**Tuesday February 11th: Integral Projection Models: construction**

Today you will construct one IPM, from raw data. You can use the script IPM\_construction\_acai.R

After installing and loading packages, and reading the data, conduct the following 9 steps. You can run the code and try to understand and interpret the output.

**Step 1. Explore data**

* To get an impression of the format and type of data, we first explore the data. The dataset contains 5 variables.
* In this dataset, each row is one individual that was included in one of the two censuses that was performed (1997 and 1998). So, also individuals that were not present at the start (1997), but recruited in 1998 are included in the dataset. These get ‘NA’ values for variables in 1997. Data for the second year are indicated as size\_next. Size is expressed in m stem height of the palms.
* The Text box below contains the types of demographic data can used to construct the vital rate models.

**Text box: Variables used as input in the ipmr package (Not all are required)**

Id: an identifier of the individual

size: size of individuals in census time t

size\_next: size of individuals in census time t +1 !

survival: survival of individuals from census time t to t +1 (contains: 0 for death or 1 for survival)

reproduction: this can consist of multiple variables, each with a unique name. Possible variables: probability of reproduction (0 for a non-reproductive indvidual at time t or 1 for a reproductive individual), number of flowers or fruits produced

treatment, population, environmental condition: information to allow making separate kernels for different conditions, or for variable environments.

* The code contains some examples of histograms and plots to explore the data.
* The variables repro and survival contain information on whether individuals were reproductive (i.e., had reproductive structures) and whether they survived to the next time step.

**Step 2. Create growth models**

* To construct the IPM, we need to establish the relations between vital rates (survival, growth and reproduction) with size (height). This is done by statistical analyses (glm, lm or non-linear functions) and results of these analyses are included in an **object**.
* We first make such an object for height growth, using three possible models
* In growth model 1, size\_next is a linear function of size
* In the two other growth models, we use a non-linear model to relate size\_next to size plus a growth term, which contains non-linear parts. We implement this using the nls function in r, a non-linear curve fitting function
* In growth model 2, we use quadratic growth
* In growth model 3, we use the Hossfeld model, which is often used to describe tree growth
* We then compare these models visually (in graphs).
* We also compare the AIC values of the models: their goodness of fit
* It seems model 3 has the lowest AIC values
* Finally, we calculate the standard deviations of the growth residuals around the regression. These are needed to determine the distribution of growth rates in the IPM.

**Step 3. Create survival models**

* Next, we create vital rates models for **survival.** Again, we build a regression model, but in this case a logistic regression using the glm function, because the data have a binary distribution.
* We make three models: a constant, a linear and a quadratic function
* We plot the data and model, and compare the AIC values.
* It seems the survival model 1 is the most supported

**Step 4. Create reproduction models**

* Next, we create vital rates models for **reproduction.** Again, we use logistic regressions with the glm function, because the data have a binary distribution.
* We make three models: a constant, a linear and a quadratic function
* We plot the data and model, and compare the AIC values.
* It seems the reproduction model 2 is the most supported

**Step 5. Recruitment parameters**

* In addition to information on reproduction, we also need to know specify how many recruits (small seedlings) are produced per reproductive palm per year. This number (5.9) is directly obtained from the permanent sample plots. As we do not have a seedling category in this model, we need to adjust this to account for survival until reaching 0.4 m in palm height. We estimate this to be 5%.
* And the final pieces of information on vital rates are the mean and sd of the sizes of the recruits. New recruits are those individuals with NA for size(t), and non-NA for size(t + 1).

**Step 6. Prepare info for IPM model**

* We’ll now bundle all paramater values and statistical models together. Note that this is a mix of model objects and simple parameters.
* Now we can define the size range and the number of classes (= mesh size) of the large matrix that is built from the fecundity, growth and survival objects. The minimum size is simply the size of the smallest observed height. The maximum size is taken to be 20% larger than the maximum size.
* The mesh size is now taken to be 200, and it depends on both the size range and the growth variation. For species with large size ranges compared to their mean growth, a larger mesh is needed. For species with small variation in growth, also a larger mesh is needed (to realistically describe the growth transitions). The effect of mesh size on model output can be tested; one would like to have a mesh size at which an increasing in mesh does not change model output.
* Just as a quick check of the width of the categories in our IPM, we divide the range by the mesh size and see that for the L, U and mesh paramaters, the width is 16 cm of palm height.
* The final preparation is to establish the starting population. To get this, we first create the list of breakpoints.

**Step 7. Create the IPM model**

* The ipmr package calculates lambda by iterations (i.e., not by matrix calculation), so we need to define the number of iterations. Larger matrices and those for species with small rates of size change relative to the maximum size require a large number of iterations (i.e., 500) to reach the stable size distribution.
* The code to construct the IPM is very long and contains a series of steps to construct the IPM. All steps are linked in one pipeline.
* The IPM is initialized by setting the type of IPM: in our case this is a “simple” IPM, and density independent (“di”) and deterministic (“de”).
* The function define\_kernel combines the statistical models for each of the sub-kernels. The first sub-kernel is for survival and growth, called “P”, with transitions between continuous categories, in which the survival and growth rates per size class are multiplied. This sub-kernel contains the results of two regression models, one for survival (s) and one for growth (mean growth, hence g\_mu). For each of these the predictons are made based on the statistical models.
* The remaining information to be specified within the define\_kernel function deal with the variation in growth rates (g, a normal distribution of growth rates, with mean=g\_my and SD = grow\_sd based on the residuals around the growth model), the name(s) of the size variable used, the name of the data\_list and information on how to prevent eviction (i.e., loosing individuals at the edges of the size range.
* Next, the define\_kernel function is used again, now to add the Fecundity kernel to the IPM. As with the previous step, a formula shows the calculation (multiplying the probability of reproduction by the number of recruits per reproductive individual and by the size distribution of new recruits.
* After having defined the statistical relations, the two parts of the kernels need to be described generally. This is done using the following code:
* The next step is to define the size range of the IPM:
* Before actual construction and running of the IPM, the starting distribution is provided to be able to run the model.
* This final step in the IPM construction pipeline is the actual construction of the IPM. In this step the statistical relations are combined, the kernels are produced and the discretization is done, and the model is run for n\_iter time steps.
* You can now check what’s in the resulting IPM. This object contains a list with IPM settings and model output (projections and lambda). This command gives the type of IPM and the lambda value.

**Step 8. Visualize the IPM model**

* We can plot the IPMs in 2D or 3D.
* For 2D plots, we can use ggplot, but then first store the IPM as a dataframe. We can also zoom in on parts of the matrix
* For 3D plots, we can use standard plotting from R base

**Step 9. Do some checks of IPM**

* We can produce some scatter plots of survival (column sum of P), reproduction and stasis to further explore the IPM matrix.
* Slices of the matrix can also be shown. These are examples the transitions from the category at which palms are 2 and 10 m tall. You can observe that the highest transition probability is to keep almost the same height, but also a small chance to reduce in size. This retrogression (moving to smaller categories) is due to the measurement errors, but also to the assumption of a normal (and thus symmetric) distribution of growth residuals the linear regression. For 32 m tall palms, note that the distribution is not symmetric as we’ve chosen to prevent losing individuals from the edges of the IPM.
* The transitions for palms of 32 m height to much larger sizes are impossible because we set U to 32.4 m. But what happens to these transitions? When we implemented the define\_kernel functions, we specified to correct for eviction. Thus, individuals that would be predicted to grow out of the bounds of the IPM (above U or below L) are spread over the other categories to which palms of 32 m can grow.