An Integral Projection Model for Giant Hogweed

by

Jonathan Peter Drake

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Master of Mathematics
in
Applied Ecology

Waterloo, Ontario, Canada, 2019

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Abstract

In this thesis I investigate the impact that the choice of model may have on predictions of giant hogweed population dynamics. It has been shown in two case studies that population growth rates predicted by matrix population models may be biased if the number of plants sampled is low and plants are classified based on a continuous measure of their size [79]. These same studies have shown that integral projection models do not provide biased estimates of population growth rates for populations with few plants sampled. In chapter 2 I construct a density-independent integral projection model for giant hogweed population dynamics and I use it to verify that the population growth rates do not significantly differ from a previously published matrix model of giant hogweed population dynamics [39]. This research indicates that the conclusions made using the matrix population model are unaffected by the decision to discretize plant size.

A second major topic of research for giant hogweed populations is to predict the rate at which the species spreads. It has been shown that the rate of spread of a plant may be affected by the number of seeds produced by the population [68]. In chapter 3 I develop two density-dependent integral projection models for giant hogweed populations and compare the total seed production predicted by each model. In both models I allow for recruitment to be limited by competition among seedlings. However, in the second model I also allow the probability of flowering to depend on intraspecific density. I find that the model with density-dependent flowering has oscillatory seed production and that the model predicts significantly fewer seeds every other year compared to the model with density-independent flowering. I conclude that accounting for competition among adult plants may affect predictions of the rate of spread of giant hogweed.

Acknowledgements

I would like to thank my supervisor, Dr. Kim Cuddington, for her invaluable guidance throughout my research and through the writing of this thesis. In particular, I'd like to thank her for helping me to determine the topic for my thesis, for helping me to formulate hypotheses to test for each chapter, for helping me identify and overcome obstacles in my research, and for all her time helping me edit and improve my work. Additionally, I would like to thank Dr. Brad Fedy and Dr. Brian Ingalls for helping edit this thesis.

I would like to thank those who collected the data used throughout this thesis. Thank you to Dr. Jörg Hüls for braving dozens of giant hogweed stands in order to collect most of the demographic data needed to parametrize my models, and thank you to Dr. Cuddington and Meghan Grguric for collecting data on seedling germination and establishment. And thank you to Markus Wieland at the Geospatial Center for helping me find GIS data to use in a spatial model (even though that model was cut from the final thesis).

I am grateful to my officemates Matt Ambacher, Anthony Caterini, Shawn Corvec, and Tony-Pierre Kim, to my labmate Jody Daniel, to Paul Tiede, to the people in the Bauch lab, and to all other members of the applied math and biology departments who made writing this thesis bearable. Finally, I am grateful to my family for helping to support and encourage me for the past three years.

Dedication

To my mom, Joan, and my dad, Peter

Table of Contents

List of Tables x

List of Figures xii

1 Literature Review 1

1.1 Background and motivation . . . 1

1.2 Matrix population models . . . 2

1.2.1 Introduction to matrix models . . . 2

1.2.2 Previous matrix models for giant hogweed . . . 3

1.3 Integral projection models . . . 5

1.3.1 Why an integral projection model? . . . 5

1.3.2 Overview of integral projection models . . . 6

1.3.3 Density-dependent integral projection models . . . 7

1.4 Outline of thesis . . . 7

2 An Integral Projection Model for Giant Hogweed Management Decisions 8

2.1 Introduction . . . 8

2.2 Methods . . . 10

2.2.1 Study species . . . 10

2.2.2 Experimental data collection . . . 11

2.2.3 Model formulation . . . 11

2.2.4 Parameterization . . . 13
2.2.5 Life cycle analysis . . . 17
2.2.6 Bootstrap confidence intervals . . . 18
2.3 Results . . . 19
2.3.1 Results of statistical analysis . . . 19
2.3.2 Results of demographic analysis . . . 21
2.3.3 Impact of outliers . . . 23
2.4 Discussion . . . 23

3 The Impact of Density-Dependent Flowering on *H. mantegazzianum* Seed Production 28
3.1 Introduction . . . 28
3.2 Methods . . . 30
3.2.1 Study species . . . 30
3.2.2 Experimental data collection . . . 30
3.2.3 Towards a density-dependent integral projection model . . . 31
3.2.4 Parameterization . . . 35
3.2.5 Simulation results . . . 42
3.2.6 Equilibrium size distribution and stability . . . 42
3.3 Results . . . 42
3.3.1 Stability results for each model . . . 42
3.3.2 Seed production predicted by each model . . . 43
3.3.3 Differences in mean plant size . . . 44
3.4 Discussion . . . 45

4 Conclusions and Future Work 50
4.1 Conclusions . . . 50
4.2 Future work . . . 50

References **52**

Appendices **62**

A Chapter 2 Code **63**

A.1 Determining Population Growth Rates 63

A.2 Diagnostics 72

A.2.1 Unintentional Eviction 72

A.2.2 Form of Growth 75

A.2.3 Normality of Residuals and Homoskedasticity 77

A.2.4 Log-normality of the Recruit Size Distribution 78

A.3 Bootstrapping 78

A.3.1 Bootstrap population growth rate 78

A.4 Methods Plots 88

A.4.1 Plot vital rate functions 88

A.4.2 Plot recruit size distribution 90

A.5 Results Plots 91

A.5.1 Plot predicted size distribution vs observations 91

B Chapter 3 Code **94**

B.1 Simulation Function 94

B.2 Diagnostics 104

B.2.1 Form of Growth 104

B.2.2 Density-Dependence and Site-Specific Effects 106

B.2.3 Normality of Residuals and Homoskedasticity 107

B.2.4 Log-normality of the Recruit Size Distribution 108

B.3 Bootstrapping 108

B.3.1 Bootstrap population growth rate 108

B.4 Determine Equilibrium and Stablility 123
B.4.1 Determine equilibrium 123
B.4.2 Determine stability of equilibrium 129
B.5 Methods Plots 130
B.5.1 Plot vital rate functions 130
B.5.2 Plot recruit size distribution 132
B.6 Results Plots 133
B.6.1 Simulate for 30 years 133
B.6.2 Mean height comparison 134

List of Figures

1.1 Life cycle diagram of the matrix models developed by Hüls and coauthors [38] and by Pergl and coauthors [73]. Nodes pictured represent (S) small vegetative plants, (M) medium vegetative plants, (L) large vegetative plants, and (R) reproductive plants. Dashed lines indicate reproduction. Arrows marked with an asterisk (*) indicate transitions that were included in the model developed by Hüls and coauthors but not the model developed by Pergl and coauthors. Retrogression is possible due to grazing or as a response to stress [85]. 4

2.1 Data and best-fit functions for relationships between plant height and vital rates for open and dense stands. Survival probability fit using logistic regression (a, b), probability of reproduction fit using logistic regression (c, d), and expected height of surviving plants fit using linear regression (e, f) in open (a, c, e), and dense (b, d, f) stands of giant hogweed in the Hesse region of Germany. Ticks on the figures fit using logistic regression represent data points with plant height indicated. Plants that survive (a, b) and plants that flower (c, d) have ticks on the top while plants that do not survive (a, b) or do not flower (c, d) have ticks on the bottom. All data collected by Hüls and coauthors [39]. 15

2.2 Histogram of the observed distribution of recruit size and fitted log-normal distribution for open (n=8, a) and dense (n=67, b) stands of giant hogweed in the Hesse region of Germany. All data collected by Hüls and coauthors [39]. 20

2.3 Histogram of observed size distribution in 2003 vs predicted stable size distribution for open (a) and dense (b) stands. All data collected by Hüls and coauthors [39]. 22

3.1 Data and best-fit functions for relationships between plant height and vital rates. Survival probability (a), expected height of vegetative (b, black dots) and reproductive plants (b, unfilled rhombi), and probabilities of reproduction for the recruit-limited model (c) and adult competition model with zero biomass (d). All data collected by Hüls and coauthors [39]. 38

3.2 Number of seedlings as a function of the seed set in the plot (a) and histogram of observed distribution of recruit size along with the fitted log-normal distribution for stands of giant hogweed in the Hesse region of Germany ($n = 75$, b). 41

3.3 Number of seeds in each year predicted by the recruit-limited model (circles) and the model with adult competition (rhombi). 43

List of Tables

2.1 Site information for data collected by Hüls and coauthors [39]. The maximum distance between sites is 56 km. 11

2.2 Vital rate function forms and parameter estimates for the integral projection model. Standard errors for parameter estimates are in parentheses. All data collected by Hüls and coauthors [39]. 17

2.3 Evaluation of the linear and nonlinear functions used to model inter-annual growth of giant hogweed in Germany [39]. The best performing fit for each data set is in bold. 19

2.4 Population growth rate (λ), bootstrap estimate of population growth rate (λ_b), and bootstrap 95% confidence interval (CI-, CI+). Bootstrap values were computed using 5,000 bootstrap samples. All data collected by Hüls and coauthors [39]. 21

2.5 Population growth rate (λ), bootstrap estimate of population growth rate (λ_b), and bootstrap 95% confidence interval (CI-, CI+) as the bootstrap replicate sample size (N) increases. Bootstrap values were computed using 5,000 bootstrap samples. All data collected by Hüls and coauthors [39]. . . 22

2.6 Mean plant height ($\bar{h}$), bootstrap mean plant height ($\bar{h}_b$), and bootstrap 95% confidence intervals (CI-, CI+). Bootstrap values were computed using 5,000 bootstrap samples. All data collected by Hüls and coauthors [39]. . . 23

3.1 Estimates of seed production from the literature. 31

3.2 Seedling germination data collected in Ontario. 32

3.3 Impact of biomass and site-specific effects on fits for vital rate functions for giant hogweed. Vital rates were each fit twice - the first fit does not use biomass as a covariate in regression while the second fit does use biomass as a covariate. The best performing fit for each data set is in bold. All data collected by Hüls and coauthors [39]. 36

3.4 The final forms of the vital rate functions. All functions were fit using data collected by Hüls and coauthors [39]. RL=recruit-limited and AC=adult competition. Standard error is indicated in parentheses. The Michaelis constant for mean recruits was fit using data collected by Cuddington and Grguric. 37

3.5 Bootstrap mean seed production and confidence intervals for years 2, 3, 4, 5, 6, 7, 29, and 30. RL=recruit-limited and AC=adult competition. 500 bootstrap simulations were performed for each model. 44

3.6 Bootstrap observed mean height and predicted mean height of vegetative and reproductive plants. 500 bootstrap simulations were performed for each model. All heights presented in centimeters. 45

Chapter 1

Literature Review

> The spread of *H. mantegazzianum* is likely to continue through much of southern Canada over the next 25-100 yr with worsening ecological, economic, and health effects
>
> *Page et al., 2006 [69]*

1.1 Background and motivation

Giant hogweed (*Heracleum mantegazzianum* Somm. & Lev.) is a large, monocarpic perennial terrestrial plant native to the Caucasus mountain range [93]. The species is highly invasive, having successfully invaded 19 countries in Europe over the past 70 years, and is in the process of colonizing northeastern and northwestern North America [69]. As with many invasive, non-native plants, giant hogweed poses a serious threat to biodiversity in its invaded habitat [76]. Giant hogweed plants are able to suppress native plant species due to their large size and rapid sprouting in the spring [93] and due to their production of allelopathic chemicals that inhibit germination of competitors [42].

In addition to the impact on native flora, giant hogweed is a pest to humans. Giant hogweed plants produce phototoxic furanocoumarins as an insecticide, but these furanocoumarins will also cause third-degree burns or blindness in humans [93]. Finally, as an invasive riverside plant giant hogweed frequently blocks access to amenity areas [86]

and, during winter, plants die back leading to bare ground which makes the riverbank more susceptible to erosion [86]. For these reasons giant hogweed has been designated as a noxious weed in numerous jurisdictions, including the United States [1] and Ontario [2].

There has been more research on the European hogweed invasion since the invasion began earlier and is more advanced than the invasion in North America; however, giant hogweed invasion in North America remains a serious threat [69, 72, 90]. Although the species is unlikely to be eradicated in either Europe or North America, there is a silver lining in that much of what has been learned from the invasion of Europe may be applied to mitigate the effects of giant hogweed invasion in North America. In addition to *in situ* evaluations of giant hogweed control strategies, mathematical models of giant hogweed population dynamics have been constructed to better understand factors influencing giant hogweed invasion and control [38, 39, 60, 62, 63, 73, 72].

1.2 Matrix population models

1.2.1 Introduction to matrix models

For many plant species, including giant hogweed, the vital rates of individuals within a population differ with the age, stage, or size of the individual. To account for this heterogeneity, structured population models can be used which allow for the vital rates of individuals to be dependent on their state. Allowing vital rates to depend on state is particularly important for modelling invasive species since researchers can determine which individuals have the largest impact on population growth [18]. This knowledge may be used to develop effective management strategies.

By far the most common type of structured model used to describe plant population dynamics are matrix population models, with one recent study finding 355 out of 396 plant population modelling papers surveyed incorporating matrix models in some capacity [16]. Matrix models are discrete-time population models in which individuals are classified into a finite number of discrete classes [12]. The class of an individual is assumed to be representative of its state so that the vital rates of the individual are solely determined by the class it occupies.

To model the dynamics of a given population, matrix population models assume the number of individuals in each class in the next time-step will rely on how many individuals transition to that state. More specifically, matrix models assume the number of individuals in each class after one time-step may be written as a linear combination of the current

number of individuals in each class. The advantage of this approach is that simple matrix models are easy to construct, easy to parameterize, and methods to analyse matrix models are well developed.

For deterministic, density-independent matrix models of the form $n(t+1) = An(t)$ the population vector, $n(t)$, will converge to a stable population distribution dependent on the demographic matrix, A [12]. Once the population has reached its stable distribution, the population vector at the next time step will be a multiple of the current population vector. That is, $n(t+1) = \lambda n(t)$ for some value λ. The largest λ that satisfies the equation is the dominant eigenvalue, which is the population growth rate in a matrix model. In addition, sensitivity and elasticity analysis may be performed to determine how small perturbations in the elements of A affect population growth rate [12].

1.2.2 Previous matrix models for giant hogweed

Hüls and coauthors collected data from the Hesse region of Central Germany from 2002-2004 [38, 39, 71] while Pergl and coauthors collected data from the Slavkovský les region in the Western part of the Czech Republic from 2002-2005 [71, 72, 75]. Both sets of authors classified individuals into three stages of small, medium, or large vegetative plants along with a fourth stage for flowering plants. The classification made by Hüls and coauthors was dependent on plant height and the laminar width of the largest leaf while the classification made by Pergl and coauthors defined seedlings as small plants, juveniles as medium sized plants, and rosette plants as large plants. In both studies, the state of each plant was recorded every year which allowed the authors to determine the probability a plant in one class may transition to another state. Each research group then used the transition data to construct density-independent matrix models to predict the size and structure of the population in future years (Figure 1.1).

In addition to classification based on size, Hüls and coauthors classified each 2.5 m^2 plot for which they collected data as either dense or open stands depending on the dominance of giant hogweed in the area [39]. The authors had originally hypothesized that open stands occur at the front of giant hogweed invasions and that these sparse stands are precursors to dense stands. However, the authors' analysis of their matrix model indicated that open stands do not have a higher intrinsic population growth rate than dense stands. They concluded that dense stands are saturated with large individuals which suppress germination while open stands are likely limited by irregular disturbances such as by mowing or grazing. Furthermore, the authors found that population growth rate in dense stands is more sensitive to survival of existing plants while population growth rate is more sensitive to the growth of individuals in open stands.

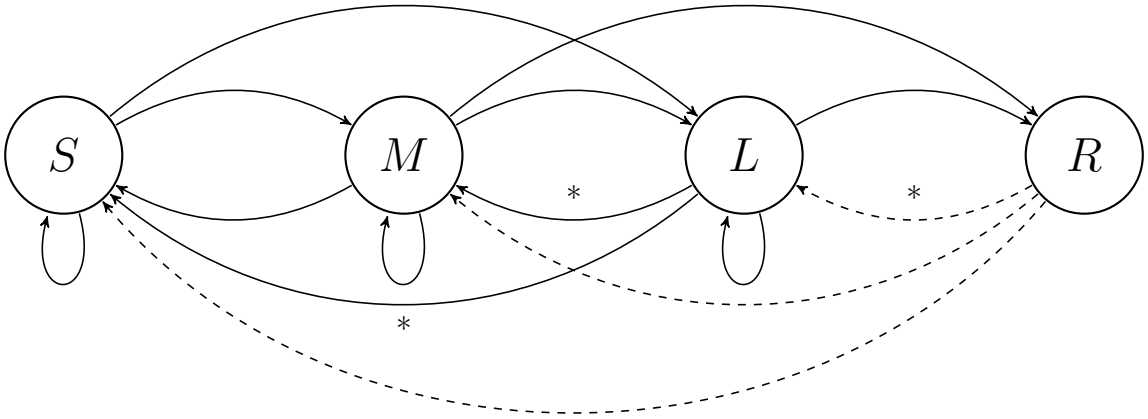

Figure 1.1: Life cycle diagram of the matrix models developed by Hüls and coauthors [38] and by Pergl and coauthors [73]. Nodes pictured represent (S) small vegetative plants, (M) medium vegetative plants, (L) large vegetative plants, and (R) reproductive plants. Dashed lines indicate reproduction. Arrows marked with an asterisk (*) indicate transitions that were included in the model developed by Hüls and coauthors but not the model developed by Pergl and coauthors. Retrogression is possible due to grazing or as a response to stress [85].

For 2002-2003, Hüls and coauthors found that populations were in decline [39]. The authors attributed this decline to the unusually hot and dry conditions of the summer of 2003. These conditions led to low productivity which resulted in increased mortality. They found that plants responded to this increased mortality by increasing reproductive output to take advantage of the decrease in ground cover. Pergl and coauthors [71] compared their results to those of Hüls and coauthors and found that their surveyed populations are decreasing in the Czech Republic, complementing the conclusion by Hüls and coauthors that populations are decreasing in Germany. However, the German and Czech study periods overlapped and thus the decrease in populations in the Czech Republic may also be due to the extreme temperature during the study period.

1.3 Integral projection models

1.3.1 Why an integral projection model?

A major drawback of matrix projection models, such as those developed by Hüls and coauthors [38, 39, 71] and Pergl and coauthors [71, 72, 73], is that they require modellers to classify plants into discrete stages in their life-cycle. In a plant such as giant hogweed, the divisions between classes are often based on continuous variables. The resulting classes are constructed arbitrarily with little regard for the underlying biology of the plant. For example, in the matrix model by Hüls and coauthors stage 1 plants were defined as having laminar width between 3.5-20 cm and height less than 50 cm while stage 2 plants have laminar width between 20-50 cm and height greater than 50 cm but not reaching the canopy. The matrix models cannot account for variation in vital rates for plants within each class. As a result, the choice of boundaries between classes may have an impact on model predictions [79]. Additionally, the number of classes has been shown to bias predicted population growth rate, and population growth rate may be underestimated when there are not very many individuals [79].

There are several continuous size individual-based models (IBMs) for giant hogweed [96, 63, 60, 61, 62]. The hypotheses in this thesis may be answered using IBMs similar to previously published IBMs; however, they do not offer any advantages over integral projection models. Furthermore, the hypotheses in this thesis relate to previously published matrix models. I opt to use integral projection models throughout this thesis due to their structural similarity to matrix models.

1.3.2 Overview of integral projection models

In his 1998 dissertation Easterling introduced integral projection models (IPMs) [25]. These models are discrete-time, size-structured population models that allow for individuals within a population to be described by a continuous trait using an integral operator [25] (as cited in [24]). This type of model allows researchers to avoid arbitrary classifications of individuals based on size by allowing vital rates in the population to depend on a continuous measure of size. For example, we may use the height of a plant as the relevant trait, in which case the population may be represented by a size distribution $n(x,t)$ where x is the height of an individual, t is the time, and $n(\cdot,t)$ is the distribution of individuals at time t. Projection from one time-step to the next relies on a kernel, $K(y,x)$, that yields the probability that a plant will transition from state x to state y over a timestep. This kernel is comprised of various size-dependent vital rates functions that may be determined through regression. The expected population at a given size in the next time step is dependent on the contribution of every plant that may transition to that size. Thus we integrate the kernel multiplied by the size distribution at time t over the size x to get the projected size distribution at the next time-step. Mathematically, this is written as:

$$n(y,t+1) = \int_{\omega} K(y,x)n(x,t)dx \tag{1.1}$$

The form of the kernel will be determined by the life cycle of the species and parametrization is then an exercise in statistical regression to fit survival, growth, and reproduction components of the kernel. The survival and reproduction functions are typically found using logistic regression while the growth function may be found using linear or non-linear regression. Since being introduced by Easterling, IPMs have been developed to accommodate discrete states, time lags, spatial structure, environmental stochasticity, and demographic stochasticity [26]. Additionally, in making the adjustment from matrix models to IPMs, the potential for analysis is not greatly diminished. The theory used to analyze IPMs is slightly less developed; however, most of the useful properties of matrix models such as the population growth rate, stable size distribution, and net reproductive ratio may still be computed for IPMs [24, 26].

1.3.3 Density-dependent integral projection models

The probability of survival, probability of flowering, expected increase in plant size, and expected number of recruits are all functions that may depend on the presence of other plants. Accounting for the competition among giant hogweed plants will therefore be necessary when modelling the course of giant hogweed invasion since plant density will change over time. However, there are limitations to the construction of density-dependent IPMs.

As is necessary for density-dependent matrix models, the forms of the vital rate functions must be restricted in order to parameterize the model. However, density-dependent IPMs have an advantage over matrix models in that they may be parameterized in cases where matrix models cannot. In particular, in matrix models each element may depend on giant hogweed density, but not every transition will have enough observations for such a function to be parameterized. This limitation is overcome in integral projection models by assuming that population dynamics are describes using a continuous kernel and assuming that the functions describing the kernel may be fit using regression.

1.4 Outline of thesis

This thesis is divided into two main chapters. In the first chapter I develop a density-independent integral projection model analogous to the matrix model constructed by Hüls and coauthors [39]. I compare the predicted population growth rates for each model to determine if the conclusions originally reached by Hüls and coauthors are affected by their use of a matrix model, as implied by the results of Ramula and coauthors [79]. In the second chapter I develop a density-dependent integral projection model in order to examine the transient and asymptotic behaviour of giant hogweed populations. I use this model to determine the impact that competition among adult plants will have on seed production in giant hogweed stands. The two chapters are united in that they investigate how increasing model complexity affect our understanding of giant hogweed spread.

Chapter 2

An Integral Projection Model for Giant Hogweed Management Decisions

Les fleurs sont si contradictoires!

Antoine de Saint-Exupéry,
Le Petit Prince

2.1 Introduction

A fundamental concern in the management of invasive species is to assess the rate at which invasive population size changes. For many populations, researchers construct matrix models in which individuals are classified into discrete stages based on the individual's state and use these models to determine long-term population growth rates [12]. However, the population growth rate predicted by a matrix model may be biased if this classification is based on a continuous measure of size and the sample size is small [20, 79]. This potential for bias in estimated population growth rate may affect management priorities or may lead officials to underestimate the threat of an invasive species.

Giant hogweed (*Heracleum mantegazzianum* Somm. & Lev.) is an invasive species that has become naturalized in parts of Europe and North America [69]. Giant hogweed plants pose a threat to biodiversity in its invaded ranges [76], can negatively affect human

health [93], and facilitate riverside erosion [86]. These potential impacts have prompted researchers to evaluate control strategies for the species. For example, Hüls and coauthors [39] investigated the relation between populations in dense, monospecific stands near the centre of infestations and populations in open stands near the margins that were intermixed with surrounding vegetation. The authors collected demographic data from the two types of giant hogweed stands and used the data to parameterize matrix models for the populations [39]. The authors noted that if the population growth rate in open stands was higher than the population growth rate in dense stands then they may represent the front of an invasion and should be prioritized for control. Using their matrix model, Hüls and coauthors found no significant difference in population growth rates between the stand types. Since open stands do not represent the front of the invasion, the authors concluded that they should not be prioritized for control. However, the use of a stage-structured model for giant hogweed populations may be unsuitable to address the authors' hypothesis. In particular, the low sample size of the open stand data set may produce a biased estimate of the population growth rate [20, 79], which in turn may affect the conclusions made by Hüls and coauthors.

Integral projection models (IPMs) are an alternative to matrix models that avoid discretizing populations that have continuous traits such as size [24]. Previous authors have speculated that avoiding this discretization would prevent bias in the predicted population growth rate since plants vital rate functions would be more precise [27]. Ramula and coauthors tested the impact that sample size has on matrix models and IPMs and found that for large demographic data sets IPMs yield the same population growth rate as matrix models [79]. As sample size decreased, the authors found that the mean estimate of population growth rate predicted by the IPMs remained the same; however, the mean estimate of population growth rate predicted by the matrix models changed. Ramula and coauthors concluded that matrix models may yield biased predictions of population growth rate when the data set used to parameterize the model is small. For both of the species studied by Ramula and coauthors, the population growth rate predicted by the matrix model decreased once the sampled population dropped below 300 plants.

The results of Hüls and coauthors may be affected since the data sets used to parameterize their matrix model vary in size [39]. The open stand data set is small enough to produce biased estimates of population growth rate. In contrast, the dense stand data set should yield an unbiased estimate of population growth rate. The conclusions of Hüls and coauthors that depend on these predicted population growth rates could therefore change with the use of an integral projection model. This possibility that the selection of the modelling approach may affect management decisions is my primary motivation for revisiting the data set collected by Hüls and coauthors.

In this chapter I investigate whether the decision to use a matrix model for giant hogweed population dynamics affected the conclusions made by Hüls and coauthors [39]. The results of Ramula and coauthors imply that the demographic data set used to parameterize the matrix model for dense stands is large enough to produce an unbiased estimate of population growth rate [79]. However, the data set of plants in open stands is too small to guarantee that the estimate of population growth rate is unbiased. To test this hypothesis I constructed an integral projection model and parameterized it for both stand types using data collected by Hüls and coauthors during the 2002-2003 growing season. I then used the population growth rates predicted by the integral projection model to revisit the conclusion made by Hüls and coauthors that open stands are not precursors to dense stands. My analysis will demonstrate whether the choice between using a matrix model or integral projection model for populations with a low number of records may affect management recommendations.

2.2 Methods

2.2.1 Study species

Heracleum mantegazzianum is a monocarpic perennial species in the family Apiaceae [69] that propagates exclusively by seed [78]. Seeds undergo dormancy breaking via wet and cold stratification [58] and typically germinate in spring the year after they set, but under suitable conditions may germinate the same year in autumn [93]. In the year a plant germinates it will direct much of its resources towards its taproot [69], causing the accumulation of aboveground biomass to be slow the first year [93]. Growth is much more vigorous in subsequent years once the root system is developed. After 3-5 years of growth plants will have accumulated enough resources to begin flowering [69].

If root reserves are sufficient at the beginning of the growing season a plant will initiate flowering that year [69]. Plants that will flower begin the year with vigorous vegetative growth, followed by stem elongation and umbel formation sometime between late April and early June [10, 93]. This vegetative growth and stem elongation causes flowering plants to grow extremely large, with most flowering plants between 3 m and 4 m tall and some recorded up to 5.5 m in height [78]. Flowering typically occurs over 5 or 6 weeks in early-mid summer while seeds ripen later in the summer in August and September [69].

Occurance of *H. mantegazzianum* in its invaded range may be limited to sparse stands of only a few individuals, linear stands along dispersal corridors, or dominant stands in

which the species attains nearly 100% ground cover [39]. Plants have an affinity for sites that have been disturbed by human activity or flooding. Once established, plants suppress competing species and promote the growth of conspecifics [39].

2.2.2 Experimental data collection

I used the data set collected in Germany by Hüls and coauthors to parameterize my model [39]. I opted to use these data since the difference in the size of the open stand data set and the size of the dense stand data set may be large enough to influence the conclusions of the authors' original paper [20, 79]. Annual transition data for vegetative plants were collected from seventy-six plots located in five sites in Germany between 2002 and 2004 by Dr. Jörg Hüls and coauthors [38, 39]. Sites were located in Allendorf, Druseltal, Frankenberg (Burgwald), Kassel (Dönche), and Viermünden (Table 2.1). I ultimately did not use data collected from Allendorf since the site was disturbed by cattle. Measurements of height, the petiole diameter for the largest leaf, and laminar width of the largest leaf were recorded in 2002 and 2003 for all plants that were 10 cm in height or taller. Data were also collected on individuals in 2004; however, since the data set was originally intended to parameterize a matrix model, plants were classified on site without recording morphological data during data collection in 2004. As a result, I used only the 2002-2003 data sets collected by Hüls and coauthors to parameterize the model.

Table 2.1: Site information for data collected by Hüls and coauthors [39]. The maximum distance between sites is 56 km.

