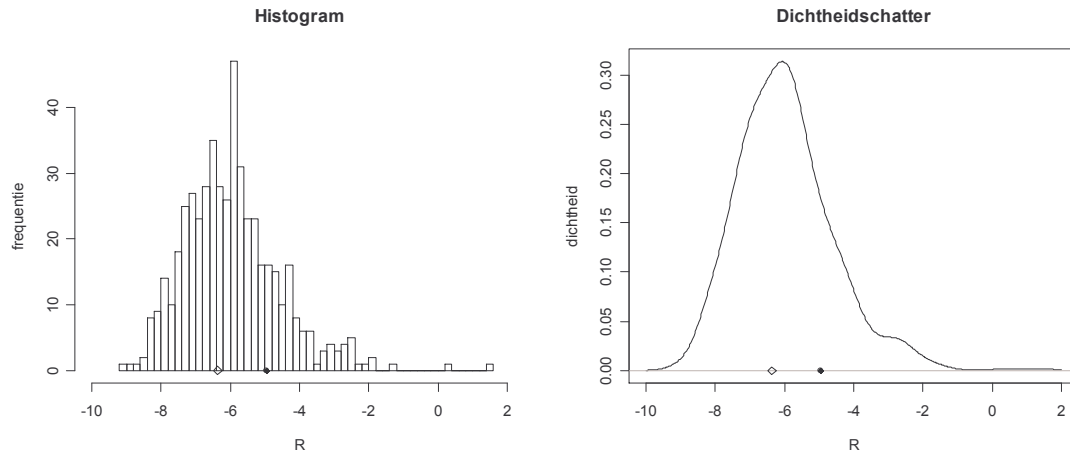
Tabel 19: Karakteristieken van de aandelenportefeuille Japan.

Portefeuille:	<b>Aandelen Japan</b>
Omschrijving:	Selectie op universum van 314 aandelen.
Meetperiode:	Het jaar 2006.
Mandaatregels:	<ul style="list-style-type: none"> <li>• Universum bestaat uit 314 stukken: de Japanse aandelen die genoteerd staan bij de MSCI.</li> <li>• Exposure gewichten liggen voor alle aandelen binnen het interval [0%, 5%].</li> <li>• Het aantal stukken met een positief gewicht ligt tussen 50 en 80.</li> <li>• Ex-ante mnd. tracking error is ten hoogste 1.7%.</li> </ul>

De ex-ante tracking error wordt weer geschat over de 12 maanden van 2005. Voor de POD analyse is gebruik gemaakt van algoritme 3. Hierbij zijn de stukken ingedeeld in groepen van maximaal 29 stukken, zodat er in totaal 11 groepen zijn. We vergelijken bij deze case de uitkomsten ingeval van uniforme trekkingen, én van trekkingen uit de kapitalisatie-gewogen verdeling.

### POD analyse: uniforme verdeling

Met algoritme 3 zijn 500 trekkingen uit de uniforme verdeling gedaan. Dit duurde ongeveer 25 minuten. Per trekking zijn er 1500 iteraties uitgevoerd in de fase met discretisatiestap 1% en 1000 in de fase met stap 0.1%. Figuur 50 toont het histogram en de dichtheidschatter. Het gerealiseerde rendement over 2006 is -6.359% en het benchmark rendement is -4.959%.



