

HomeWork-5

Solayman Hossain Emon; ID: 80744292

In [92]: `using CSV, DataFrames, GLM, RCall, EvalMetrics, MLBase`

In [93]: `ENV["COLUMNs"] = 1000`

Out[93]: 1000

In [94]: `data = CSV.read("divorce.csv", DataFrame)`

Out[94]: 170 rows × 55 columns

	Atr1	Atr2	Atr3	Atr4	Atr5	Atr6	Atr7	Atr8	Atr9	Atr10	Atr11	Atr12	Atr13	Atr14	Atr15	Atr16	Atr17	Atr18	Atr19	Atr2
	Int64	Int64	Int64	Int64	Int64	Int64	Int64	Int64	Int64	Int64	Int64	Int64	Int64	Int64	Int64	Int64	Int64	Int64	Int64	Int6
1	2	2	4	1	0	0	0	0	0	0	1	0	1	1	0	1	0	0	0	
2	4	4	4	4	4	0	0	4	4	4	4	3	4	0	4	4	4	4	3	
3	2	2	2	2	1	3	2	1	1	2	3	4	2	3	3	3	3	3	3	
4	3	2	3	2	3	3	3	3	3	3	4	3	3	4	3	3	3	3	3	
5	2	2	1	1	1	1	0	0	0	0	0	1	0	1	1	1	1	1	2	
6	0	0	1	0	0	2	0	0	0	1	0	2	1	0	2	0	2	1	0	
7	3	3	3	2	1	3	4	3	2	2	2	2	2	3	2	3	3	3	3	
8	2	1	2	2	2	1	0	3	3	2	4	3	2	3	4	3	2	3	2	
9	2	2	1	0	0	4	1	3	3	3	3	3	3	3	3	3	3	3	3	
10	1	1	1	1	1	2	0	2	2	2	3	0	0	2	1	0	1	2	1	
11	4	4	4	3	4	0	0	4	4	3	4	4	4	4	4	3	4	4	4	

	Atr1	Atr2	Atr3	Atr4	Atr5	Atr6	Atr7	Atr8	Atr9	Atr10	Atr11	Atr12	Atr13	Atr14	Atr15	Atr16	Atr17	Atr18	Atr19	Atr2
	Int64	Int64	Int64	Int64	Int64	Int64	Int64	Int64	Int64	Int64	Int64	Int64	Int64	Int64	Int64	Int64	Int64	Int64	Int64	Int6
12	4	4	4	3	4	0	0	4	4	3	4	4	4	4	4	3	4	4	4	
13	3	4	3	4	3	0	1	4	3	4	3	4	3	4	3	4	3	4	3	
14	3	4	3	4	3	0	1	4	3	4	3	4	3	4	3	4	3	4	3	
15	3	4	3	4	3	0	1	4	3	4	3	4	3	4	3	4	3	4	3	
16	4	4	3	2	4	0	0	4	3	2	4	4	4	4	3	2	4	4	4	
17	4	4	3	2	4	0	0	4	3	2	4	4	4	4	3	2	4	4	4	
18	4	4	4	3	4	0	0	4	4	3	4	4	4	4	4	3	4	4	4	
19	3	3	4	4	3	1	1	3	4	4	3	3	3	3	4	4	3	3	3	
20	4	4	4	3	4	0	0	4	4	3	4	4	4	4	4	3	4	4	4	
21	4	3	3	3	4	1	0	3	3	3	4	3	4	3	3	3	4	3	4	
22	4	3	3	3	4	1	0	3	3	3	4	3	4	3	3	3	4	3	4	
23	3	4	4	4	3	0	1	4	4	4	3	4	3	4	4	4	3	4	3	
24	3	3	3	4	3	1	1	3	3	4	3	3	3	3	3	4	3	3	3	
25	4	2	3	4	4	2	0	2	3	4	4	2	4	2	3	4	4	2	4	
26	3	3	3	4	3	1	1	3	3	4	3	3	3	3	3	4	3	3	3	
27	3	3	4	3	3	1	1	3	4	3	3	3	3	3	4	3	3	3	3	
28	3	3	3	4	3	1	1	3	3	4	3	3	3	3	3	4	3	3	3	
29	3	4	3	2	3	0	1	4	3	2	3	4	3	4	3	2	3	4	3	
30	4	3	3	2	4	1	0	3	3	2	4	3	4	3	3	2	4	3	4	
:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	

In [95]:

```
dat = data[!, Not(:Class)];
```

Atr1+Atr2+Atr3+Atr4+Atr5+Atr6+Atr7+Atr8+Atr9+Atr10+Atr11+Atr12+Atr13+Atr14+Atr15+Atr16+Atr17+Atr18+Atr19+Atr20+Atr21+Atr22·

Train & Test Dataset

```
In [96]: n, p = size(data)
```

```
Out[96]: (170, 55)
```

```
In [97]: using MLJ
traindf, testdf = partition(data, 0.7, shuffle=true); # 70:30 split
```

```
In [98]: print(testdf.Class)
```

```
[1, 1, 0, 0, 0, 1, 1, 0, 0, 0, 0, 1, 1, 1, 1, 1, 1, 0, 0, 1, 1, 1, 1, 0, 1, 0, 0, 1, 1, 0, 1, 0, 0, 1, 1, 0, 0, 1, 1, 1,
0, 0, 1, 1, 0, 1, 0, 0, 0, 0]
```

(a) Fitting a Classification Tree

```
In [99]: m = @formula(Class ~ Atr1+Atr2+Atr3+Atr4+Atr5+Atr6+Atr7+Atr8+Atr9+Atr10+Atr11+Atr12+Atr13+Atr14+Atr15+Atr16+Atr17+Atr18+A
```

```
Out[99]: FormulaTerm
Response:
  Class(unknown)
Predictors:
  Atr1(unknown)
  Atr2(unknown)
  Atr3(unknown)
  Atr4(unknown)
  Atr5(unknown)
  Atr6(unknown)
  Atr7(unknown)
  Atr8(unknown)
  Atr9(unknown)
  Atr10(unknown)
  Atr11(unknown)
  Atr12(unknown)
  Atr13(unknown)
  Atr14(unknown)
  Atr15(unknown)
```

Atr16(unknown)
Atr17(unknown)
Atr18(unknown)
Atr19(unknown)
Atr20(unknown)
Atr21(unknown)
Atr22(unknown)
Atr23(unknown)
Atr24(unknown)
Atr25(unknown)
Atr26(unknown)
Atr27(unknown)
Atr28(unknown)
Atr29(unknown)
Atr30(unknown)
Atr31(unknown)
Atr32(unknown)
Atr33(unknown)
Atr34(unknown)
Atr35(unknown)
Atr36(unknown)
Atr37(unknown)
Atr38(unknown)
Atr39(unknown)
Atr40(unknown)
Atr41(unknown)
Atr42(unknown)
Atr43(unknown)
Atr44(unknown)
Atr45(unknown)
Atr46(unknown)
Atr47(unknown)
Atr48(unknown)
Atr49(unknown)
Atr50(unknown)
Atr51(unknown)
Atr52(unknown)
Atr53(unknown)
Atr54(unknown)

In [100...

```
@rlibrary rpart
```

```
treeModel = rpart(m, data=traindf)
```

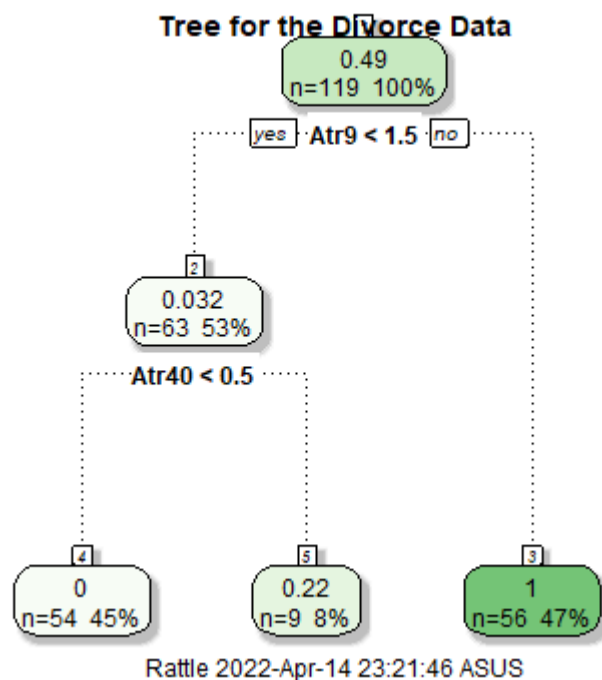
```
Out[100... RObject{VecSxp}
n= 119
```

```
node), split, n, deviance, yval
* denotes terminal node
```

```
1) root 119 29.731090 0.48739500
 2) Atr9 < 1.5 63 1.936508 0.03174603
   4) Atr40 < 0.5 54 0.000000 0.00000000 *
   5) Atr40 >= 0.5 9 1.555556 0.22222220 *
 3) Atr9 >= 1.5 56 0.000000 1.00000000 *
```

```
In [101... @rlibrary rattle

fancyRpartPlot(treeModel, main="Tree for the Divorce Data")
```



```
Out[101... RObject{NilSxp}
NULL
```

```
In [102... printcp(treeModel)
```

```

Regression tree:
(function (formula, data, weights, subset, na.action = na.rpart,
  method, model = FALSE, x = FALSE, y = TRUE, parms, control,
  cost, ...)
{
  Call <- match.call()
  if (is.data.frame(model)) {
    m <- model
    model <- FALSE
  }
  else {
    indx <- match(c("formula", "data", "weights", "subset"),
      names(Call), nomatch = 0)
    if (indx[1] == 0)
      stop("a 'formula' argument is required")
    temp <- Call[c(1, indx)]
    temp$na.action <- na.action
    temp[[1]] <- quote(stats::model.frame)
    m <- eval.parent(temp)
  }
  Terms <- attr(m, "terms")
  if (any(attr(Terms, "order") > 1))
    stop("Trees cannot handle interaction terms")
  Y <- model.response(m)
  wt <- model.weights(m)
  if (any(wt < 0))
    stop("negative weights not allowed")
  if (!length(wt))
    wt <- rep(1, nrow(m))
  offset <- model.offset(m)
  X <- rpart.matrix(m)
  nobs <- nrow(X)
  nvar <- ncol(X)
  if (missing(method)) {
    method <- if (is.factor(Y) || is.character(Y))
      "class"
    else if (inherits(Y, "Surv"))
      "exp"
    else if (is.matrix(Y))
      "poisson"
    else "anova"
  }
  if (is.list(method)) {
    mlist <- method
    method <- "user"
  }
}

```

```

init <- if (missing(parms))
  mlist$init(Y, offset, wt = wt)
else mlist$init(Y, offset, parms, wt)
keep <- rpartcallback(mlist, nobs, init)
method.int <- 4
parms <- init$parms
}
else {
  method.int <- pmatch(method, c("anova", "poisson", "class",
    "exp"))
  if (is.na(method.int))
    stop("Invalid method")
  method <- c("anova", "poisson", "class", "exp")[method.int]
  if (method.int == 4)
    method.int <- 2
  init <- if (missing(parms))
    get(paste("rpart", method, sep = "."), envir = environment())(Y,
      offset, , wt)
  else get(paste("rpart", method, sep = "."), envir = environment())(Y,
    offset, parms, wt)
  ns <- asNamespace("rpart")
  if (!is.null(init$print))
    environment(init$print) <- ns
  if (!is.null(init$summary))
    environment(init$summary) <- ns
  if (!is.null(init$text))
    environment(init$text) <- ns
}
Y <- init$y
xlevels <- .getXlevels(Terms, m)
cats <- rep(0, ncol(X))
if (!is.null(xlevels)) {
  xlevels <- xlevels[names(xlevels) %in% colnames(X)]
  cats[match(names(xlevels), colnames(X))] <- unlist(lapply(xlevels,
    length))
}
extraArgs <- list(...)
if (length(extraArgs)) {
  controlargs <- names(formals(rpart.control))
  indx <- match(names(extraArgs), controlargs, nomatch = 0)
  if (any(indx == 0))
    stop(gettextf("Argument %s not matched", names(extraArgs)[indx ==
      0]), domain = NA)
}
controls <- rpart.control(...)

```

```

if (!missing(control))
  controls[names(control)] <- control
xval <- controls$xval
if (is.null(xval) || (length(xval) == 1 && xval == 0) ||
    method == "user") {
  xgroups <- 0
  xval <- 0
}
else if (length(xval) == 1) {
  xgroups <- sample(rep(1:xval, length.out = nobs), nobs,
    replace = FALSE)
}
else if (length(xval) == nobs) {
  xgroups <- xval
  xval <- length(unique(xgroups))
}
else {
  if (!is.null(attr(m, "na.action"))) {
    temp <- as.integer(attr(m, "na.action"))
    xval <- xval[-temp]
    if (length(xval) == nobs) {
      xgroups <- xval
      xval <- length(unique(xgroups))
    }
    else stop("Wrong length for 'xval'")
  }
  else stop("Wrong length for 'xval'")
}
if (missing(cost))
  cost <- rep(1, nvar)
else {
  if (length(cost) != nvar)
    stop("Cost vector is the wrong length")
  if (any(cost <= 0))
    stop("Cost vector must be positive")
}
tfun <- function(x) if (is.matrix(x))
  rep(is.ordered(x), ncol(x))
else is.ordered(x)
labs <- sub("^(.*)`$", "\\1", attr(Terms, "term.labels"))
isord <- unlist(lapply(m[labs], tfun))
storage.mode(X) <- "double"
storage.mode(wt) <- "double"
temp <- as.double(unlist(init$params))
if (!length(temp))

```



```

temp <- 0
rpfit <- .Call(C_rpart, ncat = as.integer(cats * !isord),
  method = as.integer(method.int), as.double(unlist(controls)),
  temp, as.integer(xval), as.integer(xgroups), as.double(t(init$y)),
  X, wt, as.integer(init$numy), as.double(cost))
nsplit <- nrow(rpfit$split)
ncat <- if (!is.null(rpfit$csplit))
  nrow(rpfit$csplit)
else 0
if (nsplit == 0)
  xval <- 0
numcp <- ncol(rpfit$cptable)
temp <- if (nrow(rpfit$cptable) == 3)
  c("CP", "nsplit", "rel error")
else c("CP", "nsplit", "rel error", "xerror", "xstd")
dimnames(rpfit$cptable) <- list(temp, 1:numcp)
tname <- c("<leaf>", colnames(X))
splits <- matrix(c(rpfit$split[, 2:3], rpfit$dsplit), ncol = 5,
  dimnames = list(tname[rpfit$split[, 1] + 1], c("count",
    "ncat", "improve", "index", "adj")))
index <- rpfit$inode[, 2]
nadd <- sum(isord[rpfit$split[, 1]])
if (nadd > 0) {
  newc <- matrix(0, nadd, max(cats))
  cvar <- rpfit$split[, 1]
  indx <- isord[cvar]
  cdir <- splits[indx, 2]
  ccut <- floor(splits[indx, 4])
  splits[indx, 2] <- cats[cvar[indx]]
  splits[indx, 4] <- ncat + 1:nadd
  for (i in 1:nadd) {
    newc[i, 1:(cats[(cvar[indx])[i]])] <- -as.integer(cdir[i])
    newc[i, 1:ccut[i]] <- as.integer(cdir[i])
  }
  catmat <- if (ncat == 0)
    newc
  else {
    cs <- rpfit$csplit
    ncs <- ncol(cs)
    ncc <- ncol(newc)
    if (ncs < ncc)
      cs <- cbind(cs, matrix(0, nrow(cs), ncc - ncs))
    rbind(cs, newc)
  }
  ncat <- ncat + nadd
}

```

```

}
else catmat <- rpfit$csplit
if (nsplit == 0) {
  frame <- data.frame(row.names = 1, var = "<leaf>", n = rpfit$inode[,
    5], wt = rpfit$dnode[, 3], dev = rpfit$dnode[, 1],
    yval = rpfit$dnode[, 4], complexity = rpfit$dnode[,
    2], ncompete = 0, nsurrogate = 0)
}
else {
  temp <- ifelse(index == 0, 1, index)
  svar <- ifelse(index == 0, 0, rpfit$split[temp, 1])
  frame <- data.frame(row.names = rpfit$inode[, 1], var = tname[svar +
    1], n = rpfit$inode[, 5], wt = rpfit$dnode[, 3],
    dev = rpfit$dnode[, 1], yval = rpfit$dnode[, 4],
    complexity = rpfit$dnode[, 2], ncompete = pmax(0,
    rpfit$inode[, 3] - 1), nsurrogate = rpfit$inode[,
    4])
}
if (method.int == 3) {
  numclass <- init$numresp - 2
  nodeprob <- rpfit$dnode[, numclass + 5]/sum(wt)
  temp <- pmax(1, init$counts)
  temp <- rpfit$dnode[, 4 + (1:numclass)] %*% diag(init$parms$prior/temp)
  yprob <- temp/rowSums(temp)
  yval2 <- matrix(rpfit$dnode[, 4 + (0:numclass)], ncol = numclass +
    1)
  frame$yval2 <- cbind(yval2, yprob, nodeprob)
}
else if (init$numresp > 1)
  frame$yval2 <- rpfit$dnode[, -(1:3), drop = FALSE]
if (is.null(init$summary))
  stop("Initialization routine is missing the 'summary' function")
functions <- if (is.null(init$print))
  list(summary = init$summary)
else list(summary = init$summary, print = init$print)
if (!is.null(init$text))
  functions <- c(functions, list(text = init$text))
if (method == "user")
  functions <- c(functions, mlist)
where <- rpfit$which
names(where) <- row.names(m)
ans <- list(frame = frame, where = where, call = Call, terms = Terms,
  cptable = t(rpfit$cptable), method = method, parms = init$parms,
  control = controls, functions = functions, numresp = init$numresp)
if (nsplit)

```

```

      ans$splits = splits
    if (ncat > 0)
      ans$csplit <- catmat + 2
    if (nsplit)
      ans$variable.importance <- importance(ans)
    if (model) {
      ans$model <- m
      if (missing(y))
        y <- FALSE
    }
    if (y)
      ans$y <- Y
    if (x) {
      ans$x <- X
      ans$wt <- wt
    }
    ans$ordered <- isord
    if (!is.null(attr(m, "na.action")))
      ans$na.action <- attr(m, "na.action")
    if (!is.null(xlevels))
      attr(ans, "xlevels") <- xlevels
    if (method == "class")
      attr(ans, "ylevels") <- init$ylevels
    class(ans) <- "rpart"
    ans
  })(formula = Class ~ Atr1 + Atr2 + Atr3 + Atr4 + Atr5 + Atr6 +
    Atr7 + Atr8 + Atr9 + Atr10 + Atr11 + Atr12 + Atr13 + Atr14 +
    Atr15 + Atr16 + Atr17 + Atr18 + Atr19 + Atr20 + Atr21 + Atr22 +
    Atr23 + Atr24 + Atr25 + Atr26 + Atr27 + Atr28 + Atr29 + Atr30 +
    Atr31 + Atr32 + Atr33 + Atr34 + Atr35 + Atr36 + Atr37 + Atr38 +
    Atr39 + Atr40 + Atr41 + Atr42 + Atr43 + Atr44 + Atr45 + Atr46 +
    Atr47 + Atr48 + Atr49 + Atr50 + Atr51 + Atr52 + Atr53 + Atr54,
    data = list(c(3, 4, 3, 4, 0, 1, 3, 3, 2, 3, 3, 3, 0, 3, 4,
0, 3, 0, 0, 3, 0, 0, 0, 3, 3, 3, 1, 3, 0, 4, 1, 0, 0, 2,
4, 0, 0, 3, 2, 3, 0, 0, 0, 0, 4, 2, 3, 3, 0, 0, 4, 3, 1,
3, 0, 0, 2, 0, 3, 4, 4, 4, 0, 0, 3, 4, 1, 0, 4, 0, 0, 4,
0, 2, 2, 3, 3, 3, 2, 0, 3, 0, 0, 4, 0, 3, 0, 1, 1, 0, 0,
0, 0, 3, 0, 0, 3, 0, 0, 0, 0, 3, 0, 4, 0, 4, 2, 4, 3, 4,
0, 0, 3, 3, 3, 0, 3, 3, 4), c(2, 2, 4, 4, 1, 0, 3, 4, 2,
4, 1, 4, 0, 2, 3, 0, 3, 0, 1, 3, 0, 0, 0, 0, 3, 3, 1, 2,
0, 4, 0, 1, 0, 2, 3, 0, 0, 3, 0, 3, 0, 0, 0, 0, 4, 0, 4,
3, 0, 0, 3, 3, 2, 4, 0, 0, 1, 2, 2, 3, 2, 3, 0, 0, 2, 2,
1, 0, 4, 2, 1, 4, 0, 2, 1, 3, 4, 3, 2, 0, 4, 0, 1, 4, 0,
3, 0, 0, 1, 0, 0, 0, 0, 3, 0, 0, 3, 1, 0, 0, 1, 3, 0, 3,
0, 3, 2, 3, 2, 4, 1, 0, 3, 3, 2, 0, 4, 3, 4), c(2, 3, 2,

```

```

4, 0, 1, 2, 3, 4, 3, 1, 3, 2, 3, 2, 0, 4, 0, 0, 3, 0, 0,
0, 0, 2, 3, 0, 4, 0, 4, 1, 1, 0, 2, 2, 0, 0, 3, 2, 2, 1,
0, 0, 0, 3, 1, 4, 2, 1, 0, 3, 3, 0, 3, 1, 0, 0, 2, 4, 3,
3, 4, 0, 0, 3, 3, 2, 0, 4, 0, 2, 4, 0, 1, 1, 3, 3, 3, 3,
0, 3, 0, 0, 4, 0, 3, 0, 0, 1, 0, 1, 0, 0, 3, 0, 0, 4, 0,
0, 0, 1, 3, 0, 3, 0, 3, 3, 4, 3, 4, 0, 1, 3, 3, 3, 1, 3,
4, 3), c(3, 2, 3, 3, 0, 0, 3, 4, 1, 4, 0, 3, 4, 2, 3, 0,
4, 0, 1, 4, 0, 0, 0, 0, 3, 2, 0, 3, 0, 3, 0, 0, 0, 3, 3,
0, 0, 2, 4, 3, 0, 0, 0, 0, 3, 0, 4, 3, 0, 0, 3, 2, 0, 4,
1, 0, 2, 1, 3, 4, 4, 4, 0, 0, 2, 2, 0, 0, 3, 0, 1, 3, 0,
0, 0, 2, 4, 4, 2, 0, 2, 0, 0, 4, 0, 2, 0, 0, 1, 0, 0, 0,
0, 3, 0, 0, 3, 1, 0, 0, 0, 4, 0, 3, 0, 3, 0, 3, 4, 3, 0,
0, 3, 2, 2, 0, 4, 4, 2), c(3, 4, 3, 4, 0, 0, 3, 3, 0, 3,
0, 3, 0, 3, 4, 0, 3, 0, 0, 3, 0, 0, 0, 0, 3, 3, 0, 3, 0,
4, 1, 0, 0, 2, 4, 0, 0, 3, 2, 3, 1, 0, 0, 0, 4, 0, 3, 3,
0, 0, 4, 3, 0, 3, 0, 0, 0, 0, 3, 4, 4, 0, 0, 0, 3, 4, 2,
0, 4, 0, 0, 4, 0, 0, 0, 3, 3, 3, 2, 0, 3, 0, 0, 4, 0, 1,
0, 0, 1, 0, 0, 0, 1, 3, 0, 0, 3, 0, 0, 0, 0, 3, 0, 4, 0,
4, 0, 4, 3, 4, 0, 0, 3, 3, 3, 0, 3, 3, 4), c(2, 2, 0, 2,
0, 0, 1, 0, 0, 0, 0, 0, 0, 2, 1, 0, 1, 0, 0, 1, 2, 2, 0,
0, 1, 1, 0, 2, 0, 0, 1, 1, 0, 3, 1, 2, 0, 1, 2, 1, 0, 0,
1, 0, 0, 0, 0, 1, 1, 0, 1, 1, 0, 0, 0, 0, 0, 0, 2, 1, 2,
0, 0, 0, 2, 2, 1, 0, 0, 1, 0, 0, 0, 4, 2, 1, 0, 1, 2, 1,
0, 0, 1, 0, 0, 3, 1, 1, 0, 0, 0, 0, 0, 1, 1, 0, 1, 0, 2,
0, 2, 1, 0, 1, 0, 1, 1, 1, 2, 0, 0, 2, 1, 1, 2, 0, 0, 1,
0), c(1, 0, 1, 4, 0, 0, 1, 1, 0, 1, 0, 1, 0, 1, 0, 0, 1,
0, 0, 1, 1, 0, 0, 0, 1, 1, 0, 3, 0, 0, 0, 0, 0, 2, 0, 0,
0, 1, 4, 1, 0, 0, 0, 0, 4, 0, 1, 1, 0, 0, 0, 1, 0, 1, 0,
0, 0, 0, 3, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 1,
0, 1, 1, 1, 2, 0, 1, 0, 0, 0, 0, 4, 0, 0, 0, 0, 0, 0, 0,
1, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0,
1, 1, 1, 0, 1, 1, 0), c(2, 2, 4, 4, 0, 1, 3, 4, 0, 4, 0,
4, 0, 2, 3, 0, 3, 0, 0, 3, 0, 0, 0, 0, 3, 3, 0, 2, 0, 4,
0, 0, 0, 1, 3, 0, 0, 3, 3, 3, 0, 0, 0, 0, 2, 0, 4, 3, 0,
0, 3, 3, 0, 4, 0, 0, 0, 0, 2, 3, 2, 0, 0, 0, 2, 2, 2, 0,
4, 0, 0, 4, 0, 3, 0, 3, 4, 3, 2, 0, 4, 0, 2, 4, 0, 3, 0,
0, 1, 0, 1, 0, 0, 3, 1, 0, 3, 0, 0, 1, 0, 3, 0, 3, 0, 3,
0, 3, 2, 4, 1, 0, 3, 3, 2, 0, 4, 3, 4), c(2, 3, 2, 4, 0,
0, 3, 3, 0, 3, 0, 3, 0, 3, 2, 0, 4, 0, 0, 3, 0, 0, 0, 0,
2, 3, 0, 2, 0, 4, 0, 0, 0, 3, 2, 0, 0, 3, 4, 2, 0, 0, 0,
0, 4, 0, 4, 2, 0, 0, 3, 3, 0, 3, 0, 0, 0, 0, 2, 3, 3, 0,
0, 0, 3, 3, 1, 0, 4, 0, 0, 4, 0, 3, 0, 3, 3, 3, 3, 0, 3,
0, 0, 4, 0, 2, 0, 0, 1, 0, 1, 0, 0, 3, 1, 0, 4, 0, 0, 0,
0, 3, 0, 3, 0, 3, 0, 4, 3, 4, 0, 0, 3, 3, 3, 0, 3, 4, 3),
c(3, 2, 3, 3, 0, 0, 3, 4, 0, 4, 0, 3, 2, 2, 3, 0, 4,

```

```

2, 1, 4, 0, 1, 0, 0, 3, 2, 1, 3, 0, 3, 1, 0, 1, 2, 3,
0, 0, 2, 3, 3, 0, 1, 0, 0, 4, 2, 4, 3, 0, 0, 3, 2, 0,
4, 2, 1, 1, 2, 3, 4, 4, 0, 1, 0, 2, 2, 2, 0, 3, 0, 0,
3, 0, 3, 2, 2, 4, 4, 2, 0, 2, 0, 0, 4, 0, 2, 1, 1, 1,
0, 0, 0, 0, 3, 0, 0, 3, 1, 0, 0, 0, 4, 0, 3, 0, 3, 1,
3, 4, 3, 0, 1, 3, 2, 2, 0, 4, 4, 2), c(3, 4, 3, 4, 0,
1, 4, 3, 1, 3, 1, 3, 0, 3, 4, 0, 3, 0, 0, 3, 0, 0, 0,
0, 3, 3, 0, 4, 0, 4, 0, 0, 0, 1, 4, 0, 0, 3, 2, 3, 0,
0, 0, 0, 3, 1, 3, 3, 0, 0, 4, 3, 0, 3, 1, 0, 1, 1, 4,
4, 4, 0, 0, 0, 3, 4, 1, 0, 4, 0, 0, 4, 0, 3, 0, 3, 3,
3, 2, 1, 3, 0, 1, 4, 0, 2, 0, 1, 1, 0, 0, 0, 0, 3, 0,
0, 3, 1, 0, 0, 0, 3, 0, 4, 0, 4, 1, 4, 3, 4, 0, 0, 3,
3, 3, 0, 3, 3, 4), c(2, 2, 4, 4, 0, 0, 3, 4, 0, 4, 0,
4, 0, 2, 3, 0, 3, 1, 0, 3, 0, 0, 0, 1, 3, 3, 0, 3, 0,
4, 0, 0, 1, 2, 3, 0, 0, 3, 3, 3, 1, 1, 0, 0, 4, 0, 4,
3, 0, 1, 3, 3, 0, 4, 2, 0, 1, 1, 3, 3, 2, 0, 0, 0, 2,
2, 1, 0, 4, 0, 1, 4, 0, 3, 1, 3, 4, 3, 2, 1, 4, 1, 0,
3, 0, 2, 1, 0, 0, 0, 0, 1, 0, 3, 0, 0, 3, 2, 0, 1, 2,
3, 0, 3, 0, 3, 1, 3, 2, 4, 0, 2, 3, 3, 2, 0, 4, 3, 4),
c(3, 4, 3, 4, 0, 1, 3, 3, 1, 3, 0, 3, 0, 3, 4, 1, 3,
1, 0, 3, 0, 0, 0, 1, 3, 3, 1, 2, 0, 4, 1, 0, 0, 2, 4,
1, 0, 3, 3, 3, 0, 2, 1, 0, 4, 1, 3, 3, 0, 0, 4, 3, 0,
3, 1, 0, 1, 1, 2, 4, 4, 0, 0, 1, 3, 4, 2, 0, 4, 0, 1,
4, 0, 3, 1, 3, 3, 3, 2, 1, 3, 1, 1, 4, 2, 2, 1, 1, 0,
0, 0, 0, 1, 3, 1, 0, 3, 2, 0, 1, 1, 3, 0, 4, 0, 4, 1,
4, 3, 4, 0, 1, 3, 3, 3, 0, 3, 3, 4), c(2, 2, 4, 4, 0,
0, 3, 4, 1, 4, 1, 4, 0, 2, 3, 1, 3, 0, 1, 3, 0, 0, 0,
2, 3, 3, 0, 2, 0, 4, 1, 0, 0, 2, 3, 1, 0, 3, 0, 3, 0,
0, 0, 0, 4, 0, 4, 3, 0, 0, 3, 3, 1, 4, 1, 0, 0, 0, 3,
3, 2, 0, 0, 0, 2, 2, 0, 0, 4, 0, 0, 4, 0, 3, 1, 3, 4,
3, 2, 0, 4, 1, 0, 0, 0, 3, 0, 1, 0, 0, 1, 0, 0, 3, 1,
0, 3, 0, 0, 1, 0, 3, 0, 3, 0, 3, 1, 3, 2, 4, 1, 0, 3,
3, 2, 0, 4, 3, 4), c(2, 3, 2, 4, 0, 0, 3, 3, 0, 3, 1,
3, 0, 3, 2, 0, 4, 0, 0, 3, 0, 1, 0, 1, 2, 3, 0, 1, 0,
4, 1, 0, 0, 3, 2, 0, 0, 3, 4, 2, 0, 1, 0, 0, 4, 2, 4,
2, 0, 0, 3, 3, 0, 3, 1, 0, 1, 0, 1, 3, 3, 0, 0, 0, 3,
3, 2, 0, 4, 0, 0, 4, 0, 3, 1, 3, 3, 3, 3, 0, 3, 1, 0,
4, 0, 2, 0, 1, 0, 0, 0, 0, 1, 3, 0, 0, 4, 0, 0, 0, 0,
3, 0, 3, 0, 3, 1, 4, 3, 4, 0, 2, 3, 3, 3, 0, 3, 4, 3),
c(3, 2, 3, 4, 0, 1, 3, 4, 1, 4, 1, 3, 0, 2, 3, 0, 4,
0, 0, 4, 0, 0, 0, 0, 3, 2, 0, 3, 0, 3, 0, 0, 0, 3, 3,
0, 0, 2, 4, 3, 0, 0, 0, 0, 4, 2, 4, 3, 0, 0, 3, 2, 0,
4, 0, 0, 0, 0, 3, 4, 4, 0, 0, 0, 2, 2, 1, 0, 3, 0, 0,
3, 0, 3, 0, 2, 4, 4, 2, 0, 2, 0, 0, 4, 0, 3, 1, 1, 0,
0, 0, 0, 0, 3, 0, 0, 3, 0, 0, 0, 0, 4, 0, 3, 0, 3, 1,

```

3, 4, 3, 0, 0, 3, 2, 2, 0, 4, 4, 2), c(3, 4, 3, 4, 1,
 0, 3, 3, 0, 3, 0, 3, 0, 3, 4, 0, 3, 0, 0, 3, 0, 1, 0,
 0, 3, 3, 0, 3, 0, 4, 0, 0, 0, 3, 4, 0, 0, 3, 3, 3, 0,
 0, 0, 0, 4, 1, 3, 3, 0, 1, 4, 3, 0, 3, 0, 0, 1, 0, 3,
 4, 4, 0, 0, 0, 3, 4, 2, 0, 4, 0, 1, 4, 0, 3, 0, 3, 3,
 3, 2, 0, 3, 0, 0, 4, 0, 3, 0, 0, 0, 0, 0, 0, 3, 0,
 0, 3, 0, 0, 0, 0, 3, 0, 4, 0, 4, 1, 4, 3, 4, 0, 2, 3,
 3, 3, 0, 3, 3, 4), c(2, 2, 4, 4, 0, 0, 3, 4, 0, 4, 0,
 4, 0, 2, 3, 0, 3, 0, 0, 3, 0, 0, 0, 1, 3, 3, 0, 3, 0,
 4, 1, 0, 0, 3, 3, 0, 0, 3, 3, 3, 0, 0, 0, 0, 4, 0, 4,
 3, 0, 0, 3, 3, 0, 4, 0, 0, 1, 0, 3, 3, 2, 0, 0, 0, 2,
 2, 1, 0, 4, 0, 0, 4, 0, 3, 0, 3, 4, 3, 2, 0, 4, 0, 0,
 4, 0, 3, 0, 1, 0, 0, 0, 0, 0, 3, 0, 0, 3, 0, 0, 0, 0,
 3, 0, 3, 0, 3, 1, 3, 2, 4, 0, 1, 3, 3, 2, 1, 4, 3, 4),
 c(3, 4, 3, 4, 0, 1, 4, 3, 0, 3, 0, 3, 2, 3, 4, 0, 3,
 0, 0, 3, 0, 1, 0, 1, 3, 3, 0, 3, 0, 4, 0, 0, 0, 3, 4,
 0, 0, 3, 3, 3, 0, 0, 0, 0, 4, 0, 3, 3, 0, 0, 4, 3, 1,
 3, 0, 0, 0, 0, 3, 4, 4, 0, 0, 0, 3, 4, 0, 0, 4, 0, 1,
 4, 0, 3, 0, 3, 3, 3, 2, 1, 3, 0, 0, 3, 0, 3, 0, 0, 0,
 0, 0, 0, 0, 3, 0, 0, 3, 0, 0, 0, 0, 3, 0, 4, 0, 4, 1,
 4, 3, 4, 0, 0, 3, 3, 3, 0, 3, 3, 4), c(2, 2, 4, 4, 0,
 0, 3, 4, 1, 4, 0, 4, 0, 2, 3, 0, 3, 0, 0, 3, 0, 0, 0,
 0, 3, 3, 0, 4, 0, 4, 1, 0, 0, 2, 3, 0, 0, 3, 2, 3, 0,
 0, 0, 0, 4, 0, 4, 3, 0, 0, 3, 3, 0, 4, 0, 0, 0, 0, 4,
 3, 2, 0, 0, 0, 2, 2, 0, 0, 4, 0, 0, 4, 0, 3, 0, 3, 4,
 3, 2, 0, 4, 0, 0, 2, 0, 2, 0, 0, 0, 0, 0, 0, 0, 3, 0,
 0, 3, 0, 0, 1, 0, 3, 0, 3, 0, 3, 1, 3, 2, 4, 0, 1, 3,
 3, 2, 0, 4, 3, 4), c(2, 3, 2, 4, 0, 0, 3, 3, 0, 3, 0,
 3, 0, 3, 2, 1, 4, 1, 0, 3, 0, 0, 0, 0, 2, 3, 1, 2, 0,
 4, 0, 0, 0, 2, 2, 0, 0, 3, 2, 2, 0, 0, 0, 0, 0, 1, 4,
 2, 0, 1, 3, 3, 0, 3, 0, 0, 0, 0, 2, 3, 3, 0, 0, 0, 3,
 3, 0, 0, 4, 0, 0, 4, 0, 2, 0, 3, 3, 3, 3, 1, 3, 1, 0,
 1, 0, 3, 0, 0, 0, 0, 0, 0, 0, 3, 1, 0, 4, 0, 0, 1, 0,
 3, 0, 3, 0, 3, 1, 4, 3, 4, 0, 0, 3, 3, 3, 0, 3, 4, 3),
 c(3, 2, 3, 3, 0, 0, 3, 4, 0, 4, 0, 3, 0, 2, 3, 0, 4,
 0, 0, 4, 0, 0, 0, 0, 3, 2, 0, 1, 0, 3, 0, 0, 0, 1, 3,
 0, 0, 2, 1, 3, 0, 0, 0, 0, 0, 0, 4, 3, 0, 0, 3, 2, 1,
 4, 0, 0, 0, 0, 1, 4, 4, 0, 0, 0, 2, 2, 1, 0, 3, 0, 0,
 3, 0, 2, 0, 2, 4, 4, 2, 0, 2, 1, 0, 1, 0, 3, 0, 0, 0,
 0, 0, 0, 0, 3, 1, 0, 3, 0, 0, 0, 0, 4, 0, 3, 0, 3, 0,
 3, 4, 3, 0, 0, 3, 2, 2, 0, 4, 4, 2), c(3, 4, 3, 0, 0,
 0, 3, 3, 0, 3, 0, 3, 0, 3, 4, 0, 3, 0, 0, 3, 0, 0, 0,
 0, 3, 3, 0, 0, 0, 4, 0, 0, 0, 1, 4, 0, 0, 3, 2, 3, 0,
 0, 0, 0, 2, 0, 3, 3, 0, 0, 4, 3, 1, 3, 0, 0, 0, 0, 0,
 4, 4, 0, 0, 0, 3, 4, 0, 0, 4, 0, 0, 4, 0, 2, 0, 3, 3,

```

3, 2, 0, 3, 0, 2, 0, 0, 3, 0, 0, 0, 0, 0, 0, 3, 1,
0, 3, 0, 0, 1, 0, 3, 0, 4, 0, 4, 0, 4, 3, 4, 0, 0, 3,
3, 3, 2, 3, 3, 4), c(2, 2, 4, 4, 0, 0, 4, 4, 0, 4, 0,
4, 0, 2, 3, 0, 3, 1, 1, 3, 0, 0, 0, 1, 3, 3, 0, 2, 0,
4, 0, 0, 0, 3, 3, 0, 0, 3, 0, 3, 0, 0, 0, 4, 1, 4,
3, 2, 0, 3, 3, 1, 4, 0, 1, 1, 0, 2, 3, 2, 2, 1, 0, 2,
2, 1, 0, 4, 0, 0, 4, 0, 3, 0, 3, 4, 3, 2, 0, 4, 0, 2,
2, 0, 3, 0, 1, 0, 0, 0, 0, 0, 3, 0, 0, 3, 0, 2, 1, 0,
3, 0, 3, 0, 3, 0, 3, 2, 4, 0, 0, 3, 3, 2, 2, 4, 3, 4),
c(3, 4, 3, 0, 0, 0, 3, 3, 0, 3, 0, 3, 0, 3, 4, 1, 3,
1, 0, 3, 0, 0, 0, 0, 3, 3, 1, 2, 0, 4, 0, 0, 0, 3, 4,
1, 0, 3, 0, 3, 0, 0, 0, 0, 4, 0, 3, 3, 0, 0, 4, 3, 1,
3, 1, 1, 1, 1, 2, 4, 4, 0, 1, 0, 3, 4, 1, 0, 4, 1, 0,
4, 1, 2, 1, 3, 3, 3, 2, 0, 3, 0, 0, 2, 0, 2, 2, 1, 0,
0, 0, 0, 1, 3, 1, 0, 3, 0, 0, 0, 1, 3, 1, 4, 0, 4, 0,
4, 3, 4, 1, 2, 3, 3, 3, 1, 3, 3, 4), c(2, 2, 4, 4, 0,
0, 3, 4, 0, 4, 0, 4, 0, 2, 3, 0, 3, 0, 0, 3, 0, 0, 0,
0, 3, 3, 0, 2, 0, 4, 0, 0, 0, 3, 3, 1, 0, 3, 0, 3, 0,
0, 0, 0, 4, 1, 4, 3, 0, 0, 3, 3, 1, 4, 0, 0, 1, 0, 2,
3, 2, 0, 1, 0, 2, 2, 0, 0, 4, 1, 0, 4, 1, 3, 1, 3, 4,
3, 2, 0, 4, 0, 0, 1, 0, 3, 1, 0, 0, 0, 0, 0, 1, 3, 0,
0, 3, 1, 0, 0, 0, 3, 1, 3, 0, 3, 0, 3, 2, 4, 1, 2, 3,
3, 2, 1, 4, 3, 4), c(2, 3, 2, 4, 0, 0, 2, 3, 0, 3, 0,
3, 0, 3, 2, 0, 4, 0, 0, 3, 0, 0, 0, 1, 2, 3, 1, 2, 0,
4, 0, 0, 0, 2, 2, 1, 0, 3, 0, 2, 0, 0, 0, 0, 4, 0, 4,
2, 0, 0, 3, 3, 0, 3, 0, 0, 0, 0, 2, 3, 3, 0, 0, 0, 3,
3, 1, 0, 4, 0, 0, 4, 0, 2, 0, 3, 3, 3, 3, 0, 3, 0, 1,
2, 0, 3, 1, 0, 0, 0, 0, 0, 0, 3, 0, 0, 4, 1, 0, 0, 0,
3, 1, 3, 0, 3, 0, 4, 3, 4, 1, 0, 3, 3, 3, 1, 3, 4, 3),
c(3, 2, 3, 0, 0, 0, 4, 4, 0, 4, 0, 3, 0, 2, 3, 1, 4,
0, 0, 4, 0, 0, 0, 0, 3, 2, 1, 2, 0, 3, 0, 0, 0, 3, 3,
0, 0, 2, 0, 3, 0, 0, 0, 0, 1, 0, 4, 3, 0, 0, 3, 2, 0,
4, 0, 0, 0, 0, 2, 4, 4, 0, 0, 0, 2, 2, 0, 0, 3, 0, 0,
3, 0, 3, 0, 2, 4, 4, 2, 0, 2, 1, 1, 0, 0, 2, 0, 0, 0,
0, 0, 0, 0, 3, 0, 0, 3, 1, 0, 0, 0, 4, 1, 3, 0, 3, 0,
3, 4, 3, 0, 0, 3, 2, 2, 0, 4, 4, 2), c(3, 4, 3, 4, 0,
0, 3, 3, 0, 3, 0, 3, 0, 3, 4, 1, 3, 1, 0, 3, 0, 0, 0,
0, 3, 3, 1, 2, 0, 4, 0, 0, 0, 2, 4, 0, 0, 3, 2, 3, 0,
0, 0, 0, 1, 0, 3, 3, 0, 0, 4, 3, 0, 3, 0, 0, 0, 0, 2,
4, 4, 0, 0, 0, 3, 4, 0, 0, 4, 0, 0, 4, 1, 2, 0, 3, 3,
3, 2, 0, 3, 0, 0, 1, 0, 2, 0, 0, 0, 0, 0, 0, 1, 3, 0,
0, 3, 0, 0, 0, 0, 3, 1, 4, 0, 4, 0, 4, 3, 4, 0, 0, 3,
3, 3, 1, 3, 3, 4), c(2, 2, 4, 4, 0, 0, 4, 4, 1, 4, 0,
4, 0, 2, 3, 1, 3, 0, 0, 3, 0, 1, 0, 0, 3, 3, 1, 3, 0,
4, 0, 0, 0, 2, 3, 0, 0, 3, 2, 3, 0, 0, 0, 0, 0, 4,

```

```

3, 1, 0, 3, 3, 0, 4, 1, 0, 0, 0, 3, 3, 2, 0, 0, 0, 2,
2, 0, 0, 4, 2, 0, 4, 0, 3, 1, 3, 4, 3, 2, 0, 4, 0, 0,
1, 0, 2, 0, 1, 0, 0, 0, 0, 0, 3, 1, 0, 3, 0, 0, 0, 1,
3, 1, 3, 0, 3, 0, 3, 2, 4, 0, 0, 3, 3, 2, 0, 4, 3, 4),
c(4, 4, 4, 0, 0, 2, 4, 3, 1, 4, 0, 3, 0, 4, 3, 1, 4,
2, 0, 3, 1, 0, 0, 0, 4, 4, 0, 3, 4, 4, 1, 0, 2, 3, 4,
1, 0, 4, 3, 4, 2, 2, 2, 0, 4, 2, 4, 4, 0, 0, 4, 4, 2,
4, 2, 0, 0, 1, 3, 4, 4, 0, 2, 0, 4, 4, 0, 0, 4, 0, 0,
4, 0, 1, 1, 3, 4, 4, 4, 0, 4, 0, 1, 0, 2, 1, 0, 1, 0,
2, 1, 0, 0, 4, 1, 1, 4, 2, 0, 1, 0, 4, 1, 4, 0, 4, 2,
4, 4, 4, 0, 4, 3, 4, 4, 0, 3, 3, 4), c(4, 4, 4, 4, 0,
0, 4, 3, 2, 4, 0, 4, 2, 4, 4, 0, 4, 0, 1, 4, 0, 1, 0,
0, 3, 4, 0, 3, 4, 4, 1, 0, 2, 3, 3, 0, 0, 3, 3, 4, 1,
1, 0, 1, 2, 0, 3, 4, 2, 0, 3, 4, 1, 4, 1, 0, 0, 3, 3,
3, 4, 0, 1, 0, 4, 4, 0, 0, 3, 0, 0, 3, 0, 1, 1, 4, 4,
4, 4, 0, 4, 0, 0, 4, 0, 2, 0, 0, 2, 1, 0, 0, 0, 3, 1,
1, 4, 1, 0, 3, 2, 4, 0, 4, 0, 3, 0, 3, 4, 4, 0, 1, 3,
4, 4, 0, 3, 3, 4), c(4, 4, 4, 4, 0, 0, 4, 3, 1, 4, 0,
3, 0, 4, 3, 0, 4, 0, 1, 3, 0, 0, 0, 0, 4, 4, 0, 3, 3,
4, 1, 0, 0, 3, 4, 0, 1, 4, 3, 4, 2, 1, 0, 0, 0, 0, 4,
4, 0, 0, 4, 4, 0, 4, 0, 0, 1, 0, 3, 4, 4, 0, 1, 0, 4,
4, 0, 0, 4, 0, 0, 4, 0, 1, 2, 3, 4, 4, 4, 0, 4, 0, 0,
2, 0, 2, 0, 0, 0, 1, 0, 0, 0, 4, 1, 0, 4, 0, 0, 0, 0,
4, 0, 4, 0, 4, 0, 4, 4, 4, 0, 1, 3, 4, 4, 0, 3, 3, 4),
c(4, 4, 4, 0, 0, 0, 4, 3, 2, 4, 0, 4, 1, 4, 4, 0, 4,
1, 1, 4, 0, 1, 0, 0, 3, 4, 0, 3, 4, 4, 1, 0, 0, 3, 3,
0, 1, 3, 3, 4, 0, 1, 1, 0, 0, 0, 3, 4, 1, 0, 3, 4, 1,
4, 2, 0, 0, 0, 3, 3, 4, 0, 1, 1, 4, 4, 1, 0, 3, 1, 0,
3, 0, 1, 2, 4, 4, 4, 4, 0, 4, 1, 0, 3, 0, 1, 0, 0, 0,
1, 1, 1, 0, 3, 1, 0, 4, 0, 0, 0, 0, 4, 0, 4, 0, 3, 0,
3, 4, 4, 0, 1, 3, 4, 4, 0, 3, 3, 4), c(4, 4, 4, 1, 0,
0, 4, 3, 0, 4, 0, 3, 0, 4, 3, 0, 4, 0, 0, 3, 0, 0, 0,
0, 4, 4, 0, 3, 0, 4, 0, 0, 0, 3, 4, 0, 0, 4, 3, 4, 0,
0, 0, 0, 0, 0, 4, 4, 0, 0, 4, 4, 0, 4, 0, 0, 1, 0, 3,
4, 4, 0, 0, 0, 4, 4, 0, 0, 4, 0, 0, 4, 0, 1, 0, 3, 4,
4, 4, 0, 4, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 4, 0,
0, 4, 0, 0, 0, 0, 4, 0, 4, 0, 4, 0, 4, 4, 0, 1, 3,
4, 4, 0, 3, 3, 4), c(4, 4, 4, 0, 0, 0, 4, 3, 1, 4, 0,
4, 0, 4, 4, 0, 4, 0, 0, 4, 0, 0, 0, 0, 3, 4, 0, 3, 0,
4, 0, 0, 0, 3, 3, 0, 0, 3, 3, 4, 0, 0, 0, 0, 0, 3,
4, 0, 0, 3, 4, 0, 4, 0, 0, 0, 0, 3, 3, 4, 0, 0, 0, 4,
4, 0, 0, 3, 0, 0, 3, 0, 1, 0, 4, 4, 4, 4, 0, 4, 0, 0,
2, 0, 2, 0, 0, 0, 0, 0, 0, 0, 3, 0, 0, 4, 0, 0, 0, 0,
4, 0, 4, 0, 3, 0, 3, 4, 4, 0, 1, 3, 4, 4, 0, 3, 3, 4),
c(4, 4, 4, 2, 0, 0, 4, 3, 2, 4, 0, 3, 0, 4, 3, 0, 4,