Site	Position		Elevation	Area	Number of Plots	Number of Individuals
	Lat (N)	*Lon (E)*				
Burgwald	51°01'37"	8°45'03"	330m	600	16	216
Dönche	51°17'50"	9°25'22"	290m	4500	18	126
Druseltal	51°18'15"	9°24'49"	335m	30	10	27
Viermünden	51°05'28"	8°49'41"	315m	1400	16	216

2.2.3 Model formulation

The matrix model of Hüls and coauthors consisted of four classes in total: three for vegetative plants classified by size and one for reproductive plants [39]. In order to be able to

compare my model predictions to the prediction made by Hüls and coauthors, I first constructed an integral projection model as similar as possible to the original matrix model. Instead of discretizing vegetative plants based on size I used plant height (cm) as a continuous measure of plant size. Since the species is monocarpic I did not include a separate class for reproductive plants.

If we let $n(x,t)$ be the size distribution of plants at time t with lower and upper bounds L and U, respectively, then the population distribution at the next time step will be governed by the integral equation [24]:

$$n(y,t+1) = \int_L^U K(y,x)n(x,t)\mathrm{d}x \tag{2.1}$$

for an appropriate kernel $K(y,x)$. The kernel may be decomposed into the sum of survival-growth ($P(y,x)$) and fecundity ($F(y,x)$) parts so that [24]:

$$K(y,x) = P(y,x) + F(y,x). \tag{2.2}$$

The survival-growth kernel is the product of the probability a plant of size x is still a vegetative plant the next year multiplied by the probability the plant will grow to size y. The probability a plant remains vegetative may itself be decomposed into the product of surviving to the next year, $p_s(x)$, and the probability the plant does not flower, $1-p_r(x)$, where $p_r(x)$ is the probability a plant of size x flowers [83]. I defined the growth kernel, $G(y,x)$, to be the conditional probability that a plant of height x that remains vegetative will have height y in the next year. Thus the survival-growth kernel may be written as [83]:

$$P(y,x) = p_s(x)[1-p_r(x)]G(y,x). \tag{2.3}$$

The fecundity kernel is the product of the expected number of new plants that survive until the annual census multiplied by their expected size distribution [52]. I defined recruits to be plants that germinated and established earlier in the year of the census that have at least one leaf with a laminar width of 3.5 cm or greater. First-year plants that did not have a leaf with a laminar width of at least 3.5 cm were not included in the census since their survival to the following year was negligible [39]. The expected number of recruits is the

product of the probability a plant of size x survives one more year and the probability that the plant reproduces in that year, $p_s(x)p_r(x)$, multiplied by the expected number of recruits produced by one flowering plant, f_e. I assumed mixing-at-birth for recruit size distribution [13], meaning that the size of the parent plant does not affect recruit size. Under this assumption I denoted recruit size distribution by $c_0(y)$. Altogether, the fecundity kernel may be written as [52]:

$$F(y, x) = p_s(x)p_r(x)f_e c_0(y). \tag{2.4}$$

Thus population dynamics will be described entirely by the integral equation:

$$n(y, t+1) = \int_L^U p_s(x) \left[(1 - p_r(x))\, G(y, x) + p_r(x) f_e c_0(y)\right] n(x, t) \mathrm{d}x. \tag{2.5}$$

2.2.4 Parameterization

Bounds on plant size

The lower and upper bounds of plant size, L and U, must be chosen carefully in integral projection models in order to minimize the number of plants that are needlessly excluded by the model. This phenomenon, known as unintentional eviction, occurs when plants are projected to have a size smaller than the lower bound or larger than the upper bound [27, 101]. Since only plants within the bounds are included in the population vector, plants that are projected to leave the bounds are "evicted" from the population and no longer have an impact on population dynamics.

The lower and upper bounds were initially selected to be the heights of the shortest (10 cm) and tallest (228 cm) plants, respectively. However, plants may be slightly shorter or taller than these bounds and so I extend the bounds to $L = 5$ cm and $U = 250$ cm. I used the process outlined by Williams and coauthors to determine if unintentional eviction affects my results [101].

In my model eviction may occur either in the fecundity kernel or the survival-growth kernel. Eviction through the fecundity kernel was prevented numerically by truncating and normalizing the recruit size distribution, $c_0(y)$. Unfortunately, eviction through the survival-growth kernel is more difficult to mitigate.

The unconditional probability of eviction through the survival-growth kernel is a function defined to be $\rho(x) = p_s(x)\left[1 - \int_L^U G(y,x)\mathrm{d}y\right]$. A maximum of 0.016% of plants may be evicted in open stands while a maximum of 10.3% of plants may be evicted in dense stands. However, the maximum percent of evicted plants is not useful without accounting for the size distribution of plants.

A more meaningful measure of eviction is to take the inner product between the unconditional probability of eviction and the stable size distribution. I first break down the unconditional probability of eviction into the probability of eviction through the lower bound, $\rho_L(x) = p_s(x)\int_{-\infty}^{L} G(y,x)\mathrm{d}y$, and the probability of eviction through the upper bound, $\rho_U(x) = p_s(x)\int_U^{\infty} G(y,x)\mathrm{d}y$. I found the inner product between the probabilities of eviction with the stable size distribution to determine the proportion of plants that are evicted through each bound at the stable size distribution. For open stands, I found that 0.00022% of plants are evicted through the lower bound while 0.00000051% of plants are evicted through the upper bound. For dense stands, I found that 2.13% of plants are evicted through the lower bound and 0.041% of plants are evicted through the upper bound.

I then determined the rate at which eviction will decrease as the bounds change. For open stands, the rate at which eviction decreases as the lower bound decreases is $\mathrm{d}\lambda_L = 9.3 \times 10^{-7}$ and the rate at which eviction decreases as the upper bound increases is $\mathrm{d}\lambda_U = 4.6 \times 10^{-9}$. For dense stands, the corresponding rates of change are $\mathrm{d}\lambda_L = 0.00845$ and $\mathrm{d}\lambda_U = 0.000441$. None of these values are particularly concerning except for $\mathrm{d}\lambda_L$ for open stands.

In order to determine if eviction through the lower bound in dense stands is a problem, I lowered L to the minimum physically possible height of 0 cm and recomputed the population growth rate. Since the population growth rates were identical up to two significant figures, I concluded that a minimum bound of $L = 5$ cm is reasonable to use in the model.

Probability of survival and reproduction

Both the probability of survival and the probability of flowering for the data collected by Hüls and coauthors [39] were fit using logistic regression with height as the explanatory variable (Figure 2.1). For direct comparison to the predictions of the matrix model, the probability of survival and probability of flowering were fit once using only plants from open stands then again using only plants from dense stands.

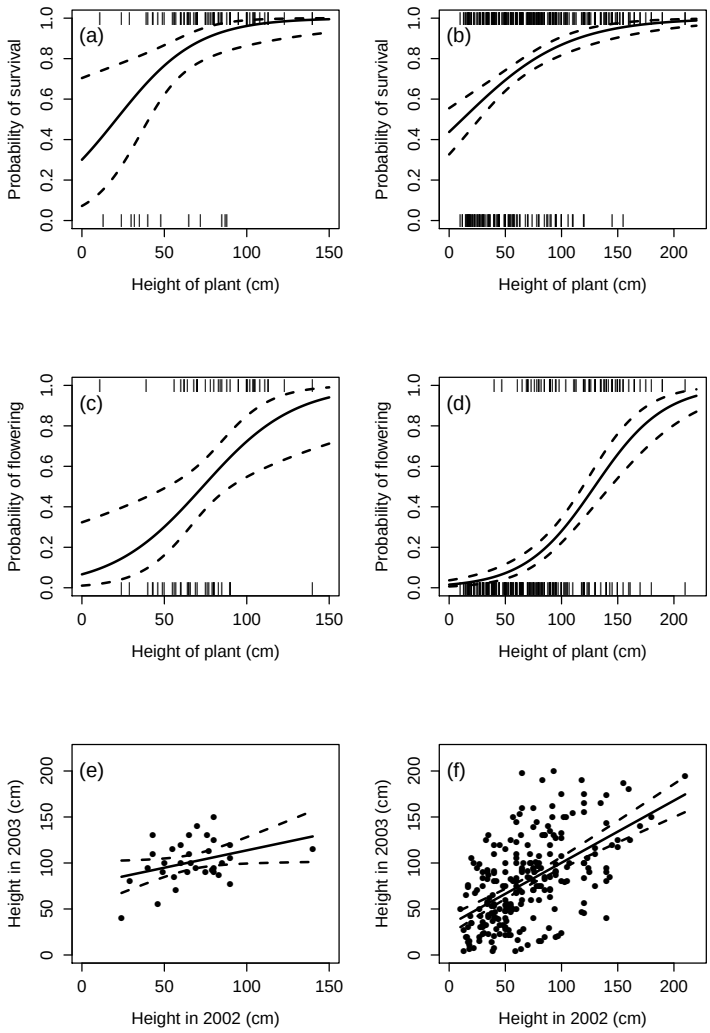

Figure 2.1: Data and best-fit functions for relationships between plant height and vital rates for open and dense stands. Survival probability fit using logistic regression (a, b), probability of reproduction fit using logistic regression (c, d), and expected height of surviving plants fit using linear regression (e, f) in open (a, c, e), and dense (b, d, f) stands of giant hogweed in the Hesse region of Germany. Ticks on the figures fit using logistic regression represent data points with plant height indicated. Plants that survive (a, b) and plants that flower (c, d) have ticks on the top while plants that do not survive (a, b) or do not flower (c, d) have ticks on the bottom. All data collected by Hüls and coauthors [39].

Inter-annual growth

Inter-annual growth was fit for both open and dense stands using linear regression (Figure 2.1). I also fit inter-annual growth using several nonlinear parametric functions and compared these fits to the linear model using the Akaike information criterion (AIC; Table 2.3) [9]. Since weak nonlinearity has been previously shown to significantly affect predicted population growth rate [19], I tested to see if any of the nonlinear models with a lower AIC value yielded different values for predicted population growth rate. In addition to the parametric nonlinear models, I fit a generalized additive model using the gam function from the mgcv package in R [102]. The generalized additive model was fit using default parameters and the results of the IPM that used the GAM growth function was compared to the results of the IPM that assumed the growth function was linear [82].

The suitability of the linear regression was evaluated by testing if the data is homoskedastic and whether residuals are normally distributed. I used the Breusch-Pagan test to test for heteroskedasticity and the Shapiro-Wilk test to check if residuals are normally distributed. Furthermore, in order to determine the impact that nonnormal variance in residuals may have on estimated population growth rate I fit both a linear model and a generalized additive model that each had their error distribution determined using kernel density estimation [82].

Recruitment

The expected number of recruits, f_e, was determined using the same method as Hüls and coauthors [39]. In order to find the expected number of recruits I determined the total number of new plants in each stand type and divided by the total number of flowering plants in the respective stand type. Instead of categorizing these plants by stage as was done in the matrix model, recruit size was assumed to follow a log-normal distribution; however, in order to prevent unintentional eviction I used a truncated log-normal distribution in the numerical implementation. The validity of assuming a log-normal distribution was tested by log-transforming the height of each recruit and using the Shapiro-Wilk test for normality.

Outliers

One plant was removed from the analysis of open stands since its height in 2003 was clearly recorded incorrectly. The plant reportedly decreased in size from 85 cm to 7 cm, yet the

Table 2.2: Vital rate function forms and parameter estimates for the integral projection model. Standard errors for parameter estimates are in parentheses. All data collected by Hüls and coauthors [39].

Function	Stand type	Fitted function
Survival (p_s)	Open	$\text{Logit}(p_s(z)) = -0.845_{(0.873)} + 0.041_{(0.014)}z$
	Dense	$\text{Logit}(p_s(z)) = -0.251_{(0.241)} + 0.021_{(0.004)}z$
Reproduction (p_r)	Open	$\text{Logit}(p_r(z)) = -2.645_{(0.973)} + 0.036_{(0.012)}z$
	Dense	$\text{Logit}(p_r(z)) = -4.156_{(0.451)} + 0.032_{(0.004)}z$
Growth (G)	Open	$G(z_1, z) = 75.96_{(12.64)} + 0.38_{(0.18)}z + \mathcal{N}(0, 22.15^2)$
	Dense	$G(z_1, z) = 32.70_{(5.25)} + 0.68_{(0.07)}z + \mathcal{N}(0, 38.40^2)$
Mean recruits (f_e)	Open	$f_e = 0.62$
	Dense	$f_e = 0.92$
Recruit size (c_0)	Open	$\ln(z_1) \sim \mathcal{N}(4.17, 0.28^2)$
	Dense	$\ln(z_1) \sim \mathcal{N}(3.78, 0.56^2)$

plant remained a stage 2 individual (>50 cm) with a laminar width of 61 cm for its largest leaf. In addition to this record, one outlier was removed from the analysis of dense stands since it had an unusually large height increase from 32 cm to 228 cm in one year. When growth was fit the standard deviation of the residuals was 37 cm while the residual for this point was 155 cm. Since its growth was more than four standard deviations larger than the mean increase I concluded height was recorded incorrectly for this plant as well and removed it from the data set for dense stands.

2.2.5 Life cycle analysis

The population growth rate for each model was determined by finding the dominant eigenvalue of the demographic kernel, defined to be the largest value of λ that satisfies [24]:

$$\lambda w(y) = \int_L^U K(y, x) w(x) \mathrm{d}x \tag{2.6}$$

for some function $w(x)$ called the right eigenvector. The eigenvalue and right eigenvector were computed numerically by first using the midpoint rule to produce a matrix equation,

$n(t+1) = An(t)$, to approximate the IPM. The code for this approximation was extensively modified from code provided by Rees and coauthors [82] and may be found on UWSpace. I then found the largest λ and its corresponding right eigenvector w that satisfy the equation $\lambda w = Aw$.

A unique population growth rate and corresponding right eigenvector are guaranteed since the kernel is power positive [27]. The dominant right eigenvector of the IPM represents the stable size distribution of the population which is the size distribution the population will converge to over time. In addition to the right eigenvector, there is a left eigenvector corresponding to the dominant eigenvalue that satisfies:

$$\lambda v(x) = \int_L^U K(y,x)v(y)\mathrm{d}y. \tag{2.7}$$

The dominant left eigenvector is also guaranteed to exist since the kernel is power positive and the dominant left eigenvector provides the relative reproductive values for the population [26]. The relative reproductive values are a measure of how a plant of a given size will impact future population size.

2.2.6 Bootstrap confidence intervals

I used bootstrapping to compare between different values for population growth rate and to compare mean observed plant height to the mean height predicted by the IPM. For population growth rate and mean predicted height, I computed 5,000 bootstrap samples for both the open stand data set and the dense stand data set. Each bootstrap sample was used to reparameterize the integral projection model, which was then used to determine the bootstrap statistics, $\bar{h}_b$ and λ_b. The bootstrap 95% confidence intervals were given by the 2.5% and 97.5% quantiles of the bootstrap distribution. It should be noted that any bootstrap sample that had two or fewer recruits were discarded and another sample drawn in its place. This requirement was necessary to ensure the recruit size distribution could be determined for the IPM. Bootstrapping for mean observed height was performed at the same time as population growth rate.

2.3 Results

2.3.1 Results of statistical analysis

Inter-annual growth

I found that the linear model yielded the second lowest AIC value for the dense stand data set and the fifth lowest AIC value for the open stand data set. However, the differences in AIC values between the linear model and nonlinear models were small enough to not be of much concern ($\Delta\text{AIC} < 4$; Table 2.3). Furthermore, I found that the estimates of population growth rate provided by the nonlinear models were identical to the estimates of population growth rate provided by the linear model up to two significant figures. Since the potential presence of nonlinearity did not affect population growth rate and the quality of the fit (as measured by AIC) was only slightly improved by including nonlinearity I decided to use the linear model for growth.

Table 2.3: Evaluation of the linear and nonlinear functions used to model inter-annual growth of giant hogweed in Germany [39]. The best performing fit for each data set is in bold.

Form	df	AIC	
		Open stands	*Dense stands*
$y = ax + b$	2	302	2416
$y = ax^2 + bx$	2	303	2425
$y = ax^2 + bx + c$	3	302	2417
$y = axe^{-bx}$	2	299	2421
$y = a(1 - e^{-bx})$	2	**298**	2420
$y = ax/(b + x)$	2	299	2418
$y = ax^b$	2	300	**2415**

The Breusch-Pagan test failed to reject the hypothesis that the data is homoskedastic in both open stands ($p = 0.23$) and dense stands ($p = 0.074$). The Shapiro-Wilk test failed to reject the hypothesis that residuals are normally distributed for open stands ($p = 0.84$); however, the test does reject the hypothesis that residuals are normally distributed for dense stands ($p = 0.016$). The predicted population growth rate for populations in dense stands was determined in three ways: using linear regression for growth with normally distributed errors, using linear regression for growth with the error distribution found using kernel

density estimation, and using a generalized additive model with the error distribution found using kernel density estimation. The predicted population growth rates found using each of the three methods were identical up to two significant figures. Therefore, I concluded that the growth kernel may be represented by a Gaussian conditional probability distribution with mean scaling linearly with height and constant variance.

Recruitment

I log-transformed the heights of all seedlings and tested the normality of the distribution using the Shapiro-Wilk test. The Shapiro-Wilk test yielded $p = 0.68$ for recruits in open stands, $p = 0.77$ for recruits in dense stands, and $p = 0.58$ for recruits in the combined data set. These results indicate that I cannot reject the hypothesis that recruit size follows a log-normal distribution. However, I note that the high p-value for open stands may be due to the low sample size which makes the null hypothesis unlikely to be rejected. A histogram of log-transformed recruit height in open stands indicates the distribution may be uniform (Figure 2.2); however, since the sample size is low and the recruit size distribution in dense stands appears to be log-normal (Figure 2.2), I assumed that the distribution in open stands is log-normal as well.

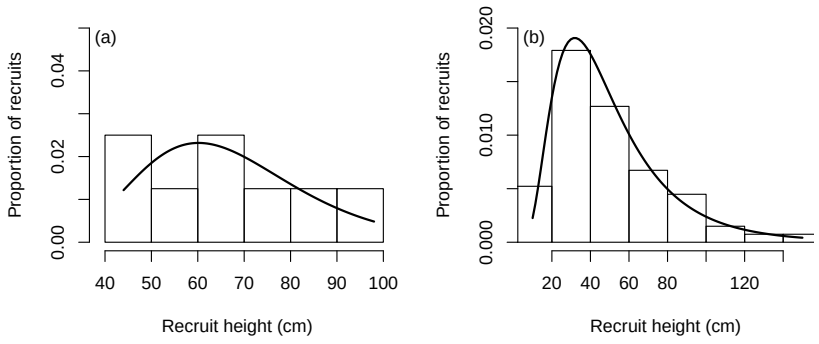

Figure 2.2: Histogram of the observed distribution of recruit size and fitted log-normal distribution for open (n=8, a) and dense (n=67, b) stands of giant hogweed in the Hesse region of Germany. All data collected by Hüls and coauthors [39].

2.3.2 Results of demographic analysis

The population growth rates I calculated from the integral projection model are much less than one, like those found by Hüls and coauthors using a matrix model [39], and indicate that populations in both open and dense stands are in decline (Table 2.4). Similar to the results of the matrix model, the bootstrap confidence intervals for population growth rate in open and dense stands overlap. This overlap indicates that the difference in population growth rates between the two stand types is not statistically significant.

Table 2.4: Population growth rate (λ), bootstrap estimate of population growth rate (λ_b), and bootstrap 95% confidence interval (CI-, CI+). Bootstrap values were computed using 5,000 bootstrap samples. All data collected by Hüls and coauthors [39].

Stand type	Model	N	λ	λ_b	(CI-, CI+)
Open	Matrix	103	0.76	0.77	(0.55, 1.05)
	IPM	103	0.69	0.72	(0.45, 1.13)
Dense	Matrix	554	0.75	0.75	(0.70, 0.81)
	IPM	554	0.73	0.73	(0.67, 0.79)

The large confidence interval for open stands in consistent with the results of Ramula and coauthors [79]. To determine if sample size affects my results, I performed bootstrapping with larger sample sizes for each replicate (Table 2.5). As sample size increased, the size of the confidence intervals decreased. However, even with the increased sample size I conclude that there is no significant difference in population growth rate between open and dense stands.

The stable size distributions for open and dense stands were computed and compared to the observed size distributions (Figure 2.3). The mean values for plant height from the stable size distributions are 86 cm for open stands and 77 cm for dense stands. In contrast, the mean heights determined from data recorded in 2003 are 95 cm for open stands and 73 cm for dense stands. The bootstrap confidence intervals for the predicted and observed values of height overlap in both open and dense stands (Table 2.6). Therefore, the mean heights at stable size distribution are not significantly different than their corresponding observed heights. These results are in contrast to the matrix model which determined that the mean predicted height for open stands is significantly higher than the mean observed height [39].

Table 2.5: Population growth rate (λ), bootstrap estimate of population growth rate (λ_b), and bootstrap 95% confidence interval (CI-, CI+) as the bootstrap replicate sample size (N) increases. Bootstrap values were computed using 5,000 bootstrap samples. All data collected by Hüls and coauthors [39].

Stand type	N	λ	λ_b	(CI-, CI+)
Open	100	0.69	0.72	(0.44, 1.15)
	200	0.69	0.70	(0.50, 0.96)
	300	0.69	0.70	(0.53, 0.91)
	400	0.69	0.70	(0.55, 0.88)
	500	0.69	0.69	(0.56, 0.85)
	600	0.69	0.69	(0.57, 0.84)
Dense	100	0.73	0.74	(0.59, 0.92)
	200	0.73	0.73	(0.63, 0.86)
	300	0.73	0.73	(0.65, 0.83)
	400	0.73	0.73	(0.66, 0.81)
	500	0.73	0.73	(0.67, 0.81)
	600	0.73	0.73	(0.67, 0.79)

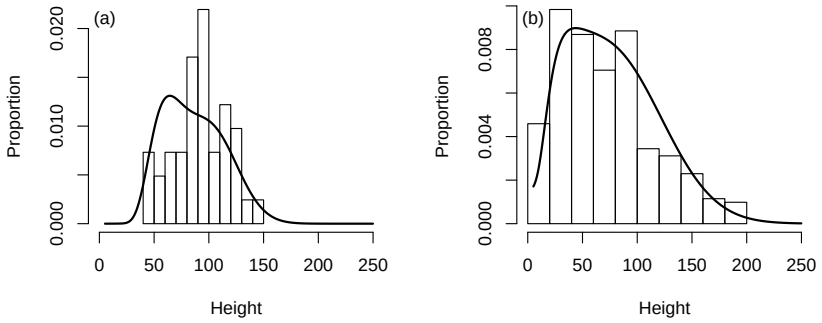

Figure 2.3: Histogram of observed size distribution in 2003 vs predicted stable size distribution for open (a) and dense (b) stands. All data collected by Hüls and coauthors [39].

Table 2.6: Mean plant height ($\bar{h}$), bootstrap mean plant height ($\bar{h}_b$), and bootstrap 95% confidence intervals (CI-, CI+). Bootstrap values were computed using 5,000 bootstrap samples. All data collected by Hüls and coauthors [39].

Stand type	Source	$\bar{h}$	$\bar{h}_b$	(CI-, CI+)
Open	Observed	95	95	(87, 103)
	Predicted	86	87	(76, 98)
Dense	Observed	73	73	(68, 78)
	Predicted	77	78	(73, 84)

2.3.3 Impact of outliers

I re-parameterized the integral projection model using both the open and dense stand data sets without outliers to see how the exclusion of these plants affect model predictions. In both cases the population growth rate was unaffected by the removal of outliers up to two significant figures.

2.4 Discussion

In this chapter I developed a density-independent integral projection model for giant hogweed and parameterized it using two different data sets collected by Hüls and coauthors [39]. I found that the decision to categorize plants into discrete stages based on size had no significant impact on the conclusions of Hüls and coauthors and therefore did not affect the authors' suggested management priorities. This result does not support the hypothesis that model discretization significantly affects predicted population growth rate [20, 79].

Several authors have concluded that matrix models will not be biased for large demographic data sets [20, 79]. The threshold for this bias to become negligible is dependent on both the life history of the plant and how many plants were sampled. Ramula and coauthors determined that for the monocarpic perennial *Cirsium palustre* both the matrix model and the integral projection model are unbiased for a data set consisting of more than 300 records, resulting in nearly identical predicted values for population growth rate [79]. This claim is supported by my result that the matrix model and IPM do not produce significantly different values for population growth rate in dense stands with numerous individuals ($n = 554$ plants). However, due to the large size of the confidence intervals,

my analysis does not support the conclusion made by Ramula and coauthors that bias in estimated population growth rate becomes significant for smaller data sets.

When parameterized using a small demographic data set ($n = 103$ plants), the bootstrap distributions derived from each model have large variance [79]. The large variance in each estimate for population growth rate makes it difficult to derive strong conclusions from either model. Ramula and coauthors reported that, for small demographic data sets (fewer than 300 plants), IPMs have lower variance than matrix models in estimated population growth rate [79]. However, I did not observe this result for the open stand data set. For both the matrix model and the IPM the bootstrap confidence interval for dense stands is entirely within the bootstrap confidence interval for open stands.

I conclude that for both open and dense stands the management priorities recommended by Hüls and coauthors are unaffected by their decision to use a matrix model [39]. Specifically, the population growth rates predicted by the IPM support the authors' observation that open stands do not have a higher intrinsic population growth rate than dense stands.

The results of this chapter support the conclusion made by Hüls and coauthors that the populations growth rate in open stands is not significantly higher than population growth rate in dense stands [39]. The lack of a significant difference between open and dense stands can be contrasted with barbed goatgrass (*Aegilops triuncialis*), a grass invasive in North America that has been shown to have similar growth rates between 'core' and 'edge' populations [92]. Unlike giant hogweed, the 'core' and 'edge' populations of barbed goatgrass are caused by soil conditions rather than management. Due to the rocky soil conditions, spikes of barbed goatgrass in edge populations have an unusual ability to move long distances and colonize new habitats. In contrast to giant hogweed, edge populations of barbed goatgrass represent the invasion front and should be prioritized for control.

More similar to giant hogweed, some invasive species have two distinct subpopulations caused by control measures. For example, invasive lionfish species (*Pterois volitans* and *Pterois miles*) in the Caribbean are controlled by culling; however, due to diving limitations only reefs up to 30 m deep may be targeted for control [5]. This inability to cull populations in mesophotic reefs creates two subpopulations, one with low population density and one with high populations density, similar to open and dense stands of giant hogweed. Andradi-Brown and coauthors found that the ability for lionfish to quickly repopulate shallow reefs post-culling undermined management efforts. The authors concluded that enhancing management efforts to include culling in mesophotic reefs is necessary to control the species. Although giant hogweed is sessile and thus will replenish managed sites more slowly, expanding management to dense stands may be necessary to control the species.

Even though my results support the conclusions of Hüls and coauthors [39], they dispute the applicability of the findings of Ramula and coauthors towards the development of management regimes for invasive or endangered populations [79]. Ramula and coauthors concluded that small demographic data sets, such as the open stand data set, will have a biased population growth rate which may in turn affect management decisions. However, I found that the small size of the open stand data set makes it difficult to find significant differences in population growth rate. These results suggest that the decision to use an integral projection model may not reduce the difficulty in drawing conclusions from small demographic data sets in general.

The strength of the integral projection model approach is the use of continuous vital rate functions to construct the kernel. Unfortunately, this regression-based approach to describe population dynamics is dependent on the how well the vital rate functions approximate reality. I assumed that linear regression with normally distributed residuals would adequately describe plant growth. I tested these assumptions in section 2.2.4 and found that the fit could be improved upon by adding nonlinearities or by allowing residual errors to follow a different distribution.

Dahlgren and coauthors demonstrated that nonlinearities in the fitted growth function could have a significant impact on predicted population growth rate [19]. To address this concern I parameterized the IPM with each nonlinear model that had a lower AIC value than the linear model as well as a generalized additive model [82]. The population growth rates predicted by each of these models did not differ significantly from the population growth rate predicted by the IPM fit with linear growth. Similarly, the linear model with non-normal error distribution and generalized additive model with non-normal error distribution did not yield different predictions for population growth rate than the linear model used throughout this chapter.

Although predicted population growth rate was robust to changes in the growth fit and the error distribution, the selection of the linear fit with normal errors could affect other results of this chapter. The stable size distributions and the bootstrap confidence intervals for predicted population growth rate are both dependent on the form used for growth. The use of a nonlinear function to describe growth or relaxing the assumption that errors are normally distributed may affect the conclusions in this chapter; however, further investigation of the impact of growth fit is beyond the scope of this chapter.

In addition to the potential issue with growth, the fit for recruitment may be problematic. Since only eight recruits were present in open stands in 2003 [39], there is structural uncertainty in the recruit size distribution which may bias the estimate of population growth rate or stable size distribution. I used a log-normal distribution for both data sets

since it was a good fit for recruit size distribution in dense stands and since it is a commonly used distribution to describe recruit size [55]. However, the recruit size distribution may have a different form in open stands due to the difference in competitive pressure.

