



Utrecht University

Leeftijdsafhankelijk Matrix Model en Stochasticiteit

Mohammed El Badaoui

6032745

Begeleider:

Dr. Karma Dajani

Bachelorscriptie

21 Juni 2024

20 juni 2024

Abstract

In deze scriptie bestuderen we de populatiedynamiek door gebruik te maken van een leeftijdsafhankelijk matrixmodel, bekend als de Leslie-matrix, gecombineerd met stochastische elementen van vertakkingsprocessen. Het Leslie-model biedt een deterministische benadering voor het voorspellen van populatiegroei, gebaseerd op leeftijdsgebonden overlevings- en vruchtbaarheidscijfers. We verrijken dit model door stochastische variabelen toe te voegen, wat resulteert in een realistischer weergave van populatiedynamiek. Daarnaast onderzoeken we het langetermijngedrag van het deterministische model door de dominante eigenwaarde en de stabiele leeftijdsverdeling te bepalen, wat cruciale inzichten biedt in de asymptotische groei en de leeftijdsstructuur van populaties.

Deze scriptie integreert deterministische en stochastische methoden in een leeftijdsafhankelijk matrixmodel en demonstreert hun effect via simulaties. De resultaten laten zien dat stochastische modellen een weergave geven van de onvoorspelbare aspecten van populatiedynamiek, terwijl de analyse van de dominante eigenwaarde en stabiele leeftijdsverdeling binnen het deterministische model van groot belang is voor het begrijpen van het langetermijngedrag van populaties.

Mijn oprechte dank gaat uit naar mijn begeleider Dr. Karma Dajani, die nooit te beroerd was om mijn vragen te beantwoorden en voor haar onschatbare begeleiding en ondersteuning tijdens het schrijven van mijn scriptie. Ik ben ook zeer dankbaar voor de waardevolle feedback en samenwerking van mijn studiegenoot Harm Verheggen, die een belangrijke rol heeft gespeeld tijdens dit proces.

Inhoudsopgave

1	Inleiding	5
2	Leeftijdsafhankelijk matrixmodel (De Leslie Matrix)	7
2.1	Het Leslie-model	7
2.1.1	Eigenschappen van de Leslie matrix	10
2.2	Perron en Perron-Frobenius Stelling	14
2.3	Perron	15
2.4	Perron-Frobenius	15
2.5	Bewijs van de Perron-Frobenius Stelling	17
2.6	Toepassing op een willekeurige diagonaliseerbare matrix	22
3	Vertakkingsprocessen	24
3.1	Galton-Watson vertakkingsprocessen	24
3.1.1	kansgenererende functie vertakkingsproces	24
3.2	Uitsterving in Vertakkingsprocessen	28
3.2.1	Classificatie van de Verwachte Waarde	28
4	Van Deterministische naar Stochastisch	31
4.1	Stochastische Simulatie	31
4.2	Bespreking van de codes	33
4.2.1	Deterministische code Bijlage A.1	34
4.2.2	Stochastische code bijlage A.2	35
4.3	Simulaties	36
4.3.1	$\mathbf{n}(0) := [n_1 \ n_2 \ n_3]^T = [1 \ 1 \ 1]^T$	38
4.3.2	$\mathbf{n}(0) := [n_1 \ n_2 \ n_3]^T = [10 \ 10 \ 10]^T$	39
4.3.3	$\mathbf{n}(0) := [n_1 \ n_2 \ n_3]^T = [1000 \ 1000 \ 1000]^T$	40
4.4	Bevindingen simulaties	41

5 Conclusie	42
5.1 Epiloog	42
A Bijlagen	45
A.1 Code voor Deterministische Populatieprojectie in Python	45
A.2 Coder voor Stochastische Populatieprojectie in Python	46

1 Inleiding

In deze scriptie behandelen we een leeftijdsafhankelijk matrixmodel, namelijk de Leslie-matrix. De Leslie-matrix wordt gebruikt om projecties van populaties te maken. Dit houdt in dat het model, gegeven bepaalde hypothesen, bekijkt wat er zal gebeuren als je de huidige situatie naar de toekomst projecteert. Dit verschilt van voorspellingen, waarbij uitspraken worden gedaan over wat er zal gebeuren, terwijl projecties beschrijven wat er zal gebeuren gegeven bepaalde hypothesen.

Bijvoorbeeld, stel dat we een populatie van konijnen hebben en we gebruiken een Leslie-matrix om de populatiegroei te projecteren. Als we aannemen dat elke volwassen konijn elk jaar een bepaald aantal jongen krijgt en dat er een vast percentage van de konijnen elk jaar sterft, kunnen we een projectie maken van de populatiegroei over de komende jaren. Dit is een projectie omdat het gebaseerd is op de aannames over geboortecijfers en sterftcijfers. Een voorspelling zou echter proberen te voorspellen wat er daadwerkelijk met de konijnenpopulatie zal gebeuren, rekening houdend met mogelijke onverwachte gebeurtenissen zoals ziektes of veranderingen in het milieu, die niet in de projectie zijn opgenomen.

Deze hypothesen voor het matrixmodel lijken in eerste instantie sterk en onrealistisch. Door de Leslie-matrix echter een stochastisch karakter te geven, kan een model worden gecreëerd dat dichter bij de realiteit ligt. Dit kan onder andere worden gedaan door (een deel van) het Leslie-model te modelleren als een vertakkingsproces. De geboortecijfers en overgangscijfers, die in het normale Leslie-model deterministisch worden gegeven omdat het vaste percentages per leeftijdsgroep betreft, zullen dan worden omgezet naar stochastische variabelen. Door zowel het deterministische model als het stochastische model te vergelijken, kunnen we uitspraken doen over de bevindingen en hoe het deterministische model zich verhoudt tot het realistischere stochastische model. Dit kan worden gedaan door voor beide modellen simulaties uit te voeren en deze voor verschillende parameters met elkaar te vergelijken, zoals tijd en populatiegrootte.

In de volgende hoofdstukken van deze scriptie zullen we dieper ingaan op verschillende aspecten en uitbreidingen van het Leslie-model en de toepassing van stochastische processen. In Hoofdstuk 2, "Leeftijdsgeclassificeerde Modellen (De Leslie-Matrix)", beginnen we met een gedetailleerde bespreking van het Leslie-model, inclusief de structuur en eigenschappen van de Leslie-matrix [3], [7]. We behandelen ook de Perron-Frobenius-stelling en de toepassing ervan op de Leslie-matrix om belangrijke demografische parameters zoals asymptotische groeisnelheid en stabiele leeftijdsverdeling te bepalen [3], [12], [8]. In Hoofdstuk 3, "Vertakkingsprocessen", introduceren we de theorie van vertakkingsprocessen, met een focus op het Galton-Watson vertakkingsproces, en bespreken we de kansgenererende functies en de dynamiek van populatiegroei en uitsterving binnen deze context [2], [4]. In Hoofdstuk 4, "Van Deterministisch naar Stochastisch", integreren we stochastische elementen van vertakkingsprocessen in het deterministische Leslie-model door gebruik te maken van de Poisson-verdeling voor geboortecijfers en de binomiale verdeling voor overlevingscijfers, waardoor we een meer realistisch model ontwikkelen dat rekening houdt met de variabiliteit in populatiedynamiek [3], [14], [5]. Ook voeren we simulaties uit voor zowel het deterministische als het stochastische model, vergelijken we de resultaten onder verschillende scenario's en

parameterwaarden om inzicht te krijgen in de effecten van stochastische variabiliteit op populatieprojecties, en bespreken we de implicaties van onze bevindingen [13], [15]. In Hoofdstuk 5, "Conclusie", bespreken we de resultaten en afwegingen, en werpen we een blik verder dan alleen de bevindingen van dit onderzoek [3], [4], [15], [2].

Deze scriptie probeert inzicht te bieden in de ontwikkeling van populaties door zowel deterministische als stochastische factoren te combineren, om zo een realistischer en completer beeld te geven van de populatiedynamica.

2 Leeftijdsafhankelijk matrixmodel (De Leslie Matrix)

Een leeftijdsafhankelijk matrixmodel is een wiskundig model dat wordt gebruikt om de dynamiek van populaties te beschrijven. In dit model worden individuen in de populatie geïnclassificeerd op basis van hun leeftijd. Het model maakt gebruik van matrices om de overlevings- en voortplantingskansen voor verschillende leeftijdsgroepen weer te geven. Een bekend voorbeeld hiervan is de Leslie-matrix, die de populatiegroei voorspelt door demografische parameters, zoals leeftijdsspecifieke vruchtbaarheid en overlevingspercentages, in een matrixvorm te organiseren. Deze matrices worden vervolgens gebruikt om de toekomstige populatiestructuur en -groei te berekenen door herhaaldelijke matrixvermenigvuldiging [3].

2.1 Het Leslie-model

Het Leslie-matrixmodel wordt uitgebreid gebruikt in demografische en ecologische studies om populatiedynamiek te vereenvoudigen en te analyseren door individuen in te delen in discrete leeftijdsklassen. Dit model werkt onder de aanname dat overlevingskansen s_i en vruchtbaarheidscijfers f_i constant blijven over deze klassen in de tijd, waarbij een gesloten populatie zonder migratie wordt verondersteld, beoordeeld op regelmatige intervallen. Dit betekent dat de populatie op vaste tijdstippen wordt geëvalueerd, bijvoorbeeld jaarlijks of maandelijks. De tijdseenheid komt overeen met de leeftijdsklasse, aangeduid als het **projectie-interval** (Δt) , waarbij (Δt) een geheel getal is. Als (Δt) gelijk is aan 1, betekent dit één tijdstap. Een projectie-interval is dus de specifieke tijdseenheid tussen opeenvolgende evaluaties van de toestand van een populatie, overeenkomend met de duur die nodig is voor individuen om van de ene leeftijdsklasse naar de volgende over te gaan. Het Leslie-model richt zich in de meeste gevallen op één geslacht, vaak de vrouwen, vanwege hun reproductieve rol.

Het model begint met een specifieke initiële leeftijdsverdelingsvector $\mathbf{n}(0)$ op een begintijdstip $t = 0$. Individen gaan opeenvolgend van de ene leeftijdsklasse naar de volgende, zonder een klasse over te slaan of terug te keren naar een eerdere klasse. Het Leslie-model segmenteert de populatie in discrete leeftijdsklassen binnen een discrete tijdsstructuur. Deze leeftijdsklassen worden geïndexeerd vanaf 1, waarbij leeftijdsklasse i correspondeert met het continu-tijds leeftijdsinterval $i - 1 \leq x \leq i$. De tijd vordert in discrete stappen, die de overgang van de ene leeftijdsklasse naar de andere weerspiegelen, in tegenstelling tot de continue stroom van tijd in werkelijke leeftijdsdemografie.

De populatievector op tijdstip $t + 1$ wordt gegeven door $\mathbf{n}(t + 1) = \mathbf{A}\mathbf{n}(t)$, waarbij \mathbf{A} de Leslie-matrix is. De Leslie-matrix \mathbf{A} bevat de overlevingscijfers en vruchtbaarheidscijfers: de eerste rij bestaat uit vruchtbaarheidscijfers f_i , en de sub-diagonaal bestaat uit overlevingscijfer s_i .

Hoewel deze benadering niet perse elk detail vastlegt, biedt het een gestructureerd systeem voor het maken van projecties in plaats van nauwkeurige voorspellingen.

Ondanks de vereenvoudigende aannames biedt het vermogen van het Leslie Matrix-model om overlevingscijfers en vruchtbaarheidscijfers over leeftijdsklassen te integreren waardevolle inzichten.

Beschouw $i = 1, \dots, m$ als aanduiding voor de i -de leeftijdsklasse. Definieer $n_i(t)$ als het aantal individuen binnen elke leeftijdsklasse op tijdstip t .

Ons doel is om de bevolkingsverdeling te projecteren van tijdstip t naar tijdstip $t + 1$, ervan uitgaande dat de tijdseenheid dezelfde is als de breedte van de leeftijdsklasse, wat dus het projectie-interval is. De populatievector op tijdstip t wordt dan gegeven door:

$$\mathbf{n}(t) = \begin{bmatrix} n_1(t) \\ n_2(t) \\ \vdots \\ n_m(t) \end{bmatrix}$$

De overgang tussen leeftijdsklassen van tijdstip t naar $t + 1$ kan worden omschreven door de volgende vergelijkingen, waarbij s_i de overlevingskans vertegenwoordigt van leeftijdsklasse i naar $i + 1$:

$$\begin{aligned} n_2(t + 1) &= s_1 \cdot n_1(t) \\ n_3(t + 1) &= s_2 \cdot n_2(t) \\ &\vdots \\ n_{i+1}(t + 1) &= s_i \cdot n_i(t) \\ &\vdots \\ n_m(t + 1) &= s_{m-1} \cdot n_{m-1}(t) \end{aligned}$$

De overlevingskansen kunnen worden weergegeven in een overlevingsmatrix:

$$\mathbf{S} = \begin{bmatrix} 0 & 0 & 0 & \dots & 0 & 0 \\ s_1 & 0 & 0 & \dots & 0 & 0 \\ 0 & s_2 & 0 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & s_{m-1} & 0 \end{bmatrix} \quad (1)$$

In de matrix geeft elke s_i het leeftijd specifieke percentage aan waarmee individuen in leeftijdsklasse i overleven en overgaan naar de leeftijdsklasse $i + 1$ tijdens het projectie-interval.

Hoewel de overlevingsmatrix (1) het overleven van individuen van de ene leeftijdsklasse naar de andere leeftijdsklasse beschrijft, incorporeert het geen nieuwe individuen die de eerste leeftijdsklasse binnentreden of expliciet de overleving of sterfte van de oudste leeftijdsklasse, $m - 1$ naar m . De matrix bevat geen overlevingscijfers s_m omdat er vanuit gegaan wordt dat de individuen sterven na de laatste leeftijdsklasse.

De vruchtbaarheidsmatrix \mathbf{F} definieert hoe elke leeftijdsklasse bijdraagt aan de bevolkingsgroei door vruchtbaarheidscijfers f_1, f_2, \dots, f_m in de eerste rij te plaatsen, terwijl alle andere elementen nul zijn. Deze structuur is gebaseerd op de aanname dat alleen de jongste leeftijdsklasse deze reproductieve input ontvangt. De vruchtbaarheidsmatrix wordt weergegeven als:

$$\mathbf{F} = \begin{bmatrix} f_1 & f_2 & \dots & f_{m-1} & f_m \\ 0 & 0 & \dots & 0 & 0 \\ 0 & 0 & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & 0 & 0 \end{bmatrix} \quad (2)$$

Door de matrices \mathbf{S} en \mathbf{F} te combineren, verkrijgen we $\mathbf{S} + \mathbf{F} = \mathbf{A}$, waarbij \mathbf{A} de populatieprojectiematrix wordt genoemd. De leeftijdsgeclassificeerde versie van \mathbf{A} , die we hebben geconstrueerd, staat bekend als de Leslie-matrix. In matrixvorm:

$$\mathbf{A} = \begin{bmatrix} f_1 & f_2 & \dots & f_{m-1} & f_m \\ s_1 & 0 & \dots & 0 & 0 \\ 0 & s_2 & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & s_{m-1} & 0 \end{bmatrix}$$

Nu we de Leslie-matrix \mathbf{A} hebben opgesteld, kunnen we de bevolkingsverdeling projecteren van tijdstip $n(t)$ naar tijdstip $n(t+1)$ door matrix \mathbf{A} te vermenigvuldigen met de populatieverdelingsvector $n(t)$ als volgt:

$$\mathbf{n}(t+1) = \mathbf{A} \cdot \mathbf{n}(t) = \begin{bmatrix} f_1 & f_2 & \dots & f_{m-1} & f_m \\ s_1 & 0 & \dots & 0 & 0 \\ 0 & s_2 & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & s_{m-1} & 0 \end{bmatrix} \begin{bmatrix} n_1(t) \\ n_2(t) \\ \vdots \\ n_{m-1}(t) \\ n_m(t) \end{bmatrix}$$