```



```

1, 0, 3, 0, 1, 0, 0, 4, 4, 0, 4, 4, 4, 1, 1, 0, 3, 4,
0, 0, 4, 3, 4, 2, 1, 1, 0, 3, 2, 4, 4, 0, 0, 4, 4, 1,
4, 1, 0, 1, 2, 4, 4, 4, 0, 2, 0, 4, 4, 0, 0, 4, 1, 0,
4, 0, 1, 0, 3, 4, 4, 4, 0, 4, 1, 0, 3, 4, 3, 0, 0, 0,
0, 0, 1, 0, 4, 1, 0, 4, 1, 1, 0, 0, 4, 0, 4, 2, 4, 0,
4, 4, 4, 0, 1, 3, 4, 4, 0, 3, 3, 4), c(4, 4, 4, 0, 0,
1, 4, 3, 1, 4, 3, 4, 2, 4, 4, 0, 4, 0, 2, 4, 0, 1, 0,
0, 3, 4, 0, 4, 0, 4, 0, 0, 0, 3, 3, 0, 1, 3, 3, 4, 0,
1, 0, 0, 0, 2, 3, 4, 0, 0, 4, 4, 0, 4, 1, 0, 0, 1, 4,
3, 4, 0, 0, 0, 4, 4, 0, 0, 3, 1, 0, 3, 0, 2, 0, 4, 4,
4, 4, 0, 4, 0, 0, 4, 0, 2, 0, 0, 0, 1, 0, 0, 0, 3, 0,
0, 4, 1, 0, 0, 0, 4, 0, 4, 1, 3, 0, 3, 4, 4, 0, 2, 3,
4, 4, 0, 3, 3, 4), c(4, 4, 4, 4, 0, 0, 4, 3, 3, 4, 0,
3, 0, 4, 3, 0, 4, 2, 1, 3, 0, 1, 0, 1, 4, 4, 0, 4, 1,
4, 1, 0, 0, 4, 4, 0, 1, 4, 1, 4, 1, 2, 0, 0, 4, 1, 4,
4, 0, 0, 3, 4, 0, 4, 1, 0, 1, 0, 4, 4, 4, 0, 1, 0, 4,
4, 0, 0, 4, 2, 1, 4, 0, 2, 0, 3, 4, 4, 4, 0, 4, 1, 0,
2, 0, 2, 0, 0, 0, 1, 0, 4, 0, 4, 0, 0, 4, 2, 0, 0, 0,
4, 0, 4, 1, 4, 0, 4, 4, 4, 0, 0, 3, 4, 4, 0, 3, 3, 4),
c(4, 4, 4, 0, 0, 0, 4, 3, 3, 4, 0, 4, 0, 4, 4, 0, 4,
0, 0, 4, 0, 0, 0, 1, 3, 4, 0, 4, 0, 4, 1, 0, 0, 4, 3,
0, 0, 3, 3, 4, 0, 1, 0, 0, 4, 0, 3, 4, 0, 0, 4, 4, 0,
4, 1, 0, 0, 2, 4, 3, 4, 0, 0, 0, 4, 4, 0, 0, 3, 0, 0,
3, 0, 2, 0, 4, 4, 4, 4, 0, 4, 0, 0, 4, 0, 3, 0, 0, 0,
0, 0, 1, 0, 3, 0, 0, 4, 1, 0, 0, 0, 4, 0, 4, 0, 3, 0,
3, 4, 4, 0, 2, 3, 4, 4, 0, 3, 3, 4), c(4, 4, 4, 2, 0,
0, 4, 3, 2, 4, 1, 3, 2, 4, 3, 0, 4, 1, 0, 3, 0, 0, 0,
0, 4, 4, 0, 3, 1, 4, 1, 0, 0, 4, 4, 0, 0, 4, 2, 4, 0,
1, 1, 0, 4, 0, 4, 4, 0, 1, 4, 4, 1, 4, 0, 0, 0, 0, 4,
4, 4, 0, 1, 0, 4, 4, 0, 1, 4, 1, 0, 4, 1, 2, 1, 3, 4,
4, 4, 0, 4, 0, 3, 2, 0, 3, 0, 0, 0, 0, 0, 2, 0, 4, 0,
0, 4, 0, 0, 0, 0, 4, 2, 4, 1, 4, 0, 4, 4, 4, 0, 2, 3,
4, 4, 0, 3, 3, 4), c(4, 4, 4, 4, 2, 0, 3, 3, 1, 4, 1,
4, 2, 4, 4, 1, 4, 3, 0, 4, 0, 4, 1, 0, 3, 4, 1, 0, 0,
4, 1, 1, 4, 4, 3, 0, 3, 3, 3, 4, 0, 3, 1, 0, 4, 2, 3,
4, 0, 0, 4, 4, 0, 4, 2, 0, 0, 0, 3, 3, 4, 0, 0, 0, 4,
4, 0, 2, 3, 2, 2, 3, 0, 2, 0, 4, 4, 4, 3, 0, 4, 0, 1,
2, 0, 3, 0, 2, 0, 3, 2, 1, 2, 3, 0, 0, 4, 2, 0, 2, 2,
4, 0, 4, 3, 3, 0, 3, 4, 4, 0, 1, 3, 4, 4, 0, 3, 3, 4),
c(4, 4, 4, 4, 4, 2, 4, 3, 1, 4, 1, 3, 3, 4, 3, 3, 4,
2, 2, 3, 0, 2, 1, 3, 4, 4, 2, 0, 0, 4, 1, 2, 4, 3, 4,
2, 2, 4, 3, 4, 2, 2, 2, 4, 4, 2, 4, 4, 3, 1, 4, 4, 2,
4, 2, 1, 1, 0, 0, 4, 4, 0, 0, 3, 4, 4, 0, 4, 4, 3, 2,
4, 2, 2, 1, 3, 4, 4, 4, 3, 4, 2, 1, 3, 0, 3, 1, 4, 0,
2, 2, 1, 3, 4, 0, 2, 4, 2, 2, 2, 2, 4, 3, 4, 4, 4, 2,

```

```

4, 4, 4, 3, 2, 3, 4, 4, 4, 3, 3, 4), c(4, 4, 4, 1, 4,
0, 3, 3, 2, 4, 0, 4, 0, 4, 4, 2, 4, 0, 1, 4, 2, 0, 1,
2, 3, 4, 0, 3, 0, 4, 1, 0, 0, 4, 3, 1, 1, 3, 4, 4, 0,
0, 0, 0, 4, 1, 3, 4, 0, 0, 3, 4, 0, 4, 0, 0, 0, 0, 0,
3, 4, 0, 0, 0, 4, 4, 0, 2, 3, 0, 0, 3, 0, 2, 2, 4, 3,
4, 4, 0, 4, 2, 0, 4, 0, 4, 1, 1, 0, 1, 0, 0, 1, 3, 0,
0, 4, 2, 2, 0, 2, 4, 0, 4, 0, 3, 0, 3, 4, 4, 1, 3, 3,
4, 4, 4, 3, 3, 4), c(4, 4, 3, 4, 4, 2, 3, 3, 3, 4, 3,
3, 3, 4, 3, 1, 4, 1, 1, 3, 1, 1, 2, 4, 4, 3, 2, 0, 0,
4, 1, 1, 4, 2, 4, 0, 3, 3, 4, 4, 1, 2, 1, 4, 4, 2, 3,
4, 0, 0, 4, 4, 0, 4, 0, 3, 0, 0, 0, 4, 4, 0, 0, 0, 4,
4, 1, 3, 4, 1, 1, 4, 1, 2, 0, 3, 4, 3, 4, 3, 4, 2, 1,
2, 0, 3, 0, 0, 0, 2, 3, 4, 3, 4, 0, 0, 3, 1, 4, 1, 2,
4, 3, 4, 4, 4, 2, 4, 3, 4, 0, 0, 3, 4, 4, 4, 3, 3, 4),
c(4, 4, 4, 0, 4, 0, 3, 3, 2, 4, 3, 4, 2, 4, 4, 3, 4,
3, 1, 4, 4, 2, 3, 4, 3, 4, 1, 0, 1, 4, 0, 4, 1, 2, 3,
3, 3, 3, 2, 4, 2, 1, 2, 3, 2, 3, 3, 4, 2, 0, 4, 4, 2,
4, 2, 2, 3, 0, 0, 3, 4, 0, 0, 0, 4, 4, 1, 1, 3, 1, 0,
3, 1, 1, 1, 4, 4, 4, 4, 4, 4, 0, 3, 2, 3, 3, 4, 2, 2,
2, 3, 3, 3, 3, 0, 2, 4, 0, 4, 3, 2, 4, 3, 4, 3, 3, 2,
3, 4, 4, 3, 2, 3, 4, 4, 4, 3, 3, 4), c(4, 4, 4, 4, 4,
2, 4, 3, 1, 4, 2, 3, 0, 4, 3, 0, 4, 3, 1, 3, 2, 1, 1,
0, 4, 4, 0, 0, 0, 4, 0, 4, 4, 3, 4, 0, 0, 4, 3, 4, 1,
2, 2, 1, 4, 0, 4, 4, 2, 1, 3, 4, 1, 4, 1, 0, 0, 0, 0,
4, 4, 0, 0, 0, 4, 4, 1, 1, 4, 2, 1, 4, 1, 1, 0, 3, 4,
4, 4, 3, 4, 0, 3, 2, 0, 2, 1, 1, 2, 1, 3, 3, 3, 4, 0,
0, 4, 1, 4, 4, 2, 4, 1, 4, 4, 4, 1, 4, 4, 4, 0, 2, 3,
4, 4, 4, 3, 3, 4), c(4, 4, 4, 4, 4, 3, 4, 3, 3, 4, 2,
4, 2, 4, 4, 2, 4, 2, 2, 4, 3, 3, 0, 1, 3, 4, 0, 3, 4,
4, 2, 3, 2, 3, 3, 2, 2, 3, 2, 4, 3, 3, 2, 3, 4, 2, 3,
4, 0, 0, 4, 4, 2, 4, 2, 2, 2, 3, 3, 3, 4, 4, 2, 2, 4,
4, 0, 2, 3, 2, 2, 3, 1, 1, 3, 4, 4, 4, 4, 0, 4, 2, 2,
3, 2, 3, 1, 2, 3, 2, 3, 2, 2, 3, 2, 2, 4, 3, 2, 4, 2,
4, 2, 4, 3, 3, 1, 3, 4, 4, 1, 1, 3, 4, 4, 0, 3, 3, 4),
c(4, 4, 4, 4, 4, 0, 4, 3, 3, 4, 0, 3, 4, 4, 3, 3, 4,
2, 1, 3, 0, 1, 0, 0, 4, 4, 1, 3, 1, 4, 1, 3, 3, 2, 4,
3, 3, 4, 2, 4, 3, 2, 2, 1, 3, 0, 4, 4, 0, 0, 4, 4, 2,
4, 1, 2, 2, 1, 3, 4, 4, 4, 2, 0, 4, 4, 1, 0, 4, 1, 0,
4, 0, 1, 1, 3, 4, 4, 4, 2, 4, 2, 0, 4, 2, 2, 4, 1, 3,
3, 0, 2, 1, 4, 1, 3, 4, 2, 2, 0, 0, 4, 1, 4, 1, 4, 1,
4, 4, 4, 0, 2, 3, 4, 4, 4, 3, 3, 4), c(4, 4, 4, 3, 2,
3, 4, 3, 3, 4, 2, 3, 0, 4, 3, 3, 4, 1, 1, 3, 0, 1, 1,
0, 4, 4, 1, 3, 1, 4, 2, 1, 2, 1, 4, 4, 2, 4, 3, 4, 2,
2, 2, 3, 4, 0, 4, 4, 1, 0, 3, 4, 2, 4, 2, 2, 2, 1, 3,
4, 4, 4, 2, 0, 4, 4, 2, 0, 4, 0, 0, 4, 0, 1, 1, 3, 4,

```

```

4, 4, 2, 4, 2, 0, 4, 2, 3, 4, 2, 3, 2, 1, 3, 1, 4, 2,
2, 4, 2, 2, 1, 2, 4, 1, 4, 3, 4, 1, 4, 4, 4, 0, 1, 3,
4, 4, 0, 3, 3, 4), c(4, 4, 4, 4, 2, 3, 3, 3, 2, 4, 2,
4, 0, 4, 4, 2, 4, 1, 1, 4, 0, 1, 1, 1, 3, 4, 1, 3, 4,
4, 2, 2, 2, 1, 3, 3, 2, 3, 3, 4, 2, 2, 2, 3, 4, 2, 3,
4, 1, 1, 3, 4, 2, 4, 2, 2, 2, 1, 3, 3, 4, 4, 2, 1, 4,
4, 2, 1, 3, 1, 0, 3, 0, 1, 3, 4, 4, 4, 4, 2, 4, 2, 0,
4, 2, 3, 2, 2, 3, 2, 3, 2, 1, 3, 2, 2, 4, 2, 2, 3, 1,
4, 1, 4, 3, 3, 1, 3, 4, 4, 0, 1, 3, 4, 4, 0, 3, 3, 4),
c(4, 4, 4, 4, 0, 3, 3, 3, 3, 4, 0, 3, 1, 4, 3, 1, 4,
2, 0, 3, 2, 1, 0, 4, 4, 4, 1, 3, 2, 4, 2, 3, 0, 2, 4,
3, 4, 4, 4, 4, 2, 1, 2, 3, 1, 1, 4, 4, 3, 0, 4, 4, 2,
4, 1, 2, 2, 0, 3, 4, 4, 0, 1, 2, 4, 4, 0, 4, 4, 3, 3,
4, 1, 1, 1, 3, 4, 4, 4, 0, 4, 1, 4, 4, 4, 2, 2, 4, 1,
3, 3, 4, 1, 4, 3, 1, 4, 1, 4, 3, 1, 4, 0, 4, 0, 4, 1,
4, 4, 4, 2, 1, 3, 4, 4, 0, 3, 3, 4), c(4, 4, 4, 4, 0,
2, 4, 3, 2, 4, 0, 4, 0, 4, 4, 0, 4, 0, 0, 4, 4, 1, 0,
1, 3, 4, 1, 3, 2, 4, 2, 1, 0, 2, 3, 0, 1, 3, 2, 4, 2,
1, 2, 1, 1, 0, 3, 4, 4, 0, 4, 4, 1, 4, 0, 2, 2, 3, 3,
3, 4, 0, 0, 2, 4, 4, 1, 1, 3, 2, 2, 3, 1, 1, 1, 4, 4,
4, 4, 0, 4, 1, 4, 2, 4, 2, 0, 2, 1, 1, 3, 3, 1, 3, 3,
1, 4, 0, 2, 3, 1, 4, 1, 4, 1, 3, 1, 3, 4, 4, 0, 2, 3,
4, 4, 0, 3, 3, 4), c(4, 4, 4, 4, 0, 2, 4, 3, 1, 4, 4,
3, 0, 4, 3, 0, 4, 0, 0, 3, 1, 0, 0, 0, 4, 4, 1, 3, 2,
4, 3, 1, 0, 2, 4, 2, 1, 4, 2, 4, 0, 0, 2, 0, 0, 0, 4,
4, 0, 1, 3, 4, 0, 4, 0, 2, 1, 0, 3, 4, 4, 0, 0, 1, 4,
4, 1, 2, 4, 2, 1, 4, 1, 1, 1, 3, 4, 4, 4, 0, 4, 0, 2,
2, 0, 2, 1, 0, 0, 1, 1, 1, 1, 4, 2, 0, 4, 0, 2, 1, 0,
4, 0, 4, 0, 4, 1, 4, 4, 4, 0, 0, 3, 4, 4, 0, 3, 3, 4),
c(1, 1, 1, 1, 0, 0, 1, 1, 1, 1, 0, 1, 0, 1, 1, 0, 1,
0, 0, 1, 0, 0, 0, 0, 1, 1, 0, 1, 0, 1, 0, 0, 0, 1, 1,
0, 0, 1, 1, 1, 0, 0, 0, 0, 1, 0, 1, 1, 0, 0, 1, 1, 0,
1, 0, 0, 0, 0, 1, 1, 1, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0,
1, 0, 1, 0, 1, 1, 1, 1, 0, 1, 0, 0, 1, 0, 1, 0, 0, 0,
0, 0, 0, 0, 1, 0, 0, 1, 0, 0, 0, 0, 1, 0, 1, 0, 1, 0,
1, 1, 1, 0, 1, 1, 1, 1, 0, 1, 1, 1)))

```

Variables actually used in tree construction:

[1] Atr40 Atr9

Root node error: 29.731/119 = 0.24984

n= 119

CP nsplit rel error xerror xstd

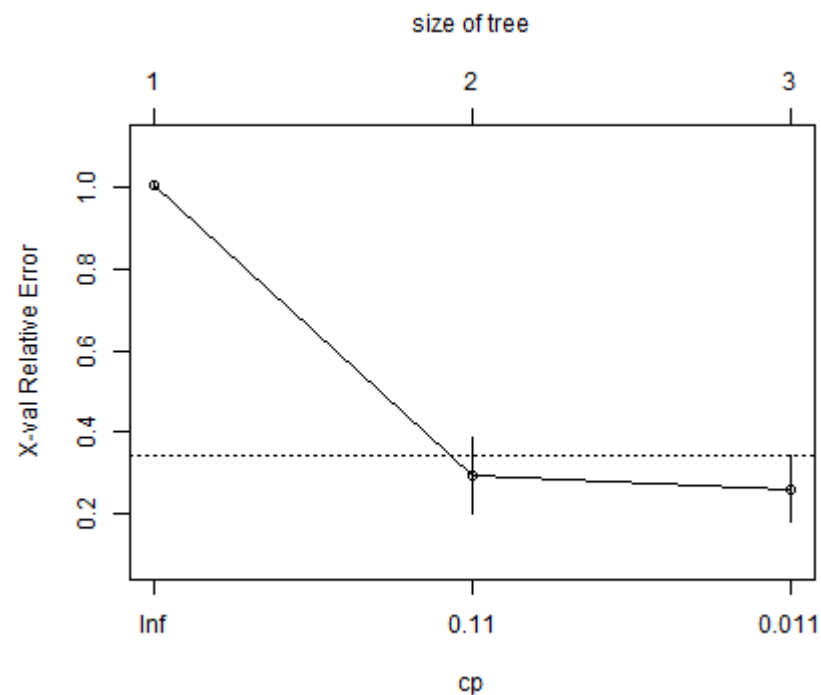
```

1 0.934866      0  1.000000 1.00932 0.0061696
2 0.012813     1  0.065134 0.29326 0.0935181
3 0.010000     2  0.052321 0.26024 0.0829647
RObject{RealSxp}
Out[102...      CP nsplit  rel error    xerror      xstd
1 0.93486590     0 1.00000000 1.0093171 0.006169639
2 0.01281327     1 0.06513410 0.2932585 0.093518134
3 0.01000000     2 0.05232083 0.2602386 0.082964739

```

Complexity Parametere (CP) would be: 0.01 for nsplit: 2

```
In [103... plotcp(treeModel)
```



```
Out[103... RObject{NilSxp}
NULL
```

```
In [104... ptreeModel = prune(treeModel,cp=0.01)
```

```
Out[104... RObject{VecSxp}
n= 119
```

```
node), split, n, deviance, yval
  * denotes terminal node
```

```
1) root 119 29.731090 0.48739500
 2) Atr9< 1.5 63 1.936508 0.03174603
   4) Atr40< 0.5 54 0.000000 0.00000000 *
   5) Atr40>=0.5 9 1.555556 0.22222220 *
 3) Atr9>=1.5 56 0.000000 1.00000000 *
```

In [105...

```
@rimport base as rbase
rbase.summary(ptreeModel)
```

Call:

```
(function (formula, data, weights, subset, na.action = na.rpart,
  method, model = FALSE, x = FALSE, y = TRUE, parms, control,
  cost, ...)
{
  Call <- match.call()
  if (is.data.frame(model)) {
    m <- model
    model <- FALSE
  }
  else {
    indx <- match(c("formula", "data", "weights", "subset"),
      names(Call), nomatch = 0)
    if (indx[1] == 0)
      stop("a 'formula' argument is required")
    temp <- Call[c(1, indx)]
    temp$na.action <- na.action
    temp[[1]] <- quote(stats::model.frame)
    m <- eval.parent(temp)
  }
  Terms <- attr(m, "terms")
  if (any(attr(Terms, "order") > 1))
    stop("Trees cannot handle interaction terms")
  Y <- model.response(m)
  wt <- model.weights(m)
  if (any(wt < 0))
    stop("negative weights not allowed")
  if (!length(wt))
    wt <- rep(1, nrow(m))
  offset <- model.offset(m)
  X <- rpart.matrix(m)
```

```

nobs <- nrow(X)
nvar <- ncol(X)
if (missing(method)) {
  method <- if (is.factor(Y) || is.character(Y))
    "class"
  else if (inherits(Y, "Surv"))
    "exp"
  else if (is.matrix(Y))
    "poisson"
  else "anova"
}
if (is.list(method)) {
  mlist <- method
  method <- "user"
  init <- if (missing(parms))
    mlist$init(Y, offset, wt = wt)
  else mlist$init(Y, offset, parms, wt)
  keep <- rpartcallback(mlist, nobs, init)
  method.int <- 4
  parms <- init$parms
}
else {
  method.int <- pmatch(method, c("anova", "poisson", "class",
    "exp"))
  if (is.na(method.int))
    stop("Invalid method")
  method <- c("anova", "poisson", "class", "exp")[method.int]
  if (method.int == 4)
    method.int <- 2
  init <- if (missing(parms))
    get(paste("rpart", method, sep = "."), envir = environment())(Y,
      offset, , wt)
  else get(paste("rpart", method, sep = "."), envir = environment())(Y,
    offset, parms, wt)
  ns <- asNamespace("rpart")
  if (!is.null(init$print))
    environment(init$print) <- ns
  if (!is.null(init$summary))
    environment(init$summary) <- ns
  if (!is.null(init$text))
    environment(init$text) <- ns
}
Y <- init$y
xlevels <- .getXlevels(Terms, m)
cats <- rep(0, ncol(X))

```

```

if (!is.null(xlevels)) {
  xlevels <- xlevels[names(xlevels) %in% colnames(X)]
  cats[match(names(xlevels), colnames(X))] <- unlist(lapply(xlevels,
    length))
}
extraArgs <- list(...)
if (length(extraArgs)) {
  controlargs <- names(formals(rpart.control))
  indx <- match(names(extraArgs), controlargs, nomatch = 0)
  if (any(indx == 0))
    stop(gettextf("Argument %s not matched", names(extraArgs)[indx ==
      0]), domain = NA)
}
controls <- rpart.control(...)
if (!missing(control))
  controls[names(control)] <- control
xval <- controls$xval
if (is.null(xval) || (length(xval) == 1 && xval == 0) ||
  method == "user") {
  xgroups <- 0
  xval <- 0
}
else if (length(xval) == 1) {
  xgroups <- sample(rep(1:xval, length.out = nobs), nobs,
    replace = FALSE)
}
else if (length(xval) == nobs) {
  xgroups <- xval
  xval <- length(unique(xgroups))
}
else {
  if (!is.null(attr(m, "na.action"))) {
    temp <- as.integer(attr(m, "na.action"))
    xval <- xval[-temp]
    if (length(xval) == nobs) {
      xgroups <- xval
      xval <- length(unique(xgroups))
    }
    else stop("Wrong length for 'xval'")
  }
  else stop("Wrong length for 'xval'")
}
if (missing(cost))
  cost <- rep(1, nvar)
else {

```

```

    if (length(cost) != nvar)
      stop("Cost vector is the wrong length")
    if (any(cost <= 0))
      stop("Cost vector must be positive")
  }
  tfun <- function(x) if (is.matrix(x))
    rep(is.ordered(x), ncol(x))
  else is.ordered(x)
  labs <- sub("^(.*)`$", "\\1", attr(Terms, "term.labels"))
  isord <- unlist(lapply(m[lab], tfun))
  storage.mode(X) <- "double"
  storage.mode(wt) <- "double"
  temp <- as.double(unlist(init$parms))
  if (!length(temp))
    temp <- 0
  rpfit <- .Call(C_rpart, ncat = as.integer(cats * !isord),
    method = as.integer(method.int), as.double(unlist(controls)),
    temp, as.integer(xval), as.integer(xgroups), as.double(t(init$y)),
    X, wt, as.integer(init$numy), as.double(cost))
  nsplit <- nrow(rpfit$split)
  ncat <- if (!is.null(rpfit$csplit))
    nrow(rpfit$csplit)
  else 0
  if (nsplit == 0)
    xval <- 0
  numcp <- ncol(rpfit$cptable)
  temp <- if (nrow(rpfit$cptable) == 3)
    c("CP", "nsplit", "rel error")
  else c("CP", "nsplit", "rel error", "xerror", "xstd")
  dimnames(rpfit$cptable) <- list(temp, 1:numcp)
  tname <- c("<leaf>", colnames(X))
  splits <- matrix(c(rpfit$split[, 2:3], rpfit$dsplit), ncol = 5,
    dimnames = list(tname[rpfit$split[, 1] + 1], c("count",
      "ncat", "improve", "index", "adj")))
  index <- rpfit$inode[, 2]
  nadd <- sum(isord[rpfit$split[, 1]])
  if (nadd > 0) {
    newc <- matrix(0, nadd, max(cats))
    cvar <- rpfit$split[, 1]
    indx <- isord[cvar]
    cdir <- splits[indx, 2]
    ccut <- floor(splits[indx, 4])
    splits[indx, 2] <- cats[cvar[indx]]
    splits[indx, 4] <- ncat + 1:nadd
    for (i in 1:nadd) {

```



```

      newc[i, 1:(cats[(cvar[indx])[i]])] <- -as.integer(cdir[i])
      newc[i, 1:ccut[i]] <- as.integer(cdir[i])
    }
    catmat <- if (ncat == 0)
      newc
    else {
      cs <- rpfit$csplit
      ncs <- ncol(cs)
      ncc <- ncol(newc)
      if (ncs < ncc)
        cs <- cbind(cs, matrix(0, nrow(cs), ncc - ncs))
      rbind(cs, newc)
    }
    ncat <- ncat + nadd
  }
  else catmat <- rpfit$csplit
  if (nsplit == 0) {
    frame <- data.frame(row.names = 1, var = "<leaf>", n = rpfit$inode[,
      5], wt = rpfit$dnode[, 3], dev = rpfit$dnode[, 1],
      yval = rpfit$dnode[, 4], complexity = rpfit$dnode[,
        2], ncompete = 0, nsurrogate = 0)
  }
  else {
    temp <- ifelse(index == 0, 1, index)
    svar <- ifelse(index == 0, 0, rpfit$isplit[temp, 1])
    frame <- data.frame(row.names = rpfit$inode[, 1], var = tname[svar +
      1], n = rpfit$inode[, 5], wt = rpfit$dnode[, 3],
      dev = rpfit$dnode[, 1], yval = rpfit$dnode[, 4],
      complexity = rpfit$dnode[, 2], ncompete = pmax(0,
        rpfit$inode[, 3] - 1), nsurrogate = rpfit$inode[,
          4])
  }
  if (method.int == 3) {
    numclass <- init$numresp - 2
    nodeprob <- rpfit$dnode[, numclass + 5]/sum(wt)
    temp <- pmax(1, init$counts)
    temp <- rpfit$dnode[, 4 + (1:numclass)] %*% diag(init$params$prior/temp)
    yprob <- temp/rowSums(temp)
    yval2 <- matrix(rpfit$dnode[, 4 + (0:numclass)], ncol = numclass +
      1)
    frame$yval2 <- cbind(yval2, yprob, nodeprob)
  }
  else if (init$numresp > 1)
    frame$yval2 <- rpfit$dnode[, -(1:3), drop = FALSE]
  if (is.null(init$summary))

```

```

      stop("Initialization routine is missing the 'summary' function")
functions <- if (is.null(init$print))
  list(summary = init$summary)
else list(summary = init$summary, print = init$print)
if (!is.null(init$text))
  functions <- c(functions, list(text = init$text))
if (method == "user")
  functions <- c(functions, mlist)
where <- rpfif$which
names(where) <- row.names(m)
ans <- list(frame = frame, where = where, call = Call, terms = Terms,
  cptable = t(rpfif$cptable), method = method, parms = init$parms,
  control = controls, functions = functions, numresp = init$numresp)
if (nsplit)
  ans$splits = splits
if (ncat > 0)
  ans$csplit <- catmat + 2
if (nsplit)
  ans$variable.importance <- importance(ans)
if (model) {
  ans$model <- m
  if (missing(y))
    y <- FALSE
}
if (y)
  ans$y <- Y
if (x) {
  ans$x <- X
  ans$wt <- wt
}
ans$ordered <- isord
if (!is.null(attr(m, "na.action")))
  ans$na.action <- attr(m, "na.action")
if (!is.null(xlevels))
  attr(ans, "xlevels") <- xlevels
if (method == "class")
  attr(ans, "ylevels") <- init$ylevels
class(ans) <- "rpart"
ans
})(formula = Class ~ Atr1 + Atr2 + Atr3 + Atr4 + Atr5 + Atr6 +
  Atr7 + Atr8 + Atr9 + Atr10 + Atr11 + Atr12 + Atr13 + Atr14 +
  Atr15 + Atr16 + Atr17 + Atr18 + Atr19 + Atr20 + Atr21 + Atr22 +
  Atr23 + Atr24 + Atr25 + Atr26 + Atr27 + Atr28 + Atr29 + Atr30 +
  Atr31 + Atr32 + Atr33 + Atr34 + Atr35 + Atr36 + Atr37 + Atr38 +
  Atr39 + Atr40 + Atr41 + Atr42 + Atr43 + Atr44 + Atr45 + Atr46 +

```

```

Atr47 + Atr48 + Atr49 + Atr50 + Atr51 + Atr52 + Atr53 + Atr54,
data = list(c(3, 4, 3, 4, 0, 1, 3, 3, 2, 3, 3, 3, 0, 3, 4,
0, 3, 0, 0, 3, 0, 0, 0, 3, 3, 3, 1, 3, 0, 4, 1, 0, 0, 2,
4, 0, 0, 3, 2, 3, 0, 0, 0, 0, 4, 2, 3, 3, 0, 0, 4, 3, 1,
3, 0, 0, 2, 0, 3, 4, 4, 4, 0, 0, 3, 4, 1, 0, 4, 0, 0, 4,
0, 2, 2, 3, 3, 3, 2, 0, 3, 0, 0, 4, 0, 3, 0, 1, 1, 0, 0,
0, 0, 3, 0, 0, 3, 0, 0, 0, 0, 3, 0, 4, 0, 4, 2, 4, 3, 4,
0, 0, 3, 3, 3, 0, 3, 3, 4), c(2, 2, 4, 4, 1, 0, 3, 4, 2,
4, 1, 4, 0, 2, 3, 0, 3, 0, 1, 3, 0, 0, 0, 0, 3, 3, 1, 2,
0, 4, 0, 1, 0, 2, 3, 0, 0, 3, 0, 3, 0, 0, 0, 0, 4, 0, 4,
3, 0, 0, 3, 3, 2, 4, 0, 0, 1, 2, 2, 3, 2, 3, 0, 0, 2, 2,
1, 0, 4, 2, 1, 4, 0, 2, 1, 3, 4, 3, 2, 0, 4, 0, 1, 4, 0,
3, 0, 0, 1, 0, 0, 0, 0, 3, 0, 0, 3, 1, 0, 0, 1, 3, 0, 3,
0, 3, 2, 3, 2, 4, 1, 0, 3, 3, 2, 0, 4, 3, 4), c(2, 3, 2,
4, 0, 1, 2, 3, 4, 3, 1, 3, 2, 3, 2, 0, 4, 0, 0, 3, 0, 0,
0, 0, 2, 3, 0, 4, 0, 4, 1, 1, 0, 2, 2, 0, 0, 3, 2, 2, 1,
0, 0, 0, 3, 1, 4, 2, 1, 0, 3, 3, 0, 3, 1, 0, 0, 2, 4, 3,
3, 4, 0, 0, 3, 3, 2, 0, 4, 0, 2, 4, 0, 1, 1, 3, 3, 3, 3,
0, 3, 0, 0, 4, 0, 3, 0, 0, 1, 0, 1, 0, 0, 3, 0, 0, 4, 0,
0, 0, 1, 3, 0, 3, 0, 3, 3, 4, 3, 4, 0, 1, 3, 3, 3, 1, 3,
4, 3), c(3, 2, 3, 3, 0, 0, 3, 4, 1, 4, 0, 3, 4, 2, 3, 0,
4, 0, 1, 4, 0, 0, 0, 0, 3, 2, 0, 3, 0, 3, 0, 0, 0, 3, 3,
0, 0, 2, 4, 3, 0, 0, 0, 0, 3, 0, 4, 3, 0, 0, 3, 2, 0, 4,
1, 0, 2, 1, 3, 4, 4, 4, 0, 0, 2, 2, 0, 0, 3, 0, 1, 3, 0,
0, 0, 2, 4, 4, 2, 0, 2, 0, 0, 4, 0, 2, 0, 0, 1, 0, 0, 0,
0, 3, 0, 0, 3, 1, 0, 0, 0, 4, 0, 3, 0, 3, 0, 3, 4, 3, 0,
0, 3, 2, 2, 0, 4, 4, 2), c(3, 4, 3, 4, 0, 0, 3, 3, 0, 3,
0, 3, 0, 3, 4, 0, 3, 0, 0, 3, 0, 0, 0, 0, 3, 3, 0, 3, 0,
4, 1, 0, 0, 2, 4, 0, 0, 3, 2, 3, 1, 0, 0, 0, 4, 0, 3, 3,
0, 0, 4, 3, 0, 3, 0, 0, 0, 0, 3, 4, 4, 0, 0, 0, 3, 4, 2,
0, 4, 0, 0, 4, 0, 0, 0, 3, 3, 3, 2, 0, 3, 0, 0, 4, 0, 1,
0, 0, 1, 0, 0, 0, 1, 3, 0, 0, 3, 0, 0, 0, 0, 3, 0, 4, 0,
4, 0, 4, 3, 4, 0, 0, 3, 3, 3, 0, 3, 3, 4), c(2, 2, 0, 2,
0, 0, 1, 0, 0, 0, 0, 0, 0, 2, 1, 0, 1, 0, 0, 1, 2, 2, 0,
0, 1, 1, 0, 2, 0, 0, 1, 1, 0, 3, 1, 2, 0, 1, 2, 1, 0, 0,
1, 0, 0, 0, 0, 1, 1, 0, 1, 1, 0, 0, 0, 0, 0, 0, 2, 1, 2,
0, 0, 0, 2, 2, 1, 0, 0, 1, 0, 0, 0, 4, 2, 1, 0, 1, 2, 1,
0, 0, 1, 0, 0, 3, 1, 1, 0, 0, 0, 0, 0, 1, 1, 0, 1, 0, 2,
0, 2, 1, 0, 1, 0, 1, 1, 1, 2, 0, 0, 2, 1, 1, 2, 0, 0, 1,
0), c(1, 0, 1, 4, 0, 0, 1, 1, 0, 1, 0, 1, 0, 1, 0, 0, 1,
0, 0, 1, 1, 0, 0, 0, 1, 1, 0, 3, 0, 0, 0, 0, 0, 2, 0, 0,
0, 1, 4, 1, 0, 0, 0, 0, 4, 0, 1, 1, 0, 0, 0, 1, 0, 1, 0,
0, 0, 0, 3, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 1,
0, 1, 1, 1, 2, 0, 1, 0, 0, 0, 0, 4, 0, 0, 0, 0, 0, 0, 0,
1, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0,

```

```

1, 1, 1, 0, 1, 1, 0), c(2, 2, 4, 4, 0, 1, 3, 4, 0, 4, 0,
4, 0, 2, 3, 0, 3, 0, 0, 3, 0, 0, 0, 3, 3, 0, 2, 0, 4,
0, 0, 0, 1, 3, 0, 0, 3, 3, 3, 0, 0, 0, 0, 2, 0, 4, 3, 0,
0, 3, 3, 0, 4, 0, 0, 0, 0, 2, 3, 2, 0, 0, 0, 2, 2, 2, 0,
4, 0, 0, 4, 0, 3, 0, 3, 4, 3, 2, 0, 4, 0, 2, 4, 0, 3, 0,
0, 1, 0, 1, 0, 0, 3, 1, 0, 3, 0, 0, 1, 0, 3, 0, 3, 0, 3,
0, 3, 2, 4, 1, 0, 3, 3, 2, 0, 4, 3, 4), c(2, 3, 2, 4, 0,
0, 3, 3, 0, 3, 0, 3, 0, 3, 2, 0, 4, 0, 0, 3, 0, 0, 0, 0,
2, 3, 0, 2, 0, 4, 0, 0, 0, 3, 2, 0, 0, 3, 4, 2, 0, 0, 0,
0, 4, 0, 4, 2, 0, 0, 3, 3, 0, 3, 0, 0, 0, 0, 2, 3, 3, 0,
0, 0, 3, 3, 1, 0, 4, 0, 0, 4, 0, 3, 0, 3, 3, 3, 3, 0, 3,
0, 0, 4, 0, 2, 0, 0, 1, 0, 1, 0, 0, 3, 1, 0, 4, 0, 0, 0,
0, 3, 0, 3, 0, 3, 0, 4, 3, 4, 0, 0, 3, 3, 3, 0, 3, 4, 3),
  c(3, 2, 3, 3, 0, 0, 3, 4, 0, 4, 0, 3, 2, 2, 3, 0, 4,
    2, 1, 4, 0, 1, 0, 0, 3, 2, 1, 3, 0, 3, 1, 0, 1, 2, 3,
    0, 0, 2, 3, 3, 0, 1, 0, 0, 4, 2, 4, 3, 0, 0, 3, 2, 0,
    4, 2, 1, 1, 2, 3, 4, 4, 0, 1, 0, 2, 2, 2, 0, 3, 0, 0,
    3, 0, 3, 2, 2, 4, 4, 2, 0, 2, 0, 0, 4, 0, 2, 1, 1, 1,
    0, 0, 0, 0, 3, 0, 0, 3, 1, 0, 0, 0, 4, 0, 3, 0, 3, 1,
    3, 4, 3, 0, 1, 3, 2, 2, 0, 4, 4, 2), c(3, 4, 3, 4, 0,
    1, 4, 3, 1, 3, 1, 3, 0, 3, 4, 0, 3, 0, 0, 3, 0, 0, 0,
    0, 3, 3, 0, 4, 0, 4, 0, 0, 0, 1, 4, 0, 0, 3, 2, 3, 0,
    0, 0, 0, 3, 1, 3, 3, 0, 0, 4, 3, 0, 3, 1, 0, 1, 1, 4,
    4, 4, 0, 0, 0, 3, 4, 1, 0, 4, 0, 0, 4, 0, 3, 0, 3, 3,
    3, 2, 1, 3, 0, 1, 4, 0, 2, 0, 1, 1, 0, 0, 0, 0, 3, 0,
    0, 3, 1, 0, 0, 0, 3, 0, 4, 0, 4, 1, 4, 3, 4, 0, 0, 3,
    3, 3, 0, 3, 3, 4), c(2, 2, 4, 4, 0, 0, 3, 4, 0, 4, 0,
    4, 0, 2, 3, 0, 3, 1, 0, 3, 0, 0, 0, 1, 3, 3, 0, 3, 0,
    4, 0, 0, 1, 2, 3, 0, 0, 3, 3, 3, 1, 1, 0, 0, 4, 0, 4,
    3, 0, 1, 3, 3, 0, 4, 2, 0, 1, 1, 3, 3, 2, 0, 0, 0, 2,
    2, 1, 0, 4, 0, 1, 4, 0, 3, 1, 3, 4, 3, 2, 1, 4, 1, 0,
    3, 0, 2, 1, 0, 0, 0, 0, 1, 0, 3, 0, 0, 3, 2, 0, 1, 2,
    3, 0, 3, 0, 3, 1, 3, 2, 4, 0, 2, 3, 3, 2, 0, 4, 3, 4),
  c(3, 4, 3, 4, 0, 1, 3, 3, 1, 3, 0, 3, 0, 3, 4, 1, 3,
    1, 0, 3, 0, 0, 0, 1, 3, 3, 1, 2, 0, 4, 1, 0, 0, 2, 4,
    1, 0, 3, 3, 3, 0, 2, 1, 0, 4, 1, 3, 3, 0, 0, 4, 3, 0,
    3, 1, 0, 1, 1, 2, 4, 4, 0, 0, 1, 3, 4, 2, 0, 4, 0, 1,
    4, 0, 3, 1, 3, 3, 3, 2, 1, 3, 1, 1, 4, 2, 2, 1, 1, 0,
    0, 0, 0, 1, 3, 1, 0, 3, 2, 0, 1, 1, 3, 0, 4, 0, 4, 1,
    4, 3, 4, 0, 1, 3, 3, 3, 0, 3, 3, 4), c(2, 2, 4, 4, 0,
    0, 3, 4, 1, 4, 1, 4, 0, 2, 3, 1, 3, 0, 1, 3, 0, 0, 0,
    2, 3, 3, 0, 2, 0, 4, 1, 0, 0, 2, 3, 1, 0, 3, 0, 3, 0,
    0, 0, 0, 4, 0, 4, 3, 0, 0, 3, 3, 1, 4, 1, 0, 0, 0, 3,
    3, 2, 0, 0, 0, 2, 2, 0, 0, 4, 0, 0, 4, 0, 3, 1, 3, 4,
    3, 2, 0, 4, 1, 0, 0, 0, 3, 0, 1, 0, 0, 1, 0, 0, 3, 1,

```

```

0, 3, 0, 0, 1, 0, 3, 0, 3, 0, 3, 1, 3, 2, 4, 1, 0, 3,
3, 2, 0, 4, 3, 4), c(2, 3, 2, 4, 0, 0, 3, 3, 0, 3, 1,
3, 0, 3, 2, 0, 4, 0, 0, 3, 0, 1, 0, 1, 2, 3, 0, 1, 0,
4, 1, 0, 0, 3, 2, 0, 0, 3, 4, 2, 0, 1, 0, 0, 4, 2, 4,
2, 0, 0, 3, 3, 0, 3, 1, 0, 1, 0, 1, 3, 3, 0, 0, 0, 3,
3, 2, 0, 4, 0, 0, 4, 0, 3, 1, 3, 3, 3, 3, 0, 3, 1, 0,
4, 0, 2, 0, 1, 0, 0, 0, 0, 1, 3, 0, 0, 4, 0, 0, 0, 0,
3, 0, 3, 0, 3, 1, 4, 3, 4, 0, 2, 3, 3, 3, 0, 3, 4, 3),
c(3, 2, 3, 4, 0, 1, 3, 4, 1, 4, 1, 3, 0, 2, 3, 0, 4,
0, 0, 4, 0, 0, 0, 0, 3, 2, 0, 3, 0, 3, 0, 0, 0, 3, 3,
0, 0, 2, 4, 3, 0, 0, 0, 0, 4, 2, 4, 3, 0, 0, 3, 2, 0,
4, 0, 0, 0, 0, 3, 4, 4, 0, 0, 0, 2, 2, 1, 0, 3, 0, 0,
3, 0, 3, 0, 2, 4, 4, 2, 0, 2, 0, 0, 4, 0, 3, 1, 1, 0,
0, 0, 0, 0, 3, 0, 0, 3, 0, 0, 0, 0, 4, 0, 3, 0, 3, 1,
3, 4, 3, 0, 0, 3, 2, 2, 0, 4, 4, 2), c(3, 4, 3, 4, 1,
0, 3, 3, 0, 3, 0, 3, 0, 3, 4, 0, 3, 0, 0, 3, 0, 1, 0,
0, 3, 3, 0, 3, 0, 4, 0, 0, 0, 3, 4, 0, 0, 3, 3, 3, 0,
0, 0, 0, 4, 1, 3, 3, 0, 1, 4, 3, 0, 3, 0, 0, 1, 0, 3,
4, 4, 0, 0, 0, 3, 4, 2, 0, 4, 0, 1, 4, 0, 3, 0, 3, 3,
3, 2, 0, 3, 0, 0, 4, 0, 3, 0, 0, 0, 0, 0, 0, 0, 3, 0,
0, 3, 0, 0, 0, 0, 3, 0, 4, 0, 4, 1, 4, 3, 4, 0, 2, 3,
3, 3, 0, 3, 3, 4), c(2, 2, 4, 4, 0, 0, 3, 4, 0, 4, 0,
4, 0, 2, 3, 0, 3, 0, 0, 3, 0, 0, 0, 1, 3, 3, 0, 3, 0,
4, 1, 0, 0, 3, 3, 0, 0, 3, 3, 3, 0, 0, 0, 0, 4, 0, 4,
3, 0, 0, 3, 3, 0, 4, 0, 0, 1, 0, 3, 3, 2, 0, 0, 0, 2,
2, 1, 0, 4, 0, 0, 4, 0, 3, 0, 3, 4, 3, 2, 0, 4, 0, 0,
4, 0, 3, 0, 1, 0, 0, 0, 0, 0, 3, 0, 0, 3, 0, 0, 0, 0,
3, 0, 3, 0, 3, 1, 3, 2, 4, 0, 1, 3, 3, 2, 1, 4, 3, 4),
c(3, 4, 3, 4, 0, 1, 4, 3, 0, 3, 0, 3, 2, 3, 4, 0, 3,
0, 0, 3, 0, 1, 0, 1, 3, 3, 0, 3, 0, 4, 0, 0, 0, 3, 4,
0, 0, 3, 3, 3, 0, 0, 0, 0, 4, 0, 3, 3, 0, 0, 4, 3, 1,
3, 0, 0, 0, 0, 3, 4, 4, 0, 0, 0, 3, 4, 0, 0, 4, 0, 1,
4, 0, 3, 0, 3, 3, 3, 2, 1, 3, 0, 0, 3, 0, 3, 0, 0, 0,
0, 0, 0, 0, 3, 0, 0, 3, 0, 0, 0, 0, 3, 0, 4, 0, 4, 1,
4, 3, 4, 0, 0, 3, 3, 3, 0, 3, 3, 4), c(2, 2, 4, 4, 0,
0, 3, 4, 1, 4, 0, 4, 0, 2, 3, 0, 3, 0, 0, 3, 0, 0, 0,
0, 3, 3, 0, 4, 0, 4, 1, 0, 0, 2, 3, 0, 0, 3, 2, 3, 0,
0, 0, 0, 4, 0, 4, 3, 0, 0, 3, 3, 0, 4, 0, 0, 0, 0, 4,
3, 2, 0, 0, 0, 2, 2, 0, 0, 4, 0, 0, 4, 0, 3, 0, 3, 4,
3, 2, 0, 4, 0, 0, 2, 0, 2, 0, 0, 0, 0, 0, 0, 0, 3, 0,
0, 3, 0, 0, 1, 0, 3, 0, 3, 0, 3, 1, 3, 2, 4, 0, 1, 3,
3, 2, 0, 4, 3, 4), c(2, 3, 2, 4, 0, 0, 3, 3, 0, 3, 0,
3, 0, 3, 2, 1, 4, 1, 0, 3, 0, 0, 0, 0, 2, 3, 1, 2, 0,
4, 0, 0, 0, 2, 2, 0, 0, 3, 2, 2, 0, 0, 0, 0, 0, 1, 4,
2, 0, 1, 3, 3, 0, 3, 0, 0, 0, 0, 2, 3, 3, 0, 0, 0, 3,