The uncertainty in recruit size is not the only issue with recruitment. In addition to the low number of recruits and uncertainty in recruit size distribution, there is uncertainty in how the number of flowering plants relates to the expected number of recruits. In particular, there were a few plots in the Hüls data set in which recruits were recorded in 2003 yet no flowering plants were recorded in 2002. These seeds may have originated from flowering plants just outside the plot or may have come from the seed bank [51, 57]. In either case, due to the unknown origin of these recruits we cannot expect the number of flowering plants to strongly correlate with the number of recruits in each plot.

As in the matrix model, the IPM predicts that populations are in decline. This decline was explained by Hüls and coauthors as being caused by drought conditions [39]. Hüls and coauthors found that populations rebounded in the 2004 growing season under normal weather conditions with the matrix model predicting population growth rates well above 1. Unfortunately, it is not possible to find the corresponding population growth rate using the IPM since plants were classified on site during the 2004 growing season. This dependence of population growth rate on realized weather conditions highlights the importance of multi-year studies of invasive populations since without the 2004 census Hüls and coauthors may have underestimated the risk of invasion by giant hogweed.

The low predicted population growth rates for giant hogweed due to the 2003 drought is not unusual; however, it is also common for invasive species to benefit from unusual weather conditions. Jackson reports that invasion of California grasslands by annual Mediterranean grasses was facilitated by drought conditions [41]. The author's hypothesis was later supported by a mechanistic resource availability model developed by Evarard and coauthors that finds that non-native annual grass species, such as great brome (*bromus diandrus*), competitively exclude perennial grasses native to Californian grasslands under drought conditions [31]. Subsequent research by Kimball and coauthors found that drought slowed recovery of Californian shrubland during postfire succession which may allow for the conversion of shrubland to grassland by non-native annual grasses [46].

The use of an integral projection model to estimate population growth rate for giant hogweed populations in this chapter indicates that open stands do not have a larger growth rate than dense stands, supporting the conclusions of Hüls and coauthors [39] that open stands are suppressed by biotic factors such as mowing or grazing. Furthermore, I found a practical limitation to the results of Ramula and coauthors [79] since the variance in bootstrap estimates for population growth rate is correlated with the magnitude of bias in

population growth rate. The high variance makes it difficult for significant differences to be reported for small demographic data sets. This limitation is exacerbated by the difficulty in computing bootstrap statistics using an IPM with only a few recruits.

Chapter 3

The Impact of Density-Dependent Flowering on *H. mantegazzianum* Seed Production

3.1 Introduction

The growth of a plant population must necessarily occur in an environment with finite resources. As a result of this resource limitation, the vital rate functions that govern plant population dynamics will vary as population density changes. Proper understanding of the transient and long-term dynamics of a population must therefore account for the impact of intraspecific competition on plant survival, growth, reproduction, fecundity, germination, and establishment. Although competition is important to account for, not each of these vital rates will have a significant impact on spread and their inclusion may needlessly complicate the model.

In the previous chapter I used a density-independent integral projection model to compare giant hogweed populations that have been classified as either open or dense stands. This classification was based on the level of interspecific competition giant hogweed plants face [39]. However, it has been reported by Pergl and coauthors that flowering is delayed in the presence of conspecifics [73] and it has been reported by Hüls and coauthors that competition from adult conspecifics results in longer generation times and a different size distribution [39]. In this chapter I extend the integral projection model to account for intraspecific competition. Before I describe how this is done, I must review previously published density-dependent models for giant hogweed.

Previous models for giant hogweed population dynamics have either ignored intraspecific density-dependence [39, 73], assumed there is a ceiling capacity below which populations are unaffected [63], or used a ceiling capacity with a smoothing function for survival [56]. However, the imposition of a ceiling capacity is not justified and will affect population dynamics. In particular, the predicted number of seeds produced will differ if a different form of density-dependence is used in the model. This difference in predicted seed set may have implications on the rate of spread of the species [68].

The most common form of density-dependence in integral projection models is recruitment limitation [81]. Recruitment limitation is most commonly modelled since small plants are more vulnerable than adult plants; however, another factor contributing to the focus on recruitment limitation is the relative ease of analysing models with only recruitment limitation when compared to models that account for adult density [81, 22, 23, 21]. The dearth of IPMs that allow for competition among adult conspecifics provides additional motivation for this chapter. This chapter will therefore have two models, one that only accounts for competition among recruits and a second that also includes adult competition.

Each recruit will compete for available microsites and resources in order to establish. I incorporate recruitment limitation in each of the models presented in this chapter. I then extend one of the models to allow for adult plants to compete with one another. This competition may result in reduced survival, growth, or probability of reproduction as adult plants compete for resources [98]. It will be determined during parameterization which density-dependent effects must be accounted for. However, if any of the aforementioned vital rates are affected by density then the total seed set may be affected.

It has previously been shown that the total seed set has an impact on the rate of spread of species, particularly when habitats are fragmented [68]. The rate of spread for giant hogweed is highly variable and sensitive to long-distance dispersal events that colonize disturbed sites [59, 72, 56]. The rate of spread is also dependent largely on the dispersal of seeds along corridors such as roads and waterways [93]. The accurate prediction of seed production in giant hogweed populations is therefore essential for realistic predictions of the spread of the species.

However, Pergl and coauthors [73] reported that flowering of giant hogweed plants is delayed in stands with high population density, which will in turn affect the seed production. Therefore, I expect that a negative relation between population density and the probability of flowering may initially result in higher population growth rates if there are significantly more seeds produced immediately following invasion as compared to when the populations have reached carrying capacity. I also expect that population growth rates over time will be lower if there are fewer seeds once populations have reached carrying capacity.

In this chapter I test this hypothesis using a structured population model of giant hogweed that includes negative density-dependence in two different forms; the first form is competition between seedlings for available microsites while the second form is the negative relation between the probability of flowering and population density. I compare the seed number predicted by the model with and without the delay in flowering and find that significantly fewer seeds are produced over time when the delay in flowering is modelled. Additionally, I find the equilibria for each model and the dominant eigenvalues for their Jacobians in order to determine if the course of invasion differs qualitatively. These differences in predicted seed production may in turn predict a different rate of spread for the species when competition among adult plants is accounted for.

3.2 Methods

3.2.1 Study species

Giant hogweed is a monocarpic perennial herb which may grow up to 5.5 m tall [69]. The species is native to the Caucasus and has successfully invaded much of Europe and North America [69]. The invasiveness of giant hogweed is bolstered by the extremely large seed set consisting of thousands or tens of thousands of seeds which may disperse along roads and waterways [56]. Adult plants outshade competitors and may potentially produce allelopathic substances to encourage the establishment of their offspring [100]; however, the large seed set left behind leads to high intraspecific competition among seedlings. Populations are typically not seed limited, but in patchy environments seed limitation has the potential to impact invasion speed [68].

3.2.2 Experimental data collection

I used the annual transition data collected in Germany between 2002 and 2003 by Dr. Jörg Hüls and coauthors [39] to parameterize adult population dynamics (Table 2.1). Hüls and coauthors collected measurements of height, the petiole diameter for the largest leaf, and laminar width of the largest leaf for vegetative plants in each year. As was done in the previous chapter, I used height as the measure of plant size.

Mean seed production was also recorded in the original study; however, seed production differed greatly from most previously published estimates [66, 74, 77, 94], yet agreed with estimates from giant hogweed populations in Ireland [10, 75]. However, it is speculated

that the estimates of seed production from the study in Ireland are overestimated due to the author estimating fruit number by counting both male and female flowers [75]. Rather than use the estimates produced by Hüls and coauthors I estimated mean seed production using a meta-analysis from the literature. Mean seed production was estimated from the weighted mean of estimates from four sources from the literature (Table 3.1, excluding Caffrey).

Monthly seedling survival data were collected in the original study by Hüls and coauthors [39]. Plots with an area of 0.1 m^2 were saturated with seeds so that further addition of seeds would not yield more seedlings (Jörg Hüls, personal communication). Hüls and coauthors recorded the number of surviving seedlings each month. I used the average number of seedlings measured in July as the maximum number of seedlings per plot. However, since plots were saturated with seeds I needed additional data to parameterize a recruitment function with few seeds.

To finish parametrizing the model, Grguric and Cuddington set up 3 sites in Ontario with 4 quadrants each to estimate germination (Table 3.2). Fifty seeds were sowed in each plot in Fall of 2016 and the number of resulting seedlings recorded in Spring of 2017.

Table 3.1: Estimates of seed production from the literature.

Author(s)	Location	Mean seeds	N	Source
Caffrey	Ireland	41202	80	[10]
Ochsmann	Germany	9695	33	[66]
Perglova	Czech Republic	20671	98	[74]
Pysek	Czech Republic	16140	8	[77]
Tiley, Philp	Scotland	15729	4	[94]
Weighted mean (excluding Caffrey)		17746		

3.2.3 Towards a density-dependent integral projection model

State variables

In order to account for density-dependence I first needed mathematical definitions for density. For competition between recruits I used the total number of seeds at time t, $n_s(t)$, while for adult competition I used the total biomass in the plot, $b(t)$. The decision to use total plot biomass as a measure of competition was based on previous studies

Table 3.2: Seedling germination data collected in Ontario.

Location	2016 census	2017 census	Quadrant	2016 seeds	2017 seedlings
Airport Rd	Oct 18	Apr 28	1	50	33
			2	50	28
			3	50	26
			4	50	23
Hwy 9	Oct 18	Apr 28	1	50	10
			2	50	21
			3	50	22
			4	50	12
Woodstock	Oct 19	May 12	1	50	4
			2	50	5
			3	50	6
			4	50	3

that show that the relative growth rates of target plants are significantly affected by the biomass of neighbouring plants [32, 33, 34, 80]. Since flowering plants will be competing for resources I needed to track the size distribution of both vegetative and flowering plants to determine total biomass. I define $n_v(x,t)$ to be the size distribution of vegetative plants at time t, $n_r(x,t)$ to be the size distribution of reproductive plants at time t, and $n(x,t) = [n_v(x,t), n_r(x,t)]$ to be the total state of the population at time t.

I assumed that the number of seeds produced is proportional to the number of flowering plants. Previous authors have observed that the number of flowers will increase with plant size; however, the majority of the excess flowers do not produce fruit [75]. Let $r_t(t)$ be the total number of flowering plants at time t. Then $r_t(t)$ may be found by integrating over the size distribution of flowering plants, $r_t(t) = \int_L^U n_r(x,t)\mathrm{d}x$, where L and U are the lower and upper bounds for individual size, respectively. The total number of seeds at time t may then be expressed as an integral, $n_s(t) = m_s r_t(t) = \int_L^U m_s n_r(x,t)\mathrm{d}x$, where m_s is the mean number of seeds a reproductive plant produces (Table 3.1). This mean number of seeds produced by each plant is assumed to be independent of plant density [73] and independent of plant size [75].

The measure of adult competition used in the model was necessarily a function of plant heights within each plot. This restriction was due to limitations in the data set collected by Hüls and coauthors [39] and due to height being the only state variable measured in the

model. I assumed that competition between adults is size-symmetric and dependent on the total biomass of the population [87, 97, 98]. Unfortunately, measurements of biomass were not collected by Hüls and coauthors. Instead, the total biomass of adult plants in the population was estimated using the allometric scaling law of West, Brown, and Enquist [28, 29, 30, 84, 99]. This law states that the biomass of a terrestrial plant, b_0, scales with its height, x_0, to the fourth power. I applied this relation to vegetative and reproductive plants at each height and integrated in order to find the total biomass at time t:

$$b(t) = \int_L^U x^4[n_v(x,t) + n_r(x,t)]\mathrm{d}x. \tag{3.1}$$

The scaling law of West, Brown, and Enquist has previously been criticized for making assumptions that do not necessarily hold for every plant species [49, 50]. Instead, other authors give a range between 3 and 6 for the exponent [3]. I reparameterized the model using the values of 3 and 6 for the scaling exponent and reran the simulations in the results section to determine if this uncertainty in the allometric scaling law affects my results. These findings are briefly mentioned in the discussion.

Vital rate functions

Excluding immigration, there are three processes that describe giant hogweed population dynamics: the survival and growth of vegetative plants, the transition of vegetative plants to their reproductive state, and the production of recruits by the reproductive plants. These processes of survival-growth, reproduction, and fecundity may be governed by integral kernels similar to the density-independent integral projection model from chapter 2.

In the recruit-limited model I assumed that the number of recruits is a monotonically increasing, bounded function of the number of number of seeds produced in the plot. The fecundity kernel, F, is therefore a function of the number of seeds, n_s. Specifically, the function that gives the number of surviving recruits has a Michaelis-Menten form [22]. The survival-growth and reproduction kernels are assumed to be density-independent as in chapter 2. The recruit-limited model has the form:

$$n_v(y,t+1) = \int_L^U P(y,x)n_v(x,t)\mathrm{d}x + \int_L^U F(y,x,n_s)n_r(x,t)\mathrm{d}x \tag{3.2}$$

$$n_r(y,t+1) = \int_L^U R(y,x)n_v(x,t)\mathrm{d}x, \tag{3.3}$$

where $P(y,x)$ is the survival-growth kernel, $R(y,x)$ is the reproduction kernel, and $F(y,x,n_s)$ is the fecundity kernel.

In the adult competition model I allowed the fecundity kernel to depend on the total number of seeds. However, this model differs from the recruit-limited model in that I allowed the functions describing survival, growth, and the probability of reproduction to depend on total plant biomass. The general form of the adult competition IPM is:

$$n_v(y,t+1) = \int_L^U P(y,x,b)n_v(x,t)\mathrm{d}x + \int_L^U F(y,x,n_s)n_r(x,t)\mathrm{d}x \tag{3.4}$$

$$n_r(y,t+1) = \int_L^U R(y,x,b)n_v(x,t)\mathrm{d}x, \tag{3.5}$$

where $P(y,x,b)$ is the survival-growth kernel, $R(y,x,b)$ is the reproduction kernel, and $F(y,x,n_s)$ is the fecundity kernel.

Determining the integral kernels

The survival-growth kernel for the adult competition model has the same form as survival-growth kernels in other IPMs for monocarpic perennials [14], albeit with vital rates dependent on adult plant biomass:

$$P(y,x,b) = p_s(x,b)[1-p_r(x,b)]G_v(y,x,b), \tag{3.6}$$

where $p_s(x,b)$ is the probability a plant of size x in a plot of biomass b survives to the next census, $p_r(x,b)$ is the probability a plant of size x in a plot of biomass b reproduces before the next census, and $G_v(y,x,b)$ is the probability a vegetative plant of size x in a plot of biomass b that remains vegetative grows to size y in the next census. The survival-growth kernel for the recruit-limited model is identical except for the lack of dependence

on biomass.

The reproduction kernel for the adult competition model is the probability a vegetative plant survives, reproduces, and grows to a given size:

$$R(y, x, b) = p_s(x, b)p_r(x, b)G_r(y, x, b), \tag{3.7}$$

where $G_r(y, x, b)$ is the probability a plant of size x in a plot of biomass b that reproduces grows to size y the next census. The reproduction kernel for the recruit-limited model is identical except for the lack of dependence on biomass.

Lastly, I found the fecundity kernel for both the recruit-limited model and the adult competition model. This kernel was found by first integrating over the fecundity kernel multiplied by the distribution of reproductive plants. This integral must be equal to the total number of recruits, $f_e(n_s)$, multiplied by the recruit size distribution, $c_0(y)$. Therefore $\int_L^U F(y, x, n_s(t))n_r(x, t)\mathrm{d}x = g_e(n_s(t))n_s(t)c_0(y) = \int_L^U m_s g_e(n_s(t))c_0(y)n_r(x, t)\mathrm{d}x$, where $g_e(n_s)$ is the probability a seedling establishes when n_s seeds are produced. This equality implies that:

$$F(y, x, n_s) = m_s g_e(n_s(t))c_0(y). \tag{3.8}$$

3.2.4 Parameterization

Bounds on plant size

The lower bound, L, was chosen to be the same as the lower bound in the density-independent IPM, $L = 5$ cm. I selected the upper bound of $U = 550$ cm since that value is a commonly reported upper bound for flowering giant hogweed plants [78, 69].

Probability of survival and reproduction

I assumed that the probability of survival and the probability of reproduction were both density-independent functions for the recruit-limited model. I used logistic regression with height as a covariate to fit these vital rate functions. However, this assumption is not necessarily valid since intraspecific competition for resources may affect plant survival or

the timing of reproduction. Therefore, for the adult competition model I fit the probability of survival and probability of reproduction using logistic regression with total plot biomass as a covariate in addition to plant height.

I compared the AIC values of the density-independent and density-dependent fits in order to determine if the vital rates in the adult competition model should depend on adult biomass. I found that the probability of survival had no improvement in fit when biomass was accounted for (Table 3.3). I therefore used the same density-independent function as was used in the recruit-limited model for the probability of survival. However, including biomass as a covariate in regression for the probability of reproduction did yield a significantly better fit than the density-independent fit (ΔAIC= 13, $p = 0.013$ from the likelihood ratio test). Therefore, for the adult competition model I used a function for the probability of reproduction that depends on the biomass of neighbouring plants (Table 3.4).

Table 3.3: Impact of biomass and site-specific effects on fits for vital rate functions for giant hogweed. Vital rates were each fit twice - the first fit does not use biomass as a covariate in regression while the second fit does use biomass as a covariate. The best performing fit for each data set is in bold. All data collected by Hüls and coauthors [39].

Vital Rate	Density-dependent?	df	AIC
Survival (p_s)	No	2	**483**
	Yes	3	**483**
Reproduction (p_r)	No	2	383
	Yes	3	**370**
Growth of vegetative plants (G_v)	No	2	**2739**
	Yes	3	**2739**
Growth of reproductive plants (G_r)	No	2	1117
	Yes	3	**1115**

Table 3.4: The final forms of the vital rate functions. All functions were fit using data collected by Hüls and coauthors [39]. RL=recruit-limited and AC=adult competition. Standard error is indicated in parentheses. The Michaelis constant for mean recruits was fit using data collected by Cuddington and Grguric.

Function	Notes	Fitted function
Survival (p_s)		$\text{Logit}(p_s(x)) = -0.284_{(0.235)} + 0.023_{(0.004)}x$
Reproduction (p_r)	RL	$\text{Logit}(p_r(x)) = -3.340_{(0.349)} + 0.028_{(0.004)}x$
	AC	$\text{Logit}(p_r(x,b)) = -2.889_{(0.370)} + 0.030_{(0.004)}x - 0.003_{(0.0008)}b$
Growth (G)	V	$G_v(y,x) = 36.68_{(4.96)} + 0.66_{(0.06)}x + \mathcal{N}(0, 37.52^2)$
	R	$G_r(y,x) = 217.29_{(10.85)} + 0.60_{(0.09)}x + \mathcal{N}(0, 36.30^2)$
Mean recruits (f_e)		$f_e(n_s) = \frac{127.5}{n_s+273.4}n_s$
Recruit size (c_0)		$\ln(y) \sim \mathcal{N}(3.82, 0.55^2)$

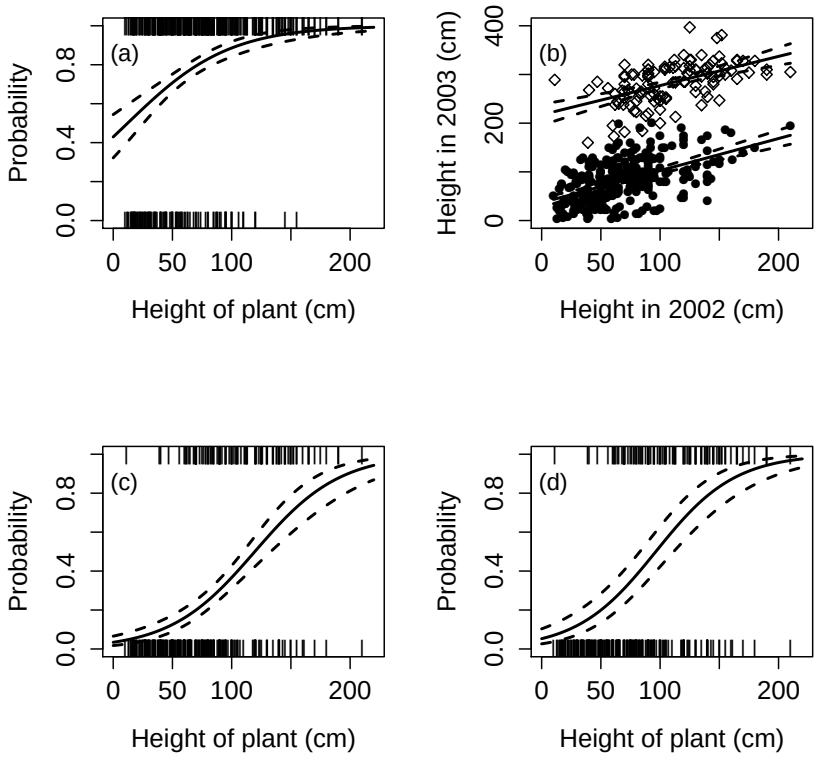

Figure 3.1: Data and best-fit functions for relationships between plant height and vital rates. Survival probability (a), expected height of vegetative (b, black dots) and reproductive plants (b, unfilled rhombi), and probabilities of reproduction for the recruit-limited model (c) and adult competition model with zero biomass (d). All data collected by Hüls and coauthors [39].

Inter-annual growth

I fit seven common functional forms to describe the growth of vegetative and reproductive plants [70]. I then compared each fit using their AIC values to determine which form to use. I found that the discrete logistic and power law fits had a slightly lower AIC value than the linear model for growth of vegetative plants. However, this difference is very small ($\Delta\text{AIC} \leq 3$) and so I opted to use the linear model. For the growth of reproductive plants, I found that the linear fit performed better than each nonlinear fit. I therefore selected

linear functions to describe the growth rates of vegetative and reproductive plants.

To test the possibility that density may affect growth rates I fit functions for the growth of vegetative and reproductive plants using linear regression with plant height and population biomass as covariates. Although the AIC value for the function describing the growth of vegetative plants was not improved by using density as a covariate, the function describing the growth of reproductive plants did improve slightly (Table 3.3). However, since this improvement was small ($\Delta\text{AIC} = 2$), I decided to use the linear model with only height as a covariate for both growth functions in both models.

In order to evaluate if the linear functions are suitable, I tested both fits for heteroskedasticity and normality of residuals. The Breusch-Pagan test indicated that the hypothesis that variance is constant cannot be rejected ($p = 0.0887$ for vegetative plant growth and $p = 0.06978$ for flowering plant growth) and therefore the assumptions of homoskedasticity are not violated. Similarly, the Shapiro-Wilk test does not reject the hypothesis that residuals are normally distributed ($p = 0.1223$ for vegetative plant growth and 0.2917 for flowering plant growth). I therefore concluded that the growth kernel is a Gaussian conditional probability distribution with mean scaling linearly with height and constant variance.

Expected number of recruits

Like Eager and coauthors [22], I used a Michaelis-Menten function to describe the expected number of recruits for a given number of seeds. The Michaelis-Menten function was derived from first principles by Eager and coauthors by assuming seeds compete for available microsites in which the seeds may establish. Let n_s be the number of seeds, $f_e(n_s)$ be the total number of recruits that will result from n_s seeds, α be the maximum number of seedlings a plot may sustain, β be the number of seeds that would result in $\alpha/2$ seedlings, and $g_e(n_s)$ be the probability a seed germinates and establishes given n_s seeds. The form used for recruitment is:

$$f_e(n_s) = \frac{\alpha}{\beta + n_s} n_s = g_e(n_s) n_s. \tag{3.9}$$

I used a combination of data collected by Hüls and coauthors [39] and data collected by Cuddington and Grguric to parameterize the recruitment function.

Hüls and coauthors [39] conducted seed sowing experiments in which 0.1 m^2 plots were saturated with giant hogweed seeds. The exact number of seeds used in the experiments

is unknown and so I cannot use regression to fit the Michaelis-Menten function. However, by assuming a sufficiently large number of seeds were used in each experiment, I can assume that the mean number of seedlings in each plot is the expected maximum number of seedlings that may establish in 0.1 m^2. The mean number of seedlings that established in these plots was 5.1. Since the plots in which Hüls and coauthors collected all their other data each have an area of 2.5 m^2, this translates to a maximum seedling capacity of $\alpha = 127.5$.

In order to determine β, I used the data collected by Cuddington and Grguric in which 50 seeds were each sowed in twelve 1 m^2 plots (Table 3.2). The mean number of seedlings which resulted were 16 seedlings/m^2. In a plot with an area of 2.5 m^2, this is equivalent to 125 seeds sowed and a resulting 40 seedlings. Therefore:

$$f_e(125) = \frac{127.5}{\beta + 125} 125 = 40, \tag{3.10}$$

which implies that $\beta = 273.4375$.

Recruit size distribution

As was done in chapter 2, I assumed that recruit size follows a log-normal distribution. I tested this assumption by log-transforming recruit height and using the Shapiro-Wilk test for normality. The Shapiro-Wilk test did not reject the hypothesis that the log-transformed values are normally distributed ($p = 0.58$). I denote the recruit size distribution by $c_0(y)$ (Table 3.4). Unintentional eviction was prevented by truncating the distribution at $L = 5$ cm and $U = 550$ cm and normalizing.

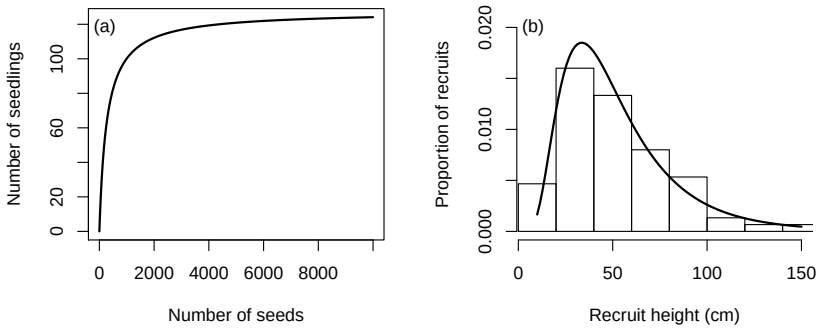

Figure 3.2: Number of seedlings as a function of the seed set in the plot (a) and histogram of observed distribution of recruit size along with the fitted log-normal distribution for stands of giant hogweed in the Hesse region of Germany ($n = 75$, b).

Site-specific effects

I used mixed-effects modelling with a random intercept to test whether any vital rate functions were significantly affected by sampling from different sites [7]. The AIC values for the mixed-effects models for survival, reproduction, and the growth of vegetative plants were higher than their corresponding fixed-effects models. The AIC value for the mixed-effects model for the growth of reproductive plants was lower than the AIC value for the corresponding fixed-effects model. However, this difference was not significant as determined by the likelihood-ratio test ($\Delta\text{AIC} < 2$; $p = 0.07345$). Therefore, I did not use mixed-effects modelling in the final model.

Outliers

I removed the same two outliers that were excluded from the density-independent model in the previous chapter. However, there was the potential for more outliers to be identified since the density-dependent model must necessarily track the heights of reproductive plants. I found one reproductive plant that had the magnitude of its residual for growth more than 4 times larger than the standard deviation of the residuals for the function describing the growth of reproductive plants. This plant appears to have had its height recorded incorrectly and so I excluded it from the density-dependent model.

3.2.5 Simulation results

The initial population in each simulation was one seed and no vegetative plants nor flowering plants. These initial conditions were selected to simulate the local population dynamics of a population from the start of an invasion. Populations were simulated for 30 years to determine how seed production changes over the course of an invasion.

The number of seeds produced in the second year were compared to determine how short distance dispersal would be affected. The number of seeds produced in years 3-7 were compared to determine if seed production was significantly affected early in the invasion. Finally, the number of seeds produced annually once populations became established were compared to estimate the relative likelihood of overcoming significant barriers to dispersal, such as forests or managed sites.

3.2.6 Equilibrium size distribution and stability

I determined the existence of an equilibrium for the recruit-limited model and its stabliity using the same technique presented by Rebarber and coauthors [81]. I then modified this technique to find the equilibrium for the adult competition model. Stability of the equilibrium for the adult competition model was determined by computing the dominant eigenvalue of the Jacobian using methods modified from Ellner and Rees [27].

3.3 Results

3.3.1 Stability results for each model

The recruit-limited model satisfies the conditions outlined by Rebarber and coauthors to guarantee the existence of a globally asymptotically stable equilibrium [81]. However, the model that allows for adult competition does not satisfy these conditions and so simulations were performed to determine stability. Simulation results indicate that populations are oscillatory with a period of two years (Figure 3.3). This behaviour was confirmed by modifying the results of Rebarber and coauthors [81] to find the equilibrium for the adult competition model numerically. The Jacobian of this equilibrium has a dominant eigenvalue of -1.04, confirming that the population dynamics around the equilibrium are unstable and oscillatory.

