Appendix D: Model Code

Philipp Neubauer[1], Edward R. Abraham[1], and Katrin Berkenbusch[1]

[1]**Dragonfly Data Science, PO Box 27535, Wellington 6141, New Zealand**

The models included some additions of small quantities to some parameters to improve numerical stability for gamma models.

1 BASE MODEL

```
model{
    for (s in 1:surveys){

        #observation model - only observed at some time points
        abu[s] ~ dnorm(t_abu[bs[s],ys[s]],known_tau[s])T(0,)
    }

    for (b in 1:beaches){

        #starting values a year 1
        t_abu[b,1] ~ dgamma(nu[b,1]*sc[b],sc[b])
        nu[b,1]~ dgamma(mu[b]*0.001,0.001)T(1,)

        # cross-correlation bits
        dby[b,1] <- t_abu[b,1]-mu[b]
        dbysq[b,1] <- dby[b,1]*dby[b,1]

        for (y in 2:nyears){

            # process model - add constant for numerical stability
            t_abu[b,y] ~ dgamma(nu[b,y]*sc[b]+0.0001,sc[b])
            nu[b,y] <- mu[b] + rho[b]*(t_abu[b,y-1]-mu[b])

            # cross-correlation bits
            dby[b,y] <- t_abu[b,y] - mb[b]
            dbysq[b,y] <- dby[b,y]*dby[b,y]
}

        # beach parameters
        mu[b] ~ dgamma(0.5,0.001)
        rho[b] ~ dbeta(d,1)
        # added constant for numerical stability
        sc[b] ~ dgamma(a+0.0001,a+0.0001)

        # cross-correlation bits
        # added constant for numerical stability
        sdb[b] <-  sqrt(sum((dbysq[b,1:years]+0.0001)/years))
        mb[b] <- mean(t_abu[b,1:years])

        # calculate empirical cross-correlation
  for(j in 1:beaches) {
            ccv[b,j] <- (dby[b,1:years] %*% dby[j,1:years])/(years*sdb[b]*sdb[j
        }
    }
```

```
    #hyper-priors
    alpha ~ dgamma(0.0001,0.0001)
    delta ~ dgamma(1,1)
}
```

2 EXTENDED MODEL: ADDED CLOSURE EFFECT AND HGF OBSERVATION MODEL

```
model{

    ## AKI  forum observation model
    for (s in 1:surveys){

        abu[s] ~ dnorm(t_abu[bs[s],ys[s]],known_tau[s])T(0,)
    }

    ## HGF  forum observation model
    for (hs in 1:HGF.surveys){

        HGF.abu[hs] ~ dnorm(t_abu[HGF.bs[hs],HGF.ys[hs]]+offset[HGF.ns[hs]]*ove

    }

    ## hierarchical prior for HGF offset
    for (hb in 1:HGF.beach){
        offset[hb] ~ dnorm(offsmean,offsetvar+0.001)
    }

    offsmean ~ dnorm(0,0.001)
    offsetvar ~ dgamma(0.001,0.001)

    for (b in 1:beaches){
        ##starting values a year 1
        t_abu[b,1] ~ dgamma(nu[b,1]*sc[b],sc[b])
        nu[b,1]~ dgamma(exp(mu[b])*3e-3,3e-3)T(1,)

        # cross-correlation bits
        dby[b,1] <- t_abu[b,1]-mu[b]
        dbysq[b,1] <- dby[b,1]*dby[b,1]

        for (y in 2:nyears){

            # process model - add constant for numerical stability
            t_abu[b,y] ~ dgamma(nu[b,y]*sc[b]+0.0001,sc[b])

            ## Trend formulation for beach closures
            trend[b,y] <- exp(mu[b] + beta[b] * closures[b,y-1])
            nu[b,y] <- trend[b,y] + rho[b]*(t_abu[b,y-1]-trend[b,y])

            # cross-correlation bits
            dby[b,y] <- t_abu[b,y] - mb[b]
            dbysq[b,y] <- dby[b,y]*dby[b,y]
}
```

```
        # cross-correlation bits
        # added constant for numerical stability
sdb[b] <-  sqrt(sum((dbysq[b,1:years]+0.0001)/years))
        mb[b] <- mean(t_abu[b,1:years])

        # calculate empirical cross-correlation
  for(j in 1:beaches) {
            ccv[b,j] <- (dby[b,1:years] %*% dby[j,1:years])/((years)*sdb[b]*sdb
        }

        # beach parameters
        mu[b] ~ dnorm(0,1e9)
        rho[b] ~ dbeta(d,1)
        sc[b] ~ dgamma(a+0.00001,a+0.00001)
        beta[b] ~ dnorm(bmu,btau)

    }

    # hyper-prior
    bmu ~ dnorm(0,0.01)
    btau ~ dgamma(0.01,0.01)
    a ~ dgamma(0.0001,0.0001)
    d ~ dgamma(1,1)
}
```