**zone**

$ zoneC :List of 16

$ zoneD :List of 11

$ glb :List of 8

$ constants :num [1:32, 1:16] 1 6 24 2.5 135.3 ...

$ product :num [1:81, 1:6, 1:16] 207 198 190 182 175 ...

$ ctrl :List of 11

$ zone1 :List of 13

**zoneC**

List of 16

$ :List of 19

..- attr(\*, "class")= chr "abpop"

$ :List of 19

..- attr(\*, "class")= chr "abpop"

$ :List of 19

..- attr(\*, "class")= chr "abpop"

…

**abpop**

List of 19

$ Me : num 0.148

$ R0 : num 91129

$ B0 : num 201

$ ExB0 : num 207

$ MSY : num 12.7

$ MSYDepl : num 0.249

$ bLML : num 154

$ popq : num 0.00121

$ SaM : num 133

$ popdef : Named num [1:19] 26.45 135.13 170.73 3.39 -16 ...

..- attr(\*, "names")= chr [1:19] "DLMax" "L50" "L95" "SigMax" ...

$ LML : Named num [1:47] 132 132 132 132 132 132 132 132 132 132 ...

..- attr(\*, "names")= chr [1:47] "1973" "1974" "1975" "1976" ...

$ G : 'STM' num [1:105, 1:105] 3.16e-14 2.36e-12 1.26e-10 4.81e-09 1.30e-07 ...

$ Maturity: num [1:105] 1.43e-07 1.82e-07 2.32e-07 2.95e-07 3.76e-07 ...

$ WtL : num [1:105] 0.000351 0.003438 0.013072 0.033719 0.070319 ...

$ Emergent: num [1:105] 1.92e-16 3.44e-16 6.18e-16 1.11e-15 1.99e-15 ...

$ Select : num [1:105, 1:47] 2.87e-127 2.61e-125 2.37e-123 2.16e-121 1.96e-119 ...

$ SelWt : num [1:105, 1:47] 1.01e-130 8.97e-128 3.10e-125 7.28e-123 1.38e-120 ...

$ MatWt : num [1:105] 5.02e-11 6.27e-10 3.03e-09 9.96e-09 2.64e-08 ...

$ SAU : num 6

- attr(\*, "class")= chr "abpop"

**zoneD**

List of 11

$ SAU : num [1:16] 6 6 7 7 8 8 9 9 10 10 ...

$ matureB : num [1:47, 1:16] 201 0 0 0 0 ...

$ exploitB: num [1:47, 1:16] 207 0 0 0 0 ...

$ catch : num [1:47, 1:16] 0 0 0 0 0 0 0 0 0 0 ...

$ harvestR: num [1:47, 1:16] 0 0 0 0 0 0 0 0 0 0 ...

$ cpue : num [1:47, 1:16] 251 0 0 0 0 ...

$ recruit : num [1:47, 1:16] 91129 0 0 0 0 ...

$ deplsB : num [1:47, 1:16] 1 0 0 0 0 ...

$ depleB : num [1:47, 1:16] 0.931 0 0 0 0 ...

$ catchN : num [1:105, 1:47, 1:16] 0 0 0 0 0 0 0 0 0 0 ...

$ Nt : num [1:105, 1:47, 1:16] 9.11e+04 1.86e-07 9.94e-06 3.78e-04 1.02e-02 ...

**glb**

List of 8

$ numpop : num 16

$ nSAU : num 8

$ midpts : num [1:105] 2 4 6 8 10 12 14 16 18 20 ...

$ Nclass : num 105

$ Nyrs : int 47

$ larvdisp: num 0.005

$ sauindex: num [1:16] 1 1 2 2 3 3 4 4 5 5 ...

$ move : num [1:16, 1:16] 0.9975 0.0025 0 0 0 ...

$ SAUpop : num [1:27] 6 6 7 7 7 8 8 9 9 10 ...

$ saunames: num [1:8] 6 7 8 9 10 11 12 13

**constants** num [1:32, 1:16] 1 6 24 2.5 135.3 ...

**product** num [1:81, 1:6, 1:16] 207 198 190 182 175 ...

**ctrl**

List of 11

$ runlabel : chr "zoneX"

$ datafile : chr "zonewest.csv"

$ batch : num 0

$ reps : num 100

$ randseed : num 4024136

$ randseedP : num 8e+06

$ withsigR : num 0.3

$ withsigB : num 0.05

$ withsigCE : num 1e-08

$ catches : num 47

$ projection: num 30

$ bysau : num 1

**zone1**

List of 13

$ SAUnames : chr [1:8] "blk6" "blk7" "blk8" "blk9" ...

$ SAUpop : num [1:8] 2 2 2 2 2 2 2 2

$ minc : num 2

$ cw : num 2

$ larvdisp : num 0.005

$ randomseed: num 4024136

$ initLML : num 140

$ condC :List of 8

$ projC :List of 8

$ globals :List of 8 = glb

$ ctrl :List of 11 = ctrl

$ catches : num 47

$ projyrs : num 30

**condC**

List of 8

$ histCatch: num [1:47, 1:8] 0 8 21 30 19 28 31 42 48 30 ...