3.3.2 Seed production predicted by each model

The expected number of seeds produced in the second year is predicted to be 1037 seeds in the recruit-limited model and 1504 seeds in the adult competition model. Similarly, the expected number of seeds in the third year is predicted to be 1013 seeds in the recruit-limited model and 1220 seeds in the adult competition model. A second generation of plants may sprout by the fourth year, bringing the expected number of seeds up to 226142 in the recruit-limited model and 313768 in the adult competition model. However, none of these increases in seed production are significant (Table 3.5). Furthermore, these increases in seed production are transient behaviour.

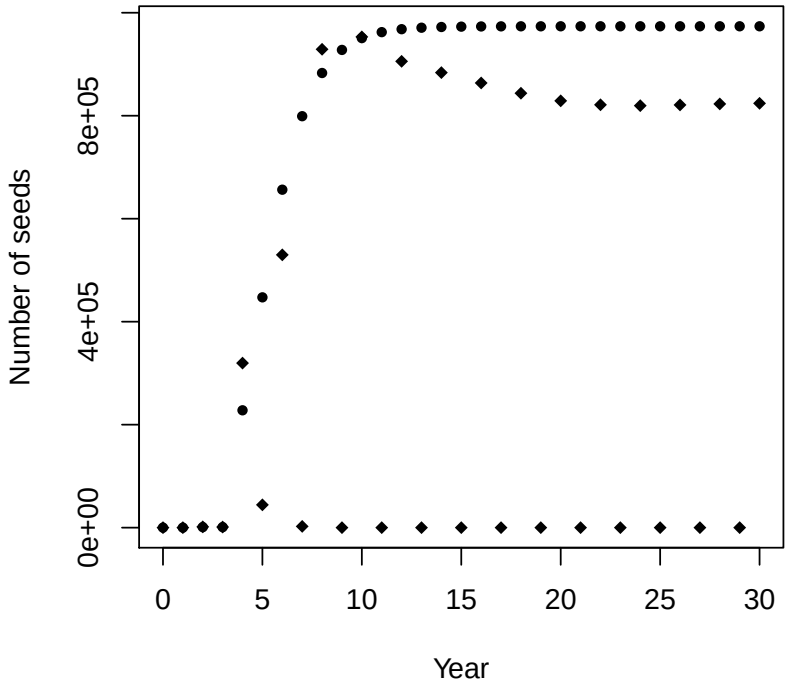

Figure 3.3: Number of seeds in each year predicted by the recruit-limited model (circles) and the model with adult competition (rhombi).

Starting in the fifth year, the density-dependent probability of flowering results in

Table 3.5: Bootstrap mean seed production and confidence intervals for years 2, 3, 4, 5, 6, 7, 29, and 30. RL=recruit-limited and AC=adult competition. 500 bootstrap simulations were performed for each model.

Year	Model	Mean seeds	Bootstrap CI	Plot biomass	Flowering Plants
2	RL	1038	(745, 1344)	4	0.059
	AC	1501	(1020, 1981)	5	0.086
3	RL	1012	(823, 1215)	53	0.057
	AC	1218	(1036, 1414)	57	0.069
4	RL	226298	(150294, 307064)	899	12.84
	AC	311849	(213527, 406929)	1183	18.01
5	RL	443927	(331568, 566017)	1822	25.21
	AC	63781	(5744, 208389)	426	2.50
6	RL	651931	(507199, 802756)	2717	36.99
	AC	505463	(291989, 767949)	2524	29.85
7	RL	794209	(635208, 963349)	3352	45.03
	AC	29916	(4, 217112)	423	0.14
29	RL	969529	(808318, 1149654)	4168	54.87
	AC	2974	(0.1384728, 4120)	303	0.0013
30	RL	969529	(808318, 1149654)	4168	54.87
	AC	832050	(53407, 1261324)	3930	46.44

oscillations in seed production in the adult competition model (Figure 3.3). In odd years, the seed production predicted by the model with adult competition is significantly lower than the seed production predicted by the recruit-limited model. However, in even years the number of seeds produced does not significantly differ between the models.

3.3.3 Differences in mean plant size

The mean height of vegetative plants was observed to be 75 cm with a 95% confidence interval of (71 cm, 80 cm) in 2003. This confidence interval overlapped with the bootstrap equilibrium mean height of vegetative plants predicted by the recruit-limited model, as well as the mean plant heights predicted for each year in the adult competition model (Table 3.6). These bootstrap confidence intervals indicate that there is no significant difference between the observed mean height and the mean heights predicted by either

model.

The mean height of reproductive plants was observed to be 283 cm with a 95% confidence interval of (275 cm, 291 cm) in 2003 (Table 3.6). This confidence interval overlapped with the bootstrap equilibrium mean height of reproductive plants predicted by the recruit-limited model. The confidence interval also overlapped with the corresponding 95% confidence interval predicted by the model with competition among adults. However, the bootstrap confidence interval for the mean height of reproductive plants predicted by the adult competition model in odd years did not overlap with the confidence interval of observed heights of reproductive plants in 2003. This result indicates there is a significant difference between the mean height of reproductive plants in odd years predicted by the model with adult competition when compared to the observed mean height of reproductive plants.

Table 3.6: Bootstrap observed mean height and predicted mean height of vegetative and reproductive plants. 500 bootstrap simulations were performed for each model. All heights presented in centimeters.

Plant type	Source	Year	Bootstrap mean	Bootstrap CI
Vegetative	Observed	2003	75	(71, 80)
	Recruit-limited	30	69	(63, 74)
	Adult Competition	29	76	(66, 97)
	Adult Competition	30	83	(73, 102)
Reproductive	Observed	2003	283	(275, 291)
	Recruit-limited	30	280	(271, 290)
	Adult Competition	29	323	(295, 389)
	Adult Competition	30	293	(279, 327)

3.4 Discussion

In this chapter I demonstrated that competition among adult giant hogweed plants results in significantly fewer seeds produced long-term, without a significant difference in initial seed production. I therefore suggest that modelling the delay in flowering may result in significantly lower rates of spread of giant hogweed [68]. In addition, the model with adult competition predicts oscillations in giant hogweed seed production. This prediction is in contrast with the model without adult competition which found that populations converge

to a stable equilibrium [81]. The number of seeds produced in the more fruitful years of the adult competition model is not significantly less than the number of seeds produced in the recruit-limited model. However, the number of seeds produced in less fruitful years of the adult competition model is significantly lower than the number of seeds predicted by the recruit-limited model. Therefore, the results of this chapter suggest the long-term population growth rate and rate of spread will be lower in the model with adult competition.

In order to parameterize this model, I had to combine data from a few different sources. I used data collected by Hüls and coauthors [39] to parameterize functions that describe survival, reproduction, and growth of adult plants. However, the open stand data set and dense stand data set were insufficient on their own to parameterize the model. Therefore, I combined the open and dense stand data sets to use in this chapter. Hüls and coauthors state that the stand types differ in the level of interspecific competition for each stand; however, in practice the authors classified stands based on the ground cover of giant hogweed. The differences between open and dense stands should therefore be accounted for by the measure of intraspecific competition used in this chapter, which will correlate with giant hogweed ground cover. Unfortunately, even after aggregating the data collected by Hüls and coauthors the data was still insufficient to parameterize the model.

To complete the data collection, Cuddington and Grguric collected recruit germination and survival data at three sites in Southern Ontario. The combination of this data set with the data set collected in Germany was necessary to parameterize the function for the expected number of recruits. Climatic conditions are somewhat different between the two locations and it is not known how results will change if recruitment data were collected at the same time as the adult plant demographic data. Although this model may not necessarily describe population dynamics in Germany or Canada, it will provide insights into the general course of giant hogweed establishment and long-term population dynamics. Unfortunately, the collection of adult data and recruit data in two separate locations does introduce an additional issue in the model.

As a result of the collection of demographic data in two locations, it was not possible to determine the effect that adult plants have on the survival and growth of recruits. This interaction is typically ignored in density-dependent models similar to the model presented in this chapter; however, competition between adult plants and seedlings will affect the equilibrium and may affect the results of my stability analysis [21]. In addition to the lack of adult-recruit interaction, I was unable to model the seed bank for this species [51, 57, 23]. The seed bank may help dampen or eliminate oscillations in the model if enough seeds persist in the seed bank to fill in after less fruitful years.

Finally, in order to get a measure of density I assumed that competition was propor-

tional to the biomass of all plants in the plot. This assumption is based off the work of West, Brown, and Enquist who derived allometric scaling relations between different measures of plant size [28, 29, 30, 84, 99]. Numerous researchers have criticised the result [49, 50, 3], arguing that the exponent will be species-dependent. Instead, the exponent for the model is likely in the range from 3 to 6 [40, 3].

I reran the simulations from subsection 3.3.2 with an exponent of 3 and with an exponent of 6 to determine if the uncertainty in the scaling law affects model results. Simulating population growth using the relation for biomass $M \propto h^6$ yields dynamics that are qualitatively the same as the results given in subsection 3.3.2; however, using the relation $M \propto h^3$ predicts that populations will initially reach a high density before settling into a low equilibrium density. Therefore, the uncertainty in the allometric scaling law could lead to drastically different conclusions. In addition, another assumption in my model may be causing oscillations.

I assumed that competition was symmetric since asymmetric competition is more difficult to model and more computationally expensive. However, it has previously been shown that models with symmetric competition may predict oscillations where similar models with asymmetric competition do not [67, 11]. To my knowledge oscillations have not been observed in giant hogweed populations, and so it is possible that the assumption that competition is symmetric may be leading to inaccurate population dynamics. However, even if the oscillations are an artefact of the choice of model, the impact of the density-dependent flowering will likely affect the conclusions of previously published matrix models and integral projection models.

Among matrix models, Pergl and coauthors developed a stochastic matrix model with a given probability for seeds to disperse a long distance away from the mother plant [72]. The authors then simulated the spread of giant hogweed with several different probabilities of long-distance dispersal and found that the values that best agreed with observed rates of spread is between 0.1% and 7.5%. However, the matrices used in the simulation model were density-independent and did not account for the decrease in seed production that may occur in high density populations. This decrease in seed production may lead to different estimates of the probability of long-distance dispersal. Furthermore, Moenickes and Thiele developed a spatial matrix model to determine the factors that will have a significant impact on the rate of spread of the species [56]. However, the hypotheses the authors test each relate to various methods of spread, recruitment limitation, or succession, without concern for the importance of seed production on the rate of spread.

Among individual-based models, Wadsworth and Collingham produced an IBM that does not account for a decrease in seed production [96] and Nehrbass and coauthors produce

four individual-based models in their investigation of giant hogweed spread and control that assume a simple ceiling carrying capacity for the number of plants [63, 60, 61, 62]. In each of these models, population density may have a significant impact on the timing of flowering which in turn may have a significant impact on the results of these studies. However, further research is necessary to understand the implications my results may have on previous studies, if at all.

In each of these previous models, accounting for the delay in flowering may result in significantly fewer seeds being produced in some years. This decrease in seed production is likely to affect predictions of population spread. Pachepsky and Levine [68] observed that decreased seed production resulted in a lower rate of invasion in patchy habitats. This result is applicable to giant hogweed spread since the pattern of spread is typically patchy followed by infilling [62, 54]. The density-dependent decrease in seed production could have an impact on previous models of giant hogweed spread.

Integrodifference models are commonly used to determine the rate of spread for invasive plants [48]. These models make use of a dispersal kernel to describe the dispersal of seeds and the dispersal kernel is commonly assumed to have exponentially bounded tails since leptokurtic dispersal kernels can lead to an unbounded rate of spread. With such dispersal kernels the invasion front moves at a constant speed. Furthermore, van den Bosch and coauthors demonstrate that in the absence of an Allee effect and the absence of long-distance dispersal the rate of spread is only dependent on the population growth rate at the front of the invasion [95]. However, most invasive plants have some long-distance dispersal that creates irregular invasion fronts and allow for plants that aren't at the front of the invasion to influence the rate of spread.

One particularly well-studied example of long-distance dispersal is the post-glacial migration of trees that occurred in the early Holocene. Clark used an integrodifference equation model to predict the rate of spread for various tree species and found that a leptokurtic dispersal kernel is necessary to explain the observed rates of spread [15]. More recently, Neubert and Caswell predicted the rate of spread of teasel in North America using a stage-structured integrodifference model, yet the predicted rate of 0.5639 m/yr is well below the observed rate of 27 km/yr [64]. The authors concluded that the large discrepancy between observed and predicted invasion speed was due to long-distance dispersal along waterways or multiple introductions of the species.

Similarly, Jongejans and coauthors used a stage-structured integrodifference model to predict the spread of musk thistle (*Carduus nutans*) in various non-native ranges [44]. The authors predicted spread rates of 7 m/yr, 9 m/yr, and 29 m/yr in Australia, Kansas, and New Zealand respectively. Jongejans and coauthors followed up with an integrodifference

model coupled with an IPM for musk thistle that predicted a rate of spread of 13.9 m/yr in New Zealand [43]. However, each estimate differed greatly from the observed spread rates of 146 m/yr in Pennsylvania. The authors concluded that the rate of spread in both of their models was underestimated since they did not account for long-distance dispersal. Rather than a uniformly advancing invasion front as predicted by Kot and coauthors [48], the front of musk thistle invasion is highly irregular due to long-distance dispersal. This pattern of invasion is relevant to this chapter since giant hogweed invasion may be driven by long-distance dispersal [72]. The density-dependent dynamics described in this chapter will be necessary to understand for giant hogweed spread due to it's sensitivity to long-distance dispersal.

The work in this chapter advances our understanding of the growth of giant hogweed in environments with limited resources. In particular, this work demonstrates that the predicted number of seeds produced by giant hogweed populations may be significantly affected if flowering is allowed to depend on density. This density-dependent flowering may in turn affect the rate of spread of the species and may need to be accounted for in future models of giant hogweed spread.

Chapter 4

Conclusions and Future Work

4.1 Conclusions

In this thesis I developed several models for giant hogweed population dynamics and investigated how model complexity may affect predicted population parameters.

From the results of chapter 2 I concluded that using an integral projection model rather than a matrix model did not change the main conclusion made by Hüls and coauthors [39]. That is, the integral projection model did not predict a significant difference in population growth rates between open and dense stands.

I concluded from the results of chapter 3 that allowing adult plants to compete with one another may result in significantly fewer seeds produced every other year. This decrease in seed production may lead to lower predictions for the rate of spread of the species [68]. Furthermore, as a result of the oscillations predicted by the adult competition model, the species may be forced to take advantage of invasion windows in order to establish new stands [37]. This result may impact previously published models of giant hogweed spread [96, 72, 56].

4.2 Future work

Endless research topics may be found by investigating how model complexity affects predicted population dynamics. A natural extension to the work presented in chapter 2 is to determine if population parameters other than population growth rate are affected by

the use of a matrix model or an integral projection model. Specifically, the predicted net reproductive ratio and predicted generation time for a population could be biased if predicted by a matrix model. Such results could further the research performed by Ramula and coauthors that proved that population growth rate may be biased by the discretization used to create some matrix models [79].

The work in chapter 3 was focussed on the impact that competition may have on total seed production and the possibility that the difference in seed production may lead to a higher or lower rate of spread. This work may be extended by developing stochastic, spatial integral projection models with and without adult competition and verifying if the rate of spread differs between the models. Such work would drastically improve the conclusions of chapter 3 by determining if a significantly different rate of spread is found, rather than simply speculating.

References

[1] Federal noxious weed list, 2010.

[2] Noxious weeds in ontario, 2015.

[3] Paul S Agutter and Jack A Tuszynski. Analytic theories of allometric scaling. *Journal of Experimental Biology*, 214(7):1055–1062, 2011.

[4] Mark C Andersen. Potential applications of population viability analysis to risk assessment for invasive species. *Human and Ecological Risk Assessment*, 11(6):1083–1095, 2005.

[5] Dominic A Andradi-Brown, Mark JA Vermeij, Marc Slattery, Michael Lesser, Ivonne Bejarano, Richard Appeldoorn, Gretchen Goodbody-Gringley, Alex D Chequer, Joanna M Pitt, Corey Eddy, et al. Large-scale invasion of western atlantic mesophotic reefs by lionfish potentially undermines culling-based management. *Biological invasions*, 19(3):939–954, 2017.

[6] Steven M Bartell and Shyam K Nair. Establishment risks for invasive species. *Risk Analysis*, 24(4):833–845, 2004.

[7] Douglas Bates, Martin Mächler, Ben Bolker, and Steve Walker. Fitting linear mixed-effects models using lme4. *Journal of Statistical Software*, 67(1):1–48, 2015.

[8] GM Berntson and PM Wayne. Characterizing the size dependence of resource acquisition within crowded plant populations. *Ecology*, 81(4):1072–1085, 2000.

[9] Kenneth P Burnham and David R Anderson. *Model selection and multimodel inference: a practical information-theoretic approach*. Springer Science & Business Media, 2003.

[10] Joe M Caffrey. Phenology and long-term control of heracleum mantegazzianum. In *Biology, Ecology and Management of Aquatic Plants*, pages 223–228. Springer, 1999.

[11] Paul Caplat, Madhur Anand, and Chris Bauch. Symmetric competition causes population oscillations in an individual-based model of forest dynamics. *Ecological Modelling*, 211(3-4):491–500, 2008.

[12] Hal Caswell. *Matrix population models*. Wiley Online Library, 1989.

[13] Dylan Z Childs, Mark Rees, Karen E Rose, Peter J Grubb, and Stephen P Ellner. Evolution of complex flowering strategies: an age–and size–structured integral projection model. *Proceedings of the Royal Society of London B: Biological Sciences*, 270(1526):1829–1838, 2003.

[14] Dylan Z Childs, Mark Rees, Karen E Rose, Peter J Grubb, and Stephen P Ellner. Evolution of size–dependent flowering in a variable environment: construction and analysis of a stochastic integral projection model. *Proceedings of the Royal Society of London B: Biological Sciences*, 271(1537):425–434, 2004.

[15] James S Clark. Why trees migrate so fast: confronting theory with dispersal biology and the paleorecord. *The American Naturalist*, 152(2):204–224, 1998.

[16] Elizabeth E Crone, Eric S Menges, Martha M Ellis, Timothy Bell, Paulette Bierzychudek, Johan Ehrlén, Thomas N Kaye, Tiffany M Knight, Peter Lesica, William F Morris, et al. How do plant ecologists use matrix population models? *Ecology Letters*, 14(1):1–8, 2011.

[17] Elizabeth E Crone and Joshua M Rapp. Resource depletion, pollen coupling, and the ecology of mast seeding. *Annals of the New York Academy of Sciences*, 1322(1):21–34, 2014.

[18] Deborah T Crouse, Larry B Crowder, and Hal Caswell. A stage-based population model for loggerhead sea turtles and implications for conservation. *Ecology*, 68(5):1412–1423, 1987.

[19] Johan P Dahlgren, María B García, and Johan Ehrlén. Nonlinear relationships between vital rates and state variables in demographic models. *Ecology*, 92(5):1181–1187, 2011.

[20] Daniel F Doak, Kevin Gross, and William F Morris. Understanding and predicting the effects of sparse data on demographic analyses. *Ecology*, 86(5):1154–1163, 2005.

[21] Eric Alan Eager. Modelling and analysis of population dynamics using lure systems accounting for competition from adult conspecifics. *Letters in Biomathematics*, 3(1):41–58, 2016.

[22] Eric Alan Eager, Richard Rebarber, and Brigitte Tenhumberg. Choice of density-dependent seedling recruitment function affects predicted transient dynamics: a case study with platte thistle. *Theoretical ecology*, 5(3):387–401, 2012.

[23] Eric Alan Eager, Richard Rebarber, and Brigitte Tenhumberg. Global asymptotic stability of plant-seed bank models. *Journal of mathematical biology*, 69(1):1–37, 2014.

[24] Michael R Easterling, Stephen P Ellner, and Philip M Dixon. Size-specific sensitivity: applying a new structured population model. *Ecology*, 81(3):694–708, 2000.

[25] Michael Robert Easterling. *The integral projection model: theory, analysis and application*. 1998.

[26] Stephen P Ellner, Dylan Z Childs, and Mark Rees. *Data-driven modelling of structured populations*. Springer, 2016.

[27] Stephen P Ellner and Mark Rees. Integral projection models for species with complex demography. *The American Naturalist*, 167(3):410–428, 2006.

[28] Brian J Enquist, James H Brown, and Geoffrey B West. Allometric scaling of plant energetics and population density. *Nature*, 395(6698):163, 1998.

[29] Brian J Enquist and Karl J Niklas. Invariant scaling relations across tree-dominated communities. *Nature*, 410(6829):655–660, 2001.

[30] Brian J Enquist and Karl J Niklas. Global allocation rules for patterns of biomass partitioning in seed plants. *Science*, 295(5559):1517–1520, 2002.

[31] Katherine Everard, Eric W Seabloom, W Stanley Harpole, and Claire de Mazancourt. Plant water use affects competition for nitrogen: why drought favors invasive species in california. *The American Naturalist*, 175(1):85–97, 2009.

[32] Deborah E Goldberg. Neighborhood competition in an old-field plant community. *Ecology*, 68(5):1211–1223, 1987.

[33] Deborah E Goldberg and Linda Fleetwood. Competitive effect and response in four annual plants. *The Journal of Ecology*, pages 1131–1143, 1987.

[34] Deborah E Goldberg and Keith Landa. Competitive effect and response: hierarchies and correlated traits in the early stages of competition. *The Journal of Ecology*, pages 1013–1030, 1991.

[35] Volker Grimm, Uta Berger, Finn Bastiansen, Sigrunn Eliassen, Vincent Ginot, Jarl Giske, John Goss-Custard, Tamara Grand, Simone K Heinz, Geir Huse, et al. A standard protocol for describing individual-based and agent-based models. *Ecological modelling*, 198(1):115–126, 2006.

[36] MP Hassell. Density-dependence in single-species populations. *The Journal of animal ecology*, pages 283–295, 1975.

[37] Richard J Hobbs and Laura F Huenneke. Disturbance, diversity, and invasion: implications for conservation. *Conservation biology*, 6(3):324–337, 1992.

[38] Jörg Hüls. Populationsbiologische untersuchung von heracleum mantegazzianum somm. et lev. *Subpopulationen unterschiedlicher Individuendichte. Dissertation, University of Giessen*, 2005.

[39] Jörg Hüls, Annette Otte, and R Lutz Eckstein. Population life-cycle and stand structure in dense and open stands of the introduced tall herb heracleum mantegazzianum. *Biological Invasions*, 9(7):799–811, 2007.

[40] Nick JB Isaac and Chris Carbone. Why are metabolic scaling exponents so controversial? quantifying variance and testing hypotheses. *Ecology letters*, 13(6):728–735, 2010.

[41] Louise E Jackson. Ecological origins of california's mediterranean grasses. *Journal of Biogeography*, pages 349–361, 1985.

[42] Kateřina Jandová, Petr Dostál, Tomáš Cajthaml, and Zdeněk Kameník. Intraspecific variability in allelopathy of heracleum mantegazzianum is linked to the metabolic profile of root exudates. *Annals of botany*, 115(5):821–831, 2015.

[43] Eelke Jongejans, Katriona Shea, Olav Skarpaas, Dave Kelly, and Stephen P Ellner. Importance of individual and environmental variation for invasive species spread: a spatial integral projection model. *Ecology*, 92(1):86–97, 2011.

[44] Eelke Jongejans, Katriona Shea, Olav Skarpaas, Dave Kelly, Andy W Sheppard, and Tim L Woodburn. Dispersal and demography contributions to population spread of carduus nutans in its native and invaded ranges. *Journal of Ecology*, 96(4):687–697, 2008.

[45] Dave Kelly. The evolutionary ecology of mast seeding. *Trends in ecology & evolution*, 9(12):465–470, 1994.

[46] Sarah Kimball, Michael L Goulden, Katharine N Suding, and Scot Parker. Altered water and nitrogen input shifts succession in a southern california coastal sage community. *Ecological Applications*, 24(6):1390–1404, 2014.

[47] Éva Kisdi. Year-class coexistence in biennial plants. *Theoretical population biology*, 82(1):18–21, 2012.

[48] Mark Kot, Mark A Lewis, and Pauline van den Driessche. Dispersal data and the spread of invading organisms. *Ecology*, 77(7):2027–2042, 1996.

[49] J Kozłowski and M Konarzewski. Is west, brown and enquist's model of allometric scaling mathematically correct and biologically relevant? *Functional Ecology*, 18(2):283–289, 2004.

[50] J Kozłowski and M Konarzewski. West, brown and enquist's model of allometric scaling again: the same questions remain. *Functional Ecology*, 19(4):739–743, 2005.

[51] Lukáš Krinke, Lenka Moravcová, Petr Pyšek, Vojtěch Jarošík, Jan Pergl, and Irena Perglová. Seed bank of an invasive alien, heracleum mantegazzianum, and its seasonal dynamics. *Seed science research*, 15(03):239–248, 2005.

[52] Patrick Kuss, Mark Rees, Hafdís Hanna Ægisdóttir, Stephen P Ellner, and Jürg Stöcklin. Evolutionary demography of long-lived monocarpic perennials: a time-lagged integral projection model. *Journal of Ecology*, 96(4):821–832, 2008.

[53] Monte Lloyd and Henry S Dybas. The periodical cicada problem. i. population ecology. *Evolution*, 20(2):133–149, 1966.

[54] Piotr Mędrzycki, Ingeborga Jarzyna, Artur Obidziński, Barbara Tokarska-Guzik, Zofia Sotek, Piotr Pabjanek, Adam Pytlarczyk, and Izabela Sachajdakiewicz. Simple yet effective: Historical proximity variables improve the species distribution models for invasive giant hogweed (heracleum mantegazzianum sl) in poland. *PloS one*, 12(9):e0184677, 2017.

[55] Cory Merow, Johan P Dahlgren, C Jessica E Metcalf, Dylan Z Childs, Margaret EK Evans, Eelke Jongejans, Sydne Record, Mark Rees, Roberto Salguero-Gómez, and Sean M McMahon. Advancing population ecology with integral projection models: a practical guide. *Methods in Ecology and Evolution*, 5(2):99–110, 2014.

[56] Sylvia Moenickes and Jan Thiele. What shapes giant hogweed invasion? answers from a spatio-temporal model integrating multiscale monitoring data. *Biological invasions*, 15(1):61–73, 2013.

[57] Lenka Moravcova, P Pyšek, Lukas Krinke, Jan Pergl, Irena Perglova, Ken Thompson, et al. Seed germination, dispersal and seed bank in heracleum mantegazzianum. *Ecology and management of giant hogweed*, pages 74–91, 2007.

[58] Lenka Moravcova, P Pyšek, Jan Pergl, Irena Perglova, V Jarošík, et al. Seasonal pattern of germination and seed longevity in the invasive species heracleum mantegazzianum. *Preslia*, 78(3):287–301, 2006.

[59] Jana Müllerová, Petr Pyšek, Vojtěch Jarošík, and Jan Pergl. Aerial photographs as a tool for assessing the regional dynamics of the invasive plant species heracleum mantegazzianum. *Journal of Applied Ecology*, 42(6):1042–1053, 2005.

[60] Nana Nehrbass and Eckart Winkler. Is the giant hogweed still a threat? an individual-based modelling approach for local invasion dynamics of heracleum mantegazzianum. *Ecological Modelling*, 201(3):377–384, 2007.

[61] Nana Nehrbass, Eckart Winkler, et al. Model-assisted evaluation of control strategies for heracleum mantegazzianum. *Ecology and Management of Giant Hogweed (Heracleum Mantegazziannum)*, page 284, 2007.

[62] Nana Nehrbass, Eckart Winkler, Jana Müllerová, Jan Pergl, Petr Pyšek, and Irena Perglová. A simulation model of plant invasion: long-distance dispersal determines the pattern of spread. *Biological Invasions*, 9(4):383–395, 2007.

[63] Nana Nehrbass, Eckart Winkler, Jan Pergl, Irena Perglová, and Petr Pyšek. Empirical and virtual investigation of the population dynamics of an alien plant under the constraints of local carrying capacity: Heracleum mantegazzianum in the czech republic. *Perspectives in Plant Ecology, Evolution and Systematics*, 7(4):253–262, 2006.

[64] Michael G Neubert and Hal Caswell. Demography and dispersal: calculation and sensitivity analysis of invasion speed for structured populations. *Ecology*, 81(6):1613–1628, 2000.

[65] Charlotte Nielsen, Hans Peter Ravn, Wolfgang Nentwig, and Max Wade. The giant hogweed best practice manual. guidelines for the management and control of an invasive weed in europe. *Forest and Landscape Denmark, Hoersholm*, 2005.

[66] J Ochsmann et al. Heracleum mantegazzianum sommier & levier (apiaceae) in germany. studies on biology, distribution, morphology and taxonomy. *Feddes Repertorium*, 107(7-8):557–595, 1996.

[67] SW Pacala and J Weiner. Effects of competitive asymmetry on a local density model of plant interference. *Journal of Theoretical Biology*, 149(2):165–179, 1991.