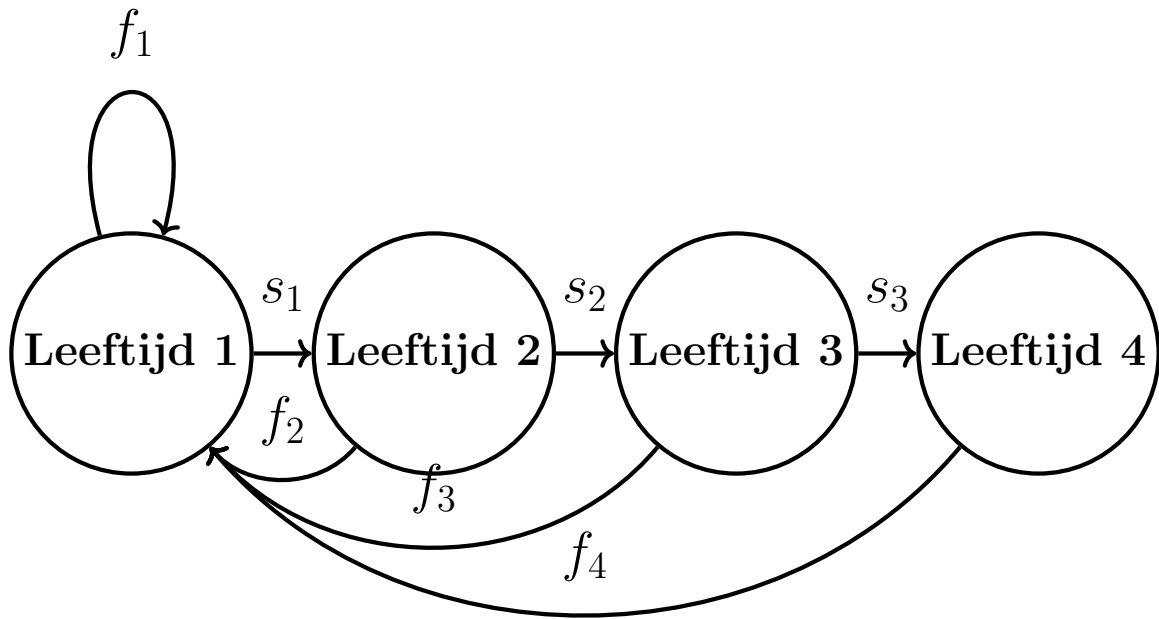
Echter is deze projectie slechts voor één tijdstap. Voor meerdere tijdstappen hebben we de volgende recursie:

$$\begin{aligned} \mathbf{n}(t+1) &= \mathbf{A} \cdot \mathbf{n}(t) \\ \mathbf{n}(t+2) &= \mathbf{A} \cdot \mathbf{n}(t+1) = \mathbf{A} \cdot \mathbf{A} \cdot \mathbf{n}(t) = \mathbf{A}^2 \cdot \mathbf{n}(t) \\ \mathbf{n}(t+3) &= \mathbf{A} \cdot \mathbf{n}(t+2) = \mathbf{A} \cdot \mathbf{A}^2 \cdot \mathbf{n}(t) = \mathbf{A}^3 \cdot \mathbf{n}(t) \\ &\vdots \\ \mathbf{n}(t+k) &= \mathbf{A} \cdot \mathbf{n}(t+(k-1)) = \mathbf{A} \cdot \mathbf{A}^{k-1} \cdot \mathbf{n}(t) = \mathbf{A}^k \cdot \mathbf{n}(t) \end{aligned}$$

Dus, voor k tijdstappen vooruit kan het bevolkingsgroeimodel worden uitgedrukt als:

$$\mathbf{n}(t+k) = \mathbf{A}^k \cdot \mathbf{n}(t) \quad (3)$$

Deze vergelijking beschrijft de bevolkingsverdeling k tijdstappen in de toekomst. Hier stelt \mathbf{A}^k de k -de macht van de Leslie-matrix \mathbf{A} voor, en $\mathbf{n}(t)$ vertegenwoordigt de populatievector op tijdstip t .



Figuur 1: De levenscyclus graaf van een populatie met 4 verschillende leeftijdsklassen.

Door de matrix te vermenigvuldigen met de vector $\mathbf{n}(t)$ krijgen we de volgende lineaire vergelijking:

$$\mathbf{n}(t + 1) = \mathbf{A} \cdot \mathbf{n}(t)$$

Dit lineaire systeem kan worden weergegeven door het volgende levenscyclusdiagram.

2.1.1 Eigenschappen van de Leslie matrix

We zullen nu een aantal resultaten presenteren die afgeleid kunnen worden voor willekeurige Leslie-matrices en die als opstapje dienen voor de verdere theorie van de dominante eigenwaarde en stabiele leeftijdsverdeling, welke we in hoofdstuk 3 zullen behandelen:

Lemma 2.1. *De karakteristieke vergelijking van een Leslie-matrix wordt gegeven door [3]:*

$$1 = \sum_i \left(\prod_{j=1}^{i-1} s_j \right) f_i \lambda^{-i} \quad (4)$$

Bewijs. [7]

Om deze stelling te bewijzen, beginnen we met de karakteristieke vergelijking van de Leslie-matrix \mathbf{A} .

Beschouw de determinant van de matrix $\mathbf{A} - \lambda \mathbf{I}$:

$$\begin{vmatrix} f_1 - \lambda & f_2 & f_3 & \cdots & f_k \\ s_1 & -\lambda & 0 & \cdots & 0 \\ 0 & s_2 & -\lambda & \cdots & 0 \\ 0 & 0 & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & s_{k-1} & -\lambda \end{vmatrix} \quad (5)$$

Hier zetten we de matrix A op door λ maal de identiteitsmatrix I af te trekken. Dit is nodig om de karakteristieke polynoom te vinden waarvan de wortels de eigenwaarden van A zijn.

Nu breiden we de determinant uit langs de eerste rij:

$$\begin{aligned} \det(\mathbf{A}) &= (-1)^{1+1}(f_1 - \lambda) \begin{vmatrix} -\lambda & 0 & \cdots & 0 \\ s_2 & -\lambda & \cdots & 0 \\ 0 & \ddots & \ddots & \vdots \\ 0 & \cdots & s_{k-1} & -\lambda \end{vmatrix} \\ &+ (-1)^{1+2} f_2 \begin{vmatrix} s_1 & 0 & \cdots & 0 \\ 0 & -\lambda & \cdots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & s_{k-1} & -\lambda \end{vmatrix} \\ &+ (-1)^{1+3} f_3 \begin{vmatrix} s_1 & -\lambda & \cdots & 0 \\ 0 & s_2 & \cdots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & s_{k-1} & -\lambda \end{vmatrix} \\ &+ \cdots + (-1)^{1+k} f_k \begin{vmatrix} s_1 & -\lambda & \cdots & 0 \\ 0 & s_2 & \cdots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & s_{k-2} \end{vmatrix} \end{aligned}$$

Vervolgens gebruiken we de stelling voor de Laplace-uitbreiding.

Herken het patroon in de cofactoren:

$$\begin{aligned} &= (f_1 - \lambda)(-1)^{k-1} \lambda^{k-1} - f_2 s_1 (-1)^{k-2} \lambda^{k-2} + f_3 s_1 s_2 (-1)^{k-3} \lambda^{k-3} - \cdots \\ &+ (-1)^{1+k} f_k s_1 s_2 \cdots s_{k-1} \end{aligned} \quad (6)$$

De determinanten van de minoren volgen dus een herkenbaar patroon door de structuur van de Leslie-matrix. Elke minor determinant is zelf een kleinere Leslie-achtige matrix waarvan de karakteristieke polynoom recursief bepaald kan worden.

We kunnen nu vergelijking (6) als volgt uitdrukken:

$$(-1)^{k-1} [\lambda^k - f_1 \lambda^{k-1} - s_1 f_2 \lambda^{k-2} - s_1 s_2 f_3 \lambda^{k-3} - \dots - s_1 s_2 \dots s_{k-1} f_k]$$

Hier combineren we de termen van het patroon dat in de vorige stap werd herkend. Elke term in de uitbreiding draagt bij aan de karakteristieke polynoom, wat leidt tot een polynoom in λ .

Nu vereenvoudigen we:

$$\begin{aligned} 0 &= \lambda^k - f_1 \lambda^{k-1} - s_1 f_2 \lambda^{k-2} - s_1 s_2 f_3 \lambda^{k-3} - \dots \\ &= \lambda^k - \sum_{i=1}^k \left(\prod_{j=1}^{i-1} s_j \right) f_i \lambda^{k-i} \end{aligned}$$

Door de polynoom op deze manier te schrijven, zien we duidelijk de karakteristieke polynoom voor A . De wortels van deze polynoom zijn de eigenwaarden van de Leslie-matrix.

Herschrijven geeft vervolgens:

$$\lambda^k = \sum_{i=1}^k \left(\prod_{j=1}^{i-1} s_j \right) f_i \lambda^{k-i}$$

Ten slotte delen we beide zijden van de vergelijking door λ^k :

$$1 = \sum_{i=1}^k \left(\prod_{j=1}^{i-1} s_j \right) f_i \lambda^{-i}$$

Hieruit volgt dat de eigenwaarden van A de oplossingen zijn van de vergelijking

$$1 = \sum_i \left(\prod_{j=1}^{i-1} s_j \right) f_i \lambda^{-i}$$

□

Stelling 2.2. *De algemene oplossing voor een vergelijking zoals*

$$\mathbf{n}(t+k) = \mathbf{A}^k \cdot \mathbf{n}(t)$$

met \mathbf{A} een diagonaliseerbare Leslie-matrix, is

$$n_i(t+k) = \sum_j c_{ij}(t) \lambda_j^k,$$

waarbij λ_j de eigenwaarden van A zijn en $c_{ij}(t)$ constanten zijn waarvan de waarden afhangen van de initiële populatievector $\mathbf{n}(t)$. Hier is $\mathbf{n}(t)$ de populatieverdeling op tijdstip t of de initiële populatieverdeling [7].

Bewijs. Beschouw de vergelijking

$$\mathbf{n}(t+k) = \mathbf{A}^k \cdot \mathbf{n}(t).$$

En laat \mathbf{A} een diagonaliseerbare matrix zijn, dan kunnen we \mathbf{A} uitdrukken als

$$\mathbf{A} = \mathbf{PDP}^{-1},$$

waarbij \mathbf{P} een matrix is waarvan de kolommen de eigenvectoren van \mathbf{A} zijn, en \mathbf{D} een diagonale matrix is waarvan de diagonaalelementen de eigenwaarden λ_i van \mathbf{A} zijn.

Om \mathbf{A}^k te vinden, schrijven we:

$$\mathbf{A}^k = (\mathbf{PDP}^{-1})^k.$$

Gebruikmakend van de eigenschappen van matrix-exponentiatie, hebben we:

$$\mathbf{A}^k = (\mathbf{PDP}^{-1})(\mathbf{PDP}^{-1}) \dots (\mathbf{PDP}^{-1}) \quad (\text{k keer}).$$

Door de associativiteit van matrixvermenigvuldiging, vereenvoudigt dit tot:

$$\mathbf{A}^k = \mathbf{PD}(\mathbf{P}^{-1}\mathbf{P})\mathbf{D}(\mathbf{P}^{-1}\mathbf{P}) \dots (\mathbf{P}^{-1}\mathbf{P})\mathbf{D}\mathbf{P}^{-1}. \quad (7)$$

Aangezien $\mathbf{P}^{-1}\mathbf{P}$ de identiteitsmatrix I is, vereenvoudigt dit tot:

$$\mathbf{A}^k = \mathbf{PD}^k\mathbf{P}^{-1}.$$

Dus, de vergelijking wordt:

$$\mathbf{n}(t+k) = \mathbf{A}^k \cdot \mathbf{n}(t) = \mathbf{PD}^k\mathbf{P}^{-1} \cdot \mathbf{n}(t).$$

Aangezien de matrix \mathbf{D} diagonaal is, betekent dit dat:

$$\mathbf{D}^k = \text{diag}(\lambda_1^k, \lambda_2^k, \dots, \lambda_n^k). \quad (8)$$

Daarom hebben we:

$$\mathbf{n}(t+k) = \mathbf{P} \text{diag}(\lambda_1^k, \lambda_2^k, \dots, \lambda_n^k) \mathbf{P}^{-1} \cdot \mathbf{n}(t). \quad (9)$$

De i -de component van de vector $\mathbf{n}(t+k)$, aangeduid met $n_i(t+k)$, is een lineaire combinatie van de termen λ_j^k vanwege de matrixvermenigvuldigingen met \mathbf{P} en \mathbf{P}^{-1} :

$$n_i(t+k) = \sum_j c_{ij}(t) \lambda_j^k, \quad (10)$$

waarbij $c_{ij}(t)$ constanten zijn afgeleid van de elementen van de eigenvector-matrix \mathbf{P} , de inverse \mathbf{P}^{-1} , en de initiële populatie vector gegeven door $\mathbf{n}(t)$.

□

We hebben nu in Stelling 2.2 gezien dat we de populatieprojectievergelijking (3) met een diagonaliseerbare Leslie-matrix \mathbf{A} , voor een willekeurig aantal tijdstappen k kunnen herschrijven. Hierbij verkrijgen we vergelijkingen waarbij de totale populatie per leeftijdsgroep, en dus per component van de populatieverdelingsvector op tijdstip $t + k$, kan worden uitgedrukt als een som van termen. Elke term bestaat uit een constante die afhangt van de initiële populatie en de eigenwaarde van de matrix \mathbf{A} opgeheven tot de macht k .

Dit betekent dat we met minder iteratieve berekeningen voor elk toekomstige tijdstip de populatieverdeling kunnen bepalen. De termen λ_j^k geven aan hoe verschillende componenten bijdragen aan de groei of afname van de populatie. Dit stelt ons in staat om een beter inzicht te krijgen in de dynamiek van de populatieverandering, inclusief welke leeftijdsgroepen significant bijdragen aan de totale populatiegroei of -afname. Door de bijdrage van elke eigenwaarde te analyseren, kunnen we identificeren welke factoren dominant zijn in het bepalen van de toekomstige populatie structuur en waar interventies of beleidsmaatregelen mogelijk het meest effectief kunnen zijn.

We zullen dieper ingaan op deze dynamiek en de rol van de eigenwaarden en eigenvectoren van de Leslie-matrix in de volgende paragraaf. Hierbij zullen we kijken naar de Perron en Perron-Frobenius stelling, hoe deze is afgeleid en hoe deze kan worden gebruikt voor Leslie-matrices.

2.2 Perron en Perron-Frobenius Stelling

De stellingen van Perron en Perron-Frobenius spelen een grote rol in de theorie van lineaire algebra en hebben brede toepassingen in verschillende vakgebieden [8], [3], [12]. In het bijzonder zijn deze stellingen essentieel voor het analyseren van Leslie-matrices omdat dit meestal niet-negatieve matrices zijn.

Een matrix $\mathbf{A} \in \mathbb{R}^{m \times n}$ wordt **positief** genoemd als elk element $a_{ij} > 0$, wat genoteerd wordt als $\mathbf{A} > \mathbf{0}$. In het algemeen betekent $\mathbf{A} > \mathbf{B}$ dat elke $a_{ij} > b_{ij}$. Evenzo is \mathbf{A} een **niet-negatieve matrix** als elk element $a_{ij} \geq 0$, genoteerd als $\mathbf{A} \geq \mathbf{0}$. Ook impliceert $\mathbf{A} \geq \mathbf{B}$ dat elke $a_{ij} \geq b_{ij}$.

Het begrijpen van de eigenschappen van de eigenwaarden en eigenvectoren van deze matrices, met behulp van de Perron en Perron-Frobenius stellingen, stelt ons in staat om belangrijke demografische parameters te bepalen en hier uitspraken over te kunnen doen. De belangrijkste van deze parameters zijn de asymptotische groeisnelheid en de stabiele leeftijdsverdeling van de populatie. De asymptotische groeisnelheid, die correspondeert met de grootste eigenwaarde (Perron-eigenwaarde) van de Leslie-matrix, geeft aan hoe de totale populatie groeit op de lange termijn. De bijbehorende eigenvector (Perron-vector) geeft de stabiele leeftijdsverdeling van de populatie, dit betekent dat het percentage van individuen in elke leeftijdsgroep constant blijft in de loop van de tijd, ondanks veranderingen in de totale populatiegrootte. Met deze inzichten kunnen uitspraken gedaan worden over de populatiedynamiek (op de langetermijn). Als gevolg hiervan kunnen bijvoorbeeld effectieve beheerstrategieën voor bevolkingen worden ontwikkeld. Ook reduceren de Perron en Perron-Frobenius de complexiteit van problemen door slechts te focussen op de grootste eigenwaarde

en de corresponderende eigenvector. Dit maakt het mogelijk om met minder rekenkracht toch belangrijke eigenschappen van de matrices te bepalen.

In de volgende paragrafen zullen we de twee stellingen formeel presenteren en vervolgens zullen we een bewijs geven voor de Perron-Frobenius stelling.

2.3 Perron

In 1907 formuleerde Oskar Perron een stelling die zich richtte op de eigenschappen van strikt positieve matrices. Deze stelling, die bekend werd als de Perron stelling, stelt dat een dergelijke matrix een unieke grootste eigenwaarde heeft die reëel en positief is. Verder kan de bijbehorende eigenvector zo worden gekozen dat al zijn componenten positief zijn. Deze unieke grootste eigenwaarde wordt vaak de Perron-eigenwaarde genoemd en de corresponderende eigenvector de Perron-eigenvector [8], [12]. Bovendien kan deze Perron-eigenvector genormaliseerd worden zodat de som van zijn componenten gelijk is aan 1, wat kan worden geschreven als $\|\mathbf{p}\|_1 = 1$. Voordat we de stelling presenteren, zullen we eerst de definitie van de spectrale straal geven:

Definitie 2.3. (Spectrale Straal): Voor een vierkante matrix A wordt de spectrale straal, aangeduid met $\rho(A)$, gedefinieerd als de grootste absolute waarde van de eigenwaarden van A . Formeel,

$$\rho(A) = \max_{\lambda \in \sigma(A)} |\lambda| \quad (11)$$

waarbij $\sigma(A)$ de verzameling van alle eigenwaarden van A vertegenwoordigt [12].