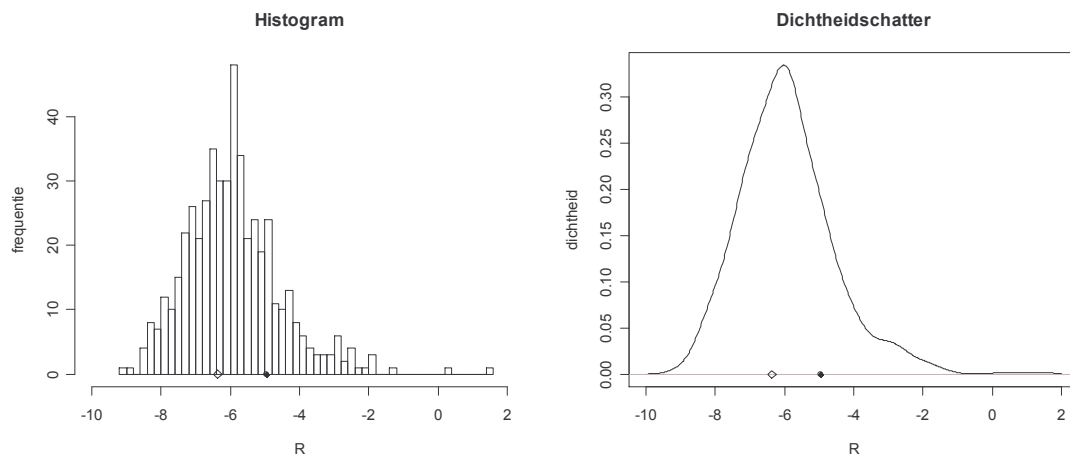
Figuur 50: Histogram en dichtheidschatter bij uniform verdeelde trekkingen.

Er is zo te zien een lichte onderperformance geleverd over het jaar 2006. De POD ranking wordt inderdaad geschat op hoger dan 0.5:

$$\hat{\theta}_n = 0.584, \text{ met } 95\% \text{ b.t.i. } \theta \in [0.541, 0.627].$$

### POD analyse: marktverdeling

Vervolgens bekijken we wat de performance is wanneer met marktwegingen rekening wordt gehouden. Met algoritme 3 zijn er 500 trekkingen uit de kapitalisatie-gewogen verdeling gegenereerd. Behalve de kapitalisatie-wegingen is het algoritme verder op dezelfde manier gebruikt. Dus per trekking zijn er 1500 iteraties uitgevoerd in de fase met discretisatiestap 1% en 1000 in de fase met stap 0.1%. In totaal is het algoritme ongeveer 25 minuten bezig geweest met trekken. Figuur 51 laat het resulterende histogram en de dichtheidschatter zien.



Figuur 51: Histogram en dichtheidschatter bij marktverdeelde trekkingen.

Het resultaat lijkt weinig te verschillen met het geval van uniforme trekkingen. Wanneer we de POD-ranking schatten blijkt deze iets groter te zijn dan in het uniforme geval:

$$\hat{\theta}_n = 0.610, \text{ met } 95\% \text{ b.t.i. } \theta \in [0.567, 0.653].$$

We concluderen voor deze case dat er een kleine onderperformance is geleverd ten opzichte van de POD. Dit zowel voor een uniforme als marktgewogen POD.

## 5.7 Simulatie met herbalancering

Tot besluit van dit hoofdstuk doen we een kleine uitbreiding op de simulatie van alle in dit hoofdstuk beschouwde mandaten. Tot nu toe is steeds de startportefeuille random getrokken en zijn de getrokken portefeuilles niet gewijzigd tijdens de meetperiode. Daardoor blijven de exposure gewichten niet constant maar fluctueren deze gedurende de periode (door koersschommelingen).

In deze paragraaf bekijken we wat het effect op de POD is indien de gewichten na iedere maand worden teruggebracht op hun startwaarden. Hiervoor zijn transacties nodig en dus leidt dit tot transactiekosten. Dit experiment voeren we onder andere uit naar aanleiding van het ontbreken van transacties in de huidige POD simulatie. Dit is een manier om te schatten wat het effect is van regelmatige herbalancerings van de portefeuille.

Het schatten van dit effect doen we als volgt. Aan het einde van iedere maand berekenen we voor ieder object in het universum hoe groot het exposure gewicht van dat object is geworden. Het absolute verschil met het startgewicht geeft aan voor hoeveel gewicht er transacties gedaan moeten worden. Dit getal sommeren over alle objecten én over alle maanden geeft aan voor hoeveel gewicht (ten opzichte van de startwaarde van de portefeuille) er transacties zijn gedaan. We noemen dit de **turnover**.