[68] Elizaveta Pachepsky and Jonathan M Levine. Density dependence slows invader spread in fragmented landscapes. *The American Naturalist*, 177(1):18–28, 2010.

[69] Nicholas A Page, Ronald E Wall, Stephen J Darbyshire, and Gerald A Mulligan. The biology of invasive alien plants in canada. 4. heracleum mantegazzianum sommier & levier. *Canadian Journal of Plant Science*, 86(2):569–589, 2006.

[70] CE Paine, Toby R Marthews, Deborah R Vogt, Drew Purves, Mark Rees, Andy Hector, and Lindsay A Turnbull. How to fit nonlinear plant growth models and calculate growth rates: an update for ecologists. *Methods in Ecology and Evolution*, 3(2):245–256, 2012.

[71] Jan Pergl, Jörg Hüls, Irena Perglova, R Lutz Eckstein, P Pyšek, Annette Otte, et al. Population dynamics of heracleum mantegazzianum. *Ecology and management of giant hogweed*, pages 92–111, 2007.

[72] Jan Pergl, Jana Müllerová, Irena Perglová, Tomáš Herben, and Petr Pyšek. The role of long-distance seed dispersal in the local population dynamics of an invasive plant species. *Diversity and Distributions*, 17(4):725–738, 2011.

[73] Jan Pergl, Irena Perglova, Petr Pyšek, and Hansjörg Dietz. Population age structure and reproductive behavior of the monocarpic perennial heracleum mantegazzianum (apiaceae) in its native and invaded distribution ranges. *American Journal of Botany*, 93(7):1018–1028, 2006.

[74] Irena Perglova, Jan Pergl, P Pyšek, et al. Flowering phenology and reproductive effort of the invasive alien plant heracleum mantegazzianum. *Preslia*, 78(3):265–285, 2006.

[75] Irena Perglova, Jan Pergl, P Pyšek, et al. Reproductive ecology of heracleum mantegazzianum. *Ecology and management of giant hogweed*, pages 55–73, 2007.

[76] Petr Pyšek. Ecological aspects of invasion by heracleum mantegazzianum in the czech republic. *Ecology and management of invasive riverside plants*, pages 45–54, 1994.

[77] Petr Pyšek, T Kucera, J Puntieri, and B Mandák. Regeneration in heracleum mantegazzianum-response to removal of vegetative and generative parts. *PRESLIA-PRAHA-*, 67:161–172, 1995.

[78] Petr Pyšek and Antonín Pyšek. Invasion by heracleum mantegazzianum in different habitats in the czech republic. *Journal of vegetation science*, 6(5):711–718, 1995.

[79] Satu Ramula, Mark Rees, and Yvonne M Buckley. Integral projection models perform better for small demographic data sets than matrix population models: a case study of two perennial herbs. *Journal of Applied Ecology*, 46(5):1048–1053, 2009.

[80] RJ Reader, SD Wilson, JW Belcher, I Wisheu, PA Keddy, D Tilman, EC Morris, JB Grace, JB McGraw, H Olff, et al. Plant competition in relation to neighbor biomass: an intercontinental study with poa pratensis. *Ecology*, 75(6):1753–1760, 1994.

[81] Richard Rebarber, Brigitte Tenhumberg, and Stuart Townley. Global asymptotic stability of density dependent integral population projection models. *Theoretical Population Biology*, 81(1):81–87, 2012.

[82] Mark Rees, Dylan Z Childs, and Stephen P Ellner. Building integral projection models: a user's guide. *Journal of Animal Ecology*, 83(3):528–545, 2014.

[83] Mark Rees and Karen E Rose. Evolution of flowering strategies in oenothera glazioviana: an integral projection model approach. *Proceedings of the Royal Society of London B: Biological Sciences*, 269(1499):1509–1515, 2002.

[84] Lawren Sack, Teodoro Marañón, and Peter J Grubb. Global allocation rules for patterns of biomass partitioning. *Science*, 296(5575):1923–1923, 2002.

[85] Roberto Salguero-Gómez and Brenda B Casper. Keeping plant shrinkage in the demographic loop. *Journal of Ecology*, 98(2):312–323, 2010.

[86] Clare Sampson, LD Waal, LE Child, PM Wade, and JH Brock. Cost and impact of current control methods used against heracleum mantegazzianum (giant hogweed) and the case for instigating a biological control programme. *Ecology and management of invasive riverside plants*, pages 55–65, 1994.

[87] Susanne Schwinning and Jacob Weiner. Mechanisms determining the degree of size asymmetry in competition among plants. *Oecologia*, 113(4):447–455, 1998.

[88] Vidar Selås. Cyclic population fluctuations of herbivores as an effect of cyclic seed cropping of plants: the mast depression hypothesis. *Oikos*, pages 257–268, 1997.

[89] Isabel M Smallegange, Hal Caswell, Marjolein EM Toorians, and André M Roos. Mechanistic description of population dynamics using dynamic energy budget theory incorporated into integral projection models. *Methods in Ecology and Evolution*, 8(2):146–154, 2017.

[90] Jan Thiele, Johannes Kollmann, Bo Markussen, and Annette Otte. Impact assessment revisited: improving the theoretical basis for management of invasive alien species. *Biological Invasions*, 12(7):2025–2035, 2010.

[91] Jan Thiele and Annette Otte. Invasion patterns of heracleum mantegazzianum in germany on the regional and landscape scales. *Journal for Nature Conservation*, 16(2):61–71, 2008.

[92] Diane M Thomson. Do source–sink dynamics promote the spread of an invasive grass into a novel habitat? *Ecology*, 88(12):3126–3134, 2007.

[93] Gordon ED Tiley, Felicite S Dodd, and PM Wade. Heracleum mantegazzianum sommier & levier. *Journal of Ecology*, 84(2):297–319, 1996.

[94] Gordon ED Tiley, Bruce Philp, et al. Effects of cutting flowering stems of giant hogweed heracleum mantegazzianum on reproductive performance. *Aspects of Applied Biology*, (58):77–80, 2000.

[95] Frank Van den Bosch, Johan Anton Jacob Metz, and Odo Diekmann. The velocity of spatial population expansion. *Journal of Mathematical Biology*, 28(5):529–565, 1990.

[96] RA Wadsworth, YC Collingham, SG Willis, B Huntley, and PE Hulme. Simulating the spread and management of alien riparian weeds: are they out of control? *Journal of Applied Ecology*, 37(s1):28–38, 2000.

[97] Jacob Weiner. The influence of competition on plant reproduction. *Plant reproductive ecology: patterns and strategies*, pages 228–245, 1988.

[98] Jacob Weiner. Asymmetric competition in plant populations. *Trends in ecology & evolution*, 5(11):360–364, 1990.

[99] Geoffrey B West, James H Brown, and Brian J Enquist. A general model for the structure and allometry of plant vascular systems. *Nature*, 400(6745):664, 1999.

[100] Wibke Wille, Jan Thiele, Emer A Walker, and Johannes Kollmann. Limited evidence for allelopathic effects of giant hogweed on germination of native herbs. *Seed Science Research*, 23(2):157–162, 2013.

[101] Jennifer L Williams, Tom EX Miller, and Stephen P Ellner. Avoiding unintentional eviction from integral projection models. *Ecology*, 93(9):2008–2014, 2012.

[102] S.N Wood. *Generalized Additive Models: An Introduction with R*. Chapman and Hall/CRC, 2 edition, 2017.

Appendices

Appendix A

Chapter 2 Code

A.1 Determining Population Growth Rates

```
1  ### ------------------------------------------------------------
2  ## -- Initialization
3
4  # Clear workspace
5  rm(list=ls())
6
7  # Library
8  library(MASS)    # Needed for 'fitdistr' function
9  library(lmtest) # Needed for 'bptest' function
10 library(mgcv)    # Needed for GAM fitting
11
12 # These are decisions that must be made
13 exclude_outliers <- T
14 growth_model <- "linear" # can be "linear", "ricker", "skellam",
       "bevholt, "power", or "gam"
15 normal_variance <- T
16
17
18 ### ------------------------------------------------------------
19 ## -- Read in the data and define state variable
20
```

```
21 ##################################################################
22 # Data collected by Huls and coauthors:
23 #  joerghuels@web.de
24 #
25 # Each row contains data for one plant
26 #
27 # The first column has an ID for the plant
28 # The first letter of the ID indicates the site: A=Allendorf,
29 #  D=Druseltal, F=Frankenberg, K=Kassel, V=Viermunden
30 # The second letter indicates the stand type
31 #  A,B,C,D,E=Dense stand; X=Open stand
32 # The Roman numerals indicate the plot
33 # The number indicates the plant
34 #
35 # Stage, height, leaf stem diameter, and leaf blade width
36 #  were recorded for 2002 and 2003
37 # All measurements are in cm
38 # Stage 5 indicates the plant died that year
39 ##################################################################
40
41 hmdata <- read.csv("./data/IPMdata.csv", na.strings="␣")
42
43 # I use height as the state variable
44 hmdata["h"]  <- hmdata[["Height2002"]]
45 hmdata["h1"] <- hmdata[["Height2003"]]
46
47
48 ### ---------------------------------------------------
49 ## -- Remove Allendorf data, divide plots based on density
50
51 # The site is the first letter of the ID
52 hmdata["site"] <- substr(hmdata$ID,1,1) # first letter of
       individual ID
53
54 # Remove Allendorf data due to grazing damage
55 hmdata <- hmdata[hmdata$site!="A",]
56
57 # The density is the second letter of the ID
```

```
58 hmdata["dens"] <- ifelse(substr(hmdata$ID,2,2)=="X", "Low", "High")
59
60 # Determine the plot number
61 hmdata["plot"] <- substr(hmdata$ID,1,as.numeric(lapply(as.character(hmdata$ID),nchar))-2)
62
63
64 ### ---------------------------------------------------
65 ## -- Survival, reproduction, and classification (i.e. seedling or adult)
66
67 # 1 means survived, 0 means did not survive, NA means flowered (so of course it's dead)
68 hmdata["survived"] <- ifelse(hmdata[["Stage2002"]] == 4, NA, ifelse(hmdata[["Stage2003"]] == 5, 0, 1))
69
70 # 1 means reproduced, 0 means did not reproduce, NA means individual died without reproducing
71 hmdata["reproduced"] <- ifelse(hmdata[["Stage2003"]] == 4, 1, ifelse(hmdata[["Stage2003"]] == 5, NA, 0))
72
73 # If a plant was recorded in 2003 but not 2002 it was a seedling in 2002
74 hmdata["class"] <- ifelse(is.na(hmdata[["Stage2002"]]), "Seedling", "Adult")
75
76
77 ### ---------------------------------------------------
78 ## -- Create another dataframe without the outlier
79
80 # Help identify outliers
81 out1 <- as.character(hmdata[hmdata$h>80 & hmdata$h1<10 & !is.na(hmdata$h) & !is.na(hmdata$h1),"ID"])
82 out2 <- as.character(hmdata[hmdata$h<50 & hmdata$h1>200 & hmdata$reproduced==0 & !is.na(hmdata$h) & !is.na(hmdata$h1),"ID"])
83 out3 <- as.character(hmdata[is.na(hmdata$h) & is.na(hmdata$h1),"ID"])
```

```
84
85 # Take a look at the outliers
86 hmdata[hmdata$ID==out1,]
87 hmdata[hmdata$ID==out2,]
88 hmdata[hmdata$ID==out3,]
89
90 # Create a new data frame without them
91 hmdata_no_outlier <- hmdata[hmdata$ID!=out1 & hmdata$ID!=out2 &
       hmdata$ID!=out3,]
92
93
94 ### ------------------------------------------------------------
95 ## -- Split single data frame into one for seedlings and one for
       adult plants
96
97 seedlingdata <- hmdata[hmdata$class=="Seedling",]
98 adultdata    <- hmdata[hmdata$class!="Seedling",]
99
100 seedlingdata_no_outlier <- hmdata_no_outlier[hmdata_no_outlier$
       class=="Seedling",]
101 adultdata_no_outlier    <- hmdata_no_outlier[hmdata_no_outlier$
       class!="Seedling",]
102
103
104 ### ------------------------------------------------------------
105 ## -- The data set to perform regression on
106
107 # Which subset of data to use
108 if (exclude_outliers) {
109   dat <- adultdata_no_outlier
110 } else {dat <- adultdata}
111
112 dat_low <- dat[dat$dens=="Low",]
113 dat_high <- dat[dat$dens=="High",]
114
115
116 ### ------------------------------------------------------------
117 ## -- Probability of survival
```

```
118
119 # Logistic regression for survival probability depending on leaf
    stem diameter
120 surv_fit_low  <- glm(survived ~ h, data=dat_low, family="binomial
    ")
121 surv_fit_high <- glm(survived ~ h, data=dat_high, family="
    binomial")
122
123 # Survival function
124 p_s <- function(h, dens){
125   tempdf <- data.frame(h=h)
126   if (dens=="Low") {out<-predict(surv_fit_low, tempdf, type="
      response")}
127   else             {out<-predict(surv_fit_high, tempdf, type="
      response")}
128   return(out)
129 }
130
131
132 ### ---------------------------------------------
133 ## -- Probability of reproduction
134
135 # Logistic regression for probability of reproduction depending
    on leaf stem diameter
136 repr_fit_low  <- glm(reproduced ~ h, data=dat_low, family="
    binomial")
137 repr_fit_high <- glm(reproduced ~ h, data=dat_high, family="
    binomial")
138
139 # Reproduction function
140 p_r <- function(h, dens){
141   tempdf <- data.frame(h=h)
142   if (dens=="Low"){out<-predict(repr_fit_low, tempdf, type="
      response")}
143   else            {out<-predict(repr_fit_high, tempdf, type="
      response")}
144   return(out)
145 }
```

```
146
147
148 ### ------------------------------------------------
149 ## --- Determine growth
150
151 # I need to exclude flowering plants and dead plants
152 dat_adult <- dat[dat$Stage2003!=4 & dat$Stage2003!=5,]
153 dat_adult_low <- dat_adult[dat_adult$dens=="Low",]
154 dat_adult_high <- dat_adult[dat_adult$dens=="High",]
155
156 # Linear regression
157 if (growth_model=="linear") {
158   grow_fit_low <- lm(h1~h, data=dat_adult_low)
159   grow_fit_high <- lm(h1~h, data=dat_adult_high)
160 } else if (growth_model=="ricker") {
161   rick_fun <- function(x, coefs) {a<-coefs[1]; b<-coefs[2];
        return(a*x*exp(-b*x))}
162   grow_fit_low <- nls(h1 ~ rick_fun(h,c(a,b)), data=dat_adult_low
        , start=list(a=1,b=0.01))
163   grow_fit_high <- nls(h1 ~ rick_fun(h,c(a,b)), data=dat_adult_
        high, start=list(a=1,b=0.01))
164 } else if (growth_model=="skellam") {
165   skel_fun <- function(x, coefs) {a<-coefs[1]; b<-coefs[2];
        return(a*(1-exp(-b*x)))}
166   grow_fit_low <- nls(h1 ~ skel_fun(h,c(a,b)), data=dat_adult_low
        , start=list(a=100,b=0.01))
167   grow_fit_high <- nls(h1 ~ skel_fun(h,c(a,b)), data=dat_adult_
        high, start=list(a=100,b=0.01))
168 } else if (growth_model=="power") {
169   powr_fun <- function(x,coefs) {a<-coefs[1]; b<-coefs[2]; return
        (a*x^b)}
170   grow_fit_low <- nls(h1 ~ powr_fun(h,c(a,b)), data=dat_adult_low
        , start=list(a=1,b=1))
171   grow_fit_high <- nls(h1 ~ powr_fun(h,c(a,b)), data=dat_adult_
        high, start=list(a=1,b=1))
172 } else if (growth_model=="bevholt") {
173   bvht_fun <- function(x, coefs) {a<-coefs[1]; b<-coefs[2];
        return(a*x/(b+x))}
```

```
174   grow_fit_low <- nls(h1 ~ bvht_fun(h,c(a,b)), data=dat_adult_low
          , start=list(a=1,b=1))
175   grow_fit_high <- nls(h1 ~ bvht_fun(h,c(a,b)), data=dat_adult_
          high, start=list(a=1,b=1))
176 } else if (growth_model=="gam") {
177   grow_fit_low <- gam(h1~s(h),data=dat_adult_low)
178   grow_fit_high <- gam(h1~s(h),data=dat_adult_high)
179 } else {rm}
180
181
182 ### ---------------------------------------------
183 ## -- Growth kernel
184
185 # Define growth kernel
186 g_k <- function(h1,h, dens) {
187
188   # Define nice data frame to use
189   newdata <- data.frame(h=h)
190
191   # Two cases: open or dense stands
192   if (dens=="Low") {
193     h1bar  <- predict(grow_fit_low, newdata=newdata, type="
          response")
194     res    <- residuals(grow_fit_low)
195     df_res <- df.residual(grow_fit_low)}
196   else {
197     h1bar  <- predict(grow_fit_high, newdata=newdata, type="
          response")
198     res    <- residuals(grow_fit_high)
199     df_res <- df.residual(grow_fit_high)}
200
201   # I need standard error and estimated standard deviation
202   sse   <- sum(res^2)
203   sdhat <- sqrt(sse/df_res)
204
205   # Two cases: normal variance or nonparametric variance
206   #  see 'kernelExample.R' from Rees, Childs, Ellner (2014)
207   if (normal_variance) {
```

```
208       out <- dnorm(h1, mean=h1bar, sd=sdhat)
209     } else {
210       bw <- bw.SJ(res); alpha <- sdhat/sqrt(sdhat^2+bw^2); bw_res
            <- alpha*res
211       kfun <- function(h) {mean(dnorm(h,mean=bw_res,sd=bw))}; kfun
            <- Vectorize(kfun)
212       out <- kfun(h1-h1bar)}
213
214     # Finally, output the growth
215     return(out)
216 }
217
218
219 ### ---------------------------------------------
220 ## -- Recruitment function
221
222 # Expected number of offspring
223 expected_offspring <- function(dens) {
224     num_new  <- nrow(seedlingdata[seedlingdata$dens==dens,])
225     num_flow <- nrow(dat[dat$Stage2002==4 & dat$dens==dens,])
226     return(num_new/num_flow)
227 }
228
229 # Fit recruit size to a log-normal distribution
230 r_size <- fitdistr(seedlingdata$Height2003, "lognormal")$estimate
231 r_size_lo <- fitdistr(seedlingdata[seedlingdata$dens=="Low",]$
      Height2003, "lognormal")$estimate
232 r_size_hi <- fitdistr(seedlingdata[seedlingdata$dens=="High",]$
      Height2003, "lognormal")$estimate
233
234 # Function for probability of recruit size
235 recruit_size <- function(zvec,dens) {
236     if       (dens=="Low")  {my_dist <- r_size_lo}
237     else if (dens=="High") {my_dist <- r_size_hi}
238     else                    {my_dist <- r_size}
239     return(dlnorm(zvec, meanlog=my_dist["meanlog"], sdlog=my_dist["
          sdlog"], log=F))
240 }
```

```
241
242
243 ### ------------------------------------------------
244 ## -- Kernel functions
245
246 # Survival-growth kernel
247 P_k <- function(h,meshpts,dens) {
248   fxn <- function(z1,z,dens) {
249     return(p_s(z,dens)*(1-p_r(z,dens))*g_k(z1,z,dens))}
250   return(h*(outer(meshpts,meshpts,fxn,dens)))}
251
252 # Reproduction kernel
253 R_k <- function(h,meshpts,dens,combine_repr) {
254   fxn <- function(z1,z,dens) {
255     return(p_s(z,dens)*p_r(z,dens)*expected_offspring(dens)*
          recruit_size(z1,combine_repr)/(h*sum(recruit_size(meshpts,
          combine_repr))))}
256   return(h*(outer(meshpts,meshpts,fxn,dens)))}
257
258
259 ### ------------------------------------------------
260 ## -- Find kernels and population growth rates
261
262 # Function to do so
263 ipm_kernel <- function(h,meshpts,dens,combine_repr) {
264   surv_growth <- P_k(h,meshpts,dens)
265   fecundity <- R_k(h,meshpts,dens,combine_repr)
266   kernel <- surv_growth + fecundity
267   return(kernel)}
268
269 # Specify bounds, mesh, spacing
270 L <- 5
271 U <- 250
272 m <- 1000
273 h <- (U-L)/m
274 meshpts <- L + (1:m)*h - h/2
275
276 # Find kernels
```

```
k_lo <- ipm_kernel(h,meshpts,"Low","Low")
k_hi <- ipm_kernel(h,meshpts,"High","High")

# Find eigenvalues of kernels
lam_lo <- Re(eigen(k_lo,only.values=T)$values[1])
lam_hi <- Re(eigen(k_hi,only.values=T)$values[1])

# Print out the final values
lam_lo
lam_hi


### ------------------------------------------------
## --- Find eigenvectors

# Find eigenvectors of kernels
v_lo <- Re(eigen(t(k_lo))$vectors[,1])
v_hi <- Re(eigen(t(k_hi))$vectors[,1])
w_lo <- Re(eigen(k_lo)$vectors[,1])
w_hi <- Re(eigen(k_hi)$vectors[,1])

# Normalize eigenvectors
v_lo <- v_lo/sum(v_lo)
v_hi <- v_hi/sum(v_hi)
w_lo <- w_lo/sum(w_lo)
w_hi <- w_hi/sum(w_hi)
```

A.2 Diagnostics

A.2.1 Unintentional Eviction

```
### ------------------------------------------------
## --- Compute epsilons and rhos

# Functions to integrate over
not_evicted <- function(x,dens) {
```

```
6     helper_fxn <- function(y) {g_k(y,x,dens)}
7     integrate(helper_fxn,L,U)}
8
9   ne_lo <- function(x) {not_evicted(x,"Low")}
10  ne_hi <- function(x) {not_evicted(x,"High")}
11
12  # Integrate over growth functions
13  temp_lo <- lapply(meshpts,ne_lo)
14  temp_hi <- lapply(meshpts,ne_hi)
15
16  # Find conditional and unconditional eviction
17  eps_lo <- rep.int(0,times=m)
18  eps_hi <- rep.int(0,times=m)
19  rho_lo <- rep.int(0,times=m)
20  rho_hi <- rep.int(0,times=m)
21
22  for (i in 1:m) {
23    eps_lo[i] <- 1-temp_lo[[i]]$value
24    eps_hi[i] <- 1-temp_hi[[i]]$value
25    rho_lo[i] <- p_s(meshpts[i],"Low")*eps_lo[i]
26    rho_hi[i] <- p_s(meshpts[i],"High")*eps_hi[i]
27  }
28
29  # Maximum values
30  max(rho_lo)
31  max(rho_hi)
32
33  # Plot conditional and unconditional eviction
34  par(mfrow=c(2,2))
35  plot(meshpts,eps_lo,xlab="Height",ylab="epsilon",main="Open␣
      stands")
36  plot(meshpts,eps_hi,xlab="Height",ylab="epsilon",main="Dense␣
      stands")
37  plot(meshpts,rho_lo,xlab="Height",ylab="rho",main="Open␣stands")
38  plot(meshpts,rho_hi,xlab="Height",ylab="rho",main="Dense␣stands")
39  par(mfrow=c(1,1))
40
41
```

```
### ------------------------------------------------
## --- Compute eviction at the stable size distribution

# Probability of eviction below L
L_evicted <- function(x,dens) {
  helper_fxn <- function(y) {p_s(x,dens)*g_k(y,x,dens)}
  integrate(helper_fxn,-Inf,L)}

rho_lo_L <- function(x) {L_evicted(x,"Low")}
rho_hi_L <- function(x) {L_evicted(x,"High")}
temp_lo_L <- lapply(meshpts,rho_lo_L)
temp_hi_L <- lapply(meshpts,rho_hi_L)
rho_lo_L_vec <- array(as.numeric(unlist(lapply(1:m,function(i){as
    .numeric(temp_lo_L[[i]]$value)}))))
rho_hi_L_vec <- array(as.numeric(unlist(lapply(1:m,function(i){as
    .numeric(temp_hi_L[[i]]$value)}))))

# Probability of eviction above U
U_evicted <- function(x,dens) {
  helper_fxn <- function(y) {p_s(x,dens)*g_k(y,x,dens)}
  integrate(helper_fxn,U,Inf)}

rho_lo_U <- function(x) {U_evicted(x,"Low")}
rho_hi_U <- function(x) {U_evicted(x,"High")}
temp_lo_U <- lapply(meshpts,rho_lo_U)
temp_hi_U <- lapply(meshpts,rho_hi_U)
rho_lo_U_vec <- array(as.numeric(unlist(lapply(1:m,function(i){as
    .numeric(temp_lo_U[[i]]$value)}))))
rho_hi_U_vec <- array(as.numeric(unlist(lapply(1:m,function(i){as
    .numeric(temp_hi_U[[i]]$value)}))))

# Unintentional eviction
format(w_lo %*% rho_lo_L_vec,scientific=F)
format(w_lo %*% rho_lo_U_vec,scientific=F)

w_hi %*% rho_hi_L_vec
w_hi %*% rho_hi_U_vec

```

```
76
77 ### ------------------------------------------------
78 ## --- Computing d lambda
79
80 # Find d lambdas
81 inner_lo_L <- rho_lo_L_vec %*% w_lo
82 inner_hi_L <- rho_hi_L_vec %*% w_hi
83
84 inner_lo_U <- rho_lo_U_vec %*% w_lo
85 inner_hi_U <- rho_hi_U_vec %*% w_hi
86
87 inner_lo <- v_lo %*% w_lo
88 inner_hi <- v_hi %*% w_hi
89
90 dlam_lo_L <- v_lo[1] * inner_lo_L / inner_lo
91 dlam_hi_L <- v_hi[1] * inner_hi_L / inner_hi
92 dlam_lo_U <- v_lo[m] * inner_lo_U / inner_lo
93 dlam_hi_U <- v_hi[m] * inner_hi_U / inner_hi
94
95 # Print thems
96 dlam_lo_L
97 dlam_lo_U
98 dlam_hi_L
99 dlam_hi_U
100
101 dlam_lo_L + dlam_lo_U
102 dlam_hi_L + dlam_hi_U
```

A.2.2 Form of Growth

```
1 ### ------------------------------------------------
2 ## --- Table 2.2: Forms for growth (and first paragraph)
3
4 # Functions for the forms
5 quad_fun <- function(x, coefs) {
6   a<-coefs[1]; b<-coefs[2]; return(a*x^2+b*x)}
```

```
7  quad3_fun <- function(x, coefs) {
8    a<-coefs[1]; b<-coefs[2]; c<-coefs[3]; return(a*x^2+b*x+c)}
9  rick_fun <- function(x, coefs) {
10   a<-coefs[1]; b<-coefs[2]; return(a*x*exp(-b*x))}
11 skel_fun <- function(x, coefs) {
12   a<-coefs[1]; b<-coefs[2]; return(a*(1-exp(-b*x)))}
13 bvht_fun <- function(x, coefs) {
14   a<-coefs[1]; b<-coefs[2]; return(a*x/(b+x))}
15 powr_fun <- function(x,coefs) {
16   a<-coefs[1]; b<-coefs[2]; return(a*x^b)}
17
18 # Fit the fits
19 dat_asymp <- dat_adult_low
20 quad_fit <- nls(h1 ~ quad_fun(h,c(a,b)), data=dat_asymp, start=
       list(a=1,b=1))
21 quad3_fit<- lm(h1~poly(h,2), data=dat_asymp)
22 rick_fit <- nls(h1 ~ rick_fun(h,c(a,b)), data=dat_asymp, start=
       list(a=1,b=0.01))
23 skel_fit <- nls(h1 ~ skel_fun(h,c(a,b)), data=dat_asymp, start=
       list(a=100,b=0.01))
24 bvht_fit <- nls(h1 ~ bvht_fun(h,c(a,b)), data=dat_asymp, start=
       list(a=1,b=1))
25 powr_fit <- nls(h1 ~ powr_fun(h,c(a,b)), data=dat_asymp, start=
       list(a=1,b=1))
26
27 # AIC for the above fits
28 AIC(grow_fit_low)
29 AIC(quad_fit)
30 AIC(quad3_fit)
31 AIC(rick_fit)
32 AIC(skel_fit)
33 AIC(bvht_fit)
34 AIC(powr_fit)
35
36 # Fit some more fits
37 dat_asymp <- dat_adult_high
38 quad_fit <- nls(h1 ~ quad_fun(h,c(a,b)), data=dat_asymp, start=
       list(a=1,b=1))
```

```
39 quad3_fit<- lm(h1~poly(h,2), data=dat_asymp)
40 rick_fit <- nls(h1 ~ rick_fun(h,c(a,b)), data=dat_asymp, start=
       list(a=1,b=0.01))
41 skel_fit <- nls(h1 ~ skel_fun(h,c(a,b)), data=dat_asymp, start=
       list(a=100,b=0.01))
42 bvht_fit <- nls(h1 ~ bvht_fun(h,c(a,b)), data=dat_asymp, start=
       list(a=1,b=1))
43 powr_fit <- nls(h1 ~ powr_fun(h,c(a,b)), data=dat_asymp, start=
       list(a=1,b=1))
44
45 # AIC for even more fits
46 AIC(grow_fit_high)
47 AIC(quad_fit)
48 AIC(quad3_fit)
49 AIC(rick_fit)
50 AIC(skel_fit)
51 AIC(bvht_fit)
52 AIC(powr_fit)
```