Stelling 2.4. Stelling van Perron voor een positieve matrix $\mathbf{A} \in \mathbb{R}^{n \times n}$ met spectrale straal $r = \rho(\mathbf{A})$, gelden de volgende uitspraken:

1. $r \in \sigma(\mathbf{A})$ (r wordt de Perron-eigenwaarde genoemd) en $r > 0$.
2. De algebraïsche multipliciteit van r is 1.
3. Er bestaat een eigenvector $\mathbf{x} > \mathbf{0}$ zodanig dat $\mathbf{A}\mathbf{x} = r\mathbf{x}$.
4. De Perron-vector is de unieke vector gedefinieerd door

$$\mathbf{A}\mathbf{p} = r\mathbf{p}, \quad \mathbf{p} > \mathbf{0}, \quad \text{en} \quad \|\mathbf{p}\|_1 = 1, \quad (12)$$

en, behalve positieve veelvouden van \mathbf{p} , zijn er geen andere niet-negatieve eigenvectoren voor \mathbf{A} , ongeacht de eigenwaarde [12].

2.4 Perron-Frobenius

Georg Frobenius breidde in 1912 de resultaten van Perron uit naar een bredere klasse van niet-negatieve matrices. Zijn werk, bekend als de Perron-Frobenius stelling, stelt dat een irreducibele niet-negatieve matrix een unieke grootste reële en positieve eigenwaarde heeft.

Voor zulke matrices hebben de bijbehorende eigenvectoren niet-negatieve componenten. Dit betekent dat, zelfs als niet alle elementen van de matrix strikt positief zijn, de grootste eigenwaarde nog steeds reëel en positief is en de corresponderende eigenvectoren niet-negatief zijn [8].

Definitie 2.5. De graaf $G(A)$ van een matrix A wordt gedefinieerd als de gerichte graaf op n knopen $\{N_1, N_2, \dots, N_n\}$ waarbij er een gerichte rand is van knoop N_i naar knoop N_j als en alleen als $a_{ij} \neq 0$ [12].

Definitie 2.6. Een graaf $G(A)$ is sterk **verbonden** als er voor elk paar knopen (N_i, N_k) een gerichte pad bestaat van N_i naar N_k [12].

Een voorbeeld van zo'n graaf is de levenscyclusgraaf die we zagen in Figuur 2 van Hoofdstuk 2. Hierin is te zien dat voor $s_i > 0$ en $f_i > 0$ alle knopen, en dus de leeftijdsgroepen, sterk met elkaar verbonden zijn. Deze sterke verbondenheid kan ook worden geanalyseerd met behulp van de matrix die bij deze graaf hoort. De volgende definitie (2.7) is hierbij van belang.

Definitie 2.7. Een matrix $A \in \mathbb{R}^{n \times n}$ wordt **reducibel** genoemd als er een permutatiematrix P bestaat zodanig dat:

$$P^T A P = \begin{pmatrix} X & Y \\ 0 & Z \end{pmatrix} \quad (13)$$

waarbij X en Z vierkante matrices zijn. Als zo'n permutatiematrix P niet gevonden kan worden, dan noemen we A **irreducibel** [12]

Uit deze definitie kunnen we afleiden dat een matrix A als irreducibel wordt geclassificeerd als deze niet kan worden getransformeerd in een blok boven-driehoeksvorm door permutatie van de rijen en kolommen. Met andere woorden, er is geen permutatiematrix P zodanig dat $P^T A P$ resulteert in een blok boven-driehoeksmatrix. Deze eigenschap geeft aan dat de matrix A sterk verbonden is (2.6), wat betekent dat de componenten niet kunnen worden gescheiden in onafhankelijke submatrices door herordening [9]. De stelling van Perron-Frobenius wordt nu als volgt gegeven:

Stelling 2.8. *Als $\mathbf{A} \in \mathbb{R}^{n \times n}$ een irreducibele niet-negatieve matrix is, dan gelden de volgende eigenschappen:*

1. De spectrale straal $r = \rho(\mathbf{A})$ (ook wel de Perron-eigenwaarde genoemd) en $r > 0$.
2. De algebraïsche multipliciteit van r is 1.
3. Er bestaat een positieve eigenvector $\mathbf{x} > \mathbf{0}$ zodanig dat $\mathbf{A}\mathbf{x} = r\mathbf{x}$.
4. Er is een unieke vector \mathbf{p} gedefinieerd door

$$\mathbf{A}\mathbf{p} = r\mathbf{p}, \quad \mathbf{p} > \mathbf{0}, \quad \text{en} \quad \|\mathbf{p}\|_1 = 1, \quad (14)$$

die de Perron-vector wordt genoemd. Er bestaan geen andere niet-negatieve eigenvectoren voor \mathbf{A} behalve positieve veelvoudigen van \mathbf{p} , ongeacht de eigenwaarde [12].

2.5 Bewijs van de Perron-Frobenius Stelling

Omdat het helpt bij het begrijpen van hoe functies van matrices worden gedefinieerd, met name in het geval van matrices die niet diagonaliseerbaar zijn is het zaak dat we eerst de volgende definitie aankaarten:

Definitie 2.9. Voor een $k \times k$ Jordanblok \mathbf{J} met eigenwaarde λ , en voor een functie $f(z)$ zodanig dat $f(\lambda), f'(\lambda), \dots, f^{(k-1)}(\lambda)$ bestaan, wordt $f(\mathbf{J})$ gedefinieerd door [12]

$$f(\mathbf{J}) = f \begin{pmatrix} \lambda & 1 & & & \\ & \lambda & 1 & & \\ & & \ddots & \ddots & \\ & & & \ddots & \lambda \end{pmatrix} = \begin{pmatrix} f(\lambda) & f'(\lambda) & \frac{f''(\lambda)}{2!} & \dots & \frac{f^{(k-1)}(\lambda)}{(k-1)!} \\ & f(\lambda) & f'(\lambda) & \dots & \frac{f^{(k-2)}(\lambda)}{(k-2)!} \\ & & \ddots & \ddots & \vdots \\ & & & f(\lambda) & f'(\lambda) \\ & & & & f(\lambda) \end{pmatrix}. \quad (15)$$

Lemma 2.10. Spectrale Afbeeldingseigenschap

Als (λ, \mathbf{x}) een eigenwaarde-eigenvectorpaar is voor \mathbf{A} , dan is $(f(\lambda), \mathbf{x})$ een eigenwaarde-eigenvectorpaar voor $f(\mathbf{A})$ wanneer $f(\mathbf{A})$ bestaat. Daarom geldt $f(\mathbf{A})\mathbf{x}_h = f(\lambda_h)\mathbf{x}_h$. Met de eigenschap dat de algebraïsche multipliciteit van λ voor \mathbf{A} gelijk is aan de algebraïsche multipliciteit van $f(\lambda)$ voor $f(\mathbf{A})$ [12].

Dit lemma (2.10) stelt dat de spectrale eigenschappen van een matrix behouden blijven onder een functionele transformatie. Dit is essentieel in het bewijs van de Perron-Frobenius stelling omdat het helpt bij het vertalen van eigenschappen van de oorspronkelijke matrix naar transformaties van die matrix, zoals het verhogen van de matrix tot een macht of het toepassen van een matrixfunctie.

Het volgende lemma (2.11) toont aan dat een positieve matrix een niet-negatieve vector omzet in een strikt positieve vector. Dit is resultaat is van belang voor de Perron en Perron-Frobenius stelling.

Lemma 2.11. Stel dat \mathbf{P} een positieve matrix is. Als \mathbf{x} een niet-negatieve vector is (wat betekent dat alle elementen van \mathbf{x} niet-negatief zijn) en $\mathbf{x} \neq \mathbf{0}$ (wat betekent dat ten minste één element van \mathbf{x} strikt positief is), dan zal $\mathbf{P}\mathbf{x}$ een positieve vector zijn [12].

Bewijs. Stel dat $\mathbf{P} > 0$ en $\mathbf{x} \geq 0$ met $\mathbf{x} \neq \mathbf{0}$. Omdat \mathbf{P} een positieve matrix is, is elk element p_{ij} van \mathbf{P} strikt positief ($p_{ij} > 0$). Omdat \mathbf{x} een niet-negatieve vector is, voldoet elk element x_j van \mathbf{x} aan $x_j \geq 0$. Bovendien, aangezien $\mathbf{x} \neq \mathbf{0}$, is er ten minste één element $x_j > 0$.

Beschouw het i -de element van de productvector $\mathbf{P}\mathbf{x}$, gegeven door:

$$(\mathbf{P}\mathbf{x})_i = \sum_j p_{ij}x_j \quad (16)$$

Elk term in de som $p_{ij}x_j$ is niet-negatief omdat $p_{ij} > 0$ en $x_j \geq 0$. Bovendien, omdat er ten minste één $x_j > 0$ is en het bijbehorende $p_{ij} > 0$, is ten minste één term in de som strikt positief.

Daarom is $(\mathbf{P}\mathbf{x})_i$ een som van niet-negatieve termen, met ten minste één strikt positieve term, wat garandeert dat $(\mathbf{P}\mathbf{x})_i > 0$ voor alle i . Dus, $\mathbf{P}\mathbf{x} > 0$ \square

Lemma 2.12. *Voor de matrix $\mathbf{A}_{n \times n} > \mathbf{0}$ is de enige niet-negatieve eigenvector de Perron-vector \mathbf{p} en eventuele positieve veelvoud daarvan [12].*

Bewijs. Laat (λ, \mathbf{y}) een eigenwaarde-eigenvectorpaar zijn van de matrix \mathbf{A} met $\mathbf{y} \geq \mathbf{0}$. Stel dat \mathbf{p} , de Perron-vector van \mathbf{A}^T , voldoet aan $\mathbf{p} > \mathbf{0}$. Volgens 2.11 moet het inwendig product $\mathbf{p}^T \mathbf{y}$ positief zijn omdat \mathbf{p} strikt positief is en \mathbf{y} niet-negatief.

Toepassend de eigenschappen van de Perron-vector, hebben we:

$$\mathbf{p}^T \mathbf{A} = \rho(\mathbf{A}) \mathbf{p}^T \quad (17)$$

Rekening houdend met het eigenwaarde-eigenvectorpaar (λ, \mathbf{y}) , resulteert dit in:

$$\rho(\mathbf{A}) \mathbf{p}^T \mathbf{y} = \mathbf{p}^T \mathbf{A} \mathbf{y} = \lambda \mathbf{p}^T \mathbf{y} \quad (18)$$

Aangezien $\mathbf{p}^T \mathbf{y} > 0$, volgt hieruit:

$$\rho(\mathbf{A}) = \lambda \quad (19)$$

Dus, λ is gelijk aan $\rho(\mathbf{A})$, en aangezien \mathbf{y} geassocieerd is met deze eigenwaarde, moet \mathbf{y} de Perron-vector \mathbf{p} zijn of een positieve scalaire veelvoud van \mathbf{p} . Dit bevestigt dat \mathbf{y} de unieke niet-negatieve eigenvector van \mathbf{A} is. \square

Lemma 2.12 ondersteunt de uniciteit van de Perron-vector voor niet-negatieve irreducibele matrices. Het bevestigt dat er slechts één niet-negatieve eigenvector is, wat essentieel is voor de eigenschap van de Perron-Frobenius stelling die zegt dat er een unieke positieve eigenvector bestaat die hoort bij de grootste eigenwaarde.

De binomiumstelling voor matrices (2.13) hebben we nu nodig, omdat we hiermee de eigenschappen van matrices zoals $\mathbf{I} + \mathbf{A}$ kunnen analyseren:

Lemma 2.13. Binomiumstelling voor matrices *Voor twee vierkante matrices \mathbf{A} en \mathbf{B} van dezelfde grootte die commuteren (d.w.z. $\mathbf{A}\mathbf{B} = \mathbf{B}\mathbf{A}$), stelt de binomiumstelling voor matrices dat voor elke positief geheel getal n de volgende vergelijking geldt [9]:*

$$(\mathbf{A} + \mathbf{B})^n = \sum_{k=0}^n \binom{n}{k} \mathbf{A}^{n-k} \mathbf{B}^k, \quad (20)$$

Stelling 2.14. *Laat \mathbf{I} de eenheidsmatrix van grootte $n \times n$ zijn en \mathbf{A} een irreducibele niet-negatieve matrix van dezelfde grootte. Dan heeft $(\mathbf{I} + \mathbf{A})^{n-1}$ alleen positieve elementen [12].*

Bewijs. Beschouw $a_{ij}^{(k)}$ als het (i, j) -de element in \mathbf{A}^k . Merk op dat:

$$a_{ij}^{(k)} = \sum_{h_1, h_2, \dots, h_{k-1}} a_{ih_1} a_{h_1 h_2} \cdots a_{h_{k-1} j},$$

wat impliceert dat $a_{ij}^{(k)} > 0$ als en alleen als er een reeks indices h_1, h_2, \dots, h_{k-1} bestaat zodanig dat:

$$a_{ih_1} > 0, a_{h_1h_2} > 0, \dots, a_{h_{k-1}j} > 0.$$

Deze reeks duidt erop dat er een pad van lengte k is van knoop N_i naar knoop N_j in de graafrepresentatie $G(\mathbf{A})$ van de matrix \mathbf{A} .

Aangezien \mathbf{A} irreducibel is, is de graaf $G(\mathbf{A})$ sterk verbonden. Dit betekent dat er een pad is tussen elk paar knopen N_i en N_j voor een bepaalde lengte $k < n$. Dus, voor elk paar (i, j) bestaat er een k zodanig dat $0 \leq k \leq n - 1$ en $a_{ij}^{(k)} > 0$.

Om de positiviteit van $(\mathbf{I} + \mathbf{A})^{n-1}$ vast te stellen, beschouw 2.13

$$(\mathbf{I} + \mathbf{A})^{n-1} = \sum_{k=0}^{n-1} \binom{n-1}{k} \mathbf{A}^k.$$

Het (i, j) -de element van $(\mathbf{I} + \mathbf{A})^{n-1}$ is:

$$[(\mathbf{I} + \mathbf{A})^{n-1}]_{ij} = \left[\sum_{k=0}^{n-1} \binom{n-1}{k} \mathbf{A}^k \right]_{ij} = \sum_{k=0}^{n-1} \binom{n-1}{k} a_{ij}^{(k)}.$$

Aangezien er voor elk paar (i, j) ten minste één k is zodanig dat $a_{ij}^{(k)} > 0$ (door de irreducibiliteit van \mathbf{A}), en alle binomiale coëfficiënten $\binom{n-1}{k}$ positief zijn, volgt hieruit dat:

$$\sum_{k=0}^{n-1} \binom{n-1}{k} a_{ij}^{(k)} > 0.$$

Dus, $(\mathbf{I} + \mathbf{A})^{n-1}$ is een matrix met alle positieve elementen, wat bevestigt dat:

$$(\mathbf{I} + \mathbf{A})^{n-1} > \mathbf{0}.$$

□

Deze stelling (2.14) speelt een sleutelrol in het bewijs van de Perron-Frobenius stelling omdat het aantoont dat de matrix $(\mathbf{I} + \mathbf{A})^{n-1}$ strikt positief is. Dit is belangrijk om te bewijzen dat de grootste eigenwaarde van \mathbf{A} , namelijk $\rho(\mathbf{A})$, een unieke positieve eigenvector heeft.

Lemma 2.15. *Voor een niet-negatieve matrix $\mathbf{A} \in \mathbb{R}^{n \times n}$ met spectrale straal $r = \rho(\mathbf{A})$, gelden de volgende punten:*

1. r is een eigenwaarde van \mathbf{A} (merk op dat r nul kan zijn).
2. Er bestaat een vector \mathbf{z} zodanig dat $\mathbf{z} \in \mathcal{N} = \{\mathbf{x} \mid \mathbf{x} \geq \mathbf{0}, \mathbf{x} \neq \mathbf{0}\}$, wat betekent dat \mathcal{N} bestaat uit alle vectoren \mathbf{x} zodanig dat elke component van \mathbf{x} niet-negatief is en \mathbf{x} niet de nulvector is, zodanig dat $\mathbf{Az} = r\mathbf{z}$ [12].

Bewijs. Beschouw de reeks positieve matrices $\mathbf{A}_k = \mathbf{A} + (1/k)\mathbf{E} > 0$, waarbij \mathbf{E} de matrix van alleen enen is. Laat $r_k > 0$ en $\mathbf{p}_k > 0$ de Perron-wortel en Perron-vector zijn voor \mathbf{A}_k , respectievelijk. Aangezien de reeks $\{\mathbf{p}_k\}$ binnen de eenheids 1-bol in \mathbb{R}^n ligt, is deze begrensd. Volgens de stelling van Bolzano-Weierstrass bestaat er dan een convergente deelreeks $\{\mathbf{p}_{k_i}\}$ zodanig dat:

$$\mathbf{p}_{k_i} \rightarrow \mathbf{z}$$

met $\mathbf{z} \geq 0$ en $\mathbf{z} \neq 0$ omdat $\mathbf{p}_{k_i} > 0$ en $\|\mathbf{p}_{k_i}\|_1 = 1$.