Wanneer we de turnover vermenigvuldigen met een schatting voor de transactiekosten per euro (in procenten), dan geeft het resultaat aan met welk percentage het behaalde rendement verminderd moet worden door de transactiekosten. Uit de literatuur is bekend dat transactiekosten per euro praktisch altijd tussen 0% en 1% liggen. Wij zullen het meest extreme geval beschouwen en transactiekosten 1% nemen.

Aldus is voor de volgende cases de POD zowel mét als zonder maandelijks herbalancering berekend.

1. Allocatie Fixed Income
2. Allocatie Equity
3. Allocatie naar sectoren
4. Selectie USA
5. Selectie Telecom Services
6. Selectie Japan

In tabel 20 staan de basis kengetallen (gemiddelde en spreiding) van beide PODs voor de zes cases. Ter informatie zijn ook de kengetallen van de turnover opgenomen. Tevens is in de tabel de ratio van de transactiekosten (1% van de turnover) en de standaarddeviatie zonder herbalancering opgenomen. Deze ratio kan geïnterpreteerd worden als een maat voor het effect van de transactiekosten op de locatie van de verdeling.

Tabel 20: Kengetallen van PODs voor zes cases.

Case	1	2	3	4	5	6
Algoritme	2	2	2	2	2	3
Mean zonder rebal. (%)	3.444	18.02	13.87	1.148	20.64	-5.960
Mean met rebal. (%)	3.416	17.77	13.54	1.144	20.47	-6.056
Std. zonder rebal. (%)	0.2518	0.4924	2.076	2.268	2.859	1.391
Std. met rebal. (%)	0.2445	0.4438	2.081	2.213	2.863	1.389
Mean turnover (%)	11.74	12.38	65.00	58.41	51.08	51.17
Std. turnover (%)	3.496	2.654	10.77	2.817	2.261	1.883
1% * Mean turnover (%) / Std. zonder rebal. (%)	0.466	0.251	0.313	0.257	0.179	0.368

In de tabel zien we dat de gemiddelden en standaarddeviaties niet schrikwekkend verschillen voor beide soorten PODs. Dit is te verklaren uit het feit dat de turnover in alle gevallen niet echt groot is (in ieder geval minder dan 100%, dus er is nooit meer geld gemoeid bij transacties dan de startwaarde van de portefeuille). Verder zijn transactiekosten nooit groot ten opzichte van de waarde van de transacties, zelfs niet wanneer we deze zo groot mogelijk (1%) nemen. Hierdoor is ook de ratio transactiekosten/standaarddeviatie relatief klein, zodat we kunnen concluderen dat de transactiekosten geen grote invloed hebben op de locatie van de verdeling.

## 6. Conclusies en aanbevelingen

In de voorgaande hoofdstukken heb ik de techniek van portfolio opportunity distributions besproken. Dit afsluitende hoofdstuk zet de belangrijkste conclusies op een rij en bespreekt de nodige aanbevelingen die gedaan kunnen worden naar aanleiding van het onderzoek. Ook doe ik enkele suggesties voor vervolgonderzoek.

Allereerst kom ik kort terug op de klassieke methoden van performance evaluatie: de benchmarks en peer groups. Deze methoden hebben een aantal tekortkomingen waaraan de POD methode niet onderhevig is.

Bij benchmarks zijn de voornaamste problemen dat de gekozen index meestal niet volledig representatief is voor het mandaat van de manager, en dat de tijdreeks te weinig data geeft (bijvoorbeeld één waarneming per maand) om op statistische gronden een performance uitspraak te kunnen doen. Wanneer echter de POD techniek gebruikt wordt, maken alle regels van het mandaat deel uit van de inputgegevens, en dus weerspiegelt een POD het mandaat van de manager beter dan traditionele benchmarks. Bovendien geeft een POD per subperiode niet één getal, maar een kansverdeling, waarvan het gemiddelde opgevat kan worden als het rendement van een ideale benchmark. De kansverdeling wordt geschat door middel van een dataset met een groot aantal waarnemingen, zodat de performance via statistieken bepaald kan worden.

