Tutorial-1.1-Weighted_predictions_matrix_approach

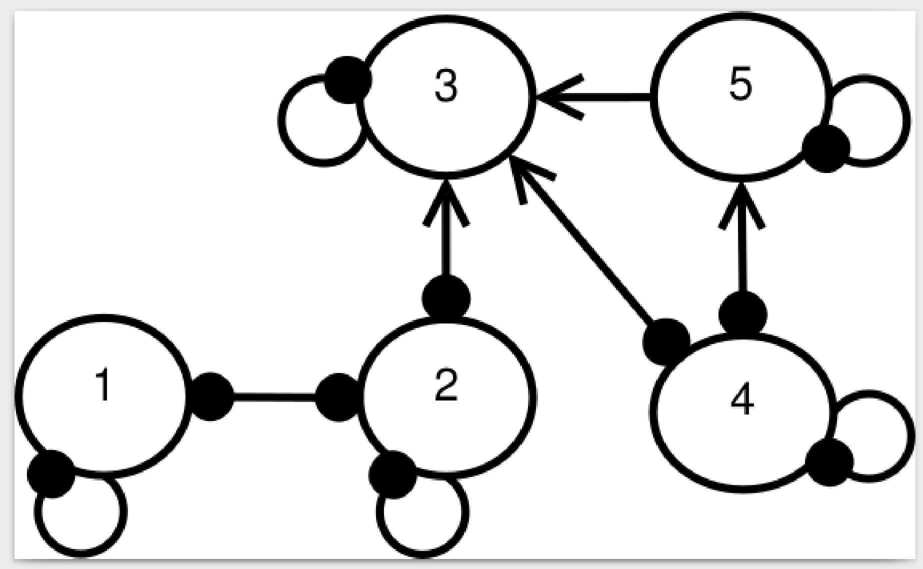
February 1, 2019

1 Weighted-predictions matrix

The weighted-predictions matrix, W, introduced by Dambacher et al. (2002), summarises the positive and negative effects of a press-perturbation on species in an interaction network. In this tutorial, we obtain W for the example from the paper, and use it to predict the effects of a negative press-perturbation of species 3 on species 3, 4, and 5 in the network below.

```
In [20]: from IPython.display import IFrame
         from IPython.display import Image
         import os

         # display the interaction network used in the example
         #IFrame("fivevariable2.pdf", width=500, height=400)
```



fivevariable2.png

1.1 Obtain the weighted-predictions matrix

1.1.1 Step 1: Create the qualitative community matrix

It is convenient to start by encoding the interaction network in a flexible way. Below we have a list of species, spp_list, a dictionary that specifies the positive interactions between species, positive_edges_dict, and a dictionary that specifies the negative interactions between species, negative_edges_dict.

```
In [21]: #
         spp_list = ['s1','s2','s3','s4','s5']

         # key is recipient of a positive effect
         positive_edges_dict = {
         's5': ['s4'],
         's3': ['s2', 's4', 's5'],
         }

         # key is recipient of a negative effect
         negative_edges_dict = {
         's5': ['s5'],
         's4': ['s4', 's5', 's3'],
         's3': ['s3'],
         's2': ['s2', 's1', 's3'],
         's1': ['s1', 's2'],
         }
```

The qualitative community matrix, denoted $^\circ A$ in Dambacher et al. (2002), has a -1 for negative species interactions, +1 for positive interactions, and 0 otherwise.

The first step is to assign each species name to an index in the matrix.

```
In [22]: sz = len(spp_list) # the number of species

         # a dictionary that maps from a species' name to its index in the matrix
         spp2idx = { spp_name: idx for idx, spp_name in enumerate(spp_list) }
         spp2idx

Out[22]: {'s1': 0, 's2': 1, 's3': 2, 's4': 3, 's5': 4}
```

