1 Changes between versions 2.7 and 2.5

added sections on covariates and lag-p model added these sections to the user guide.

MCInit not working for non-diagonal R and Q replaced function for randomly drawing matrices with a random draw from a wishart distribution.

m not getting assigned in MARSSPopWrap was missing some of the allowable cases for Z and m was not being set in those cases.

added more info re R or Q not positive-definite If the user specifies an illegal variance covariance structure from a general estimation perspective (nothing to do with MARSS), they can get the ”not positive-definite” error. Added some text in Troubleshooting section to help if they get this error.

MARSSsimulate bug MARSSsimulate was broken for multivariate simulation since I forgot that rmvnorm returns a 1 x p matrix even if the mean is p x 1. Wrapped the rmvnorm call in a array() to fix the dim setting.

error in x0 update when R=0 and x0 fixed If x_1 has fixed elements, estimates should not used for those elements. Code was missing some `d$x0` bits. This means that the user can fix x_1 when $R = 0$ to a value not equal to the corresponding y_1 value. This would mean an illogical model so a check was added to stop and give warning if that happens.

2 Changes between versions 2.5 and 2.2

factor option for all but Z removed same functionality is provided via list matrices

removed fixed/free args from MARSS() same functionality is provided via list matrices

constraint arg changed to model in MARSS() just the name of the argument was changed to be more intuitive

rewrote user guide to reflect above changes

added case studies to user guide dynamic factor analysis and species interactions with covariates

3 Changes between versions 2.2 and 2.0

allow x10 and x00 This was broken during MARSS 2.0 changes

added diffuse priors For method="BFGS" and kf.x0="x10"

incorporate KFAS package Their Kalman filter is faster but only for x10. Added MARSSkfas function.

changed Q/R estimation in optim to allow off-diagonal terms.

added V0 estimation option works like other parameters now

LL calc when R=0 Fixed LL calc in MARSSkfas to deal with 0s on diag of Ft[,,1] so can do R=0

replaced show.doc() Replaced show.doc() with RShowDoc()

default miss.value changed Changed to the more standard NA where NA is as.numeric(NA) rather than logical.

4 Changes between versions 2.0 and 1.1

allow B and Z estimation This was the main objective of MARSS 2.0

allow constrained B and Z estimation This was the second main objective of MARSS 2.0. This allows you to have fixed values or shared values in your B or Z matrices.

allow more types of element sharing in the Q and R estimation In MARSS 1.1, you were limited to diagonal, equal var-cov, and unconstrained. Now various types of block-diagonal matrices are allowed.

allow some Q or R variances to be set to 0 This allows partially deterministic systems (Q=0) and systems with no observation error (R=0)

fixed V0=0 case I was using a work-around to do the fixed x at t=0 case (V0=0). I figured out how to do this correctly. There is no iter.V0 control element anymore.

changed logLik conv test I was doing the log-log test against logLik instead of log(logLik). I think the test works better using the log of the log-likelihood.

detect degeneracy and set Q or R element to zero Now instead of the variance walking to log(negative infinity) in an infinite number of interations, the algorithm detects that a variance is going to zero and tries setting it to zero.

more general way to deal with missing values This is described in the Derivations paper. It doesn't affect the user, but allows the code to be expanded to more types of models much more easily.

using list matrices to describe models Now you can essentially write the way your model looks on paper (in matrix form) as a list matrix in R and it will run. No more fixed and free matrices—at least from the user's perspective.

some code optimization I cleaned up some of the things that really slowed down 1.1. 2.0 is now about as fast as 1.0 was.

big revamp of Derivations.pdf I cleaned up my derivation a lot. I'm especially happy with the sections on dealing missing values part of the derivation. It's much more elegant and logical now. The sections on degenerate matrices are cluttered and the notation is painful, but I will leave them be for awhile.

bug in miss.value=NA When miss.value=NA, class for NA was logical. Needed to be numeric.

5 Changes between versions 1.0 and 1.1

Bugs and issues:

error printing formatting issues Cleaned up and standardized the error message printing across all files.

allow NA and NaN to be used for miss.value In vrs 1.0, the code would crash if you tried to use miss.value=NA or NaN.

bug in MARSSmcinit MCMC init function would crash for anything except the default model.

ungraceful exiting when minit > maxit This was not being caught in is.marss().

ungraceful exiting when method=BFGS threw error This was actually a formatting issue. I had the try() error appended to the errors element of the output. I removed the try() output from the errors messages.

Add more info to ?MARSS and help("MARSS-package") Changed MARSS.Rd and MARSS-package to have reference to user guide, index, and MARSS-package help page.

Change convergence test In the convergence diagnostics test, we check that the slope of logLik vs (log iteration num) is close to zero. This is a standard convergence test. But Shumway and Stoffers code uses a delta logLik test which checks that the logLik.new-logLik.old is less than some absolute (user specified) tolerance. This turns out to be a bad convergence test because the log-log plot (described above) can still have a fairly clear slope. I switched over to using the log-log test as the default test, but I allow the user to specify a abstol (delta logLik) if they want that instead. This change slows down model fitting considerably but model fits that are actually converged.

fix to is.design() function A design matrix must have more or equal rows than columns.

R was changing dims on some matrices in MARSSkf R has a flaw in terms of how it behaves when you subscript a matrix and the new matrix has a dimension length of 1 for one (or more dimensions). For example, if a=array(0,dim=c(1,2,4)), then a[„1] is no longer a matrix but instead is a vector and dim(a[„1]) is NULL. This can cause all sorts of mysterious bugs. Sometimes adding drop=FALSE will prevent this unpleasant behavior. If b=matrix(0,2,2), dim(b[,1,drop=FALSE]) is c(2,1) while dim(b[,1]) is NULL. drop=FALSE works great with 2-dimensional matrices, but with 3-dimensional matrices it doesn't work. If a=array(0,dim=c(1,2,4)), dim(a[„1,drop=FALSE]) is c(1,2,1) instead of c(1,2) which is what you want if a[„1] is what is going to appear in some matrix operation. This problem came up in the Kt[, , t] %*% innov[, t] line in MARSSkf. Normally Kt[„t] is square and a square matrix or a scalar is returned, but if Kt[„t] happened to be something like dim=c(1,3,20) then Kt[„t] returned a VECTOR of length 3. In this case, Kt[, , t] %*% innov[, t] crashed the code. I had to use a kluge to force R to keep the dimensions after subscripting. This bug only occurred in models where Z is not a design matrix.

formatting issues in summary(marssm object) The naming of elements in the model matrices did not match summary(marssMLE object).

allow list matrices In version 2.0, the standard way to specify model parameters with fixed and free values will be with a list matrix. a=matrix(list(0,"a",1,"a")) for example. I changed checkpopWrap() and as_marssm() to allow this although the documentation will be updated when 2.0 is released as this will mean a fairly major revision to the user guide to emphasize the list matrices over the text shortcuts. The use of factor() will be de-emphasized although that feature will remain.

added function MARSSoptions() This allows you to change the defaults for the MARSS() function. See ?MARSSoptions.

added function MARSSLLprofile() This allows you to plot some basic log-likelihood profiles. See ?MARSSLLprofile.

typos in user guide Made some updates to the text in the user guide per comments from our August 2010 workshop.