```

```

3, 0, 0, 4, 0, 0, 4, 0, 2, 0, 3, 3, 3, 3, 1, 3, 1, 0,
1, 0, 3, 0, 0, 0, 0, 0, 0, 0, 3, 1, 0, 4, 0, 0, 1, 0,
3, 0, 3, 0, 3, 1, 4, 3, 4, 0, 0, 3, 3, 3, 0, 3, 4, 3),
c(3, 2, 3, 3, 0, 0, 3, 4, 0, 4, 0, 3, 0, 2, 3, 0, 4,
0, 0, 4, 0, 0, 0, 0, 3, 2, 0, 1, 0, 3, 0, 0, 0, 1, 3,
0, 0, 2, 1, 3, 0, 0, 0, 0, 0, 0, 4, 3, 0, 0, 3, 2, 1,
4, 0, 0, 0, 0, 1, 4, 4, 0, 0, 0, 2, 2, 1, 0, 3, 0, 0,
3, 0, 2, 0, 2, 4, 4, 2, 0, 2, 1, 0, 1, 0, 3, 0, 0, 0,
0, 0, 0, 0, 3, 1, 0, 3, 0, 0, 0, 0, 4, 0, 3, 0, 3, 0,
3, 4, 3, 0, 0, 3, 2, 2, 0, 4, 4, 2), c(3, 4, 3, 0, 0,
0, 3, 3, 0, 3, 0, 3, 0, 3, 4, 0, 3, 0, 0, 3, 0, 0, 0,
0, 3, 3, 0, 0, 0, 4, 0, 0, 0, 1, 4, 0, 0, 3, 2, 3, 0,
0, 0, 0, 2, 0, 3, 3, 0, 0, 4, 3, 1, 3, 0, 0, 0, 0, 0,
4, 4, 0, 0, 0, 3, 4, 0, 0, 4, 0, 0, 4, 0, 2, 0, 3, 3,
3, 2, 0, 3, 0, 2, 0, 0, 3, 0, 0, 0, 0, 0, 0, 0, 3, 1,
0, 3, 0, 0, 1, 0, 3, 0, 4, 0, 4, 0, 4, 3, 4, 0, 0, 3,
3, 3, 2, 3, 3, 4), c(2, 2, 4, 4, 0, 0, 4, 4, 0, 4, 0,
4, 0, 2, 3, 0, 3, 1, 1, 3, 0, 0, 0, 1, 3, 3, 0, 2, 0,
4, 0, 0, 0, 3, 3, 0, 0, 3, 0, 3, 0, 0, 0, 0, 4, 1, 4,
3, 2, 0, 3, 3, 1, 4, 0, 1, 1, 0, 2, 3, 2, 2, 1, 0, 2,
2, 1, 0, 4, 0, 0, 4, 0, 3, 0, 3, 4, 3, 2, 0, 4, 0, 2,
2, 0, 3, 0, 1, 0, 0, 0, 0, 0, 3, 0, 0, 3, 0, 2, 1, 0,
3, 0, 3, 0, 3, 0, 3, 2, 4, 0, 0, 3, 3, 2, 2, 4, 3, 4),
c(3, 4, 3, 0, 0, 0, 3, 3, 0, 3, 0, 3, 0, 3, 4, 1, 3,
1, 0, 3, 0, 0, 0, 0, 3, 3, 1, 2, 0, 4, 0, 0, 0, 3, 4,
1, 0, 3, 0, 3, 0, 0, 0, 0, 4, 0, 3, 3, 0, 0, 4, 3, 1,
3, 1, 1, 1, 1, 2, 4, 4, 0, 1, 0, 3, 4, 1, 0, 4, 1, 0,
4, 1, 2, 1, 3, 3, 3, 2, 0, 3, 0, 0, 2, 0, 2, 2, 1, 0,
0, 0, 0, 1, 3, 1, 0, 3, 0, 0, 0, 1, 3, 1, 4, 0, 4, 0,
4, 3, 4, 1, 2, 3, 3, 3, 1, 3, 3, 4), c(2, 2, 4, 4, 0,
0, 3, 4, 0, 4, 0, 4, 0, 2, 3, 0, 3, 0, 0, 3, 0, 0, 0,
0, 3, 3, 0, 2, 0, 4, 0, 0, 0, 3, 3, 1, 0, 3, 0, 3, 0,
0, 0, 0, 4, 1, 4, 3, 0, 0, 3, 3, 1, 4, 0, 0, 1, 0, 2,
3, 2, 0, 1, 0, 2, 2, 0, 0, 4, 1, 0, 4, 1, 3, 1, 3, 4,
3, 2, 0, 4, 0, 0, 1, 0, 3, 1, 0, 0, 0, 0, 0, 1, 3, 0,
0, 3, 1, 0, 0, 0, 3, 1, 3, 0, 3, 0, 3, 2, 4, 1, 2, 3,
3, 2, 1, 4, 3, 4), c(2, 3, 2, 4, 0, 0, 2, 3, 0, 3, 0,
3, 0, 3, 2, 0, 4, 0, 0, 3, 0, 0, 0, 1, 2, 3, 1, 2, 0,
4, 0, 0, 0, 2, 2, 1, 0, 3, 0, 2, 0, 0, 0, 0, 4, 0, 4,
2, 0, 0, 3, 3, 0, 3, 0, 0, 0, 0, 2, 3, 3, 0, 0, 0, 3,
3, 1, 0, 4, 0, 0, 4, 0, 2, 0, 3, 3, 3, 3, 0, 3, 0, 1,
2, 0, 3, 1, 0, 0, 0, 0, 0, 0, 3, 0, 0, 4, 1, 0, 0, 0,
3, 1, 3, 0, 3, 0, 4, 3, 4, 1, 0, 3, 3, 3, 1, 3, 4, 3),
c(3, 2, 3, 0, 0, 0, 4, 4, 0, 4, 0, 3, 0, 2, 3, 1, 4,
0, 0, 4, 0, 0, 0, 0, 3, 2, 1, 2, 0, 3, 0, 0, 0, 3, 3,

```

```

0, 0, 2, 0, 3, 0, 0, 0, 0, 1, 0, 4, 3, 0, 0, 3, 2, 0,
4, 0, 0, 0, 0, 2, 4, 4, 0, 0, 0, 2, 2, 0, 0, 3, 0, 0,
3, 0, 3, 0, 2, 4, 4, 2, 0, 2, 1, 1, 0, 0, 2, 0, 0, 0,
0, 0, 0, 0, 3, 0, 0, 3, 1, 0, 0, 0, 4, 1, 3, 0, 3, 0,
3, 4, 3, 0, 0, 3, 2, 2, 0, 4, 4, 2), c(3, 4, 3, 4, 0,
0, 3, 3, 0, 3, 0, 3, 0, 3, 4, 1, 3, 1, 0, 3, 0, 0, 0,
0, 3, 3, 1, 2, 0, 4, 0, 0, 0, 2, 4, 0, 0, 3, 2, 3, 0,
0, 0, 0, 1, 0, 3, 3, 0, 0, 4, 3, 0, 3, 0, 0, 0, 0, 2,
4, 4, 0, 0, 0, 3, 4, 0, 0, 4, 0, 0, 4, 1, 2, 0, 3, 3,
3, 2, 0, 3, 0, 0, 1, 0, 2, 0, 0, 0, 0, 0, 0, 1, 3, 0,
0, 3, 0, 0, 0, 0, 3, 1, 4, 0, 4, 0, 4, 3, 4, 0, 0, 3,
3, 3, 1, 3, 3, 4), c(2, 2, 4, 4, 0, 0, 4, 4, 1, 4, 0,
4, 0, 2, 3, 1, 3, 0, 0, 3, 0, 1, 0, 0, 3, 3, 1, 3, 0,
4, 0, 0, 0, 2, 3, 0, 0, 3, 2, 3, 0, 0, 0, 0, 0, 0, 4,
3, 1, 0, 3, 3, 0, 4, 1, 0, 0, 0, 3, 3, 2, 0, 0, 0, 2,
2, 0, 0, 4, 2, 0, 4, 0, 3, 1, 3, 4, 3, 2, 0, 4, 0, 0,
1, 0, 2, 0, 1, 0, 0, 0, 0, 0, 3, 1, 0, 3, 0, 0, 0, 1,
3, 1, 3, 0, 3, 0, 3, 2, 4, 0, 0, 3, 3, 2, 0, 4, 3, 4),
c(4, 4, 4, 0, 0, 2, 4, 3, 1, 4, 0, 3, 0, 4, 3, 1, 4,
2, 0, 3, 1, 0, 0, 0, 4, 4, 0, 3, 4, 4, 1, 0, 2, 3, 4,
1, 0, 4, 3, 4, 2, 2, 2, 0, 4, 2, 4, 4, 0, 0, 4, 4, 2,
4, 2, 0, 0, 1, 3, 4, 4, 0, 2, 0, 4, 4, 0, 0, 4, 0, 0,
4, 0, 1, 1, 3, 4, 4, 4, 0, 4, 0, 1, 0, 2, 1, 0, 1, 0,
2, 1, 0, 0, 4, 1, 1, 4, 2, 0, 1, 0, 4, 1, 4, 0, 4, 2,
4, 4, 4, 0, 4, 3, 4, 4, 0, 3, 3, 4), c(4, 4, 4, 4, 0,
0, 4, 3, 2, 4, 0, 4, 2, 4, 4, 0, 4, 0, 1, 4, 0, 1, 0,
0, 3, 4, 0, 3, 4, 4, 1, 0, 2, 3, 3, 0, 0, 3, 3, 4, 1,
1, 0, 1, 2, 0, 3, 4, 2, 0, 3, 4, 1, 4, 1, 0, 0, 3, 3,
3, 4, 0, 1, 0, 4, 4, 0, 0, 3, 0, 0, 3, 0, 1, 1, 4, 4,
4, 4, 0, 4, 0, 0, 4, 0, 2, 0, 0, 2, 1, 0, 0, 0, 3, 1,
1, 4, 1, 0, 3, 2, 4, 0, 4, 0, 3, 0, 3, 4, 4, 0, 1, 3,
4, 4, 0, 3, 3, 4), c(4, 4, 4, 4, 0, 0, 4, 3, 1, 4, 0,
3, 0, 4, 3, 0, 4, 0, 1, 3, 0, 0, 0, 0, 4, 4, 0, 3, 3,
4, 1, 0, 0, 3, 4, 0, 1, 4, 3, 4, 2, 1, 0, 0, 0, 0, 4,
4, 0, 0, 4, 4, 0, 4, 0, 0, 1, 0, 3, 4, 4, 0, 1, 0, 4,
4, 0, 0, 4, 0, 0, 4, 0, 1, 2, 3, 4, 4, 4, 0, 4, 0, 0,
2, 0, 2, 0, 0, 0, 1, 0, 0, 0, 4, 1, 0, 4, 0, 0, 0, 0,
4, 0, 4, 0, 4, 0, 4, 4, 4, 0, 1, 3, 4, 4, 0, 3, 3, 4),
c(4, 4, 4, 0, 0, 0, 4, 3, 2, 4, 0, 4, 1, 4, 4, 0, 4,
1, 1, 4, 0, 1, 0, 0, 3, 4, 0, 3, 4, 4, 1, 0, 0, 3, 3,
0, 1, 3, 3, 4, 0, 1, 1, 0, 0, 0, 3, 4, 1, 0, 3, 4, 1,
4, 2, 0, 0, 0, 3, 3, 4, 0, 1, 1, 4, 4, 1, 0, 3, 1, 0,
3, 0, 1, 2, 4, 4, 4, 4, 0, 4, 1, 0, 3, 0, 1, 0, 0, 0,
1, 1, 1, 0, 3, 1, 0, 4, 0, 0, 0, 0, 4, 0, 4, 0, 3, 0,
3, 4, 4, 0, 1, 3, 4, 4, 0, 3, 3, 4), c(4, 4, 4, 1, 0,

```

0, 4, 3, 0, 4, 0, 3, 0, 4, 3, 0, 4, 0, 0, 3, 0, 0, 0,
 0, 4, 4, 0, 3, 0, 4, 0, 0, 0, 3, 4, 0, 0, 4, 3, 4, 0,
 0, 0, 0, 0, 0, 4, 4, 0, 0, 4, 4, 0, 4, 0, 0, 1, 0, 3,
 4, 4, 0, 0, 0, 4, 4, 0, 0, 4, 0, 0, 4, 0, 1, 0, 3, 4,
 4, 4, 0, 4, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 4, 0,
 0, 4, 0, 0, 0, 0, 4, 0, 4, 0, 4, 0, 4, 4, 4, 0, 1, 3,
 4, 4, 0, 3, 3, 4), c(4, 4, 4, 0, 0, 0, 4, 3, 1, 4, 0,
 4, 0, 4, 4, 0, 4, 0, 0, 4, 0, 0, 0, 0, 3, 4, 0, 3, 0,
 4, 0, 0, 0, 3, 3, 0, 0, 3, 3, 4, 0, 0, 0, 0, 0, 0, 3,
 4, 0, 0, 3, 4, 0, 4, 0, 0, 0, 0, 3, 3, 4, 0, 0, 0, 4,
 4, 0, 0, 3, 0, 0, 3, 0, 1, 0, 4, 4, 4, 4, 0, 4, 0, 0,
 2, 0, 2, 0, 0, 0, 0, 0, 0, 0, 3, 0, 0, 4, 0, 0, 0, 0,
 4, 0, 4, 0, 3, 0, 3, 4, 4, 0, 1, 3, 4, 4, 0, 3, 3, 4),
 c(4, 4, 4, 2, 0, 0, 4, 3, 2, 4, 0, 3, 0, 4, 3, 0, 4,
 1, 0, 3, 0, 1, 0, 0, 4, 4, 0, 4, 4, 4, 1, 1, 0, 3, 4,
 0, 0, 4, 3, 4, 2, 1, 1, 0, 3, 2, 4, 4, 0, 0, 4, 4, 1,
 4, 1, 0, 1, 2, 4, 4, 4, 0, 2, 0, 4, 4, 0, 0, 4, 1, 0,
 4, 0, 1, 0, 3, 4, 4, 4, 0, 4, 1, 0, 3, 4, 3, 0, 0, 0,
 0, 0, 1, 0, 4, 1, 0, 4, 1, 1, 0, 0, 4, 0, 4, 2, 4, 0,
 4, 4, 4, 0, 1, 3, 4, 4, 0, 3, 3, 4), c(4, 4, 4, 0, 0,
 1, 4, 3, 1, 4, 3, 4, 2, 4, 4, 0, 4, 0, 2, 4, 0, 1, 0,
 0, 3, 4, 0, 4, 0, 4, 0, 0, 0, 3, 3, 0, 1, 3, 3, 4, 0,
 1, 0, 0, 0, 2, 3, 4, 0, 0, 4, 4, 0, 4, 1, 0, 0, 1, 4,
 3, 4, 0, 0, 0, 4, 4, 0, 0, 3, 1, 0, 3, 0, 2, 0, 4, 4,
 4, 4, 0, 4, 0, 0, 4, 0, 2, 0, 0, 0, 1, 0, 0, 0, 3, 0,
 0, 4, 1, 0, 0, 0, 4, 0, 4, 1, 3, 0, 3, 4, 4, 0, 2, 3,
 4, 4, 0, 3, 3, 4), c(4, 4, 4, 4, 0, 0, 4, 3, 3, 4, 0,
 3, 0, 4, 3, 0, 4, 2, 1, 3, 0, 1, 0, 1, 4, 4, 0, 4, 1,
 4, 1, 0, 0, 4, 4, 0, 1, 4, 1, 4, 1, 2, 0, 0, 4, 1, 4,
 4, 0, 0, 3, 4, 0, 4, 1, 0, 1, 0, 4, 4, 4, 0, 1, 0, 4,
 4, 0, 0, 4, 2, 1, 4, 0, 2, 0, 3, 4, 4, 4, 0, 4, 1, 0,
 2, 0, 2, 0, 0, 0, 1, 0, 4, 0, 4, 0, 0, 4, 2, 0, 0, 0,
 4, 0, 4, 1, 4, 0, 4, 4, 4, 0, 0, 3, 4, 4, 0, 3, 3, 4),
 c(4, 4, 4, 0, 0, 0, 4, 3, 3, 4, 0, 4, 0, 4, 4, 0, 4,
 0, 0, 4, 0, 0, 0, 1, 3, 4, 0, 4, 0, 4, 1, 0, 0, 4, 3,
 0, 0, 3, 3, 4, 0, 1, 0, 0, 4, 0, 3, 4, 0, 0, 4, 4, 0,
 4, 1, 0, 0, 2, 4, 3, 4, 0, 0, 0, 4, 4, 0, 0, 3, 0, 0,
 3, 0, 2, 0, 4, 4, 4, 4, 0, 4, 0, 0, 4, 0, 3, 0, 0, 0,
 0, 0, 1, 0, 3, 0, 0, 4, 1, 0, 0, 0, 4, 0, 4, 0, 3, 0,
 3, 4, 4, 0, 2, 3, 4, 4, 0, 3, 3, 4), c(4, 4, 4, 2, 0,
 0, 4, 3, 2, 4, 1, 3, 2, 4, 3, 0, 4, 1, 0, 3, 0, 0, 0,
 0, 4, 4, 0, 3, 1, 4, 1, 0, 0, 4, 4, 0, 0, 4, 2, 4, 0,
 1, 1, 0, 4, 0, 4, 4, 0, 1, 4, 4, 1, 4, 0, 0, 0, 0, 4,
 4, 4, 0, 1, 0, 4, 4, 0, 1, 4, 1, 0, 4, 1, 2, 1, 3, 4,
 4, 4, 0, 4, 0, 3, 2, 0, 3, 0, 0, 0, 0, 0, 2, 0, 4, 0,


```

0, 4, 0, 0, 0, 0, 4, 2, 4, 1, 4, 0, 4, 4, 4, 0, 2, 3,
4, 4, 0, 3, 3, 4), c(4, 4, 4, 4, 2, 0, 3, 3, 1, 4, 1,
4, 2, 4, 4, 1, 4, 3, 0, 4, 0, 4, 1, 0, 3, 4, 1, 0, 0,
4, 1, 1, 4, 4, 3, 0, 3, 3, 3, 4, 0, 3, 1, 0, 4, 2, 3,
4, 0, 0, 4, 4, 0, 4, 2, 0, 0, 0, 3, 3, 4, 0, 0, 0, 4,
4, 0, 2, 3, 2, 2, 3, 0, 2, 0, 4, 4, 4, 3, 0, 4, 0, 1,
2, 0, 3, 0, 2, 0, 3, 2, 1, 2, 3, 0, 0, 4, 2, 0, 2, 2,
4, 0, 4, 3, 3, 0, 3, 4, 4, 0, 1, 3, 4, 4, 0, 3, 3, 4),
c(4, 4, 4, 4, 4, 2, 4, 3, 1, 4, 1, 3, 3, 4, 3, 3, 4,
2, 2, 3, 0, 2, 1, 3, 4, 4, 2, 0, 0, 4, 1, 2, 4, 3, 4,
2, 2, 4, 3, 4, 2, 2, 2, 4, 4, 2, 4, 4, 3, 1, 4, 4, 2,
4, 2, 1, 1, 0, 0, 4, 4, 0, 0, 3, 4, 4, 0, 4, 4, 3, 2,
4, 2, 2, 1, 3, 4, 4, 4, 3, 4, 2, 1, 3, 0, 3, 1, 4, 0,
2, 2, 1, 3, 4, 0, 2, 4, 2, 2, 2, 2, 4, 3, 4, 4, 4, 2,
4, 4, 4, 3, 2, 3, 4, 4, 4, 3, 3, 4), c(4, 4, 4, 1, 4,
0, 3, 3, 2, 4, 0, 4, 0, 4, 4, 2, 4, 0, 1, 4, 2, 0, 1,
2, 3, 4, 0, 3, 0, 4, 1, 0, 0, 4, 3, 1, 1, 3, 4, 4, 0,
0, 0, 0, 4, 1, 3, 4, 0, 0, 3, 4, 0, 4, 0, 0, 0, 0, 0,
3, 4, 0, 0, 0, 4, 4, 0, 2, 3, 0, 0, 3, 0, 2, 2, 4, 3,
4, 4, 0, 4, 2, 0, 4, 0, 4, 1, 1, 0, 1, 0, 0, 1, 3, 0,
0, 4, 2, 2, 0, 2, 4, 0, 4, 0, 3, 0, 3, 4, 4, 1, 3, 3,
4, 4, 4, 3, 3, 4), c(4, 4, 3, 4, 4, 2, 3, 3, 3, 4, 3,
3, 3, 4, 3, 1, 4, 1, 1, 3, 1, 1, 2, 4, 4, 3, 2, 0, 0,
4, 1, 1, 4, 2, 4, 0, 3, 3, 4, 4, 1, 2, 1, 4, 4, 2, 3,
4, 0, 0, 4, 4, 0, 4, 0, 3, 0, 0, 0, 4, 4, 0, 0, 0, 4,
4, 1, 3, 4, 1, 1, 4, 1, 2, 0, 3, 4, 3, 4, 3, 4, 2, 1,
2, 0, 3, 0, 0, 0, 2, 3, 4, 3, 4, 0, 0, 3, 1, 4, 1, 2,
4, 3, 4, 4, 4, 2, 4, 3, 4, 0, 0, 3, 4, 4, 4, 3, 3, 4),
c(4, 4, 4, 0, 4, 0, 3, 3, 2, 4, 3, 4, 2, 4, 4, 3, 4,
3, 1, 4, 4, 2, 3, 4, 3, 4, 1, 0, 1, 4, 0, 4, 1, 2, 3,
3, 3, 3, 2, 4, 2, 1, 2, 3, 2, 3, 3, 4, 2, 0, 4, 4, 2,
4, 2, 2, 3, 0, 0, 3, 4, 0, 0, 0, 4, 4, 1, 1, 3, 1, 0,
3, 1, 1, 1, 4, 4, 4, 4, 4, 4, 0, 3, 2, 3, 3, 4, 2, 2,
2, 3, 3, 3, 3, 0, 2, 4, 0, 4, 3, 2, 4, 3, 4, 3, 3, 2,
3, 4, 4, 3, 2, 3, 4, 4, 4, 3, 3, 4), c(4, 4, 4, 4, 4,
2, 4, 3, 1, 4, 2, 3, 0, 4, 3, 0, 4, 3, 1, 3, 2, 1, 1,
0, 4, 4, 0, 0, 0, 4, 0, 4, 4, 3, 4, 0, 0, 4, 3, 4, 1,
2, 2, 1, 4, 0, 4, 4, 2, 1, 3, 4, 1, 4, 1, 0, 0, 0, 0,
4, 4, 0, 0, 0, 4, 4, 1, 1, 4, 2, 1, 4, 1, 1, 0, 3, 4,
4, 4, 3, 4, 0, 3, 2, 0, 2, 1, 1, 2, 1, 3, 3, 3, 4, 0,
0, 4, 1, 4, 4, 2, 4, 1, 4, 4, 4, 1, 4, 4, 4, 0, 2, 3,
4, 4, 4, 3, 3, 4), c(4, 4, 4, 4, 4, 3, 4, 3, 3, 4, 2,
4, 2, 4, 4, 2, 4, 2, 2, 4, 3, 3, 0, 1, 3, 4, 0, 3, 4,
4, 2, 3, 2, 3, 3, 2, 2, 3, 2, 4, 3, 3, 2, 3, 4, 2, 3,
4, 0, 0, 4, 4, 2, 4, 2, 2, 2, 3, 3, 3, 4, 4, 2, 2, 4,

```

```

4, 0, 2, 3, 2, 2, 3, 1, 1, 3, 4, 4, 4, 4, 0, 4, 2, 2,
3, 2, 3, 1, 2, 3, 2, 3, 2, 2, 3, 2, 2, 4, 3, 2, 4, 2,
4, 2, 4, 3, 3, 1, 3, 4, 4, 1, 1, 3, 4, 4, 0, 3, 3, 4),
c(4, 4, 4, 4, 4, 0, 4, 3, 3, 4, 0, 3, 4, 4, 3, 3, 4,
2, 1, 3, 0, 1, 0, 0, 4, 4, 1, 3, 1, 4, 1, 3, 3, 2, 4,
3, 3, 4, 2, 4, 3, 2, 2, 1, 3, 0, 4, 4, 0, 0, 4, 4, 2,
4, 1, 2, 2, 1, 3, 4, 4, 4, 2, 0, 4, 4, 1, 0, 4, 1, 0,
4, 0, 1, 1, 3, 4, 4, 4, 2, 4, 2, 0, 4, 2, 2, 4, 1, 3,
3, 0, 2, 1, 4, 1, 3, 4, 2, 2, 0, 0, 4, 1, 4, 1, 4, 1,
4, 4, 4, 0, 2, 3, 4, 4, 4, 3, 3, 4), c(4, 4, 4, 3, 2,
3, 4, 3, 3, 4, 2, 3, 0, 4, 3, 3, 4, 1, 1, 3, 0, 1, 1,
0, 4, 4, 1, 3, 1, 4, 2, 1, 2, 1, 4, 4, 2, 4, 3, 4, 2,
2, 2, 3, 4, 0, 4, 4, 1, 0, 3, 4, 2, 4, 2, 2, 2, 1, 3,
4, 4, 4, 2, 0, 4, 4, 2, 0, 4, 0, 0, 4, 0, 1, 1, 3, 4,
4, 4, 2, 4, 2, 0, 4, 2, 3, 4, 2, 3, 2, 1, 3, 1, 4, 2,
2, 4, 2, 2, 1, 2, 4, 1, 4, 3, 4, 1, 4, 4, 4, 0, 1, 3,
4, 4, 0, 3, 3, 4), c(4, 4, 4, 4, 2, 3, 3, 3, 2, 4, 2,
4, 0, 4, 4, 2, 4, 1, 1, 4, 0, 1, 1, 1, 3, 4, 1, 3, 4,
4, 2, 2, 2, 1, 3, 3, 2, 3, 3, 4, 2, 2, 2, 3, 4, 2, 3,
4, 1, 1, 3, 4, 2, 4, 2, 2, 2, 1, 3, 3, 4, 4, 2, 1, 4,
4, 2, 1, 3, 1, 0, 3, 0, 1, 3, 4, 4, 4, 4, 2, 4, 2, 0,
4, 2, 3, 2, 2, 3, 2, 3, 2, 1, 3, 2, 2, 4, 2, 2, 3, 1,
4, 1, 4, 3, 3, 1, 3, 4, 4, 0, 1, 3, 4, 4, 0, 3, 3, 4),
c(4, 4, 4, 4, 0, 3, 3, 3, 3, 4, 0, 3, 1, 4, 3, 1, 4,
2, 0, 3, 2, 1, 0, 4, 4, 4, 1, 3, 2, 4, 2, 3, 0, 2, 4,
3, 4, 4, 4, 4, 2, 1, 2, 3, 1, 1, 4, 4, 3, 0, 4, 4, 2,
4, 1, 2, 2, 0, 3, 4, 4, 0, 1, 2, 4, 4, 0, 4, 4, 3, 3,
4, 1, 1, 1, 3, 4, 4, 4, 0, 4, 1, 4, 4, 4, 2, 2, 4, 1,
3, 3, 4, 1, 4, 3, 1, 4, 1, 4, 3, 1, 4, 0, 4, 0, 4, 1,
4, 4, 4, 2, 1, 3, 4, 4, 0, 3, 3, 4), c(4, 4, 4, 4, 0,
2, 4, 3, 2, 4, 0, 4, 0, 4, 4, 0, 4, 0, 0, 4, 4, 1, 0,
1, 3, 4, 1, 3, 2, 4, 2, 1, 0, 2, 3, 0, 1, 3, 2, 4, 2,
1, 2, 1, 1, 0, 3, 4, 4, 0, 4, 4, 1, 4, 0, 2, 2, 3, 3,
3, 4, 0, 0, 2, 4, 4, 1, 1, 3, 2, 2, 3, 1, 1, 1, 4, 4,
4, 4, 0, 4, 1, 4, 2, 4, 2, 0, 2, 1, 1, 3, 3, 1, 3, 3,
1, 4, 0, 2, 3, 1, 4, 1, 4, 1, 3, 1, 3, 4, 4, 0, 2, 3,
4, 4, 0, 3, 3, 4), c(4, 4, 4, 4, 0, 2, 4, 3, 1, 4, 4,
3, 0, 4, 3, 0, 4, 0, 0, 3, 1, 0, 0, 0, 4, 4, 1, 3, 2,
4, 3, 1, 0, 2, 4, 2, 1, 4, 2, 4, 0, 0, 2, 0, 0, 0, 4,
4, 0, 1, 3, 4, 0, 4, 0, 2, 1, 0, 3, 4, 4, 0, 0, 1, 4,
4, 1, 2, 4, 2, 1, 4, 1, 1, 1, 3, 4, 4, 4, 0, 4, 0, 2,
2, 0, 2, 1, 0, 0, 1, 1, 1, 1, 4, 2, 0, 4, 0, 2, 1, 0,
4, 0, 4, 0, 4, 1, 4, 4, 4, 0, 0, 3, 4, 4, 0, 3, 3, 4),
c(1, 1, 1, 1, 0, 0, 1, 1, 1, 1, 0, 1, 0, 1, 1, 0, 1,
0, 0, 1, 0, 0, 0, 0, 1, 1, 0, 1, 0, 1, 0, 0, 1, 1,

```

```

0, 0, 1, 1, 1, 0, 0, 0, 0, 1, 0, 1, 1, 0, 0, 1, 1, 0,
1, 0, 0, 0, 0, 1, 1, 1, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0,
1, 0, 1, 0, 1, 1, 1, 1, 0, 1, 0, 0, 1, 0, 1, 0, 0, 0,
0, 0, 0, 0, 1, 0, 0, 1, 0, 0, 0, 0, 1, 0, 1, 0, 1, 0,
1, 1, 1, 0, 1, 1, 1, 1, 0, 1, 1, 1)))
n= 119

```

	CP	nsplit	rel error	xerror	xstd
1	0.93486590	0	1.00000000	1.0093171	0.006169639
2	0.01281327	1	0.06513410	0.2932585	0.093518134
3	0.01000000	2	0.05232083	0.2602386	0.082964739

Variable importance

Atr20	Atr18	Atr9	Atr11	Atr16	Atr17
17	17	17	16	16	16

```
Node number 1: 119 observations,      complexity param=0.9348659
  mean=0.487395, MSE=0.2498411
  left son=2 (63 obs) right son=3 (56 obs)
  Primary splits:
```

```
Atr9 < 1.5 to the left, improve=0.9348659, (0 missing)
Atr18 < 1.5 to the left, improve=0.9348659, (0 missing)
Atr36 < 0.5 to the left, improve=0.9348659, (0 missing)
Atr20 < 1.5 to the left, improve=0.9348659, (0 missing)
Atr17 < 1.5 to the left, improve=0.9338617, (0 missing)
```

Surrogate splits:

```
Atr18 < 1.5 to the left, agree=1.000, adj=1.000, (0 split)
Atr20 < 1.5 to the left, agree=1.000, adj=1.000, (0 split)
Atr11 < 1.5 to the left, agree=0.992, adj=0.982, (0 split)
Atr16 < 1.5 to the left, agree=0.992, adj=0.982, (0 split)
Atr17 < 2.5 to the left, agree=0.992, adj=0.982, (0 split)
```

```
Node number 2: 63 observations,      complexity param=0.01281327
  mean=0.03174603, MSE=0.03073822
  left son=4 (54 obs) right son=5 (9 obs)
```

Primary splits:

```
Atr40 < 0.5 to the left, improve=0.19672130, (0 missing)
Atr33 < 0.5 to the left, improve=0.12610340, (0 missing)
Atr44 < 1.5 to the left, improve=0.12610340, (0 missing)
Atr38 < 0.5 to the left, improve=0.09631148, (0 missing)
Atr41 < 0.5 to the left, improve=0.07049180, (0 missing)
```

Surrogate splits:

Atr12 < 1.5 to the left, agree=0.889, adj=0.222, (0 split)
 Atr36 < 0.5 to the left, agree=0.889, adj=0.222, (0 split)
 Atr39 < 1.5 to the left, agree=0.889, adj=0.222, (0 split)

Atr20 < 0.5 to the left, agree=0.873, adj=0.111, (0 split)

Node number 3: 56 observations
mean=1, MSE=0

Node number 4: 54 observations
mean=0, MSE=0

Node number 5: 9 observations
mean=0.2222222, MSE=0.1728395

Out[105... RObject{VecSxp}
n= 119

node), split, n, deviance, yval
* denotes terminal node

1) root 119 29.731090 0.48739500
2) Atr9< 1.5 63 1.936508 0.03174603
4) Atr40< 0.5 54 0.000000 0.00000000 *

5) Atr40>=0.5 9 1.555556 0.22222220 *
3) Atr9>=1.5 56 0.000000 1.00000000 *

Variable importance

Atr20 Atr18 Atr9 Atr11 Atr16 Atr17

17 17 17 16 16 16

(b) Fitting Random Forest

In [106... @rlibrary(randomForest)
rfmodel=randomForest(m, data = traindf)
rfmodel

Warning: RCall.jl: Warning in randomForest.default(m, y, ...) :
The response has five or fewer unique values. Are you sure you want to do regression?
@ RCall C:\Users\ASUS\julia\packages\RCall\6kphM\src\io.jl:172

Out[106... RObject{VecSxp}

Call:

```

randomForest(formula = Class ~ Atr1 + Atr2 + Atr3 + Atr4 + Atr5 +      Atr6 + Atr7 + Atr8 + Atr9 + Atr10 + Atr11 + Atr12
+ Atr13 +      Atr14 + Atr15 + Atr16 + Atr17 + Atr18 + Atr19 + Atr20 + Atr21 +      Atr22 + Atr23 + Atr24 + Atr25 + Atr26
+ Atr27 + Atr28 + Atr29 +      Atr30 + Atr31 + Atr32 + Atr33 + Atr34 + Atr35 + Atr36 + Atr37 +      Atr38 + Atr39 + Atr40
+ Atr41 + Atr42 + Atr43 + Atr44 + Atr45 +      Atr46 + Atr47 + Atr48 + Atr49 + Atr50 + Atr51 + Atr52 + Atr53 +      Atr5
4, data = structure(list(Atr1 = c(3L, 4L, 3L, 4L, 0L, 1L, 3L, 3L, 2L, 3L, 3L, 3L, 0L, 3L, 4L, 0L, 3L, 0L, 0L, 3L, 0L, 0
L, 0L, 3L, 3L, 3L, 1L, 3L, 0L, 4L, 1L, 0L, 0L, 2L, 4L, 0L, 0L, 3L, 2L, 3L, 0L, 0L, 0L, 0L, 4L, 2L, 3L, 3L, 0L, 0L, 4L, 3
L, 1L, 3L, 0L, 0L, 2L, 0L, 3L, 4L, 4L, 4L, 0L, 0L, 3L, 4L, 1L, 0L, 4L, 0L, 0L, 4L, 0L, 2L, 2L, 3L, 3L, 3L, 2L, 0L, 3L,
0L, 0L, 4L, 0L, 3L, 0L, 1L, 1L, 0L, 0L, 0L, 0L, 3L, 0L, 0L, 3L, 0L, 0L, 0L, 3L, 0L, 4L, 0L, 4L, 2L, 4L, 3L, 4L, 0L,
0L, 3L, 3L, 3L, 0L, 3L, 3L, 4L), Atr2 = c(2L, 2L, 4L, 4L, 1L, 0L, 3L, 4L, 2L, 4L, 1L, 4L, 0L, 2L, 3L, 0L, 3L, 0L, 1L, 3
L, 0L, 0L, 0L, 0L, 3L, 3L, 1L, 2L, 0L, 4L, 0L, 1L, 0L, 2L, 3L, 0L, 0L, 3L, 0L, 3L, 0L, 0L, 0L, 0L, 4L, 0L, 4L, 3L, 0L,
0L, 3L, 3L, 2L, 4L, 0L, 0L, 1L, 2L, 2L, 3L, 2L, 3L, 0L, 0L, 2L, 2L, 1L, 0L, 4L, 2L, 1L, 4L, 0L, 2L, 1L, 3L, 4L, 3L, 2L,
0L, 4L, 0L, 1L, 4L, 0L, 3L, 0L, 0L, 1L, 0L, 0L, 0L, 0L, 3L, 0L, 0L, 3L, 1L, 0L, 0L, 1L, 3L, 0L, 3L, 0L, 3L, 2L, 3L, 2L,
4L, 1L, 0L, 3L, 3L, 2L, 0L, 4L, 3L, 4L), Atr3 = c(2L, 3L, 2L, 4L, 0L, 1L, 2L, 3L, 4L, 3L, 1L, 3L, 2L, 3L, 2L, 0L, 4L, 0
L, 0L, 3L, 0L, 0L, 0L, 0L, 2L, 3L, 0L, 4L, 0L, 4L, 1L, 1L, 0L, 2L, 2L, 0L, 0L, 3L, 2L, 2L, 1L, 0L, 0L, 0L, 3L, 1L, 4L, 2
L, 1L, 0L, 3L, 3L, 0L, 3L, 1L, 0L, 0L, 2L, 4L, 3L, 3L, 4L, 0L, 0L, 3L, 3L, 2L, 0L, 4L, 0L, 2L, 4L, 0L, 1L, 1L, 3L, 3L,
3L, 3L, 0L, 3L, 0L, 0L, 4L, 0L, 3L, 0L, 0L, 1L, 0L, 1L, 0L, 0L, 3L, 0L, 0L, 4L, 0L, 0L, 0L, 1L, 3L, 0L, 3L, 0L, 3L, 3L,
4L, 3L, 4L, 0L, 1L, 3L, 3L, 3L, 1L, 3L, 4L, 3L), Atr4 = c(3L, 2L, 3L, 3L, 0L, 0L, 3L, 4L, 1L, 4L, 0L, 3L, 4L, 2L, 3L, 0
L, 4L, 0L, 1L, 4L, 0L, 0L, 0L, 0L, 3L, 2L, 0L, 3L, 0L, 3L, 0L, 0L, 3L, 3L, 0L, 0L, 0L, 3L, 3L, 0L, 0L, 0L, 3L, 3L, 0L,
0L, 4L, 3L, 0L, 0L, 3L, 2L, 0L, 4L, 1L, 0L, 2L, 1L, 3L, 4L, 4L, 4L, 0L, 0L, 2L, 2L, 0L, 0L, 3L, 0L, 1L, 3L, 0L, 0L, 0L,
2L, 4L, 4L, 2L, 0L, 2L, 0L, 0L, 4L, 0L, 2L, 0L, 0L, 1L, 0L, 0L, 0L, 0L, 3L, 0L, 0L, 3L, 1L, 0L, 0L, 0L, 4L, 0L, 3L, 0L,
3L, 0L, 3L, 0L, 3L, 4L, 3L, 0L, 0L, 3L, 2L, 2L, 0L, 4L, 4L, 2L ), Atr5 = c(3L, 4L, 3L, 4L, 0L, 0L, 3L, 3L, 0L, 3L, 0L, 3L,
0L, 3L, 0L, 3L, 0L, 0L, 3L, 0L, 0L, 0L, 0L, 3L, 3L, 0L, 3L, 0L, 4L, 1L, 0L, 0L, 2L, 4L, 0L, 0L, 3L, 2L, 3L, 1L, 0L, 0L, 0
L, 4L, 0L, 3L, 3L, 0L, 0L, 4L, 3L, 0L, 3L, 0L, 0L, 0L, 0L, 3L, 4L, 4L, 0L, 0L, 0L, 3L, 4L, 2L, 0L, 4L, 0L, 0L, 4L, 0L,
0L, 0L, 3L, 3L, 3L, 2L, 0L, 3L, 0L, 0L, 4L, 0L, 1L, 0L, 0L, 1L, 0L, 0L, 0L, 1L, 3L, 0L, 0L, 3L, 0L, 0L, 0L, 0L, 3L, 0L,
4L, 0L, 4L, 0L, 4L, 3L, 4L, 0L, 0L, 3L, 3L, 3L, 0L, 3L, 3L, 4L), Atr6 = c(2L, 2L, 0L, 2L, 0L, 0L, 1L, 0L, 0L, 0L, 0L, 0L,
0L, 2L, 1L, 0L, 1L, 0L, 0L, 1L, 2L, 2L, 0L, 0L, 1L, 1L, 0L, 2L, 0L, 0L, 1L, 1L, 0L, 3L, 1L, 2L, 0L, 1L, 2L, 1L, 0L,
0L, 1L, 0L, 0L, 0L, 0L, 1L, 1L, 0L, 1L, 1L, 0L, 0L, 0L, 0L, 2L, 1L, 2L, 0L, 0L, 0L, 0L, 0L, 0L, 1L, 0L, 0L, 0L, 0L,
0L, 0L, 0L, 0L, 1L, 0L, 1L, 1L, 1L, 2L, 0L, 1L, 0L, 0L, 0L, 0L, 4L, 0L, 0L, 0L, 0L, 0L, 0L, 0L, 0L, 1L, 0L, 0L, 1L, 0L,
0L, 0L, 0L, 1L, 0L, 0L, 0L, 0L, 0L, 0L, 1L, 0L, 0L, 0L, 1L, 1L, 1L, 0L, 1L, 1L, 0L), Atr7 = c(1L, 0L, 1L, 4L, 0L, 0L, 1L, 1L, 0L, 1
L, 0L, 1L, 0L, 1L, 0L, 0L, 1L, 0L, 0L, 1L, 1L, 0L, 0L, 0L, 1L, 1L, 0L, 3L, 0L, 0L, 0L, 0L, 0L, 2L, 0L, 0L, 0L, 1L, 4L, 1
L, 0L, 0L, 0L, 0L, 0L, 4L, 0L, 1L, 1L, 0L, 0L, 0L, 1L, 0L, 1L, 0L, 0L, 0L, 0L, 3L, 0L, 0L, 0L, 0L, 0L, 1L, 0L, 0L, 0L, 0L,
0L, 0L, 0L, 0L, 1L, 0L, 1L, 1L, 1L, 2L, 0L, 1L, 0L, 0L, 0L, 0L, 4L, 0L, 0L, 0L, 0L, 0L, 0L, 0L, 0L, 1L, 0L, 0L, 1L, 0L, 0L,
0L, 0L, 1L, 0L, 0L, 0L, 0L, 0L, 0L, 1L, 0L, 0L, 0L, 1L, 1L, 1L, 0L, 1L, 1L, 0L), Atr8 = c(2L, 2L, 4L, 4L, 0L, 1L, 3L, 4
L, 0L, 4L, 0L, 4L, 0L, 2L, 3L, 0L, 3L, 0L, 0L, 0L, 0L, 0L, 0L, 3L, 3L, 0L, 2L, 0L, 4L, 0L, 0L, 0L, 1L, 3L, 0L, 0L,
3L, 3L, 3L, 0L, 0L, 0L, 0L, 2L, 0L, 4L, 3L, 0L, 0L, 3L, 3L, 0L, 4L, 0L, 0L, 0L, 0L, 2L, 3L, 2L, 0L, 0L, 0L, 2L, 2L,
0L, 4L, 0L, 0L, 4L, 0L, 3L, 0L, 3L, 4L, 3L, 2L, 0L, 4L, 0L, 2L, 4L, 0L, 3L, 0L, 0L, 1L, 0L, 1L, 0L, 0L, 3L, 1L, 0L, 3L,
0L, 0L, 1L, 0L, 3L, 0L, 3L, 0L, 3L, 0L, 3L, 2L, 4L, 1L, 0L, 3L, 3L, 2L, 0L, 4L, 3L, 4L), Atr9 = c(2L, 3L, 2L, 4L, 0L, 0
L, 3L, 3L, 0L, 3L, 0L, 3L, 0L, 3L, 2L, 0L, 4L, 0L, 0L, 3L, 0L, 0L, 0L, 0L, 2L, 3L, 0L, 2L, 0L, 4L, 0L, 0L, 0L, 3L, 2L,
0L, 0L, 3L, 4L, 2L, 0L, 0L, 0L, 0L, 4L, 0L, 4L, 2L, 0L, 0L, 3L, 3L, 0L, 3L, 0L, 0L, 0L, 2L, 3L, 3L, 0L, 0L, 0L, 3L,
3L, 1L, 0L, 4L, 0L, 0L, 4L, 0L, 3L, 0L, 3L, 3L, 3L, 3L, 0L, 3L, 0L, 0L, 4L, 0L, 2L, 0L, 0L, 1L, 0L, 1L, 0L, 0L, 3L, 1L,
0L, 4L, 0L, 0L, 0L, 0L, 3L, 0L, 3L, 0L, 3L, 0L, 4L, 3L, 4L, 0L, 0L, 3L, 3L, 3L, 0L, 3L, 4L, 3L ), Atr10 = c(3L, 2L, 3L,
3L, 0L, 0L, 3L, 4L, 0L, 4L, 0L, 3L, 2L, 2L, 3L, 0L, 4L, 2L, 1L, 4L, 0L, 1L, 0L, 0L, 3L, 2L, 1L, 3L, 0L, 3L, 1L, 0L, 1L,
2L, 3L, 0L, 0L, 2L, 3L, 3L, 0L, 1L, 0L, 0L, 4L, 2L, 4L, 3L, 0L, 0L, 3L, 2L, 0L, 4L, 2L, 1L, 1L, 2L, 3L, 4L, 4L, 0L, 1L,
0L, 2L, 2L, 2L, 0L, 3L, 0L, 0L, 3L, 0L, 3L, 2L, 2L, 4L, 4L, 2L, 0L, 2L, 0L, 0L, 4L, 0L, 2L, 1L, 1L, 1L, 0L, 0L, 0L, 0L,

