PVA.R Users Guide

**Brett van Poorten**

**University of British Columbia, Institute for the Oceans and Fisheries**

**Bill Pine**

**University of Florida, Department of Wildlife and Conservation**

Introduction 3

Model Description 3

Parameters and controls 4

Running the model 6

set.pars( ) 7

VPA( controls, parameters ) 7

scenario.switch( scenario = c(“base”, “epis”, “recfail”, “springs”, “hab”, “dam”, “stock”, “remove.brood”), set.ymax=2000, plot.switch=TRUE ) 7

heat.proj( controls, parameters, set.ymax, plot.switch ) 8

Scenarios 8

# Introduction

**PVA.R** is a population viability analysis simulator designed to quickly and accurately reflect uncertainty in the fate of imperiled populations. The model can be used for any species or population. Populations are separated into three life stages: pre-recruits, juveniles/sub-adults and adults. Pre-recruits are subject to density-dependent mortality and can in turn by separated into multiple stanzas, where the strength of density dependence in each stanza is user defined.

This users guide first describes the biological model so user-provided inputs can be understood within the context of the model. User-provided parameters and controls are then defined. Finally, a step-by-step guide to using the functions will be provided with examples to show users how to evaluate new populations with the model.

# Model Description

The model begins by establishing stock-recruitment parameters from calculations of equilibrium population structure when unfished. Length- (T1.2), weight- (T1.3) and fecundity (T1.4) at age are first calculated, followed by size-based survival for recruited age-classes (T1.5). The product of survivorship and fecundity-at-age are summed to calculate equilibrium egg density per recruit (T1.8). Beverton-Holt recruitment function parameters *α* (maximum survival at low population density) and *β* (carrying capacity parameter) are calculated based on eggs-per-recruit, Goodyear compensation ratio (*κ*) and unfished recruits (*R*0) (T1.9-T1.10; Walters and Martell, 2004).

Recruitment to each pre-recruit stanza is calculated based on mortality rates () and density effects () for each life history stanza relative to each other (T1.11-T1.12; Walters and Martell 2004; Pine et al. 2013).

Number of recruits in the first simulated year is calculated as the ratio of user-defined vulnerable biomass in the first year and the sum of vulnerable survivorship (T1.13). Numbers of fish in the first year is calculated as the product of survivorship and first year recruitment (T1.14). Each individual fish-*i* are distributed among age-classes using a multinomially distributed random number where probability of a fish being in any age-class is given by the product of age-specific survivorship (T1.5) and a lognormally distributed recruitment deviate where is a normally distributed random variable with a mean of zero and standard deviation defined by the user. The number of fish that survive from one age and year to the next is a random binomial sample with trials and probability of success given by the annual survival at age (T1.5). Number of eggs produced each year is the sum of fecundity among adult fish (T1.15). The number of eggs surviving through each pre-recruit stanza is a random binomial sample with a probability of success given by the density-dependent survival rate given for each stanza.

# Parameters and controls

The biological population viability function in the R script (the PVA() function) runs given inputs of controls and parameters. Parameters define the biology of the population being modeled and are described in Table 2. Parameter values listed in Table 2 are those used for the Gulf sturgeon case study shown below, but another population or species could be quickly substituted by simply changing the model parameter values in the code (found in the set.pars() function).

Controls are used to define important indices (e.g. *T*, *A*, *ns*, etc.) and provide important changes to the model associated with scenarios evaluating population impacts or habitat changes. Some controls can be logical tests used to indicate if a certain condition is met (e.g. *opt*, *ep.spr*, *recfail*), frequencies of occurrence (ep.fr), proportional impacts of interventions (e.g. *h.st*, *ep.J.M*, *ep.A.M*, *ep.fec*, *habitat*), or absolute additions of stocked fish (*TP.st*).

To simulate implications of stocking, pre-recruit fish may be removed from one of the stanzas or absolute numbers may be added to one of the stanzas. To simulate removal of eggs or juveniles from a donor population for hatchery rearing, the in eq. T1.11 is multiplied by , which reduces the number of individuals remaining in that stanza. The *h.st* control is a vector (length = *ns*) of constant proportions where each element is applied to the respective stanza in all years. For example, *h.st* = {0.2,0} would remove 20% of eggs and no stanza-2 juveniles from the population. Likewise, to simulate additions of hatchery eggs or juveniles to a supplemented population, *TP.st* fish are added to the relevant stanza. The *TP.st* control is a matrix (dimensions = {*T*,*ns*}) added to respective stanzas in specified years. This allows stocking to occur on specific years or at specific frequencies (*i.e.* every other year).