A.2.3 Normality of Residuals and Homoskedasticity

```
1 ### ------------------------------------------------
2 ## -- Normality of residuals and heteroskedasticity (second
       growth paragraph)
3
4 # Low dens diagnostics
5 plot(dat_adult_low$h,dat_adult_low$h1)
6 plot(dat_adult_low$h,grow_fit_low$residuals)
7 plot(dat_adult_low$h,hat(model.matrix(grow_fit_low)))
8 plot(dat_adult_low$h,cooks.distance(grow_fit_low))
9 qqnorm(grow_fit_low$residuals)
10 shapiro.test(grow_fit_low$residuals)
11 bptest(grow_fit_low)
12 AIC(grow_fit_low)
13
14 # High dens diagnostics
```

```
plot(dat_adult_high$h,dat_adult_high$h1)
plot(dat_adult_high$h,grow_fit_high$residuals)
plot(dat_adult_high$h,hat(model.matrix(grow_fit_high)))
plot(dat_adult_high$h,cooks.distance(grow_fit_high))
qqnorm(grow_fit_high$residuals)
shapiro.test(grow_fit_high$residuals)
bptest(grow_fit_high)
AIC(grow_fit_high)
```

A.2.4 Log-normality of the Recruit Size Distribution

```
### ------------------------------------------------
## -- Test fit of recruit dist'n

df <- seedlingdata$h1
df_lo <- seedlingdata[seedlingdata$dens=="Low",]$h1
df_hi <- seedlingdata[seedlingdata$dens=="High",]$h1

t.test(log(df_lo),log(df_hi), paired=F,var.equal=F, conf.level
    =0.95)
shapiro.test(log(df))
shapiro.test(log(df_lo))
shapiro.test(log(df_hi))
sd(log(df_lo))
sd(log(df_hi))
```

A.3 Bootstrapping

A.3.1 Bootstrap population growth rate

```
bootstrap_helper <- function(adult_dat,seedling_dat) {

  ### ------------------------------------------------
  ## -- The data set to perform regression on
```

```
5
6   # Which subset of data to use
7   dat <- adult_dat
8
9   # Use this one for growth
10  #dat_adult <- dat[dat$Stage2003!=1 & dat$Stage2003!=4 & dat$
        Stage2003!=5,]
11  dat_adult <- dat[dat$Stage2003!=4 & dat$Stage2003!=5,]
12
13  ### -----------------------------------------------
14  ## -- Survival and reproduction
15
16  # Logistic regression
17  surv_fit <- glm(survived~h, data=dat, family="binomial")
18  repr_fit <- glm(reproduced~h, data=dat, family="binomial")
19
20  # Response functions
21  p_s <- function(h){return(predict(surv_fit, data.frame(h=h),
        type="response"))}
22  p_r <- function(h){return(predict(repr_fit, data.frame(h=h),
        type="response"))}
23
24  ### -----------------------------------------------
25  ## -- Growth
26
27  # Fit growth
28  grow_fit <- lm(h1~h, data=dat_adult)
29
30  # Growth kernel
31  g_k <- function(h1,h) {
32    h1bar <- predict(grow_fit, newdata=data.frame(h=h), type="
        response")
33    sdhat <- sqrt(sum(residuals(grow_fit)^2)/df.residual(grow_fit
        ))
34    return(dnorm(h1, mean=h1bar, sd=sdhat))}
35
36  ### -----------------------------------------------
37  ## -- Recruitment kernel
```

```
38
39  # Expected number of offspring (num_new / num_flowering)
40  expected_offspring <- nrow(seedling_dat)/nrow(dat[dat$Stage2002
        ==4,])
41
42  # Fit recruit size to a log-normal distribution
43  r_size <- fitdistr(seedling_dat$Height2003, "lognormal")$
        estimate
44
45  # Function for probability of recruit size
46  recruit_size <- function(zvec) {return(dlnorm(zvec, meanlog=r_
        size["meanlog"], sdlog=r_size["sdlog"], log=F))}
47
48  ### ---------------------------------------------
49  ## -- Implement kernels
50
51  # Survival-growth
52  P_k <- function(h,meshpts) {
53    fxn <- function(z1,z) {return(p_s(z)*(1-p_r(z))*g_k(z1,z))}
54    return(h*(outer(meshpts,meshpts,fxn)))}
55
56  # Reproduction
57  R_k <- function(h,meshpts,dens,combine_repr) {
58    fxn <- function(z1,z) {return(p_s(z)*p_r(z)*expected_
          offspring*recruit_size(z1))}
59    return(h*(outer(meshpts,meshpts,fxn)))}
60
61  # Full kernel
62  ipm_kernel <- function(h,meshpts) {return(P_k(h,meshpts)+R_k(h,
        meshpts))}
63
64  ### ---------------------------------------------
65  ## -- Find growth rate
66
67  # Specify bounds, mesh, spacing
68  L <- 5
69  U <- 250
70  m <- 100
```

```
71    h <- (U-L)/m
72    meshpts <- L + (1:m)*h - h/2
73
74    # Determine kernels and eigenvectors
75    P <- P_k(h,meshpts)
76    R <- R_k(h,meshpts)
77    kernel <- P+R
78    fund_mat <- R %*% solve(diag(m)-P)
79
80    w <- Re(eigen(kernel)$vectors[,1])
81    w <- w/sum(w)
82
83    # Determine values to report
84    height <- w %*% meshpts
85    lambda <- Re(eigen(kernel)$values[1])
86    R0 <- Re(eigen(fund_mat)$values[1])
87    Tval <- log(R0)/log(lambda)
88    return(list("height"=height,"lambda"=lambda,"R0"=R0,"Tval"=Tval
          ))
89  }
90
91
92  ### ---------------------------------------------------
93  ## -- Function to verify if subset is ok for bootstrapping
94
95  verify_sample <- function(dat) {
96    # By default it should be fine, I just need to check if there
          might be a problem
97    out <- 0
98    # These subsets will help, I remove "NA's"
99    seed_dat <- dat[dat$class=="Seedling",]
100   adlt_dat <- dat[dat$class!="Seedling" & !is.na(dat$survived),]
101   # More helpful subsets: first digit is survival, second is
          reproduction
102   sr_dat_00 <- adlt_dat[adlt_dat$survived==0,]
103   sr_dat_10 <- adlt_dat[adlt_dat$survived==1 & adlt_dat$
          reproduced==0,]
```

```
104     sr_dat_11 <- adlt_dat[adlt_dat$survived==1 & adlt_dat$
          reproduced==1,]
105     # Need to have enough seedlings to parameterize
106     if       (nrow(seed_dat)<=2) {out <- NA}
107     # Return whether the data set is ok or not
108     return(out)
109   }
110
111
112   ### ---------------------------------------------
113   ## -- Function for bootstrapping
114
115   bootstrap_main <- function(dat, sample_size, N) {
116     # Number of rows to sample from
117     nr <- nrow(dat)
118     # Vector to store output
119     heights <- array(NA,dim=c(N))
120     lambdas <- array(NA,dim=c(N))
121     R0s <- array(NA,dim=c(N))
122     Ts <- array(NA,dim=c(N))
123     # Index
124     i = 1
125     j = 0
126     # Bootstrap step
127     while (i <= N) {
128       # Randomly sample from data set
129       indices <- sample.int(nr,sample_size,replace=T)
130       dat_new <- dat[indices,]
131       verf <- verify_sample(dat_new)
132       # Check to see if data will work
133       if (!is.na(verf)) {
134         out <- bootstrap_helper(dat_new[dat_new$class!="Seedling"
              ,],dat_new[dat_new$class=="Seedling",])
135         heights[i] <- out$height
136         lambdas[i] <- out$lambda
137         R0s[i] <- out$R0
138         Ts[i] <- out$Tval
139         print(i)
```

```
140         i <- i+1
141       }
142       else {
143         print(c(i,"bad"))
144         j <- j+1
145       }
146     }
147     print(c("There were",j,"errors"))
148     return(list("heights"=heights,"lambdas"=lambdas,"R0s"=R0s,"Ts"=
          Ts))
149 }
150
151
152 ### ---------------------------------------------
153 ## -- Bootstrap
154
155 # Subset data into high and low
156 dat_lo <- hmdata_no_outlier[hmdata_no_outlier$dens=="Low",]
157 dat_hi <- hmdata_no_outlier[hmdata_no_outlier$dens=="High",]
158
159 # Compute the bootstrap confidence intervals
160 boot_distn_lo <- bootstrap_main(dat_lo,nrow(dat_lo),5000)
161 boot_distn_hi <- bootstrap_main(dat_lo,nrow(dat_hi),5000)
162
163 # Break into constituent parts
164 h_distn_lo <- boot_distn_lo$heights
165 h_distn_hi <- boot_distn_hi$heights
166
167 lam_distn_lo <- boot_distn_lo$lambdas
168 lam_distn_hi <- boot_distn_hi$lambdas
169
170 # Ensure each value is valid
171 length(lam_distn_lo[lam_distn_lo<0])
172 length(lam_distn_hi[lam_distn_hi<0])
173 length(lam_distn_lo[is.na(lam_distn_lo)])
174 length(lam_distn_hi[is.na(lam_distn_hi)])
175
176 # Bootstrap means
```

```
177 h_boot_mu_lo <- mean(h_distn_lo)
178 h_boot_mu_hi <- mean(h_distn_hi)
179
180 lam_boot_mu_lo <- mean(lam_distn_lo)
181 lam_boot_mu_hi <- mean(lam_distn_hi)
182
183 # Bootstrap confidence intervals
184 h_boot_ci_hi <- quantile(h_distn_hi,probs=c(0.025,0.975))
185 h_boot_ci_lo <- quantile(h_distn_lo,probs=c(0.025,0.975))
186
187 lam_boot_ci_hi <- quantile(lam_distn_hi,probs=c(0.025,0.975))
188 lam_boot_ci_lo <- quantile(lam_distn_lo,probs=c(0.025,0.975))
189
190
191 ### ------------------------------------------------
192 ## -- Height table
193
194 # Using normal bootstrapping sample size
195 w_lo %*% meshpts
196 h_boot_mu_lo
197 h_boot_ci_lo
198
199 w_hi %*% meshpts
200 h_boot_mu_hi
201 h_boot_ci_hi
202
203
204 ### ------------------------------------------------
205 ## -- Population growth rate table
206
207 # Using normal bootstrapping sample size
208 lam_lo
209 lam_boot_mu_lo
210 lam_boot_ci_lo
211
212 lam_hi
213 lam_boot_mu_hi
214 lam_boot_ci_hi
```

```
215
216
217 ### ------------------------------------------------------
218 ## --- Upsampling data
219
220 # Bootstrap
221 boot_distn_lo_100 <- bootstrap_main(dat_lo,100,5000)
222 boot_distn_lo_200 <- bootstrap_main(dat_lo,200,5000)
223 boot_distn_lo_300 <- bootstrap_main(dat_lo,300,5000)
224 boot_distn_lo_400 <- bootstrap_main(dat_lo,400,5000)
225 boot_distn_lo_500 <- bootstrap_main(dat_lo,500,5000)
226 boot_distn_lo_600 <- bootstrap_main(dat_lo,600,5000)
227
228 boot_distn_hi_100 <- bootstrap_main(dat_hi,100,5000)
229 boot_distn_hi_200 <- bootstrap_main(dat_hi,200,5000)
230 boot_distn_hi_300 <- bootstrap_main(dat_hi,300,5000)
231 boot_distn_hi_400 <- bootstrap_main(dat_hi,400,5000)
232 boot_distn_hi_500 <- bootstrap_main(dat_hi,500,5000)
233 boot_distn_hi_600 <- bootstrap_main(dat_hi,600,5000)
234
235 # Give name to bootstrap distributions
236 lam_distn_lo_100 <- boot_distn_lo_100$lambdas
237 lam_distn_lo_200 <- boot_distn_lo_200$lambdas
238 lam_distn_lo_300 <- boot_distn_lo_300$lambdas
239 lam_distn_lo_400 <- boot_distn_lo_400$lambdas
240 lam_distn_lo_500 <- boot_distn_lo_500$lambdas
241 lam_distn_lo_600 <- boot_distn_lo_600$lambdas
242
243 lam_distn_hi_100 <- boot_distn_hi_100$lambdas
244 lam_distn_hi_200 <- boot_distn_hi_200$lambdas
245 lam_distn_hi_300 <- boot_distn_hi_300$lambdas
246 lam_distn_hi_400 <- boot_distn_hi_400$lambdas
247 lam_distn_hi_500 <- boot_distn_hi_500$lambdas
248 lam_distn_hi_600 <- boot_distn_hi_600$lambdas
249
250 # Compute bootstrap means
251 lam_boot_mu_lo_100 <- mean(lam_distn_lo_100)
252 lam_boot_mu_lo_200 <- mean(lam_distn_lo_200)
```

```
lam_boot_mu_lo_300 <- mean(lam_distn_lo_300)
lam_boot_mu_lo_400 <- mean(lam_distn_lo_400)
lam_boot_mu_lo_500 <- mean(lam_distn_lo_500)
lam_boot_mu_lo_600 <- mean(lam_distn_lo_600)

lam_boot_mu_hi_100 <- mean(lam_distn_hi_100)
lam_boot_mu_hi_200 <- mean(lam_distn_hi_200)
lam_boot_mu_hi_300 <- mean(lam_distn_hi_300)
lam_boot_mu_hi_400 <- mean(lam_distn_hi_400)
lam_boot_mu_hi_500 <- mean(lam_distn_hi_500)
lam_boot_mu_hi_600 <- mean(lam_distn_hi_600)

# Compute bootstrap CIs
lam_boot_ci_lo_100 <- quantile(lam_distn_lo_100,probs=c
    (0.025,0.975))
lam_boot_ci_lo_200 <- quantile(lam_distn_lo_200,probs=c
    (0.025,0.975))
lam_boot_ci_lo_300 <- quantile(lam_distn_lo_300,probs=c
    (0.025,0.975))
lam_boot_ci_lo_400 <- quantile(lam_distn_lo_400,probs=c
    (0.025,0.975))
lam_boot_ci_lo_500 <- quantile(lam_distn_lo_500,probs=c
    (0.025,0.975))
lam_boot_ci_lo_600 <- quantile(lam_distn_lo_600,probs=c
    (0.025,0.975))

lam_boot_ci_hi_100 <- quantile(lam_distn_hi_100,probs=c
    (0.025,0.975))
lam_boot_ci_hi_200 <- quantile(lam_distn_hi_200,probs=c
    (0.025,0.975))
lam_boot_ci_hi_300 <- quantile(lam_distn_hi_300,probs=c
    (0.025,0.975))
lam_boot_ci_hi_400 <- quantile(lam_distn_hi_400,probs=c
    (0.025,0.975))
lam_boot_ci_hi_500 <- quantile(lam_distn_hi_500,probs=c
    (0.025,0.975))
lam_boot_ci_hi_600 <- quantile(lam_distn_hi_600,probs=c
    (0.025,0.975))
```

```
279
280 # Print values
281 lam_boot_mu_lo_100
282 lam_boot_ci_lo_100
283
284 lam_boot_mu_lo_200
285 lam_boot_ci_lo_200
286
287 lam_boot_mu_lo_300
288 lam_boot_ci_lo_300
289
290 lam_boot_mu_lo_400
291 lam_boot_ci_lo_400
292
293 lam_boot_mu_lo_500
294 lam_boot_ci_lo_500
295
296 lam_boot_mu_lo_600
297 lam_boot_ci_lo_600
298
299
300 lam_boot_mu_hi_100
301 lam_boot_ci_hi_100
302
303 lam_boot_mu_hi_200
304 lam_boot_ci_hi_200
305
306 lam_boot_mu_hi_300
307 lam_boot_ci_hi_300
308
309 lam_boot_mu_hi_400
310 lam_boot_ci_hi_400
311
312 lam_boot_mu_hi_500
313 lam_boot_ci_hi_500
314
315 lam_boot_mu_hi_600
316 lam_boot_ci_hi_600
```

A.4 Methods Plots

A.4.1 Plot vital rate functions

```
### ------------------------------------------------
## --- Plot vital rates

# Specify data set to use
dat <- adultdata_no_outlier

# Real data to plot (ticks or points as necessary)
#  Survival ticks
dat_surv <- dat[!is.na(dat$survived),]
dat_surv_lo <- dat_surv[dat_surv$dens=="Low",]
dat_surv_hi <- dat_surv[dat_surv$dens=="High",]
#  Reproduction ticks
dat_repr <- dat[!is.na(dat$reproduced),]
dat_repr_lo <- dat_repr[dat_repr$dens=="Low",]
dat_repr_hi <- dat_repr[dat_repr$dens=="High",]
#  Growth points
dat_veg <- dat[dat$Stage2003!=4 & dat$Stage2003!=5 & !is.na(dat$h
    ),]
dat_veg_lo <- dat_veg[dat_veg$dens=="Low",]
dat_veg_hi <- dat_veg[dat_veg$dens=="High",]

# Make lines that will be plotted
xmax_lo <- 150
xmax_hi <- 220

x_lo <- seq(0,xmax_lo, length=100)
x_hi <- seq(0,xmax_hi, length=100)

s_lo <- p_s(x_lo,"Low")
s_hi <- p_s(x_hi,"High")
```

```
30
31 r_lo <- p_r(x_lo,"Low")
32 r_hi <- p_r(x_hi,"High")
33
34 # PLOTS
35 par(mfrow=c(3,2))
36 dx <- 0.03
37 dy <- 1
38 line_frac <- 1
39 scale_text <- 1.3
40
41 plot(x_lo,s_lo, type="l",lwd=2, xlim=c(0,xmax_lo),ylim=c(0,1),
42       xlab="Height of plant (cm)",ylab="Probability of survival",
          cex.lab=scale_text,cex.axis=scale_text)
43 points(dat_surv_lo$h,dat_surv_lo$survived,pch="|",cex=line_frac)
44 mtext("(a)",side=3,adj=dx,padj=dy,line=-1.3)
45 plot(x_hi,s_hi, type="l",lwd=2, xlim=c(0,xmax_hi),ylim=c(0,1),
46       xlab="Height of plant (cm)",ylab="Probability of survival",
          cex.lab=scale_text,cex.axis=scale_text)
47 points(dat_surv_hi$h,dat_surv_hi$survived,pch="|",cex=line_frac)
48 mtext("(b)",side=3,adj=dx,padj=dy,line=-1.3)
49
50 plot(x_lo,r_lo, type="l",lwd=2, xlim=c(0,xmax_lo),ylim=c(0,1),
51       xlab="Height of plant (cm)",ylab="Probability of flowering",
          cex.lab=scale_text,cex.axis=scale_text)
52 points(dat_repr_lo$h,dat_repr_lo$reproduced,pch="|",cex=line_frac
    )
53 mtext("(c)",side=3,adj=dx,padj=dy,line=-1.3)
54 plot(x_hi,r_hi, type="l",lwd=2, xlim=c(0,xmax_hi),ylim=c(0,1),
55       xlab="Height of plant (cm)",ylab="Probability of flowering",
          cex.lab=scale_text,cex.axis=scale_text)
56 points(dat_repr_hi$h,dat_repr_hi$reproduced,pch="|",cex=line_frac
    )
57 mtext("(d)",side=3,adj=dx,padj=dy,line=-1.3)
58
59 plot(dat_veg_lo$h,dat_veg_lo$h1, pch=16, xlim=c(0,xmax_lo),ylim=c
    (0,220),
```

```
60        xlab="Height in 2002 (cm)", ylab="Height in 2003 (cm)",cex.
              lab=scale_text,cex.axis=scale_text)
61  segments(min(dat_veg_lo$h), y0=as.numeric(predict(grow_fit_low,
        data.frame(h=min(dat_veg_lo$h)))),
62            max(dat_veg_lo$h), y1=as.numeric(predict(grow_fit_low,
                  data.frame(h=max(dat_veg_lo$h)))),lwd=2)
63  mtext("(e)",side=3,adj=dx,padj=dy,line=-1.3)
64  plot(dat_veg_hi$h,dat_veg_hi$h1, pch=16, xlim=c(0,xmax_hi),ylim=c
        (0,220),
65        xlab="Height in 2002 (cm)", ylab="Height in 2003 (cm)",cex.
              lab=scale_text,cex.axis=scale_text)
66  segments(min(dat_veg_hi$h), y0=as.numeric(predict(grow_fit_high,
        data.frame(h=min(dat_veg_hi$h)))),
67            max(dat_veg_hi$h), y1=as.numeric(predict(grow_fit_high,
                  data.frame(h=max(dat_veg_hi$h)))),lwd=2)
68  mtext("(f)",side=3,adj=dx,padj=dy,line=-1.3)
69
70  par(mfrow=c(1,1))
```

A.4.2 Plot recruit size distribution

```
1  ### ------------------------------------------------
2  ## --- Plot recruit size distn
3
4  # Specify data set to use
5  dat <- seedlingdata_no_outlier
6  dat_recr_lo <- dat[dat$dens=="Low",]$Height2003
7  dat_recr_hi <- dat[dat$dens=="High",]$Height2003
8
9  # Make lines to plot
10 xmin_lo <- min(dat_recr_lo)
11 xmax_lo <- max(dat_recr_lo)
12 x_lo <- seq(xmin_lo,xmax_lo, length=100)
13 y_lo <- recruit_size(x_lo,"Low")
14
15 xmin_hi <- min(dat_recr_hi)
```

```
xmax_hi <- max(dat_recr_hi)
x_hi <- seq(xmin_hi,xmax_hi, length=100)
y_hi <- recruit_size(x_hi,"High")

# PLOTS
par(mfrow=c(1,2))
distnce <- 0.02 # -0.22
scale_text <- 1
hist(dat_recr_lo,breaks=7, xlim=c(xmin_lo-5,xmax_lo),ylim=c(0,0.05),
     xlab="Recruit height (cm)",ylab="Proportion of recruits",
     cex.lab=scale_text,cex.axis=scale_text,main="",freq=FALSE)
lines(x_lo,y_lo,type="l",lwd=2)
mtext("(a)",side=3,adj=distnce,line=-1.3)
hist(dat_recr_hi,breaks=9, xlim=c(xmin_hi-2,xmax_hi),ylim=c(0,0.02),
     xlab="Recruit height (cm)",ylab="Proportion of recruits",
     cex.lab=scale_text,cex.axis=scale_text,main="",freq=FALSE)
lines(x_hi,y_hi,type="l",lwd=2)
mtext("(b)",side=3,adj=distnce,line=-1.3)
par(mfrow=c(1,1))
```

A.5 Results Plots

A.5.1 Plot predicted size distribution vs observations

```
# Main data frames
dat_a <- adultdata_no_outlier[adultdata_no_outlier$Stage2003!=4,]
dat_s <- seedlingdata_no_outlier

# Neat little data frames
adlt_lo <- dat_a[dat_a$dens=="Low",]
adlt_hi <- dat_a[dat_a$dens=="High",]
seed_lo <- dat_s[dat_s$dens=="Low",]
seed_hi <- dat_s[dat_s$dens=="High",]

```

```
11 # Isolate heights w/o NAs
12 h_adlt_lo <- adlt_lo$h1[!is.na(adlt_lo$h1)]
13 h_seed_lo <- seed_lo$h1[!is.na(seed_lo$h1)]
14 h_adlt_hi <- adlt_hi$h1[!is.na(adlt_hi$h1)]
15 h_seed_hi <- seed_hi$h1[!is.na(seed_hi$h1)]
16
17 # Combine adult and seedling data
18 observed_lo <- c(h_adlt_lo,h_seed_lo)
19 observed_hi <- c(h_adlt_hi,h_seed_hi)
20
21 # Predicted values
22 predicted_lo <- w_lo/h
23 predicted_hi <- w_hi/h
24
25 # Mean heights
26 mean_h_lo_prd <- w_lo %*% meshpts
27 mean_h_hi_prd <- w_hi %*% meshpts
28 mean_h_lo_obs <- mean(observed_lo)
29 mean_h_hi_obs <- mean(observed_hi)
30
31 mean_h_lo_obs
32 mean_h_lo_prd
33 mean_h_hi_obs
34 mean_h_hi_prd
35
36 # Values are found by running 'ch2_bootstrapping.R' and added
      manually
37 #  I do not do this programmatically since running the
38 #  bootstrap file every time I want to make this fig
39 boot_summary_lo <- c(86.65732, 94.60086, 102.26890)
40 boot_summary_hi <- c(67.99328, 73.05466,  78.14434)
41
42 # PLOTS
43 par(mfrow=c(1,2))
44 distnce <- 0.02
45 scale_text <- 1
46 hist(observed_lo,breaks=10,freq=F,xlim=c(0,U),
47       xlab="Height", ylab="Proportion",main="",
```

```
48        cex.lab=scale_text,cex.axis=scale_text)
49 lines(meshpts,predicted_lo,lwd=2)
50 mtext("(a)",side=3,adj=distnce,line=-1.3)
51 hist(observed_hi,breaks=10,freq=F,xlim=c(0,U),
52        xlab="Height", ylab="Proportion",main="",
53        cex.lab=scale_text,cex.axis=scale_text)
54 lines(meshpts,predicted_hi,lwd=2)
55 mtext("(b)",side=3,adj=distnce,line=-1.3)
56 par(mfrow=c(1,1))
```