Now we can populate the qualitative community matrix, Aq, with zeros, ones, and negative ones.

```
In [23]: Aq = matrix(QQ, sz, sz)


         for recipient, giverList in positive_edges_dict.items():
             for giver in giverList:
                 Aq[ spp2idx[recipient], spp2idx[giver]] = 1

         for recipient, giverList in negative_edges_dict.items():
             for giver in giverList:
                 Aq[ spp2idx[recipient], spp2idx[giver]] = -1

         Aq
```

```
Out[23]: [-1 -1  0  0  0]
         [-1 -1 -1  0  0]
         [ 0  1 -1  1  1]
         [ 0  0 -1 -1 -1]
         [ 0  0  0  1 -1]
```

1.1.2 Step 2: Find the adjoint of the qualitative community matrix

The classical adjoint or adjugate of the qualitative community matrix, $\text{adj}(-^{\circ}A)$, provides an analogue of the sensitivity matrix

$$-^{\circ}A^{-1} = \frac{\text{adj}(-^{\circ}A)}{\det(-^{\circ}A)}$$

where $\det()$ is the determinant, and $\text{adj}()$ is the classical adjoint or adjugate. It equally weights each feedback loop between species in the system (magnitude $= 1$ for all), and counts the sum of those positive (+1) and negative (-1) feedbacks.

```
In [24]: adj = (-Aq).adjoint()
         adj
```

```
Out[24]: [ 6 -4  2  2  0]
         [-4  4 -2 -2  0]
         [-2  2  0  0  0]
         [ 1 -1  0  1 -1]
         [ 1 -1  0  1  1]
```

Because the elements in $\text{adj}(-^{\circ}A)$ are the sum of positive and negative feedbacks, their signs are indicative of the response of each row-species to a negative press-perturbation of each column-species. For example (taken from Dambacher et al. (2002)), if there are 4 positive feedback loops between two species, then this would result in an element with value $+4$. However, an element with value $+4$ may also result from 44 positive and 40 negative loops, or 6 positive and 2 negative loops, etc. Because the elements are a simple sum, more information is needed to interpret the result probabilistically; one needs to additionally know what the total number of feedback loops is.

1.1.3 Step 3: Find the absolute feedback matrix from the binary community matrix

To obtain the total number of feedback loops, one first creates the binary community matrix. The binary community matrix, $^{\bullet}A$, has a `+1` for any interaction between species, regardless of their sign, and a `0` otherwise.

```
In [25]: # create binary community matrix
         Ab = matrix(QQ, Aq.nrows(), Aq.ncols())
         for position in Aq.nonzero_positions():
             Ab[position] = 1
         Ab
```

```
Out[25]: [1 1 0 0 0]
         [1 1 1 0 0]
```

```
[0 1 1 1 1]
[0 0 1 1 1]
[0 0 0 1 1]
```

The total count of feedback loops is then obtained from the absolute feedback matrix, T. T is from ${}^\bullet A$

$$T_{ji} = \text{per}(\min {}^\bullet A_{ij}),$$

where 'per' is the matrix permanent, and where $\min {}^\bullet A_{ij}$ is the submatrix found by removing row i and column j of ${}^\bullet A$. The matrix permanent is calculated in a similar way to the determinant but with all terms in the sum having a positive sign.

```
In [26]: # absolute feedback matrix
         T = matrix(QQ,sz,sz)
         for row in range(sz):
             for col in range(sz):
                 keeprows = range(sz)
                 keeprows.remove(row)
                 keepcols = range(sz)
                 keepcols.remove(col)
                 T[col,row] = Ab.matrix_from_rows_and_columns(keeprows, keepcols).permanent()

         T
```

```
Out[26]: [6 4 2 2 2]
         [4 4 2 2 2]
         [2 2 4 4 4]
         [1 1 2 3 5]
         [1 1 2 3 5]
```

1.1.4 Step 4: Use the adjoint of the qualitative community matrix and the absolute feedback matrix to find the weighted-predictions matrix