Aangezien $\mathbf{A}_k > \mathbf{A}_{k+1} > \dots > \mathbf{A}$, moeten we het volgende resultaat vaststellen: Voor elke matrix \mathbf{A} , duiden we met $|\mathbf{A}|$ de matrix met elementen $|a_{ij}|$ aan. Als $|\mathbf{A}| \leq \mathbf{B}$, waarbij \mathbf{B} ook een matrix is, dan geldt:

$$\rho(\mathbf{A}) \leq \rho(|\mathbf{A}|) \leq \rho(\mathbf{B}).$$

De driehoeksongelijkheid geeft dan $|\mathbf{A}^k| \leq |\mathbf{A}|^k$ voor elke positief geheel getal k . Bovendien impliceert $|\mathbf{A}| \leq \mathbf{B}$ dat $|\mathbf{A}|^k \leq \mathbf{B}^k$. Dus, we hebben:

$$\begin{aligned} \|\mathbf{A}^k\|_\infty &= \| |\mathbf{A}^k| \|_\infty \leq \| |\mathbf{A}|^k \|_\infty \leq \|\mathbf{B}^k\|_\infty \\ &\implies \|\mathbf{A}^k\|_\infty^{1/k} \leq \| |\mathbf{A}|^k \|_\infty^{1/k} \leq \|\mathbf{B}^k\|_\infty^{1/k} \\ &\implies \lim_{k \rightarrow \infty} \|\mathbf{A}^k\|_\infty^{1/k} \leq \lim_{k \rightarrow \infty} \| |\mathbf{A}|^k \|_\infty^{1/k} \leq \lim_{k \rightarrow \infty} \|\mathbf{B}^k\|_\infty^{1/k} \\ &\implies \rho(\mathbf{A}) \leq \rho(|\mathbf{A}|) \leq \rho(\mathbf{B}). \end{aligned}$$

Dus, $r_1 \geq r_2 \geq \dots \geq r$. Daarom is $\{r_k\}$ een monotone reeks van positieve getallen die van onderen begrensd zijn door r , dus:

$$\lim_{k \rightarrow \infty} r_k = r' \geq r.$$

Aangezien $\lim_{k \rightarrow \infty} \mathbf{A}_k = \mathbf{A}$, en gebruikmakend van het feit dat de limiet van een product de product van de limieten is (mits alle limieten bestaan), verkrijgen we:

$$\mathbf{Az} = \lim_{i \rightarrow \infty} \mathbf{A}_{k_i} \mathbf{p}_{k_i} = \lim_{i \rightarrow \infty} r_{k_i} \mathbf{p}_{k_i} = r' \mathbf{z}.$$

Dus, $\mathbf{Az} = r' \mathbf{z}$ met $\mathbf{z} \geq 0$ en $\mathbf{z} \neq \mathbf{0}$. Aangezien $r' \geq r$ en $r = \rho(\mathbf{A})$, volgt hieruit dat $r' = r$. Daarom:

$$\mathbf{Az} = r \mathbf{z}.$$

Bijgevolg is $r^* = r$, en $\mathbf{Az} = r \mathbf{z}$ met $\mathbf{z} > \mathbf{0}$ en $\mathbf{z} \neq \mathbf{0}$. Dit voltooit het bewijs van de stelling. \square

We hebben nu aangetoond dat de spectrale straal van een niet-negatieve matrix een corresponderende niet-negatieve eigenvector heeft. Nu we alle bouwstenen hebben zijn we klaar om het bewijs van de Perron-Frobenius stelling te geven.

Bewijs. (Perron-Frobenius) Uit 2.15 weten we dat als \mathbf{A} een niet-negatieve irreducibele matrix is, dan geldt dat $r = \rho(\mathbf{A}) \in \sigma(\mathbf{A})$, wat betekent dat r een eigenwaarde is van \mathbf{A} .

Om aan te tonen dat de algebraïsche multipliciteit van r gelijk is aan 1, beschouwen we de matrix $\mathbf{B} = (\mathbf{I} + \mathbf{A})^{n-1}$, die positief is volgens 2.14. Nu geldt volgens 2.9, $\lambda \in \sigma(\mathbf{A})$ als en alleen als $(1 + \lambda)^{n-1} \in \sigma(\mathbf{B})$, en de algebraïsche multipliciteit van λ in \mathbf{A} is hetzelfde als die van $(1 + \lambda)^{n-1}$ in \mathbf{B} .

Laat $\mu = \rho(\mathbf{B})$. Dan,

$$\mu = \max_{\lambda \in \sigma(\mathbf{A})} |1 + \lambda|^{n-1} = \left(\max_{\lambda \in \sigma(\mathbf{A})} |1 + \lambda| \right)^{n-1} = (1 + r)^{n-1}.$$

Dus, $\mu = (1 + r)^{n-1}$.

Stel dat de algebraïsche multipliciteit van r in \mathbf{A} groter is dan 1. Bijgevolg zou de algebraïsche multipliciteit van μ in \mathbf{B} ook groter zijn dan 1. Dit is echter in tegenspraak met het feit dat \mathbf{B} positief is, omdat een positieve matrix geen eigenwaarde kan hebben met een algebraïsche multipliciteit groter dan 1. Daarom moet de algebraïsche multipliciteit van r in \mathbf{A} 1 zijn.

Om te bewijzen dat \mathbf{A} een positieve eigenvector heeft die correspondeert met r , herinner dat er uit 2.15 een niet-negatieve eigenvector $\mathbf{x} \geq 0$ bestaat die geassocieerd is met r .

als we nu 2.10 toepassen, volgt dat als (λ, \mathbf{x}) een eigenwaarde-eigenvectorpaar is voor \mathbf{A} , dan is $(f(\lambda), \mathbf{x})$ een eigenwaarde-eigenvectorpaar voor $f(\mathbf{A})$. Specifiek, (r, \mathbf{x}) zijnde een eigenwaarde-eigenvectorpaar voor \mathbf{A} impliceert dat (μ, \mathbf{x}) een eigenwaarde-eigenvectorpaar is voor \mathbf{B} .

Aangezien \mathbf{B} positief is, moet \mathbf{x} volgens 2.12 een positieve veelvoud zijn van de Perron-vector van \mathbf{B} . Daarom moet \mathbf{x} inderdaad positief zijn.

Ten slotte, $r > 0$; anders zou $\mathbf{A}\mathbf{x} = 0$ zijn, wat onmogelijk is omdat $\mathbf{A} \geq 0$ en $\mathbf{x} > 0$ impliceert dat $\mathbf{A}\mathbf{x} > 0$. Daarom is $r > 0$.

Dit voltooit het bewijs van de Perron-Frobenius stelling. De spectrale straal r van een niet-negatieve irreducibele matrix \mathbf{A} is een eigenwaarde met algebraïsche multipliciteit 1, en \mathbf{A} heeft een positieve eigenvector die geassocieerd is met r . \square

2.6 Toepassing op een willekeurige diagonaliseerbare matrix

Voor een niet-negatieve, irreducibele en diagonaliseerbare projectiematrix \mathbf{A} zoals in 2.2, kan het asymptotische gedrag worden onderzocht via de stelling van Perron-Frobenius 2.8. Volgens 2.8 bestaat er een unieke grootste reële eigenwaarde $\lambda_1 = \rho(\mathbf{A})$, de spectrale straal, die correspondeert met de eigenvector \mathbf{v}_1 met strikt positieve componenten. Alle andere eigenwaarden λ_i voldoen aan de voorwaarde $|\lambda_i| < \lambda_1$.

Om de toepassing van de van de Perron-Frobenius stelling op deze projectie matrix nog wat verder te kunnen uitbreiden, geven we de volgende definitie en stelling:

Definitie 2.16. Een niet-negatieve en irreducibele matrix \mathbf{A} wordt **primitief** genoemd als deze positief wordt wanneer deze tot een voldoende hoge macht wordt verheven, dat wil zeggen, als A^k strikt positief is voor een geheel getal $k > 0$ [3], [12], [11].

Stelling 2.17. Zij $\mathbf{A} \geq 0$ een niet-negatieve, primitieve matrix. Laat $\rho(\mathbf{A}) = \lambda_1$ de spectrale straal van \mathbf{A} zijn. Dan geldt het volgende [12]:

1. Er bestaat een unieke (rechter) eigenvector \mathbf{v}_1 van \mathbf{A} , die positief genomen kan worden, zodanig dat $\mathbf{A}\mathbf{v}_1 = \lambda_1\mathbf{v}_1$ en $\|\mathbf{v}_1\|_1 = v_{11} + \dots + v_{1n} = 1$.
2. Er bestaat een unieke (linker) eigenvector \mathbf{u}_1^T van \mathbf{A} , die positief genomen kan worden, zodanig dat $\mathbf{u}_1^T\mathbf{A} = \lambda_1\mathbf{u}_1^T$ en $\mathbf{u}_1^T \cdot \mathbf{v}_1 = v_{11}u_{11} + \dots + v_{1n}u_{1n} = 1$.
3. $\lim_{m \rightarrow \infty} \left(\frac{1}{\lambda_1} \mathbf{A} \right)^m = \frac{\mathbf{v}_1\mathbf{u}_1^T}{\mathbf{u}_1^T \cdot \mathbf{v}_1} = \mathbf{v}_1\mathbf{u}_1^T > 0$.

Stelling 2.17(3) zegt dus dat als we de matrix \mathbf{A} schalen door $\frac{1}{\lambda_1}$ en deze resulterende matrix tot de macht t verheffen, de limiet van deze operatie als t naar oneindig gaat, zal convergeren naar de genormaliseerde $n \times n$ projectiematrix $\mathbf{v}_1\mathbf{u}_1^T$, voor het bewijs van stelling 2.17, zie Meyer [12]. Zij \mathbf{A} nu een diagonaliseerbare en primitieve matrix, dan kan deze worden uitgedrukt als $\mathbf{A} = \mathbf{P}\mathbf{D}\mathbf{P}^{-1}$, waarbij \mathbf{D} een diagonaal matrix is met de eigenwaarden $\lambda_1, \lambda_2, \dots, \lambda_n$ op de diagonaal, en \mathbf{P} een matrix is waarvan de kolommen de corresponderende eigenvectoren zijn. Als we nu alle eigenwaarde van de diagonaal matrix \mathbf{D} verheffen tot de macht t en λ_1^t uit de matrix factoreren, krijgen we de volgende matrix:

$$\mathbf{D}^t = \lambda_1^t \begin{bmatrix} 1 & 0 & \dots & 0 \\ 0 & \left(\frac{\lambda_2}{\lambda_1}\right)^t & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \left(\frac{\lambda_n}{\lambda_1}\right)^t \end{bmatrix} \quad (21)$$

Voor grote waarde van t kan \mathbf{D}^t dan worden geschat als:

$$\mathbf{D}^t \approx \lambda_1^t \begin{bmatrix} 1 & 0 & \dots & 0 \\ 0 & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 0 \end{bmatrix} \quad (22)$$

Aangezien $|\lambda_i/\lambda_1| < 1$ voor alle $i \geq 2$, worden hun bijdragen verwaarloosbaar voor grote t .

Als we nu dit resultaat 22 substitueren in de projectievergelijking en stelling 2.17 toepassen verkrijgen we: $\mathbf{n}(t) = \mathbf{A}^t \mathbf{n}(0) = \mathbf{P} \mathbf{D}^t \mathbf{P}^{-1} \mathbf{n}(0)$, verkrijgen we:

$$\mathbf{n}(t) \approx \lambda_1^t \mathbf{P} \begin{bmatrix} 1 & 0 & \dots & 0 \\ 0 & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 0 \end{bmatrix} \mathbf{P}^{-1} \mathbf{n}(0) = \lambda_1^t \mathbf{v}_1 \mathbf{u}_1^T \mathbf{n}(0) \quad (23)$$

waarbij \mathbf{v}_1 en \mathbf{u}_1 de genormaliseerde rechter- en linkereigenvectoren zijn die overeenkomen met λ_1 . Dit geeft aan dat de populatie exponentieel groeit met een snelheid λ_1 en convergeert naar een stabiele leeftijdsverdeling gegeven door \mathbf{v}_1 .

De dominante eigenwaarde λ_1 bepaalt of de populatie groeit, stabiel blijft of afneemt: $\lambda_1 > 1$ duidt op groei, $\lambda_1 = 1$ duidt op stabiliteit, en $\lambda_1 < 1$ duidt op afname. De genormaliseerde dominante eigenvector \mathbf{v}_1 geeft de stabiele leeftijdsverdeling, deze is essentieel voor het begrijpen van de langetermijnstructuur van de populatie. Als gevolg van de toepassing van 2.17(3) in (24) kunnen we nu de sterke ergodische stelling voor de populatieprojectie vergelijking geven, die stelt dat het langetermijngedrag van de populatie wordt beschreven door λ_1 en \mathbf{v}_1 [3].

Voor $t \rightarrow \infty$ geldt de relatie

$$\lim_{t \rightarrow \infty} \frac{\mathbf{n}(t)}{\lambda_1^t} = c_1 \mathbf{v}_1 \quad (24)$$

waarbij

$$c_1 = \mathbf{u}_1^T \mathbf{n}(0) \quad (25)$$

De constante c_1 is dus een scalaire waarde die de projectie van de initiële toestand $\mathbf{n}(0)$ op de bijbehorende linker-eigenvector \mathbf{u}_1 van de grootste eigenwaarde λ_1 weergeeft. c_1 geeft dus weer hoe sterk de initiële toestand $\mathbf{n}(0)$ geprojecteerd is op de richting van de rechter-eigenvector \mathbf{v}_1 . Met andere woorden, het geeft aan hoeveel $\mathbf{n}(0)$ in de richting van \mathbf{v}_1 ligt. Voor $t \rightarrow \infty$, zal $\mathbf{n}(t)$ dus voornamelijk bepaald worden door de grootste eigenwaarde λ_1 en de bijbehorende eigenvector \mathbf{v}_1 . De term $\lambda_1^t \mathbf{v}_1$ zal dominant zijn in de uitdrukking voor $\mathbf{n}(t)$, en c_1 geeft de grootte van deze dominante term aan.

Daarom kan het langetermijngedrag van een populatie gemodelleerd door een projectiematrix worden benaderd met behulp van de dominante eigenwaarde en de bijbehorende eigenvectoren. Het begrijpen van de dominante eigenwaarde en de bijbehorende eigenvectoren maakt het mogelijk om de stabiele leeftijdsverdeling en de algemene groeitrend van de populatie in de loop van de tijd te voorspellen.

3 Vertakkingsprocessen

Voordat we het Leslie model omzetten naar een stochastisch Leslie model met behulp de theorie van vertakkingsprocesses, is het nodig om kennis te hebben van vertakkingsprocessen.

Deze processen werden voor het eerst geïntroduceerd aan het eind van de 19e eeuw door Francis Galton en Henry H.W. Watson. Zij waren pioniers in de ontwikkeling van deze theorie. Ze gebruikte deze theorie om het evolutionaire traject en het uitsterven van bepaalde soorten te beschrijven in de tijd. Interessant genoeg was Francis Galton, bekend om zijn bijdragen aan de biologie, ook de neef van Charles Darwin, die onder andere zeer bekend is vanwege zijn werk "On the Origin of Species".

We beginnen ons begrip van vertakkingsprocessen met het eenvoudigste model van vertakkingsprocessen, bekend als het Galton-Watson vertakkingsproces. Dit proces beschrijft de reproductie van individuen binnen een populatie in de tijd. Het kan bijvoorbeeld worden toegepast op voortplantende dieren, een besmettelijk virus of delende kankercellen. We zullen nu een definitie geven van het Galton-Watson vertakkingsproces [4], [2], [6].

3.1 Galton-Watson vertakkingsprocessen

Definitie 3.1. Een Galton-Watson vertakkingsproces is een stochastisch proces dat begint met een enkel individu op tijdstip $n = 0$. Elk individu in het proces leeft precies één tijds-eenheid en produceert dan een willekeurig aantal nakomelingen, aangeduid met Y , volgens een kansverdeling $\mathbb{P}(Y = k) = p_k$ voor $k = 0, 1, 2, \dots$, waarbij $(p_k)_{k \geq 0}$ de **nakomelingenverdeling** wordt genoemd met de eigenschap dat $\sum_{k=0}^{\infty} p_k = 1$. Alle individuen reproduceren onafhankelijk, en elke nakomeling is van hetzelfde type als de ouder. Individuen $1, 2, \dots, n$ hebben families van grootte Y_1, Y_2, \dots, Y_n , waarbij elke Y_i dezelfde verdeling heeft als Y . Laat Z_n het aantal individuen zijn dat geboren wordt op tijdstip n , voor $n = 0, 1, 2, \dots$, wat wordt geïnterpreteerd als de grootte van generatie n . Dan wordt het vertakkingsproces gedefinieerd door de reeks $\{Z_n : n \in \mathbb{N}\}$ [4], [2].

Voorbeeld 3.2. Laat Z_n het aantal cellen op tijdstip n zijn en laat $p_k \in \{p_0 = \frac{1}{6}, p_1 = \frac{2}{3}, p_2 = \frac{1}{6}\}$. Dit betekent dat we op tijdstip $n = 0$ één cel hebben, dus $Z_0 = 1$, en de kans dat deze cel geen nakomelingen heeft ($Y = 0$) is $\frac{1}{6}$, de kans op één nakomeling ($Y = 1$) is $\frac{2}{3}$, en de kans dat deze cel twee nakomelingen heeft ($Y = 2$) is $\frac{1}{6}$.