Appendix B

Chapter 3 Code

B.1 Simulation Function

```
# Import libraries
library(lme4)        # for mixed-effects modelling
library(MASS)        # for fitting negative binomials
library(lmtest)      # for performing likelihood ratio tests


### ------------------------------------------------
## -- Read in the data

# The same data as was used in Chapter 2
hmdata <- read.csv("./data/HuelsNewData.csv", na.strings="␣")

# The number of seedlings per plot
seedling_numbers <- read.csv("./data/HuelsSeedlingData.csv", na.
    strings="␣")


### ------------------------------------------------
## -- Define state variable

# I use height as the state variable
```

```
21 hmdata["h"]  <- hmdata[["Height2002"]]
22 hmdata["h1"] <- hmdata[["Height2003"]]
23
24
25 ### ---------------------------------------------------
26 ## -- Create columns for site and plot
27
28 # The site is the first letter of the ID
29 hmdata["site"] <- substr(hmdata$ID,1,1) # first letter of
       individual ID
30
31 # Determine the plot number
32 hmdata["plot"] <- substr(hmdata$ID,1,as.numeric(lapply(as.
       character(hmdata$ID),nchar))-2)
33
34
35 ### ---------------------------------------------------
36 ## -- Remove bad data
37
38 # Remove Allendorf data due to grazing damage
39 hmdata <- hmdata[hmdata$site!="A",]
40
41
42 ### ---------------------------------------------------
43 ## -- Find biomass in each plot
44
45 # NOTES:
46 #  mass_exp: "Global allocation rules for patterns of biomass
       partitioning in seed plants"
47 #  I divide by 100 to convert to meters (so the numbers are a
       reasonable size)
48
49 # Estimate biomass of each plant
50 euclid <- 3     # exponent=1
51 wbe <- 4        # one-fourth law of West, Brown, and Enquist
52 otl <- 9/2      # one-third law
53 upper_exp <- 6 # max value
54 mass_exp <- wbe
```

```
55
56 hmdata["mass"]  <- ifelse(!is.na(hmdata$h),(hmdata["h"]/100)^mass
       _exp,0)
57 hmdata["mass1"] <- ifelse(!is.na(hmdata$h1),(hmdata["h1"]/100)^
       mass_exp,0)
58
59 # Find the density of each plot based on biomass
60 dens_df <- aggregate(mass ~ plot, hmdata, sum)
61 names(dens_df)[names(dens_df)=="mass"] <- "dens"
62 dens1_df <- aggregate(mass1 ~ plot, hmdata, sum)
63 names(dens1_df)[names(dens1_df)=="mass1"] <- "dens1"
64
65 # Merging the data frames to get seedlings/plot and biomass/plot
66 hmdata <- merge(hmdata, seedling_numbers, by="plot")
67 hmdata <- merge(hmdata, dens_df,  by="plot")
68 hmdata <- merge(hmdata, dens1_df, by="plot")
69
70
71 ### ---------------------------------------------
72 ## -- Survival, reproduction, and classification (i.e. seedling
       or adult)
73
74 # 1 means survived, 0 means did not survive, NA means flowered (
       so of course it's dead)
75 hmdata["survived"]  <- ifelse(hmdata[["Stage2002"]] == 4, NA,
       ifelse(hmdata[["Stage2003"]] == 5, 0, 1))
76
77 # 1 means reproduced, 0 means did not reproduce, NA means
       individual died without reproducing
78 hmdata["reproduced"] <- ifelse(hmdata[["Stage2003"]] == 4, 1,
       ifelse(hmdata[["Stage2003"]] == 5, NA, 0))
79
80 # If a plant was recorded in 2003 but not 2002 it was a seedling
       in 2002
81 hmdata["class"] <- ifelse(is.na(hmdata[["Stage2002"]]), "Seedling
       ", "Adult")
82
83
```

```
84  ### ------------------------------------------------------------
85  ## -- Remove outliers
86
87  # Remove outliers (has to be done AFTER mass calcs)
88  hmdata <- hmdata[hmdata$ID!="KXIII09" & hmdata$ID!="VCI04",]
89
90  # Remove a reproductive outlier
91  hmdata <- hmdata[hmdata$reproduced==0 | hmdata$h1>100 | is.na(
        hmdata$reproduced),]
92
93
94  ### ------------------------------------------------------------
95  ## -- Split single data frame into one for seedlings and one for
        adult plants
96
97  seedlingdata <- hmdata[hmdata$class=="Seedling",]
98  hmdata       <- hmdata[hmdata$class!="Seedling",]
99
100
101 ### ------------------------------------------------------------
102 ## -- The data set to perform regression on
103
104 # Which subset of data to use
105 dat <- hmdata
106
107
108 ### ------------------------------------------------------------
109 ## -- Probability of survival
110
111 # Logistic regression for survival probability depending on leaf
        stem diameter
112 surv_fit <- glm(survived ~ h, data=dat, family="binomial")
113
114 # Survival function
115 p_s <- function(h) {
116   tempdf <- data.frame(h=h)
117   return(predict(surv_fit, tempdf, type="response"))
118 }
```

```
119
120
121 ### ---------------------------------------------------
122 ## --- Probability of reproduction
123
124 # Logistic regression for probability of reproduction depending
      on leaf stem diameter
125 repr_fit_rl <- glm(reproduced ~ h, data=dat, family="binomial")
126 repr_fit_ac <- glm(reproduced ~ h+dens, data=dat, family="
      binomial")
127
128 # Reproduction function
129 #  I have two cases since I have two models
130 p_r <- function(h, dens, modl) {
131   dens <- as.numeric(dens)
132   if (modl=="rl") {
133     tempdf <- data.frame(h=h)
134     my_fit <- repr_fit_rl}
135   else if (modl=="ac") {
136     tempdf <- data.frame(h=h, dens=dens)
137     my_fit <- repr_fit_ac}
138   return(predict(my_fit, tempdf, type="response"))
139 }
140
141
142 ### ---------------------------------------------------
143 ## --- Expected size increase (non-flowering plants)
144
145 # I need to exclude flowering plants and dead plants...
146 hmdata_adult <- hmdata[hmdata$Stage2003!=4 & hmdata$Stage2003!=
      5,]
147 # ...and don't forget seedlings
148 hmdata_adult <- hmdata_adult[!is.na(hmdata_adult$Height2002),]
149
150 # Fit growth using a linear model
151 grow_fit_a <- lm(h1~h, data=hmdata_adult)
152
153 # Define growth kernel
```

```
154 g_a <- function(h1,h) {
155   newdata <- data.frame(h=h)
156   h1bar   <- predict(grow_fit_a, newdata=newdata, type="response"
          )
157   res     <- residuals(grow_fit_a)
158   df.res  <- df.residual(grow_fit_a)
159   sse     <- sum(res^2)
160   sdhat   <- sqrt(sse/df.res)
161   return(dnorm(h1, mean=h1bar, sd=sdhat))
162 }
163
164
165 ### ------------------------------------------------
166 ## -- Expected size increase (flowering plants)
167
168 # I only include flowering plants
169 hmdata_flow <- hmdata[hmdata$Stage2003==4,]
170
171 # Fit growth using a linear model
172 grow_fit_r <- lm(h1~h, data=hmdata_flow)
173
174 # Define growth kernel
175 g_r <- function(h1,h) {
176   newdata <- data.frame(h=h)
177   h1bar   <- predict(grow_fit_r, newdata=newdata, type="response"
          )
178   res     <- residuals(grow_fit_r)
179   df.res  <- df.residual(grow_fit_r)
180   sse     <- sum(res^2)
181   sdhat   <- sqrt(sse/df.res)
182   return(dnorm(h1, mean=h1bar, sd=sdhat))
183 }
184
185
186 ### ------------------------------------------------
187 ## -- Number of recruits
188
```

```
189 # Number of seeds (from meta-analysis), num produced is indep of
    height
190 mean_seeds <- function(z) {
191   return(rep(17746,times=length(z)))
192 }
193
194 # Expected number of seeds per plot
195 seedling_numbers$Seeds2002 <- seedling_numbers$NumFlow2002*mean_
    seeds(1)
196 seedling_pos <- seedling_numbers[seedling_numbers$Seeds2002!=0,]
    # only plots that have seeds
197
198 # Seedling survival parameters
199 # max_seedlings <- 127.5 #max_seedlings <- mean(seedling_pos$
    Seedlings2003)
200 # prob_surv <- 0.32
201 alpha <- 127.5
202 beta  <- 273.4375
203
204 # Function to return expected number of seedlings given number of
     flowering plants
205 expected_seedlings <- function(ns) {
206   num_seedlings <- alpha*ns/(beta+ns)
207   return(num_seedlings)
208 }
209
210 per_capita_offspring <- function(ns) {
211   num_offspring <- alpha/(beta+ns)
212   return(num_offspring)
213 }
214
215
216 ### ------------------------------------------------
217 ## -- Recruit size
218
219 # Fit recruit size to a log-normal distribution
220 r_size <- fitdistr(seedlingdata$Height2003, "lognormal")$estimate
221
```

```
222 # Function for probability of recruit size
223 recruit_size <- function(zvec) {
224   return(dlnorm(zvec, meanlog=r_size["meanlog"], sdlog=r_size["
        sdlog"], log=F))
225 }
226
227
228 ### ---------------------------------------------
229 ## -- Implement kernels
230
231 # Survival-growth
232 P_k <- function(h,meshpts,dens,modl) {
233   fxn <- function(z1,z,params) {
234     dens <- params[1]; modl <- params[2]
235     return(p_s(z)*(1-p_r(z,dens,modl))*g_a(z1,z))}
236   params <- c(dens,modl)
237   return(h*(outer(meshpts,meshpts,fxn,params)))}
238
239 # Reproduction
240 R_k <- function(h,meshpts,dens,modl) {
241   fxn <- function(z1,z,params) {
242     dens <- params[1]; modl <- params[2]
243     return(p_s(z)*p_r(z,dens,modl)*g_r(z1,z))}
244   params <- c(dens,modl)
245   return(h*(outer(meshpts,meshpts,fxn,params)))}
246
247 # Fecundity
248 F_k <- function(h,recruit_dist,ns) {
249   seedlings <- expected_seedlings(ns)
250   offspring <- recruit_dist*seedlings#*h #(no need to multiply by
        h)
251   return(offspring)}
252
253 F_k_mod <- function(h,recruit_dist,ns) {
254   per_capita <- per_capita_offspring(ns)*mean_seeds(1)
255   matrix <- h*outer(recruit_dist,rep(per_capita,times=m),FUN="*")
256   return(matrix)
257 }
```

```
258
259 #  --     --  #
260 #  |  P  F  | #
261 #  |  R  0  | #
262 #  --     --  #
263
264
265 ### ------------------------------------------------
266 ## -- Functions to update state
267
268 # For both types of simulations
269
270 update_seeds <- function(h,nr) {
271   ns <- sum(nr)*h*mean_seeds(1)
272   return(ns)}
273
274 update_adults <- function(h,meshpts,recruit_dist,dens,nt,nr,modl)
      {
275   ns <- sum(nr)*h*mean_seeds(1)
276   recruits  <- F_k(h,recruit_dist,ns)
277   survivors <- P_k(h,meshpts,dens,modl) %*% nt
278   nt1 <- recruits + survivors
279   return(nt1)}
280
281 update_adults_ws <- function(h,meshpts,recruit_dist,dens,ns,nt,nr
    ,modl) {
282   #ns <- sum(nr)*h*mean_seeds(1)
283   recruits  <- F_k(h,recruit_dist,ns)
284   survivors <- P_k(h,meshpts,dens,modl) %*% nt
285   nt1 <- recruits + survivors
286   return(nt1)}
287
288 update_flowering <- function(h,meshpts,dens,nt,modl) {
289   nr <- R_k(h,meshpts,dens,modl) %*% nt
290   return(nr)}
291
292 update_density <- function(h,weights,nt,nr, area_frac,quality) {
293   if (is.na(quality) | sum(nt+nr)==0) {return(0)}
```

```
294     else {return((weights %*% (nt+nr)*h)/(area_frac*quality))}}
295
296
297 # Function to simulate population dynamics
298 master_local_simulation_ws <- function(L,U,m, num_steps, ns_init,
        nt_init,nr_init, modl) {
299
300   ### ------------------------------------------------
301   ## -- Initial processing
302
303   # Design mesh
304   h <- (U-L)/m
305   meshpts <- L + (1:m)*h - h/2
306   weights <- (meshpts/100)^mass_exp
307   recruit_dist <- recruit_size(meshpts)
308   print("Status: meshes made")
309
310   ### ------------------------------------------------
311   ## -- Run the simulation
312
313   # Initialize vectors
314   ns <- rep(0,times=num_steps+1)
315   nt <- array(0,dim=c(num_steps+1,m))
316   nr <- array(0,dim=c(num_steps+1,m))
317   dt <- rep(0,times=num_steps+1)
318
319   # Initial values
320   ns[1]  <- ns_init
321   nt[1,] <- nt_init
322   nr[1,] <- nr_init
323   dt[1]  <- update_density(h,weights,nt[1,],nr[1,], 1,1)
324   print("Status: simulation initialized")
325
326   # Loop through each time step
327   for (t in 1:num_steps) {
328     nt[t+1,] <- update_adults_ws(h,meshpts,recruit_dist,dt[t],ns[
          t],nt[t,],nr[t,], modl)
329     nr[t+1,] <- update_flowering(h,meshpts,dt[t],nt[t,], modl)
```

```
330       ns[t+1]  <- update_seeds(h,nr[t+1,])
331       dt[t+1]  <- update_density(h,weights,  nt[t+1,],nr[t+1,],  1,1)
332       print(ns[t+1])
333     }
334
335     # Return the density over time
336     return(list("adults"=nt,"flowering"=nr,"densities"=dt,"seeds"=
            ns))
337 }
```

B.2 Diagnostics

B.2.1 Form of Growth

```
1  # Library
2  library(lme4)
3
4  ### -----------------------------------------------
5  ## -- Other forms for growth
6
7  # Functions for the forms
8  quad_fun <- function(x, coefs) {
9    a<-coefs[1]; b<-coefs[2]; return(a*x^2+b*x)}
10 quad3_fun <- function(x, coefs) {
11   a<-coefs[1]; b<-coefs[2]; c<-coefs[3]; return(a*x^2+b*x+c)}
12 rick_fun <- function(x, coefs) {
13   a<-coefs[1]; b<-coefs[2]; return(a*x*exp(-b*x))}
14 skel_fun <- function(x, coefs) {
15   a<-coefs[1]; b<-coefs[2]; return(a*(1-exp(-b*x)))}
16 bvht_fun <- function(x, coefs) {
17   a<-coefs[1]; b<-coefs[2]; return(a*x/(b+x))}
18 powr_fun <- function(x,coefs) {
19   a<-coefs[1]; b<-coefs[2]; return(a*x^b)}
20
21 # Fit the fits
22 dat_asymp <- hmdata_adult[!is.na(hmdata_adult$Height2002),]
```

```
23 line_fit <- lm(h1 ~ h, data=dat_asymp)
24 quad_fit <- nls(h1 ~ quad_fun(h,c(a,b)), data=dat_asymp, start=
       list(a=1,b=1))
25 quad3_fit<- lm(h1~poly(h,2), data=dat_asymp)
26 rick_fit <- nls(h1 ~ rick_fun(h,c(a,b)), data=dat_asymp, start=
       list(a=1,b=0.01))
27 skel_fit <- nls(h1 ~ skel_fun(h,c(a,b)), data=dat_asymp, start=
       list(a=100,b=0.01))
28 bvht_fit <- nls(h1 ~ bvht_fun(h,c(a,b)), data=dat_asymp, start=
       list(a=1,b=1))
29 powr_fit <- nls(h1 ~ powr_fun(h,c(a,b)), data=dat_asymp, start=
       list(a=1,b=1))
30
31 # AIC for the above fits
32 round(AIC(line_fit))
33 round(AIC(quad_fit))
34 round(AIC(quad3_fit))
35 round(AIC(rick_fit))
36 round(AIC(skel_fit))
37 round(AIC(bvht_fit))
38 round(AIC(powr_fit))
39
40 # Fit some more fits
41 dat_asymp <- hmdata_flow
42 line_fit <- lm(h1 ~ h, data=dat_asymp)
43 quad_fit <- nls(h1 ~ quad_fun(h,c(a,b)), data=dat_asymp, start=
       list(a=1,b=1))
44 quad3_fit<- lm(h1~poly(h,2), data=dat_asymp)
45 rick_fit <- nls(h1 ~ rick_fun(h,c(a,b)), data=dat_asymp, start=
       list(a=1,b=0.01))
46 skel_fit <- nls(h1 ~ skel_fun(h,c(a,b)), data=dat_asymp, start=
       list(a=100,b=0.01))
47 bvht_fit <- nls(h1 ~ bvht_fun(h,c(a,b)), data=dat_asymp, start=
       list(a=1,b=1))
48 powr_fit <- nls(h1 ~ powr_fun(h,c(a,b)), data=dat_asymp, start=
       list(a=1,b=1))
49
50 # AIC for even more fits
```

```
round(AIC(line_fit))
round(AIC(quad_fit))
round(AIC(quad3_fit))
round(AIC(rick_fit))
round(AIC(skel_fit))
round(AIC(bvht_fit))
round(AIC(powr_fit))
```

B.2.2 Density-Dependence and Site-Specific Effects

```
### ------------------------------------------------
## --- Test density-dependence and site-specific effects

# Survival
surv_fit_orig <- glm(survived ~ h, data=dat, family="binomial")
surv_fit_dens <- glm(survived ~ h+dens, data=dat, family="
    binomial")
surv_fit_site <- glmer(survived ~ h+(1|h), data=dat, family="
    binomial")

AIC(surv_fit_orig)
AIC(surv_fit_dens)
AIC(surv_fit_site)

# Reproduction
repr_fit_orig <- glm(reproduced ~ h, data=dat, family="binomial")
repr_fit_dens <- glm(reproduced ~ h+dens, data=dat, family="
    binomial")
repr_fit_site <- glmer(reproduced ~ h+(1|h), data=dat, family="
    binomial")

AIC(repr_fit_orig)
AIC(repr_fit_dens)
AIC(repr_fit_site)
lrtest(repr_fit_orig,repr_fit_dens)

```

```
# Vegetative growth
grow_fit_a_orig <- lm(h1~h, data=hmdata_adult)
grow_fit_a_dens <- lm(h1~h+dens, data=hmdata_adult)
grow_fit_a_site <- lmer(h1~h+(1|h), data=hmdata_adult)

AIC(grow_fit_a_orig)
AIC(grow_fit_a_dens)
AIC(grow_fit_a_site)
lrtest(grow_fit_a_orig,grow_fit_a_dens)

# Vegetative growth
grow_fit_r_orig <- lm(h1~h, data=hmdata_flow)
grow_fit_r_dens <- lm(h1~h+dens, data=hmdata_flow)
grow_fit_r_site <- lmer(h1~h+(1|h), data=hmdata_flow)

AIC(grow_fit_r_orig)
AIC(grow_fit_r_dens)
AIC(grow_fit_r_site)
lrtest(grow_fit_r_orig,grow_fit_r_dens)
lrtest(grow_fit_r_orig,grow_fit_r_site)
```

B.2.3 Normality of Residuals and Homoskedasticity

```
### ------------------------------------------------
## -- Normality of residuals and heteroskedasticity

# For vegetative plants
plot(hmdata_adult$h,grow_fit_a$residuals)
shapiro.test(grow_fit_a$residuals)
bptest(grow_fit_a)

# For reproductive plants
plot(hmdata_flow$h,hmdata_flow$h1)
plot(hmdata_flow$h,grow_fit_r$residuals)
shapiro.test(grow_fit_r$residuals)
bptest(grow_fit_r)
```

B.2.4 Log-normality of the Recruit Size Distribution

```
### ------------------------------------------------
## -- Test recruit fit

shapiro.test(log(seedlingdata$Height2003))
sd(log(seedlingdata$Height2003))
qqnorm(log(seedlingdata$Height2003))
```

B.3 Bootstrapping

B.3.1 Bootstrap population growth rate

```
# Library to test if intervals overlap
library(DescTools)

# Function to simulate population given a bootstrap replicate
bootstrap_helper <- function(adult_dat,seedling_dat,modl,L,U,m) {

  ### ------------------------------------------------
  ## -- The data set to perform regression on

  # Which subset of data to use
  dat <- adult_dat

  # Use this one for growth
  dat_adult <- dat[dat$Stage2003!=4 & dat$Stage2003!=5,]
  dat_flow  <- dat[dat$Stage2003==4,]

  ### ------------------------------------------------
  ## -- Preamble to finding equilibria

  # Specify mesh and weights
  h <- (U-L)/m; meshpts <- L + (1:m)*h - h/2
  weights <- (meshpts/100)^mass_exp

```

```
24  ### ------------------------------------------------------
25  ## --- Survival and reproduction
26
27  # Survival
28  surv_fit <- glm(survived~h, data=dat, family="binomial")
29  p_s <- function(h) {return(predict(surv_fit, data.frame(h=h),
        type="response"))}
30
31  # Growth
32  if (modl=="rl") {
33    repr_fit <- glm(reproduced~h, data=dat, family="binomial")
34    p_r <- function(h,dens) {
35      return(predict(repr_fit, data.frame(h=h), type="response"))
          }
36  } else if (modl=="ac") {
37    repr_fit <- glm(reproduced~h+dens, data=dat, family="binomial
        ")
38    p_r <- function(h,dens) {
39      return(predict(repr_fit, data.frame(h=h,dens=dens), type="
          response"))}
40  }
41
42  ### ------------------------------------------------------
43  ## --- Growth
44
45  # Fit growth
46  grow_fit_a <- lm(h1~h, data=dat_adult)
47  grow_fit_r <- lm(h1~h, data=dat_flow)
48
49  # Growth kernel
50  g_a <- function(h1,h) {
51    h1bar <- predict(grow_fit_a, newdata=data.frame(h=h), type="
        response")
52    sdhat <- sqrt(sum(residuals(grow_fit_a)^2)/df.residual(grow_
        fit_a))
53    return(dnorm(h1, mean=h1bar, sd=sdhat))}
54  g_r <- function(h1,h) {
```

```
55      h1bar <- predict(grow_fit_r, newdata=data.frame(h=h), type="
            response")
56      sdhat <- sqrt(sum(residuals(grow_fit_r)^2)/df.residual(grow_
            fit_r))
57      return(dnorm(h1, mean=h1bar, sd=sdhat))}
58
59    ### ---------------------------------------------
60    ## -- Number of recruits
61
62    # Seedling survival parameters
63    # max_seedlings <- 127.5 #max_seedlings <- mean(seedling_pos$
          Seedlings2003)
64    # prob_surv <- 0.32
65    alpha <- 127.5
66    beta  <- 273.4375
67
68    # Function to return expected number of seedlings given number
          of flowering plants
69    expected_seedlings <- function(ns) {
70      num_seedlings <- alpha*ns/(beta+ns)
71      return(num_seedlings)
72    }
73
74    per_capita_offspring <- function(ns) {
75      num_offspring <- alpha/(beta+ns)
76      return(num_offspring)
77    }
78
79    # Fit recruit size to a log-normal distribution
80    r_size <- fitdistr(seedling_dat$Height2003, "lognormal")$
          estimate
81
82    # Function for probability of recruit size
83    recruit_size <- function(zvec) {return(dlnorm(zvec, meanlog=r_
          size["meanlog"], sdlog=r_size["sdlog"], log=F))}
84
85    ### ---------------------------------------------
86    ## -- Find kernels
```

```
87
88  # Survival-growth
89  P_k <- function(h,meshpts,dens) {
90    fxn <- function(z1,z,dens) {
91      return(p_s(z)*(1-p_r(z,dens))*g_a(z1,z))}
92    return(h*(outer(meshpts,meshpts,fxn,dens)))}
93
94  # Reproduction
95  R_k <- function(h,meshpts,dens) {
96    fxn <- function(z1,z,dens) {
97      return(p_s(z)*p_r(z,dens)*g_r(z1,z))}
98    return(h*(outer(meshpts,meshpts,fxn,dens)))}
99
100 # Fecundity
101 F_k <- function(h,recruit_dist,ns) {
102   seedlings <- expected_seedlings(ns)
103   offspring <- recruit_dist*seedlings#*h #(no need to multiply
        by h?)
104   return(offspring)}
105
106 ### ------------------------------------------------
107 ## -- Functions to update state
108
109 update_seeds <- function(h,nr) {
110   ns <- sum(nr)*h*mean_seeds(1)
111   return(ns)}
112
113 update_adults_ws <- function(h,meshpts,recruit_dist,dens,ns,nt,
      nr,modl) {
114   recruits  <- F_k(h,recruit_dist,ns)
115   survivors <- P_k(h,meshpts,dens) %*% nt
116   nt1 <- recruits + survivors
117   return(nt1)}
118
119 update_flowering <- function(h,meshpts,dens,nt,modl) {
120   nr <- R_k(h,meshpts,dens) %*% nt
121   return(nr)}
122
```

```
123  update_density <- function(h,weights,nt,nr, area_frac,quality)
       {
124    if (is.na(quality) | sum(nt+nr)==0) {return(0)}
125    else {return((weights %*% (nt+nr)*h)/(area_frac*quality))}}
126
127  ### ---------------------------------------------
128  ## -- Function to iterate
129
130  master_local_simulation_ws <- function(L,U,m, num_steps, ns_
       init,nt_init,nr_init, modl) {
131
132    # Design mesh
133    h <- (U-L)/m
134    meshpts <- L + (1:m)*h - h/2
135    weights <- (meshpts/100)^mass_exp
136    recruit_dist <- recruit_size(meshpts)
137    print("Status: meshes made")
138
139    # Initialize vectors
140    ns <- rep(0,times=num_steps+1)
141    nt <- array(0,dim=c(num_steps+1,m))
142    nr <- array(0,dim=c(num_steps+1,m))
143    dt <- rep(0,times=num_steps+1)
144
145    # Initial values
146    ns[1]  <- ns_init
147    nt[1,] <- nt_init
148    nr[1,] <- nr_init
149    dt[1]  <- update_density(h,weights,nt[1,],nr[1,], 1,1)
150
151    # Loop through each time step
152    for (t in 1:num_steps) {
153      nt[t+1,] <- update_adults_ws(h,meshpts,recruit_dist,dt[t],
         ns[t],nt[t,],nr[t,], modl)
154      nr[t+1,] <- update_flowering(h,meshpts,dt[t],nt[t,], modl)
155      ns[t+1]  <- update_seeds(h,nr[t+1,])
156      dt[t+1]  <- update_density(h,weights, nt[t+1,],nr[t+1,],
         1,1)
```

```
157       }
158
159       # Return the density over time
160       return(list("adults"=nt,"flowering"=nr,"densities"=dt,"seeds"
              =ns))
161     }
162
163     ### ---------------------------------------------------
164     ## -- Run the simulation
165
166     # Initialize
167     num_steps <- 30
168     ns_init <- 1
169     nt_init <- rep(0,length=m)
170     nr_init <- rep(0,length=m)
171
172     # Run simulations
173     sim_res <- master_local_simulation_ws(L,U,m, num_steps, ns_init
          ,nt_init,nr_init, modl)
174
175     ### ---------------------------------------------------
176     ## -- Return things
177
178     # Return the number of seeds and distribution of adult and
          flowering plants
179     return(list("seeds"=sim_res$seeds,"adults"=sim_res$adults,"
          flowering"=sim_res$flowering))
180 }
181
182
183 ### ---------------------------------------------------
184 ## -- Function for bootstrapping
185
186 bootstrap_main <- function(dat, N, modl) {
187   # Initialize things
188   L <- 5; U <- 550; m <- 50
189   h <- (U-L)/m; meshpts <- L + (1:m)*h - h/2
190   weights <- (meshpts/100)^mass_exp
```

```
191   # Number of rows to sample from
192   nr <- nrow(dat)
193   # Vectors to store initial seed values
194   seeds0 <- array(NA,dim=c(N))
195   seeds1 <- array(NA,dim=c(N))
196   seeds2 <- array(NA,dim=c(N))
197   seeds3 <- array(NA,dim=c(N))
198   seeds4 <- array(NA,dim=c(N))
199   seeds5 <- array(NA,dim=c(N))
200   seeds6 <- array(NA,dim=c(N))
201   seeds7 <- array(NA,dim=c(N))
202   # Vectors to store asymptotic seed values
203   seeds29 <- array(NA,dim=c(N))
204   seeds30 <- array(NA,dim=c(N))
205   # Vectors to store heights of vegetative plants, reproductive
          plants, and biomass
206   # NOTE: 's' means 'simulated' while 'o' means 'observed'
207   height_as29 <- array(NA,dim=c(N))
208   height_fs29 <- array(NA,dim=c(N))
209   height_as30 <- array(NA,dim=c(N))
210   height_fs30 <- array(NA,dim=c(N))
211   height_ao   <- array(NA,dim=c(N))
212   height_fo   <- array(NA,dim=c(N))
213   biomass     <- array(NA,dim=c(N))
214   # Index
215   i = 1
216   # Bootstrap step
217   while (i <= N) {
218     # Randomly sample from data set
219     indices <- sample.int(nr,nr,replace=T)
220     dat_new <- dat[indices,]
221     dat_adult <- dat_new[dat_new$Stage2003!=4 & dat_new$Stage2003
            !=5,]
222     dat_flow  <- dat_new[dat_new$Stage2003==4,]
223     # Do bootstrapping
224     out <- bootstrap_helper(dat_new[dat_new$class!="Seedling",],
            dat_new[dat_new$class=="Seedling",],modl,L,U,m)
225     # Give useful names to final versions
```

```
226     num_steps  <- length(out$seeds)
227     adlt_2last <- out$adults[num_steps-1,]/sum(out$adults[num_
            steps-1,])
228     flow_2last <- out$flowering[num_steps-1,]/sum(out$flowering[
            num_steps-1,])
229     adlt_last  <- out$adults[num_steps,]/sum(out$adults[num_steps
            ,])
230     flow_last  <- out$flowering[num_steps,]/sum(out$flowering[num
            _steps,])
231     # Get statistics
232     seeds0[i]  <- out$seeds[1]
233     seeds1[i]  <- out$seeds[2]
234     seeds2[i]  <- out$seeds[3]
235     seeds3[i]  <- out$seeds[4]
236     seeds4[i]  <- out$seeds[5]
237     seeds5[i]  <- out$seeds[6]
238     seeds6[i]  <- out$seeds[7]
239     seeds7[i]  <- out$seeds[8]
240     seeds29[i] <- out$seeds[num_steps-1]
241     seeds30[i] <- out$seeds[num_steps]
242     biomass[i] <- weights %*% (out$adults[num_steps,]+out$
            flowering[num_steps,]) * h
243     height_as29[i] <- meshpts %*% adlt_2last
244     height_fs29[i] <- meshpts %*% flow_2last
245     height_as30[i] <- meshpts %*% adlt_last
246     height_fs30[i] <- meshpts %*% flow_last
247     height_ao[i] <- mean(dat_adult$h1,na.rm=T)
248     height_fo[i] <- mean(dat_flow$h1)
249     print(i)
250     i <- i+1
251   }
252   return(list("seeds0"=seeds0,"seeds1"=seeds1,"seeds2","
          seeds3"=seeds3,
253                 "seeds4"=seeds4,"seeds5"=seeds5,"seeds6","
                      seeds7"=seeds7,
254                 "seeds29"=seeds29,"seeds30","biomass"=
                      biomass,
```

```
255                 "height_as29"=height_as29,"height_fs29"=height_fs29
                        ,
256                 "height_as30"=height_as30,"height_fs30"=height_fs30
                        ,
257                 "height_ao"=height_ao,"height_fo"=height_fo))
258 }
259
260
261 ### ------------------------------------------------
262 ## --- Run bootstrap simulations
263
264 # Subset data into high and low
265 dat <- rbind(hmdata,seedlingdata)
266
267 # Compute the bootstrap confidence intervals
268 #  Took 65 seconds when N=100 for each
269 tm <- proc.time()
270 boot_distn_rl <- bootstrap_main(dat,500,"rl")
271 boot_distn_ac <- bootstrap_main(dat,500,"ac")
272 print(proc.time()-tm)
273
274
275 ### ------------------------------------------------
276 ## --- Seed informations
277
278 # Get seed numbers
279 s0_distn_rl  <- boot_distn_rl$seeds0
280 s1_distn_rl  <- boot_distn_rl$seeds1
281 s2_distn_rl  <- boot_distn_rl$seeds2
282 s3_distn_rl  <- boot_distn_rl$seeds3
283 s4_distn_rl  <- boot_distn_rl$seeds4
284 s5_distn_rl  <- boot_distn_rl$seeds5
285 s6_distn_rl  <- boot_distn_rl$seeds6
286 s7_distn_rl  <- boot_distn_rl$seeds7
287 s29_distn_rl <- boot_distn_rl$seeds29
288 s30_distn_rl <- boot_distn_rl$seeds30
289
290 s0_distn_ac  <- boot_distn_ac$seeds0
```

```
s1_distn_ac  <- boot_distn_ac$seeds1
s2_distn_ac  <- boot_distn_ac$seeds2
s3_distn_ac  <- boot_distn_ac$seeds3
s4_distn_ac  <- boot_distn_ac$seeds4
s5_distn_ac  <- boot_distn_ac$seeds5
s6_distn_ac  <- boot_distn_ac$seeds6
s7_distn_ac  <- boot_distn_ac$seeds7
s29_distn_ac <- boot_distn_ac$seeds29
s30_distn_ac <- boot_distn_ac$seeds30

# Bootstrap means
s0_boot_mu_rl  <- mean(s0_distn_rl)
s1_boot_mu_rl  <- mean(s1_distn_rl)
s2_boot_mu_rl  <- mean(s2_distn_rl)
s3_boot_mu_rl  <- mean(s3_distn_rl)
s4_boot_mu_rl  <- mean(s4_distn_rl)
s5_boot_mu_rl  <- mean(s5_distn_rl)
s6_boot_mu_rl  <- mean(s6_distn_rl)
s7_boot_mu_rl  <- mean(s7_distn_rl)
s29_boot_mu_rl <- mean(s29_distn_rl)
s30_boot_mu_rl <- mean(s30_distn_rl)

s0_boot_mu_ac  <- mean(s0_distn_ac)
s1_boot_mu_ac  <- mean(s1_distn_ac)
s2_boot_mu_ac  <- mean(s2_distn_ac)
s3_boot_mu_ac  <- mean(s3_distn_ac)
s4_boot_mu_ac  <- mean(s4_distn_ac)
s5_boot_mu_ac  <- mean(s5_distn_ac)
s6_boot_mu_ac  <- mean(s6_distn_ac)
s7_boot_mu_ac  <- mean(s7_distn_ac)
s29_boot_mu_ac <- mean(s29_distn_ac)
s30_boot_mu_ac <- mean(s30_distn_ac)

# Bootstrap confidence intervals
s0_boot_ci_rl  <- quantile(s0_distn_rl,probs=c(0.025,0.975))
s1_boot_ci_rl  <- quantile(s1_distn_rl,probs=c(0.025,0.975))
s2_boot_ci_rl  <- quantile(s2_distn_rl,probs=c(0.025,0.975))
s3_boot_ci_rl  <- quantile(s3_distn_rl,probs=c(0.025,0.975))
```

```
s4_boot_ci_rl  <- quantile(s4_distn_rl,probs=c(0.025,0.975))
s5_boot_ci_rl  <- quantile(s5_distn_rl,probs=c(0.025,0.975))
s6_boot_ci_rl  <- quantile(s6_distn_rl,probs=c(0.025,0.975))
s7_boot_ci_rl  <- quantile(s7_distn_rl,probs=c(0.025,0.975))
s29_boot_ci_rl <- quantile(s29_distn_rl,probs=c(0.025,0.975))
s30_boot_ci_rl <- quantile(s30_distn_rl,probs=c(0.025,0.975))

s0_boot_ci_ac  <- quantile(s0_distn_ac,probs=c(0.025,0.975))
s1_boot_ci_ac  <- quantile(s1_distn_ac,probs=c(0.025,0.975))
s2_boot_ci_ac  <- quantile(s2_distn_ac,probs=c(0.025,0.975))
s3_boot_ci_ac  <- quantile(s3_distn_ac,probs=c(0.025,0.975))
s4_boot_ci_ac  <- quantile(s4_distn_ac,probs=c(0.025,0.975))
s5_boot_ci_ac  <- quantile(s5_distn_ac,probs=c(0.025,0.975))
s6_boot_ci_ac  <- quantile(s6_distn_ac,probs=c(0.025,0.975))
s7_boot_ci_ac  <- quantile(s7_distn_ac,probs=c(0.025,0.975))
s29_boot_ci_ac <- quantile(s29_distn_ac,probs=c(0.025,0.975))
s30_boot_ci_ac <- quantile(s30_distn_ac,probs=c(0.025,0.975))


### ------------------------------------------------
## -- Seed table

# Compare second year
s2_boot_mu_rl
s2_boot_ci_rl

s2_boot_mu_ac
s2_boot_ci_ac

# Compare third year
s3_boot_mu_rl
s3_boot_ci_rl

s3_boot_mu_ac
s3_boot_ci_ac

# Compare fourth year
s4_boot_mu_rl
```

```
s4_boot_ci_rl

s4_boot_mu_ac
s4_boot_ci_ac

# Compare fifth year
s5_boot_mu_rl
s5_boot_ci_rl

s5_boot_mu_ac
s5_boot_ci_ac

# Compare sixth year
s6_boot_mu_rl
s6_boot_ci_rl

s6_boot_mu_ac
s6_boot_ci_ac

# Compare seventh year
s7_boot_mu_rl
s7_boot_ci_rl

s7_boot_mu_ac
s7_boot_ci_ac

# Compare twenty-ninth year
s29_boot_mu_rl
s29_boot_ci_rl

s29_boot_mu_ac
s29_boot_ci_ac

# Compare thirtieth year
s30_boot_mu_rl
s30_boot_ci_rl

s30_boot_mu_ac
```

```
s30_boot_ci_ac

### ------------------------------------------------
## -- Check if the differences are significant

s2_boot_ci_rl %overlaps% s2_boot_ci_ac
s3_boot_ci_rl %overlaps% s3_boot_ci_ac
s4_boot_ci_rl %overlaps% s4_boot_ci_ac
s5_boot_ci_rl %overlaps% s5_boot_ci_ac
s6_boot_ci_rl %overlaps% s6_boot_ci_ac
s7_boot_ci_rl %overlaps% s7_boot_ci_ac

s29_boot_ci_rl %overlaps% s29_boot_ci_ac
s30_boot_ci_rl %overlaps% s30_boot_ci_ac


### ------------------------------------------------
## -- Height information

# Get heights and biomass
has29_distn_rl <- boot_distn_rl$height_as29
has29_distn_ac <- boot_distn_ac$height_as29
hfs29_distn_rl <- boot_distn_rl$height_fs29
hfs29_distn_ac <- boot_distn_ac$height_fs29

has30_distn_rl <- boot_distn_rl$height_as30
has30_distn_ac <- boot_distn_ac$height_as30
hfs30_distn_rl <- boot_distn_rl$height_fs30
hfs30_distn_ac <- boot_distn_ac$height_fs30

hao_distn_rl <- boot_distn_rl$height_ao
hao_distn_ac <- boot_distn_ac$height_ao
hfo_distn_rl <- boot_distn_rl$height_fo
hfo_distn_ac <- boot_distn_ac$height_fo

b_distn_rl <- boot_distn_rl$biomass
b_distn_ac <- boot_distn_ac$biomass