```

```

3L, 0L, 0L, 3L, 1L, 0L, 0L, 0L, 4L, 0L, 3L, 0L, 3L, 1L, 3L, 4L, 3L, 0L, 1L, 3L, 2L, 2L, 0L, 4L, 4L, 2L), Atr11 = c(3L, 4
L, 3L, 4L, 0L, 1L, 4L, 3L, 1L, 3L, 1L, 3L, 0L, 3L, 4L, 0L, 3L, 0L, 0L, 3L, 0L, 0L, 0L, 3L, 3L, 0L, 4L, 0L, 4L, 0L,
0L, 0L, 1L, 4L, 0L, 0L, 3L, 2L, 3L, 0L, 0L, 0L, 0L, 3L, 1L, 3L, 3L, 0L, 0L, 4L, 3L, 0L, 3L, 1L, 0L, 1L, 1L, 4L, 4L, 4L,
0L, 0L, 0L, 3L, 4L, 1L, 0L, 4L, 0L, 0L, 4L, 0L, 3L, 0L, 3L, 3L, 2L, 1L, 3L, 0L, 1L, 4L, 0L, 2L, 0L, 1L, 1L, 0L, 0L,
0L, 0L, 3L, 0L, 0L, 3L, 1L, 0L, 0L, 0L, 3L, 0L, 4L, 0L, 4L, 1L, 4L, 3L, 4L, 0L, 0L, 3L, 3L, 3L, 0L, 3L, 3L, 4L), Atr12
= c(2L, 2L, 4L, 4L, 0L, 0L, 3L, 4L, 0L, 4L, 0L, 4L, 0L, 2L, 3L, 0L, 3L, 1L, 0L, 3L, 0L, 0L, 0L, 1L, 3L, 3L, 0L, 3L, 0L,
4L, 0L, 0L, 1L, 2L, 3L, 0L, 0L, 3L, 3L, 3L, 1L, 1L, 0L, 0L, 4L, 0L, 4L, 3L, 0L, 1L, 3L, 3L, 0L, 4L, 2L, 0L, 1L, 1L, 3L,
3L, 2L, 0L, 0L, 0L, 2L, 2L, 1L, 0L, 4L, 0L, 1L, 4L, 0L, 3L, 1L, 3L, 4L, 3L, 2L, 1L, 4L, 1L, 0L, 3L, 0L, 2L, 1L, 0L, 0L,
0L, 0L, 1L, 0L, 3L, 0L, 0L, 3L, 2L, 0L, 1L, 2L, 3L, 0L, 3L, 0L, 3L, 1L, 3L, 2L, 4L, 0L, 2L, 3L, 3L, 2L, 0L, 4L, 3L, 4L),
Atr13 = c(3L, 4L, 3L, 4L, 0L, 1L, 3L, 3L, 1L, 3L, 0L, 3L, 0L, 3L, 4L, 1L, 3L, 1L, 0L, 3L, 0L, 0L, 0L, 1L, 3L, 3L, 1
L, 2L, 0L, 4L, 1L, 0L, 0L, 2L, 4L, 1L, 0L, 3L, 3L, 3L, 0L, 2L, 1L, 0L, 4L, 1L, 3L, 3L, 0L, 0L, 4L, 3L, 0L, 3L,
1L, 0L, 1L, 1L, 2L, 4L, 4L, 0L, 0L, 1L, 3L, 4L, 2L, 0L, 4L, 0L, 1L, 4L, 0L, 3L, 1L, 3L, 3L, 3L, 2L, 1L, 3L, 1L,
1L, 4L, 2L, 2L, 1L, 1L, 0L, 0L, 0L, 0L, 1L, 3L, 1L, 0L, 3L, 2L, 0L, 1L, 1L, 3L, 0L, 4L, 0L, 4L, 1L, 4L, 3L, 4L,
0L, 1L, 3L, 3L, 3L, 0L, 3L, 3L, 4L), Atr14 = c(2L, 2L, 4L, 4L, 0L, 0L, 3L, 4L, 1L, 4L, 1L, 4L, 0L, 0L, 0L, 0L,
3L, 0L, 1L, 3L, 0L, 0L, 0L, 2L, 3L, 3L, 0L, 2L, 0L, 4L, 1L, 0L, 0L, 2L, 3L, 1L, 0L, 3L, 0L, 3L, 0L, 0L, 0L, 0L,
4L, 0L, 4L, 3L, 0L, 0L, 3L, 3L, 1L, 4L, 1L, 0L, 0L, 0L, 3L, 3L, 2L, 0L, 0L, 0L, 2L, 2L, 0L, 0L, 4L, 0L, 0L, 4L,
0L, 3L, 1L, 3L, 4L, 3L, 2L, 0L, 4L, 1L, 0L, 0L, 0L, 3L, 0L, 1L, 0L, 0L, 1L, 0L, 0L, 3L, 1L, 0L, 3L, 0L, 0L, 1L,
0L, 3L, 0L, 3L, 0L, 3L, 1L, 3L, 2L, 4L, 1L, 0L, 3L, 3L, 2L, 0L, 4L, 3L, 4L), Atr15 = c(2L, 3L, 2L, 4L, 0L, 0L, 3L, 3
L, 0L, 3L, 1L, 3L, 0L, 3L, 2L, 0L, 4L, 0L, 0L, 3L, 0L, 1L, 0L, 1L, 2L, 3L, 0L, 1L, 0L, 4L, 1L, 0L, 0L, 3L, 2L,
0L, 0L, 3L, 4L, 2L, 0L, 1L, 0L, 0L, 4L, 2L, 4L, 2L, 0L, 0L, 3L, 3L, 0L, 3L, 1L, 0L, 1L, 0L, 1L, 3L, 3L, 0L, 0L,
0L, 3L, 3L, 2L, 0L, 4L, 0L, 0L, 4L, 0L, 3L, 1L, 3L, 3L, 3L, 3L, 0L, 3L, 1L, 0L, 4L, 0L, 2L, 0L, 1L, 0L, 0L, 0L,
0L, 1L, 3L, 0L, 0L, 4L, 0L, 0L, 0L, 0L, 3L, 0L, 3L, 0L, 3L, 1L, 4L, 3L, 4L, 0L, 2L, 3L, 3L, 3L, 0L, 3L, 4L, 3
L), Atr16 = c(3L, 2L, 3L, 4L, 0L, 1L, 3L, 4L, 1L, 4L, 1L, 3L, 0L, 2L, 3L, 0L, 4L, 0L, 0L, 4L, 0L, 0L, 0L, 0L, 3
L, 2L, 0L, 3L, 0L, 3L, 0L, 0L, 0L, 3L, 3L, 0L, 0L, 2L, 4L, 3L, 0L, 0L, 0L, 3L, 2L, 0L, 4L, 0L, 0L, 0L, 3L,
0L, 4L, 0L, 0L, 0L, 0L, 3L, 4L, 4L, 0L, 0L, 0L, 2L, 2L, 1L, 0L, 3L, 0L, 0L, 3L, 0L, 2L, 4L, 4L, 2L, 0L, 2L,
0L, 0L, 4L, 0L, 3L, 1L, 1L, 0L, 0L, 0L, 0L, 0L, 3L, 0L, 0L, 3L, 0L, 0L, 0L, 4L, 0L, 3L, 0L, 3L, 1L, 3L, 4L, 3L,
0L, 0L, 3L, 2L, 2L, 0L, 4L, 4L, 2L), Atr17 = c(3L, 4L, 3L, 4L, 1L, 0L, 3L, 3L, 0L, 3L, 0L, 3L, 0L, 3L, 0L, 3L, 0L, 3L, 4L, 0L,
3L, 0L, 0L, 3L, 0L, 1L, 0L, 0L, 3L, 3L, 0L, 3L, 0L, 4L, 0L, 0L, 0L, 3L, 4L, 0L, 0L, 0L, 3L, 4L, 0L, 0L, 0L, 0L,
4L, 1L, 3L, 3L, 0L, 1L, 4L, 3L, 0L, 3L, 0L, 0L, 1L, 0L, 3L, 4L, 4L, 0L, 0L, 0L, 3L, 4L, 2L, 0L, 4L, 0L, 1L, 4L,
0L, 3L, 0L, 3L, 3L, 3L, 2L, 0L, 3L, 0L, 0L, 4L, 0L, 3L, 0L, 0L, 0L, 0L, 0L, 0L, 0L, 0L, 3L, 0L, 0L, 3L, 0L, 0L, 0L,
0L, 3L, 0L, 4L, 0L, 4L, 1L, 4L, 3L, 4L, 0L, 2L, 3L, 3L, 0L, 3L, 3L, 0L, 3L, 3L, 4L), Atr18 = c(2L, 2L, 4L, 4L, 0L, 0L,
3L, 4L, 0L, 4L, 0L, 4L, 0L, 2L, 3L, 0L, 3L, 0L, 0L, 3L, 0L, 0L, 0L, 1L, 3L, 3L, 0L, 3L, 0L, 4L, 1L, 0L, 0L, 3L,
3L, 0L, 0L, 3L, 3L, 3L, 0L, 0L, 0L, 0L, 4L, 0L, 4L, 3L, 0L, 0L, 3L, 3L, 0L, 4L, 0L, 0L, 1L, 0L, 3L, 3L, 2L, 0L,
0L, 0L, 2L, 2L, 1L, 0L, 4L, 0L, 0L, 4L, 0L, 3L, 0L, 3L, 4L, 3L, 2L, 0L, 4L, 0L, 0L, 4L, 0L, 3L, 0L, 1L, 0L, 0L, 0L,
0L, 0L, 3L, 0L, 0L, 3L, 0L, 0L, 0L, 0L, 3L, 0L, 3L, 0L, 3L, 1L, 3L, 2L, 4L, 0L, 1L, 3L, 3L, 2L, 1L, 4L, 3L, 4
L), Atr19 = c(3L, 4L, 3L, 4L, 0L, 1L, 4L, 3L, 0L, 3L, 0L, 3L, 2L, 3L, 4L, 0L, 3L, 0L, 0L, 3L, 0L, 1L, 0L, 1L, 3
L, 3L, 0L, 3L, 0L, 4L, 0L, 0L, 0L, 3L, 4L, 0L, 0L, 3L, 3L, 3L, 0L, 0L, 0L, 0L, 4L, 0L, 3L, 3L, 0L, 0L, 4L, 3L,
1L, 3L, 0L, 0L, 0L, 0L, 3L, 4L, 4L, 0L, 0L, 0L, 3L, 4L, 0L, 0L, 4L, 0L, 1L, 4L, 0L, 3L, 0L, 3L, 3L, 3L, 2L, 1L,
3L, 0L, 0L, 3L, 0L, 3L, 0L, 0L, 0L, 0L, 0L, 0L, 0L, 0L, 0L, 0L, 3L, 0L, 4L, 0L, 4L, 1L, 4L, 3L, 4L, 0L, 0L, 3L, 3L, 3L, 0L, 3L, 3L, 4L), Atr20 = c(2L, 2L, 4L, 4L, 0L, 0L, 3L, 4L, 1L, 4L, 0L, 4L, 0L, 2L, 3L, 0
L, 3L, 0L, 0L, 3L, 0L, 0L, 0L, 0L, 3L, 3L, 0L, 4L, 0L, 4L, 1L, 0L, 0L, 2L, 3L, 0L, 0L, 3L, 2L, 3L, 0L, 0L, 0L,
0L, 4L, 0L, 4L, 3L, 0L, 0L, 3L, 3L, 0L, 4L, 0L, 0L, 0L, 0L, 4L, 3L, 2L, 0L, 0L, 0L, 2L, 2L, 0L, 0L, 4L, 0L, 0L,
4L, 0L, 3L, 0L, 3L, 4L, 3L, 2L, 0L, 4L, 0L, 0L, 2L, 0L, 2L, 0L, 0L, 0L, 0L, 0L, 0L, 0L, 0L, 0L, 0L, 0L, 3L, 0L, 0L, 0L,
1L, 0L, 3L, 0L, 3L, 0L, 3L, 1L, 3L, 2L, 4L, 0L, 1L, 3L, 3L, 2L, 0L, 4L, 3L, 4L), Atr21 = c(2L, 3L, 2L, 4L, 0L,
0L, 3L, 3L, 0L, 3L, 0L, 3L, 0L, 3L, 2L, 1L, 4L, 1L, 0L, 3L, 0L, 0L, 0L, 0L, 2L, 3L, 1L, 2L, 0L, 4L, 0L, 0L, 0L,

```

```

2L, 2L, 0L, 0L, 3L, 2L, 2L, 0L, 0L, 0L, 0L, 0L, 1L,      4L, 2L, 0L, 1L, 3L, 3L, 0L, 3L, 0L, 0L, 0L, 0L, 2L, 3L, 3L,
0L, 0L, 0L, 3L, 3L, 0L, 0L, 4L, 0L, 0L, 4L, 0L, 2L, 0L, 3L,      3L, 3L, 3L, 1L, 3L, 1L, 0L, 1L, 0L, 3L, 0L, 0L, 0L, 0L,
0L,      0L, 0L, 3L, 1L, 0L, 4L, 0L, 0L, 1L, 0L, 3L, 0L, 3L, 0L, 3L,      1L, 4L, 3L, 4L, 0L, 0L, 3L, 3L, 3L, 0L, 3L, 4L,
3L), Atr22 = c(3L,      2L, 3L, 3L, 0L, 0L, 3L, 4L, 0L, 4L, 0L, 3L, 0L, 2L, 3L, 0L,      4L, 0L, 0L, 4L, 0L, 0L, 0L, 0L,
3L, 2L, 0L, 1L, 0L, 3L, 0L,      0L, 0L, 1L, 3L, 0L, 0L, 2L, 1L, 3L, 0L, 0L, 0L, 0L, 0L,      4L, 3L, 0L, 0L, 3L, 2L,
1L, 4L, 0L, 0L, 0L, 0L, 1L, 4L, 4L,      0L, 0L, 0L, 2L, 2L, 1L, 0L, 3L, 0L, 0L, 3L, 0L, 2L, 0L, 2L,      4L, 4L, 2L, 0L,
2L, 1L, 0L, 1L, 0L, 3L, 0L, 0L, 0L, 0L,      0L, 0L, 3L, 1L, 0L, 3L, 0L, 0L, 0L, 0L, 4L, 0L, 3L, 0L, 3L,      0L, 3L,
4L, 3L, 0L, 0L, 3L, 2L, 2L, 0L, 4L, 4L, 2L), Atr23 = c(3L,      4L, 3L, 0L, 0L, 0L, 3L, 3L, 0L, 3L, 0L, 3L, 0L, 3L, 4L, 0
L,      3L, 0L, 0L, 3L, 0L, 0L, 0L, 0L, 3L, 3L, 0L, 0L, 0L, 4L, 0L,      0L, 0L, 1L, 4L, 0L, 0L, 3L, 2L, 3L, 0L, 0L, 0L,
0L, 2L, 0L,      3L, 3L, 0L, 0L, 4L, 3L, 1L, 3L, 0L, 0L, 0L, 0L, 0L, 4L, 4L,      0L, 0L, 0L, 3L, 4L, 0L, 0L, 4L, 0L, 0L,
4L, 0L, 2L, 0L, 3L,      3L, 3L, 2L, 0L, 3L, 0L, 2L, 0L, 0L, 3L, 0L, 0L, 0L, 0L,      0L, 0L, 3L, 1L, 0L, 3L, 0L, 0L,
1L, 0L, 3L, 0L, 4L, 0L, 4L,      0L, 4L, 3L, 4L, 0L, 0L, 3L, 3L, 3L, 2L, 3L, 3L, 4L), Atr24 = c(2L,      2L, 4L, 4L, 0L,
0L, 4L, 4L, 0L, 4L, 0L, 4L, 0L, 2L, 3L, 0L,      3L, 1L, 1L, 3L, 0L, 0L, 0L, 1L, 3L, 3L, 0L, 2L, 0L, 4L, 0L,      0L, 0L,
3L, 3L, 0L, 0L, 3L, 0L, 3L, 0L, 0L, 0L, 0L, 4L, 1L,      4L, 3L, 2L, 0L, 3L, 3L, 1L, 4L, 0L, 1L, 1L, 0L, 2L, 3L, 2L,
2L, 1L, 0L, 2L, 2L, 1L, 0L, 4L, 0L, 0L, 4L, 0L, 3L, 0L, 3L,      4L, 3L, 2L, 0L, 4L, 0L, 2L, 2L, 0L, 3L, 0L, 1L, 0L, 0L,
0L,      0L, 0L, 3L, 0L, 0L, 3L, 0L, 2L, 1L, 0L, 3L, 0L, 3L, 0L, 3L,      0L, 3L, 2L, 4L, 0L, 0L, 3L, 3L, 2L, 2L, 4L, 3L,
4L), Atr25 = c(3L,      4L, 3L, 0L, 0L, 0L, 3L, 3L, 0L, 3L, 0L, 3L, 0L, 3L, 4L, 1L,      3L, 1L, 0L, 3L, 0L, 0L, 0L, 0L,
3L, 3L, 1L, 2L, 0L, 4L, 0L,      0L, 0L, 3L, 4L, 1L, 0L, 3L, 0L, 3L, 0L, 0L, 0L, 4L, 0L,      3L, 3L, 0L, 0L, 4L, 3L,
1L, 3L, 1L, 1L, 1L, 1L, 2L, 4L, 4L,      0L, 1L, 0L, 3L, 4L, 1L, 0L, 4L, 1L, 0L, 4L, 1L, 2L, 1L, 3L,      3L, 3L, 2L, 0L,
3L, 0L, 0L, 2L, 0L, 2L, 2L, 1L, 0L, 0L, 0L,      0L, 1L, 3L, 1L, 0L, 3L, 0L, 0L, 0L, 1L, 3L, 1L, 4L, 0L, 4L,      0L, 4L,
3L, 4L, 1L, 2L, 3L, 3L, 3L, 1L, 3L, 3L, 4L), Atr26 = c(2L,      2L, 4L, 4L, 0L, 0L, 3L, 4L, 0L, 4L, 0L, 4L, 0L, 2L, 3L, 0
L,      3L, 0L, 0L, 3L, 0L, 0L, 0L, 0L, 3L, 3L, 0L, 2L, 0L, 4L, 0L,      0L, 0L, 3L, 3L, 1L, 0L, 3L, 0L, 3L, 0L, 0L, 0L,
0L, 4L, 1L,      4L, 3L, 0L, 0L, 3L, 3L, 1L, 4L, 0L, 0L, 1L, 0L, 2L, 3L, 2L,      0L, 1L, 0L, 2L, 2L, 0L, 0L, 4L, 1L, 0L,
4L, 1L, 3L, 1L, 3L,      4L, 3L, 2L, 0L, 4L, 0L, 0L, 1L, 0L, 3L, 1L, 0L, 0L, 0L,      0L, 1L, 3L, 0L, 0L, 3L, 1L, 0L,
0L, 0L, 3L, 1L, 3L, 0L, 3L,      0L, 3L, 2L, 4L, 1L, 2L, 3L, 3L, 2L, 1L, 4L, 3L, 4L), Atr27 = c(2L,      3L, 2L, 4L, 0L,
0L, 2L, 3L, 0L, 3L, 0L, 3L, 0L, 3L, 2L, 0L,      4L, 0L, 0L, 3L, 0L, 0L, 0L, 1L, 2L, 3L, 1L, 2L, 0L, 4L, 0L,      0L, 0L,
2L, 2L, 1L, 0L, 3L, 0L, 2L, 0L, 0L, 0L, 0L, 4L, 0L,      4L, 2L, 0L, 0L, 3L, 3L, 0L, 3L, 0L, 0L, 0L, 0L, 2L, 3L, 3L,
0L, 0L, 0L, 3L, 3L, 1L, 0L, 4L, 0L, 0L, 4L, 0L, 2L, 0L, 3L,      3L, 3L, 3L, 0L, 3L, 0L, 1L, 2L, 0L, 3L, 1L, 0L, 0L, 0L,
0L,      0L, 0L, 3L, 0L, 0L, 4L, 1L, 0L, 0L, 0L, 3L, 1L, 3L, 0L, 3L,      0L, 4L, 3L, 4L, 1L, 0L, 3L, 3L, 3L, 1L, 3L, 4L,
3L), Atr28 = c(3L,      2L, 3L, 0L, 0L, 0L, 4L, 4L, 0L, 4L, 0L, 3L, 0L, 2L, 3L, 1L,      4L, 0L, 0L, 4L, 0L, 0L, 0L, 0L,
3L, 2L, 1L, 2L, 0L, 3L, 0L,      0L, 0L, 3L, 3L, 0L, 0L, 2L, 0L, 3L, 0L, 0L, 0L, 0L, 1L, 0L,      4L, 3L, 0L, 0L, 3L, 2L,
0L, 4L, 0L, 0L, 0L, 0L, 2L, 4L, 4L,      0L, 0L, 0L, 2L, 2L, 0L, 0L, 3L, 0L, 0L, 3L, 0L, 3L, 0L, 2L,      4L, 4L, 2L, 0L,
2L, 1L, 1L, 0L, 0L, 2L, 0L, 0L, 0L, 0L,      0L, 0L, 3L, 0L, 0L, 3L, 1L, 0L, 0L, 0L, 4L, 1L, 3L, 0L, 3L,      0L, 3L,
4L, 3L, 0L, 0L, 3L, 2L, 2L, 0L, 4L, 4L, 2L), Atr29 = c(3L,      4L, 3L, 4L, 0L, 0L, 3L, 3L, 0L, 3L, 0L, 3L, 0L, 3L, 4L, 1
L,      3L, 1L, 0L, 3L, 0L, 0L, 0L, 0L, 3L, 3L, 1L, 2L, 0L, 4L, 0L,      0L, 0L, 2L, 4L, 0L, 0L, 3L, 2L, 3L, 0L, 0L, 0L,
0L, 1L, 0L,      3L, 3L, 0L, 0L, 4L, 3L, 0L, 3L, 0L, 0L, 0L, 0L, 2L, 4L, 4L,      0L, 0L, 0L, 3L, 4L, 0L, 0L, 4L, 0L, 0L,
4L, 1L, 2L, 0L, 3L,      3L, 3L, 2L, 0L, 3L, 0L, 0L, 1L, 0L, 2L, 0L, 0L, 0L, 0L,      0L, 1L, 3L, 0L, 0L, 3L, 0L, 0L,
0L, 0L, 3L, 1L, 4L, 0L, 4L,      0L, 4L, 3L, 4L, 0L, 0L, 3L, 3L, 3L, 1L, 3L, 3L, 4L), Atr30 = c(2L,      2L, 4L, 4L, 0L,
0L, 4L, 4L, 1L, 4L, 0L, 4L, 0L, 2L, 3L, 1L,      3L, 0L, 0L, 3L, 0L, 1L, 0L, 0L, 3L, 3L, 1L, 3L, 0L, 4L, 0L,      0L, 0L,
2L, 3L, 0L, 0L, 3L, 2L, 3L, 0L, 0L, 0L, 0L, 0L, 0L,      4L, 3L, 1L, 0L, 3L, 3L, 0L, 4L, 1L, 0L, 0L, 0L, 3L, 3L, 2L,
0L, 0L, 0L, 2L, 2L, 0L, 0L, 4L, 2L, 0L, 4L, 0L, 3L, 1L, 3L,      4L, 3L, 2L, 0L, 4L, 0L, 0L, 1L, 0L, 2L, 0L, 1L, 0L, 0L,
0L,      0L, 0L, 3L, 1L, 0L, 3L, 0L, 0L, 0L, 1L, 3L, 1L, 3L, 0L, 3L,      0L, 3L, 2L, 4L, 0L, 0L, 3L, 3L, 2L, 0L, 4L, 3L,
4L), Atr31 = c(4L,      4L, 4L, 0L, 0L, 2L, 4L, 3L, 1L, 4L, 0L, 3L, 0L, 4L, 3L, 1L,      4L, 2L, 0L, 3L, 1L, 0L, 0L, 0L,
4L, 4L, 0L, 3L, 4L, 4L, 1L,      0L, 2L, 3L, 4L, 1L, 0L, 4L, 3L, 4L, 2L, 2L, 2L, 0L, 4L, 2L,      4L, 4L, 0L, 0L, 4L, 4L,
2L, 4L, 2L, 0L, 0L, 1L, 3L, 4L, 4L,      0L, 2L, 0L, 4L, 4L, 0L, 0L, 4L, 0L, 0L, 4L, 0L, 1L, 1L, 3L,      4L, 4L, 4L, 0L,

```

```

4L, 0L, 1L, 0L, 2L, 1L, 0L, 1L, 0L, 2L, 1L, 0L, 0L, 4L, 1L, 1L, 4L, 2L, 0L, 1L, 0L, 4L, 1L, 4L, 0L, 4L, 2L, 4L,
4L, 4L, 0L, 4L, 3L, 4L, 4L, 0L, 3L, 3L, 4L), Atr32 = c(4L, 4L, 4L, 0L, 0L, 4L, 3L, 2L, 4L, 0L, 4L, 2L, 4L, 4L, 0
L, 4L, 0L, 1L, 4L, 0L, 1L, 0L, 0L, 3L, 4L, 0L, 3L, 4L, 4L, 1L, 0L, 2L, 3L, 3L, 0L, 0L, 3L, 3L, 4L, 1L, 1L, 0L,
1L, 2L, 0L, 3L, 4L, 2L, 0L, 3L, 4L, 1L, 4L, 1L, 0L, 0L, 3L, 3L, 3L, 4L, 0L, 1L, 0L, 4L, 4L, 0L, 0L, 3L, 0L, 0L,
3L, 0L, 1L, 1L, 4L, 4L, 4L, 0L, 4L, 0L, 0L, 4L, 0L, 2L, 0L, 0L, 2L, 1L, 0L, 0L, 0L, 3L, 1L, 1L, 4L, 1L, 0L,
3L, 2L, 4L, 0L, 4L, 0L, 3L, 0L, 3L, 4L, 4L, 0L, 1L, 3L, 4L, 4L, 0L, 3L, 3L, 4L), Atr33 = c(4L, 4L, 4L, 4L, 0L,
0L, 4L, 3L, 1L, 4L, 0L, 3L, 0L, 4L, 3L, 0L, 4L, 0L, 1L, 3L, 0L, 0L, 0L, 0L, 4L, 4L, 0L, 3L, 3L, 4L, 1L, 0L, 0L,
3L, 4L, 0L, 1L, 4L, 3L, 4L, 2L, 1L, 0L, 0L, 0L, 0L, 4L, 4L, 0L, 0L, 4L, 4L, 0L, 4L, 0L, 0L, 1L, 0L, 3L, 4L, 4L,
0L, 1L, 0L, 4L, 4L, 0L, 0L, 4L, 0L, 0L, 4L, 0L, 1L, 2L, 3L, 4L, 4L, 4L, 0L, 4L, 0L, 0L, 2L, 0L, 2L, 0L, 0L, 0L, 1L,
0L, 0L, 0L, 4L, 1L, 0L, 4L, 0L, 0L, 0L, 0L, 4L, 0L, 4L, 0L, 4L, 0L, 4L, 4L, 4L, 0L, 1L, 3L, 4L, 4L, 0L, 3L, 3L,
4L), Atr34 = c(4L, 4L, 4L, 0L, 0L, 0L, 4L, 3L, 2L, 4L, 0L, 4L, 1L, 4L, 4L, 0L, 4L, 1L, 1L, 4L, 0L, 1L, 0L, 0L,
3L, 4L, 0L, 3L, 4L, 4L, 1L, 0L, 0L, 3L, 3L, 0L, 1L, 3L, 3L, 4L, 0L, 1L, 1L, 0L, 0L, 0L, 3L, 4L, 1L, 0L, 3L, 4L,
1L, 4L, 2L, 0L, 0L, 0L, 3L, 3L, 4L, 0L, 1L, 1L, 4L, 4L, 1L, 0L, 3L, 1L, 0L, 3L, 0L, 1L, 2L, 4L, 4L, 4L, 4L, 0L,
4L, 1L, 0L, 3L, 0L, 1L, 0L, 0L, 0L, 1L, 1L, 1L, 0L, 3L, 1L, 0L, 4L, 0L, 0L, 0L, 0L, 4L, 0L, 4L, 0L, 3L, 0L, 3L,
4L, 4L, 0L, 1L, 3L, 4L, 4L, 0L, 3L, 3L, 4L), Atr35 = c(4L, 4L, 4L, 1L, 0L, 0L, 4L, 3L, 0L, 4L, 0L, 3L, 0L, 4L, 3L, 0
L, 4L, 0L, 0L, 3L, 0L, 0L, 0L, 0L, 4L, 4L, 0L, 3L, 0L, 4L, 0L, 0L, 0L, 3L, 4L, 0L, 0L, 4L, 3L, 4L, 0L, 0L, 0L,
0L, 0L, 0L, 4L, 4L, 0L, 0L, 4L, 4L, 0L, 4L, 0L, 0L, 1L, 0L, 3L, 4L, 4L, 0L, 0L, 0L, 4L, 4L, 0L, 0L, 4L, 0L, 0L,
4L, 0L, 1L, 0L, 3L, 4L, 4L, 4L, 0L, 4L, 0L, 0L, 0L, 0L, 1L, 0L, 0L, 0L, 0L, 0L, 0L, 0L, 4L, 0L, 0L, 0L, 0L,
0L, 0L, 4L, 0L, 4L, 0L, 4L, 0L, 4L, 0L, 4L, 0L, 1L, 3L, 4L, 4L, 0L, 3L, 3L, 4L), Atr36 = c(4L, 4L, 4L, 0L, 0L,
0L, 4L, 3L, 1L, 4L, 0L, 4L, 0L, 4L, 4L, 0L, 4L, 0L, 0L, 4L, 0L, 0L, 0L, 3L, 4L, 0L, 3L, 0L, 4L, 0L, 0L, 0L,
3L, 3L, 0L, 0L, 3L, 3L, 4L, 0L, 0L, 0L, 0L, 0L, 0L, 3L, 4L, 0L, 0L, 3L, 4L, 0L, 4L, 0L, 0L, 0L, 0L, 3L, 3L, 4L,
0L, 0L, 0L, 4L, 4L, 0L, 0L, 3L, 0L, 0L, 3L, 0L, 1L, 0L, 4L, 4L, 4L, 4L, 0L, 4L, 0L, 0L, 2L, 0L, 2L, 0L, 0L, 0L, 0L,
0L, 0L, 0L, 3L, 0L, 0L, 4L, 0L, 0L, 0L, 0L, 4L, 0L, 4L, 0L, 3L, 0L, 3L, 4L, 4L, 0L, 1L, 3L, 4L, 4L, 0L, 3L, 3L,
4L), Atr37 = c(4L, 4L, 4L, 2L, 0L, 0L, 4L, 3L, 2L, 4L, 0L, 3L, 0L, 4L, 3L, 0L, 4L, 1L, 0L, 3L, 0L, 1L, 0L, 0L,
4L, 4L, 0L, 4L, 4L, 4L, 1L, 1L, 0L, 3L, 4L, 0L, 0L, 4L, 3L, 4L, 2L, 1L, 1L, 0L, 3L, 2L, 4L, 4L, 0L, 0L, 4L, 4L,
1L, 4L, 1L, 0L, 1L, 2L, 4L, 4L, 4L, 0L, 2L, 0L, 4L, 4L, 0L, 0L, 4L, 1L, 0L, 4L, 0L, 1L, 0L, 3L, 4L, 4L, 4L, 0L,
4L, 1L, 0L, 3L, 4L, 3L, 0L, 0L, 0L, 0L, 0L, 1L, 0L, 4L, 1L, 0L, 4L, 1L, 1L, 0L, 0L, 4L, 0L, 4L, 2L, 4L, 0L, 4L,
4L, 4L, 0L, 1L, 3L, 4L, 4L, 0L, 3L, 3L, 4L), Atr38 = c(4L, 4L, 4L, 0L, 0L, 1L, 4L, 3L, 1L, 4L, 3L, 4L, 2L, 4L, 4L, 0
L, 4L, 0L, 2L, 4L, 0L, 1L, 0L, 0L, 3L, 4L, 0L, 4L, 0L, 4L, 0L, 0L, 0L, 0L, 3L, 3L, 0L, 1L, 3L, 3L, 4L, 0L, 1L, 0L,
0L, 0L, 2L, 3L, 4L, 0L, 0L, 4L, 4L, 0L, 4L, 1L, 0L, 0L, 1L, 4L, 3L, 4L, 0L, 0L, 0L, 4L, 4L, 0L, 0L, 3L, 1L, 0L,
3L, 0L, 2L, 0L, 4L, 4L, 4L, 4L, 0L, 4L, 0L, 0L, 4L, 0L, 2L, 0L, 0L, 0L, 1L, 0L, 0L, 0L, 3L, 0L, 0L, 4L, 1L, 0L,
0L, 0L, 4L, 0L, 4L, 1L, 3L, 0L, 3L, 4L, 4L, 0L, 2L, 3L, 4L, 4L, 0L, 3L, 3L, 4L), Atr39 = c(4L, 4L, 4L, 4L, 0L,
0L, 4L, 3L, 3L, 4L, 0L, 3L, 0L, 4L, 3L, 0L, 4L, 2L, 1L, 3L, 0L, 1L, 0L, 1L, 4L, 4L, 0L, 4L, 1L, 4L, 1L, 0L, 0L,
4L, 4L, 0L, 1L, 4L, 1L, 4L, 1L, 2L, 0L, 0L, 4L, 1L, 4L, 4L, 0L, 0L, 3L, 4L, 0L, 4L, 1L, 0L, 1L, 0L, 4L, 4L, 4L,
0L, 1L, 0L, 4L, 4L, 0L, 0L, 4L, 2L, 1L, 4L, 0L, 2L, 0L, 3L, 4L, 4L, 4L, 0L, 4L, 1L, 0L, 2L, 0L, 2L, 0L, 0L, 0L, 1L,
0L, 4L, 0L, 4L, 0L, 0L, 4L, 2L, 0L, 0L, 0L, 4L, 0L, 4L, 1L, 4L, 0L, 4L, 4L, 4L, 0L, 0L, 3L, 4L, 4L, 0L, 3L, 3L,
4L), Atr40 = c(4L, 4L, 4L, 0L, 0L, 0L, 4L, 3L, 3L, 4L, 0L, 4L, 0L, 4L, 4L, 0L, 4L, 0L, 0L, 4L, 0L, 0L, 0L, 1L,
3L, 4L, 0L, 4L, 0L, 4L, 1L, 0L, 0L, 4L, 3L, 0L, 0L, 3L, 3L, 4L, 0L, 1L, 0L, 0L, 4L, 0L, 3L, 4L, 0L, 0L, 4L, 4L,
0L, 4L, 1L, 0L, 0L, 2L, 4L, 3L, 4L, 0L, 0L, 0L, 4L, 4L, 0L, 0L, 3L, 0L, 0L, 3L, 0L, 2L, 0L, 4L, 4L, 4L, 4L, 0L,
4L, 0L, 0L, 4L, 0L, 3L, 0L, 0L, 0L, 0L, 0L, 1L, 0L, 3L, 0L, 0L, 4L, 1L, 0L, 0L, 0L, 4L, 0L, 4L, 0L, 3L, 0L, 3L,
4L, 4L, 0L, 2L, 3L, 4L, 4L, 0L, 3L, 3L, 4L), Atr41 = c(4L, 4L, 4L, 2L, 0L, 0L, 4L, 3L, 2L, 4L, 1L, 3L, 2L, 4L, 3L, 0
L, 4L, 1L, 0L, 3L, 0L, 0L, 0L, 0L, 4L, 4L, 0L, 3L, 1L, 4L, 1L, 0L, 0L, 4L, 4L, 0L, 0L, 4L, 2L, 4L, 0L, 1L, 1L,
0L, 4L, 0L, 4L, 4L, 0L, 1L, 4L, 4L, 1L, 4L, 0L, 0L, 0L, 0L, 4L, 4L, 4L, 0L, 1L, 0L, 4L, 4L, 0L, 1L, 4L, 1L, 0L,
4L, 1L, 2L, 1L, 3L, 4L, 4L, 4L, 0L, 4L, 0L, 3L, 2L, 0L, 3L, 0L, 0L, 0L, 0L, 0L, 2L, 0L, 4L, 0L, 0L, 4L, 0L, 0L,
0L, 0L, 4L, 2L, 4L, 1L, 4L, 0L, 4L, 4L, 4L, 0L, 2L, 3L, 4L, 4L, 0L, 3L, 3L, 4L), Atr42 = c(4L, 4L, 4L, 4L, 2L,

```



```

0L, 3L, 3L, 1L, 4L, 1L, 4L, 2L, 4L, 4L, 1L,      4L, 3L, 0L, 4L, 0L, 4L, 1L, 0L, 3L, 4L, 1L, 0L, 0L, 4L, 1L,      1L, 4L,
4L, 3L, 0L, 3L, 3L, 3L, 4L, 0L, 3L, 1L, 0L, 4L, 2L,      3L, 4L, 0L, 0L, 4L, 4L, 0L, 4L, 2L, 0L, 0L, 0L, 3L, 3L, 4L,
0L, 0L, 0L, 4L, 4L, 0L, 2L, 3L, 2L, 2L, 3L, 0L, 2L, 0L, 4L,      4L, 4L, 3L, 0L, 4L, 0L, 1L, 2L, 0L, 3L, 0L, 2L, 0L, 3L,
2L,      1L, 2L, 3L, 0L, 0L, 4L, 2L, 0L, 2L, 2L, 4L, 0L, 4L, 3L, 3L,      0L, 3L, 4L, 4L, 0L, 1L, 3L, 4L, 4L, 0L, 3L, 3L,
4L), Atr43 = c(4L,      4L, 4L, 4L, 4L, 2L, 4L, 3L, 1L, 4L, 1L, 3L, 3L, 4L, 3L, 3L,      4L, 2L, 2L, 3L, 0L, 2L, 1L, 3L,
4L, 4L, 2L, 0L, 0L, 4L, 1L,      2L, 4L, 3L, 4L, 2L, 2L, 4L, 3L, 4L, 2L, 2L, 2L, 4L, 4L, 2L,      4L, 4L, 3L, 1L, 4L, 4L,
2L, 4L, 2L, 1L, 1L, 0L, 0L, 4L, 4L,      0L, 0L, 3L, 4L, 4L, 0L, 4L, 4L, 3L, 2L, 4L, 2L, 2L, 1L, 3L,      4L, 4L, 4L, 3L,
4L, 2L, 1L, 3L, 0L, 3L, 1L, 4L, 0L, 2L, 2L,      1L, 3L, 4L, 0L, 2L, 4L, 2L, 2L, 2L, 2L, 2L, 4L, 3L, 4L, 4L, 4L,      2L, 4L,
4L, 4L, 3L, 2L, 3L, 4L, 4L, 4L, 3L, 3L, 4L), Atr44 = c(4L,      4L, 4L, 1L, 4L, 0L, 3L, 3L, 2L, 4L, 0L, 4L, 0L, 4L, 4L, 2
L,      4L, 0L, 1L, 4L, 2L, 0L, 1L, 2L, 3L, 4L, 0L, 3L, 0L, 4L, 1L,      0L, 0L, 4L, 3L, 1L, 1L, 3L, 4L, 4L, 0L, 0L, 0L,
0L, 4L, 1L,      3L, 4L, 0L, 0L, 3L, 4L, 0L, 4L, 0L, 0L, 0L, 0L, 0L, 3L, 4L,      0L, 0L, 0L, 4L, 4L, 0L, 2L, 3L, 0L, 0L,
3L, 0L, 2L, 2L, 4L,      3L, 4L, 4L, 0L, 4L, 2L, 0L, 4L, 0L, 4L, 1L, 1L, 0L, 1L, 0L,      0L, 1L, 3L, 0L, 0L, 4L, 2L, 2L,
0L, 2L, 4L, 0L, 4L, 0L, 3L,      0L, 3L, 4L, 4L, 1L, 3L, 3L, 4L, 4L, 4L, 3L, 3L, 4L), Atr45 = c(4L,      4L, 3L, 4L, 4L,
2L, 3L, 3L, 3L, 4L, 3L, 3L, 3L, 4L, 3L, 1L,      4L, 1L, 1L, 3L, 1L, 1L, 2L, 4L, 4L, 3L, 2L, 0L, 0L, 4L, 1L,      1L, 4L,
2L, 4L, 0L, 3L, 3L, 4L, 4L, 1L, 2L, 1L, 4L, 4L, 2L,      3L, 4L, 0L, 0L, 4L, 4L, 0L, 4L, 0L, 3L, 0L, 0L, 0L, 4L, 4L,
0L, 0L, 0L, 4L, 4L, 1L, 3L, 4L, 1L, 1L, 4L, 1L, 2L, 0L, 3L,      4L, 3L, 4L, 3L, 4L, 2L, 1L, 2L, 0L, 3L, 0L, 0L, 0L, 2L,
3L,      4L, 3L, 4L, 0L, 0L, 3L, 1L, 4L, 1L, 2L, 4L, 3L, 4L, 4L, 4L,      2L, 4L, 3L, 4L, 0L, 0L, 3L, 4L, 4L, 4L, 3L, 3L,
4L), Atr46 = c(4L,      4L, 4L, 0L, 4L, 0L, 3L, 3L, 2L, 4L, 3L, 4L, 2L, 4L, 4L, 3L,      4L, 3L, 1L, 4L, 4L, 2L, 3L, 4L,
3L, 4L, 1L, 0L, 1L, 4L, 0L,      4L, 1L, 2L, 3L, 3L, 3L, 3L, 2L, 4L, 2L, 1L, 2L, 3L, 2L, 3L,      3L, 4L, 2L, 0L, 4L, 4L,
2L, 4L, 2L, 2L, 3L, 0L, 0L, 3L, 4L,      0L, 0L, 0L, 4L, 4L, 1L, 1L, 3L, 1L, 0L, 3L, 1L, 1L, 1L, 4L,      4L, 4L, 4L, 4L,
4L, 0L, 3L, 2L, 3L, 3L, 4L, 2L, 2L, 2L, 3L,      3L, 3L, 3L, 0L, 2L, 4L, 0L, 4L, 3L, 2L, 4L, 3L, 4L, 3L, 3L,      2L, 3L,
4L, 4L, 3L, 2L, 3L, 4L, 4L, 4L, 3L, 3L, 4L), Atr47 = c(4L,      4L, 4L, 4L, 4L, 2L, 4L, 3L, 1L, 4L, 2L, 3L, 0L, 4L, 3L, 0
L,      4L, 3L, 1L, 3L, 2L, 1L, 1L, 0L, 4L, 4L, 0L, 0L, 0L, 4L, 0L,      4L, 4L, 3L, 4L, 0L, 0L, 4L, 3L, 4L, 1L, 2L, 2L,
1L, 4L, 0L,      4L, 4L, 2L, 1L, 3L, 4L, 1L, 4L, 1L, 0L, 0L, 0L, 0L, 4L, 4L,      0L, 0L, 0L, 4L, 4L, 1L, 1L, 4L, 2L, 1L,
4L, 1L, 1L, 0L, 3L,      4L, 4L, 4L, 3L, 4L, 0L, 3L, 2L, 0L, 2L, 1L, 1L, 2L, 1L, 3L,      3L, 3L, 4L, 0L, 0L, 4L, 1L, 4L,
4L, 2L, 4L, 1L, 4L, 4L, 4L,      1L, 4L, 4L, 4L, 0L, 2L, 3L, 4L, 4L, 4L, 3L, 3L, 4L), Atr48 = c(4L,      4L, 4L, 4L, 4L,
3L, 4L, 3L, 3L, 4L, 2L, 4L, 2L, 4L, 4L, 2L,      4L, 2L, 2L, 4L, 3L, 3L, 0L, 1L, 3L, 4L, 0L, 3L, 4L, 4L, 2L,      3L, 2L,
3L, 3L, 2L, 2L, 3L, 2L, 4L, 3L, 3L, 2L, 3L, 4L, 2L,      3L, 4L, 0L, 0L, 4L, 4L, 2L, 4L, 2L, 2L, 2L, 3L, 3L, 3L, 4L,
4L, 2L, 2L, 4L, 4L, 0L, 2L, 3L, 2L, 2L, 3L, 1L, 1L, 3L, 4L,      4L, 4L, 4L, 0L, 4L, 2L, 2L, 3L, 2L, 3L, 1L, 2L, 3L, 2L,
3L,      2L, 2L, 3L, 2L, 2L, 4L, 3L, 2L, 4L, 2L, 4L, 2L, 4L, 3L, 3L,      1L, 3L, 4L, 4L, 1L, 1L, 3L, 4L, 4L, 0L, 3L, 3L,
4L), Atr49 = c(4L,      4L, 4L, 4L, 4L, 0L, 4L, 3L, 3L, 4L, 0L, 3L, 4L, 4L, 3L, 3L,      4L, 2L, 1L, 3L, 0L, 1L, 0L, 0L,
4L, 4L, 1L, 3L, 1L, 4L, 1L,      3L, 3L, 2L, 4L, 3L, 3L, 4L, 2L, 4L, 3L, 2L, 2L, 1L, 3L, 0L,      4L, 4L, 0L, 0L, 4L, 4L,
2L, 4L, 1L, 2L, 2L, 1L, 3L, 4L, 4L,      4L, 2L, 0L, 4L, 4L, 1L, 0L, 4L, 0L, 1L, 1L, 3L,      4L, 4L, 4L, 2L,
4L, 2L, 0L, 4L, 2L, 2L, 4L, 1L, 3L, 3L, 0L,      2L, 1L, 4L, 1L, 3L, 4L, 2L, 2L, 0L, 0L, 4L, 1L, 4L, 1L, 4L,      1L, 4L,
4L, 4L, 0L, 2L, 3L, 4L, 4L, 4L, 3L, 3L, 4L), Atr50 = c(4L,      4L, 4L, 3L, 2L, 3L, 4L, 3L, 3L, 4L, 2L, 3L, 0L, 4L, 3L, 3
L,      4L, 1L, 1L, 3L, 0L, 1L, 1L, 0L, 4L, 4L, 1L, 3L, 1L, 4L, 2L,      1L, 2L, 1L, 4L, 4L, 2L, 4L, 3L, 4L, 2L, 2L, 2L,
3L, 4L, 0L,      4L, 4L, 1L, 0L, 3L, 4L, 2L, 4L, 2L, 2L, 1L, 3L, 4L, 4L,      4L, 2L, 0L, 4L, 4L, 2L, 0L, 4L, 0L, 0L,
4L, 0L, 1L, 1L, 3L,      4L, 4L, 4L, 2L, 4L, 2L, 0L, 4L, 2L, 3L, 4L, 2L, 3L, 2L, 1L,      3L, 1L, 4L, 2L, 2L, 4L, 2L, 2L,
1L, 2L, 4L, 1L, 4L, 3L, 4L,      1L, 4L, 4L, 4L, 0L, 1L, 3L, 4L, 4L, 0L, 3L, 3L, 4L), Atr51 = c(4L,      4L, 4L, 4L, 2L,
3L, 3L, 3L, 2L, 4L, 2L, 4L, 0L, 4L, 4L, 2L,      4L, 1L, 1L, 4L, 0L, 1L, 1L, 1L, 3L, 4L, 1L, 3L, 4L, 4L, 2L,      2L, 2L,
1L, 3L, 3L, 2L, 3L, 3L, 4L, 2L, 2L, 2L, 3L, 4L, 2L,      3L, 4L, 1L, 1L, 3L, 4L, 2L, 4L, 2L, 2L, 2L, 1L, 3L, 3L, 4L,
4L, 2L, 1L, 4L, 4L, 2L, 1L, 3L, 1L, 0L, 3L, 0L, 1L, 3L, 4L,      4L, 4L, 4L, 2L, 4L, 2L, 0L, 4L, 2L, 3L, 2L, 2L, 3L, 2L,
3L,      2L, 1L, 3L, 2L, 2L, 4L, 2L, 2L, 3L, 1L, 4L, 1L, 4L, 3L, 3L,      1L, 3L, 4L, 4L, 0L, 1L, 3L, 4L, 4L, 0L, 3L, 3L,
4L), Atr52 = c(4L,      4L, 4L, 4L, 0L, 3L, 3L, 3L, 3L, 4L, 0L, 3L, 1L, 4L, 3L, 1L,      4L, 2L, 0L, 3L, 2L, 1L, 0L, 4L,
4L, 4L, 1L, 3L, 2L, 4L, 2L,      3L, 0L, 2L, 4L, 3L, 4L, 4L, 4L, 2L, 1L, 2L, 3L, 1L, 1L,      4L, 4L, 3L, 0L, 4L, 4L,

```

```

2L, 4L, 1L, 2L, 2L, 0L, 3L, 4L, 4L,      0L, 1L, 2L, 4L, 4L, 0L, 4L, 4L, 3L, 3L, 4L, 1L, 1L, 1L, 3L,      4L, 4L, 4L, 0L,
4L, 1L, 4L, 4L, 4L, 2L, 2L, 4L, 1L, 3L, 3L,      4L, 1L, 4L, 3L, 1L, 4L, 1L, 4L, 3L, 1L, 4L, 0L, 4L, 0L, 4L,      1L, 4L,
4L, 4L, 2L, 1L, 3L, 4L, 4L, 0L, 3L, 3L, 4L), Atr53 = c(4L,      4L, 4L, 4L, 0L, 2L, 4L, 3L, 2L, 4L, 0L, 4L, 0L, 4L, 4L, 0
L,      4L, 0L, 0L, 4L, 4L, 1L, 0L, 1L, 3L, 4L, 1L, 3L, 2L, 4L, 2L,      1L, 0L, 2L, 3L, 0L, 1L, 3L, 2L, 4L, 2L, 1L, 2L,
1L, 1L, 0L,      3L, 4L, 4L, 0L, 4L, 4L, 1L, 4L, 0L, 2L, 2L, 3L, 3L, 3L, 4L,      0L, 0L, 2L, 4L, 4L, 1L, 1L, 3L, 2L, 2L,
3L, 1L, 1L, 1L, 4L,      4L, 4L, 4L, 0L, 4L, 1L, 4L, 2L, 4L, 2L, 0L, 2L, 1L, 1L, 3L,      3L, 1L, 3L, 3L, 1L, 4L, 0L, 2L,
3L, 1L, 4L, 1L, 4L, 1L, 3L,      1L, 3L, 4L, 4L, 0L, 2L, 3L, 4L, 4L, 0L, 3L, 3L, 4L), Atr54 = c(4L,      4L, 4L, 4L, 0L,
2L, 4L, 3L, 1L, 4L, 4L, 3L, 0L, 4L, 3L, 0L,      4L, 0L, 0L, 3L, 1L, 0L, 0L, 0L, 4L, 4L, 1L, 3L, 2L, 4L, 3L,      1L, 0L,
2L, 4L, 2L, 1L, 4L, 2L, 4L, 0L, 0L, 2L, 0L, 0L, 0L,      4L, 4L, 0L, 1L, 3L, 4L, 0L, 4L, 0L, 2L, 1L, 0L, 3L, 4L, 4L,
0L, 0L, 1L, 4L, 4L, 1L, 2L, 4L, 2L, 1L, 4L, 1L, 1L, 1L, 3L,      4L, 4L, 4L, 0L, 4L, 0L, 2L, 2L, 0L, 2L, 1L, 0L, 0L, 1L,
1L,      1L, 1L, 4L, 2L, 0L, 4L, 0L, 2L, 1L, 0L, 4L, 0L, 4L, 0L, 4L,      1L, 4L, 4L, 4L, 0L, 0L, 3L, 4L, 4L, 0L, 3L, 3L,
4L), Class = c(1L,      1L, 1L, 1L, 0L, 0L, 1L, 1L, 1L, 1L, 0L, 1L, 0L, 1L, 1L, 0L,      1L, 0L, 0L, 1L, 0L, 0L, 0L, 0L,
1L, 1L, 0L, 1L, 0L, 1L, 0L,      0L, 0L, 1L, 1L, 0L, 0L, 1L, 1L, 1L, 0L, 0L, 0L, 0L, 1L, 0L,      1L, 1L, 0L, 0L, 1L, 1L,
0L, 1L, 0L, 0L, 0L, 0L, 1L, 1L, 1L,      0L, 0L, 0L, 1L, 1L, 0L, 0L, 1L, 0L, 0L, 1L, 0L, 1L,      1L, 1L, 1L, 0L,
1L, 0L, 0L, 1L, 0L, 1L, 0L, 0L, 0L, 0L,      0L, 0L, 1L, 0L, 0L, 1L, 0L, 0L, 0L, 0L, 1L, 0L, 1L, 0L, 1L,      0L, 1L,
1L, 1L, 0L, 1L, 1L, 1L, 0L, 1L, 1L, 1L)), class = "data.frame", row.names = c(NA, 119L))

```

Type of random forest: regression

Number of trees: 500

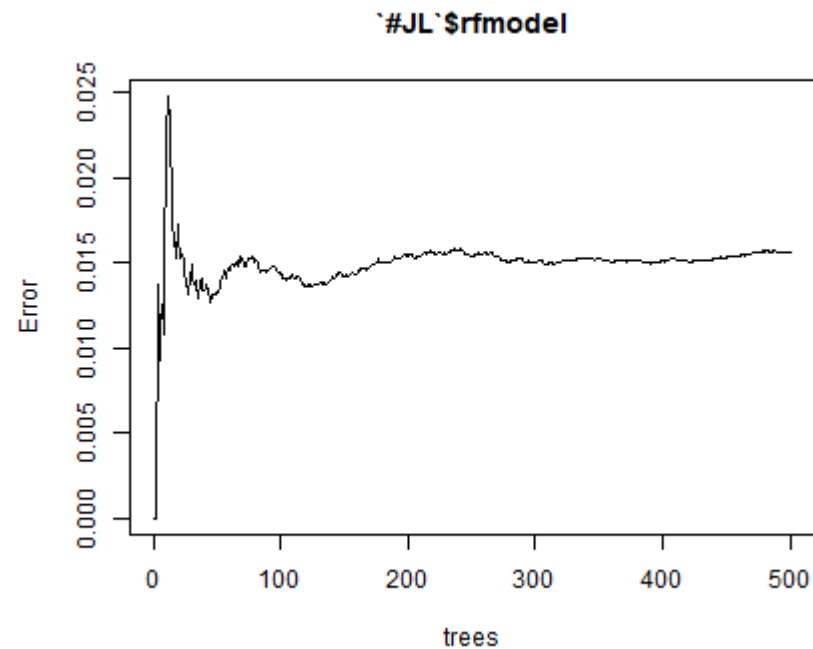
No. of variables tried at each split: 18