Episodic events may include factors that affect recruitment, subadult mortality, adult mortality, or egg deposition in a year (*e.g.* through changes in skip-spawning frequency). The frequency with which these events occur is controlled by *ep.fr* (the number of years between episodic events; annual occurrences would be *ep.fr* = 1). The impact of these episodic events is controlled by *ep.fec* (proportional change in annual fecundity) *ep.J.M* (additional subadult annual mortality), *ep.A.M* (additional adult annual mortality). If the flag *ep.spr* is TRUE, egg deposition in affected years is automatically reduced by 80%. If the flag *recfail* is TRUE, there is complete recruitment failure in affected years. The flag *opt* may be used to simulate episodic events occurring for two years consecutively (*opt* = TRUE) or only one (*opt* = FALSE).

Habitat availability may be reduced (*e.g.* through dams limiting access to rearing sites) or improved (*e.g.* through habitat restoration). The impact of habitat changes is only impacts life stages where density-dependence occurs (e.g. eggs and pre-recruits). To simulate changes to the amount of available habitat used by any pre-recruit stanza (relative to the first year), is divided by the quantity of habitat available to each stanza relative to the first year. The *habitat* control is a matrix (dimensions = {*T*,*ns*}) applied to respective stanzas in specified years. In this way, successive changes to habitat over time (improvements or reductions) may be simulated to evaluate how these changes affect recovery projections.

# Running the model

There are five functions included in **PVA.R**, including: set.pars(); PVA(); scenario.switch(); heat.proj(); and vwReg2(). Each function controls a different aspect of calculating population projections and displaying results.

To begin running the **PVA.R** script, open an R console (or an R user interface like RStudio) and source the program **PVA.R**. This loads the script into memory so R can be run quickly.

A simple model run can be completed and visually evaluated by entering heat.proj( controls=controls, parameters=parameters ). This calls the heat.proj() function using baseline controls and parameters provided in set.pars(). The heat.proj() function (details below) calls the PVA() function, which calculates the population trajectories across each of n.sim simulations. The probability of population extirpation is reported and results are sent to vwReg2(), which plots the distribution of population projections over *T* years.

Scenarios affecting the population may be evaluated using scenario.switch() (details below). This function asks for user input to define the initial population and details of the scenario being evaluated (e.g. mortality rate at different ages, habitat changes, frequency of episodic events, numbers of pre-recruit fish removed or added, etc.). Using input provided by the user, the heat.proj() is used to calculate population trajectories and then plotted.

Each function provided in the script is described below. Details of the various pre-loaded scenarios are provided in the Scenarios section.

### set.pars( )

The set.pars() function builds the list of population parameters and baseline controls for future projections. The build a population viability analysis for a new population or a new situation (*e.g.* change the number of years, or number of situations), values in this function must be altered. Calling this function resets all parameters and controls back to base conditions

### VPA( controls, parameters )

The function VPA() calculates the fate of each simulation according to the parameters and controls provided (parameters and controls are inputs to the program and functions described in Table 1). The function returns calculated multi-stanza Beverton-Holt parameters, numbers at age and time for each simulation as well as the probability of extirpation after 50, 100 and 200 years (if *T* is at least 50, 100 or 200 years long).

### scenario.switch( scenario = c(“base”, “epis”, “recfail”, “springs”, “hab”, “dam”, “stock”, “remove.brood”), set.ymax = 2000, plot.switch = TRUE )

The function scenario.switch() allows users to evaluate the impact of one of several pre-programmed scenarios on the species being examined. Scenarios include “base”, ”epis”, “recfail”, ”springs”, “hab”, “dam”, “stock”, and “remove”. Additional optional input includes set.ymax, and plot.switch. Input set.ymax can be set to TRUE, which automatically calculates a maximum extent of the y-axis of the resulting population trajectory, or a value can be provided by the user. Input plot.switch, which is a logical value indicating whether results should be plotted (TRUE) or returned as a list of the outputs from the PVA() function. The scenario being evaluated must be entered as text as an input to the function. The function then asks the user to enter required information regarding adult mortality rate, initial vulnerable abundance and details regarding the scenario being evaluated. Details regarding each scenario are provided in the Scenario section, below. Once all inputs are collected, the scenario.switch() calls heat.proj(), calculates population projections

### heat.proj( controls, parameters, set.ymax = 2000, plot.switch = TRUE )

The heat.proj() function calls the PVA() function to calculate population trajectories based on controls and parameters provided. It reports probabilities of extirpation after 50, 100 and 200 years (if *T* exceeds those lengths) and either returns outputs from PVA as a list (if plot.switch = FALSE) or plots trajectories by calling vwReg2(). Input set.ymax can be set to TRUE, which automatically calculates a maximum extent of the y-axis of the resulting population trajectory, or a value can be provided by the user.