3.1.1 kansgenererende functie vertakkingsproces

Bij het onderzoeken van de definitie van een vertakkingsproces, kan men de complexiteit opmerken die voortkomt uit de vele verschillende nakomelingskansen die moeten worden bewaard. Daarom ontstaat er een noodzaak voor een eenduidige representatie die al deze verschillende kansen binnen een enkele entiteit omvat. Dit wordt bereikt door het gebruik van wat bekend staat als de kansgenererende functie. Deze functie dient als een methode om een oneindige hoeveelheid gegevens samen te persen tot één uitdrukking, wat een beter beheersbare en beknoptere analyse van het vertakkingsproces mogelijk maakt. De definitie van de kansgenererende functie is als volgt:

Definitie 3.3. Laat X een discrete willekeurige variabele zijn die waarden aanneemt in de niet-negatieve gehele getallen $\{0, 1, 2, \dots\}$. De kansgenererende functie (PGF) van X wordt gedefinieerd door

$$G_X(s) = E[s^X] = \sum_{x=0}^{\infty} s^x \mathbb{P}(X = x) \quad (26)$$

voor alle $s \in \mathbb{R}$ waarvoor de som convergeert [4].

Specifiek omvat het domein van $G_X(s)$, aangeduid met $D = \{s \in \mathbb{R} : \sum_{x=0}^{\infty} |s|^x \mathbb{P}(X = x) < \infty\}$, alle waarden van s waarvoor de reeks absoluut convergeert.

Men ziet gemakkelijk dat $[-1, 1] \subseteq D$, met andere woorden, als $|s| \leq 1$, volgt dat $\sum_{x=0}^{\infty} |s|^x \mathbb{P}(X = x) \leq \sum_{x=0}^{\infty} \mathbb{P}(X = x) = 1$, aangezien de kansen $\mathbb{P}(X = x)$ optellen tot 1 (als eigenschap van kansverdelingen) en $|s|^x \leq 1$ voor elke x . Dit garandeert de convergentie van $G_X(s)$ binnen het interval $[-1, 1]$ en benadrukt de absolute convergentie van de PGF in dit bereik, wat bevestigt dat D inderdaad ten minste het interval $[-1, 1]$ omvat.

Het kennen van de kansgenererende functie (PGF) maakt de directe berekening van verschillende eigenschappen van de willekeurige variabele X mogelijk. Dit omvat de afleiding van kansmassafuncties, waarbij de kans $\mathbb{P}(X = n)$ dat X een specifieke waarde n aanneemt, wordt gegeven door $\frac{1}{n!} G_X^{(n)}(0)$. Bovendien vergemakkelijkt de PGF de berekening van momenten van X . Bijvoorbeeld, de verwachte waarde $\mathbb{E}(X)$ wordt bepaald door $G'_X(1)$ te evalueren, terwijl hogere momenten zoals $\mathbb{E}\{X(X-1)\dots(X-k+1)\}$ kunnen worden gevonden met behulp van $G_X^{(k)}(1)$. Aangezien de kansgenererende functie uniek is, volgt hieruit dat twee willekeurige variabelen die dezelfde PGF hebben, ook dezelfde verdeling hebben. Met andere woorden, de PGF van een willekeurige variabele X vertelt ons alles wat er te weten valt over de verdeling van X . Een andere opmerkelijke eigenschap van de kansgenererende functie is het vermogen om een som om te zetten in een product [4].

Stelling 3.4. *Stel dat X_1, \dots, X_n onafhankelijke willekeurige variabelen zijn, en laat $Y = X_1 + \dots + X_n$ zijn. Dan geldt*

$$G_Y(s) = \prod_{i=1}^n G_{X_i}(s)$$

[4]

Aangezien de theorie van vertakkingsprocessen draait om onafhankelijke willekeurige variabelen, kan men al raden hoe belangrijk de kansgenererende functie is voor de ontwikkeling van de theorie in deze scriptie.

Veel van de fascinerende eigenschappen van het vertakkingsproces draaien om de verdeling van Z_n , wat de populatiegrootte op tijdstip n vertegenwoordigt. De hoeveelheid Z_n , die het aantal individuen op tijdstip n aangeeft, is gelijk aan het totale aantal nakomelingen geproduceerd door individuen $1, 2, \dots, Z_{n-1}$. Met andere woorden, Z_n kan worden uitgedrukt als de som van de individuele aantallen nakomelingen: $Z_n = \sum_{i=1}^{Z_{n-1}} Y_i$. Aangezien

Z_n een willekeurig gestopte som is vanwege het variabele aantal termen, kan volgens Stelling 4.6 in [4], de kansgenererende functie (PGF) van Z_n worden gegeven door de recursieve formule [4]:

$$G_{Z_n}(s) = G_{Z_{n-1}}(G_Y(s)),$$

waarbij $G_{Z_{n-1}}$ de PGF van de willekeurige variabele Z_{n-1} vertegenwoordigt.

Voor de eenvoud in de tekst zullen we de volgende notatie gebruiken:

$$G_{Z_n}(s) = G_n(s), \quad G_{Z_{n-1}}(s) = G_{n-1}(s),$$

enzovoort.

Het is belangrijk te benadrukken dat Z_1 correspondeert met Y (het aantal individuen geboren op tijdstip $n = 1$), wat vereenvoudigt tot: $G_Y(s) = G_1(s) = G(s)$.

Nu zijn we klaar om de recursieformule voor vertakkingsprocessen te presenteren zoals vermeld in de volgende stelling [4]:

Stelling 3.5. *Laat $G(s) = \mathbb{E}(s^Y) = \sum_{y=0}^{\infty} p_y s^y$ de PGF zijn van de familiegrootteverdeling, Y . Laat $Z_0 = 1$ (starten met een enkel individu op tijdstip 0), en laat Z_n de populatiegrootte op tijdstip n zijn ($n = 0, 1, 2, \dots$). Laat $G_n(s)$ de PGF zijn van de willekeurige variabele Z_n . Dan geldt,*

$$G_n(s) = G_{n-1}(G(s)) = \underbrace{G(G(G(\dots G(s) \dots)))}_{n \text{ keer}} = G(G_{n-1}(s)). \quad (27)$$

[4]

Het begrijpen van de verwachte populatiegrootte, $E(Z_n)$, over n generaties is cruciaal voor ons begrip van vertakkingsprocessen. Even belangrijk is het begrijpen van de betekenis van $\text{Var}(Z_n)$. Z_n volgt typisch een "boom-or-bust" verdeling, waarbij de populatie ofwel snelle groei (boom) ervaart of volledig instort (bust). Als de populatie faalt, gebeurt dit vaak snel binnen de eerste generaties. Dit benadrukt het belang van $\text{Var}(Z_n)$: een grote variantie duidt op significante afwijkingen van de gemiddelde waarden, wat suggereert dat alleen vertrouwen op het gemiddelde mogelijk niet voldoende is om het gedrag van het proces te voorspellen.

Om $E(Z_n)$ en $\text{Var}(Z_n)$ te berekenen, moeten we deze uitdrukken in termen van het gemiddelde en de variantie van de familiegrootteverdeling, Y . Laat $E(Y) = \mu$ en $\text{Var}(Y) = \sigma^2$ zijn, wat het gemiddelde en de variantie vertegenwoordigt van het aantal nakomelingen van een enkel individu [4].

Stelling 3.6. *Laat $\{Z_0, Z_1, Z_2, \dots\}$ een vertakkingsproces zijn met $Z_0 = 1$ (starten met een enkel individu). Laat Y de familiegrootteverdeling aanduiden, en veronderstel dat $\mathbb{E}(Y) = \mu$. Dan geldt*

$$\mathbb{E}(Z_n) = \mu^n$$

Stelling 3.7. *Laat $\{Z_0, Z_1, Z_2, \dots\}$ een vertakkingsproces zijn met $Z_0 = 1$ (starten met een enkel individu). Laat Y de familie-grootteverdeling aanduiden, en veronderstel dat $\mathbb{E}(Y) = \mu$ en $\text{Var}(Y) = \sigma^2$. Dan geldt*

$$\text{Var}(Z_n) = \begin{cases} \sigma^2 n & \text{als } \mu = 1, \\ \sigma^2 \mu^{n-1} \left(\frac{1-\mu^n}{1-\mu} \right) & \text{als } \mu \neq 1 \quad (> 1 \text{ of } < 1). \end{cases}$$

Ondanks de mogelijkheid van significante afwijkingen van de werkelijke resultaten, blijft de verwachte waarde van Z_n belangrijk in de studie van limietstellingen voor vertakkingsprocessen. Naarmate n naar oneindig nadert, wordt het begrijpen van het gedrag van de verwachte waarde cruciaal voor het afleiden van asymptotische resultaten, inclusief de kritische voorwaarden voor het voortbestaan van de populatie, zoals we in de volgende paragraaf zullen verkennen.

3.2 Uitsterving in Vertakkingsprocessen

Voortbouwend op de discussies uit het vorige paragrafen begrijpen we dat de verwachte waarde en variantie van Z_n essentieel zijn voor het begrijpen van vertakkingsprocessen. Deze statistische maatstaven zijn bijzonder belangrijk bij het begrijpen van het limietgedrag van de processen en bij het berekenen van hun uitstervingskansen, een fascinerende en praktische toepassing binnen dit vakgebied. De uitstervingskans en uiteindelijke uitstervingskans, respectievelijk aangeduid met $\gamma_n = \mathbb{P}(E_n) = \mathbb{P}(\text{uitgestorven bij generatie } n) = \mathbb{P}(Z_n = 0)$ en $\mathbb{P}(\text{uiteindelijke uitsterving}) = \mathbb{P}(\lim_{n \rightarrow \infty} E_n) = \lim_{n \rightarrow \infty} \mathbb{P}(E_n)$, spelen een sleutelrol in onze analyse.

Verder kunnen vertakkingsprocessen worden geclassificeerd op basis van de criticaliteitsparameter μ , die het gemiddelde van de nakomelingenverdeling Y is. Deze indeling in drie typen is cruciaal voor het begrijpen van de dynamiek en potentiële uitkomsten van de processen en het voorspellen en beheren van populatiegedrag in verschillende domeinen.

Het belang van dit hoofdstuk ligt in het onderzoeken van de basisprincipes voor het berekenen van uitstervingskansen in vertakkingsprocessen, wat een cruciaal inzicht biedt om verschillende redenen. Het begrijpen van de uitstervingskansen van populaties, inclusief die van schadelijke soorten of cellen (bijv. kankercellen), kan risicobeoordeling en beheerstrategieën in de landbouw, gezondheidszorg en milieubeheer informeren [4], [6], [10]. De uitstervingskans geeft ons veel informatie over het langetermijngedrag en de eindtoestand van het proces in kwestie.

3.2.1 Classificatie van de Verwachte Waarde

Als $Z_n = 0$, betekent dit dat de populatie is uitgestorven, wat inhoudt dat $Z_t = 0$ voor alle $t \geq n$. Definieer de gebeurtenis E_n als $E_n = \{Z_n = 0\}$, wat het uitsterven van de populatie tegen de n -de generatie vertegenwoordigt.

Zoals besproken aan het einde van het vorige hoofdstuk, zijn de verwachte waarde en variantie van Z_n cruciaal voor het begrijpen van vertakkingsprocessen. Deze maatstaven zijn bijzonder belangrijk bij het onderzoeken van het limietgedrag en het berekenen van uitstervingskansen in vertakkingsprocessen. Het bepalen van de uitstervingskans is een significante en intrigerende toepassing binnen dit vakgebied.

Laat $\gamma_n = \mathbb{P}(E_n) = \mathbb{P}(\text{uitgestorven bij generatie } n) = \mathbb{P}(Z_n = 0)$.

De volgende classificatie is belangrijk [4], [10]: De criticaliteitsparameter μ , die het gemiddelde aantal nakomelingen in een vertakkingsproces aangeeft, categoriseert het proces in drie typen: superkritisch, kritiek en subkritisch. Volgens stelling 2.8 is de verwachte waarde van de populatiegrootte $Z_n \mu^n$. Daarom, wanneer $\mu > 1$, wordt verwacht dat de populatie exponentieel groeit, wat wijst op een superkritische toestand waarbij $\mathbb{E}[Z_n]$ naar oneindig neigt. Als $\mu = 1$, is het proces in een kritieke toestand, waarbij de verwachte waarde constant 1 blijft. Omgekeerd, voor $\mu < 1$, neemt de verwachte populatiegrootte in de tijd af, wat een subkritische toestand kenmerkt waarbij $\mathbb{E}[Z_n]$ naar nul neigt.

In een subkritisch proces ($\mu < 1$) is uitsterven zeker, met $\gamma = 1$, wat betekent dat de verwachte populatiegrootte $\mathbb{E}(Z_n) = \mu^n$ naar nul neigt naarmate n naar oneindig nadert.

Voor een kritisch proces ($\mu = 1$) is uitsterven ook onvermijdelijk, tenzij elke generatie consequent precies één nakomeling produceert. Hier is $\mathbb{E}(Z_n) = \mu^n = 1$ voor alle n , ondanks de zekerheid van uitsterven. In het geval van een superkritisch proces ($\mu > 1$) is uitsterven niet gegarandeerd, met $\gamma < 1$. Hier groeit de verwachte populatiegrootte $\mathbb{E}(Z_n) = \mu^n$ onbegrensd naarmate n toeneemt, wat wijst op de mogelijkheid van onbeperkte groei.

Dit brengt ons bij de volgende stelling betreffende de uitstervingskans:

Stelling 3.8 (Uiteindelijke Uitsterving). *Voor een subkritisch proces ($\mu < 1$) is de kans op uiteindelijke uitsterving $\gamma = 1$, wat uitsterving verzekert. Voor een kritisch proces ($\mu = 1$) is uitsterving ook zeker ($\gamma = 1$), tenzij $Z_n = 1$ is voor elke generatie n . Voor een superkritisch proces ($\mu > 1$) ligt de kans op uiteindelijke uitsterving tussen 0 en 1 ($0 < \gamma < 1$)*

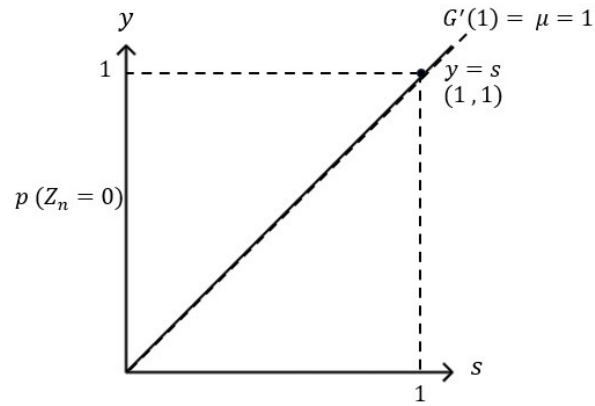
Bewijs. [4] We zullen de grafieken (op de volgende pagina) analyseren die de krommen van $G(s)$ weergeven voor verschillende waarden van de criticaliteitsparameter μ . In elk scenario voldoet de grafiek van $G(s)$ aan de volgende voorwaarden: $G(s)$ is stijgend en strikt convex wanneer $Z_n \geq 2$; $G(0) = \mathbb{P}(Z_n = 0) \geq 0$; $G(1) = 1$; $G'(1) = \mu$, wat betekent dat de helling van $G(s)$ bij $s = 1$ gelijk is aan μ ; en de uitstervingskans, γ , is de kleinste waarde ≥ 0 waarvoor $G(s) = s$.

Beschouw eerst het kritieke geval waar $\mu = 1$. In dit scenario valt de kromme $G(s)$ samen met de lijn $y = s$ bij $s = 1$. Daarom is de uitstervingskans γ gelijk aan 1, aangezien de kromme de lijn $y = s$ op geen enkel ander punt binnen het interval $[0, 1)$ snijdt. Een uitzondering doet zich voor als $Z_n = 1$ voor alle $n \in \mathbb{N}_0$, waar $G(s) = s$ voor alle $s \in [0, 1]$. In dit unieke geval is de kleinste waarde van s waar $G(s) = s$ gelijk aan 0, wat betekent dat de uitstervingskans γ gelijk is aan 0.

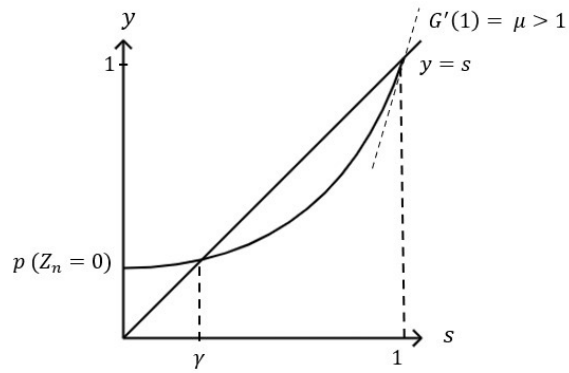
Vervolgens beschouwen we het superkritieke geval waar $\mu > 1$. Hier is de helling van $G(s)$ bij $s = 1$ steiler dan die van de lijn $y = s$. Hierdoor ligt de kromme $G(s)$ onder de lijn $y = s$ voorbij $s = 1$. Dit betekent dat de kromme de lijn $y = s$ op twee punten moet snijden: $s = 1$ en een ander punt s_1 waar $0 < s_1 < 1$. Bijgevolg is de uitstervingskans γ een positieve waarde kleiner dan 1.