Mean of squared residuals: 0.01559753

% Var explained: 93.76

In [107...

```
R"plot($rfmodel)"
```



```
Out[107...] RObject{RealSxp}
[1] 2.101014e-34 4.019332e-34 1.373626e-02 9.350935e-03 9.259259e-03
[6] 1.151786e-02 1.253687e-02 1.080548e-02 1.304950e-02 2.341867e-02
[11] 2.399351e-02 2.478265e-02 2.334380e-02 1.784910e-02 1.741231e-02
[16] 1.636247e-02 1.666018e-02 1.525210e-02 1.729987e-02 1.669695e-02
[21] 1.521275e-02 1.547608e-02 1.552779e-02 1.509915e-02 1.448103e-02
[26] 1.391231e-02 1.324781e-02 1.308556e-02 1.412612e-02 1.488009e-02
[31] 1.421810e-02 1.367845e-02 1.394944e-02 1.337749e-02 1.340885e-02
[36] 1.294962e-02 1.378342e-02 1.403463e-02 1.334888e-02 1.333733e-02
[41] 1.346776e-02 1.366251e-02 1.346090e-02 1.261886e-02 1.284258e-02
[46] 1.291345e-02 1.317037e-02 1.309087e-02 1.322946e-02 1.309815e-02
[51] 1.326961e-02 1.360271e-02 1.382014e-02 1.406839e-02 1.428479e-02
[56] 1.450775e-02 1.412774e-02 1.421053e-02 1.463124e-02 1.458029e-02
[61] 1.469495e-02 1.484070e-02 1.488585e-02 1.477891e-02 1.490864e-02
[66] 1.509830e-02 1.474480e-02 1.504240e-02 1.541408e-02 1.520209e-02
[71] 1.507078e-02 1.476700e-02 1.475491e-02 1.526265e-02 1.528538e-02
[76] 1.529783e-02 1.519275e-02 1.531394e-02 1.519241e-02 1.495605e-02
[81] 1.517333e-02 1.496196e-02 1.485062e-02 1.441694e-02 1.452646e-02
[86] 1.455643e-02 1.458116e-02 1.446053e-02 1.456266e-02 1.449237e-02
[91] 1.449272e-02 1.465982e-02 1.481125e-02 1.480386e-02 1.472061e-02
[96] 1.470745e-02 1.458031e-02 1.442104e-02 1.434631e-02 1.448193e-02
[101] 1.433119e-02 1.404964e-02 1.409035e-02 1.397378e-02 1.407256e-02
[106] 1.402772e-02 1.414416e-02 1.412340e-02 1.425341e-02 1.407341e-02
```

[111] 1.412783e-02 1.416634e-02 1.412469e-02 1.422392e-02 1.401773e-02
[116] 1.407954e-02 1.386576e-02 1.361584e-02 1.369571e-02 1.365071e-02
[121] 1.372077e-02 1.360474e-02 1.371717e-02 1.357167e-02 1.364417e-02
[126] 1.368494e-02 1.376738e-02 1.368422e-02 1.369142e-02 1.373310e-02
[131] 1.381565e-02 1.383712e-02 1.375976e-02 1.379189e-02 1.373689e-02
[136] 1.369348e-02 1.386521e-02 1.384490e-02 1.394176e-02 1.407985e-02
[141] 1.413437e-02 1.410348e-02 1.413803e-02 1.425261e-02 1.433636e-02
[146] 1.440963e-02 1.437232e-02 1.425478e-02 1.418829e-02 1.419413e-02
[151] 1.425157e-02 1.418715e-02 1.417212e-02 1.431712e-02 1.431423e-02
[156] 1.442425e-02 1.437151e-02 1.431021e-02 1.431897e-02 1.435144e-02
[161] 1.448248e-02 1.463398e-02 1.455755e-02 1.462418e-02 1.470737e-02
[166] 1.460460e-02 1.460197e-02 1.468014e-02 1.470580e-02 1.479144e-02
[171] 1.476103e-02 1.483876e-02 1.487448e-02 1.490947e-02 1.500403e-02
[176] 1.505912e-02 1.519169e-02 1.499015e-02 1.495781e-02 1.497986e-02
[181] 1.501599e-02 1.497845e-02 1.505012e-02 1.503567e-02 1.505951e-02
[186] 1.505563e-02 1.513443e-02 1.510937e-02 1.521002e-02 1.532938e-02
[191] 1.523038e-02 1.527022e-02 1.528140e-02 1.525580e-02 1.531992e-02
[196] 1.539899e-02 1.552029e-02 1.543941e-02 1.540901e-02 1.535039e-02
[201] 1.542690e-02 1.547571e-02 1.543710e-02 1.545699e-02 1.530416e-02
[206] 1.527603e-02 1.536850e-02 1.538918e-02 1.542453e-02 1.540814e-02
[211] 1.544499e-02 1.551460e-02 1.556668e-02 1.552319e-02 1.560102e-02
[216] 1.555788e-02 1.557894e-02 1.569397e-02 1.558148e-02 1.559159e-02
[221] 1.542669e-02 1.543044e-02 1.548193e-02 1.555410e-02 1.558456e-02
[226] 1.554580e-02 1.553062e-02 1.553528e-02 1.557705e-02 1.547907e-02
[231] 1.548659e-02 1.557806e-02 1.561103e-02 1.572842e-02 1.576897e-02
[236] 1.578874e-02 1.567650e-02 1.571029e-02 1.572318e-02 1.570998e-02
[241] 1.573444e-02 1.578221e-02 1.576264e-02 1.565942e-02 1.553450e-02
[246] 1.552802e-02 1.551738e-02 1.545746e-02 1.535166e-02 1.534758e-02
[251] 1.535832e-02 1.541768e-02 1.554243e-02 1.552560e-02 1.556353e-02
[256] 1.549963e-02 1.548055e-02 1.547294e-02 1.547666e-02 1.554561e-02
[261] 1.552560e-02 1.555626e-02 1.552629e-02 1.553438e-02 1.554541e-02
[266] 1.554439e-02 1.553509e-02 1.541887e-02 1.536842e-02 1.532940e-02
[271] 1.535311e-02 1.523030e-02 1.515081e-02 1.512831e-02 1.516182e-02
[276] 1.516366e-02 1.513906e-02 1.500768e-02 1.506637e-02 1.501722e-02
[281] 1.509024e-02 1.513602e-02 1.512900e-02 1.515240e-02 1.513367e-02
[286] 1.521365e-02 1.527296e-02 1.528985e-02 1.521833e-02 1.519246e-02
[291] 1.507273e-02 1.507268e-02 1.506468e-02 1.519107e-02 1.506981e-02
[296] 1.505868e-02 1.503901e-02 1.505993e-02 1.500626e-02 1.497182e-02
[301] 1.503000e-02 1.510922e-02 1.514825e-02 1.516291e-02 1.508982e-02
[306] 1.501217e-02 1.490535e-02 1.489866e-02 1.486698e-02 1.489620e-02
[311] 1.497943e-02 1.501407e-02 1.499786e-02 1.490994e-02 1.491477e-02
[316] 1.496126e-02 1.500257e-02 1.500821e-02 1.502285e-02 1.503788e-02
[321] 1.508047e-02 1.507442e-02 1.513092e-02 1.512891e-02 1.515577e-02
[326] 1.513188e-02 1.508538e-02 1.512269e-02 1.513500e-02 1.512056e-02
[331] 1.516376e-02 1.517677e-02 1.520614e-02 1.523382e-02 1.528880e-02

```
[336] 1.526162e-02 1.526394e-02 1.520734e-02 1.520187e-02 1.520820e-02
[341] 1.520969e-02 1.519139e-02 1.520732e-02 1.522502e-02 1.526981e-02
[346] 1.517319e-02 1.513906e-02 1.520090e-02 1.522647e-02 1.523453e-02
[351] 1.520969e-02 1.523035e-02 1.514037e-02 1.518238e-02 1.509477e-02
[356] 1.508926e-02 1.508718e-02 1.508039e-02 1.499799e-02 1.502096e-02
[361] 1.503159e-02 1.505136e-02 1.508288e-02 1.512070e-02 1.512954e-02
[366] 1.515326e-02 1.513829e-02 1.512529e-02 1.509335e-02 1.511463e-02
[371] 1.505925e-02 1.512340e-02 1.512943e-02 1.513907e-02 1.515676e-02
[376] 1.511233e-02 1.510808e-02 1.511788e-02 1.511112e-02 1.509767e-02
[381] 1.512310e-02 1.508196e-02 1.510366e-02 1.500903e-02 1.502198e-02
[386] 1.499246e-02 1.497719e-02 1.502746e-02 1.505707e-02 1.495507e-02
[391] 1.502936e-02 1.500189e-02 1.495751e-02 1.502854e-02 1.502982e-02
[396] 1.502114e-02 1.502880e-02 1.504804e-02 1.509265e-02 1.508919e-02
[401] 1.506759e-02 1.508984e-02 1.509655e-02 1.512801e-02 1.515601e-02
[406] 1.521297e-02 1.522500e-02 1.522236e-02 1.522571e-02 1.520411e-02
[411] 1.511467e-02 1.514360e-02 1.514569e-02 1.518495e-02 1.510408e-02
[416] 1.510206e-02 1.509341e-02 1.511358e-02 1.508592e-02 1.501388e-02
[421] 1.505540e-02 1.505502e-02 1.506741e-02 1.509686e-02 1.515213e-02
[426] 1.515769e-02 1.516553e-02 1.518646e-02 1.516485e-02 1.517324e-02
[431] 1.516560e-02 1.512571e-02 1.517651e-02 1.517509e-02 1.513173e-02
[436] 1.516301e-02 1.517132e-02 1.516650e-02 1.518871e-02 1.521158e-02
[441] 1.522765e-02 1.523785e-02 1.524367e-02 1.529882e-02 1.532352e-02
[446] 1.527501e-02 1.528492e-02 1.530212e-02 1.530428e-02 1.530291e-02
[451] 1.533005e-02 1.534294e-02 1.534086e-02 1.533740e-02 1.537216e-02
[456] 1.538494e-02 1.538761e-02 1.536780e-02 1.536314e-02 1.539315e-02
[461] 1.539752e-02 1.543399e-02 1.545616e-02 1.540453e-02 1.546495e-02
[466] 1.544540e-02 1.544857e-02 1.548733e-02 1.550875e-02 1.552070e-02
[471] 1.556171e-02 1.559205e-02 1.559572e-02 1.561561e-02 1.563191e-02
[476] 1.563004e-02 1.563467e-02 1.563759e-02 1.566336e-02 1.565072e-02
[481] 1.566669e-02 1.566396e-02 1.565513e-02 1.566695e-02 1.561759e-02
[486] 1.562871e-02 1.566529e-02 1.570026e-02 1.562194e-02 1.564811e-02
[491] 1.562448e-02 1.562085e-02 1.563478e-02 1.560871e-02 1.558675e-02
[496] 1.560299e-02 1.557734e-02 1.559865e-02 1.560246e-02 1.559753e-02
```

In [108...

```
varImpPlot(rfmodel)
```

[illegible]

```
Out[108... RObject{RealSxp}
              IncNodePurity
  Atr1    1.250515e-02
  Atr2    6.921331e-02
  Atr3    6.295343e-02
  Atr4    1.961538e-03
  Atr5    1.315906e-01
  Atr6    4.187761e-16
  Atr7    1.971831e-03
  Atr8    7.072586e-02
  Atr9    2.715662e+00
  Atr10   7.818750e-03
  Atr11   9.768458e-01
  Atr12   4.104154e-01
  Atr13   2.114477e-02
  Atr14   7.672555e-03
  Atr15   2.566604e-01
  Atr16   6.858092e-01
  Atr17   3.341735e+00
  Atr18   2.394504e+00
  Atr19   1.394640e+00
  Atr20   3.447513e+00
  Atr21   2.939288e-01
```

```

Atr22  5.828736e-03
Atr23  8.348877e-17
Atr24  3.928366e-03
Atr25  8.551768e-01
Atr26  1.335728e+00
Atr27  9.136333e-01
Atr28  4.885256e-02
Atr29  3.417709e-01
Atr30  1.145750e-16
Atr31  6.657570e-02
Atr32  1.187002e-02
Atr33  9.782229e-03
Atr34  1.129025e-01
Atr35  6.998582e-01
Atr36  3.060676e+00
Atr37  6.594643e-02
Atr38  1.967742e-03
Atr39  2.287625e-01
Atr40  4.142813e+00
Atr41  9.582542e-01
Atr42  1.966102e-03
Atr43  6.417089e-16
Atr44  1.957975e-01
Atr45  2.609024e-15
Atr46  3.656186e-15
Atr47  1.963636e-03
Atr48  5.195844e-16
Atr49  1.312259e-02
Atr50  6.117213e-03
Atr51  5.827370e-03
Atr52  2.114982e-02
Atr53  4.495968e-03
Atr54  3.679198e-03

```

```
In [109... bagmodel=randomForest(m , data = traindf,mtry=10)
```

```

└ Warning: RCall.jl: Warning in randomForest.default(m, y, ...) :
  └ The response has five or fewer unique values. Are you sure you want to do regression?
└ @ RCall C:\Users\ASUS\julia\packages\RCall\6kphM\src\io.jl:172
RObject{VecSxp}

```

```
Out[109...
```

```
Call:
```

```

randomForest(formula = Class ~ Atr1 + Atr2 + Atr3 + Atr4 + Atr5 +      Atr6 + Atr7 + Atr8 + Atr9 + Atr10 + Atr11 + Atr12
+ Atr13 +      Atr14 + Atr15 + Atr16 + Atr17 + Atr18 + Atr19 + Atr20 + Atr21 +      Atr22 + Atr23 + Atr24 + Atr25 + Atr26

```

```

+ Atr27 + Atr28 + Atr29 + Atr30 + Atr31 + Atr32 + Atr33 + Atr34 + Atr35 + Atr36 + Atr37 + Atr38 + Atr39 + Atr40
+ Atr41 + Atr42 + Atr43 + Atr44 + Atr45 + Atr46 + Atr47 + Atr48 + Atr49 + Atr50 + Atr51 + Atr52 + Atr53 + Atr5
4, data = structure(list(Atr1 = c(3L, 4L, 3L, 4L, 0L, 1L, 3L, 3L, 2L, 3L, 3L, 3L, 0L, 3L, 4L, 0L, 3L, 0L, 0L, 3L, 0L, 0
L, 0L, 3L, 3L, 3L, 1L, 3L, 0L, 4L, 1L, 0L, 0L, 2L, 4L, 0L, 0L, 3L, 2L, 3L, 0L, 0L, 0L, 0L, 4L, 2L, 3L, 3L, 0L, 0L, 4L, 3
L, 1L, 3L, 0L, 0L, 2L, 0L, 3L, 4L, 4L, 4L, 0L, 0L, 3L, 4L, 1L, 0L, 4L, 0L, 0L, 4L, 0L, 2L, 2L, 3L, 3L, 3L, 2L, 0L, 3L,
0L, 0L, 4L, 0L, 3L, 0L, 1L, 1L, 0L, 0L, 0L, 0L, 3L, 0L, 0L, 3L, 0L, 0L, 0L, 0L, 3L, 0L, 4L, 0L, 4L, 2L, 4L, 3L, 4L, 0L,
0L, 3L, 3L, 3L, 0L, 3L, 3L, 4L), Atr2 = c(2L, 2L, 4L, 4L, 1L, 0L, 3L, 4L, 2L, 4L, 1L, 4L, 0L, 2L, 3L, 0L, 3L, 0L, 1L, 3
L, 0L, 0L, 0L, 0L, 3L, 3L, 1L, 2L, 0L, 4L, 0L, 1L, 0L, 2L, 3L, 0L, 0L, 3L, 0L, 3L, 0L, 0L, 0L, 0L, 4L, 0L, 4L, 3L, 0L,
0L, 3L, 3L, 2L, 4L, 0L, 0L, 1L, 2L, 2L, 3L, 2L, 3L, 0L, 0L, 2L, 2L, 1L, 0L, 4L, 2L, 1L, 4L, 0L, 2L, 1L, 3L, 4L, 3L, 2L,
0L, 4L, 0L, 1L, 4L, 0L, 3L, 0L, 0L, 1L, 0L, 0L, 0L, 0L, 3L, 0L, 0L, 3L, 1L, 0L, 0L, 1L, 3L, 0L, 3L, 0L, 3L, 2L, 3L, 2L,
4L, 1L, 0L, 3L, 3L, 2L, 0L, 4L, 3L, 4L), Atr3 = c(2L, 3L, 2L, 4L, 0L, 1L, 2L, 3L, 4L, 3L, 1L, 3L, 2L, 3L, 2L, 0L, 4L, 0
L, 0L, 3L, 0L, 0L, 0L, 2L, 3L, 0L, 4L, 0L, 4L, 1L, 1L, 0L, 2L, 2L, 0L, 0L, 3L, 2L, 2L, 1L, 0L, 0L, 0L, 3L, 1L, 4L, 2
L, 1L, 0L, 3L, 3L, 0L, 3L, 1L, 0L, 0L, 2L, 4L, 3L, 3L, 4L, 0L, 0L, 3L, 3L, 2L, 0L, 4L, 0L, 2L, 4L, 0L, 1L, 1L, 3L, 3L,
3L, 3L, 0L, 3L, 0L, 0L, 4L, 0L, 3L, 0L, 0L, 1L, 0L, 1L, 0L, 0L, 3L, 0L, 0L, 4L, 0L, 0L, 0L, 1L, 3L, 0L, 3L, 0L, 3L, 3L,
4L, 3L, 4L, 0L, 1L, 3L, 3L, 3L, 1L, 3L, 4L, 3L), Atr4 = c(3L, 2L, 3L, 3L, 0L, 0L, 3L, 4L, 1L, 4L, 0L, 3L, 4L, 2L, 3L, 0
L, 4L, 0L, 1L, 4L, 0L, 0L, 0L, 0L, 3L, 2L, 0L, 3L, 0L, 0L, 0L, 3L, 3L, 0L, 0L, 2L, 4L, 3L, 0L, 0L, 0L, 0L, 3L, 1L, 4L, 2
L, 4L, 3L, 0L, 0L, 3L, 2L, 0L, 4L, 1L, 0L, 2L, 1L, 0L, 4L, 4L, 4L, 0L, 0L, 2L, 2L, 0L, 0L, 3L, 0L, 1L, 3L, 0L, 0L, 0L,
2L, 4L, 4L, 2L, 0L, 2L, 0L, 0L, 4L, 0L, 2L, 0L, 0L, 1L, 0L, 0L, 0L, 0L, 3L, 0L, 0L, 3L, 1L, 0L, 0L, 0L, 4L, 0L, 3L, 0L,
3L, 0L, 3L, 4L, 3L, 0L, 0L, 3L, 2L, 2L, 0L, 4L, 4L, 2L ), Atr5 = c(3L, 4L, 3L, 4L, 0L, 0L, 3L, 3L, 0L, 3L, 0L, 3L, 0L, 3
L, 4L, 0L, 3L, 0L, 0L, 3L, 0L, 0L, 0L, 3L, 3L, 0L, 3L, 0L, 4L, 1L, 0L, 0L, 2L, 4L, 0L, 0L, 3L, 2L, 3L, 1L, 0L, 0L, 0
L, 4L, 0L, 3L, 3L, 0L, 0L, 4L, 3L, 0L, 3L, 0L, 0L, 0L, 0L, 3L, 4L, 4L, 0L, 0L, 0L, 3L, 4L, 2L, 0L, 4L, 0L, 0L, 4L, 0L,
0L, 0L, 3L, 3L, 3L, 2L, 0L, 3L, 0L, 0L, 4L, 0L, 1L, 0L, 0L, 1L, 0L, 0L, 0L, 1L, 3L, 0L, 0L, 3L, 0L, 0L, 0L, 0L, 3L, 0L,
4L, 0L, 4L, 0L, 4L, 3L, 4L, 0L, 0L, 3L, 3L, 3L, 0L, 3L, 3L, 4L), Atr6 = c(2L, 2L, 0L, 2L, 0L, 0L, 1L, 0L, 0L, 0L, 0L, 0
L, 0L, 2L, 1L, 0L, 1L, 0L, 0L, 1L, 2L, 2L, 0L, 0L, 1L, 1L, 0L, 2L, 0L, 0L, 1L, 1L, 0L, 3L, 1L, 2L, 0L, 1L, 2L, 1L, 0L,
0L, 1L, 0L, 0L, 0L, 0L, 1L, 0L, 0L, 0L, 0L, 0L, 2L, 1L, 2L, 0L, 0L, 0L, 0L, 2L, 1L, 0L, 0L, 1L, 0L, 0L, 1L, 0L,
0L, 0L, 4L, 2L, 1L, 0L, 1L, 2L, 1L, 0L, 0L, 1L, 0L, 0L, 3L, 1L, 1L, 0L, 0L, 0L, 0L, 0L, 1L, 1L, 0L, 1L, 0L, 2L, 0L, 2L,
1L, 0L, 1L, 0L, 1L, 1L, 1L, 2L, 0L, 0L, 2L, 1L, 1L, 2L, 0L, 0L, 1L, 0L), Atr7 = c(1L, 0L, 1L, 4L, 0L, 0L, 1L, 1L, 0L, 1
L, 0L, 1L, 0L, 1L, 0L, 0L, 1L, 1L, 0L, 0L, 0L, 1L, 1L, 0L, 3L, 0L, 0L, 0L, 0L, 0L, 2L, 0L, 0L, 0L, 1L, 4L, 1
L, 0L, 0L, 0L, 0L, 4L, 0L, 1L, 1L, 0L, 0L, 0L, 1L, 0L, 1L, 0L, 0L, 0L, 0L, 3L, 0L, 0L, 0L, 0L, 0L, 1L, 0L, 0L, 0L, 0L,
0L, 0L, 0L, 0L, 1L, 0L, 1L, 1L, 1L, 2L, 0L, 1L, 0L, 0L, 0L, 0L, 4L, 0L, 0L, 0L, 0L, 0L, 0L, 0L, 1L, 0L, 0L, 1L, 0L, 0L,
0L, 0L, 1L, 0L, 0L, 0L, 0L, 0L, 0L, 1L, 0L, 0L, 0L, 1L, 1L, 1L, 0L, 1L, 1L, 0L), Atr8 = c(2L, 2L, 4L, 4L, 0L, 1L, 3L, 4
L, 0L, 4L, 0L, 4L, 0L, 2L, 3L, 0L, 3L, 0L, 0L, 3L, 0L, 0L, 0L, 0L, 3L, 3L, 0L, 2L, 0L, 4L, 0L, 0L, 0L, 1L, 3L, 0L, 0L,
3L, 3L, 3L, 0L, 0L, 0L, 0L, 2L, 0L, 4L, 3L, 0L, 0L, 3L, 3L, 0L, 4L, 0L, 0L, 0L, 0L, 2L, 3L, 2L, 0L, 0L, 0L, 2L, 2L, 2L,
0L, 4L, 0L, 0L, 4L, 0L, 3L, 0L, 3L, 4L, 3L, 2L, 0L, 4L, 0L, 2L, 4L, 0L, 3L, 0L, 0L, 1L, 0L, 1L, 0L, 0L, 3L, 1L, 0L, 3L,
0L, 0L, 1L, 0L, 3L, 0L, 3L, 0L, 3L, 0L, 3L, 2L, 4L, 1L, 0L, 3L, 3L, 2L, 0L, 4L, 3L, 4L), Atr9 = c(2L, 3L, 2L, 4L, 0L, 0
L, 3L, 3L, 0L, 3L, 0L, 3L, 0L, 3L, 2L, 0L, 4L, 0L, 0L, 3L, 0L, 0L, 0L, 0L, 2L, 3L, 0L
```



```

0L, 0L, 0L, 3L, 4L, 1L, 0L, 4L, 0L, 0L, 4L, 0L, 3L, 0L, 3L, 3L, 3L, 2L, 1L, 3L, 0L, 1L, 4L, 0L, 2L, 0L, 1L, 1L, 0L, 0L,
0L, 0L, 3L, 0L, 0L, 3L, 1L, 0L, 0L, 0L, 3L, 0L, 4L, 0L, 4L, 1L, 4L, 3L, 4L, 0L, 0L, 3L, 3L, 3L, 0L, 3L, 3L, 4L), Atr12
= c(2L, 2L, 4L, 4L, 0L, 0L, 3L, 4L, 0L, 4L, 0L, 4L, 0L, 2L, 3L, 0L, 3L, 1L, 0L, 3L, 0L, 0L, 0L, 1L, 3L, 3L, 0L, 3L, 0L,
4L, 0L, 0L, 1L, 2L, 3L, 0L, 0L, 3L, 3L, 3L, 1L, 1L, 0L, 0L, 4L, 0L, 4L, 3L, 0L, 1L, 3L, 3L, 0L, 4L, 2L, 0L, 1L, 1L, 3L,
3L, 2L, 0L, 0L, 0L, 2L, 2L, 1L, 0L, 4L, 0L, 1L, 4L, 0L, 3L, 1L, 3L, 4L, 3L, 2L, 1L, 4L, 1L, 0L, 3L, 0L, 2L, 1L, 0L, 0L,
0L, 0L, 1L, 0L, 3L, 0L, 0L, 3L, 2L, 0L, 1L, 2L, 3L, 0L, 3L, 0L, 3L, 1L, 3L, 2L, 4L, 0L, 2L, 3L, 3L, 2L, 0L, 4L, 3L, 4L),
Atr13 = c(3L, 4L, 3L, 4L, 0L, 1L, 3L, 3L, 1L, 3L, 0L, 3L, 0L, 3L, 4L, 1L, 3L, 1L, 0L, 3L, 0L, 0L, 0L, 1L, 3L, 3L, 1
L, 2L, 0L, 4L, 1L, 0L, 0L, 2L, 4L, 1L, 0L, 3L, 3L, 3L, 0L, 2L, 1L, 0L, 4L, 1L, 3L, 3L, 0L, 0L, 4L, 3L, 0L, 3L,
1L, 0L, 1L, 1L, 2L, 4L, 4L, 0L, 0L, 1L, 3L, 4L, 2L, 0L, 4L, 0L, 1L, 4L, 0L, 3L, 1L, 3L, 3L, 3L, 2L, 1L, 3L, 1L,
1L, 4L, 2L, 2L, 1L, 1L, 0L, 0L, 0L, 0L, 1L, 3L, 1L, 0L, 3L, 2L, 0L, 1L, 1L, 3L, 0L, 4L, 0L, 4L, 1L, 4L, 3L, 4L,
0L, 1L, 3L, 3L, 3L, 0L, 3L, 3L, 4L), Atr14 = c(2L, 2L, 4L, 4L, 0L, 0L, 3L, 4L, 1L, 4L, 1L, 4L, 0L, 2L, 3L, 1L,
3L, 0L, 1L, 3L, 0L, 0L, 0L, 2L, 3L, 3L, 0L, 2L, 0L, 4L, 1L, 0L, 0L, 2L, 3L, 1L, 0L, 3L, 0L, 3L, 0L, 0L, 0L, 0L,
4L, 0L, 4L, 3L, 0L, 0L, 3L, 3L, 1L, 4L, 1L, 0L, 0L, 0L, 3L, 3L, 2L, 0L, 0L, 0L, 2L, 2L, 0L, 0L, 4L, 0L, 0L, 4L,
0L, 3L, 1L, 3L, 4L, 3L, 2L, 0L, 4L, 1L, 0L, 0L, 0L, 3L, 0L, 1L, 0L, 0L, 1L, 0L, 0L, 3L, 1L, 0L, 3L, 0L, 0L, 1L,
0L, 3L, 0L, 3L, 0L, 3L, 1L, 3L, 2L, 4L, 1L, 0L, 3L, 3L, 2L, 0L, 4L, 3L, 4L), Atr15 = c(2L, 3L, 2L, 4L, 0L, 0L, 3L, 3
L, 0L, 3L, 1L, 3L, 0L, 3L, 2L, 0L, 4L, 0L, 0L, 3L, 0L, 1L, 0L, 1L, 2L, 3L, 0L, 1L, 0L, 4L, 1L, 0L, 0L, 3L, 2L,
0L, 0L, 3L, 4L, 2L, 0L, 1L, 0L, 0L, 4L, 2L, 4L, 2L, 0L, 0L, 3L, 3L, 0L, 3L, 1L, 0L, 1L, 0L, 1L, 3L, 3L, 0L, 0L,
0L, 3L, 3L, 2L, 0L, 4L, 0L, 0L, 4L, 0L, 3L, 1L, 3L, 3L, 3L, 3L, 0L, 3L, 1L, 0L, 4L, 0L, 2L, 0L, 1L, 0L, 0L, 0L,
0L, 1L, 3L, 0L, 0L, 4L, 0L, 0L, 0L, 0L, 3L, 0L, 3L, 0L, 3L, 1L, 4L, 3L, 4L, 0L, 2L, 3L, 3L, 3L, 0L, 3L, 4L, 3
L), Atr16 = c(3L, 2L, 3L, 4L, 0L, 1L, 3L, 4L, 1L, 4L, 1L, 3L, 0L, 2L, 3L, 0L, 4L, 0L, 0L, 4L, 0L, 0L, 0L, 0L, 3
L, 2L, 0L, 3L, 0L, 3L, 0L, 0L, 0L, 3L, 3L, 0L, 0L, 2L, 4L, 3L, 0L, 0L, 0L, 4L, 2L, 4L, 3L, 0L, 0L, 3L, 2L,
0L, 4L, 0L, 0L, 0L, 0L, 3L, 4L, 4L, 0L, 0L, 0L, 2L, 2L, 1L, 0L, 3L, 0L, 0L, 3L, 0L, 0L, 3L, 0L, 0L, 0L, 0L, 3L,
0L, 0L, 0L, 0L, 3L, 1L, 1L, 0L, 0L, 0L, 0L, 0L, 0L, 3L, 0L, 0L, 3L, 0L, 0L, 3L, 0L, 0L, 0L, 4L, 0L, 3L, 0L, 3L,
1L, 3L, 4L, 3L, 0L, 0L, 3L, 2L, 2L, 0L, 4L, 4L, 2L), Atr17 = c(3L, 4L, 3L, 4L, 1L, 0L, 3L, 3L, 0L, 3L, 0L, 3L, 0L,
3L, 0L, 0L, 3L, 0L, 1L, 0L, 0L, 3L, 3L, 0L, 3L, 0L, 4L, 0L, 0L, 0L, 3L, 4L, 0L, 0L, 0L, 0L, 3L, 4L, 0L, 0L, 0L,
0L, 0L, 0L, 3L, 0L, 1L, 0L, 0L, 0L, 3L, 3L, 0L, 3L, 0L, 4L, 0L, 0L, 0L, 3L, 4L, 0L, 0L, 0L, 0L, 0L, 0L, 0L,
4L, 1L, 3L, 3L, 0L, 1L, 4L, 3L, 0L, 3L, 0L, 0L, 1L, 0L, 3L, 4L, 4L, 0L, 0L, 0L, 3L, 4L, 2L, 0L, 4L, 0L, 1L, 4L,
0L, 3L, 0L, 3L, 3L, 3L, 2L, 0L, 3L, 0L, 0L, 4L, 0L, 3L, 0L, 0L, 0L, 0L, 0L, 0L, 0L, 0L, 0L, 0L, 3L, 0L, 0L, 3L, 0L, 0L, 0L,
0L, 3L, 0L, 4L, 0L, 4L, 1L, 4L, 3L, 4L, 0L, 2L, 3L, 3L, 3L, 0L, 3L, 3L, 4L), Atr18 = c(2L, 2L, 4L, 4L, 0L, 0L,
3L, 4L, 0L, 4L, 0L, 4L, 0L, 2L, 3L, 0L, 3L, 0L, 0L, 3L, 0L, 0L, 0L, 1L, 3L, 3L, 0L, 3L, 0L, 4L, 1L, 0L, 0L, 3L,
3L, 0L, 0L, 3L, 3L, 3L, 0L, 0L, 0L, 0L, 4L, 0L, 4L, 3L, 0L, 0L, 3L, 3L, 0L, 4L, 0L, 0L, 1L, 0L, 3L, 3L, 2L, 0L,
0L, 0L, 2L, 2L, 1L, 0L, 4L, 0L, 0L, 4L, 0L, 3L, 0L, 3L, 4L, 3L, 2L, 0L, 4L, 0L, 0L, 4L, 0L, 3L, 0L, 1L, 0L, 0L, 0L,
0L, 0L, 3L, 0L, 0L, 3L, 0L, 0L, 0L, 0L, 3L, 0L, 3L, 0L, 3L, 1L, 3L, 2L, 4L, 0L, 1L, 3L, 3L, 2L, 1L, 4L, 3L, 4
L), Atr19 = c(3L, 4L, 3L, 4L, 0L, 1L, 4L, 3L, 0L, 3L, 0L, 3L, 2L, 3L, 4L, 0L, 3L, 0L, 0L, 3L, 0L, 1L, 0L, 1L, 3
L, 3L, 0L, 3L, 0L, 4L, 0L, 0L, 0L, 3L, 4L, 0L, 0L, 3L, 3L, 3L, 0L, 0L, 0L, 0L, 4L, 0L, 3L, 3L, 0L, 0L, 4L, 3L,
1L, 3L, 0L, 0L, 0L, 0L, 3L, 4L, 4L, 0L, 0L, 0L, 3L, 4L, 0L, 0L, 4L, 0L, 1L, 4L, 0L, 3L, 0L, 3L, 3L, 3L, 2L, 1L,
3L, 0L, 0L, 3L, 0L, 3L, 0L, 0L, 0L, 0L, 0L, 0L, 0L, 0L, 0L, 0L, 3L, 0L, 4L, 0L, 4L, 1L, 4L, 0L, 4L,
3L, 4L, 0L, 0L, 3L, 3L, 3L, 0L, 3L, 3L, 4L), Atr20 = c(2L, 2L, 4L, 4L, 0L, 0L, 3L, 4L, 1L, 4L, 0L, 4L, 0L, 2L, 3L, 0
L, 3L, 0L, 0L, 3L, 0L, 0L, 0L, 0L, 3L, 3L, 0L, 4L, 0L, 4L, 1L, 0L, 0L, 2L, 3L, 0L, 0L, 3L, 2L, 3L, 0L, 0L, 0L,
0L, 4L, 0L, 4L, 3L, 0L, 0L, 3L, 3L, 0L, 4L, 0L, 0L, 0L, 0L, 4L, 3L, 2L, 0L, 0L, 0L, 2L, 2L, 0L, 0L, 4L, 0L, 0L,
4L, 0L, 3L, 0L, 3L, 4L, 3L, 2L, 0L, 4L, 0L, 0L, 2L, 0L, 2L, 0L, 0L, 0L, 0L, 0L, 0L, 0L, 0L, 0L, 0L, 0L, 0L,
1L, 0L, 3L, 0L, 3L, 0L, 3L, 1L, 3L, 2L, 4L, 0L, 1L, 3L, 3L, 2L, 0L, 4L, 3L, 4L), Atr21 = c(2L, 3L, 2L, 4L, 0L,
0L, 3L, 3L, 0L, 3L, 0L, 3L, 0L, 3L, 2L, 1L, 4L, 1L, 0L, 3L, 0L, 0L, 0L, 0L, 2L, 3L, 1L, 2L, 0L, 4L, 0L, 0L, 0L,
2L, 2L, 0L, 0L, 3L, 2L, 2L, 0L, 0L, 0L, 0L, 0L, 1L, 4L, 2L, 0L, 1L, 3L, 3L, 0L, 3L, 0L, 0L, 0L, 0L, 2L, 3L, 3L,
0L, 0L, 0L, 3L, 3L, 0L, 0L, 4L, 0L, 0L, 4L, 0L, 2L, 0L, 3L, 3L, 3L, 1L, 3L, 1L, 0L, 1L, 0L, 3L, 0L, 0L, 0L, 0L,
0L, 0L, 0L, 3L, 1L, 0L, 4L, 0L, 0L, 1L, 0L, 3L, 0L, 3L, 0L, 3L, 1L, 4L, 3L, 4L, 0L, 0L, 3L, 3L, 3L, 0L, 3L, 4L,

```

```

3L), Atr22 = c(3L,      2L, 3L, 3L, 0L, 0L, 3L, 4L, 0L, 4L, 0L, 3L, 0L, 2L, 3L, 0L,      4L, 0L, 0L, 4L, 0L, 0L, 0L, 0L,
3L, 2L, 0L, 1L, 0L, 3L, 0L,      0L, 0L, 1L, 3L, 0L, 0L, 2L, 1L, 3L, 0L, 0L, 0L, 0L, 0L,      4L, 3L, 0L, 0L, 3L, 2L,
1L, 4L, 0L, 0L, 0L, 0L, 1L, 4L, 4L,      0L, 0L, 0L, 2L, 2L, 1L, 0L, 3L, 0L, 0L, 3L, 0L, 2L, 0L, 2L,      4L, 4L, 2L, 0L,
2L, 1L, 0L, 1L, 0L, 3L, 0L, 0L, 0L, 0L, 0L,      0L, 0L, 3L, 1L, 0L, 3L, 0L, 0L, 0L, 0L, 4L, 0L, 3L, 0L, 3L,      0L, 3L,
4L, 3L, 0L, 0L, 3L, 2L, 2L, 0L, 4L, 4L, 2L), Atr23 = c(3L,      4L, 3L, 0L, 0L, 0L, 3L, 3L, 0L, 3L, 0L, 3L, 0L, 3L, 4L, 0
L,      3L, 0L, 0L, 3L, 0L, 0L, 0L, 0L, 3L, 3L, 0L, 0L, 0L, 4L, 0L,      0L, 0L, 1L, 4L, 0L, 0L, 3L, 2L, 3L, 0L, 0L, 0L,
0L, 2L, 0L,      3L, 3L, 0L, 0L, 4L, 3L, 1L, 3L, 0L, 0L, 0L, 0L, 0L, 4L, 4L,      0L, 0L, 0L, 3L, 4L, 0L, 0L, 4L, 0L, 0L,
4L, 0L, 2L, 0L, 3L,      3L, 3L, 2L, 0L, 3L, 0L, 2L, 0L, 0L, 3L, 0L, 0L, 0L, 0L,      0L, 0L, 3L, 1L, 0L, 3L, 0L, 0L,
1L, 0L, 3L, 0L, 4L, 0L, 4L,      0L, 4L, 3L, 4L, 0L, 0L, 3L, 3L, 3L, 2L, 3L, 3L, 4L), Atr24 = c(2L,      2L, 4L, 4L, 0L,
0L, 4L, 4L, 0L, 4L, 0L, 4L, 0L, 2L, 3L, 0L,      3L, 1L, 1L, 3L, 0L, 0L, 0L, 1L, 3L, 3L, 0L, 2L, 0L, 4L, 0L,      0L, 0L,
3L, 3L, 0L, 0L, 3L, 0L, 3L, 0L, 0L, 0L, 0L, 4L, 1L,      4L, 3L, 2L, 0L, 3L, 3L, 1L, 4L, 0L, 1L, 1L, 0L, 2L, 3L, 2L,
2L, 1L, 0L, 2L, 2L, 1L, 0L, 4L, 0L, 0L, 4L, 0L, 3L, 0L, 3L,      4L, 3L, 2L, 0L, 4L, 0L, 2L, 2L, 0L, 3L, 0L, 1L, 0L, 0L,
0L,      0L, 0L, 3L, 0L, 0L, 3L, 0L, 2L, 1L, 0L, 3L, 0L, 3L, 0L, 3L,      0L, 3L, 2L, 4L, 0L, 0L, 3L, 3L, 2L, 2L, 4L, 3L,
4L), Atr25 = c(3L,      4L, 3L, 0L, 0L, 0L, 3L, 3L, 0L, 3L, 0L, 3L, 0L, 3L, 4L, 1L,      3L, 1L, 0L, 3L, 0L, 0L, 0L, 0L,
3L, 3L, 1L, 2L, 0L, 4L, 0L,      0L, 0L, 3L, 4L, 1L, 0L, 3L, 0L, 3L, 0L, 0L, 0L, 4L, 0L,      3L, 3L, 0L, 0L, 4L, 3L,
1L, 3L, 1L, 1L, 1L, 1L, 2L, 4L, 4L,      0L, 1L, 0L, 3L, 4L, 1L, 0L, 4L, 1L, 0L, 4L, 1L, 2L, 1L, 3L,      3L, 3L, 2L, 0L,
3L, 0L, 0L, 2L, 0L, 2L, 2L, 1L, 0L, 0L, 0L,      0L, 1L, 3L, 1L, 0L, 3L, 0L, 0L, 0L, 1L, 3L, 1L, 4L, 0L, 4L,      0L, 4L,
3L, 4L, 1L, 2L, 3L, 3L, 3L, 1L, 3L, 3L, 4L), Atr26 = c(2L,      2L, 4L, 4L, 0L, 0L, 3L, 4L, 0L, 4L, 0L, 4L, 0L, 2L, 3L, 0
L,      3L, 0L, 0L, 3L, 0L, 0L, 0L, 0L, 3L, 3L, 0L, 2L, 0L, 4L, 0L,      0L, 0L, 3L, 3L, 1L, 0L, 3L, 0L, 3L, 0L, 0L, 0L,
0L, 4L, 1L,      4L, 3L, 0L, 0L, 3L, 3L, 1L, 4L, 0L, 0L, 1L, 0L, 2L, 3L, 2L,      0L, 1L, 0L, 2L, 2L, 0L, 0L, 4L, 1L, 0L,
4L, 1L, 3L, 1L, 3L,      4L, 3L, 2L, 0L, 4L, 0L, 0L, 1L, 0L, 3L, 1L, 0L, 0L, 0L,      0L, 1L, 3L, 0L, 0L, 3L, 1L, 0L,
0L, 0L, 3L, 1L, 3L, 0L, 3L,      0L, 3L, 2L, 4L, 1L, 2L, 3L, 3L, 2L, 1L, 4L, 3L, 4L), Atr27 = c(2L,      3L, 2L, 4L, 0L,
0L, 2L, 3L, 0L, 3L, 0L, 3L, 2L, 0L,      4L, 0L, 0L, 3L, 0L, 0L, 0L, 1L, 2L, 3L, 1L, 2L, 0L, 4L, 0L,      0L, 0L,
2L, 2L, 1L, 0L, 3L, 0L, 2L, 0L, 0L, 0L, 0L, 4L, 0L,      4L, 2L, 0L, 0L, 3L, 3L, 0L, 3L, 0L, 0L, 0L, 0L, 2L, 3L, 3L,
0L, 0L, 0L, 3L, 3L, 1L, 0L, 4L, 0L, 0L, 4L, 0L, 2L, 0L, 3L,      3L, 3L, 3L, 0L, 3L, 0L, 1L, 2L, 0L, 3L, 1L, 0L, 0L, 0L,
0L,      0L, 0L, 3L, 0L, 0L, 4L, 1L, 0L, 0L, 0L, 3L, 1L, 3L, 0L, 3L,      0L, 4L, 3L, 4L, 1L, 0L, 3L, 3L, 3L, 1L, 3L, 4L,
3L), Atr28 = c(3L,      2L, 3L, 0L, 0L, 0L, 4L, 4L, 0L, 4L, 0L, 3L, 0L, 2L, 3L, 1L,      4L, 0L, 0L, 4L, 0L, 0L, 0L, 0L,
3L, 2L, 1L, 2L, 0L, 3L, 0L,      0L, 0L, 3L, 3L, 0L, 0L, 2L, 0L, 3L, 0L, 0L, 0L, 1L, 0L,      4L, 3L, 0L, 0L, 3L, 2L,
0L, 4L, 0L, 0L, 0L, 0L, 2L, 4L, 4L,      0L, 0L, 0L, 2L, 2L, 0L, 0L, 3L, 0L, 0L, 3L, 0L, 2L,      4L, 4L, 2L, 0L,
2L, 1L, 1L, 0L, 0L, 2L, 0L, 0L, 0L, 0L,      0L, 0L, 3L, 0L, 0L, 3L, 1L, 0L, 0L, 0L, 4L, 1L, 3L, 0L, 3L,      0L, 3L,
4L, 3L, 0L, 0L, 3L, 2L, 2L, 0L, 4L, 4L, 2L), Atr29 = c(3L,      4L, 3L, 4L, 0L, 0L, 3L, 3L, 0L, 3L, 0L, 3L, 0L, 3L, 4L, 1
L,      3L, 1L, 0L, 3L, 0L, 0L, 0L, 0L, 3L, 3L, 1L, 2L, 0L, 4L, 0L,      0L, 0L, 2L, 4L, 0L, 0L, 3L, 2L, 3L, 0L, 0L, 0L,
0L, 1L, 0L,      3L, 3L, 0L, 0L, 4L, 3L, 0L, 3L, 0L, 0L, 0L, 0L, 2L, 4L, 4L,      0L, 0L, 0L, 3L, 4L, 0L, 0L, 4L, 0L, 0L,
4L, 1L, 2L, 0L, 3L,      3L, 3L, 2L, 0L, 3L, 0L, 0L, 1L, 0L, 2L, 0L, 0L, 0L, 0L,      0L, 1L, 3L, 0L, 0L, 3L, 0L, 0L,
0L, 0L, 3L, 1L, 4L, 0L, 4L,      0L, 4L, 3L, 4L, 0L, 0L, 3L, 3L, 3L, 1L, 3L, 3L, 4L), Atr30 = c(2L,      2L, 4L, 4L, 0L,
0L, 4L, 4L, 1L, 4L, 0L, 4L, 0L, 2L, 3L, 1L,      3L, 0L, 0L, 3L, 0L, 1L, 0L, 0L, 3L, 3L, 1L, 3L, 0L, 4L, 0L,      0L, 0L,
2L, 3L, 0L, 0L, 3L, 2L, 3L, 0L, 0L, 0L, 0L, 0L,      4L, 3L, 1L, 0L, 3L, 3L, 0L, 4L, 1L, 0L, 0L, 0L, 3L, 3L, 2L,
0L, 0L, 0L, 2L, 2L, 0L, 0L, 4L, 2L, 0L, 4L, 0L, 3L, 1L, 3L,      4L, 3L, 2L, 0L, 4L, 0L, 0L, 1L, 0L, 2L, 0L, 1L, 0L, 0L,
0L,      0L, 0L, 3L, 1L, 0L, 3L, 0L, 0L, 0L, 1L, 3L, 1L, 3L, 0L, 3L,      0L, 3L, 2L, 4L, 0L, 0L, 3L, 3L, 2L, 0L, 4L, 3L,
4L), Atr31 = c(4L,      4L, 4L, 0L, 0L, 2L, 4L, 3L, 1L, 4L, 0L, 3L, 0L, 4L, 3L, 1L,      4L, 2L, 0L, 3L, 1L, 0L, 0L, 0L,
4L, 4L, 0L, 3L, 4L, 4L, 1L,      0L, 2L, 3L, 4L, 1L, 0L, 4L, 3L, 4L, 2L, 2L, 2L, 0L, 4L, 2L,      4L, 4L, 0L, 0L, 4L, 4L,
2L, 4L, 2L, 0L, 0L, 1L, 3L, 4L, 4L,      0L, 2L, 0L, 4L, 4L, 0L, 0L, 4L, 0L, 0L, 4L, 0L, 1L, 1L, 3L,      4L, 4L, 4L, 0L,
4L, 0L, 1L, 0L, 2L, 1L, 0L, 1L, 0L, 2L, 1L,      0L, 0L, 4L, 1L, 1L, 4L, 2L, 0L, 1L, 0L, 4L, 1L, 4L, 0L, 4L,      2L, 4L,
4L, 4L, 0L, 4L, 3L, 4L, 4L, 0L, 3L, 3L, 4L), Atr32 = c(4L,      4L, 4L, 4L, 0L, 0L, 4L, 3L, 2L, 4L, 0L, 4L, 2L, 4L, 4L, 0
L,      4L, 0L, 1L, 4L, 0L, 1L, 0L, 0L, 3L, 4L, 0L, 3L, 4L, 4L, 1L,      0L, 2L, 3L, 3L, 0L, 0L, 3L, 3L, 4L, 1L, 1L, 0L,