## Scenarios

We describe several scenarios that have been included in the **PVA.R** code for rapid and flexible evaluation of potential population recovery possibilities. These scenarios are designed to be flexible, with the exact impact of each defined by the user. Scenarios include “base”, “epis”, “recfail”, “springs”, “hab”, “dam”, “stock”, “remove.brood”, and “supplementation”. All scenarios allow the adult mortality and initial population density to be entered by the user. Each scenario is described below and an example of each is provided, with plotted results (based on Gulf sturgeon parameters.

### base

The scenario “base” allows evaluation of adult mortality and initial abundance to address uncertainty in these two critical parameters. The user is asked to supply these two values and then runs the model conditional on all other parameters and controls defined in set.pars().

> scenario.switch("base")

Enter adult mortality: .1

Enter initial vulnerable abundance: 100

Calculating population projections ...

Time difference of 7.666284 secs

Probability of extirpation after:

50 years - 0 %

100 years - 0 %

200 years - 0 %

Computing density estimates for each vertical cut ...

|================================================| 100%

Build ggplot figure ...

Results are shown in Figure 1.

### epis

The scenario “epis” evaluates the impacts of episodic events on population persistence. Depending on the nature of the event and the habitat in which it occurs, these events could include high and immediate additional mortality on either sub-adults or adults. Sub-adults are recruited fish younger than the age at maturity, calculated as

.

The user is asked to supply adult mortality and initial abundance, then the frequency of an episodic event occurring. Finally, the user supplies the additional mortality on juvenile and adult fish. Episodic mortality events occur in random years across simulations at the mean frequency determined by the user.

> scenario.switch("epis")

Enter adult mortality: .1

Enter initial vulnerable abundance: 100

How many years between episodic mortality events: 10

Enter additional juvenile mortality during episodic events: .8

Enter additional adult mortality during episodic events: 0

Calculating population projections ...

Time difference of 7.577598 secs

Probability of extirpation after:

50 years - 0 %

100 years - 0 %

200 years - 0 %

Computing density estimates for each vertical cut ...

|================================================| 100%

Build ggplot figure ...

Results are in Figure 2.

### recfail

The scenario “recfail” evaluates the impact of occasional episodic recruitment failures on population persistence. The user is asked to supply adult mortality and initial abundance, then the mean frequency of recruitment failure. Episodic recruitment failure events occur in random years across simulations at the mean specified by the user.

> scenario.switch("recfail")

Enter adult mortality: .1

Enter initial vulnerable abundance: 100

How many years between episodic recruitment failure events: 10

Calculating population projections ...

Time difference of 7.784209 secs

Probability of extirpation after:

50 years - 0 %

100 years - 0 %

200 years - 0 %

Computing density estimates for each vertical cut ...

|================================================| 100%

Build ggplot figure ...

Results are in Figure 3.

### springs

The scenario “springs” simulates an episodic event affecting consumption or metabolism of adults, which in turn may elevate adult mortality for that year as well as impact likelihood of spawning, and therefore aggregate fecundity for the population. In Gulf sturgeon, this may occur as coldspring flows decline or cease in some years, which effects metabolism and energy reserves of adults resting during the summer. Other species may have similar environmental bottlenecks. The user is asked to supply adult mortality and initial abundance, then the frequency of an episodic event occurring. The user is then asked whether episodic events occur for a single year or two consecutive years, as well as the change to adult mortality and population fecundity.

> scenario.switch("springs")

Enter adult mortality: .1

Enter initial vulnerable abundance: 100

How many years between episodic coldspring flow reduction events: 10

Do springs dry for two years (TRUE) or just one (FALSE):FALSE

What is the additional adult mortaltiy imposed during

coldspring flow reduction events: .1

What is the percent change in egg deposition

during coldspring flow reduction events (no change=100): .8

Calculating population projections ...

Time difference of 7.697321 secs

Probability of extirpation after:

50 years - 0 %

100 years - 0 %

200 years - 0 %

Computing density estimates for each vertical cut ...

|================================================| 100%

Build ggplot figure ...

Results are in Figure 4.