```

```
443
444 # Bootstrap means
445 has29_boot_mu_rl <- mean(has29_distn_rl)
446 has29_boot_mu_ac <- mean(has29_distn_ac)
447 hfs29_boot_mu_rl <- mean(hfs29_distn_rl)
448 hfs29_boot_mu_ac <- mean(hfs29_distn_ac)
449
450 has30_boot_mu_rl <- mean(has30_distn_rl)
451 has30_boot_mu_ac <- mean(has30_distn_ac)
452 hfs30_boot_mu_rl <- mean(hfs30_distn_rl)
453 hfs30_boot_mu_ac <- mean(hfs30_distn_ac)
454
455 hao_boot_mu_rl <- mean(hao_distn_rl)
456 hao_boot_mu_ac <- mean(hao_distn_ac)
457 hfo_boot_mu_rl <- mean(hfo_distn_rl)
458 hfo_boot_mu_ac <- mean(hfo_distn_ac)
459
460 b_boot_mu_rl <- mean(b_distn_rl)
461 b_boot_mu_ac <- mean(b_distn_ac)
462
463 # Bootstrap confidence intervals
464 has29_boot_ci_rl <- quantile(has29_distn_rl,probs=c(0.025,0.975))
465 has29_boot_ci_ac <- quantile(has29_distn_ac,probs=c(0.025,0.975))
466 hfs29_boot_ci_rl <- quantile(hfs29_distn_rl,probs=c(0.025,0.975))
467 hfs29_boot_ci_ac <- quantile(hfs29_distn_ac,probs=c(0.025,0.975))
468
469 has30_boot_ci_rl <- quantile(has30_distn_rl,probs=c(0.025,0.975))
470 has30_boot_ci_ac <- quantile(has30_distn_ac,probs=c(0.025,0.975))
471 hfs30_boot_ci_rl <- quantile(hfs30_distn_rl,probs=c(0.025,0.975))
472 hfs30_boot_ci_ac <- quantile(hfs30_distn_ac,probs=c(0.025,0.975))
473
474 hao_boot_ci_rl <- quantile(hao_distn_rl,probs=c(0.025,0.975))
475 hao_boot_ci_ac <- quantile(hao_distn_ac,probs=c(0.025,0.975))
476 hfo_boot_ci_rl <- quantile(hfo_distn_rl,probs=c(0.025,0.975))
477 hfo_boot_ci_ac <- quantile(hfo_distn_ac,probs=c(0.025,0.975))
478
479 b_boot_ci_rl <- quantile(b_distn_rl,probs=c(0.025,0.975))
480 b_boot_ci_ac <- quantile(b_distn_ac,probs=c(0.025,0.975))
```

```
481
482
483 ### ------------------------------------------------
484 ## --- Height table
485
486 # Heights of vegetative plants
487 # NOTE: No need for hao_boot_mu_ac or has29_boot_mu_rl (or their
      CIs)...
488 #   ... since they're the same as hao_boot_mu_rl and has30_boot_mu
      _rl
489
490 hao_boot_mu_rl
491 hao_boot_ci_rl
492
493 has30_boot_mu_rl
494 has30_boot_ci_rl
495
496 has29_boot_mu_ac
497 has29_boot_ci_ac
498
499 has30_boot_mu_ac
500 has30_boot_ci_ac
501
502 # Heights of flowering plants
503
504 hfo_boot_mu_rl
505 hfo_boot_ci_rl
506
507 hfs30_boot_mu_rl
508 hfs30_boot_ci_rl
509
510 hfs29_boot_mu_ac
511 hfs29_boot_ci_ac
512
513 hfs30_boot_mu_ac
514 hfs30_boot_ci_ac
```

B.4 Determine Equilibrium and Stablility

B.4.1 Determine equilibrium

```
library(grDevices)


### ------------------------------------------------
## --- Find dominant eigenvalue of P

L <- 5; U <- 550; m <- 1000
h <- (U-L)/m; meshpts <- L + (1:m)*h - h/2

P <- P_k(h,meshpts,1000000,"ac")
Re(eigen(P,only.values=T)$values[1])


### ------------------------------------------------
## --- Code discretization

# Params
L <- 5; U <- 550; m <- U-L
h <- (U-L)/m; meshpts <- L + (1:m)*h - h/2
weights <- (meshpts/100)^mass_exp
recruit_dist <- recruit_size(meshpts)

# Which model to use
modl <- "rl"

### ------------------------------------------------
## --- Various helper functions

# Helper function for A(b)
A <- function(b,modl) {
   P_mat <- P_k(h,meshpts,b,modl)
   R_mat <- R_k(h,meshpts,b,modl)
   Z_mat <- matrix(0,nrow=m,ncol=m)
```

```
34    tmp <- cbind(rbind(P_mat,R_mat),rbind(Z_mat,Z_mat))
35    return(tmp)
36  }
37
38  # Helper function for n_s^*
39  find_ns <- function(pe) {
40    ns_star <- 0
41    if (pe<alpha) {
42      ns_star <- (alpha-pe*beta)/pe}
43      #ns_star <- pe*beta/(alpha-pe)}
44    return(ns_star)
45  }
46
47  ### ---------------------------------------------
48  ## -- Find equilibium for recruit-limited model
49
50  # Stability radius
51  q <- t(rbind(matrix(0,nrow=m,ncol=1),matrix(h*mean_seeds(1),nrow=
      m,ncol=1)))
52  c0 <- rbind(as.matrix(recruit_size(meshpts)),matrix(0,nrow=m,ncol
      =1))
53  tmp <- A(0,"rl")
54  fund_mat <- solve(diag(2*m)-tmp)
55  pe_rl <- 1/(q%*%fund_mat%*%c0)
56
57  # Equilibrium seeds
58  ns_eq_rl <- find_ns(pe_rl)
59
60  # Equilibrium
61  n_eq_rl <- drop(pe_rl*ns_eq_rl) * (fund_mat%*%c0)
62  nt_eq_rl <- n_eq_rl[1:m]
63  nr_eq_rl <- n_eq_rl[(m+1):(2*m)]
64
65  # Equilibrium biomass
66  b_eq_rl <- (nt_eq_rl+nr_eq_rl)%*%weights
67  b_eq_rl
68
69
```

```
70 ### ------------------------------------------------
71 ## --- Find equilibrium dens as a function of $b$
72
73 # Initialize
74 N <- 100
75 max_dens <- 4000
76 b_vec <- as.matrix(seq(1,max_dens,length=N))
77 pe_vec <- array(0,dim=N)
78 ns_vec <- array(0,dim=N)
79 n_eq_vec <- array(0,dim=c(N,2*m))
80 bs_new <- array(0,dim=N)
81
82 # Precompute these vectors for speeeeed
83 q <- t(rbind(matrix(0,nrow=m,ncol=1),matrix(h*mean_seeds(1),nrow=
       m,ncol=1)))
84 c0 <- rbind(as.matrix(recruit_size(meshpts)),matrix(0,nrow=m,ncol
       =1))
85
86 # p^*_e = 1/(q^T (I-A(b))^{-1} c_0)
87 ptm <- proc.time()
88 for (i in 1:N) {
89   b <- b_vec[i]
90   tmp <- A(b,"ac")
91   fund_mat <- solve(diag(2*m)-tmp)
92   pe_vec[i] <- 1/(q%*%fund_mat%*%c0)
93   ns_vec[i] <- find_ns(pe_vec[i])
94   n_eq_vec[i,] <- fund_mat%*%c0 *(pe_vec[i]*ns_vec[i])
95   bs_new[i] <- update_density(h,weights,n_eq_vec[i,1:m],n_eq_vec[
       i,(m+1):(2*m)],1,1)
96   print(i)
97 }
98 print(proc.time()-ptm)
99
100
101 ### ------------------------------------------------
102 ## --- Find the biomass which gives the equilibrium size
       distribution
103
```

```
104 # Params
105 L <- 5; U <- 550; m <- U-L
106 h <- (U-L)/m; meshpts <- L + (1:m)*h - h/2
107 weights <- (meshpts/100)^mass_exp
108 recruit_dist <- recruit_size(meshpts)
109
110 # Initialize
111 b_l <- 2142
112 b_u <- 2146
113 b_m <- (b_l+b_u)/2
114 b_approx <- 2500
115 err <- b_approx-b_m
116 tol <- 0.00001
117
118 # Precompute these vectors for speeeeed
119 q <- t(rbind(matrix(0,nrow=m,ncol=1),matrix(h*mean_seeds(1),nrow=
    m,ncol=1)))
120 c0 <- rbind(as.matrix(recruit_size(meshpts)),matrix(0,nrow=m,ncol
    =1))
121
122 # Run the loop
123 while (abs(err) > tol) {
124   if (err>0) {b_l <- b_m}
125   if (err<0) {b_u <- b_m}
126   b_m <- (b_l+b_u)/2
127   fund_mat <- solve(diag(2*m)-A(b_m,"ac"))
128   pe <- 1/(q%*%fund_mat%*%c0)
129   ns <- find_ns(pe)
130   bar_n <- fund_mat%*%c0 * drop(pe*ns)
131   b_approx <- update_density(h,weights,bar_n[1:m],bar_n[(m+1):(2*
      m)],1,1)
132   err <- b_approx-b_m
133   print(err)
134 }
135 b_approx
136
137 # Give better names to the final values
138 pe_ac    <- pe
```

```
139 ns_eq_ac <- ns
140 n_eq_ac  <- bar_n
141 nt_eq_ac <- n_eq_ac[1:m]
142 nr_eq_ac <- n_eq_ac[(m+1):(2*m)]
143 b_eq_ac  <- b_approx
144
145
146 ### ---------------------------------------------------
147 ## -- Save equilibria for later use in stability analysis
148
149 write.csv(n_eq_rl, file="equilibrium_rl.csv")
150 write.csv(n_eq_ac, file="equilibrium_ac.csv")
151
152
153 ### ---------------------------------------------------------------------------
154 ### ---- COMPARE RESULTS VS REALITY
155 ### ---------------------------------------------------------------------------
156
157 ### ---------------------------------------------------
158 ## -- Compare biomasses
159
160 # Equilibrium biomasses
161 b_eq_rl
162 b_eq_ac
163
164 # vs max observed biomass
165 max(hmdata$dens,hmdata$dens1)
166
167
168 ### ---------------------------------------------------
169 ## -- Compare seed production and number of flowering plants
170
171 # Equilibrium seeds
172 ns_eq_rl
```

```
173 ns_eq_ac
174
175 # Equilibrium flowering plants
176 ns_eq_rl/(mean_seeds(1))
177 ns_eq_ac/(mean_seeds(1))
178
179 # Equilibrium flowering plants (alt...should equal above)
180 sum(nr_eq_rl)*h
181 sum(nr_eq_ac)*h
182
183 # Observed flowering plants
184 total_flowering <- sum(hmdata$reproduced,na.rm=T)
185 total_plots <- length(levels(as.factor(hmdata$plot)))
186
187 total_flowering/total_plots
188
189
190 ### ------------------------------------------------
191 ## -- Compare number of vegetative plants
192
193 # Equilibrium vegetative plants
194 sum(nt_eq_rl)*h
195 sum(nt_eq_ac)*h
196
197 # Observed vegetative plants (including seedlings)
198 hmdata_veg <- rbind(seedlingdata,hmdata_adult)
199 total_vegetative <- nrow(hmdata_veg)
200 total_vegetative/total_plots
201
202
203 ### ------------------------------------------------
204 ## -- Compare mean heights
205
206 # Mean heights from recruit-limited model
207 sum(nt_eq_rl*meshpts)/sum(nt_eq_rl)
208 sum(nr_eq_rl*meshpts)/sum(nr_eq_rl)
209
210 # Mean heights from adult competition model
```

```
sum(nt_eq_ac*meshpts)/sum(nt_eq_ac)
sum(nr_eq_ac*meshpts)/sum(nr_eq_ac)

# Observed mean heights in 2003
mean(hmdata_veg$h1)
mean(hmdata_flow$h1)
```

B.4.2 Determine stability of equilibrium

```
# Helper functions for computing the Jacobian
dP_fun <- function(z1,z,dens) {
  gamma <- as.numeric(coefficients(repr_fit_ac)[3])
  return(-gamma*p_s(z)*p_r(z,dens,"ac")*(1-p_r(z,dens,"ac"))*g_a(
    z1,z))}
dR_fun <- function(z1,z,dens) {
  gamma <- as.numeric(coefficients(repr_fit_ac)[3])
  return(gamma*p_s(z)*p_r(z,dens,"ac")*(1-p_r(z,dens,"ac"))*g_r(
    z1,z))}
dF_fun<- function(z1,z,ns) {
  return(-alpha*mean_seeds(1)*recruit_size(z1)/((beta+ns)^2))}

# Compute K
P_mat <- P_k(h,meshpts,b_eq_ac,"ac")
R_mat <- R_k(h,meshpts,b_eq_ac,"ac")
F_mat <- F_k_mod(h,recruit_dist,ns_eq_ac)
Z_mat <- matrix(0,nrow=m,ncol=m)
K <- cbind(rbind(P_mat,R_mat),rbind(F_mat,Z_mat))

# Compute Q1
dP_mat <- h*outer(meshpts,meshpts,dP_fun,b_eq_ac)
dR_mat <- h*outer(meshpts,meshpts,dR_fun,b_eq_ac)
dZ_mat <- matrix(0,nrow=m,ncol=m)
dK_mat <- cbind(rbind(dP_mat,dR_mat),rbind(dZ_mat,dZ_mat))
Q1 <- dK_mat %*% n_eq_ac

# Compute W1
```

```
W1 <- as.matrix((meshpts/100)^mass_exp)
W1 <- rbind(W1,W1)

# Compute Q2
dZ_mat <- matrix(0,nrow=m, ncol=m)
dF_mat <- h*outer(meshpts, meshpts, dF_fun, ns_eq_ac)
dK_mat <- cbind(rbind(dZ_mat, dZ_mat), rbind(dF_mat, dZ_mat))
Q2 <- dK_mat %*% n_eq_ac

# Compute W2
Z_vec <- matrix(0,nrow=m, ncol=1)
one_vec <- matrix(1,nrow=m, ncol=1)
W2 <- rbind(Z_vec, one_vec)*mean_seeds(1)

# Compute J
J <- K + outer(t(Q1),W1,FUN="*")[,,,] + outer(t(Q2),W2,FUN="*")
    [,,,]

# Determine stability
Re(eigen(J, only.values=T)$values[1])
```

B.5 Methods Plots

B.5.1 Plot vital rate functions

```
### ------------------------------------------
## -- Plot vital rates

# Specify data set to use
dat <- hmdata

# Real data to plot (ticks or points as necessary)
dat_surv <- dat[!is.na(dat$survived),]
dat_repr <- dat[!is.na(dat$reproduced),]
dat_veg  <- dat[dat$Stage2003!=4 & dat$Stage2003!=5 & !is.na(dat$
   h),]
```

```
dat_flo  <- dat[dat$Stage2003==4,]

# Make points that will be plotted
xmax <- 220
x_vec <- seq(0,xmax, length=100)
s_vec <- p_s(x_vec)
r_rl_vec <- p_r(x_vec,0,"rl")
r_ac_vec <- p_r(x_vec,0,"ac")

# These values are used to create segments in the growth plot
ga_min_x <- min(dat_veg$h)
ga_min_y <- as.numeric(predict(grow_fit_a,data.frame(h=ga_min_x)))
ga_max_x <- max(dat_veg$h)
ga_max_y <- as.numeric(predict(grow_fit_a,data.frame(h=ga_max_x)))

gr_min_x <- min(dat_flo$h)
gr_min_y <- as.numeric(predict(grow_fit_r,data.frame(h=gr_min_x)))
gr_max_x <- max(dat_flo$h)
gr_max_y <- as.numeric(predict(grow_fit_r,data.frame(h=gr_max_x)))

# PLOTS
par(mfrow=c(2,2))
dx <- 0.03
dy <- 1
line_frac <- 1
scale_text <- 1.3

plot(x_vec,s_vec, type="l",lwd=2, xlim=c(0,xmax),ylim=c(0,1),
     xlab="Height of plant (cm)",ylab="Probability",cex.lab=scale
         _text,cex.axis=scale_text)
points(dat_surv$h,dat_surv$survived,pch="|",cex=line_frac)
mtext("(a)",side=3,adj=dx,padj=dy,line=-1.3)

plot(dat_veg$h,dat_veg$h1, pch=16, xlim=c(0,xmax),ylim=c(0,400),
```

```
44        xlab="Height in 2002 (cm)", ylab="Height in 2003 (cm)",cex.
              lab=scale_text,cex.axis=scale_text)
45  points(dat_flo$h,dat_flo$h1,pch=23)
46  segments(ga_min_x,y0=ga_min_y, ga_max_x,y1=ga_max_y, lwd=2)
47  segments(gr_min_x,y0=gr_min_y, gr_max_x,y1=gr_max_y, lwd=2)
48  mtext("(b)",side=3,adj=dx,padj=dy,line=-1.3)
49
50  plot(x_vec,r_rl_vec, type="l",lwd=2, xlim=c(0,xmax),ylim=c(0,1),
51        xlab="Height of plant (cm)",ylab="Probability",cex.lab=scale
              _text,cex.axis=scale_text)
52  points(dat_repr$h,dat_repr$reproduced,pch="|",cex=line_frac)
53  mtext("(c)",side=3,adj=dx,padj=dy,line=-1.3)
54
55  plot(x_vec,r_ac_vec, type="l",lwd=2, xlim=c(0,xmax),ylim=c(0,1),
56        xlab="Height of plant (cm)",ylab="Probability",cex.lab=scale
              _text,cex.axis=scale_text)
57  points(dat_repr$h,dat_repr$reproduced,pch="|",cex=line_frac)
58  mtext("(d)",side=3,adj=dx,padj=dy,line=-1.3)
59
60  par(mfrow=c(1,1))
```

B.5.2 Plot recruit size distribution

```
1  ### ------------------------------------------------
2  ## --- Plot recruit size distn
3
4  # Specify data set to use
5  dat <- seedlingdata
6  dat_recr <- dat$Height2003
7
8  # Vectors to help plot f_e
9  ns_vec <- seq(0,10000,length=1000)
10 fe_vec <- expected_seedlings(ns_vec)
11
12 # Vectors to help plot size distribution
13 xmin <- min(dat_recr)
```

```
xmax <- max(dat_recr)
x_vec <- seq(xmin,xmax, length=100)
y_vec <- recruit_size(x_vec)

# PLOTS
par(mfrow=c(1,2))
distnce <- 0.02
scale_text <- 1

plot(ns_vec,fe_vec, type="l", lwd=2,
     xlab="Number of seeds", ylab="Number of seedlings",
     cex.lab=scale_text,cex.axis=scale_text,main="")
mtext("(a)",side=3,adj=distnce,line=-1.3)

hist(dat_recr,breaks=7, xlim=c(xmin-5,xmax),ylim=c(0,0.021),
     xlab="Recruit height (cm)",ylab="Proportion of recruits",
     cex.lab=scale_text,cex.axis=scale_text,main="",freq=FALSE)
lines(x_vec,y_vec,type="l",lwd=2)
mtext("(b)",side=3,adj=distnce,line=-1.3)

par(mfrow=c(1,1))
```

B.6 Results Plots

B.6.1 Simulate for 30 years

```
### ------------------------------------------------
## --- Run simulations

# Run simulation starting from equilibrium
L <- 5; U <- 550; m <- U-L
num_steps <- 30

# With seeds
ns_init <- 1
nt_init <- rep(0,m)
```

```
nr_init <- rep(0,m)

sim_res_rl <- master_local_simulation_ws(L,U,m, num_steps, ns_init,nt_init,nr_init, "rl")
sim_res_ac <- master_local_simulation_ws(L,U,m, num_steps, ns_init,nt_init,nr_init, "ac")


### ------------------------------------------------
## --- Seed production results

# Number of years in which adult competition model predicts more seeds
year_overtook <- 0
while (sim_res_ac$seeds[year_overtook+1] >= sim_res_rl$seeds[year_overtook+1]) {
  year_overtook <- year_overtook+1
}
year_overtook
years_more <- year_overtook-1

# Comparison of asymptotic behaviour
lower_seed <- min(tail(sim_res_ac$seeds,2))
upper_seed <- max(tail(sim_res_ac$seeds,2))
vs_rl <- 1-upper_seed/max(sim_res_rl$seeds)

lower_seed
upper_seed
vs_rl

# Plot initial seed production and long-term seed production
plot(0:num_steps,sim_res_rl$seeds,col="black",pch=20,
     xlab="Year",ylab="Number of seeds")
points(0:num_steps,sim_res_ac$seeds,col="black",pch=18)
```

B.6.2 Mean height comparison

```
### ------------------------------------------------
## --- Mean height calculations

# Observed mean height of recruits, vegetative plants, and
    flowering plants
mean(seedlingdata$h1)
mean(hmdata_adult$h1)
mean(hmdata_flow$h1)

# Mean of recruits predicted by recruit size dist'n
h <- (U-L)/m; meshpts <- L + (1:m)*h - h/2
meshpts %*% recruit_size(meshpts)

# Mean of adult plants predicted at equilibrium
weighted_distn <- sim_res_rl$adults[num_steps,]/sum(sim_res_rl$
    adults[num_steps,])
meshpts %*% weighted_distn

# Mean of flowering plants predicted at equilibrium
weighted_distn <- sim_res_rl$flowering[num_steps,]/sum(sim_res_rl
    $flowering[num_steps,])
meshpts %*% weighted_distn
```