```

```

1L, 2L, 0L,      3L, 4L, 2L, 0L, 3L, 4L, 1L, 4L, 1L, 0L, 0L, 3L, 3L, 3L, 4L,      0L, 1L, 0L, 4L, 4L, 0L, 0L, 3L, 0L, 0L,
3L, 0L, 1L, 1L, 4L,      4L, 4L, 4L, 0L, 4L, 0L, 0L, 4L, 0L, 2L, 0L, 0L, 2L, 1L, 0L,      0L, 0L, 3L, 1L, 1L, 4L, 1L, 0L,
3L, 2L, 4L, 0L, 4L, 0L, 3L,      0L, 3L, 4L, 4L, 0L, 1L, 3L, 4L, 4L, 0L, 3L, 3L, 4L), Atr33 = c(4L,      4L, 4L, 4L, 0L,
0L, 4L, 3L, 1L, 4L, 0L, 3L, 0L, 4L, 3L, 0L,      4L, 0L, 1L, 3L, 0L, 0L, 0L, 4L, 4L, 0L, 3L, 3L, 4L, 1L,      0L, 0L,
3L, 4L, 0L, 1L, 4L, 3L, 4L, 2L, 1L, 0L, 0L, 0L, 0L,      4L, 4L, 0L, 0L, 4L, 4L, 0L, 4L, 0L, 0L, 1L, 0L, 3L, 4L, 4L,
0L, 1L, 0L, 4L, 4L, 0L, 0L, 4L, 0L, 0L, 4L, 0L, 1L, 2L, 3L,      4L, 4L, 4L, 0L, 4L, 0L, 0L, 2L, 0L, 2L, 0L, 0L, 0L, 1L,
0L,      0L, 0L, 4L, 1L, 0L, 4L, 0L, 0L, 0L, 0L, 4L, 0L, 4L, 0L, 4L,      0L, 4L, 4L, 4L, 0L, 1L, 3L, 4L, 4L, 0L, 3L, 3L,
4L), Atr34 = c(4L,      4L, 4L, 0L, 0L, 0L, 4L, 3L, 2L, 4L, 0L, 4L, 1L, 4L, 4L, 0L,      4L, 1L, 1L, 4L, 0L, 1L, 0L, 0L,
3L, 4L, 0L, 3L, 4L, 4L, 1L,      0L, 0L, 3L, 3L, 0L, 1L, 3L, 3L, 4L, 0L, 1L, 1L, 0L, 0L, 0L,      3L, 4L, 1L, 0L, 3L, 4L,
1L, 4L, 2L, 0L, 0L, 0L, 3L, 3L, 4L,      0L, 1L, 1L, 4L, 4L, 1L, 0L, 3L, 1L, 0L, 3L, 0L, 1L, 2L, 4L,      4L, 4L, 4L, 0L,
4L, 1L, 0L, 3L, 0L, 1L, 0L, 0L, 0L, 1L, 1L,      1L, 0L, 3L, 1L, 0L, 4L, 0L, 0L, 0L, 0L, 4L, 0L, 4L, 0L, 3L,      0L, 3L,
4L, 4L, 0L, 1L, 3L, 4L, 4L, 0L, 3L, 3L, 4L), Atr35 = c(4L,      4L, 4L, 1L, 0L, 0L, 4L, 3L, 0L, 4L, 0L, 3L, 0L, 4L, 3L, 0
L,      4L, 0L, 0L, 3L, 0L, 0L, 0L, 0L, 4L, 4L, 0L, 3L, 0L, 4L, 0L,      0L, 0L, 3L, 4L, 0L, 0L, 4L, 3L, 4L, 0L, 0L, 0L,
0L, 0L, 0L,      4L, 4L, 0L, 0L, 4L, 4L, 0L, 4L, 0L, 0L, 1L, 0L, 3L, 4L, 4L,      0L, 0L, 0L, 4L, 4L, 0L, 0L, 4L, 0L, 0L,
4L, 0L, 1L, 0L, 3L,      4L, 4L, 4L, 0L, 4L, 0L, 0L, 0L, 0L, 0L, 0L, 0L, 0L,      0L, 0L, 4L, 0L, 0L, 4L, 0L, 0L,
0L, 0L, 4L, 0L, 4L, 0L, 4L,      0L, 4L, 4L, 4L, 0L, 1L, 3L, 4L, 4L, 0L, 3L, 3L, 4L), Atr36 = c(4L,      4L, 4L, 0L, 0L,
0L, 4L, 3L, 1L, 4L, 0L, 4L, 0L, 4L, 4L, 0L,      4L, 0L, 0L, 4L, 0L, 0L, 0L, 0L, 3L, 4L, 0L, 3L, 0L, 4L, 0L,      0L, 0L,
3L, 3L, 0L, 0L, 3L, 3L, 4L, 0L, 0L, 0L, 0L, 0L, 0L,      3L, 4L, 0L, 0L, 3L, 4L, 0L, 4L, 0L, 0L, 0L, 0L, 3L, 3L, 4L,
0L, 0L, 0L, 4L, 4L, 0L, 0L, 3L, 0L, 0L, 3L, 0L, 1L, 0L, 4L,      4L, 4L, 4L, 0L, 4L, 0L, 0L, 2L, 0L, 2L, 0L, 0L, 0L, 0L,
0L,      0L, 0L, 3L, 0L, 0L, 4L, 0L, 0L, 0L, 0L, 4L, 0L, 4L, 0L, 3L,      0L, 3L, 4L, 4L, 0L, 1L, 3L, 4L, 4L, 0L, 3L, 3L,
4L), Atr37 = c(4L,      4L, 4L, 2L, 0L, 0L, 4L, 3L, 2L, 4L, 0L, 3L, 0L, 4L, 3L, 0L,      4L, 1L, 0L, 3L, 0L, 1L, 0L, 0L,
4L, 4L, 0L, 4L, 4L, 4L, 1L,      1L, 0L, 3L, 4L, 0L, 0L, 4L, 3L, 4L, 2L, 1L, 1L, 0L, 3L, 2L,      4L, 4L, 0L, 0L, 4L, 4L,
1L, 4L, 1L, 0L, 1L, 2L, 4L, 4L, 4L,      0L, 2L, 0L, 4L, 4L, 0L, 0L, 4L, 1L, 0L, 4L, 0L, 1L, 0L, 3L,      4L, 4L, 4L, 0L,
4L, 1L, 0L, 3L, 4L, 3L, 0L, 0L, 0L, 0L, 0L,      1L, 0L, 4L, 1L, 0L, 4L, 1L, 1L, 0L, 0L, 4L, 0L, 4L, 2L, 4L,      0L, 4L,
4L, 4L, 0L, 1L, 3L, 4L, 4L, 0L, 3L, 3L, 4L), Atr38 = c(4L,      4L, 4L, 0L, 0L, 1L, 4L, 3L, 1L, 4L, 3L, 4L, 2L, 4L, 4L, 0
L,      4L, 0L, 2L, 4L, 0L, 1L, 0L, 0L, 3L, 4L, 0L, 4L, 0L, 4L, 0L,      0L, 0L, 3L, 3L, 0L, 1L, 3L, 3L, 4L, 0L, 1L, 0L,
0L, 0L, 2L,      3L, 4L, 0L, 0L, 4L, 4L, 0L, 4L, 1L, 0L, 0L, 1L, 4L, 3L, 4L,      0L, 0L, 0L, 4L, 4L, 0L, 0L, 3L, 1L, 0L,
3L, 0L, 2L, 0L, 4L,      4L, 4L, 4L, 0L, 4L, 0L, 0L, 4L, 0L, 2L, 0L, 0L, 0L, 1L, 0L,      0L, 0L, 3L, 0L, 0L, 4L, 1L, 0L,
0L, 0L, 4L, 0L, 4L, 1L, 3L,      0L, 3L, 4L, 4L, 0L, 2L, 3L, 4L, 4L, 0L, 3L, 3L, 4L), Atr39 = c(4L,      4L, 4L, 4L, 0L,
0L, 4L, 3L, 3L, 4L, 0L, 3L, 0L, 4L, 3L, 0L,      4L, 2L, 1L, 3L, 0L, 1L, 0L, 1L, 4L, 4L, 0L, 4L, 1L, 4L, 1L,      0L, 0L,
4L, 4L, 0L, 1L, 4L, 1L, 4L, 1L, 2L, 0L, 0L, 4L, 1L,      4L, 4L, 0L, 0L, 3L, 4L, 0L, 4L, 1L, 0L, 1L, 0L, 4L, 4L, 4L,
0L, 1L, 0L, 4L, 4L, 0L, 0L, 4L, 2L, 1L, 4L, 0L, 2L, 0L, 3L,      4L, 4L, 4L, 0L, 4L, 1L, 0L, 2L, 0L, 2L, 0L, 0L, 0L, 1L,
0L,      4L, 0L, 4L, 0L, 0L, 4L, 2L, 0L, 0L, 0L, 4L, 0L, 4L, 1L, 4L,      0L, 4L, 4L, 4L, 0L, 0L, 3L, 4L, 4L, 0L, 3L, 3L,
4L), Atr40 = c(4L,      4L, 4L, 0L, 0L, 0L, 4L, 3L, 3L, 4L, 0L, 4L, 0L, 4L, 4L, 0L,      4L, 0L, 0L, 4L, 0L, 0L, 0L, 1L,
3L, 4L, 0L, 4L, 0L, 4L, 1L,      0L, 0L, 4L, 3L, 0L, 0L, 3L, 3L, 4L, 0L, 1L, 0L, 0L, 4L, 0L,      3L, 4L, 0L, 0L, 4L, 4L,
0L, 4L, 1L, 0L, 0L, 2L, 4L, 3L, 4L,      0L, 0L, 0L, 4L, 4L, 0L, 0L, 3L, 0L, 0L, 3L, 0L, 2L, 0L, 4L,      4L, 4L, 4L, 0L,
4L, 0L, 0L, 4L, 0L, 3L, 0L, 0L, 0L, 0L, 0L,      1L, 0L, 3L, 0L, 0L, 4L, 1L, 0L, 0L, 0L, 4L, 0L, 4L, 0L, 3L,      0L, 3L,
4L, 4L, 0L, 2L, 3L, 4L, 4L, 0L, 3L, 3L, 4L), Atr41 = c(4L,      4L, 4L, 2L, 0L, 0L, 4L, 3L, 2L, 4L, 1L, 3L, 2L, 4L, 3L, 0
L,      4L, 1L, 0L, 3L, 0L, 0L, 0L, 0L, 4L, 4L, 0L, 3L, 1L, 4L, 1L,      0L, 0L, 4L, 4L, 0L, 0L, 4L, 2L, 4L, 0L, 1L, 1L,
0L, 4L, 0L,      4L, 4L, 0L, 1L, 4L, 4L, 1L, 4L, 0L, 0L, 0L, 0L, 4L, 4L, 4L,      0L, 1L, 0L, 4L, 4L, 0L, 1L, 4L, 1L, 0L,
4L, 1L, 2L, 1L, 3L,      4L, 4L, 4L, 0L, 4L, 0L, 3L, 2L, 0L, 3L, 0L, 0L, 0L, 0L, 0L,      2L, 0L, 4L, 0L, 0L, 4L, 0L, 0L,
0L, 0L, 4L, 2L, 4L, 1L, 4L,      0L, 4L, 4L, 4L, 0L, 2L, 3L, 4L, 4L, 0L, 3L, 3L, 4L), Atr42 = c(4L,      4L, 4L, 4L, 2L,
0L, 3L, 3L, 1L, 4L, 1L, 4L, 2L, 4L, 4L, 1L,      4L, 3L, 0L, 4L, 0L, 4L, 1L, 0L, 3L, 4L, 1L, 0L, 0L, 4L, 1L,      1L, 4L,
4L, 3L, 0L, 3L, 3L, 3L, 4L, 0L, 3L, 1L, 0L, 4L, 2L,      3L, 4L, 0L, 0L, 4L, 4L, 0L, 4L, 2L, 0L, 0L, 0L, 3L, 3L, 4L,
0L, 0L, 0L, 4L, 4L, 0L, 2L, 3L, 2L, 2L, 3L, 0L, 2L, 0L, 4L,      4L, 4L, 3L, 0L, 4L, 0L, 1L, 2L, 0L, 3L, 0L, 2L, 0L, 3L,

```

```

2L,      1L, 2L, 3L, 0L, 0L, 4L, 2L, 0L, 2L, 2L, 4L, 0L, 4L, 3L, 3L,      0L, 3L, 4L, 4L, 0L, 1L, 3L, 4L, 4L, 0L, 3L, 3L,
4L), Atr43 = c(4L,      4L, 4L, 4L, 4L, 2L, 4L, 3L, 1L, 4L, 1L, 3L, 3L, 4L, 3L, 3L,      4L, 2L, 2L, 3L, 0L, 2L, 1L, 3L,
4L, 4L, 2L, 0L, 0L, 4L, 1L,      2L, 4L, 3L, 4L, 2L, 2L, 4L, 3L, 4L, 2L, 2L, 4L, 4L, 2L,      4L, 4L, 3L, 1L, 4L, 4L,
2L, 4L, 2L, 1L, 1L, 0L, 0L, 4L, 4L,      0L, 0L, 3L, 4L, 4L, 0L, 4L, 4L, 3L, 2L, 4L, 2L, 2L, 1L, 3L,      4L, 4L, 4L, 3L,
4L, 2L, 1L, 3L, 0L, 3L, 1L, 4L, 0L, 2L, 2L,      1L, 3L, 4L, 0L, 2L, 4L, 2L, 2L, 2L, 2L, 4L, 3L, 4L, 4L, 4L,      2L, 4L,
4L, 4L, 3L, 2L, 3L, 4L, 4L, 4L, 3L, 3L, 4L), Atr44 = c(4L,      4L, 4L, 1L, 4L, 0L, 3L, 3L, 2L, 4L, 0L, 4L, 0L, 4L, 4L, 2
L,      4L, 0L, 1L, 4L, 2L, 0L, 1L, 2L, 3L, 4L, 0L, 3L, 0L, 4L, 1L,      0L, 0L, 4L, 3L, 1L, 1L, 3L, 4L, 4L, 0L, 0L, 0L,
0L, 4L, 1L,      3L, 4L, 0L, 0L, 3L, 4L, 0L, 4L, 0L, 0L, 0L, 0L, 0L, 3L, 4L,      0L, 0L, 0L, 4L, 4L, 0L, 2L, 3L, 0L, 0L,
3L, 0L, 2L, 2L, 4L,      3L, 4L, 4L, 0L, 4L, 2L, 0L, 4L, 0L, 4L, 1L, 1L, 0L, 1L, 0L,      0L, 1L, 3L, 0L, 0L, 4L, 2L, 2L,
0L, 2L, 4L, 0L, 4L, 0L, 3L,      0L, 3L, 4L, 4L, 1L, 3L, 3L, 4L, 4L, 4L, 3L, 3L, 4L), Atr45 = c(4L,      4L, 3L, 4L, 4L,
2L, 3L, 3L, 3L, 4L, 3L, 3L, 3L, 4L, 3L, 1L,      4L, 1L, 1L, 3L, 1L, 1L, 2L, 4L, 4L, 3L, 2L, 0L, 0L, 4L, 1L,      1L, 4L,
2L, 4L, 0L, 3L, 3L, 4L, 4L, 1L, 2L, 1L, 4L, 4L, 2L,      3L, 4L, 0L, 0L, 4L, 4L, 0L, 4L, 0L, 3L, 0L, 0L, 0L, 4L, 4L,
0L, 0L, 0L, 4L, 4L, 1L, 3L, 4L, 1L, 1L, 4L, 1L, 2L, 0L, 3L,      4L, 3L, 4L, 3L, 4L, 2L, 1L, 2L, 0L, 3L, 0L, 0L, 0L, 2L,
3L,      4L, 3L, 4L, 0L, 0L, 3L, 1L, 4L, 1L, 2L, 4L, 3L, 4L, 4L, 4L,      2L, 4L, 3L, 4L, 0L, 0L, 3L, 4L, 4L, 4L, 3L, 3L,
4L), Atr46 = c(4L,      4L, 4L, 0L, 4L, 0L, 3L, 3L, 2L, 4L, 3L, 4L, 2L, 4L, 4L, 3L,      4L, 3L, 1L, 4L, 4L, 2L, 3L, 4L,
3L, 4L, 1L, 0L, 1L, 4L, 0L,      4L, 1L, 2L, 3L, 3L, 3L, 3L, 2L, 4L, 2L, 1L, 2L, 3L, 2L, 3L,      3L, 4L, 2L, 0L, 4L, 4L,
2L, 4L, 2L, 2L, 3L, 0L, 0L, 3L, 4L,      0L, 0L, 0L, 4L, 4L, 1L, 1L, 3L, 1L, 0L, 3L, 1L, 1L, 1L, 4L,      4L, 4L, 4L, 4L,
4L, 0L, 3L, 2L, 3L, 3L, 4L, 2L, 2L, 2L, 3L,      3L, 3L, 3L, 0L, 2L, 4L, 0L, 4L, 3L, 2L, 4L, 3L, 4L, 3L, 3L,      2L, 3L,
4L, 4L, 3L, 2L, 3L, 4L, 4L, 4L, 3L, 3L, 4L), Atr47 = c(4L,      4L, 4L, 4L, 4L, 2L, 4L, 3L, 1L, 4L, 2L, 3L, 0L, 4L, 3L, 0
L,      4L, 3L, 1L, 3L, 2L, 1L, 1L, 0L, 4L, 4L, 0L, 0L, 0L, 4L, 0L,      4L, 4L, 3L, 4L, 0L, 0L, 4L, 3L, 4L, 1L, 2L, 2L,
1L, 4L, 0L,      4L, 4L, 2L, 1L, 3L, 4L, 1L, 4L, 1L, 0L, 0L, 0L, 0L, 4L, 4L,      0L, 0L, 0L, 4L, 4L, 1L, 1L, 4L, 2L, 1L,
4L, 1L, 1L, 0L, 3L,      4L, 4L, 4L, 3L, 4L, 0L, 3L, 2L, 0L, 2L, 1L, 1L, 2L, 1L, 3L,      3L, 3L, 4L, 0L, 0L, 4L, 1L, 4L,
4L, 2L, 4L, 1L, 4L, 4L, 4L,      1L, 4L, 4L, 4L, 0L, 2L, 3L, 4L, 4L, 4L, 3L, 3L, 4L), Atr48 = c(4L,      4L, 4L, 4L, 4L,
3L, 4L, 3L, 3L, 4L, 2L, 4L, 2L, 4L, 4L, 2L,      4L, 2L, 2L, 4L, 3L, 3L, 0L, 1L, 3L, 4L, 0L, 3L, 4L, 4L, 2L,      3L, 2L,
3L, 3L, 2L, 2L, 3L, 2L, 4L, 3L, 3L, 2L, 3L, 4L, 2L,      3L, 4L, 0L, 0L, 4L, 4L, 2L, 4L, 2L, 2L, 2L, 3L, 3L, 3L, 4L,
4L, 2L, 2L, 4L, 4L, 0L, 2L, 3L, 2L, 2L, 3L, 1L, 1L, 3L, 4L,      4L, 4L, 4L, 0L, 4L, 2L, 2L, 3L, 2L, 3L, 1L, 2L, 3L, 2L,
3L,      2L, 2L, 3L, 2L, 2L, 4L, 3L, 2L, 4L, 2L, 4L, 2L, 4L, 3L, 3L,      1L, 3L, 4L, 4L, 1L, 1L, 3L, 4L, 4L, 0L, 3L, 3L,
4L), Atr49 = c(4L,      4L, 4L, 4L, 4L, 0L, 4L, 3L, 3L, 4L, 0L, 3L, 4L, 4L, 3L, 3L,      4L, 2L, 1L, 3L, 0L, 1L, 0L, 0L,
4L, 4L, 1L, 3L, 1L, 4L, 1L,      3L, 3L, 2L, 4L, 3L, 3L, 4L, 2L, 4L, 3L, 2L, 2L, 1L, 3L, 0L,      4L, 4L, 0L, 0L, 4L, 4L,
2L, 4L, 1L, 2L, 2L, 1L, 3L, 4L, 4L,      4L, 2L, 0L, 4L, 4L, 1L, 0L, 4L, 1L, 0L, 4L, 0L, 1L, 1L, 3L,      4L, 4L, 4L, 2L,
4L, 2L, 0L, 4L, 2L, 2L, 4L, 1L, 3L, 3L, 0L,      2L, 1L, 4L, 1L, 3L, 4L, 2L, 2L, 0L, 0L, 4L, 1L, 4L, 1L, 4L,      1L, 4L,
4L, 4L, 0L, 2L, 3L, 4L, 4L, 4L, 3L, 3L, 4L), Atr50 = c(4L,      4L, 4L, 3L, 2L, 3L, 4L, 3L, 3L, 4L, 2L, 3L, 0L, 4L, 3L, 3
L,      4L, 1L, 1L, 3L, 0L, 1L, 1L, 0L, 4L, 4L, 1L, 3L, 1L, 4L, 2L,      1L, 2L, 1L, 4L, 4L, 2L, 4L, 3L, 4L, 2L, 2L, 2L,
3L, 4L, 0L,      4L, 4L, 1L, 0L, 3L, 4L, 2L, 4L, 2L, 2L, 2L, 1L, 3L, 4L, 4L,      4L, 2L, 0L, 4L, 4L, 2L, 0L, 4L, 0L, 0L,
4L, 0L, 1L, 1L, 3L,      4L, 4L, 4L, 2L, 4L, 2L, 0L, 4L, 2L, 3L, 4L, 2L, 3L, 4L, 2L, 3L, 2L, 1L,      3L, 1L, 4L, 2L, 2L, 4L, 2L, 2L,
1L, 2L, 4L, 1L, 4L, 3L, 4L,      1L, 4L, 4L, 4L, 0L, 1L, 3L, 4L, 4L, 0L, 3L, 3L, 4L), Atr51 = c(4L,      4L, 4L, 4L, 2L,
3L, 3L, 3L, 2L, 4L, 2L, 4L, 0L, 4L, 4L, 2L,      4L, 1L, 1L, 4L, 0L, 1L, 1L, 1L, 3L, 4L, 1L, 3L, 4L, 4L, 2L,      2L, 2L,
1L, 3L, 3L, 2L, 3L, 3L, 4L, 2L, 2L, 2L, 3L, 4L, 2L,      3L, 4L, 1L, 1L, 3L, 4L, 2L, 4L, 2L, 2L, 2L, 1L, 3L, 3L, 4L,
4L, 2L, 1L, 4L, 4L, 2L, 1L, 3L, 1L, 0L, 3L, 0L, 1L, 3L, 4L,      4L, 4L, 4L, 2L, 4L, 2L, 0L, 4L, 2L, 3L, 2L, 2L, 3L, 2L,
3L,      2L, 1L, 3L, 2L, 2L, 4L, 2L, 2L, 3L, 1L, 4L, 1L, 4L, 3L, 3L,      1L, 3L, 4L, 4L, 0L, 1L, 3L, 4L, 4L, 0L, 3L, 3L,
4L), Atr52 = c(4L,      4L, 4L, 4L, 0L, 3L, 3L, 3L, 3L, 4L, 0L, 3L, 1L, 4L, 3L, 1L,      4L, 2L, 0L, 3L, 2L, 1L, 0L, 4L,
4L, 4L, 1L, 3L, 2L, 4L, 2L,      3L, 0L, 2L, 4L, 3L, 4L, 4L, 4L, 4L, 2L, 1L, 2L, 3L, 1L, 1L,      4L, 4L, 3L, 0L, 4L, 4L,
2L, 4L, 1L, 2L, 2L, 0L, 3L, 4L, 4L,      0L, 1L, 2L, 4L, 4L, 0L, 4L, 4L, 3L, 3L, 4L, 1L, 1L, 1L, 3L,      4L, 4L, 4L, 0L,
4L, 1L, 4L, 4L, 4L, 2L, 2L, 4L, 1L, 3L, 3L,      4L, 1L, 4L, 3L, 1L, 4L, 1L, 4L, 3L, 1L, 4L, 0L, 4L, 0L, 4L,      1L, 4L,
4L, 4L, 2L, 1L, 3L, 4L, 4L, 0L, 3L, 3L, 4L), Atr53 = c(4L,      4L, 4L, 4L, 0L, 2L, 4L, 3L, 2L, 4L, 0L, 4L, 0L, 4L, 4L, 0

```

```

L,      4L, 0L, 0L, 4L, 4L, 1L, 0L, 1L, 3L, 4L, 1L, 3L, 2L, 4L, 2L,      1L, 0L, 2L, 3L, 0L, 1L, 3L, 2L, 4L, 2L, 1L, 2L,
1L, 1L, 0L,      3L, 4L, 4L, 0L, 4L, 4L, 1L, 4L, 0L, 2L, 2L, 3L, 3L, 3L, 4L,      0L, 0L, 2L, 4L, 4L, 1L, 1L, 3L, 2L, 2L,
3L, 1L, 1L, 1L, 4L,      4L, 4L, 4L, 0L, 4L, 1L, 4L, 2L, 4L, 2L, 0L, 2L, 1L, 1L, 3L,      3L, 1L, 3L, 3L, 1L, 4L, 0L, 2L,
3L, 1L, 4L, 1L, 4L, 1L, 3L,      1L, 3L, 4L, 4L, 0L, 2L, 3L, 4L, 4L, 0L, 3L, 3L, 4L), Atr54 = c(4L,      4L, 4L, 4L, 0L,
2L, 4L, 3L, 1L, 4L, 4L, 3L, 0L, 4L, 3L, 0L,      4L, 0L, 0L, 3L, 1L, 0L, 0L, 0L, 4L, 4L, 1L, 3L, 2L, 4L, 3L,      1L, 0L,
2L, 4L, 2L, 1L, 4L, 2L, 4L, 0L, 0L, 2L, 0L, 0L, 0L,      4L, 4L, 0L, 1L, 3L, 4L, 0L, 4L, 0L, 2L, 1L, 0L, 3L, 4L, 4L,
0L, 0L, 1L, 4L, 4L, 1L, 2L, 4L, 2L, 1L, 4L, 1L, 1L, 1L, 3L,      4L, 4L, 4L, 0L, 4L, 0L, 2L, 2L, 0L, 2L, 1L, 0L, 0L, 1L,
1L,      1L, 1L, 4L, 2L, 0L, 4L, 0L, 2L, 1L, 0L, 4L, 0L, 4L, 0L, 4L,      1L, 4L, 4L, 4L, 0L, 0L, 3L, 4L, 4L, 0L, 3L, 3L,
4L), Class = c(1L,      1L, 1L, 1L, 0L, 0L, 1L, 1L, 1L, 1L, 0L, 1L, 0L, 1L, 1L, 0L,      1L, 0L, 0L, 1L, 0L, 0L, 0L, 0L,
1L, 1L, 0L, 1L, 0L, 1L, 0L,      0L, 0L, 1L, 1L, 0L, 0L, 1L, 1L, 1L, 0L, 0L, 0L, 0L, 1L, 0L,      1L, 1L, 0L, 0L, 1L, 1L,
0L, 1L, 0L, 0L, 0L, 0L, 1L, 1L, 1L,      0L, 0L, 0L, 1L, 1L, 0L, 0L, 1L, 0L, 0L, 1L, 0L, 1L,      1L, 1L, 1L, 0L,
1L, 0L, 0L, 1L, 0L, 1L, 0L, 0L, 0L, 0L,      0L, 0L, 1L, 0L, 0L, 1L, 0L, 0L, 0L, 0L, 1L, 0L, 1L, 0L, 1L,      0L, 1L,
1L, 1L, 0L, 1L, 1L, 1L, 0L, 1L, 1L, 1L)), class = "data.frame", row.names = c(NA, 119L)), mtry = 10L)

```

Type of random forest: regression

Number of trees: 500

No. of variables tried at each split: 10

Mean of squared residuals: 0.01563012

% Var explained: 93.74

```

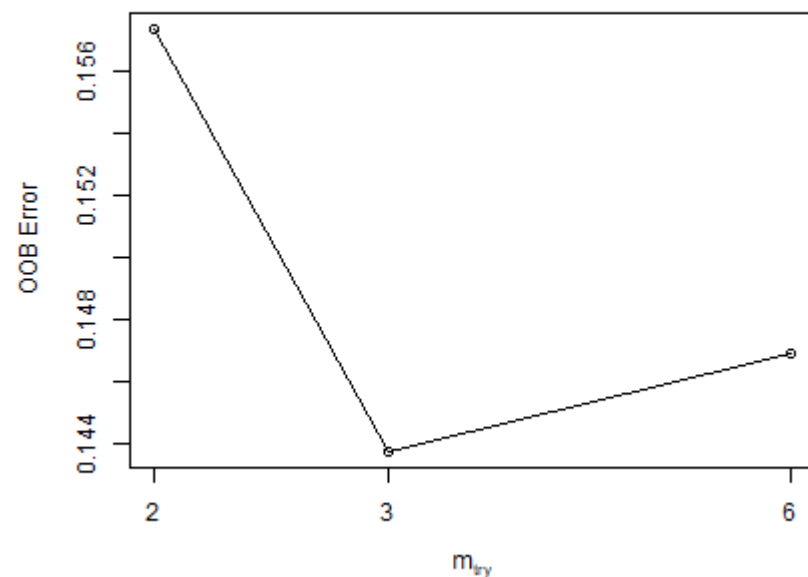
In [110... Xdf = traindf[:,1:10];
           Ydf = traindf[:,11];

```

```

In [111... trf=tuneRF(Xdf,Ydf)

```



```
mtry = 3 OOB error = 0.1437453
```

```
Searching left ...
```

```
mtry = 2 OOB error = 0.1573242
```

```
-0.09446494 0.05
```

```
Searching right ...
```

```
mtry = 6 OOB error = 0.1468726
```

```
-0.02175589 0.05
```

```
Warning: RCall.jl: Warning in randomForest.default(x, y, mtry = mtryStart, ntree = ntreeTry, :
  The response has five or fewer unique values. Are you sure you want to do regression?
Warning in randomForest.default(x, y, mtry = mtryCur, ntree = ntreeTry, :
  The response has five or fewer unique values. Are you sure you want to do regression?
Warning in randomForest.default(x, y, mtry = mtryCur, ntree = ntreeTry, :
  The response has five or fewer unique values. Are you sure you want to do regression?
@ RCall C:\Users\ASUS\julia\packages\RCall\6kphM\src\io.jl:172
```

```
Out[111...] RObject{RealSxp}
  mtry OOBError
2  2 0.1573242
3  3 0.1437453
6  6 0.1468726
```

```
In [112...]
```

```
prf= R"predict($rfmodel)"
```

Out[112... RObject{RealSxp}

	1	2	3	4	5
1.000000e+00	1.000000e+00	1.000000e+00	5.969697e-01	2.298851e-02	
6	7	8	9	10	
1.442308e-02	1.000000e+00	1.000000e+00	1.893939e-01	1.000000e+00	
11	12	13	14	15	
1.373626e-03	1.000000e+00	1.001821e-01	1.000000e+00	1.000000e+00	
16	17	18	19	20	
6.818182e-03	1.000000e+00	1.530612e-02	6.535948e-03	1.000000e+00	
21	22	23	24	25	
2.272727e-02	-2.220446e-16	1.010101e-02	-3.330669e-16	1.000000e+00	
26	27	28	29	30	
1.000000e+00	1.063830e-02	8.589744e-01	5.376344e-02	1.000000e+00	
31	32	33	34	35	
1.163873e-01	1.047120e-02	-2.775558e-16	9.416206e-01	1.000000e+00	
36	37	38	39	40	
1.317365e-02	5.681818e-03	1.000000e+00	7.303665e-01	1.000000e+00	
41	42	43	44	45	
2.285714e-03	6.172840e-03	1.025641e-02	1.129944e-03	6.806283e-01	
46	47	48	49	50	
6.723164e-02	1.000000e+00	1.000000e+00	5.347594e-03	5.952381e-03	
51	52	53	54	55	
1.000000e+00	1.000000e+00	2.721088e-03	1.000000e+00	4.051565e-02	
56	57	58	59	60	
5.235602e-03	9.126984e-02	2.279570e-01	8.596491e-01	1.000000e+00	
61	62	63	64	65	
1.000000e+00	1.209738e-01	5.617978e-03	2.197802e-02	1.000000e+00	
66	67	68	69	70	
1.000000e+00	2.116279e-01	5.319149e-03	1.000000e+00	2.717391e-02	
71	72	73	74	75	
2.923977e-02	1.000000e+00	5.291005e-03	8.279279e-01	3.597450e-02	
76	77	78	79	80	
1.000000e+00	1.000000e+00	1.000000e+00	9.553704e-01	1.058201e-02	
81	82	83	84	85	
1.000000e+00	-6.661338e-16	7.397504e-02	8.264249e-01	2.378378e-02	
86	87	88	89	90	
9.502762e-01	8.539326e-02	-4.996004e-16	1.169591e-03	5.524862e-03	
91	92	93	94	95	
-4.996004e-16	1.356383e-01	-2.775558e-16	1.000000e+00	1.212121e-02	
96	97	98	99	100	
-8.326673e-16	1.000000e+00	2.873563e-02	-7.216450e-16	7.231041e-02	
101	102	103	104	105	
2.259887e-02	1.000000e+00	5.240741e-02	1.000000e+00	7.526882e-03	
106	107	108	109	110	
1.000000e+00	9.193841e-02	1.000000e+00	1.000000e+00	1.000000e+00	

111	112	113	114	115
-5.551115e-16	2.610820e-01	1.000000e+00	1.000000e+00	1.000000e+00
116	117	118	119	
2.622673e-02	1.000000e+00	1.000000e+00	1.000000e+00	

In [113...

```
rfcv(Xdf,Ydf)
```

[illegible]


```

| The response has five or fewer unique values. Are you sure you want to do regression?
| Warning in randomForest.default(trainx[idx != i, imp.idx, drop = FALSE], :
| The response has five or fewer unique values. Are you sure you want to do regression?
| @ RCall C:\Users\ASUS\julia\packages\RCall\6kphM\src\io.jl:172
RObject{VecSxp}
Out[113...] $n.var
[1] 10 5 2 1

$error.cv
      10      5      2      1
0.1425600 0.1418292 0.2704712 0.4154540

$predicted
$predicted$`10`
 [1] 3.217002165 3.930833333 3.082450000 3.554611111 0.096876637 0.078601875
 [7] 2.959033333 3.002666667 0.435136508 3.010263333 0.342923651 3.048569683
[13] 0.918752842 3.118640548 3.969000000 0.009959272 3.089788064 0.323134806
[19] 0.219348845 3.027485814 0.189206436 0.065247056 0.015484372 0.598812792
[25] 3.148423188 3.001186098 0.529586957 3.423780159 0.015484372 3.932300000
[31] 0.577338524 0.164479530 0.023806392 1.713677886 3.971228205 0.014602192
[37] 0.012371455 3.010979308 2.188381169 3.083983333 0.212871915 0.017640875
[43] 0.228870269 0.021750418 3.791386111 0.366986100 3.102033333 3.083983333
[49] 0.167289522 0.021750418 3.999100000 3.001349913 0.509780659 3.010263333
[55] 0.278807937 0.034187872 0.588368326 0.541740298 3.458179739 3.911966667
[61] 3.942833333 1.269841270 0.028277758 0.001318015 3.040144526 3.834300000
[67] 1.293336839 0.012371455 3.932300000 0.293113565 0.532951071 3.932300000
[73] 0.009959272 1.405488095 0.760974603 2.998088071 3.001390909 3.027485814
[79] 2.271950000 0.020602192 2.975913131 0.001318015 0.719296311 3.831866667
[85] 0.021750418 2.124355526 0.144566730 0.140686788 0.457437840 0.009959272
[91] 0.086245794 0.015484372 0.092096028 3.096978947 0.291900755 0.012371455
[97] 3.202346351 0.219348845 0.116546216 0.127879045 0.082269625 3.027485814
[103] 0.021750418 3.997600000 0.015484372 3.992161538 0.804978274 3.967666667
[109] 3.208935202 3.984775000 0.081316158 0.051022380 3.084384312 2.996700000
[115] 3.118640548 0.153325106 3.000500000 3.058625541 3.851058333

$predicted$`5`
 [1] 3.36207871 3.93887816 3.20327517 3.37694794 0.07602822 0.04690119
 [7] 3.00322133 3.00337517 0.32342676 3.02337159 0.28431963 3.02337159
[13] 0.13662560 3.16508479 3.95212222 0.04606490 3.04918308 0.39938099
[19] 0.06940970 3.10044274 0.07933349 0.07602822 0.13662560 0.69063453
[25] 3.06595816 3.02721817 0.57466155 3.24039952 0.13662560 3.91776242
[31] 0.46096182 0.13662560 0.13662560 1.65701277 3.94373531 0.07602822
[37] 0.08743153 3.06860261 2.39702809 3.20445466 0.15431408 0.08743153
[43] 0.04606490 0.07933349 3.91563333 0.50228199 3.02868558 3.20445466
[49] 0.07933349 0.07933349 3.99641921 3.10044274 0.57466155 3.02337159

```

```
[55] 0.07602822 0.07933349 0.50228199 0.39938099 3.23831020 3.92664657
[61] 3.93132460 0.76419365 0.06940970 0.07602822 3.07473281 3.86865556
[67] 1.26974972 0.08743153 3.91776242 0.13662560 0.07602822 3.91776242
[73] 0.04606490 1.54316573 0.75220208 3.02959565 3.03047948 3.10044274
[79] 2.09563651 0.07602822 3.03047948 0.07602822 0.96669086 3.98507390
[85] 0.07933349 2.18251008 0.08743153 0.07665678 0.18016277 0.04606490
[91] 0.06094539 0.13662560 0.15431408 3.06860261 0.29092146 0.08743153
[97] 3.12467554 0.06940970 0.08743153 0.20335647 0.22183936 3.10044274
[103] 0.07933349 3.98620905 0.13662560 3.97375198 0.75220208 3.97850000
[109] 3.23869357 3.99186667 0.17752927 0.07602822 3.17279998 3.00322133
[115] 3.16508479 0.07933349 3.01411588 3.04802754 3.87796667
```

\$predicted\$`2`

```
[1] 3.56942643 3.32478302 3.32772286 3.64625221 0.09346492 0.10089578
[7] 3.25421334 3.25313831 0.14192120 3.31394007 0.14192120 3.31394007
[13] 0.21412794 3.07545257 3.12996122 0.14192120 3.32134562 0.14192120
[19] 0.14192120 3.31969078 0.24283847 0.09346492 0.21412794 0.19540795
[25] 3.21857258 3.18866784 0.24283847 3.27688080 0.21412794 3.63736345
[31] 0.21412794 0.21412794 0.21412794 1.57633577 3.05184485 0.09346492
[37] 0.19540795 3.08142144 3.32134562 3.32067564 0.24283847 0.19540795
[43] 0.14192120 0.24283847 3.19418614 0.69777234 3.63736345 3.32067564
[49] 0.24283847 0.24283847 3.91342491 3.31969078 0.24283847 3.31394007
[55] 0.09346492 0.24283847 0.69777234 0.14192120 3.29716810 3.18866784
[61] 3.07545257 0.21412794 0.14192120 0.09346492 3.27461706 3.07545257
[67] 1.28339311 0.19540795 3.63736345 0.21412794 0.09346492 3.63736345
[73] 0.14192120 0.69777234 0.21412794 3.28157757 3.08142144 3.31969078
[79] 3.32478302 0.09346492 3.08142144 0.09346492 1.36207798 3.91342491
[85] 0.24283847 0.78897755 0.19540795 0.19540795 0.13263621 0.14192120
[91] 0.13263621 0.21412794 0.24283847 3.08142144 0.56271791 0.19540795
[97] 3.17607236 0.14192120 0.19540795 0.45216356 0.19540795 3.31969078
[103] 0.24283847 3.31969078 0.21412794 3.18866784 0.21412794 3.32134562
[109] 3.07545257 3.66193413 0.10089578 0.09346492 3.28157757 3.25421334
[115] 3.07545257 0.24283847 3.21644972 3.08142144 3.12679013
```

\$predicted\$`1`

```
[1] 3.1933636 3.1933636 3.4614389 3.4614389 0.1687747 0.2063020 3.2167718
[8] 3.4614389 0.1403858 3.2874946 0.1403858 3.2874946 0.2099054 2.7345621
[15] 3.2360630 0.1403858 3.2360630 0.1403858 0.1403858 3.2874946 0.2499281
[22] 0.1687747 0.2099054 0.2063020 3.2389530 3.1503111 0.2499281 3.2389530
[29] 0.2099054 3.3872977 0.2099054 0.2099054 0.2099054 3.2874946 3.1078773
[36] 0.1687747 0.2063020 3.1110543 3.2360630 3.2167718 0.2499281 0.2063020
[43] 0.1403858 0.2499281 2.7345621 0.1687747 3.3872977 3.2167718 0.2499281
[50] 0.2499281 3.9379380 3.2874946 0.2499281 3.2874946 0.1687747 0.2499281
[57] 0.1687747 0.1403858 3.1078773 3.1503111 2.7345621 0.2099054 0.1403858
[64] 0.1687747 3.1503111 2.7345621 0.3133351 0.2063020 3.3872977 0.2099054
```

```
[71] 0.1687747 3.3872977 0.1403858 0.1687747 0.2099054 3.2360630 3.1110543
[78] 3.2874946 3.1933636 0.1687747 3.1110543 0.1687747 3.1933636 3.9379380
[85] 0.2499281 0.2613012 0.2063020 0.2063020 0.4496823 0.1403858 0.4496823
[92] 0.2099054 0.2499281 3.1110543 0.6072607 0.2063020 3.3872977 0.1403858
[99] 0.2063020 0.6072607 0.2063020 3.2874946 0.2499281 3.2874946 0.2099054
[106] 3.1503111 0.2099054 3.2360630 2.7345621 3.3544583 0.2063020 0.1687747
[113] 3.2360630 3.2167718 2.7345621 0.2499281 3.1503111 3.1110543 3.3544583
```

(c) Boosting Model

```
In [114... @rlibrary gbm
using CategoricalArrays
```

```
In [115... gbmfit = gbm(m,distribution="bernoulli",data=traindf,var"n.trees"=1000,var"cv.folds"=5)
```

```
Out[115... RObject{VecSxp}
(function (formula = formula(data), distribution = "bernoulli",
  data = list(), weights, var.monotone = NULL, n.trees = 100,
  interaction.depth = 1, n.minobsinnode = 10, shrinkage = 0.1,
  bag.fraction = 0.5, train.fraction = 1, cv.folds = 0, keep.data = TRUE,
  verbose = FALSE, class.stratify.cv = NULL, n.cores = NULL)
{
  mcall <- match.call()
  lVerbose <- if (!is.logical(verbose)) {
    FALSE
  }
  else {
    verbose
  }
  mf <- match.call(expand.dots = FALSE)
  m <- match(c("formula", "data", "weights",
    "offset"), names(mf), 0)
  mf <- mf[c(1, m)]
  mf$drop.unused.levels <- TRUE
  mf$na.action <- na.pass
  mf[[1]] <- as.name("model.frame")
  m <- mf
  mf <- eval(mf, parent.frame())
  Terms <- attr(mf, "terms")
  w <- model.weights(mf)
```

```

offset <- model.offset(mf)
y <- model.response(mf)
if (missing(distribution)) {
  distribution <- guessDist(y)
}
if (is.character(distribution)) {
  distribution <- list(name = distribution)
}
if (!is.element(distribution$name, getAvailableDistributions())) {
  stop("Distribution ", distribution$name, " is not supported.")
}
if (distribution$name == "multinomial") {
  warning("Setting `distribution = \"multinomial\"` is ill-advised as it is ",
    "currently broken. It exists only for backwards compatibility. ",
    "Use at your own risk.", call. = FALSE)
}
var.names <- attributes(Terms)$term.labels
x <- model.frame(terms(reformulate(var.names)), data = data,
  na.action = na.pass)
response.name <- as.character(formula[[2L]])
class.stratify.cv <- getStratify(class.stratify.cv, d = distribution)
group <- NULL
num.groups <- 0
if (distribution$name != "pairwise") {
  nTrain <- floor(train.fraction * nrow(x))
}
else {
  distribution.group <- distribution[["group"]]
  if (is.null(distribution.group)) {
    stop(paste("For pairwise regression, `distribution` must be a list of",
      "the form `list(name = \"pairwise\", group = c(\"date\",",
      "\"session\", \"category\", \"keywords\"))`.")
  )
  i <- match(distribution.group, colnames(data))
  if (any(is.na(i))) {
    stop("Group column does not occur in data: ",
      distribution.group[is.na(i)], ".")
  }
  group <- factor(do.call(paste, c(data[, distribution.group,
    drop = FALSE], sep = ":")))
  if ((!missing(weights)) && (!is.null(weights))) {
    w.min <- tapply(w, INDEX = group, FUN = min)
    w.max <- tapply(w, INDEX = group, FUN = max)
    if (any(w.min != w.max)) {
      stop("For `distribution = \"pairwise\"`, all instances for the same ",

```

```

        "group must have the same weight.")
    }
    w <- w * length(w.min)/sum(w.min)
  }
  perm.levels <- levels(group)[sample(1:nlevels(group))]
  group <- factor(group, levels = perm.levels)
  ord.group <- order(group, -y)
  group <- group[ord.group]
  y <- y[ord.group]
  x <- x[ord.group, , drop = FALSE]
  w <- w[ord.group]
  num.groups.train <- max(1, round(train.fraction * nlevels(group)))
  nTrain <- max(which(group == levels(group)[num.groups.train]))
  Misc <- group
}
cv.error <- NULL
if (cv.folds == 1) {
  cv.folds <- 0
}
if (cv.folds > 1) {
  cv.results <- gbmCrossVal(cv.folds = cv.folds, nTrain = nTrain,
    n.cores = n.cores, class.stratify.cv = class.stratify.cv,
    data = data, x = x, y = y, offset = offset, distribution = distribution,
    w = w, var.monotone = var.monotone, n.trees = n.trees,
    interaction.depth = interaction.depth, n.minobsinnode = n.minobsinnode,
    shrinkage = shrinkage, bag.fraction = bag.fraction,
    var.names = var.names, response.name = response.name,
    group = group)
  cv.error <- cv.results$error
  p <- cv.results$predictions
}
gbm.obj <- gbm.fit(x = x, y = y, offset = offset, distribution = distribution,
  w = w, var.monotone = var.monotone, n.trees = n.trees,
  interaction.depth = interaction.depth, n.minobsinnode = n.minobsinnode,
  shrinkage = shrinkage, bag.fraction = bag.fraction, nTrain = nTrain,
  keep.data = keep.data, verbose = lVerbose, var.names = var.names,
  response.name = response.name, group = group)
gbm.obj$train.fraction <- train.fraction
gbm.obj$Terms <- Terms
gbm.obj$cv.error <- cv.error
gbm.obj$cv.folds <- cv.folds
gbm.obj$call <- mcall
gbm.obj$m <- m
if (cv.folds > 1) {
  gbm.obj$cv.fitted <- p
}

```

```

}
if (distribution$name == "pairwise") {
  gbm.obj$ord.group <- ord.group
  gbm.obj$fit <- gbm.obj$fit[order(ord.group)]
}
gbm.obj
})(formula = Class ~ Atr1 + Atr2 + Atr3 + Atr4 + Atr5 + Atr6 +
  Atr7 + Atr8 + Atr9 + Atr10 + Atr11 + Atr12 + Atr13 + Atr14 +
  Atr15 + Atr16 + Atr17 + Atr18 + Atr19 + Atr20 + Atr21 + Atr22 +
  Atr23 + Atr24 + Atr25 + Atr26 + Atr27 + Atr28 + Atr29 + Atr30 +
  Atr31 + Atr32 + Atr33 + Atr34 + Atr35 + Atr36 + Atr37 + Atr38 +
  Atr39 + Atr40 + Atr41 + Atr42 + Atr43 + Atr44 + Atr45 + Atr46 +
  Atr47 + Atr48 + Atr49 + Atr50 + Atr51 + Atr52 + Atr53 + Atr54,
  distribution = "bernoulli", data = list(Atr1 = c(3L,
  4L, 3L, 4L, 0L, 1L, 3L, 3L, 2L, 3L, 3L, 3L, 0L, 3L, 4L, 0L,
  3L, 0L, 0L, 3L, 0L, 0L, 0L, 3L, 3L, 3L, 1L, 3L, 0L, 4L, 1L,
  0L, 0L, 2L, 4L, 0L, 0L, 3L, 2L, 3L, 0L, 0L, 0L, 0L, 4L, 2L,
  3L, 3L, 0L, 0L, 4L, 3L, 1L, 3L, 0L, 0L, 2L, 0L, 3L, 4L, 4L,
  4L, 0L, 0L, 3L, 4L, 1L, 0L, 4L, 0L, 0L, 4L, 0L, 2L, 2L, 3L,
  3L, 3L, 2L, 0L, 3L, 0L, 0L, 4L, 0L, 3L, 0L, 1L, 1L, 0L, 0L,
  0L, 0L, 3L, 0L, 0L, 3L, 0L, 0L, 0L, 0L, 3L, 0L, 4L, 0L, 4L,
  2L, 4L, 3L, 4L, 0L, 0L, 3L, 3L, 3L, 0L, 3L, 3L, 4L), Atr2 = c(2L,
  2L, 4L, 4L, 1L, 0L, 3L, 4L, 2L, 4L, 1L, 4L, 0L, 2L, 3L, 0L,
  3L, 0L, 1L, 3L, 0L, 0L, 0L, 0L, 3L, 3L, 1L, 2L, 0L, 4L, 0L,
  1L, 0L, 2L, 3L, 0L, 0L, 3L, 0L, 3L, 0L, 0L, 0L, 0L, 4L, 0L,
  4L, 3L, 0L, 0L, 3L, 3L, 2L, 4L, 0L, 0L, 1L, 2L, 2L, 3L, 2L,
  3L, 0L, 0L, 2L, 2L, 1L, 0L, 4L, 2L, 1L, 4L, 0L, 2L, 1L, 3L,
  4L, 3L, 2L, 0L, 4L, 0L, 1L, 4L, 0L, 3L, 0L, 0L, 1L, 0L, 0L,
  0L, 0L, 3L, 0L, 0L, 3L, 1L, 0L, 0L, 1L, 3L, 0L, 3L, 0L, 3L,
  2L, 3L, 2L, 4L, 1L, 0L, 3L, 3L, 2L, 0L, 4L, 3L, 4L), Atr3 = c(2L,
  3L, 2L, 4L, 0L, 1L, 2L, 3L, 4L, 3L, 1L, 3L, 2L, 3L, 2L, 0L,
  4L, 0L, 0L, 3L, 0L, 0L, 0L, 0L, 2L, 3L, 0L, 4L, 0L, 4L, 1L,
  1L, 0L, 2L, 2L, 0L, 0L, 3L, 2L, 2L, 1L, 0L, 0L, 0L, 3L, 1L,
  4L, 2L, 1L, 0L, 3L, 3L, 0L, 3L, 1L, 0L, 0L, 2L, 4L, 3L, 3L,
  4L, 0L, 0L, 3L, 3L, 2L, 0L, 4L, 0L, 2L, 4L, 0L, 1L, 1L, 3L,
  3L, 3L, 3L, 0L, 3L, 0L, 0L, 4L, 0L, 3L, 0L, 0L, 1L, 0L, 1L,
  0L, 0L, 3L, 0L, 0L, 4L, 0L, 0L, 0L, 1L, 3L, 0L, 3L, 0L, 3L,
  3L, 4L, 3L, 4L, 0L, 1L, 3L, 3L, 3L, 1L, 3L, 4L, 3L), Atr4 = c(3L,
  2L, 3L, 3L, 0L, 0L, 3L, 4L, 1L, 4L, 0L, 3L, 4L, 2L, 3L, 0L,
  4L, 0L, 1L, 4L, 0L, 0L, 0L, 0L, 3L, 2L, 0L, 3L, 0L, 3L, 0L,
  0L, 0L, 3L, 3L, 0L, 0L, 2L, 4L, 3L, 0L, 0L, 0L, 0L, 3L, 0L,
  4L, 3L, 0L, 0L, 3L, 2L, 0L, 4L, 1L, 0L, 2L, 1L, 3L, 4L, 4L,
  4L, 0L, 0L, 2L, 2L, 0L, 0L, 3L, 0L, 1L, 3L, 0L, 0L, 0L, 2L,
  4L, 4L, 2L, 0L, 2L, 0L, 0L, 4L, 0L, 2L, 0L, 0L, 1L, 0L, 0L,
  0L, 0L, 3L, 0L, 0L, 3L, 1L, 0L, 0L, 0L, 4L, 0L, 3L, 0L, 3L,