Peer groups zijn onderhevig aan diverse onzuiverheden, die voortkomen uit het feit dat er over het algemeen weinig concurrent-managers bestaan met een overeenkomstig mandaat, en tevens uit het gegeven dat slecht presterende portefeuilles worden opgeheven en daarmee verdwijnen uit de databases van peer group providers. Het laatstgenoemde verschijnsel treedt niet op bij PODs, omdat deze met gesimuleerde portefeuilles worden bepaald. Ook aan het eerste probleem zijn PODs niet onderhevig, omdat alle getrokken portefeuilles aan het mandaat van de manager voldoen, en het tevens mogelijk is om een zeer groot aantal trekkingen te verrichten.

Om op basis van POD trekkingen tot een performance uitspraak te kunnen komen, dient het door de manager gerealiseerde rendement vergeleken te worden met de POD rendementen. Hiervoor heb ik verschillende statistische technieken besproken.

Allereerst kan een beeld van de POD zelf (de locatie, vorm en spreiding) verkregen worden middels een verdelingsonderzoek. Vervolgens kunnen schattingen, betrouwbaarheidsgebieden, en overschrijdingskansen van toetsen worden berekend, waarbij de getrokken rendementen worden gerelateerd aan het gerealiseerde rendement. De belangrijkste en meest voor de hand liggende statistieken zijn de POD ranking en de p-waarde van een mediaantoets met als nulhypothese dat de “aap” niet is verslagen. Eventueel kan een bootstrap worden gebruikt ingeval het aantal waarnemingen tegenvalt.

Dergelijke statistische analyses geven niet alleen een performance resultaat, zij vergroten ook het inzicht in het effect dat de mandaatregels op de performance hebben. Zo is gebleken dat een extra mandaatregel (bij hetzelfde gerealiseerde rendement) niet noodzakelijk tot een betere performance leidt. Hoe de performance verandert door toevoeging van een regel hangt af van de aard van die regel, én van de grootte van het gerealiseerde rendement.

Naast deze statistische beschouwingen is ook de algoritmische kant van de methode belicht. Voor het trekken van portefeuilles uit de toegelaten verzameling heb ik drie algoritmes ontworpen en uitgewerkt in een klein computerprogramma (dat draait onder DOS en is geschreven in C++).

Het eerste algoritme, genoemd algoritme 1, genereert alle portefeuilles op een discreet rooster, en benadert daarmee de POD zonder random trekkingen te verrichten. Door gebruik te maken van een acceptatie-rejectie procedure worden de portefeuilles die niet aan het mandaat voldoen uit de dataset verwijderd. Helaas is deze methode, vanwege de rekentijd, alleen bruikbaar wanneer de dimensie van het probleem (i.e. het aantal objecten waarin de manager mag beleggen) zeer klein is (maximaal 5). Dit is bijvoorbeeld het geval bij allocatiemandaten met een klein aantal deelsegmenten. Bij dit soort mandaten zijn soms ook boven- en ondergrenzen op de exposure-gewichten aanwezig. Met een dergelijke regel kan algoritme 1 rekening houden vóórdat de acceptatie-rejectie wordt aangeroepen. Dit kan de rekentijd significant verbeteren, maar neemt niet weg dat algoritme 1 slechts bruikbaar is bij kleine dimensies.

Algoritme 2 trekt random portefeuilles uit de volledige eenheidssimplex en verwijdert, net als algoritme 1, de trekkingen die buiten de toegelaten verzameling vallen door middel van acceptatie-rejectie. Het trekken van de portefeuilles verloopt zeer snel en de rekentijd is lineair in de dimensie. Dit komt doordat de trekkingsprocedure in essentie bestaat uit een reeks trekkingen uit een exponentiële verdeling. Het aantal keren per trekking dat dit nodig is, is gelijk aan de dimensie van het mandaat. Vanwege de korte rekentijd is algoritme 2 vaak nuttig voor de toepassing. Helaas echter is algoritme 2 niet bruikbaar wanneer het mandaat zo restrictief is dat de acceptatie-rejectie procedure bijna alle trekkingen verwijdert. We raden daarom aan om algoritme 2 te gebruiken voor allocatiemandaten of selectiemandaten waarbij het mandaat de belegger niet te veel beperkingen oplegt.