Ten slotte beschouwen we het subkritieke geval waar $\mu < 1$. In deze situatie ligt de kromme $G(s)$ boven de lijn $y = s$ voor alle $s < 1$. Binnen het interval $[0, 1)$ is er dus geen snijpunt tussen de kromme en de lijn $y = s$ voordat de kromme de y -as bereikt. Dit toont aan dat de uiteindelijke uitstervingskans γ gelijk is aan 1, aangezien dit de kleinste waarde van $s > 0$ is waarvoor $G(s) = s$.

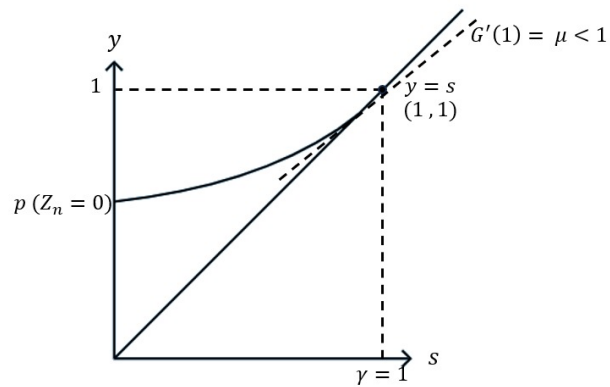
Ten slotte laten deze grafieken zien dat naarmate de criticaliteitsparameter μ afneemt, de waarschijnlijkheid $\mathbb{P}(Z_n = 0)$ toeneemt, wat de gestelde stelling bevestigt. \square



Figuur 2: Kritieke geval $\mu = 1$



Figuur 3: Superkritieke geval $\mu > 1$



Figuur 4: Subkritieke geval $\mu < 1$

4 Van Deterministische naar Stochastisch

We hebben nu het vertakkingsproces besproken, wat een stochastische model is. Ook hebben we het Leslie-model onderzocht, dat een deterministisch model is. In deze sectie zullen we deze twee benaderingen combineren om een meer omvattend model te bewerkstelligen. Specifiek zullen we de stochastische elementen van vertakkingsprocessen integreren met de Leslie-matrix. Deze integratie stelt ons in staat om de onvoorspelbaarheid in natuurlijke populaties vast te leggen, wat ons een realistischer beeld geeft van populatiedynamica.

4.1 Stochastische Simulatie

Om demografische stochastiek in het deterministische Leslie-model te integreren, moeten we de kansverdeling bepalen van het aantal en de typen individuen die op tijd $t + 1$ worden geproduceerd door een individu dat aanwezig is op tijd t . Een individu op tijd t draagt bij aan de populatie op tijd $t+1$ door overleving en reproductie (vruchtbaarheid). We kunnen de theorie van vertakkingsprocessen toepassen op een Leslie-matrix om dit te bereiken. Omdat de theorie van vertakkingsprocessen niet gelijk toepasbaar is op een Leslie matrix, gebruiken we de ontleding $\mathbf{A} = \mathbf{S} + \mathbf{F}$ van de Leslie matrix uit hoofdstuk 2. de vruchtbaarheidsmatrix kan dan worden gebruikt om de voortplanting van een populatie te simuleren als een vertakkingsproces zoals in definitie 3.1.

De vruchtbaarheidsmatrix \mathbf{F} vertegenwoordigt verwachte geboorten per individu in elke leeftijdsklasse. Om deze verwachtingen om te zetten in kansen, gebruiken we de Poisson-verdeling.

De Poisson-verdeling wordt vaak gebruikt voor het simuleren van leeftijdsspecifieke geboorten binnen een Leslie-matrix vanwege zijn eigenschappen die goed aansluiten bij geboorten in demografische studies. Elke leeftijdsklasse in de matrix heeft een specifieke geboortecijfer f_i .

De belangrijkste reden voor het gebruik van deze verdeling ligt in zijn vermogen om zeldzame, onafhankelijke gebeurtenissen over een vast interval na te bootsen, wat in de context van een Leslie-matrix verwijst naar de geboorten die plaatsvinden binnen elke leeftijdsklasse gedurende een bepaalde periode.

In grote populaties kan het totale aantal geboorten hoog zijn, maar de kans dat een individu binnen een korte periode bevalt, is relatief laag. Dit past bij de veronderstelling van de Poisson-verdeling dat gebeurtenissen (geboorten) zeldzaam zijn gezien het aantal mogelijkheden (individuen). De Poisson-verdeling vereenvoudigt het berekenen van de kans op een specifiek aantal geboorten in elke leeftijdsklasse door gebruik te maken van een enkele parameter, de snelheid (μ), die het verwachte aantal geboorten per tijdseenheid in die leeftijdsklasse vertegenwoordigt.

Definitie 4.1. Een discrete willekeurige variabele X volgt een *Poisson-verdeling* met parameter μ , waarbij $\mu > 0$, als de kansmassafunctie p gegeven wordt door

$$p(k) = \mathbb{P}(X = k) = \frac{\mu^k e^{-\mu}}{k!} \quad \text{voor } k = 0, 1, 2, \dots$$

Deze verdeling wordt aangeduid als $\text{Pois}(\mu)$.

Dit impliceert dat het aantal geboorten van een individu in leeftijdsklasse i de volgende kansverdeling volgt:

$$\mathbb{P}_X^{(i)}(X = k) = \frac{(f_i)^k e^{-f_i}}{k!} \quad \text{voor } k = 0, 1, 2, \dots \text{ met } \mathbb{P}_X^{(i)} = \mathbb{P}(X \mid \text{leeftijdsgroep } i) \quad (28)$$

We kunnen de vruchtbaarheidsmatrix \mathbf{F} nu geven door: :

$$\mathbf{F} = \begin{pmatrix} \mathbb{P}_X^{(1)} & \mathbb{P}_X^{(2)} & \dots & \mathbb{P}_X^{(m)} \\ 0 & 0 & \dots & 0 \\ 0 & 0 & \dots & 0 \\ 0 & 0 & \dots & 0 \\ 0 & 0 & \dots & 0 \end{pmatrix}$$

Met $\mathbb{P}_X^{(i)}$ een Poissonverdeling met parameter f_i , het gemiddelde aantal geboorten voor leeftijdsklasse i .

Voordat we het overlevingsproces stochastisch beschrijven, geven we eerst de volgende defniete:

Definitie 4.2. Een discrete willekeurige variabele X wordt gezegd een *binomiale verdeling* te hebben met parameters n en p , waarbij n een niet-negatief geheel getal is en $0 \leq p \leq 1$, als de kansmassafunctie p gedefinieerd is als

$$p(k) = \mathbb{P}(X = k) = \binom{n}{k} p^k (1-p)^{n-k} \quad \text{voor } k = 0, 1, 2, \dots, n$$

Deze verdeling wordt weergegeven als $\text{Binom}(n, p)$.

Voor de overlevingsmatrix \mathbf{S} , die de verwachte overlevingscijfers voor individuen in de verschillende leeftijdsklassen bevat, gebruiken we de binomiale verdeling om deze om te zetten in kansen. Het aantal individuen dat naar de volgende leeftijdsklasse gaat (overleving) wordt beschreven door de matrix \mathbf{S} . De overgang van leeftijdsklasse i naar leeftijdsklasse $i+1$ per individu wordt dan beschreven door een Bernoulli-verdeling te gebruiken. Omdat alle individuen in een leeftijdsgroep dezelfde verdeling volgen, kan dit voor een hele leeftijdsgroep beschreven worden met een binomiale verdeling met parameter s_i (overlevingskans) en n_i (aantal individuen in de desbetreffende leeftijdsklasse) [3], [14]. De overlevingsmatrix \mathbf{S} met stochastische elementen op de sub-diagonaal wordt dan gegeven door:

$$\mathbf{S} = \begin{pmatrix} 0 & 0 & \dots & 0 \\ S_1 & 0 & \dots & 0 \\ 0 & S_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & S_{m-1} \end{pmatrix}$$

En de kansverdeling voor het aantal overlevenden in leeftijdsklasse i wordt gegeven door:

$$\mathbb{P}(S_i = k) = \binom{n_i}{k} (s_i)^k (1 - s_i)^{n_i - k} \quad \text{voor } k = 0, 1, 2, \dots, n_i \quad (29)$$

met S_i het aantal individuen in leeftijdsklasse i die overleven.

We hebben gezien dat de projectievergelijking voor de deterministische Leslie matrix voor k tijdstappen wordt gegeven door:

$$\mathbf{n}(t + k) = (\mathbf{S} + \mathbf{F})^k \cdot \mathbf{n}(t)$$

Zij nu $\mathbf{A}_j = \mathbf{S}_j + \mathbf{F}_j$, met \mathbf{S}_j de overlevingsmatrix en \mathbf{F}_j de vruchtbaarheidsmatrix op tijdstip j . Dan wordt nu voor het stochastische model de projectievergelijking als volgt gegeven [3], [5]:

$$\mathbf{n}(t + k) = (\mathbf{S}_{t+k} + \mathbf{F}_{t+k})(\mathbf{S}_{t+k-1} + \mathbf{F}_{t+k-1}) \cdots (\mathbf{S}_{t+1} + \mathbf{F}_{t+1}) \cdot \mathbf{n}(t)$$

omdat elke matrix nu willekeurig is, moet op elke tijdstap een nieuwe matrix bepaald worden, de projectie vergelijking kan dus niet meer gegeven worden door slechts een zelfde leslie matrix tot een macht te verheffen. Bijgevolg is dat de samenstelling van de matrix in het stochastische model onvoorspelbaar is, het kennen van de samenstelling van de matrix op een bepaald moment geeft dus geen informatie over de samenstelling op een ander tijdstip. De bepaling van $\mathbf{n}(t + k)$ kan dus niet direct gedaan worden zoals in het deterministische geval. Voor elke tijdstap tot aan tijdstip $t + k$ moest dus een willekeurige matrix bepaald worden. Dit impliceert dus ook dat het gebruik van de Perron-Frobenius Stelling niet mogelijk is in dit geval. Per tijdstap geeft de willekeurige matrix $\mathbf{A}_j = \mathbf{S}_j + \mathbf{F}_j$ dus de verdeling van de overlevingen en van de geboortes. De nieuw geboren individuen worden dan per tijds-eenheid gegenereerd (volgens de Poisson verdeling) en de overlevenden individuen van de vorige tijdstap (volgens de Binomiale verdeling). De totale populatie wordt dan per tijdstap bepaald en is de som van de nieuw geboren individuen en de overlevenden individuen van de vorige tijdstap. de totale populatie op tijdstip $\mathbf{n}(t + k)$ wordt dan gesimuleerd als een vertakkingsproces met overleving en de nieuw geboren individuen volgens een Galton-Watson vertakkingsproces zoals in 3.1. De code voor deze simulatie word gegeven in Bijlage A.2 en al worden besproken in de volgende sectie (4.3)

4.2 Bespreking van de codes

In deze paragraaf bespreken we de belangrijkste functies van de Python-code die hoort bij de stochastische simulatie uit de voorgaande paragraaf en de deterministische simulatie uit Bijlage A.1. De code voor de stochastische simulatie is ontwikkeld op basis van de code in Bijlage A.1 [13]. Door de ideeën uit paragraaf 4.1 te communiceren met ChatGPT en via trial-and-error te verfijnen, is de uiteindelijke code in Bijlage A.2 verkregen.

4.2.1 Deterministische code Bijlage A.1

De functie `ProjecteerPopulatie` projecteert de populatie over een opgegeven aantal tijdstappen door gebruik te maken van de Leslie-matrix.

```
1 def ProjecteerPopulatie(LeslieMatrix, BeginPopulatie, AantalTijdstappen):
2     AlleTijdstappen = np.zeros((LeslieMatrix.shape[0], AantalTijdstappen +
3     1))
4     AlleTijdstappen[:, 0] = BeginPopulatie
5     for t in range(1, AantalTijdstappen + 1):
6         AlleTijdstappen[:, t] = np.dot(LeslieMatrix, AlleTijdstappen[:, t
7         - 1])
8     return AlleTijdstappen
```

Listing 1: De functie `ProjecteerPopulatie`

- `AlleTijdstappen = np.zeros((LeslieMatrix.shape[0], AantalTijdstappen + 1))`: Initialiseert een matrix `AlleTijdstappen` om de populatiegroottes voor alle tijdstappen op te slaan.
- `AlleTijdstappen[:, 0] = BeginPopulatie`: De beginpopulatie wordt in de eerste kolom van `AlleTijdstappen` geplaatst.
- `for t in range(1, AantalTijdstappen + 1)`: Itereert over elke tijdstap.
- `AlleTijdstappen[:, t] = np.dot(LeslieMatrix, AlleTijdstappen[:, t - 1])`: Voor elke tijdstap wordt de nieuwe populatiegrootte berekend door de Leslie-matrix te vermenigvuldigen met de populatiegrootte van de vorige tijdstap.
- `return AlleTijdstappen`: De functie retourneert de matrix `AlleTijdstappen`, die de populatiegroottes over alle tijdstappen bevat.

Vervolgens wordt de dominante eigenwaarde van de Leslie-matrix berekend om de asymptotische groeisnelheid van de populatie te bepalen.

```
1 lambda_value = max(np.linalg.eigvals(LeslieMatrix))
2 print(f'Lambda: {lambda_value.real}')
```

Listing 2: Berekening van de dominante eigenwaarde

- `lambda_value = max(np.linalg.eigvals(LeslieMatrix))`: De eigenwaarden van de Leslie-matrix worden berekend met `np.linalg.eigvals`. De grootste eigenwaarde (`lambda_value`) wordt bepaald, die de asymptotische groeisnelheid van de populatie vertegenwoordigt.
- `print(f'Lambda: lambda_value.real')`: De dominante eigenwaarde wordt vervolgens afgedrukt.

4.2.2 Stochastische code bijlage A.2

De functie `WillekeurigeGeboorten` in bijlage A.2 simuleert het aantal geboorten en overlevingen voor elke tijdstap. Deze functie gebruikt de stochastische distributies (Poisson voor geboorten en binomiaal voor overlevingen) zoals in paragraaf 4.1 besproken is.

```
1 def WillekeurigeGeboorten(PopulatieVector):
2     Nakomelingen = 0
3     for i, Geboortecijfer in enumerate(Geboortecijfers):
4         for _ in range(int(PopulatieVector[i])):
5             Nakomelingen += np.random.poisson(Geboortecijfer)
6     NieuweGeneratie = np.zeros(len(Geboortecijfers))
7     NieuweGeneratie[0] = Nakomelingen
8     for i in range(1, len(Geboortecijfers)):
9         NieuweGeneratie[i] = np.random.binomial(int(PopulatieVector[i-1]), Overlevingskansen[i-1])
10    return NieuweGeneratie
```

Listing 3: De functie `WillekeurigeGeboorten`

- `Nakomelingen = 0`: Initialiseert het aantal nakomelingen.
- `for i, Geboortecijfer in enumerate(Geboortecijfers)`: Itereert over elke leeftijdsgroep en hun respectievelijke geboortecijfers.
- `for _ in range(int(PopulatieVector[i]))`: Voor elk individu in de huidige leeftijdsgroep wordt een willekeurig aantal nakomelingen gegenereerd volgens een Poisson-verdeling.
- `Nakomelingen += np.random.poisson(Geboortecijfer)`: Verhoogt het aantal nakomelingen met een willekeurige waarde volgens de Poisson-verdeling.
- `NieuweGeneratie = np.zeros(len(Geboortecijfers))`: Initialiseert een nieuwe generatievector.
- `NieuweGeneratie[0] = Nakomelingen`: Plaatst het totale aantal nakomelingen in de eerste leeftijdsgroep.
- `for i in range(1, len(Geboortecijfers))`: Itereert over de leeftijdsgroepen om overlevingskansen toe te passen.
- `NieuweGeneratie[i] = np.random.binomial(int(PopulatieVector[i-1]), Overlevingskansen[i-1])`: Berekent het aantal overlevenden voor elke leeftijdsgroep volgens een binomiale verdeling.

De functie `PopulatieOpTijdstip` projecteert dan de populatie voor een gegeven aantal tijdstappen door herhaaldelijk `WillekeurigeGeboorten` aan te roepen.

```

1 def PopulatieOpTijdstip(t):
2     PopulatieVector = BeginPopulatieVector
3     for _ in range(t):
4         PopulatieVector = WillekeurigeGeboorten(PopulatieVector)
5     return PopulatieVector

```

Listing 4: De functie PopulatieOpTijdstip

- `PopulatieVector = BeginPopulatieVector`: Initialiseert de populatievector met de beginpopulatie.
- `for _ in range(t)`: Itereert over het aantal tijdstappen.
- `PopulatieVector = WillekeurigeGeboorten(PopulatieVector)`: Update de populatievector door de functie `WillekeurigeGeboorten` aan te roepen, die het aantal nakomelingen en overlevenden berekent.
- `return PopulatieVector`: Geeft de geprojecteerde populatievector na het opgegeven aantal tijdstappen terug.