$ histyr : num [1:47, 1:2] 1973 1974 1975 1976 1977 ...

$ histCE : num [1:28, 1:8] NA NA NA NA NA ...

$ yearCE : num [1:28] 1992 1993 1994 1995 1996 ...

$ initdepl : num [1:8] 0.7 0.7 0.7 0.7 0.7 0.7 0.7 0.7

$ compdat : NULL

$ Sel : NULL

$ SelWt : NULL

**projC**

List of 8

$ projLML : num [1:30] 145 145 145 145 145 145 145 145 145 145 ...

$ HS : chr "MCDA"

$ HSdetail: num 1

$ projyrs : num 30

$ inityrs : num 10

$ Sel : NULL

$ SelWt : NULL

$ histCE : num [1:28, 1:8] NA NA NA NA NA ...

**zoneDD**

List of 11

$ SAU : num [1:27] 6 6 7 7 7 8 8 9 9 10 ...

$ matureB : num [1:47, 1:27] 199 195 185 172 165 ...

$ exploitB: num [1:47, 1:27] 191 208 204 194 181 ...

$ catch : num [1:47, 1:27] 0 4.26 11.15 15.93 10.09 ...

$ harvestR: num [1:47, 1:27] 0 0.0205 0.0546 0.0819 0.0557 ...

$ cpue : num [1:47, 1:27] 440 458 442 415 392 ...

$ recruit : num [1:47, 1:27] 101716 101548 101092 100398 100007 ...

$ deplsB : num [1:47, 1:27] 1 0.98 0.93 0.862 0.828 ...

$ depleB : num [1:47, 1:27] 0.915 1.092 1.072 1.02 0.951 ...

$ catchN : num [1:105, 1:47, 1:27] 0 0 0 0 0 0 0 0 0 0 ...

$ Nt : num [1:105, 1:47, 1:27] 1.02e+05 1.24e-06 5.10e-05 1.52e-03 3.27e-02 ...

**zoneDDR**

List of 14

$ SAU : num [1:27] 6 6 7 7 7 8 8 9 9 10 ...

$ matureB : num [1:47, 1:27, 1:100] 199 195 185 172 165 ...

$ exploitB: num [1:47, 1:27, 1:100] 191 208 204 194 181 ...

$ catch : num [1:47, 1:27, 1:100] 0 4.26 11.15 15.93 10.09 ...

$ acatch : num [1:47, 1:8, 1:100] 0 0 0 0 0 0 0 0 0 0 ...

$ harvestR: num [1:47, 1:27, 1:100] 0 0.0205 0.0546 0.0819 0.0557 ...

$ cpue : num [1:47, 1:27, 1:100] 440 458 442 415 392 ...

$ cesau : num [1:47, 1:8, 1:100] 0 0 0 0 0 0 0 0 0 0 ...

$ catsau : num [1:47, 1:8, 1:100] 0 0 0 0 0 0 0 0 0 0 ...

$ recruit : num [1:47, 1:27, 1:100] 101716 101548 101092 100398 100007 ...

$ deplsB : num [1:47, 1:27, 1:100] 0 0 0 0 0 0 0 0 0 0 ...

$ depleB : num [1:47, 1:27, 1:100] 0 0 0 0 0 0 0 0 0 0 ...

$ catchN : num [1:105, 1:47, 1:27, 1:100] 0 0 0 0 0 0 0 0 0 0 ...

$ Nt : num [1:105, 1:47, 1:27, 1:100] 1.02e+05 1.24e-06 5.10e-05 1.52e-03 3.27e-02 ...

**zoneDP**

List of 14

$ SAU : num [1:27] 6 6 7 7 7 8 8 9 9 10 ...

$ matureB : num [1:30, 1:27, 1:100] 58.3 59.1 61.1 59.8 60.4 ...

$ exploitB: num [1:30, 1:27, 1:100] 66.5 63.4 61.9 63.9 61.1 ...

$ catch : num [1:30, 1:27, 1:100] 9.31 9.61 6.23 8.18 8.38 ...

$ acatch : num [1:30, 1:8, 1:100] 13.6 14.3 12.9 12.9 12.9 ...

$ harvestR: num [1:30, 1:27, 1:100] 0.14 0.151 0.101 0.128 0.137 ...

$ cpue : num [1:30, 1:27, 1:100] 138 131 131 133 127 ...

$ cesau : num [1:30, 1:8, 1:100] 123 115 116 120 121 ...

$ catsau : num [1:30, 1:8, 1:100] 16.43 15.72 9.86 12.97 13.49 ...

$ recruit : num [1:30, 1:27, 1:100] 87848 75856 86055 94454 47903 ...

$ deplsB : num [1:30, 1:27, 1:100] 0.293 0.297 0.307 0.3 0.303 ...

$ depleB : num [1:30, 1:27, 1:100] 0.349 0.333 0.325 0.335 0.321 ...

$ catchN : num [1:105, 1:30, 1:27, 1:100] 1.75e-131 8.81e-128 3.26e-124 8.77e-121 1.71e-

$ Nt : num [1:105, 1:30, 1:27, 1:100] 8.78e+04 9.04e-07 3.70e-05 1.10e-03 2.38e-02 ...