```

```

0L, 3L, 4L, 3L, 0L, 0L, 3L, 2L, 2L, 0L, 4L, 4L, 2L), Atr5 = c(3L,
4L, 3L, 4L, 0L, 0L, 3L, 3L, 0L, 3L, 0L, 3L, 0L, 3L, 4L, 0L,
3L, 0L, 0L, 3L, 0L, 0L, 0L, 0L, 3L, 3L, 0L, 3L, 0L, 4L, 1L,
0L, 0L, 2L, 4L, 0L, 0L, 3L, 2L, 3L, 1L, 0L, 0L, 0L, 4L, 0L,
3L, 3L, 0L, 0L, 4L, 3L, 0L, 3L, 0L, 0L, 0L, 0L, 3L, 4L, 4L,
0L, 0L, 0L, 3L, 4L, 2L, 0L, 4L, 0L, 0L, 4L, 0L, 0L, 0L, 3L,
3L, 3L, 2L, 0L, 3L, 0L, 0L, 4L, 0L, 1L, 0L, 0L, 1L, 0L, 0L,
0L, 1L, 3L, 0L, 0L, 3L, 0L, 0L, 0L, 0L, 3L, 0L, 4L, 0L, 4L,
0L, 4L, 3L, 4L, 0L, 0L, 3L, 3L, 3L, 0L, 3L, 3L, 4L), Atr6 = c(2L,
2L, 0L, 2L, 0L, 0L, 1L, 0L, 0L, 0L, 0L, 0L, 0L, 2L, 1L, 0L,
1L, 0L, 0L, 1L, 2L, 2L, 0L, 0L, 1L, 1L, 0L, 2L, 0L, 0L, 1L,
1L, 0L, 3L, 1L, 2L, 0L, 1L, 2L, 1L, 0L, 0L, 1L, 0L, 0L, 0L,
0L, 1L, 1L, 0L, 1L, 1L, 0L, 0L, 0L, 0L, 0L, 0L, 2L, 1L, 2L,
0L, 0L, 0L, 2L, 2L, 1L, 0L, 0L, 1L, 0L, 0L, 0L, 4L, 2L, 1L,
0L, 1L, 2L, 1L, 0L, 0L, 1L, 0L, 0L, 3L, 1L, 1L, 0L, 0L, 0L,
0L, 0L, 1L, 1L, 0L, 1L, 0L, 2L, 0L, 2L, 1L, 0L, 1L, 0L, 1L,
1L, 1L, 2L, 0L, 0L, 2L, 1L, 1L, 2L, 0L, 0L, 1L, 0L), Atr7 = c(1L,
0L, 1L, 4L, 0L, 0L, 1L, 1L, 0L, 1L, 0L, 1L, 0L, 1L, 0L, 0L,
1L, 0L, 0L, 1L, 1L, 0L, 0L, 0L, 1L, 1L, 0L, 3L, 0L, 0L, 0L,
0L, 0L, 2L, 0L, 0L, 0L, 1L, 4L, 1L, 0L, 0L, 0L, 0L, 4L, 0L,
1L, 1L, 0L, 0L, 0L, 1L, 0L, 1L, 0L, 0L, 0L, 0L, 3L, 0L, 0L,
0L, 0L, 0L, 1L, 0L, 0L, 0L, 0L, 0L, 0L, 0L, 0L, 1L, 0L, 1L,
1L, 1L, 2L, 0L, 1L, 0L, 0L, 0L, 0L, 4L, 0L, 0L, 0L, 0L, 0L,
0L, 0L, 1L, 0L, 0L, 1L, 0L, 0L, 0L, 0L, 1L, 0L, 0L, 0L, 0L,
0L, 0L, 1L, 0L, 0L, 0L, 1L, 1L, 1L, 0L, 1L, 1L, 0L), Atr8 = c(2L,
2L, 4L, 4L, 0L, 1L, 3L, 4L, 0L, 4L, 0L, 4L, 0L, 2L, 3L, 0L,
3L, 0L, 0L, 3L, 0L, 0L, 0L, 0L, 3L, 3L, 0L, 2L, 0L, 4L, 0L,
0L, 0L, 1L, 3L, 0L, 0L, 3L, 3L, 3L, 0L, 0L, 0L, 0L, 2L, 0L,
4L, 3L, 0L, 0L, 3L, 3L, 0L, 4L, 0L, 0L, 0L, 0L, 2L, 3L, 2L,
0L, 0L, 0L, 2L, 2L, 2L, 0L, 4L, 0L, 0L, 4L, 0L, 3L, 0L, 3L,
4L, 3L, 2L, 0L, 4L, 0L, 2L, 4L, 0L, 3L, 0L, 0L, 1L, 0L, 1L,
0L, 0L, 3L, 1L, 0L, 3L, 0L, 0L, 1L, 0L, 3L, 0L, 3L, 0L, 3L,
0L, 3L, 2L, 4L, 1L, 0L, 3L, 3L, 2L, 0L, 4L, 3L, 4L), Atr9 = c(2L,
3L, 2L, 4L, 0L, 0L, 3L, 3L, 0L, 3L, 0L, 3L, 0L, 3L, 2L, 0L,
4L, 0L, 0L, 3L, 0L, 0L, 0L, 0L, 2L, 3L, 0L, 2L, 0L, 4L, 0L,
0L, 0L, 3L, 2L, 0L, 0L, 3L, 4L, 2L, 0L, 0L, 0L, 0L, 4L, 0L,
4L, 2L, 0L, 0L, 3L, 3L, 0L, 3L, 0L, 0L, 0L, 0L, 2L, 3L, 3L,
0L, 0L, 0L, 3L, 3L, 1L, 0L, 4L, 0L, 0L, 4L, 0L, 3L, 0L, 3L,
3L, 3L, 3L, 0L, 3L, 0L, 0L, 4L, 0L, 2L, 0L, 0L, 1L, 0L, 1L,
0L, 0L, 3L, 1L, 0L, 4L, 0L, 0L, 0L, 0L, 3L, 0L, 3L, 0L, 3L,
0L, 4L, 3L, 4L, 0L, 0L, 3L, 3L, 3L, 0L, 3L, 4L, 3L), Atr10 = c(3L,
2L, 3L, 3L, 0L, 0L, 3L, 4L, 0L, 4L, 0L, 3L, 2L, 2L, 3L, 0L,
4L, 2L, 1L, 4L, 0L, 1L, 0L, 0L, 3L, 2L, 1L, 3L, 0L, 3L, 1L,
0L, 1L, 2L, 3L, 0L, 0L, 2L, 3L, 3L, 0L, 1L, 0L, 0L, 4L, 2L,
4L, 3L, 0L, 0L, 3L, 2L, 0L, 4L, 2L, 1L, 1L, 2L, 3L, 4L, 4L,

```

```

0L, 1L, 0L, 2L, 2L, 2L, 0L, 3L, 0L, 0L, 3L, 0L, 3L, 2L, 2L,
4L, 4L, 2L, 0L, 2L, 0L, 0L, 4L, 0L, 2L, 1L, 1L, 1L, 0L, 0L,
0L, 0L, 3L, 0L, 0L, 3L, 1L, 0L, 0L, 0L, 4L, 0L, 3L, 0L, 3L,
1L, 3L, 4L, 3L, 0L, 1L, 3L, 2L, 2L, 0L, 4L, 4L, 2L), Atr11 = c(3L,
4L, 3L, 4L, 0L, 1L, 4L, 3L, 1L, 3L, 1L, 3L, 0L, 3L, 4L, 0L,
3L, 0L, 0L, 3L, 0L, 0L, 0L, 0L, 3L, 3L, 0L, 4L, 0L, 4L, 0L,
0L, 0L, 1L, 4L, 0L, 0L, 3L, 2L, 3L, 0L, 0L, 0L, 0L, 3L, 1L,
3L, 3L, 0L, 0L, 4L, 3L, 0L, 3L, 1L, 0L, 1L, 1L, 4L, 4L, 4L,
0L, 0L, 0L, 3L, 4L, 1L, 0L, 4L, 0L, 0L, 4L, 0L, 3L, 0L, 3L,
3L, 3L, 2L, 1L, 3L, 0L, 1L, 4L, 0L, 2L, 0L, 1L, 1L, 0L, 0L,
0L, 0L, 3L, 0L, 0L, 3L, 1L, 0L, 0L, 0L, 3L, 0L, 4L, 0L, 4L,
1L, 4L, 3L, 4L, 0L, 0L, 3L, 3L, 3L, 0L, 3L, 3L, 4L), Atr12 = c(2L,
2L, 4L, 4L, 0L, 0L, 3L, 4L, 0L, 4L, 0L, 4L, 0L, 2L, 3L, 0L,
3L, 1L, 0L, 3L, 0L, 0L, 0L, 1L, 3L, 3L, 0L, 3L, 0L, 4L, 0L,
0L, 1L, 2L, 3L, 0L, 0L, 3L, 3L, 3L, 1L, 1L, 0L, 0L, 4L, 0L,
4L, 3L, 0L, 1L, 3L, 3L, 0L, 4L, 2L, 0L, 1L, 1L, 3L, 3L, 2L,
0L, 0L, 0L, 2L, 2L, 1L, 0L, 4L, 0L, 1L, 4L, 0L, 3L, 1L, 3L,
4L, 3L, 2L, 1L, 4L, 1L, 0L, 3L, 0L, 2L, 1L, 0L, 0L, 0L, 0L,
1L, 0L, 3L, 0L, 0L, 3L, 2L, 0L, 1L, 2L, 3L, 0L, 3L, 0L, 3L,
1L, 3L, 2L, 4L, 0L, 2L, 3L, 3L, 2L, 0L, 4L, 3L, 4L), Atr13 = c(3L,
4L, 3L, 4L, 0L, 1L, 3L, 3L, 1L, 3L, 0L, 3L, 0L, 3L, 4L, 1L,
3L, 1L, 0L, 3L, 0L, 0L, 0L, 1L, 3L, 3L, 1L, 2L, 0L, 4L, 1L,
0L, 0L, 2L, 4L, 1L, 0L, 3L, 3L, 3L, 0L, 2L, 1L, 0L, 4L, 1L,
3L, 3L, 0L, 0L, 4L, 3L, 0L, 3L, 1L, 0L, 1L, 1L, 2L, 4L, 4L,
0L, 0L, 1L, 3L, 4L, 2L, 0L, 4L, 0L, 1L, 4L, 0L, 3L, 1L, 3L,
3L, 3L, 2L, 1L, 3L, 1L, 1L, 4L, 2L, 2L, 1L, 1L, 0L, 0L, 0L,
0L, 1L, 3L, 1L, 0L, 3L, 2L, 0L, 1L, 1L, 3L, 0L, 4L, 0L, 4L,
1L, 4L, 3L, 4L, 0L, 1L, 3L, 3L, 3L, 0L, 3L, 3L, 4L), Atr14 = c(2L,
2L, 4L, 4L, 0L, 0L, 3L, 4L, 1L, 4L, 1L, 4L, 0L, 2L, 3L, 1L,
3L, 0L, 1L, 3L, 0L, 0L, 0L, 2L, 3L, 3L, 0L, 2L, 0L, 4L, 1L,
0L, 0L, 2L, 3L, 1L, 0L, 3L, 0L, 3L, 0L, 0L, 0L, 0L, 4L, 0L,
4L, 3L, 0L, 0L, 3L, 3L, 1L, 4L, 1L, 0L, 0L, 0L, 3L, 3L, 2L,
0L, 0L, 0L, 2L, 2L, 0L, 0L, 4L, 0L, 0L, 4L, 0L, 3L, 1L, 3L,
4L, 3L, 2L, 0L, 4L, 1L, 0L, 0L, 0L, 3L, 0L, 1L, 0L, 0L, 1L,
0L, 0L, 3L, 1L, 0L, 3L, 0L, 0L, 1L, 0L, 3L, 0L, 3L, 0L, 3L,
1L, 3L, 2L, 4L, 1L, 0L, 3L, 3L, 2L, 0L, 4L, 3L, 4L), Atr15 = c(2L,
3L, 2L, 4L, 0L, 0L, 3L, 3L, 0L, 3L, 1L, 3L, 0L, 3L, 2L, 0L,
4L, 0L, 0L, 3L, 0L, 1L, 0L, 1L, 2L, 3L, 0L, 1L, 0L, 4L, 1L,
0L, 0L, 3L, 2L, 0L, 0L, 3L, 4L, 2L, 0L, 1L, 0L, 0L, 4L, 2L,
4L, 2L, 0L, 0L, 3L, 3L, 0L, 3L, 1L, 0L, 1L, 0L, 1L, 3L, 3L,
0L, 0L, 0L, 3L, 3L, 2L, 0L, 4L, 0L, 0L, 4L, 0L, 3L, 1L, 3L,
3L, 3L, 3L, 0L, 3L, 1L, 0L, 4L, 0L, 2L, 0L, 1L, 0L, 0L, 0L,
0L, 1L, 3L, 0L, 0L, 4L, 0L, 0L, 0L, 0L, 3L, 0L, 3L, 0L, 3L,
1L, 4L, 3L, 4L, 0L, 2L, 3L, 3L, 3L, 0L, 3L, 4L, 3L), Atr16 = c(3L,
2L, 3L, 4L, 0L, 1L, 3L, 4L, 1L, 4L, 1L, 3L, 0L, 2L, 3L, 0L,

```



```

4L, 0L, 0L, 4L, 0L, 0L, 0L, 0L, 3L, 2L, 0L, 3L, 0L, 3L, 0L,
0L, 0L, 3L, 3L, 0L, 0L, 2L, 4L, 3L, 0L, 0L, 0L, 0L, 4L, 2L,
4L, 3L, 0L, 0L, 3L, 2L, 0L, 4L, 0L, 0L, 0L, 0L, 3L, 4L, 4L,
0L, 0L, 0L, 2L, 2L, 1L, 0L, 3L, 0L, 0L, 3L, 0L, 3L, 0L, 2L,
4L, 4L, 2L, 0L, 2L, 0L, 0L, 4L, 0L, 3L, 1L, 1L, 0L, 0L, 0L,
0L, 0L, 3L, 0L, 0L, 3L, 0L, 0L, 0L, 0L, 4L, 0L, 3L, 0L, 3L,
1L, 3L, 4L, 3L, 0L, 0L, 3L, 2L, 2L, 0L, 4L, 4L, 2L), Atr17 = c(3L,
4L, 3L, 4L, 1L, 0L, 3L, 3L, 0L, 3L, 0L, 3L, 0L, 3L, 4L, 0L,
3L, 0L, 0L, 3L, 0L, 1L, 0L, 0L, 3L, 3L, 0L, 3L, 0L, 4L, 0L,
0L, 0L, 3L, 4L, 0L, 0L, 3L, 3L, 3L, 0L, 0L, 0L, 0L, 4L, 1L,
3L, 3L, 0L, 1L, 4L, 3L, 0L, 3L, 0L, 0L, 1L, 0L, 3L, 4L, 4L,
0L, 0L, 0L, 3L, 4L, 2L, 0L, 4L, 0L, 1L, 4L, 0L, 3L, 0L, 3L,
3L, 3L, 2L, 0L, 3L, 0L, 0L, 4L, 0L, 3L, 0L, 0L, 0L, 0L, 0L,
0L, 0L, 3L, 0L, 0L, 3L, 0L, 0L, 0L, 0L, 3L, 0L, 4L, 0L, 4L,
1L, 4L, 3L, 4L, 0L, 2L, 3L, 3L, 3L, 0L, 3L, 3L, 4L), Atr18 = c(2L,
2L, 4L, 4L, 0L, 0L, 3L, 4L, 0L, 4L, 0L, 4L, 0L, 2L, 3L, 0L,
3L, 0L, 0L, 3L, 0L, 0L, 0L, 1L, 3L, 3L, 0L, 3L, 0L, 4L, 1L,
0L, 0L, 3L, 3L, 0L, 0L, 3L, 3L, 3L, 0L, 0L, 0L, 0L, 4L, 0L,
4L, 3L, 0L, 0L, 3L, 3L, 0L, 4L, 0L, 0L, 1L, 0L, 3L, 3L, 2L,
0L, 0L, 0L, 2L, 2L, 1L, 0L, 4L, 0L, 0L, 4L, 0L, 3L, 0L, 3L,
4L, 3L, 2L, 0L, 4L, 0L, 0L, 4L, 0L, 3L, 0L, 1L, 0L, 0L, 0L,
0L, 0L, 3L, 0L, 0L, 3L, 0L, 0L, 0L, 0L, 3L, 0L, 3L, 0L, 3L,
1L, 3L, 2L, 4L, 0L, 1L, 3L, 3L, 2L, 1L, 4L, 3L, 4L), Atr19 = c(3L,
4L, 3L, 4L, 0L, 1L, 4L, 3L, 0L, 3L, 0L, 3L, 2L, 3L, 4L, 0L,
3L, 0L, 0L, 3L, 0L, 1L, 0L, 1L, 3L, 3L, 0L, 3L, 0L, 4L, 0L,
0L, 0L, 3L, 4L, 0L, 0L, 3L, 3L, 3L, 0L, 0L, 0L, 0L, 4L, 0L,
3L, 3L, 0L, 0L, 4L, 3L, 1L, 3L, 0L, 0L, 0L, 0L, 3L, 4L, 4L,
0L, 0L, 0L, 3L, 4L, 0L, 0L, 4L, 0L, 1L, 4L, 0L, 3L, 0L, 3L,
3L, 3L, 2L, 1L, 3L, 0L, 0L, 3L, 0L, 3L, 0L, 0L, 0L, 0L, 0L,
0L, 0L, 3L, 0L, 0L, 3L, 0L, 0L, 0L, 0L, 3L, 0L, 4L, 0L, 4L,
1L, 4L, 3L, 4L, 0L, 0L, 3L, 3L, 3L, 0L, 3L, 3L, 4L), Atr20 = c(2L,
2L, 4L, 4L, 0L, 0L, 3L, 4L, 1L, 4L, 0L, 4L, 0L, 2L, 3L, 0L,
3L, 0L, 0L, 3L, 0L, 0L, 0L, 0L, 3L, 3L, 0L, 4L, 0L, 4L, 1L,
0L, 0L, 2L, 3L, 0L, 0L, 3L, 2L, 3L, 0L, 0L, 0L, 0L, 4L, 0L,
4L, 3L, 0L, 0L, 3L, 3L, 0L, 4L, 0L, 0L, 0L, 0L, 4L, 3L, 2L,
0L, 0L, 0L, 2L, 2L, 0L, 0L, 4L, 0L, 0L, 4L, 0L, 3L, 0L, 3L,
4L, 3L, 2L, 0L, 4L, 0L, 0L, 2L, 0L, 2L, 0L, 0L, 0L, 0L, 0L,
0L, 0L, 3L, 0L, 0L, 3L, 0L, 0L, 1L, 0L, 3L, 0L, 3L, 0L, 3L,
1L, 3L, 2L, 4L, 0L, 1L, 3L, 3L, 2L, 0L, 4L, 3L, 4L), Atr21 = c(2L,
3L, 2L, 4L, 0L, 0L, 3L, 3L, 0L, 3L, 0L, 3L, 0L, 3L, 2L, 1L,
4L, 1L, 0L, 3L, 0L, 0L, 0L, 0L, 2L, 3L, 1L, 2L, 0L, 4L, 0L,
0L, 0L, 2L, 2L, 0L, 0L, 3L, 2L, 2L, 0L, 0L, 0L, 0L, 0L, 1L,
4L, 2L, 0L, 1L, 3L, 3L, 0L, 3L, 0L, 0L, 0L, 0L, 2L, 3L, 3L,
0L, 0L, 0L, 3L, 3L, 0L, 0L, 4L, 0L, 0L, 4L, 0L, 2L, 0L, 3L,
3L, 3L, 3L, 1L, 3L, 1L, 0L, 1L, 0L, 3L, 0L, 0L, 0L, 0L, 0L,

```

```

0L, 0L, 3L, 1L, 0L, 4L, 0L, 0L, 1L, 0L, 3L, 0L, 3L, 0L, 3L,
1L, 4L, 3L, 4L, 0L, 0L, 3L, 3L, 3L, 0L, 3L, 4L, 3L), Atr22 = c(3L,
2L, 3L, 3L, 0L, 0L, 3L, 4L, 0L, 4L, 0L, 3L, 0L, 2L, 3L, 0L,
4L, 0L, 0L, 4L, 0L, 0L, 0L, 0L, 3L, 2L, 0L, 1L, 0L, 3L, 0L,
0L, 0L, 1L, 3L, 0L, 0L, 2L, 1L, 3L, 0L, 0L, 0L, 0L, 0L, 0L,
4L, 3L, 0L, 0L, 3L, 2L, 1L, 4L, 0L, 0L, 0L, 0L, 1L, 4L, 4L,
0L, 0L, 0L, 2L, 2L, 1L, 0L, 3L, 0L, 0L, 3L, 0L, 2L, 0L, 2L,
4L, 4L, 2L, 0L, 2L, 1L, 0L, 1L, 0L, 3L, 0L, 0L, 0L, 0L, 0L,
0L, 0L, 3L, 1L, 0L, 3L, 0L, 0L, 0L, 0L, 4L, 0L, 3L, 0L, 3L,
0L, 3L, 4L, 3L, 0L, 0L, 3L, 2L, 2L, 0L, 4L, 4L, 2L), Atr23 = c(3L,
4L, 3L, 0L, 0L, 0L, 3L, 3L, 0L, 3L, 0L, 3L, 0L, 3L, 4L, 0L,
3L, 0L, 0L, 3L, 0L, 0L, 0L, 0L, 3L, 3L, 0L, 0L, 0L, 4L, 0L,
0L, 0L, 1L, 4L, 0L, 0L, 3L, 2L, 3L, 0L, 0L, 0L, 0L, 2L, 0L,
3L, 3L, 0L, 0L, 4L, 3L, 1L, 3L, 0L, 0L, 0L, 0L, 0L, 4L, 4L,
0L, 0L, 0L, 3L, 4L, 0L, 0L, 4L, 0L, 0L, 4L, 0L, 2L, 0L, 3L,
3L, 3L, 2L, 0L, 3L, 0L, 2L, 0L, 0L, 3L, 0L, 0L, 0L, 0L, 0L,
0L, 0L, 3L, 1L, 0L, 3L, 0L, 0L, 1L, 0L, 3L, 0L, 4L, 0L, 4L,
0L, 4L, 3L, 4L, 0L, 0L, 3L, 3L, 3L, 2L, 3L, 3L, 4L), Atr24 = c(2L,
2L, 4L, 4L, 0L, 0L, 4L, 4L, 0L, 4L, 0L, 4L, 0L, 2L, 3L, 0L,
3L, 1L, 1L, 3L, 0L, 0L, 0L, 1L, 3L, 3L, 0L, 2L, 0L, 4L, 0L,
0L, 0L, 3L, 3L, 0L, 0L, 3L, 0L, 3L, 0L, 0L, 0L, 0L, 4L, 1L,
4L, 3L, 2L, 0L, 3L, 3L, 1L, 4L, 0L, 1L, 1L, 0L, 2L, 3L, 2L,
2L, 1L, 0L, 2L, 2L, 1L, 0L, 4L, 0L, 0L, 4L, 0L, 3L, 0L, 3L,
4L, 3L, 2L, 0L, 4L, 0L, 2L, 2L, 0L, 3L, 0L, 1L, 0L, 0L, 0L,
0L, 0L, 3L, 0L, 0L, 3L, 0L, 2L, 1L, 0L, 3L, 0L, 3L, 0L, 3L,
0L, 3L, 2L, 4L, 0L, 0L, 3L, 3L, 2L, 2L, 4L, 3L, 4L), Atr25 = c(3L,
4L, 3L, 0L, 0L, 0L, 3L, 3L, 0L, 3L, 0L, 3L, 0L, 3L, 4L, 1L,
3L, 1L, 0L, 3L, 0L, 0L, 0L, 0L, 3L, 3L, 1L, 2L, 0L, 4L, 0L,
0L, 0L, 3L, 4L, 1L, 0L, 3L, 0L, 3L, 0L, 0L, 0L, 0L, 4L, 0L,
3L, 3L, 0L, 0L, 4L, 3L, 1L, 3L, 1L, 1L, 1L, 1L, 2L, 4L, 4L,
0L, 1L, 0L, 3L, 4L, 1L, 0L, 4L, 1L, 0L, 4L, 1L, 2L, 1L, 3L,
3L, 3L, 2L, 0L, 3L, 0L, 0L, 2L, 0L, 2L, 2L, 1L, 0L, 0L, 0L,
0L, 1L, 3L, 1L, 0L, 3L, 0L, 0L, 0L, 1L, 3L, 1L, 4L, 0L, 4L,
0L, 4L, 3L, 4L, 1L, 2L, 3L, 3L, 3L, 1L, 3L, 3L, 4L), Atr26 = c(2L,
2L, 4L, 4L, 0L, 0L, 3L, 4L, 0L, 4L, 0L, 4L, 0L, 2L, 3L, 0L,
3L, 0L, 0L, 3L, 0L, 0L, 0L, 0L, 3L, 3L, 0L, 2L, 0L, 4L, 0L,
0L, 0L, 3L, 3L, 1L, 0L, 3L, 0L, 3L, 0L, 0L, 0L, 0L, 4L, 1L,
4L, 3L, 0L, 0L, 3L, 3L, 1L, 4L, 0L, 0L, 1L, 0L, 2L, 3L, 2L,
0L, 1L, 0L, 2L, 2L, 0L, 0L, 4L, 1L, 0L, 4L, 1L, 3L, 1L, 3L,
4L, 3L, 2L, 0L, 4L, 0L, 0L, 1L, 0L, 3L, 1L, 0L, 0L, 0L, 0L,
0L, 1L, 3L, 0L, 0L, 3L, 1L, 0L, 0L, 0L, 3L, 1L, 3L, 0L, 3L,
0L, 3L, 2L, 4L, 1L, 2L, 3L, 3L, 2L, 1L, 4L, 3L, 4L), Atr27 = c(2L,
3L, 2L, 4L, 0L, 0L, 2L, 3L, 0L, 3L, 0L, 3L, 0L, 3L, 2L, 0L,
4L, 0L, 0L, 3L, 0L, 0L, 0L, 1L, 2L, 3L, 1L, 2L, 0L, 4L, 0L,
0L, 0L, 2L, 2L, 1L, 0L, 3L, 0L, 2L, 0L, 0L, 0L, 4L, 0L,

```

```

4L, 2L, 0L, 0L, 3L, 3L, 0L, 3L, 0L, 0L, 0L, 0L, 2L, 3L, 3L,
0L, 0L, 0L, 3L, 3L, 1L, 0L, 4L, 0L, 0L, 4L, 0L, 2L, 0L, 3L,
3L, 3L, 3L, 0L, 3L, 0L, 1L, 2L, 0L, 3L, 1L, 0L, 0L, 0L, 0L,
0L, 0L, 3L, 0L, 0L, 4L, 1L, 0L, 0L, 0L, 3L, 1L, 3L, 0L, 3L,
0L, 4L, 3L, 4L, 1L, 0L, 3L, 3L, 3L, 1L, 3L, 4L, 3L), Atr28 = c(3L,
2L, 3L, 0L, 0L, 0L, 4L, 4L, 0L, 4L, 0L, 3L, 0L, 2L, 3L, 1L,
4L, 0L, 0L, 4L, 0L, 0L, 0L, 0L, 3L, 2L, 1L, 2L, 0L, 3L, 0L,
0L, 0L, 3L, 3L, 0L, 0L, 2L, 0L, 3L, 0L, 0L, 0L, 0L, 1L, 0L,
4L, 3L, 0L, 0L, 3L, 2L, 0L, 4L, 0L, 0L, 0L, 0L, 2L, 4L, 4L,
0L, 0L, 0L, 2L, 2L, 0L, 0L, 3L, 0L, 0L, 3L, 0L, 3L, 0L, 2L,
4L, 4L, 2L, 0L, 2L, 1L, 1L, 0L, 0L, 2L, 0L, 0L, 0L, 0L, 0L,
0L, 0L, 3L, 0L, 0L, 3L, 1L, 0L, 0L, 0L, 4L, 1L, 3L, 0L, 3L,
0L, 3L, 4L, 3L, 0L, 0L, 3L, 2L, 2L, 0L, 4L, 4L, 2L), Atr29 = c(3L,
4L, 3L, 4L, 0L, 0L, 3L, 3L, 0L, 3L, 0L, 3L, 0L, 3L, 4L, 1L,
3L, 1L, 0L, 3L, 0L, 0L, 0L, 0L, 3L, 3L, 1L, 2L, 0L, 4L, 0L,
0L, 0L, 2L, 4L, 0L, 0L, 3L, 2L, 3L, 0L, 0L, 0L, 0L, 1L, 0L,
3L, 3L, 0L, 0L, 4L, 3L, 0L, 3L, 0L, 0L, 0L, 0L, 2L, 4L, 4L,
0L, 0L, 0L, 3L, 4L, 0L, 0L, 4L, 0L, 0L, 4L, 1L, 2L, 0L, 3L,
3L, 3L, 2L, 0L, 3L, 0L, 0L, 1L, 0L, 2L, 0L, 0L, 0L, 0L, 0L,
0L, 1L, 3L, 0L, 0L, 3L, 0L, 0L, 0L, 0L, 3L, 1L, 4L, 0L, 4L,
0L, 4L, 3L, 4L, 0L, 0L, 3L, 3L, 3L, 1L, 3L, 3L, 4L), Atr30 = c(2L,
2L, 4L, 4L, 0L, 0L, 4L, 4L, 1L, 4L, 0L, 4L, 0L, 2L, 3L, 1L,
3L, 0L, 0L, 3L, 0L, 1L, 0L, 0L, 3L, 3L, 1L, 3L, 0L, 4L, 0L,
0L, 0L, 2L, 3L, 0L, 0L, 3L, 2L, 3L, 0L, 0L, 0L, 0L, 0L, 0L,
4L, 3L, 1L, 0L, 3L, 3L, 0L, 4L, 1L, 0L, 0L, 0L, 3L, 3L, 2L,
0L, 0L, 0L, 2L, 2L, 0L, 0L, 4L, 2L, 0L, 4L, 0L, 3L, 1L, 3L,
4L, 3L, 2L, 0L, 4L, 0L, 0L, 1L, 0L, 2L, 0L, 1L, 0L, 0L, 0L,
0L, 0L, 3L, 1L, 0L, 3L, 0L, 0L, 0L, 1L, 3L, 1L, 3L, 0L, 3L,
0L, 3L, 2L, 4L, 0L, 0L, 3L, 3L, 2L, 0L, 4L, 3L, 4L), Atr31 = c(4L,
4L, 4L, 0L, 0L, 2L, 4L, 3L, 1L, 4L, 0L, 3L, 0L, 4L, 3L, 1L,
4L, 2L, 0L, 3L, 1L, 0L, 0L, 0L, 4L, 4L, 0L, 3L, 4L, 4L, 1L,
0L, 2L, 3L, 4L, 1L, 0L, 4L, 3L, 4L, 2L, 2L, 2L, 0L, 4L, 2L,
4L, 4L, 0L, 0L, 4L, 4L, 2L, 4L, 2L, 0L, 0L, 1L, 3L, 4L, 4L,
0L, 2L, 0L, 4L, 4L, 0L, 0L, 4L, 0L, 0L, 4L, 0L, 1L, 1L, 3L,
4L, 4L, 4L, 0L, 4L, 0L, 1L, 0L, 2L, 1L, 0L, 1L, 0L, 2L, 1L,
0L, 0L, 4L, 1L, 1L, 4L, 2L, 0L, 1L, 0L, 4L, 1L, 4L, 0L, 4L,
2L, 4L, 4L, 4L, 0L, 4L, 3L, 4L, 4L, 0L, 3L, 3L, 4L), Atr32 = c(4L,
4L, 4L, 4L, 0L, 0L, 4L, 3L, 2L, 4L, 0L, 4L, 2L, 4L, 4L, 0L,
4L, 0L, 1L, 4L, 0L, 1L, 0L, 0L, 3L, 4L, 0L, 3L, 4L, 4L, 1L,
0L, 2L, 3L, 3L, 0L, 0L, 3L, 3L, 4L, 1L, 1L, 0L, 1L, 2L, 0L,
3L, 4L, 2L, 0L, 3L, 4L, 1L, 4L, 1L, 0L, 0L, 3L, 3L, 3L, 4L,
0L, 1L, 0L, 4L, 4L, 0L, 0L, 3L, 0L, 0L, 3L, 0L, 1L, 1L, 4L,
4L, 4L, 4L, 0L, 4L, 0L, 0L, 4L, 0L, 2L, 0L, 0L, 2L, 1L, 0L,
0L, 0L, 3L, 1L, 1L, 4L, 1L, 0L, 3L, 2L, 4L, 0L, 4L, 0L, 3L,
0L, 3L, 4L, 4L, 0L, 1L, 3L, 4L, 4L, 0L, 3L, 3L, 4L), Atr33 = c(4L,

```

```

4L, 4L, 4L, 0L, 0L, 4L, 3L, 1L, 4L, 0L, 3L, 0L, 4L, 3L, 0L,
4L, 0L, 1L, 3L, 0L, 0L, 0L, 0L, 4L, 4L, 0L, 3L, 3L, 4L, 1L,
0L, 0L, 3L, 4L, 0L, 1L, 4L, 3L, 4L, 2L, 1L, 0L, 0L, 0L, 0L,
4L, 4L, 0L, 0L, 4L, 4L, 0L, 4L, 0L, 0L, 1L, 0L, 3L, 4L, 4L,
0L, 1L, 0L, 4L, 4L, 0L, 0L, 4L, 0L, 0L, 4L, 0L, 1L, 2L, 3L,
4L, 4L, 4L, 0L, 4L, 0L, 0L, 2L, 0L, 2L, 0L, 0L, 0L, 1L, 0L,
0L, 0L, 4L, 1L, 0L, 4L, 0L, 0L, 0L, 0L, 4L, 0L, 4L, 0L, 4L,
0L, 4L, 4L, 4L, 0L, 1L, 3L, 4L, 4L, 0L, 3L, 3L, 4L), Atr34 = c(4L,
4L, 4L, 0L, 0L, 0L, 4L, 3L, 2L, 4L, 0L, 4L, 1L, 4L, 4L, 0L,
4L, 1L, 1L, 4L, 0L, 1L, 0L, 0L, 3L, 4L, 0L, 3L, 4L, 4L, 1L,
0L, 0L, 3L, 3L, 0L, 1L, 3L, 3L, 4L, 0L, 1L, 1L, 0L, 0L, 0L,
3L, 4L, 1L, 0L, 3L, 4L, 1L, 4L, 2L, 0L, 0L, 0L, 3L, 3L, 4L,
0L, 1L, 1L, 4L, 4L, 1L, 0L, 3L, 1L, 0L, 3L, 0L, 1L, 2L, 4L,
4L, 4L, 4L, 0L, 4L, 1L, 0L, 3L, 0L, 1L, 0L, 0L, 0L, 1L, 1L,
1L, 0L, 3L, 1L, 0L, 4L, 0L, 0L, 0L, 0L, 4L, 0L, 4L, 0L, 3L,
0L, 3L, 4L, 4L, 0L, 1L, 3L, 4L, 4L, 0L, 3L, 3L, 4L), Atr35 = c(4L,
4L, 4L, 1L, 0L, 0L, 4L, 3L, 0L, 4L, 0L, 3L, 0L, 4L, 3L, 0L,
4L, 0L, 0L, 3L, 0L, 0L, 0L, 0L, 4L, 4L, 0L, 3L, 0L, 4L, 0L,
0L, 0L, 3L, 4L, 0L, 0L, 4L, 3L, 4L, 0L, 0L, 0L, 0L, 0L, 0L,
4L, 4L, 0L, 0L, 4L, 4L, 0L, 4L, 0L, 0L, 1L, 0L, 3L, 4L, 4L,
0L, 0L, 0L, 4L, 4L, 0L, 0L, 4L, 0L, 0L, 4L, 0L, 1L, 0L, 3L,
4L, 4L, 4L, 0L, 4L, 0L, 0L, 0L, 0L, 1L, 0L, 0L, 0L, 0L, 0L,
0L, 0L, 4L, 0L, 0L, 4L, 0L, 0L, 0L, 0L, 4L, 0L, 4L, 0L, 4L,
0L, 4L, 4L, 4L, 0L, 1L, 3L, 4L, 4L, 0L, 3L, 3L, 4L), Atr36 = c(4L,
4L, 4L, 0L, 0L, 0L, 4L, 3L, 1L, 4L, 0L, 4L, 0L, 4L, 4L, 0L,
4L, 0L, 0L, 4L, 0L, 0L, 0L, 0L, 3L, 4L, 0L, 3L, 0L, 4L, 0L,
0L, 0L, 3L, 3L, 0L, 0L, 3L, 3L, 4L, 0L, 0L, 0L, 0L, 0L, 0L,
3L, 4L, 0L, 0L, 3L, 4L, 0L, 4L, 0L, 0L, 0L, 0L, 3L, 3L, 4L,
0L, 0L, 0L, 4L, 4L, 0L, 0L, 3L, 0L, 0L, 3L, 0L, 1L, 0L, 4L,
4L, 4L, 4L, 0L, 4L, 0L, 0L, 2L, 0L, 2L, 0L, 0L, 0L, 0L, 0L,
0L, 0L, 3L, 0L, 0L, 4L, 0L, 0L, 0L, 0L, 4L, 0L, 4L, 0L, 3L,
0L, 3L, 4L, 4L, 0L, 1L, 3L, 4L, 4L, 0L, 3L, 3L, 4L), Atr37 = c(4L,
4L, 4L, 2L, 0L, 0L, 4L, 3L, 2L, 4L, 0L, 3L, 0L, 4L, 3L, 0L,
4L, 1L, 0L, 3L, 0L, 1L, 0L, 0L, 4L, 4L, 0L, 4L, 4L, 4L, 1L,
1L, 0L, 3L, 4L, 0L, 0L, 4L, 3L, 4L, 2L, 1L, 1L, 0L, 3L, 2L,
4L, 4L, 0L, 0L, 4L, 4L, 1L, 4L, 1L, 0L, 1L, 2L, 4L, 4L, 4L,
0L, 2L, 0L, 4L, 4L, 0L, 0L, 4L, 1L, 0L, 4L, 0L, 1L, 0L, 3L,
4L, 4L, 4L, 0L, 4L, 1L, 0L, 3L, 4L, 3L, 0L, 0L, 0L, 0L, 0L,
1L, 0L, 4L, 1L, 0L, 4L, 1L, 1L, 0L, 0L, 4L, 0L, 4L, 2L, 4L,
0L, 4L, 4L, 4L, 0L, 1L, 3L, 4L, 4L, 0L, 3L, 3L, 4L), Atr38 = c(4L,
4L, 4L, 0L, 0L, 1L, 4L, 3L, 1L, 4L, 3L, 4L, 2L, 4L, 4L, 0L,
4L, 0L, 2L, 4L, 0L, 1L, 0L, 0L, 3L, 4L, 0L, 4L, 0L, 4L, 0L,
0L, 0L, 3L, 3L, 0L, 1L, 3L, 3L, 4L, 0L, 1L, 0L, 0L, 0L, 2L,
3L, 4L, 0L, 0L, 4L, 4L, 0L, 4L, 1L, 0L, 0L, 1L, 4L, 3L, 4L,
0L, 0L, 0L, 4L, 4L, 0L, 0L, 3L, 1L, 0L, 3L, 0L, 2L, 0L, 4L,

```

```

4L, 4L, 4L, 0L, 4L, 0L, 0L, 4L, 0L, 2L, 0L, 0L, 0L, 1L, 0L,
0L, 0L, 3L, 0L, 0L, 4L, 1L, 0L, 0L, 0L, 4L, 0L, 4L, 1L, 3L,
0L, 3L, 4L, 4L, 0L, 2L, 3L, 4L, 4L, 0L, 3L, 3L, 4L), Atr39 = c(4L,
4L, 4L, 4L, 0L, 0L, 4L, 3L, 3L, 4L, 0L, 3L, 0L, 4L, 3L, 0L,
4L, 2L, 1L, 3L, 0L, 1L, 0L, 1L, 4L, 4L, 0L, 4L, 1L, 4L, 1L,
0L, 0L, 4L, 4L, 0L, 1L, 4L, 1L, 4L, 1L, 2L, 0L, 0L, 4L, 1L,
4L, 4L, 0L, 0L, 3L, 4L, 0L, 4L, 1L, 0L, 1L, 0L, 4L, 4L, 4L,
0L, 1L, 0L, 4L, 4L, 0L, 0L, 4L, 2L, 1L, 4L, 0L, 2L, 0L, 3L,
4L, 4L, 4L, 0L, 4L, 1L, 0L, 2L, 0L, 2L, 0L, 0L, 0L, 1L, 0L,
4L, 0L, 4L, 0L, 0L, 4L, 2L, 0L, 0L, 0L, 4L, 0L, 4L, 1L, 4L,
0L, 4L, 4L, 4L, 0L, 0L, 3L, 4L, 4L, 0L, 3L, 3L, 4L), Atr40 = c(4L,
4L, 4L, 0L, 0L, 0L, 4L, 3L, 3L, 4L, 0L, 4L, 0L, 4L, 4L, 0L,
4L, 0L, 0L, 4L, 0L, 0L, 0L, 1L, 3L, 4L, 0L, 4L, 0L, 4L, 1L,
0L, 0L, 4L, 3L, 0L, 0L, 3L, 3L, 4L, 0L, 1L, 0L, 0L, 4L, 0L,
3L, 4L, 0L, 0L, 4L, 4L, 0L, 4L, 1L, 0L, 0L, 2L, 4L, 3L, 4L,
0L, 0L, 0L, 4L, 4L, 0L, 0L, 3L, 0L, 0L, 3L, 0L, 2L, 0L, 4L,
4L, 4L, 4L, 0L, 4L, 0L, 0L, 4L, 0L, 3L, 0L, 0L, 0L, 0L, 0L,
1L, 0L, 3L, 0L, 0L, 4L, 1L, 0L, 0L, 0L, 4L, 0L, 4L, 0L, 3L,
0L, 3L, 4L, 4L, 0L, 2L, 3L, 4L, 4L, 0L, 3L, 3L, 4L), Atr41 = c(4L,
4L, 4L, 2L, 0L, 0L, 4L, 3L, 2L, 4L, 1L, 3L, 2L, 4L, 3L, 0L,
4L, 1L, 0L, 3L, 0L, 0L, 0L, 0L, 4L, 4L, 0L, 3L, 1L, 4L, 1L,
0L, 0L, 4L, 4L, 0L, 0L, 4L, 2L, 4L, 0L, 1L, 1L, 0L, 4L, 0L,
4L, 4L, 0L, 1L, 4L, 4L, 1L, 4L, 0L, 0L, 0L, 0L, 4L, 4L, 4L,
0L, 1L, 0L, 4L, 4L, 0L, 1L, 4L, 1L, 0L, 4L, 1L, 2L, 1L, 3L,
4L, 4L, 4L, 0L, 4L, 0L, 3L, 2L, 0L, 3L, 0L, 0L, 0L, 0L, 0L,
2L, 0L, 4L, 0L, 0L, 4L, 0L, 0L, 0L, 0L, 4L, 2L, 4L, 1L, 4L,
0L, 4L, 4L, 4L, 0L, 2L, 3L, 4L, 4L, 0L, 3L, 3L, 4L), Atr42 = c(4L,
4L, 4L, 4L, 2L, 0L, 3L, 3L, 1L, 4L, 1L, 4L, 2L, 4L, 4L, 1L,
4L, 3L, 0L, 4L, 0L, 4L, 1L, 0L, 3L, 4L, 1L, 0L, 0L, 4L, 1L,
1L, 4L, 4L, 3L, 0L, 3L, 3L, 3L, 4L, 0L, 3L, 1L, 0L, 4L, 2L,
3L, 4L, 0L, 0L, 4L, 4L, 0L, 4L, 2L, 0L, 0L, 0L, 3L, 3L, 4L,
0L, 0L, 0L, 4L, 4L, 0L, 2L, 3L, 2L, 2L, 3L, 0L, 2L, 0L, 4L,
4L, 4L, 3L, 0L, 4L, 0L, 1L, 2L, 0L, 3L, 0L, 2L, 0L, 3L, 2L,
1L, 2L, 3L, 0L, 0L, 4L, 2L, 0L, 2L, 2L, 4L, 0L, 4L, 3L, 3L,
0L, 3L, 4L, 4L, 0L, 1L, 3L, 4L, 4L, 0L, 3L, 3L, 4L), Atr43 = c(4L,
4L, 4L, 4L, 4L, 2L, 4L, 3L, 1L, 4L, 1L, 3L, 3L, 4L, 3L, 3L,
4L, 2L, 2L, 3L, 0L, 2L, 1L, 3L, 4L, 4L, 2L, 0L, 0L, 4L, 1L,
2L, 4L, 3L, 4L, 2L, 2L, 4L, 3L, 4L, 2L, 2L, 2L, 4L, 4L, 2L,
4L, 4L, 3L, 1L, 4L, 4L, 2L, 4L, 2L, 1L, 1L, 0L, 0L, 4L, 4L,
0L, 0L, 3L, 4L, 4L, 0L, 4L, 4L, 3L, 2L, 4L, 2L, 2L, 1L, 3L,
4L, 4L, 4L, 3L, 4L, 2L, 1L, 3L, 0L, 3L, 1L, 4L, 0L, 2L, 2L,
1L, 3L, 4L, 0L, 2L, 4L, 2L, 2L, 2L, 2L, 4L, 3L, 4L, 4L, 4L,
2L, 4L, 4L, 4L, 3L, 2L, 3L, 4L, 4L, 4L, 3L, 3L, 4L), Atr44 = c(4L,
4L, 4L, 1L, 4L, 0L, 3L, 3L, 2L, 4L, 0L, 4L, 0L, 4L, 4L, 2L,
4L, 0L, 1L, 4L, 2L, 0L, 1L, 2L, 3L, 4L, 0L, 3L, 0L, 4L, 1L,