4.3 Simulaties

We zullen nu de resultaten van de simulaties voor verschillende populatiegroottes vergelijken voor zowel het deterministische als het stochastische model. Voor de simulaties gebruiken we de volgende 3×3 matrix:

$$A := \begin{bmatrix} f_1 & f_2 & f_3 \\ s_1 & 0 & 0 \\ 0 & s_2 & 0 \end{bmatrix} = \begin{bmatrix} 1.5 & 2.0 & 2.5 \\ 0.8 & 0 & 0 \\ 0 & 0.5 & 0 \end{bmatrix} \quad (30)$$

Door gebruik te maken van de Perron-Frobenius-stelling vinden we dat de dominante eigenwaarde $\lambda_1 \approx 2.35827$ is. Omdat deze eigenwaarde groter is dan 1, verwachten we dat de populatie exponentieel zal groeien (zie vergelijking 24). De stabiele leeftijdsverdeling die hoort bij λ_1 voor deze matrix is gegeven door:

$$\mathbf{v}_1 := \begin{bmatrix} 0.7086 \\ 0.2404 \\ 0.0510 \end{bmatrix}$$

Dit suggereert dat de populatie voornamelijk geconcentreerd zal zijn in leeftijdsklasse 1.

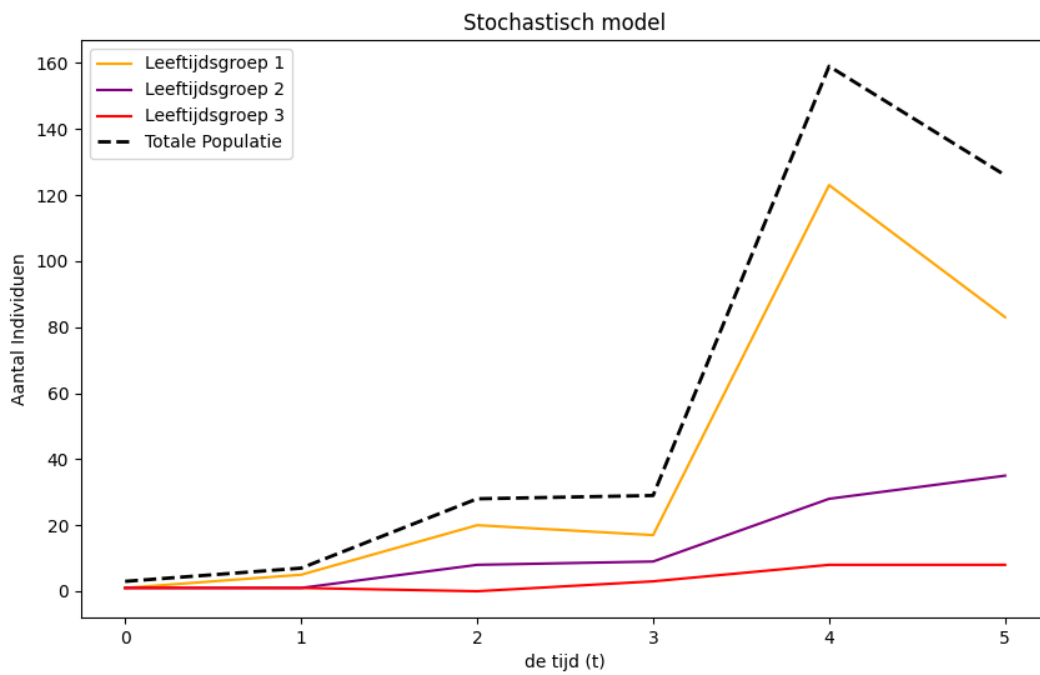
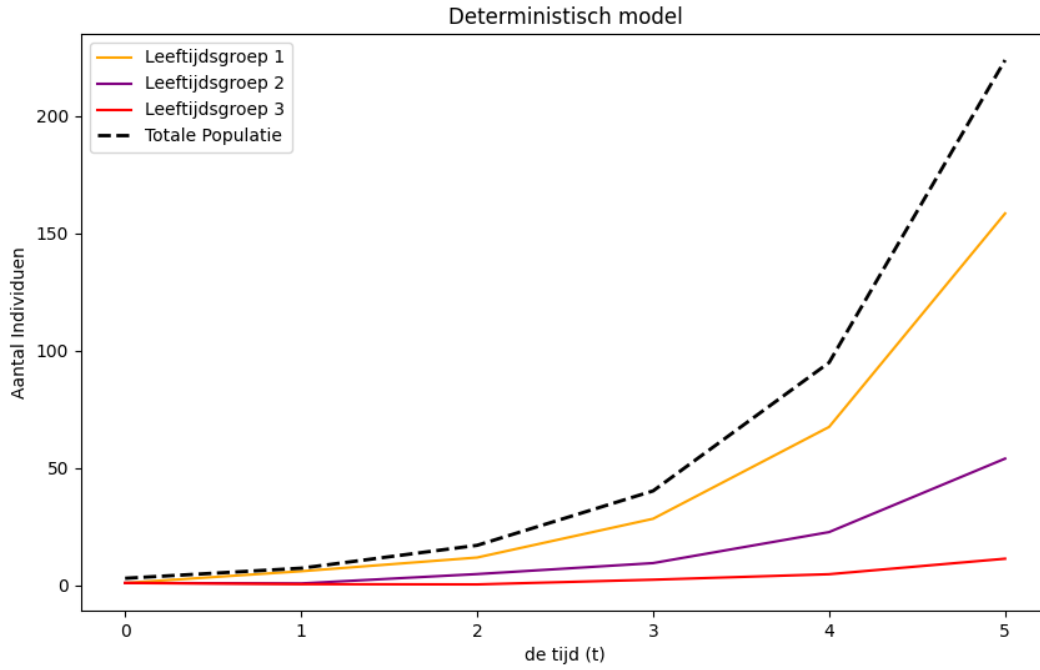
We zullen de simulaties uitvoeren voor drie verschillende initiële populatiegroottes, namelijk:

$$\mathbf{n}(0) := \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}, \quad \mathbf{n}(0) := \begin{bmatrix} 10 \\ 10 \\ 10 \end{bmatrix}, \quad \mathbf{n}(0) := \begin{bmatrix} 1000 \\ 1000 \\ 1000 \end{bmatrix}$$

We vergelijken de resultaten van het deterministische model met die van het stochastische model om inzicht te krijgen in de verschillen tussen de voorspelbare en de willekeurige dynamiek van de populatiegroei.

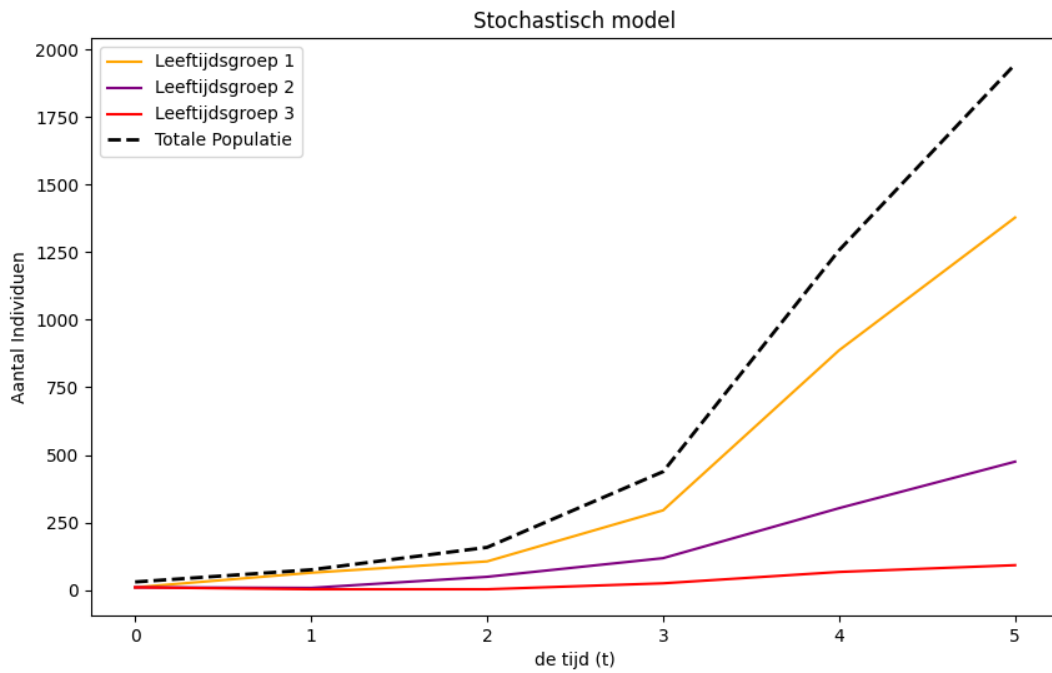
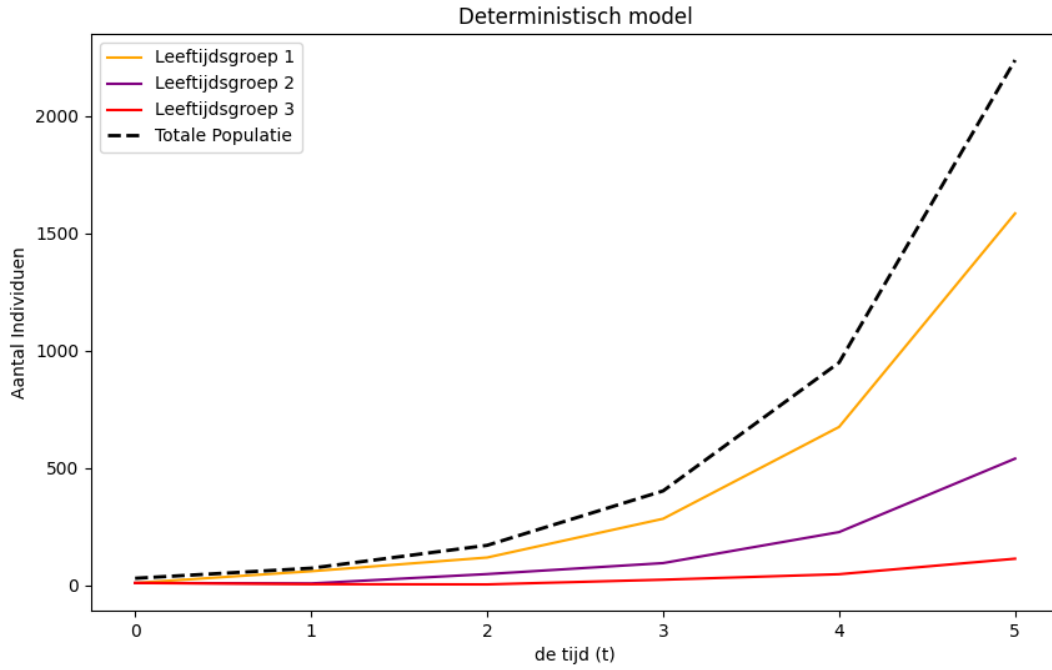
4.3.1

$$\mathbf{n}(0) := [n_1 \ n_2 \ n_3]^T = [1 \ 1 \ 1]^T$$



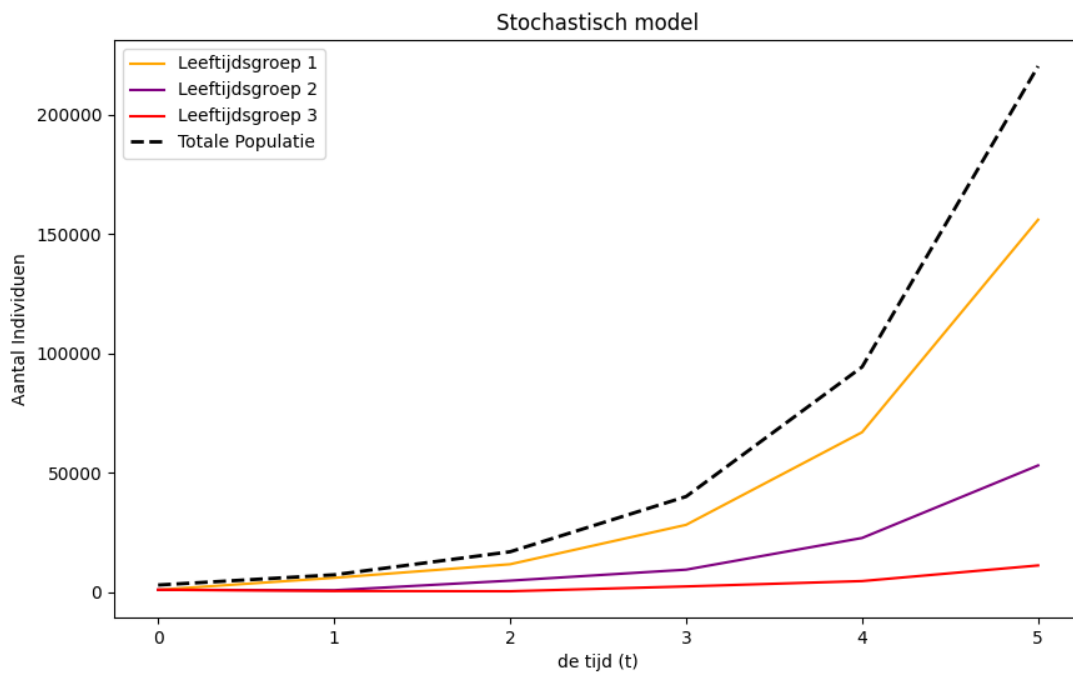
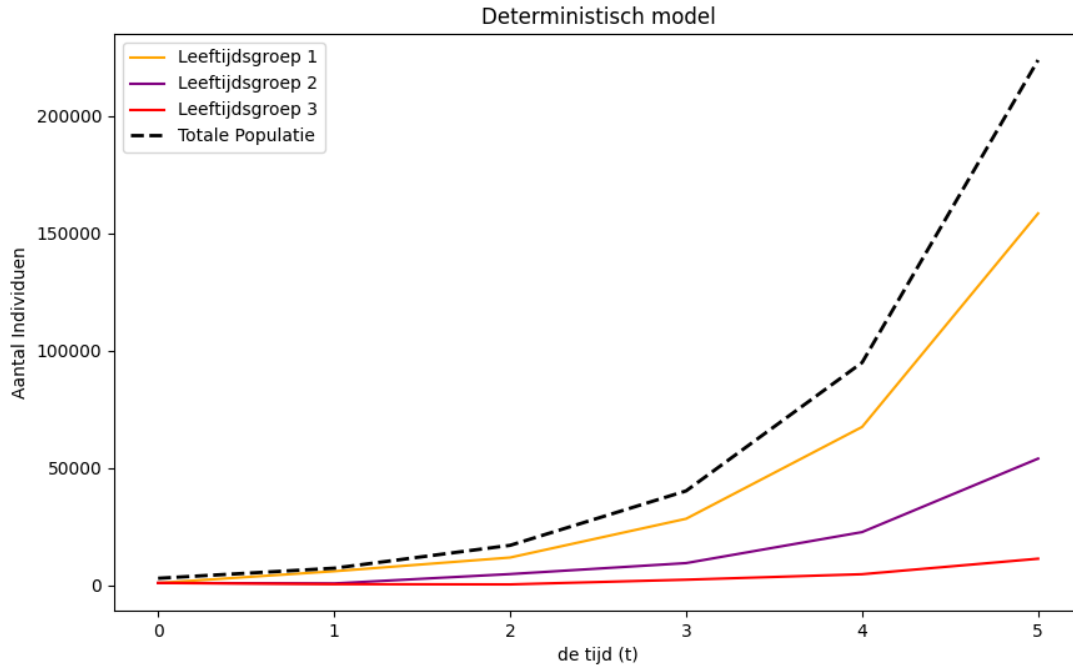
4.3.2

$$\mathbf{n}(0) := [n_1 \ n_2 \ n_3]^T = [10 \ 10 \ 10]^T$$



4.3.3

$$\mathbf{n}(0) := [n_1 \ n_2 \ n_3]^T = [1000 \ 1000 \ 1000]^T$$



4.4 Bevindingen simulaties

Uit de herhaalde simulaties van het stochastische model en de vergelijkingen met de resultaten van het deterministische model voor dezelfde populatiegroottes en tijdsintervallen blijkt dat de populatiegrootte een cruciale rol speelt in het gedrag van het stochastische model. Het stochastische model vertoont bij kleine populaties duidelijk willekeurig gedrag, waarbij de populatiegroei fluctueert en zelfs kan dalen tot nul. Echter, naarmate de populatiegrootte toeneemt, nemen de willekeurige fluctuaties af. Voor grote populaties vertoont het stochastische model (bij gebruik van de 3×3 matrix uit vergelijking 30) een exponentieel groeipatroon dat vergelijkbaar is met het deterministische model.

De afname van willekeurigheid bij grotere populaties kan worden verklaard door het feit dat bij toenemende populatiegrootte de fluctuaties in het stochastische model gemiddeld worden. Hierdoor gedraagt het systeem zich steeds meer volgens de gemiddelde deterministische dynamica. Dit fenomeen wordt ondersteund door sterke convergentieargumenten en de centrale limietstelling, die aantonen dat stochastische processen naar hun deterministische equivalenten convergeren bij grotere populatiegroottes [15].

