MARSS Reference Sheet

The default MARSS model (`form="marxss"`) is written as follows:

$$\begin{gathered}\mathbf{x}_t = \mathbf{B}_t\mathbf{x}_{t-1} + \mathbf{u}_t + \mathbf{C}_t\mathbf{c}_t + \mathbf{w}_t, \text{ where } \mathbf{w}_t \sim \text{MVN}(0, \mathbf{Q}_t) \\ \mathbf{y}_t = \mathbf{Z}_t\mathbf{x}_t + \mathbf{a}_t + \mathbf{D}_t\mathbf{d}_t + \mathbf{v}_t, \text{ where } \mathbf{v}_t \sim \text{MVN}(0, \mathbf{R}_t) \\ \mathbf{x}_1 \sim \text{MVN}(\boldsymbol{\pi}, \boldsymbol{\Lambda}) \text{ or } \mathbf{x}_0 \sim \text{MVN}(\boldsymbol{\pi}, \boldsymbol{\Lambda})\end{gathered} \tag{1}$$

c and **d** are inputs (numeric) and must have no missing values. The MARSS package is designed to handle linear constraints within the parameter matrices (the **B**, **u**, **C**, **Q**, **Z**, **a**, **D**, **R**, $\boldsymbol{\pi}$, and $\boldsymbol{\Lambda}$)). Linear constraint means you can write the elements of the matrix as a linear equation of all the other elements, although typically each matrix element is just a fixed or estimated value.

The following shows an example of a mean-reverting random walk model with three observation time series written as a MARSS model:

$$\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}_t = \begin{bmatrix} b & 0 \\ 0 & b \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}_{t-1} + \begin{bmatrix} w_1 \\ w_2 \end{bmatrix}_t, \quad \begin{bmatrix} w_1 \\ w_2 \end{bmatrix}_t \sim \text{MVN}\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} q_{11} & q_{12} \\ q_{12} & q_{22} \end{bmatrix} \right), \quad \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}_0 \sim \text{MVN}\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \right)$$

$$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix}_t = \begin{bmatrix} 1 & 1 \\ 0 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}_t + \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix}_t, \quad \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix}_t \sim MVN\left(\begin{bmatrix} a_1 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} r_{11} & 0 & 0 \\ 0 & r & 0 \\ 0 & 0 & r \end{bmatrix} \right)$$

To fit this with MARSS, we translate this model into equivalent matrices (or arrays if time-varying) in R:

```
B1=matrix(list("b",0,0,"b"),2,2)
U1=matrix(0,2,1)
Q1=matrix(c("q11","q12","q12","q22"),2,2)
Z1=matrix(c(1,0,1,1,1,0),3,2)
A1=matrix(list("a1",0,0),3,1)
R1=matrix(list("r11",0,0,0,"r",0,0,0,"r"),3,3)
pi1=matrix(0,2,1); V1=diag(1,2)
model.list=list(B=B1,U=U1,Q=Q1,Z=Z1,A=A1,R=R1,x0=pi1,V0=V1,tinitx=0)
```

Defaults and shortcuts for model specification

B		
default	”identity”	identity matrix
shortcut	”unconstrained”	all elements estimated
shortcut	”diagonal and equal”	diagonal matrix with one value on the diagonal
shortcut	”diagonal and unequal”	diagonal matrix with unique values on the diagonal
numeric matrix	diag(0.8,2)	specify as matrix with numbers
char matrix	matrix(c(”a”,”b”),2,2)	estimated matrix with only 2 estimated parameters
list matrix	matrix(list(”a”,2,1,”b”),2,2)	combine numeric (fixed) and estimated values

U and x0		
default	”unequal”	all u’s different
shortcut	”unconstrained”	same as ”unequal”
shortcut	”zero”	all zero
numeric matrix	matrix(1,2,1)	specify as matrix with numbers
char matrix	matrix(c(”a”,”a”),2,1)	all estimated parameters
list matrix	matrix(list(0,”a”),2,1)	combine numeric (fixed) and estimated values

Q and R		All variance-covariance matrices are symmetric
default	"diagonal and unequal"	diagonal matrix with unique values on the diagonal; independent with different variances
shortcut	"diagonal and equal"	diagonal with one value on diagonal; i.i.d.
shortcut	"unconstrained"	unconstrained variance-covariance matrix
shortcut	"equalvarcov"	one variance and one covariance
shortcut	"identity"	i.i.d with variance of 1
numeric matrix	diag(0.8,2)	specify as matrix with numbers
char matrix	matrix(c("a","c","c","b"),2,2)	specify constrained within a var-cov matrix; careful, matrix must be a valid var-cov structure
list matrix	matrix(list(1,0,0,"b"),2,2)	combine numeric (fixed) and estimated values; careful matrix must be a valid var-cov structure

Z		Z num of columns will specify the num of x; columns of Z must match rows in u, B, Q (otherwise Z specifies a different num of x than u, B, Q)
default	"identity"	identity matrix; means one x will be estimated for each y
numeric matrix	matrix(c(1,0,0,0,1,1),3,2)	specify as matrix with numbers; example is a design matrix with 0s and 1s (assigns each x to a y)
shortcut	"unconstrained"	all elements estimated; rarely used
shortcut	"diagonal and equal"	diagonal matrix with one value on the diagonal
shortcut	"diagonal and unequal"	diagonal matrix with unique values on the diagonal
char matrix	matrix(c("a","b"),2,2)	estimated matrix with only 2 estimated parameters
list matrix	matrix(list("a",2,1,"b"),2,2)	combine numeric (fixed) and estimated values

A		
default	”scalar”	estimates an intercept; sets the first scalar of y assoc with an x to 0
shortcut	”zero”	all zero
numeric matrix	matrix(1,2,1)	specify as matrix with numbers
list matrix	matrix(list(0,”a”),2,1)	combine numeric (fixed) and estimated values; make sure matrix does not lead to a indeterminant model

tinitx		is x0 at t=0 or t=1
default	0	at t=0
	1	at t=1

Inputs - set to zero by default

c		inputs; must be numeric; no missing values allowed; dimension is q x T (number of columns in y)
default	”zero”	if no inputs
if passed in	matrix(1,a,T)	if c passed in, must be a numeric matrix with same num of columns as y

C		if passed in; dimension is m (rows in x) $\times q(rowsinc)$
default	”zero”	if no c passed in, then default is zero
default	”unconstrained”	if c passed in, corresponding C has all elements estimated.
numeric matrix	matrix(1,m,q)	fixed values; specify as matrix with numbers
char matrix	matrix(list(”b”,”a”),m,q)	all estimated values but some shared (equal)
list matrix	matrix(list(0,”a”),m,q)	combine numeric (fixed) and estimated values

d		inputs; must be numeric; no missing values allowed; dimension is p x T (number of columns in y)
default	"zero"	if no inputs
if passed in	matrix(1,p,T)	if c or d passed in, must be a numeric matrix with same num of columns as y

D		if passed in; dimension is n (rows in y) $\times p(rowsind)$
default	"zero"	if no d passed in, then default is zero
default	"unconstrained"	if d passed in, corresponding D has all elements estimated.
numeric matrix	matrix(1,n,p)	fixed values; specify as matrix with numbers
char matrix	matrix(list("b","a"),n,p)	all estimated values but some shared (equal)
list matrix	matrix(list(0,"a"),n,p)	combine numeric (fixed) and estimated values