

---

```

%=====
% Name      : BiWeb.m
% Author    : Cesar Flores
% Created   : 23/Jan/2012
% Updated   : 23/Jan/2012
% Description : Represent a complete data structure of a bipartite network.
%=====

```

## Main class - Bipartite network

```
classdef Bipartite < handle
```

## Properties

```

properties
    webmatrix          = []; %Interaction matrix (not neccesearly a binary
    adjacency          = []; %Adjacency matrix (binary matrix)
    modules            = {}; %Modularity algorithm object
    nestedness         = {}; %Nestedness algorithm object
    num_edges          = 0; %Number of edges
    connectance        = 0; %Fill of the webmatrix
    n_rows             = 0; %Number of rows
    n_cols             = 0; %Number of columns
    size_webmatrix     = 0; %Size of the matrix (n_rows times n_cols);
    specificity        = []; %----
    row_degrees        = []; %Degrees of each row node.
    col_degrees        = []; %Degrees of each col node.
    rr                 = [];
    ssi                = [];
    bperf              = [];
    name               = {}; %Name of the network
    tests              = {}; %Object of the statistical analysis class (Tes
    nodf_strict        = 1;
    row_labels         = {}; %Labels for row nodes.
    col_labels         = {}; %Labels for col nodes.
    printer            = {}; %Object of the class Print.m for output.
    plotter            = {}; %Instance of the class PlotWebs.m for plotting
    rows_idx           = []; %Id's of the row nodes
    cols_idx           = []; %Id's of the col nodes.
    testmodule         = {}; %Instance of the class TestModules.m for multi
end

```

```
methods
```

```

function obj = Bipartite(web,namebip)

    if(nargin == 0)
        error('You need to specify a double matrix or a txt file in matrix
    end

    if(nargin == 1 && (isa(web,'double')||isa(web,'logical'))); namebip =

```

---

```

if(nargin == 1 && isa(web,'char'))
    [paths namefile ext] = fileparts(web);
    namebip = namefile;
    web = dlmread(web, ' ');
end
web = 1.0*web;
%emptyrows = sum(web,2)==0;
%emptycols = sum(web,1)==0;

%web(emptyrows,:) = [];
%web(:,emptycols) = [];

obj.name = namebip;
obj.webmatrix = web;

%General Info
[obj.n_rows obj.n_cols] = size(web);

if(obj.n_rows == 0 || obj.n_cols == 0)
    return;
end;

obj.size_webmatrix = obj.n_rows * obj.n_cols;

%Connectance
obj.adjacency = obj.webmatrix > 0;
obj.num_edges = sum(obj.adjacency(:));
obj.connectance = sum(obj.adjacency(:))/numel(obj.adjacency);

%Specificity and all
obj.row_degrees = sum(obj.adjacency,2);
obj.col_degrees = sum(obj.adjacency,1)';
obj.specificity = SpeFunc.SPECIFICITY(obj.webmatrix);
obj.rr = SpeFunc.RESOURCE_RANGE(obj.webmatrix);
obj.ssi = SpeFunc.SPECIES_SPECIFICITY_INDEX(obj.webmatrix);
obj.bperf = max(obj.webmatrix)';

obj.nestedness = NODF(obj.adjacency,obj.nodf_strict);
%obj.modules = LPBrim(obj.webmatrix);NewmanAlgorithm
%obj.modules = NewmanModularity(obj.webmatrix);
obj.modules = AdaptiveBrim(obj.webmatrix);
obj.tests = Test(obj);
obj.printer = Printer(obj);
obj.plotter = PlotWebs(obj);
obj.testmodule = TestModules(obj);

for i = 1:obj.n_rows; obj.row_labels{i} = sprintf('row%03i',i); end;
for i = 1:obj.n_cols; obj.col_labels{i} = sprintf('col%03i',i); end;

end

function set.rows_idx(obj,value)

```

---

---

```
        obj.rows_idx = value;
    end

end

Error using Bipartite (line 46)
You need to specify a double matrix or a txt file in matrix format

end
```

*Published with MATLAB® 7.13*