Verder valt op dat het deterministische model bij kleine populaties altijd een exponentiële groei vertoont, terwijl het stochastische model, ondanks de populatiegroefactor van $\lambda_1 \approx 2.35827$, soms een daling in de populatie vertoont of zelfs tot uitsterven leidt. Dit kan worden toegeschreven aan de willekeurigheid van het stochastische model. Ondanks de verwachte groefactor, bestaat er altijd een kans dat de populatie door toevallige gebeurtenissen kan uitsterven.

5 Conclusie

We hebben de dynamiek van populaties bestudeerd door middel van een leeftijdsafhankelijk matrixmodel, bekend als de Leslie-matrix, gecombineerd met stochastische elementen van vertakkingsprocessen. Het doel was om een beter begrip te krijgen van zowel deterministische als stochastische factoren die de groei en structuur van populaties beïnvloeden.

Het deterministische Leslie-model biedt een gestructureerde methode voor het projecteren van populatiegroei, gebaseerd op leeftijdsspecifieke overlevings- en vruchtbaarheidscijfers. Door gebruik te maken van de Perron-Frobenius-stelling, konden we cruciale inzichten verkrijgen in de asymptotische groei en de stabiele leeftijdsverdeling van de populatie. Dit model is echter beperkt in zijn vermogen om de variabiliteit en onzekerheid in populatiedynamiek volledig weer te geven.

Daarom hebben we het model uitgebreid met stochastische variabelen, met name door vertakkingsprocessen en kansverdelingen zoals de Poisson- en binomiale verdeling te gebruiken. Onze simulaties hebben aangetoond dat stochastische modellen beter in staat zijn om de onvoorspelbare aspecten van populatiedynamiek te weerspiegelen. Ondanks een dominante eigenwaarde groter dan 1, vertoonden kleine populaties in stochastische modellen soms een afname of zelfs uitsterving, wat de willekeurigheid en kwetsbaarheid van kleine populaties onderstreept.

Bij grotere populaties zagen we dat de resultaten van stochastische modellen steeds meer overeenkwamen met die van deterministische modellen. Dit komt doordat willekeurige fluctuaties gemiddeld worden, wat leidt tot gedrag dat vergelijkbaar is met de gemiddelde deterministische dynamica. Dit resultaat wordt ondersteund door sterke convergentieargumenten en de centrale limietstelling, die aantonen dat stochastische processen naar hun deterministische equivalenten convergeren bij grotere populatiegroottes [15].

Samenvattend hebben we aangetoond dat de integratie van deterministische en stochastische methoden in leeftijdsafhankelijke matrixmodellen een completer en realistischer beeld geeft van populatiedynamiek..

5.1 Epiloog

In deze scriptie hebben we uitsluitend een leeftijdsafhankelijk matrixmodel voor een één-type Galton-Watson proces gesimuleerd, een vertakkingsproces waarbij slechts één type individu betrokken is. Een multi-type vertakkingsproces vormt een uitbreiding van dit één-type Galton-Watson proces. In een multi-type vertakkingsproces heeft elk individu in de populatie een specifiek type, en het aantal nakomelingen dat een individu produceert, hangt af van het type van dat individu. De evolutie van de populatie in een stochastisch leeftijdsafhankelijk matrixmodel kan dan worden beschreven door een stochastische matrix met verschillende typen.

Zij $\mathbf{X}_n = (\mathbf{X}_n^1, \mathbf{X}_n^2, \dots, \mathbf{X}_n^k)$ het aantal individuen van elk type in generatie n , waarbij \mathbf{X}_n^j het aantal individuen van type j in generatie n is. Het voortplantingsproces wordt beschreven door een verzameling kansverdelingen $p_{ij}(k)$, die de kans aangeven dat een individu van type

j k nakomelingen van type i produceert. Alle individuen van een bepaald type hebben dus dezelfde nakomelingenverdeling. Deze nakomelingenverdeling heeft een corresponderende kansgenererende functie

$$G_{\mathbf{X}}^{(j)}(\mathbf{s}) = \sum_{x_1=0}^{\infty} \cdots \sum_{x_k=0}^{\infty} P[\mathbf{X}_1 = x_1, \dots, \mathbf{X}_k = x_k \mid \text{ouder } j] s_1^{x_1} \cdots s_k^{x_k}.$$

De populatiedynamiek kan dan worden beschreven door de stochastische matrix \mathbf{M} , waarbij elk element m_{ij} van de matrix \mathbf{M} de verwachte waarde van nakomelingen van type i , gegeven een ouder van type j , representeert [1]. De elementen m_{ij} van de matrix \mathbf{M} worden gegeven door

$$m_{ij} = \left. \frac{\partial G_{\mathbf{X}}^{(j)}(\mathbf{s})}{\partial s_i} \right|_{\mathbf{s}=\mathbf{e}},$$

waarbij \mathbf{e} de eenheidsvector is.

Stel nu dat $\mathbf{n}(0) = \mathbf{e}^{(j)}$. De gemiddelde groei van de populatie wordt verkregen door differentiatie met betrekking tot elk element van s voor bijvoorbeeld n_1 :

$$E(n_1(t+1)) = \left. \frac{\partial G_{\mathbf{n}(t+1)}^{(j)}(\mathbf{s})}{\partial s_1} \right|_{\mathbf{s}=\mathbf{0}}.$$

De multi-type vertakkingsprocessen kunnen worden geclassificeerd als subkritisch, kritiek, of superkritisch, afhankelijk van $\lambda < 1$, $\lambda = 1$, of $\lambda > 1$, waarbij λ de dominante eigenwaarde is van \mathbf{M} . Subkritische en kritische processen sterven met een kans van 1 uit. Superkritische processen hebben een uitstervingskans die kleiner is dan 1 [3].

Referenties

- [1] Linda J. S. Allen. *An Introduction to Stochastic Processes with Applications to Biology*. CRC Press, Taylor Francis Group, second edition edition, 2010.
- [2] K. Athreya and P. Ney. *Branching Processes*. Springer-Verlag, 1972.
- [3] Hal Caswell. *Matrix Population Models: Construction, Analysis, and Interpretation*. Sinauer Associates, Inc., second edition edition, 2001.
- [4] University of Auckland Department of Statistics. *STATS 325: Stochastic Processes Course Notes*. n.d.
- [5] Colby D. Dormann. Stochastic versions of the matrix model for population dynamics. <https://bookdown.org/cdorm/lefko3gentle/stochan.html#sec:iidseq>, 2023. Accessed: 2024-06-04.
- [6] R. Durrett. *Branching Process Models of Cancer*. Springer International Publishing, 2015.
- [7] Samy Gallego. Modelling population dynamics of elephants. *Amarula Elephant Research Programme, School of Life and Environmental Sciences, University of Natal, Durban*, 2003.
- [8] Thomas Hawkins. Continued fractions and the origins of the perron–frobenius theorem. *Archive for History of Exact Sciences*, 62(6):655–717, 2008.
- [9] Roger A. Horn and Charles R. Johnson. *Matrix Analysis*. Cambridge University Press, 1985.
- [10] Marek Kimmel and David E. Axelrod. *Branching Processes in Biology*. Springer-Verlag, 2002.
- [11] Maxim Klin. Lecture notes on monte carlo methods, 2019. Accessed: 2024-06-17.
- [12] Carl D. Meyer. *Matrix Analysis and Applied Linear Algebra*. Society for Industrial and Applied Mathematics (SIAM), 2000.
- [13] Kevin T. Shoemaker. Nres 470: Lab 4 - population genetics, 2024. Accessed: 2024-06-06.
- [14] Z.M. Sykes. Some stochastic versions of the matrix model for population dynamics. *Journal of the American Statistical Association*, 64:111–30, 1969.
- [15] Viet Chi Tran. Large population limit and time behaviour of a stochastic particle model describing an age-structured population. *ESAIM: Probability and Statistics*, 12:345–386, 2008.

A Bijlagen

A.1 Code voor Deterministische Populatieprojectie in Python

```
1 import numpy as np
2 import matplotlib.pyplot as plt
3
4 # Matrixprojectie in Python
5 def ProjecteerPopulatie(LeslieMatrix, BeginPopulatie, AantalTijdstappen):
6     AlleTijdstappen = np.zeros((LeslieMatrix.shape[0], AantalTijdstappen +
7     1))
8     AlleTijdstappen[:, 0] = BeginPopulatie
9
10    for t in range(1, AantalTijdstappen + 1):
11        AlleTijdstappen[:, t] = np.dot(LeslieMatrix, AlleTijdstappen[:, t
12        - 1])
13
14    return AlleTijdstappen
15
16 # Definieer de Leslie-matrix (LeslieMatrix)
17 LeslieMatrix = np.array([
18     [1.5, 2.0, 2.5],
19     [0.8, 0.0, 0.0],
20     [0.0, 0.5, 0.0]
21 ])
22
23 # Definieer de beginpopulatievector (BeginPopulatie)
24 BeginPopulatie = np.array([1, 1, 1])
25
26 # Stel het aantal projectietijdstappen in
27 AantalTijdstappen = 5
28
29 # Voer de projectie uit
30 AlleTijdstappen = ProjecteerPopulatie(LeslieMatrix, BeginPopulatie,
31 AantalTijdstappen)
32
33 # Plot de resultaten
34 plt.figure(figsize=(10, 6))
35 kleuren = ['orange', 'purple', 'red']
36 for i in range(LeslieMatrix.shape[0]):
37     plt.plot(AlleTijdstappen[i, :], label=f'Leeftijdsgroep {i + 1}', color
38     =kleuren[i])
39
40 # Plot de totale populatie
41 TotalePopulatie = np.sum(AlleTijdstappen, axis=0)
42 plt.plot(TotalePopulatie, label='Totale Populatie', color='black',
43     linestyle='--', linewidth=2)
44
45 plt.xlabel('de tijd (t)')
46 plt.ylabel('Aantal Individuen')
47 plt.title('Deterministisch model')
48 plt.legend()
49 plt.show()
```

```

45
46 # Bereken lambda (dominante eigenwaarde van de Leslie-matrix)
47 lambda_value = max(np.linalg.eigvals(LeslieMatrix))
48 print(f'Lambda: {lambda_value.real}')
49
50 # Bereken stabiele leeftijdsverdeling (eigenvector die overeenkomt met de
    dominante eigenwaarde)
51 _, eigenvectors = np.linalg.eig(LeslieMatrix)
52 StabieleLeeftijdsverdeling = eigenvectors[:, np.argmax(np.abs(np.linalg.
    eigvals(LeslieMatrix)))] .real
53 StabieleLeeftijdsverdeling /= sum(StabieleLeeftijdsverdeling)
54 print(f'Stabiele Leeftijdsverdeling: {StabieleLeeftijdsverdeling}')

```

Listing 5: Python code voor populatieprojectie

A.2 Coder voor Stochastische Populatieprojectie in Python

```

1 import random
2 import numpy as np
3 import matplotlib.pyplot as plt
4
5 # Formulering van het stochastische proces
6 def PopulatieOpTijdstip(t):
7     PopulatieVector = BeginPopulatieVector
8     for _ in range(t):
9         PopulatieVector = WillekeurigeGeboorten(PopulatieVector)
10    return PopulatieVector
11
12 def WillekeurigeGeboorten(PopulatieVector):
13     Nakomelingen = 0
14     for i, Geboortecijfer in enumerate(Geboortecijfers):
15         for _ in range(int(PopulatieVector[i])):
16             Nakomelingen += np.random.poisson(Geboortecijfer)
17
18     NieuweGeneratie = np.zeros(len(Geboortecijfers))
19     NieuweGeneratie[0] = Nakomelingen
20     for i in range(1, len(Geboortecijfers)):
21         NieuweGeneratie[i] = np.random.binomial(int(PopulatieVector[i-1]),
22             Overlevingskansen[i-1])
23
24     return NieuweGeneratie
25
26 # Definieer geboortecijfers en overlevingskansen
27 Geboortecijfers = [1.5, 2.0, 2.5]
28 Overlevingskansen = [0.8, 0.5, 0.0]
29
30 # Definieer begin populaties in elke leeftijdsgroep
31 BeginPopulaties = [1, 1, 1]
32
33 # Maak de Leslie-matrix met de gedefinieerde cijfers
34 LeslieMatrix = np.array([

```

```

35     [Overlevingskansen[0], 0.0, 0.0],
36     [0.0, Overlevingskansen[1], 0.0]
37 ])
38
39 # Maak de begin populatievector
40 BeginPopulatieVector = np.array(BeginPopulaties)
41
42 print("Leslie-matrix:", LeslieMatrix)
43 print("Begin populaties:", BeginPopulatieVector)
44
45 AantalTijdstappen = 5
46
47 TotalePopulatieStochastisch = []
48 PopulatieVectorsStochastisch = []
49
50 # Bereken en sla de totale populatie en verdelingen voor beide modellen op
51 for t in range(AantalTijdstappen + 1):
52     PopulatieOpT = PopulatieOpTijdstip(t)
53     TotalePopulatie = np.sum(PopulatieOpT)
54     TotalePopulatieStochastisch.append(TotalePopulatie)
55     PopulatieVectorsStochastisch.append(PopulatieOpT)
56
57     print(f"Populatie op t={t}: {PopulatieOpT}")
58     print(f"De totale populatie na {t} tijdstappen met behulp van het
59           stochastische proces is {TotalePopulatie}")
60
61 # Plot de totale populatie en leeftijdsgroepverdelingen over tijd met
62 # behulp van het stochastische proces
63 plt.figure(figsize=(10, 6))
64
65 PopulatieVectorsStochastisch = np.array(PopulatieVectorsStochastisch).T
66
67 # Plot leeftijdsgroep populaties
68 plt.plot(range(AantalTijdstappen + 1), PopulatieVectorsStochastisch[0, :],
69         label="Leeftijdsgroep 1", color="orange")
70 plt.plot(range(AantalTijdstappen + 1), PopulatieVectorsStochastisch[1, :],
71         label="Leeftijdsgroep 2", color="purple")
72 plt.plot(range(AantalTijdstappen + 1), PopulatieVectorsStochastisch[2, :],
73         label="Leeftijdsgroep 3", color="red")
74
75 # Plot de totale populatie over tijd met behulp van het stochastische
76 # proces
77 plt.plot(range(AantalTijdstappen + 1), TotalePopulatieStochastisch, label=
78         "Totale Populatie", color="black", linewidth=2, linestyle='--')
79
80 plt.xlabel("Tijd (t)")
81 plt.ylabel("Aantal Individuen")
82 plt.title("Stochastisch model")
83 plt.legend()
84 plt.show()

```

Listing 6: Python code voor populatieprojectie

```

1 import numpy as np

```



```

2 import matplotlib.pyplot as plt
3
4 # Matrixprojectie in Python
5 def ProjecteerPopulatie(LeslieMatrix, BeginPopulatie, AantalTijdstappen):
6     AlleTijdstappen = np.zeros((LeslieMatrix.shape[0], AantalTijdstappen +
7         1))
8     AlleTijdstappen[:, 0] = BeginPopulatie
9
10    for t in range(1, AantalTijdstappen + 1):
11        AlleTijdstappen[:, t] = np.dot(LeslieMatrix, AlleTijdstappen[:, t
12            - 1])
13
14    return AlleTijdstappen
15
16 # Definieer de Leslie-matrix (LeslieMatrix)
17 LeslieMatrix = np.array([
18     [1.5, 2.0, 2.5],
19     [0.8, 0.0, 0.0],
20     [0.0, 0.5, 0.0]
21 ])
22
23 # Definieer de beginpopulatievector (BeginPopulatie)
24 BeginPopulatie = np.array([1, 1, 1])
25
26 # Stel het aantal projectietijdstappen in
27 AantalTijdstappen = 5
28
29 # Voer de projectie uit
30 AlleTijdstappen = ProjecteerPopulatie(LeslieMatrix, BeginPopulatie,
31     AantalTijdstappen)
32
33 # Plot de resultaten
34 plt.figure(figsize=(10, 6))
35 kleuren = ['orange', 'purple', 'red']
36 for i in range(LeslieMatrix.shape[0]):
37     plt.plot(AlleTijdstappen[i, :], label=f'Leeftijdsgroep {i + 1}', color
38         =kleuren[i])
39
40 # Plot de totale populatie
41 TotalePopulatie = np.sum(AlleTijdstappen, axis=0)
42 plt.plot(TotalePopulatie, label='Totale Populatie', color='black',
43     linestyle='--', linewidth=2)
44
45 plt.xlabel('de tijd (t)')
46 plt.ylabel('Aantal Individuen')
47 plt.title('Deterministisch model')
48 plt.legend()
49 plt.show()
50
51 # Bereken lambda (dominante eigenwaarde van de Leslie-matrix)
52 lambda_value = max(np.linalg.eigvals(LeslieMatrix))
53 print(f'Lambda: {lambda_value.real}')
54
55 # Bereken stabiele leeftijdsverdeling (eigenvector die overeenkomt met de

```

```
    dominante eigenwaarde)
51 _, eigenvectors = np.linalg.eig(LeslieMatrix)
52 StabieleLeeftijdsverdeling = eigenvectors[:, np.argmax(np.abs(np.linalg.
    eigvals(LeslieMatrix)))] .real
53 StabieleLeeftijdsverdeling /= sum(StabieleLeeftijdsverdeling)
54 print(f'Stabiele Leeftijdsverdeling: {StabieleLeeftijdsverdeling}')
```

Listing 7: Python code voor populatieprojectie