De laatste methode, algoritme 3, trekt direct portefeuilles uit de toegelaten verzameling. Dit geschiedt door middel van Markov-keten Monte Carlo (MCMC). Hierbij wordt een trekking verkregen door een stochastisch proces op de toegelaten verzameling te simuleren voor een bepaalde tijd. Een groot probleem bij deze methode is de rekentijd: per trekking moeten duizenden tot miljoenen iteraties van het stochastisch proces worden gedaan. Wel zijn er enkele varianten en uitbreidingen bedacht om de simulatietijd te verkleinen, maar nog steeds kan het minuten lang duren voordat er één trekking is verricht. Ook is het mogelijk dat een afwijkende verdeling ontstaat wanneer de dimensie wordt verlaagd door het groeperen van de stukken. Na het toepassen van dit algoritme op verschillende cases is de conclusie getrokken dat algoritme 3 beter niet gebruikt kan worden wanneer dit niet noodzakelijk is. Maar wanneer bij algoritme 2 de verwerpingskans zo groot is dat vrijwel alle trekkingen worden geweigerd, adviseren we om de trekkingen wel via algoritme 3 te doen. Er is dan immers geen andere optie.

Naast de rekentijd is de kansverdeling een belangrijk issue bij het trekkingsprobleem. Praktisch gezien liggen twee kansverdelingen voor de hand om uit de toegelaten verzameling te trekken. Bij de uniforme verdeling wegen alle portefeuilles even zwaar mee. Bij de marktverdeling (of kapitalisatie-gewogen verdeling) wegen de portefeuilles zwaarder mee naarmate deze belegd zijn in objecten met een grotere marktkapitalisatie. Hierbij treden een aantal problemen op. In de eerste plaats een definitieprobleem: welke verdeling willen we precies? Er zijn vele verdelingen denkbaar met grotere dichtheid in portefeuilles met grotere kapitalisatie. De meest voor de hand liggende kandidaat lijkt een lineaire verdeling, dat wil zeggen dat de dichtheid een lineaire functie van de gewichtsvector is, waarbij de coëfficiënten zich verhouden volgens de marktkapitalisaties. Een nadeel van deze keuze is echter dat het gemiddelde niet gelijk is aan de benchmark portefeuille.

Behalve dit definitieprobleem bij de marktverdeling is het tevens lastig om voor een gegeven marktverdeling een procedure te verzinnen die uit deze verdeling trekt. Wel kan voor de lineaire verdeling algoritme 3 probleemloos worden uitgebreid. Ook voor algoritme 1 en 2 zijn opmerkingen gedaan over mogelijke uitbreidingen voor de marktverdeling, maar deze



methoden voldoen niet helemaal aan alle eisen. Daarom is voor de kapitalisatie-gewogen verdeling eigenlijk alleen algoritme 3 geschikt, met alle nadelen van dien.

Het is dus duidelijk dat er zowel conceptuele als praktische problemen ontstaan bij het trekken uit de marktverdeling. Daarom wordt gebruik van deze verdeling helemaal afgeraden. Men kan dus het beste altijd uit de uniforme verdeling trekken. Dit heeft een duidelijke interpretatie en kan gemakkelijk worden uitgevoerd.

Een belangrijk knelpunt van de POD methode, zoals die tot nu toe is ontwikkeld, is het ontbreken van transacties in de simulatie. Dit betekent overigens niet dat de uitkomst onzinnig is. Het behaalde rendement wordt vergeleken met alle rendementen die via een passieve strategie behaald konden worden. Desalniettemin is de simulatie zonder transacties minder realistisch en kan men tegen het probleem aanlopen dat de toegelaten verzameling leeg is. In dat geval moeten enkele mandaatregels worden verzwakt, of moet de schattingsperiode voor de ex-ante tracking error heel kort worden gemaakt.