### hab

The scenario “hab” simulates the effects of changes to habitat quality or quantity. Changes to habitat should not affect subadults or adults should not affect the population because there is no density dependent regulation at those stages: only pre-recruits experience density dependent survival. Assuming pre-recruits experience some level of ontogenetic habitat shift (*e.g.* larvae settle in one habitat, juveniles move to another habitat to rear), habitat changes will affect one or more pre-recruit stanzas. The user is asked to supply adult mortality and initial abundance, followed by an indication of the relative change (positive or negative) in habitat capacity as a percent. Finally, the user is asked which pre-recruit stanza uses the changed habitat.

> scenario.switch("hab")

Enter adult mortality: .1

Enter initial vulnerable abundance: 100

What is the percent change in habitat capacity (no change=100): 120

Which stanza is impacted by this habitat change (between 1-nS): 2

Calculating population projections ...

Time difference of 7.881265 secs

Probability of extirpation after:

50 years - 0 %

100 years - 0 %

200 years - 0 %

Computing density estimates for each vertical cut ...

|================================================| 100%

Build ggplot figure ...

Results are in Figure 5.

### dam

The scenario “dam” simulates the effects of changes in available habitat (to all pre-recruit stanzas) after installation or removal of a dam in a particular year. The user is asked to supply adult mortality and initial abundance, followed by the year the dam is installed/removed and the relative change in available habitat.

> scenario.switch("dam")

Enter adult mortality: .1

Enter initial vulnerable abundance: 100

What year is the dam installed/removed: 20

What is the percent change in available habitat

with the addition/removal of a dam (no change=100): 40

Calculating population projections ...

Time difference of 7.485055 secs

Probability of extirpation after:

50 years - 0 %

100 years - 0 %

200 years - 0 %

Computing density estimates for each vertical cut ...

|================================================| 100%

Build ggplot figure ...

Results are in Figure 6.

### stock

The scenario “stock” simulates the impact of stocking fish at some pre-recruit stage (including eggs). Adding juveniles to the pre-recruit stage will increase numbers of fish, but also result in additional density-dependence and reduced survival (see e.g. van Poorten et al., 2011 for a full discussion). The user is asked to supply adult mortality and initial abundance, followed by the year stocking begins. The user is then asked which stanza fish are stocked into (stanza-1 is eggs) and the number of fish stocked each year. Note: This scenario evaluates demographic impacts of stocking only. The effects of stocking on population genetic structure are not evaluated here and should be carefully considered before hatchery supplementation.

> scenario.switch("stock")

Enter adult mortality: .1

Enter initial vulnerable abundance: 100

Which year does stocking start: 10

Which stanza are fish stocked into: 2

How many fish are stocked annually: 100

Calculating population projections ...

Time difference of 8.356164 secs

Probability of extirpation after:

50 years - 0 %

100 years - 0 %

200 years - 0 %

Computing density estimates for each vertical cut ...

|================================================| 100%

Build ggplot figure ...

Results are in Figure 7.

### remove.brood

The scenario “remove.brood” simulates the impact of removing brood stock for the purposes of supporting hatchery introductions on a different system. Brood stock removal is assumed to be a fixed proportion of available eggs/juveniles rather than a constant number (which prevents against depensatory impacts on available juveniles at low densities). The user is asked to supply adult mortality and initial abundance and the proportion of each stanza removed to support hatchery use.

> scenario.switch("remove.brood")

Enter adult mortality: .1

Enter initial vulnerable abundance: 100

What proportion of eggs/larvae are removed from stanza 1: .6

What proportion of eggs/larvae are removed from stanza 2: 0

Calculating population projections ...

Time difference of 7.789216 secs

Probability of extirpation after:

50 years - 0 %

100 years - 0 %

200 years - 0 %

Computing density estimates for each vertical cut ...

|================================================| 100%

Build ggplot figure ...

Results are in Figure 8.

### supplementation

The scenario “supplementation” combines the previous two scenarios by simulating the impact of removing brood from one or more pre-recruit stanzas and stocking at a particular stanza. This would occur if brood stock were removed from the wild population, raised in the hatchery and stocked at a later stage to improve juvenile survival in the hatchery over what would be experienced in the wild. The user is asked to supply adult mortality and initial abundance and the proportion of each stanza removed to support hatchery use. The user is then asked which year stocking begins, which stanza fish are stocked into and the number of fish stocked each year.

> scenario.switch("supplementation")

Enter adult mortality: .1

Enter initial vulnerable abundance: 100

What proportion of eggs/larvae are removed from stanza 1: .6

What proportion of eggs/larvae are removed from stanza 2: 0

Which year does stocking start: 10

Which stanza are fish stocked into: 2

How many fish are stocked annually: 100

Calculating population projections ...