The weighted predictions matrix W is calculated by dividing the absolute value of each element of the adjoint by the corresponding element of T

$$W = \text{abs}(\text{adj}(-{}^\circ A))./T$$

where ./ indicates element-wise division. By convention, W_{ij} is set to 1 when $T_{ij} = 0$.

```
In [27]: # absolute value of adjugate matrix
         abs_adj = adj.apply_map(abs)

         # element-wise fraction of the adjugate to the absolute feedback matrix
         fnc = lambda x: 0 if x == 0 else 1/x
         W = abs_adj.elementwise_product(T.apply_map(fnc))

         # Though for independent species,
         # we set the weighted predictions matrix value to 1
         fnc = lambda x: x == 0
         for position in (T.find(fnc, indices=True)).iterkeys():
             W[position] = 1
         W
```

```
Out[27]: [  1   1   1   1   0]
         [  1   1   1   1   0]
         [  1   1   0   0   0]
         [  1   1   0 1/3 1/5]
         [  1   1   0 1/3 1/5]
```

1.2 Interpret the weighted predictions matrix

From the analysis above, we have

$$\mathrm{adj}(-^{\circ}A) = \begin{bmatrix} +6 & -4 & +2 & +2 & 0 \\ -4 & +4 & -2 & -2 & 0 \\ -2 & +2 & 0 & 0 & 0 \\ +1 & -1 & 0 & +1 & -1 \\ +1 & -1 & 0 & +1 & +1 \end{bmatrix}, T = \begin{bmatrix} 6 & 4 & 2 & 2 & 2 \\ 4 & 4 & 2 & 2 & 2 \\ 2 & 2 & 4 & 4 & 4 \\ 1 & 1 & 2 & 3 & 5 \\ 1 & 1 & 2 & 3 & 5 \end{bmatrix}, W = \begin{bmatrix} 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1/3 & 1/5 \\ 1 & 1 & 0 & 1/3 & 1/5 \end{bmatrix}$$

As explained in Dambacher et al. (2002), the elements in W, ranging between 0 and 1, indicate the determinacy of the response sign prediction obtained from $\mathrm{adj}(-^{\circ}A)$. A value of 1 indicates that all of the feedbacks between a row and column species are of the same sign. This means that the species response to a press perturbation, regardless of the specific interaction-strength values, is guaranteed to match the sign in $\mathrm{adj}(-^{\circ}A)$. Values less than one indicate indeterminacy: there is a combination of positive and negative feedback loops between the species in the system, and so the species response may be positive or negative, depending upon the specific interaction-strength values. A value of 0 indicates either that there are no feedbacks between the row and column species (see convention above), or that the number of positive and negative feedbacks are equal.

Dambacher et al. (2002) further interpreted the magnitude of $W_{ij} < 1$ values as indicative of the degree of indeterminacy in the species response, and based on Monte Carlo simulations, recommended a threshold of $W_{ij} > 0.5$ as a general guideline for high ($\approx$ 95% or more) sign determinacy (Dambacher et al. 2001), which corresponds to $>$ 75% of feedback loops having the same sign. Hosack et al. (2008) also used simulations, where interaction-strength values were sampled from various distributions, to obtain a relationship W_{ij} and sign-determinacy.

In the example in the paper, we are interested in predicting the effects of a negative press-perturbation of species 3 on species 3, 4, and 5. This corresponds to column 3, rows 3, 4, and 5 of the matrices above. In the weighted-predictions matrix, $W_{3,3} = W_{4,3} = W_{5,3} = 0$. Given that $T_{ij} \neq 0$ for the corresponding elements, this means that the response signs are maximally indeterminate: there are an equal number of positive and negative feedbacks between the perturbed species 3 and the other species.

```
In [28]: # the proportion of feedback loops that are positive
         fnc = lambda x: 0 if x == 0 else 1/x
         propn_pos = (1/2)*((adj+T).elementwise_product(T.apply_map(fnc)))
         propn_pos
```

```
Out[28]: [  1   0   1   1 1/2]
         [  0   1   0   0 1/2]
         [  0   1 1/2 1/2 1/2]
         [  1   0 1/2 2/3 2/5]
         [  1   0 1/2 2/3 3/5]
```

If the results of the weighted-predictions matrix analysis were interpreted probabilistically, then giving equal weighting to each feedback, the predictions are equivalent to coin flip between positive or negative response (c.f. Dambacher et al.'s (2002) 'old-field web' results).