Het probleem bij het simuleren van transacties is echter dat men niet precies weet hoe vaak een manager geacht wordt transacties te doen, en ook niet hoe groot deze transacties dan zijn. Dit geeft een keuzeprobleem. Verschillende keuzes kunnen tot verschillende PODs (en daarmee tot verschillende performance) leiden, hetgeen natuurlijk ongewenst is. Verder is het niet duidelijk wat voor effect transactiekosten hebben op het rendement. Deze kosten zelf zullen op een of andere manier geschat moeten worden.

Overigens is in het vijfde hoofdstuk bekeken wat het effect van transacties is wanneer de portefeuille maandelijks wordt teruggebracht op de startgewichten. Voor alle zes cases is geconcludeerd dat de invloed van deze herbalancerings (plus de transactiekosten) op de POD niet erg groot is. De basiskenngetallen (gemiddelde en spreiding) van de berekende POD veranderen niet schrikwekkend.

Verder is in hoofdstuk 5 de POD techniek geïllustreerd op de zes voorbeeldmandaten. Voor alle cases zijn grafieken van de POD opgenomen, en zijn de POD ranking en de p-waarde van de mediaantoets berekend. Daarnaast heb ik alle cases op een specifiek punt onderzocht. Dit heeft geleid tot interessante resultaten, bijvoorbeeld dat algoritme 3 een afwijkende POD kan geven door groepering van de stukken, dat een extra mandaatregel ook tot een slechtere performance kan leiden, en dat gesimuleerde rendementen in het algemeen niet normaal verdeeld zijn.

### **Suggesties voor vervolgonderzoek**

Ik sluit af met een overzicht van alle problemen die nog open staan, en wellicht aanleiding geven voor een nieuw onderzoek.

- Verbeterde schattingstechniek voor de ex-ante tracking error. De huidige standaardtechniek voor het schatten van ex-ante risicomaten is een simulatie op zich, en kost daardoor veel te veel rekentijd. Voor een goede werking van de POD methode is het belangrijk dat een portefeuille in een fractie van een seconde geëvalueerd kan worden op toelaatbaarheid met betrekking tot het mandaat. Het schatten van een ex-ante tracking error moet dan minstens zo snel kunnen verlopen. De schatter die in paragraaf 4.6 is besproken kan weliswaar snel berekend worden, maar over de nauwkeurigheid van deze schatter kan weinig zinnigs gezegd worden.
- Verbetering van algoritme 3 door middel van MCMC zonder discretisatie. De convergentie van algoritme 3 verloopt wellicht sneller wanneer de toestandsovergangen niet beperkt zijn tot enkele naaste burens, maar de toestand sprongen kan maken naar een groter gebied in de toegelaten verzameling. Dit kan mogelijk worden bereikt middels een variant op MCMC

met continue toestandsruimte. In paragraaf 4.6 is een eerste aanzet gegeven tot een (literatuur)onderzoek naar deze methode.

- Definitie van de marktverdeling. Ook al wordt gebruik van een dergelijke verdeling in het geheel afgeraden, de definitie van deze verdeling is een probleem waar in deze scriptie geen oplossing voor wordt gegeven. Van een marktgewogen verdeling verwachten we (zie ook paragraaf 4.1) dat deze een dichtheid heeft die stijgend is in de kapitalisatie van de portefeuille. In het bijzonder dienen de waarden van de dichtheid in de hoekpunten (de portefeuilles geconcentreerd in één stuk) zich onderling te verhouden volgens de marktkapitalisaties. Verder is het een gewenste eigenschap dat het gemiddelde van de verdeling gelijk is aan een benchmark portefeuille, dus dat de gewichten zich verhouden volgens de marktkapitalisaties.
- Uitbreiding op algoritme 2 voor trekkingen uit de marktverdeling. Hebben we eenmaal een geschikte definitie van de marktverdeling, dan zou het bijzonder prettig zijn als naast algoritme 3 ook algoritme 2 gebruikt kan worden voor het genereren van trekkingen uit deze verdeling.
- Uitbreiding van POD methode met gesimuleerde transacties. Zoals hierboven al aangegeven is de POD methode pas echt volledig zodra er ook transacties worden gesimuleerd. Dit vereist onderzoek naar een geschikte transactiefrequentie en een geschikte (grens op de) grootte van een transactie. Waarschijnlijk zal er dan met absolute bedragen gewerkt moeten worden in plaats van alleen met gewichten zoals in deze scriptie gedaan is. Daarnaast zal onderzocht moeten worden hoe per transactie de transactiekosten kunnen worden berekend of geschat.