Time difference of 7.485807 secs

Probability of extirpation after:

50 years - 0 %

100 years - 0 %

200 years - 0 %

Computing density estimates for each vertical cut ...

|================================================| 100%

Build ggplot figure ...

Results are in Figure 9.

Table 1:

|  |  |  |
| --- | --- | --- |
| Parameters | | |
|  | T1.1 |  |
|  | | |
| Initialization | | |
|  | T1.2 |  |
|  | T1.3 |  |
|  | T1.4 |  |
|  | T1.5 |  |
|  | T1.6 |  |
|  | T1.7 |  |
|  | T1.8 |  |
|  | T1.9 |  |
|  | T1.10 |  |
|  | T1.11 |  |
|  | T1.12 |  |
|  | T1.13 |  |
|  | T1.14 |  |
|  | T1.15 |  |

Table 2: Indices, parameters and derived variables used in PVA. Parameter values shown are biological parameters used for the Gulf sturgeon example.

|  |  |  |  |
| --- | --- | --- | --- |
| Symbol | Value | Description | Source |
| **Indices** | | | |
| *t* | {1, 2, …, *T*} | Annual time-step (*T* = 200) |  |
| *a* | {*AR*, 2, …, *A*} | Age (*A* = 60) |  |
| *i* | {1, 2, …, *Et*} | Individual number |  |
| *s* | {1, 2, …, *ns*} | Recruitment stanza (*ns* = 2) |  |
| **Model parameters** | | | |
|  | 200 | Unfished recruits | Ahrens |
| *κ* | 2.8 | Compensation ratio in recruitment | Ahrens |
| *K* | 0.13 | Metabolic rate parameter of von Bertalanffy function | Ahrens |
|  | 0.1 | Minimum adult natural mortality rate |  |
|  | 0.27 | Length at 50% selectivity | Ahrens |
|  | 0.045 | Selectivity shape parameter | Ahrens |
|  | 0.1 | Weight at maturity relative to asymptotic weight | Ahrens |
|  | 1,000 | Initial number of vulnerable fish |  |
|  | 0.5 | Standard deviation in recruitment |  |
| **Derived variables** | | | |
|  |  | Length |  |
|  |  | Weight |  |
|  |  | Survival |  |
|  |  | Selectivity |  |
|  |  | Survivorship |  |
|  |  | Unfished eggs-per-recruit |  |
|  |  | Maximum survival of Beverton-Holt recruitment function |  |
|  |  | Carrying capacity parameter of Beverton-Holt recruitment function for each stanza-*s* |  |
|  |  | Maximum survival of Beverton-Holt recruitment function |  |
|  |  | Carrying capacity parameter of Beverton-Holt recruitment function for each stanza-*s* |  |
|  |  | Recruitment in first year |  |
|  |  | Population abundance in first year |  |

Table 3: Controls used to build scenarios

|  |  |  |
| --- | --- | --- |
| Symbol | Dimensions | Description |
| *h.st* | vector(*ns*) | Proportion of each pre-recruit stanza removed for stocking (from 0-1) |
| *TP.st* | matrix(*nT*,*ns*) | Number of eggs or larvae added each year through stocking |
| *ep.fr* | numeric | Number of intervening years between episodic events |
| *ep.J.M* | numeric | Proportional decline in subadult mortality (between age at recruitment and age at maturity) |
| *ep.A.M* | numeric | Proportional decline in adult mortality |
| *ep.spr* | logical | TRUE/FALSE indicating whether coldspring flows may fluctuate annually |
| *recfail* | logical | TRUE/FALSE indicating if episodic recruitment failure may occur |
| *ep.fec* | numeric | Proportional decrease in egg deposition |
| *opt* | logical | TRUE/FALSE indicating whether episodic events occur for two (TRUE) or one (FALSE) year in a row |
| *habitat* | matrix(*nT*,*ns*) | Proportional annual change in available habitat for each stanza relative to the first year |

[Figure 1](../fig1.pdf): population projections produced using the “base” scenario.

[Figure 2](../fig2.pdf): Population projections produced using the “epis” scenario.

[Figure 3](../fig3.pdf): Population projections produced using the “recfail” scenario.

[Figure 4](../fig4.pdf): Population projections produced using the “springs” scenario.

[Figure 5](../fig5.pdf): Population projections produced using the “hab” scenario.

[Figure 6](../fig6.pdf): Population projections produced using the “dam” scenario.

[Figure 7](../fig7.pdf): Population projections produced using the “stock” scenario.

[Figure 8](../fig8.pdf): Population projections produced using the “remove.brood” scenario.

[Figure 9](../fig9.pdf): Population projections produced using the “supplementation” scenario.