Van al deze suggesties is, naar mijn persoonlijke mening, de laatste het belangrijkste. Uit het onderzoek van de scriptie is gebleken dat het verwaarlozen van het effect van transacties de resultaten van de methode vaak onzuiver maakt, vooral wanneer de meetperiode redelijk lang is en het mandaat de manager dwingt om de portefeuille regelmatig te herbalanceren.

Mocht er ooit behoefte ontstaan om de POD methode te gebruiken voor een nieuw ORTEC product, of om deze te integreren in een bestaand product zoals PEARL, dan is het van belang om eerst te inventariseren wat de mogelijkheden zijn om het gehele beleggingsproces over een gegeven meetperiode (dus inclusief transacties) te simuleren.

## Literatuurlijst

- [1] Amenc, N., Le Sourd, V., *Portfolio Theory and Performance Analysis*, Wiley, London, 2003.
- [2] Bacon, C.R., *Practical Portfolio Performance Measurement and Attribution*, Wiley, London, 2004.
- [3] Burns, P., “Performance Measurement via Random Portfolios”, working paper, 2004.
- [4] Feibel, B.J., *Investment Performance Measurement*, Wiley, London, 2003.
- [5] Fox, C., *MCMC on Continuous State Spaces*, lecture notes, University of Auckland, 2004.
- [6] Groenendijk, A., Spronk J., “Portfolio Performance Through the Eyes of Monkeys”, *Financiering en Belegging*, vol. **20**, Haveka, Alblasserdam, 1997.
- [7] Harn, K. van, Holewijn, P.J., *Markov-ketens in diskrete tijd*, Epsilon uitgaven, Utrecht, 2-de druk, 2003.
- [8] Robert, C.P., Casella, G., *Monte Carlo Statistical Models*, Springer, New York, 2nd ed., 2004.
- [9] Spaulding, D., *Measuring Investment Performance: Calculating and Evaluating Investment Risk and Return*, McGraw-Hill, New York, 1997.
- [10] Surz, R.J., “Portfolio Opportunity Distributions: An Innovation in Performance Evaluation”, *Journal of Investing*, vol. **3**, no. **2**, summer 1994.
- [11] Surz, R.J., “Portfolio Opportunity Distributions: A Solution to the Problems with Benchmarks and Peer Groups”, *Journal of Performance Measurement*, vol. **1**, no. **2**, winter 1996.
- [12] Surz, R.J., “New and Improved Investment Performance Evaluation”, *Journal of Performance Measurement*, vol. **2**, no. **1**, fall 1997.
- [13] Surz, R.J., “Testimonials on Popular Index Portfolio Opportunity Distributions (PIPODs)”, *Senior Consultant*, vol. **6**, no. **8**, September 2003.
- [14] Surz, R.J., “A Modest Proposal to Modernize Performance Evaluation of Hedge Funds”, *Journal of Performance Measurement*, fall 2005.
- [15] Surz, R.J., “A Fresh Look at Investment Performance Evaluation: Unifying Best Practices to Improve Timeliness and Reliability”, *Journal of Portfolio Management*, summer 2006.
- [16] Tijms, H.C., *A First Course In Stochastic Models*, Wiley, London, 2003.
- [17] Vaart, A. van der, Oosterhof, J, *Algemene statistiek*, collegedictaat, Vrije Universiteit Amsterdam, 2002.
- [18] Vaart, A. van der, Gunst, M. de, *Statistische data analyse*, collegedictaat, Vrije Universiteit Amsterdam, 2004.