```

```

0L, 0L, 4L, 3L, 1L, 1L, 3L, 4L, 4L, 0L, 0L, 0L, 0L, 4L, 1L,
3L, 4L, 0L, 0L, 3L, 4L, 0L, 4L, 0L, 0L, 0L, 0L, 3L, 4L,
0L, 0L, 0L, 4L, 4L, 0L, 2L, 3L, 0L, 0L, 3L, 0L, 2L, 2L, 4L,
3L, 4L, 4L, 0L, 4L, 2L, 0L, 4L, 0L, 4L, 1L, 1L, 0L, 1L, 0L,
0L, 1L, 3L, 0L, 0L, 4L, 2L, 2L, 0L, 2L, 4L, 0L, 4L, 0L, 3L,
0L, 3L, 4L, 4L, 1L, 3L, 3L, 4L, 4L, 4L, 3L, 3L, 4L), Atr45 = c(4L,
4L, 3L, 4L, 4L, 2L, 3L, 3L, 3L, 4L, 3L, 3L, 3L, 4L, 3L, 1L,
4L, 1L, 1L, 3L, 1L, 1L, 2L, 4L, 4L, 3L, 2L, 0L, 0L, 4L, 1L,
1L, 4L, 2L, 4L, 0L, 3L, 3L, 4L, 4L, 1L, 2L, 1L, 4L, 4L, 2L,
3L, 4L, 0L, 0L, 4L, 4L, 0L, 4L, 0L, 3L, 0L, 0L, 0L, 4L, 4L,
0L, 0L, 0L, 4L, 4L, 1L, 3L, 4L, 1L, 1L, 4L, 1L, 2L, 0L, 3L,
4L, 3L, 4L, 3L, 4L, 2L, 1L, 2L, 0L, 3L, 0L, 0L, 0L, 2L, 3L,
4L, 3L, 4L, 0L, 0L, 3L, 1L, 4L, 1L, 2L, 4L, 3L, 4L, 4L, 4L,
2L, 4L, 3L, 4L, 0L, 0L, 3L, 4L, 4L, 4L, 3L, 3L, 4L), Atr46 = c(4L,
4L, 4L, 0L, 4L, 0L, 3L, 3L, 2L, 4L, 3L, 4L, 2L, 4L, 4L, 3L,
4L, 3L, 1L, 4L, 4L, 2L, 3L, 4L, 3L, 4L, 1L, 0L, 1L, 4L, 0L,
4L, 1L, 2L, 3L, 3L, 3L, 3L, 2L, 4L, 2L, 1L, 2L, 3L, 2L, 3L,
3L, 4L, 2L, 0L, 4L, 4L, 2L, 4L, 2L, 2L, 3L, 0L, 0L, 3L, 4L,
0L, 0L, 0L, 4L, 4L, 1L, 1L, 3L, 1L, 0L, 3L, 1L, 1L, 1L, 4L,
4L, 4L, 4L, 4L, 4L, 0L, 3L, 2L, 3L, 3L, 4L, 2L, 2L, 2L, 3L,
3L, 3L, 3L, 0L, 2L, 4L, 0L, 4L, 3L, 2L, 4L, 3L, 4L, 3L, 3L,
2L, 3L, 4L, 4L, 3L, 2L, 3L, 4L, 4L, 4L, 3L, 3L, 4L), Atr47 = c(4L,
4L, 4L, 4L, 4L, 2L, 4L, 3L, 1L, 4L, 2L, 3L, 0L, 4L, 3L, 0L,
4L, 3L, 1L, 3L, 2L, 1L, 1L, 0L, 4L, 4L, 0L, 0L, 0L, 4L, 0L,
4L, 4L, 3L, 4L, 0L, 0L, 4L, 3L, 4L, 1L, 2L, 2L, 1L, 4L, 0L,
4L, 4L, 2L, 1L, 3L, 4L, 1L, 4L, 1L, 0L, 0L, 0L, 0L, 4L, 4L,
0L, 0L, 0L, 4L, 4L, 1L, 1L, 4L, 2L, 1L, 4L, 1L, 1L, 0L, 3L,
4L, 4L, 4L, 3L, 4L, 0L, 3L, 2L, 0L, 2L, 1L, 1L, 2L, 1L, 3L,
3L, 3L, 4L, 0L, 0L, 4L, 1L, 4L, 4L, 2L, 4L, 1L, 4L, 4L, 4L,
1L, 4L, 4L, 4L, 0L, 2L, 3L, 4L, 4L, 4L, 3L, 3L, 4L), Atr48 = c(4L,
4L, 4L, 4L, 4L, 3L, 4L, 3L, 3L, 4L, 2L, 4L, 2L, 4L, 4L, 2L,
4L, 2L, 2L, 4L, 3L, 3L, 0L, 1L, 3L, 4L, 0L, 3L, 4L, 4L, 2L,
3L, 2L, 3L, 3L, 2L, 2L, 3L, 2L, 4L, 3L, 3L, 2L, 3L, 4L, 2L,
3L, 4L, 0L, 0L, 4L, 4L, 2L, 4L, 2L, 2L, 2L, 3L, 3L, 3L, 4L,
4L, 2L, 2L, 4L, 4L, 0L, 2L, 3L, 2L, 2L, 3L, 1L, 1L, 3L, 4L,
4L, 4L, 4L, 0L, 4L, 2L, 2L, 3L, 2L, 3L, 1L, 2L, 3L, 2L, 3L,
2L, 2L, 3L, 2L, 2L, 4L, 3L, 2L, 4L, 2L, 4L, 2L, 4L, 3L, 3L,
1L, 3L, 4L, 4L, 1L, 1L, 3L, 4L, 4L, 0L, 3L, 3L, 4L), Atr49 = c(4L,
4L, 4L, 4L, 4L, 0L, 4L, 3L, 3L, 4L, 0L, 3L, 4L, 4L, 3L, 3L,
4L, 2L, 1L, 3L, 0L, 1L, 0L, 0L, 4L, 4L, 1L, 3L, 1L, 4L, 1L,
3L, 3L, 2L, 4L, 3L, 3L, 4L, 2L, 4L, 3L, 2L, 2L, 1L, 3L, 0L,
4L, 4L, 0L, 0L, 4L, 4L, 2L, 4L, 1L, 2L, 2L, 1L, 3L, 4L, 4L,
4L, 2L, 0L, 4L, 4L, 1L, 0L, 4L, 1L, 0L, 4L, 0L, 1L, 1L, 3L,
4L, 4L, 4L, 2L, 4L, 2L, 0L, 4L, 2L, 2L, 4L, 1L, 3L, 3L, 0L,
2L, 1L, 4L, 1L, 3L, 4L, 2L, 2L, 0L, 0L, 4L, 1L, 4L, 1L, 4L,

```

```

1L, 4L, 4L, 4L, 0L, 2L, 3L, 4L, 4L, 4L, 3L, 3L, 4L), Atr50 = c(4L,
4L, 4L, 3L, 2L, 3L, 4L, 3L, 3L, 4L, 2L, 3L, 0L, 4L, 3L, 3L,
4L, 1L, 1L, 3L, 0L, 1L, 1L, 0L, 4L, 4L, 1L, 3L, 1L, 4L, 2L,
1L, 2L, 1L, 4L, 4L, 2L, 4L, 3L, 4L, 2L, 2L, 2L, 3L, 4L, 0L,
4L, 4L, 1L, 0L, 3L, 4L, 2L, 4L, 2L, 2L, 2L, 1L, 3L, 4L, 4L,
4L, 2L, 0L, 4L, 4L, 2L, 0L, 4L, 0L, 0L, 4L, 0L, 1L, 1L, 3L,
4L, 4L, 4L, 2L, 4L, 2L, 0L, 4L, 2L, 3L, 4L, 2L, 3L, 2L, 1L,
3L, 1L, 4L, 2L, 2L, 4L, 2L, 2L, 1L, 2L, 4L, 1L, 4L, 3L, 4L,
1L, 4L, 4L, 4L, 0L, 1L, 3L, 4L, 4L, 0L, 3L, 3L, 4L), Atr51 = c(4L,
4L, 4L, 4L, 2L, 3L, 3L, 3L, 2L, 4L, 2L, 4L, 0L, 4L, 4L, 2L,
4L, 1L, 1L, 4L, 0L, 1L, 1L, 1L, 3L, 4L, 1L, 3L, 4L, 4L, 2L,
2L, 2L, 1L, 3L, 3L, 2L, 3L, 3L, 4L, 2L, 2L, 2L, 3L, 4L, 2L,
3L, 4L, 1L, 1L, 3L, 4L, 2L, 4L, 2L, 2L, 2L, 1L, 3L, 3L, 4L,
4L, 2L, 1L, 4L, 4L, 2L, 1L, 3L, 1L, 0L, 3L, 0L, 1L, 3L, 4L,
4L, 4L, 4L, 2L, 4L, 2L, 0L, 4L, 2L, 3L, 2L, 2L, 3L, 2L, 3L,
2L, 1L, 3L, 2L, 2L, 4L, 2L, 2L, 3L, 1L, 4L, 1L, 4L, 3L, 3L,
1L, 3L, 4L, 4L, 0L, 1L, 3L, 4L, 4L, 0L, 3L, 3L, 4L), Atr52 = c(4L,
4L, 4L, 4L, 0L, 3L, 3L, 3L, 3L, 4L, 0L, 3L, 1L, 4L, 3L, 1L,
4L, 2L, 0L, 3L, 2L, 1L, 0L, 4L, 4L, 4L, 1L, 3L, 2L, 4L, 2L,
3L, 0L, 2L, 4L, 3L, 4L, 4L, 4L, 4L, 2L, 1L, 2L, 3L, 1L, 1L,
4L, 4L, 3L, 0L, 4L, 4L, 2L, 4L, 1L, 2L, 2L, 0L, 3L, 4L, 4L,
0L, 1L, 2L, 4L, 4L, 0L, 4L, 4L, 3L, 3L, 4L, 1L, 1L, 1L, 3L,
4L, 4L, 4L, 0L, 4L, 1L, 4L, 4L, 4L, 2L, 2L, 4L, 1L, 3L, 3L,
4L, 1L, 4L, 3L, 1L, 4L, 1L, 4L, 3L, 1L, 4L, 0L, 4L, 0L, 4L,
1L, 4L, 4L, 4L, 2L, 1L, 3L, 4L, 4L, 0L, 3L, 3L, 4L), Atr53 = c(4L,
4L, 4L, 4L, 0L, 2L, 4L, 3L, 2L, 4L, 0L, 4L, 0L, 4L, 4L, 0L,
4L, 0L, 0L, 4L, 4L, 1L, 0L, 1L, 3L, 4L, 1L, 3L, 2L, 4L, 2L,
1L, 0L, 2L, 3L, 0L, 1L, 3L, 2L, 4L, 2L, 1L, 2L, 1L, 1L, 0L,
3L, 4L, 4L, 0L, 4L, 4L, 1L, 4L, 0L, 2L, 2L, 3L, 3L, 3L, 4L,
0L, 0L, 2L, 4L, 4L, 1L, 1L, 3L, 2L, 2L, 3L, 1L, 1L, 1L, 4L,
4L, 4L, 4L, 0L, 4L, 1L, 4L, 2L, 4L, 2L, 0L, 2L, 1L, 1L, 3L,
3L, 1L, 3L, 3L, 1L, 4L, 0L, 2L, 3L, 1L, 4L, 1L, 4L, 1L, 3L,
1L, 3L, 4L, 4L, 0L, 2L, 3L, 4L, 4L, 0L, 3L, 3L, 4L), Atr54 = c(4L,
4L, 4L, 4L, 0L, 2L, 4L, 3L, 1L, 4L, 4L, 3L, 0L, 4L, 3L, 0L,
4L, 0L, 0L, 3L, 1L, 0L, 0L, 0L, 4L, 4L, 1L, 3L, 2L, 4L, 3L,
1L, 0L, 2L, 4L, 2L, 1L, 4L, 2L, 4L, 0L, 0L, 2L, 0L, 0L, 0L,
4L, 4L, 0L, 1L, 3L, 4L, 0L, 4L, 0L, 2L, 1L, 0L, 3L, 4L, 4L,
0L, 0L, 1L, 4L, 4L, 1L, 2L, 4L, 2L, 1L, 4L, 1L, 1L, 1L, 3L,
4L, 4L, 4L, 0L, 4L, 0L, 2L, 2L, 0L, 2L, 1L, 0L, 0L, 1L, 1L,
1L, 1L, 4L, 2L, 0L, 4L, 0L, 2L, 1L, 0L, 4L, 0L, 4L, 0L, 4L,
1L, 4L, 4L, 4L, 0L, 0L, 3L, 4L, 4L, 0L, 3L, 3L, 4L), Class = c(1L,
1L, 1L, 1L, 0L, 0L, 1L, 1L, 1L, 1L, 0L, 1L, 0L, 1L, 1L, 0L,
1L, 0L, 0L, 1L, 0L, 0L, 0L, 0L, 1L, 1L, 0L, 1L, 0L, 1L, 0L,
0L, 0L, 1L, 1L, 0L, 0L, 1L, 1L, 1L, 0L, 0L, 0L, 0L, 1L, 0L,
1L, 1L, 0L, 0L, 1L, 1L, 0L, 1L, 0L, 0L, 0L, 1L, 1L, 1L,

```

```
0L, 0L, 0L, 1L, 1L, 0L, 0L, 1L, 0L, 0L, 1L, 0L, 1L, 0L, 1L,
1L, 1L, 1L, 0L, 1L, 0L, 0L, 1L, 0L, 1L, 0L, 0L, 0L, 0L,
0L, 0L, 1L, 0L, 0L, 1L, 0L, 0L, 0L, 0L, 1L, 0L, 1L, 0L, 1L,
0L, 1L, 1L, 1L, 0L, 1L, 1L, 1L, 1L, 0L, 1L, 1L, 1L)), n.trees = 1000L,
cv.folds = 5L)
```

A gradient boosted model with bernoulli loss function.

1000 iterations were performed.

The best cross-validation iteration was 36.

There were 54 predictors of which 16 had non-zero influence.

In [116...

```
names(gbmfit)
```

Out[116...

```
29-element Vector{Symbol}:
 :initF
 :fit
 Symbol("train.error")
 Symbol("valid.error")
 Symbol("oobag.improve")
 :trees
 Symbol("c.splits")
 Symbol("bag.fraction")
 :distribution
 Symbol("interaction.depth")
 Symbol("n.minobsinnode")
 Symbol("num.classes")
 Symbol("n.trees")
 :
 Symbol("var.levels")
 Symbol("var.monotone")
 Symbol("var.names")
 Symbol("var.type")
 :verbose
 :data
 :Terms
 Symbol("cv.error")
 Symbol("cv.folds")
 :call
 :m
 Symbol("cv.fitted")
```

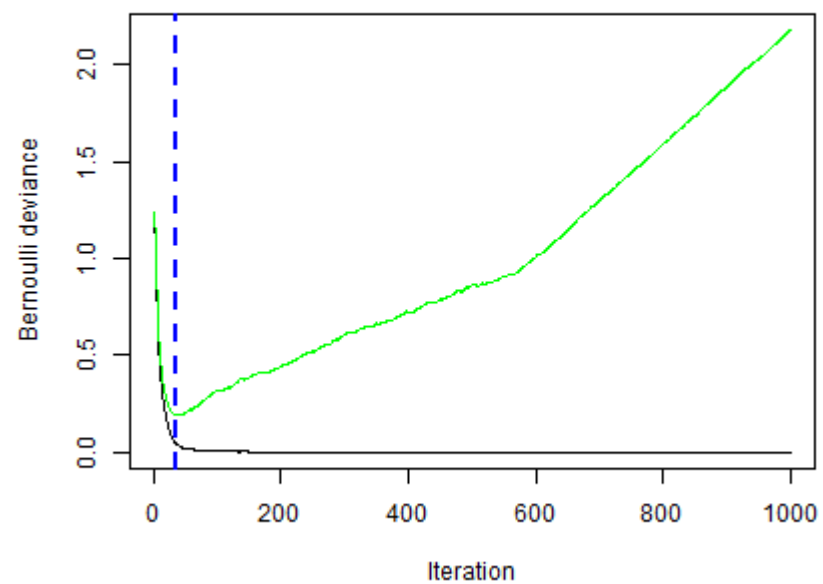
Cross Validation Error

In [117...

```
sqrt(minimum(gbmfit[Symbol("cv.error")]))
```

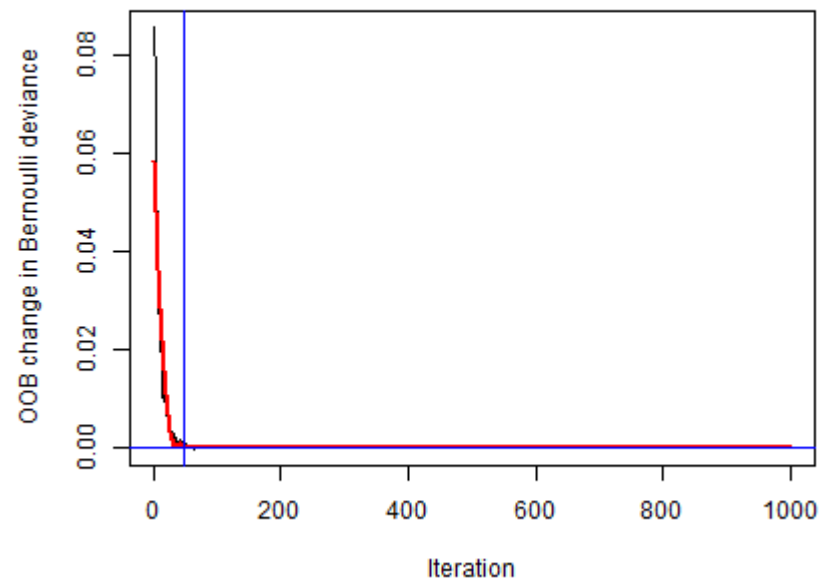
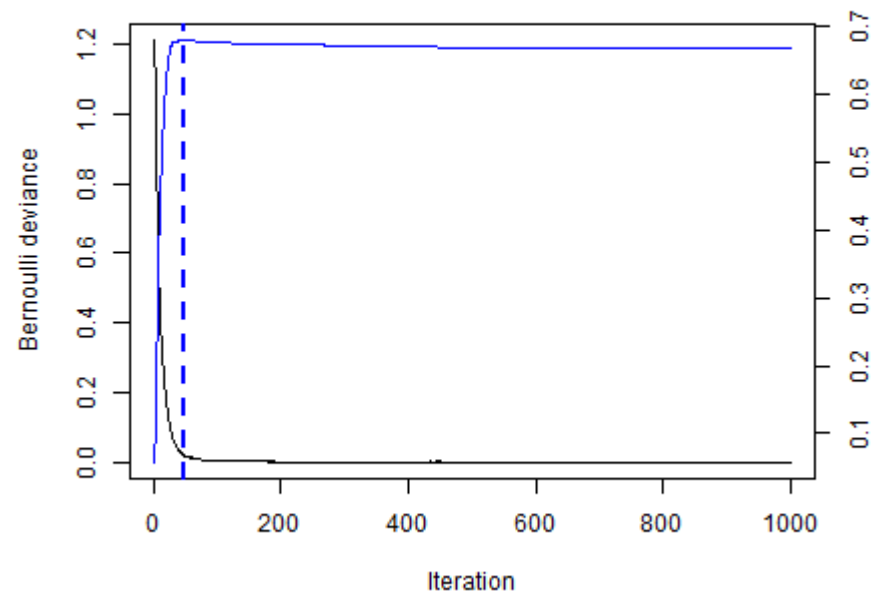

Out[117... 0.4374217524927952

In [118... `gbm_perf(gbmfit,method="cv")`



Out[118... RObject{IntSxp}
[1] 36

In [119... `gbm_perf(gbmfit, var"oobag.curve"=true,method="OOB")`



Warning: RCall.jl: OOB generally underestimates the optimal number of iterations although predictive performance is reasonably competitive. Using cv_folds>1 when calling gbm usually results in improved predictive performance.

@ RCall C:\Users\ASUS\julia\packages\RCall\6kphM\src\io.jl:172

Out[119... RObject{IntSxp}
[1] 48
attr("smoother")
Call:
loess(formula = object\$oobag.improve ~ x, enp.target = min(max(4,
length(x)/10), 50))

Number of Observations: 1000
Equivalent Number of Parameters: 40
Residual Standard Error: 0.001532

In [120... *# tuning more parameters*
creating hyperparameter grid

@rimport base as rbase

hyper_grid = rbase.expand_grid(shrinkage = [.01, .1, .3], interaction_depth = [1, 3, 5],
minobsinnode = [5, 10, 15],
bag_fraction = [.65, .8, 1],
optimal_trees = 0.0, *# a place to dump results*
min_RMSE = 0.0 *# a place to dump results*
);

rbase.dim(hyper_grid)

Out[120... RObject{IntSxp}
[1] 81 6

In [121... **for** i **in** 1:rbase.dim(hyper_grid)[1]
gbm_tune=gbm(m,distribution="bernoulli",data=traindf,var"n.trees"=1000,
var"interaction.depth"=hyper_grid[:interaction_depth][i],
shrinkage=hyper_grid[:shrinkage][i],
var"n.minobsinnode"=hyper_grid[:minobsinnode][i],
var"bag.fraction"=hyper_grid[:bag_fraction][i],
var"train.fraction"=0.6)

hyper_grid[:optimal_trees][i]= rbase.which_min(gbm_tune[Symbol("valid.error")])

hyper_grid[:min_RMSE][i]= sqrt(minimum(gbm_tune[Symbol("valid.error")]))
println(i);

end

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44

45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81

```
In [122... minindex=rbase.which_min(hyper_grid[:min_RMSE])  
  
i=convert(Int,minindex)
```

Out[122... 12

```
In [123... gbmfitfinal = gbm(m,distribution="bernoulli",data=traindf,var"n.trees"=1000,
var"interaction.depth"=hyper_grid[:interaction_depth][i],
shrinkage=hyper_grid[:shrinkage][i],
var"n.minobsinnode"=hyper_grid[:minobsinnode][i],
var"bag.fraction"=hyper_grid[:bag_fraction][i],
var"train.fraction"=0.6)
```

```
Out[123... RObject{VecSxp}
(function (formula = formula(data), distribution = "bernoulli",
  data = list(), weights, var.monotone = NULL, n.trees = 100,
  interaction.depth = 1, n.minobsinnode = 10, shrinkage = 0.1,
  bag.fraction = 0.5, train.fraction = 1, cv.folds = 0, keep.data = TRUE,
  verbose = FALSE, class.stratify.cv = NULL, n.cores = NULL)
{
  mcall <- match.call()
  lVerbose <- if (!is.logical(verbose)) {
    FALSE
  }
  else {
    verbose
  }
  mf <- match.call(expand.dots = FALSE)
  m <- match(c("formula", "data", "weights",
    "offset"), names(mf), 0)
  mf <- mf[c(1, m)]
  mf$drop.unused.levels <- TRUE
  mf$na.action <- na.pass
  mf[[1]] <- as.name("model.frame")
  m <- mf
  mf <- eval(mf, parent.frame())
  Terms <- attr(mf, "terms")
  w <- model.weights(mf)
  offset <- model.offset(mf)
  y <- model.response(mf)
  if (missing(distribution)) {
    distribution <- guessDist(y)
  }
  if (is.character(distribution)) {
    distribution <- list(name = distribution)
  }
  if (!is.element(distribution$name, getAvailableDistributions())) {
    stop("Distribution ", distribution$name, " is not supported.")
  }
  if (distribution$name == "multinomial") {
    warning("Setting `distribution = \"multinomial\"` is ill-advised as it is ",
```

```

      "currently broken. It exists only for backwards compatibility. ",
      "Use at your own risk.", call. = FALSE)
}
var.names <- attributes(Terms)$term.labels
x <- model.frame(terms(reformulate(var.names)), data = data,
  na.action = na.pass)
response.name <- as.character(formula[[2L]])
class.stratify.cv <- getStratify(class.stratify.cv, d = distribution)
group <- NULL
num.groups <- 0
if (distribution$name != "pairwise") {
  nTrain <- floor(train.fraction * nrow(x))
}
else {
  distribution.group <- distribution[["group"]]
  if (is.null(distribution.group)) {
    stop(paste("For pairwise regression, `distribution` must be a list of",
      "the form `list(name = \"pairwise\", group = c(\"date\",",
      "\"session\", \"category\", \"keywords\")`.")"))
  }
  i <- match(distribution.group, colnames(data))
  if (any(is.na(i))) {
    stop("Group column does not occur in data: ",
      distribution.group[is.na(i)], ".")
  }
  group <- factor(do.call(paste, c(data[, distribution.group,
    drop = FALSE], sep = ":")))
  if ((!missing(weights)) && (!is.null(weights))) {
    w.min <- tapply(w, INDEX = group, FUN = min)
    w.max <- tapply(w, INDEX = group, FUN = max)
    if (any(w.min != w.max)) {
      stop("For `distribution = \"pairwise\"`, all instances for the same ",
        "group must have the same weight.")
    }
    w <- w * length(w.min)/sum(w.min)
  }
  perm.levels <- levels(group)[sample(1:nlevels(group))]
  group <- factor(group, levels = perm.levels)
  ord.group <- order(group, -y)
  group <- group[ord.group]
  y <- y[ord.group]
  x <- x[ord.group, , drop = FALSE]
  w <- w[ord.group]
  num.groups.train <- max(1, round(train.fraction * nlevels(group)))
  nTrain <- max(which(group == levels(group)[num.groups.train]))

```

```

      Misc <- group
    }
    cv.error <- NULL
    if (cv.folds == 1) {
      cv.folds <- 0
    }
    if (cv.folds > 1) {
      cv.results <- gbmCrossVal(cv.folds = cv.folds, nTrain = nTrain,
        n.cores = n.cores, class.stratify.cv = class.stratify.cv,
        data = data, x = x, y = y, offset = offset, distribution = distribution,
        w = w, var.monotone = var.monotone, n.trees = n.trees,
        interaction.depth = interaction.depth, n.minobsinnode = n.minobsinnode,
        shrinkage = shrinkage, bag.fraction = bag.fraction,
        var.names = var.names, response.name = response.name,
        group = group)
      cv.error <- cv.results$error
      p <- cv.results$predictions
    }
    gbm.obj <- gbm.fit(x = x, y = y, offset = offset, distribution = distribution,
      w = w, var.monotone = var.monotone, n.trees = n.trees,
      interaction.depth = interaction.depth, n.minobsinnode = n.minobsinnode,
      shrinkage = shrinkage, bag.fraction = bag.fraction, nTrain = nTrain,
      keep.data = keep.data, verbose = lVerbose, var.names = var.names,
      response.name = response.name, group = group)
    gbm.obj$train.fraction <- train.fraction
    gbm.obj$Terms <- Terms
    gbm.obj$cv.error <- cv.error
    gbm.obj$cv.folds <- cv.folds
    gbm.obj$call <- mcall
    gbm.obj$m <- m
    if (cv.folds > 1) {
      gbm.obj$cv.fitted <- p
    }
    if (distribution$name == "pairwise") {
      gbm.obj$ord.group <- ord.group
      gbm.obj$fit <- gbm.obj$fit[order(ord.group)]
    }
    gbm.obj
  })(formula = Class ~ Atr1 + Atr2 + Atr3 + Atr4 + Atr5 + Atr6 +
    Atr7 + Atr8 + Atr9 + Atr10 + Atr11 + Atr12 + Atr13 + Atr14 +
    Atr15 + Atr16 + Atr17 + Atr18 + Atr19 + Atr20 + Atr21 + Atr22 +
    Atr23 + Atr24 + Atr25 + Atr26 + Atr27 + Atr28 + Atr29 + Atr30 +
    Atr31 + Atr32 + Atr33 + Atr34 + Atr35 + Atr36 + Atr37 + Atr38 +
    Atr39 + Atr40 + Atr41 + Atr42 + Atr43 + Atr44 + Atr45 + Atr46 +
    Atr47 + Atr48 + Atr49 + Atr50 + Atr51 + Atr52 + Atr53 + Atr54,

```



```

distribution = "bernoulli", data = list(Atr1 = c(3L,
4L, 3L, 4L, 0L, 1L, 3L, 3L, 2L, 3L, 3L, 3L, 0L, 3L, 4L, 0L,
3L, 0L, 0L, 3L, 0L, 0L, 0L, 3L, 3L, 3L, 1L, 3L, 0L, 4L, 1L,
0L, 0L, 2L, 4L, 0L, 0L, 3L, 2L, 3L, 0L, 0L, 0L, 0L, 4L, 2L,
3L, 3L, 0L, 0L, 4L, 3L, 1L, 3L, 0L, 0L, 2L, 0L, 3L, 4L, 4L,
4L, 0L, 0L, 3L, 4L, 1L, 0L, 4L, 0L, 0L, 4L, 0L, 2L, 2L, 3L,
3L, 3L, 2L, 0L, 3L, 0L, 0L, 4L, 0L, 3L, 0L, 1L, 1L, 0L, 0L,
0L, 0L, 3L, 0L, 0L, 3L, 0L, 0L, 0L, 0L, 3L, 0L, 4L, 0L, 4L,
2L, 4L, 3L, 4L, 0L, 0L, 3L, 3L, 3L, 0L, 3L, 3L, 4L), Atr2 = c(2L,
2L, 4L, 4L, 1L, 0L, 3L, 4L, 2L, 4L, 1L, 4L, 0L, 2L, 3L, 0L,
3L, 0L, 1L, 3L, 0L, 0L, 0L, 0L, 3L, 3L, 1L, 2L, 0L, 4L, 0L,
1L, 0L, 2L, 3L, 0L, 0L, 3L, 0L, 3L, 0L, 0L, 0L, 0L, 4L, 0L,
4L, 3L, 0L, 0L, 3L, 3L, 2L, 4L, 0L, 0L, 1L, 2L, 2L, 3L, 2L,
3L, 0L, 0L, 2L, 2L, 1L, 0L, 4L, 2L, 1L, 4L, 0L, 2L, 1L, 3L,
4L, 3L, 2L, 0L, 4L, 0L, 1L, 4L, 0L, 3L, 0L, 0L, 1L, 0L, 0L,
0L, 0L, 3L, 0L, 0L, 3L, 1L, 0L, 0L, 1L, 3L, 0L, 3L, 0L, 3L,
2L, 3L, 2L, 4L, 1L, 0L, 3L, 3L, 2L, 0L, 4L, 3L, 4L), Atr3 = c(2L,
3L, 2L, 4L, 0L, 1L, 2L, 3L, 4L, 3L, 1L, 3L, 2L, 3L, 2L, 0L,
4L, 0L, 0L, 3L, 0L, 0L, 0L, 0L, 2L, 3L, 0L, 4L, 0L, 4L, 1L,
1L, 0L, 2L, 2L, 0L, 0L, 3L, 2L, 2L, 1L, 0L, 0L, 0L, 3L, 1L,
4L, 2L, 1L, 0L, 3L, 3L, 0L, 3L, 1L, 0L, 0L, 2L, 4L, 3L, 3L,
4L, 0L, 0L, 3L, 3L, 2L, 0L, 4L, 0L, 2L, 4L, 0L, 1L, 1L, 3L,
3L, 3L, 3L, 0L, 3L, 0L, 0L, 4L, 0L, 3L, 0L, 0L, 1L, 0L, 1L,
0L, 0L, 3L, 0L, 0L, 4L, 0L, 0L, 0L, 1L, 3L, 0L, 3L, 0L, 3L,
3L, 4L, 3L, 4L, 0L, 1L, 3L, 3L, 3L, 1L, 3L, 4L, 3L), Atr4 = c(3L,
2L, 3L, 3L, 0L, 0L, 3L, 4L, 1L, 4L, 0L, 3L, 4L, 2L, 3L, 0L,
4L, 0L, 1L, 4L, 0L, 0L, 0L, 0L, 3L, 2L, 0L, 3L, 0L, 3L, 0L,
0L, 0L, 3L, 3L, 0L, 0L, 2L, 4L, 3L, 0L, 0L, 0L, 0L, 3L, 0L,
4L, 3L, 0L, 0L, 3L, 2L, 0L, 4L, 1L, 0L, 2L, 1L, 3L, 4L, 4L,
4L, 0L, 0L, 2L, 2L, 0L, 0L, 3L, 0L, 1L, 3L, 0L, 0L, 0L, 2L,
4L, 4L, 2L, 0L, 2L, 0L, 0L, 4L, 0L, 2L, 0L, 0L, 1L, 0L, 0L,
0L, 0L, 3L, 0L, 0L, 3L, 1L, 0L, 0L, 0L, 4L, 0L, 3L, 0L, 3L,
0L, 3L, 4L, 3L, 0L, 0L, 3L, 2L, 2L, 0L, 4L, 4L, 2L), Atr5 = c(3L,
4L, 3L, 4L, 0L, 0L, 3L, 3L, 0L, 3L, 0L, 3L, 0L, 3L, 4L, 0L,
3L, 0L, 0L, 3L, 0L, 0L, 0L, 0L, 3L, 3L, 0L, 3L, 0L, 4L, 1L,
0L, 0L, 2L, 4L, 0L, 0L, 3L, 2L, 3L, 1L, 0L, 0L, 0L, 4L, 0L,
3L, 3L, 0L, 0L, 4L, 3L, 0L, 3L, 0L, 0L, 0L, 3L, 4L, 4L,
0L, 0L, 0L, 3L, 4L, 2L, 0L, 4L, 0L, 0L, 4L, 0L, 0L, 0L, 3L,
3L, 3L, 2L, 0L, 3L, 0L, 0L, 4L, 0L, 1L, 0L, 0L, 1L, 0L, 0L,
0L, 1L, 3L, 0L, 0L, 3L, 0L, 0L, 0L, 0L, 3L, 0L, 4L, 0L, 4L,
0L, 4L, 3L, 4L, 0L, 0L, 3L, 3L, 3L, 0L, 3L, 3L, 4L), Atr6 = c(2L,
2L, 0L, 2L, 0L, 0L, 1L, 0L, 0L, 0L, 0L, 0L, 0L, 2L, 1L, 0L,
1L, 0L, 0L, 1L, 2L, 2L, 0L, 0L, 1L, 1L, 0L, 2L, 0L, 0L, 1L,
1L, 0L, 3L, 1L, 2L, 0L, 1L, 2L, 1L, 0L, 0L, 1L, 0L, 0L, 0L,
0L, 1L, 1L, 0L, 1L, 1L, 0L, 0L, 0L, 0L, 0L, 2L, 1L, 2L,

```

```

0L, 0L, 0L, 2L, 2L, 1L, 0L, 0L, 1L, 0L, 0L, 0L, 4L, 2L, 1L,
0L, 1L, 2L, 1L, 0L, 0L, 1L, 0L, 0L, 3L, 1L, 1L, 0L, 0L, 0L,
0L, 0L, 1L, 1L, 0L, 1L, 0L, 2L, 0L, 2L, 1L, 0L, 1L, 0L, 1L,
1L, 1L, 2L, 0L, 0L, 2L, 1L, 1L, 2L, 0L, 0L, 1L, 0L), Atr7 = c(1L,
0L, 1L, 4L, 0L, 0L, 1L, 1L, 0L, 1L, 0L, 1L, 0L, 1L, 0L, 0L,
1L, 0L, 0L, 1L, 1L, 0L, 0L, 0L, 1L, 1L, 0L, 3L, 0L, 0L, 0L,
0L, 0L, 2L, 0L, 0L, 0L, 1L, 4L, 1L, 0L, 0L, 0L, 0L, 4L, 0L,
1L, 1L, 0L, 0L, 0L, 1L, 0L, 1L, 0L, 0L, 0L, 0L, 3L, 0L, 0L,
0L, 0L, 0L, 1L, 0L, 0L, 0L, 0L, 0L, 0L, 0L, 0L, 1L, 0L, 1L,
1L, 1L, 2L, 0L, 1L, 0L, 0L, 0L, 0L, 4L, 0L, 0L, 0L, 0L, 0L,
0L, 0L, 1L, 0L, 0L, 1L, 0L, 0L, 0L, 0L, 1L, 0L, 0L, 0L, 0L,
0L, 0L, 1L, 0L, 0L, 0L, 1L, 1L, 1L, 0L, 1L, 1L, 0L), Atr8 = c(2L,
2L, 4L, 4L, 0L, 1L, 3L, 4L, 0L, 4L, 0L, 4L, 0L, 2L, 3L, 0L,
3L, 0L, 0L, 3L, 0L, 0L, 0L, 0L, 3L, 3L, 0L, 2L, 0L, 4L, 0L,
0L, 0L, 1L, 3L, 0L, 0L, 3L, 3L, 3L, 0L, 0L, 0L, 0L, 2L, 0L,
4L, 3L, 0L, 0L, 3L, 3L, 0L, 4L, 0L, 0L, 0L, 0L, 2L, 3L, 2L,
0L, 0L, 0L, 2L, 2L, 2L, 0L, 4L, 0L, 0L, 4L, 0L, 3L, 0L, 3L,
4L, 3L, 2L, 0L, 4L, 0L, 2L, 4L, 0L, 3L, 0L, 0L, 1L, 0L, 1L,
0L, 0L, 3L, 1L, 0L, 3L, 0L, 0L, 1L, 0L, 3L, 0L, 3L, 0L, 3L,
0L, 3L, 2L, 4L, 1L, 0L, 3L, 3L, 2L, 0L, 4L, 3L, 4L), Atr9 = c(2L,
3L, 2L, 4L, 0L, 0L, 3L, 3L, 0L, 3L, 0L, 3L, 0L, 3L, 2L, 0L,
4L, 0L, 0L, 3L, 0L, 0L, 0L, 0L, 2L, 3L, 0L, 2L, 0L, 4L, 0L,
0L, 0L, 3L, 2L, 0L, 0L, 3L, 4L, 2L, 0L, 0L, 0L, 0L, 4L, 0L,
4L, 2L, 0L, 0L, 3L, 3L, 0L, 3L, 0L, 0L, 0L, 0L, 2L, 3L, 3L,
0L, 0L, 0L, 3L, 3L, 1L, 0L, 4L, 0L, 0L, 4L, 0L, 3L, 0L, 3L,
3L, 3L, 3L, 0L, 3L, 0L, 0L, 4L, 0L, 2L, 0L, 0L, 1L, 0L, 1L,
0L, 0L, 3L, 1L, 0L, 4L, 0L, 0L, 0L, 0L, 3L, 0L, 3L, 0L, 3L,
0L, 4L, 3L, 4L, 0L, 0L, 3L, 3L, 3L, 0L, 3L, 4L, 3L), Atr10 = c(3L,
2L, 3L, 3L, 0L, 0L, 3L, 4L, 0L, 4L, 0L, 3L, 2L, 2L, 3L, 0L,
4L, 2L, 1L, 4L, 0L, 1L, 0L, 0L, 3L, 2L, 1L, 3L, 0L, 3L, 1L,
0L, 1L, 2L, 3L, 0L, 0L, 2L, 3L, 3L, 0L, 1L, 0L, 0L, 4L, 2L,
4L, 3L, 0L, 0L, 3L, 2L, 0L, 4L, 2L, 1L, 1L, 2L, 3L, 4L, 4L,
0L, 1L, 0L, 2L, 2L, 2L, 0L, 3L, 0L, 0L, 3L, 0L, 3L, 2L, 2L,
4L, 4L, 2L, 0L, 2L, 0L, 0L, 4L, 0L, 2L, 1L, 1L, 1L, 0L, 0L,
0L, 0L, 3L, 0L, 0L, 3L, 1L, 0L, 0L, 0L, 4L, 0L, 3L, 0L, 3L,
1L, 3L, 4L, 3L, 0L, 1L, 3L, 2L, 2L, 0L, 4L, 4L, 2L), Atr11 = c(3L,
4L, 3L, 4L, 0L, 1L, 4L, 3L, 1L, 3L, 1L, 3L, 0L, 3L, 4L, 0L,
3L, 0L, 0L, 3L, 0L, 0L, 0L, 0L, 3L, 3L, 0L, 4L, 0L, 4L, 0L,
0L, 0L, 1L, 4L, 0L, 0L, 3L, 2L, 3L, 0L, 0L, 0L, 0L, 3L, 1L,
3L, 3L, 0L, 0L, 4L, 3L, 0L, 3L, 1L, 0L, 1L, 1L, 4L, 4L, 4L,
0L, 0L, 0L, 3L, 4L, 1L, 0L, 4L, 0L, 0L, 4L, 0L, 3L, 0L, 3L,
3L, 3L, 2L, 1L, 3L, 0L, 1L, 4L, 0L, 2L, 0L, 1L, 1L, 0L, 0L,
0L, 0L, 3L, 0L, 0L, 3L, 1L, 0L, 0L, 0L, 3L, 0L, 4L, 0L, 4L,
1L, 4L, 3L, 4L, 0L, 0L, 3L, 3L, 3L, 0L, 3L, 3L, 4L), Atr12 = c(2L,
2L, 4L, 4L, 0L, 0L, 3L, 4L, 0L, 4L, 0L, 4L, 0L, 2L, 3L, 0L,

```

```

3L, 1L, 0L, 3L, 0L, 0L, 0L, 1L, 3L, 3L, 0L, 3L, 0L, 4L, 0L,
0L, 1L, 2L, 3L, 0L, 0L, 3L, 3L, 3L, 1L, 1L, 0L, 0L, 4L, 0L,
4L, 3L, 0L, 1L, 3L, 3L, 0L, 4L, 2L, 0L, 1L, 1L, 3L, 3L, 2L,
0L, 0L, 0L, 2L, 2L, 1L, 0L, 4L, 0L, 1L, 4L, 0L, 3L, 1L, 3L,
4L, 3L, 2L, 1L, 4L, 1L, 0L, 3L, 0L, 2L, 1L, 0L, 0L, 0L, 0L,
1L, 0L, 3L, 0L, 0L, 3L, 2L, 0L, 1L, 2L, 3L, 0L, 3L, 0L, 3L,
1L, 3L, 2L, 4L, 0L, 2L, 3L, 3L, 2L, 0L, 4L, 3L, 4L), Atr13 = c(3L,
4L, 3L, 4L, 0L, 1L, 3L, 3L, 1L, 3L, 0L, 3L, 0L, 3L, 4L, 1L,
3L, 1L, 0L, 3L, 0L, 0L, 0L, 1L, 3L, 3L, 1L, 2L, 0L, 4L, 1L,
0L, 0L, 2L, 4L, 1L, 0L, 3L, 3L, 3L, 0L, 2L, 1L, 0L, 4L, 1L,
3L, 3L, 0L, 0L, 4L, 3L, 0L, 3L, 1L, 0L, 1L, 1L, 2L, 4L, 4L,
0L, 0L, 1L, 3L, 4L, 2L, 0L, 4L, 0L, 1L, 4L, 0L, 3L, 1L, 3L,
3L, 3L, 2L, 1L, 3L, 1L, 1L, 4L, 2L, 2L, 1L, 1L, 0L, 0L, 0L,
0L, 1L, 3L, 1L, 0L, 3L, 2L, 0L, 1L, 1L, 3L, 0L, 4L, 0L, 4L,
1L, 4L, 3L, 4L, 0L, 1L, 3L, 3L, 3L, 0L, 3L, 3L, 4L), Atr14 = c(2L,
2L, 4L, 4L, 0L, 0L, 3L, 4L, 1L, 4L, 1L, 4L, 0L, 2L, 3L, 1L,
3L, 0L, 1L, 3L, 0L, 0L, 0L, 2L, 3L, 3L, 0L, 2L, 0L, 4L, 1L,
0L, 0L, 2L, 3L, 1L, 0L, 3L, 0L, 3L, 0L, 0L, 0L, 0L, 4L, 0L,
4L, 3L, 0L, 0L, 3L, 3L, 1L, 4L, 1L, 0L, 0L, 0L, 3L, 3L, 2L,
0L, 0L, 0L, 2L, 2L, 0L, 0L, 4L, 0L, 0L, 4L, 0L, 3L, 1L, 3L,
4L, 3L, 2L, 0L, 4L, 1L, 0L, 0L, 0L, 3L, 0L, 1L, 0L, 0L, 1L,
0L, 0L, 3L, 1L, 0L, 3L, 0L, 0L, 1L, 0L, 3L, 0L, 3L, 0L, 3L,
1L, 3L, 2L, 4L, 1L, 0L, 3L, 3L, 2L, 0L, 4L, 3L, 4L), Atr15 = c(2L,
3L, 2L, 4L, 0L, 0L, 3L, 3L, 0L, 3L, 1L, 3L, 0L, 3L, 2L, 0L,
4L, 0L, 0L, 3L, 0L, 1L, 0L, 1L, 2L, 3L, 0L, 1L, 0L, 4L, 1L,
0L, 0L, 3L, 2L, 0L, 0L, 3L, 4L, 2L, 0L, 1L, 0L, 0L, 4L, 2L,
4L, 2L, 0L, 0L, 3L, 3L, 0L, 3L, 1L, 0L, 1L, 0L, 1L, 3L, 3L,
0L, 0L, 0L, 3L, 3L, 2L, 0L, 4L, 0L, 0L, 4L, 0L, 3L, 1L, 3L,
3L, 3L, 3L, 0L, 3L, 1L, 0L, 4L, 0L, 2L, 0L, 1L, 0L, 0L, 0L,
0L, 1L, 3L, 0L, 0L, 4L, 0L, 0L, 0L, 0L, 3L, 0L, 3L, 0L, 3L,
1L, 4L, 3L, 4L, 0L, 2L, 3L, 3L, 3L, 0L, 3L, 4L, 3L), Atr16 = c(3L,
2L, 3L, 4L, 0L, 1L, 3L, 4L, 1L, 4L, 1L, 3L, 0L, 2L, 3L, 0L,
4L, 0L, 0L, 4L, 0L, 0L, 0L, 0L, 3L, 2L, 0L, 3L, 0L, 3L, 0L,
0L, 0L, 3L, 3L, 0L, 0L, 2L, 4L, 3L, 0L, 0L, 0L, 0L, 4L, 2L,
4L, 3L, 0L, 0L, 3L, 2L, 0L, 4L, 0L, 0L, 0L, 0L, 3L, 4L, 4L,
0L, 0L, 0L, 2L, 2L, 1L, 0L, 3L, 0L, 0L, 3L, 0L, 3L, 0L, 2L,
4L, 4L, 2L, 0L, 2L, 0L, 0L, 4L, 0L, 3L, 1L, 1L, 0L, 0L, 0L,
0L, 0L, 3L, 0L, 0L, 3L, 0L, 0L, 0L, 0L, 4L, 0L, 3L, 0L, 3L,
1L, 3L, 4L, 3L, 0L, 0L, 3L, 2L, 2L, 0L, 4L, 4L, 2L), Atr17 = c(3L,
4L, 3L, 4L, 1L, 0L, 3L, 3L, 0L, 3L, 0L, 3L, 0L, 3L, 4L, 0L,
3L, 0L, 0L, 3L, 0L, 1L, 0L, 0L, 3L, 3L, 0L, 3L, 0L, 4L, 0L,
0L, 0L, 3L, 4L, 0L, 0L, 3L, 3L, 3L, 0L, 0L, 0L, 0L, 4L, 1L,
3L, 3L, 0L, 1L, 4L, 3L, 0L, 3L, 0L, 0L, 1L, 0L, 3L, 4L, 4L,
0L, 0L, 0L, 3L, 4L, 2L, 0L, 4L, 0L, 1L, 4L, 0L, 3L, 0L, 3L,
3L, 3L, 2L, 0L, 3L, 0L, 0L, 4L, 0L, 3L, 0L, 0L, 0L, 0L,

```

```

0L, 0L, 3L, 0L, 0L, 3L, 0L, 0L, 0L, 0L, 3L, 0L, 4L, 0L, 4L,
1L, 4L, 3L, 4L, 0L, 2L, 3L, 3L, 3L, 0L, 3L, 3L, 4L), Atr18 = c(2L,
2L, 4L, 4L, 0L, 0L, 3L, 4L, 0L, 4L, 0L, 4L, 0L, 2L, 3L, 0L,
3L, 0L, 0L, 3L, 0L, 0L, 0L, 1L, 3L, 3L, 0L, 3L, 0L, 4L, 1L,
0L, 0L, 3L, 3L, 0L, 0L, 3L, 3L, 3L, 0L, 0L, 0L, 0L, 4L, 0L,
4L, 3L, 0L, 0L, 3L, 3L, 0L, 4L, 0L, 0L, 1L, 0L, 3L, 3L, 2L,
0L, 0L, 0L, 2L, 2L, 1L, 0L, 4L, 0L, 0L, 4L, 0L, 3L, 0L, 3L,
4L, 3L, 2L, 0L, 4L, 0L, 0L, 4L, 0L, 3L, 0L, 1L, 0L, 0L, 0L,
0L, 0L, 3L, 0L, 0L, 3L, 0L, 0L, 0L, 0L, 3L, 0L, 3L, 0L, 3L,
1L, 3L, 2L, 4L, 0L, 1L, 3L, 3L, 2L, 1L, 4L, 3L, 4L), Atr19 = c(3L,
4L, 3L, 4L, 0L, 1L, 4L, 3L, 0L, 3L, 0L, 3L, 2L, 3L, 4L, 0L,
3L, 0L, 0L, 3L, 0L, 1L, 0L, 1L, 3L, 3L, 0L, 3L, 0L, 4L, 0L,
0L, 0L, 3L, 4L, 0L, 0L, 3L, 3L, 3L, 0L, 0L, 0L, 0L, 4L, 0L,
3L, 3L, 0L, 0L, 4L, 3L, 1L, 3L, 0L, 0L, 0L, 0L, 3L, 4L, 4L,
0L, 0L, 0L, 3L, 4L, 0L, 0L, 4L, 0L, 1L, 4L, 0L, 3L, 0L, 3L,
3L, 3L, 2L, 1L, 3L, 0L, 0L, 3L, 0L, 3L, 0L, 0L, 0L, 0L, 0L,
0L, 0L, 3L, 0L, 0L, 3L, 0L, 0L, 0L, 0L, 3L, 0L, 4L, 0L, 4L,
1L, 4L, 3L, 4L, 0L, 0L, 3L, 3L, 3L, 0L, 3L, 3L, 4L), Atr20 = c(2L,
2L, 4L, 4L, 0L, 0L, 3L, 4L, 1L, 4L, 0L, 4L, 0L, 2L, 3L, 0L,
3L, 0L, 0L, 3L, 0L, 0L, 0L, 0L, 3L, 3L, 0L, 4L, 0L, 4L, 1L,
0L, 0L, 2L, 3L, 0L, 0L, 3L, 2L, 3L, 0L, 0L, 0L, 0L, 4L, 0L,
4L, 3L, 0L, 0L, 3L, 3L, 0L, 4L, 0L, 0L, 0L, 0L, 4L, 3L, 2L,
0L, 0L, 0L, 2L, 2L, 0L, 0L, 4L, 0L, 0L, 4L, 0L, 3L, 0L, 3L,
4L, 3L, 2L, 0L, 4L, 0L, 0L, 2L, 0L, 2L, 0L, 0L, 0L, 0L, 0L,
0L, 0L, 3L, 0L, 0L, 3L, 0L, 0L, 1L, 0L, 3L, 0L, 3L, 0L, 3L,
1L, 3L, 2L, 4L, 0L, 1L, 3L, 3L, 2L, 0L, 4L, 3L, 4L), Atr21 = c(2L,
3L, 2L, 4L, 0L, 0L, 3L, 3L, 0L, 3L, 0L, 3L, 0L, 3L, 2L, 1L,
4L, 1L, 0L, 3L, 0L, 0L, 0L, 0L, 2L, 3L, 1L, 2L, 0L, 4L, 0L,
0L, 0L, 2L, 2L, 0L, 0L, 3L, 2L, 2L, 0L, 0L, 0L, 0L, 0L, 1L,
4L, 2L, 0L, 1L, 3L, 3L, 0L, 3L, 0L, 0L, 0L, 0L, 2L, 3L, 3L,
0L, 0L, 0L, 3L, 3L, 0L, 0L, 4L, 0L, 0L, 4L, 0L, 2L, 0L, 3L,
3L, 3L, 3L, 1L, 3L, 1L, 0L, 1L, 0L, 3L, 0L, 0L, 0L, 0L, 0L,
0L, 0L, 3L, 1L, 0L, 4L, 0L, 0L, 1L, 0L, 3L, 0L, 3L, 0L, 3L,
1L, 4L, 3L, 4L, 0L, 0L, 3L, 3L, 3L, 0L, 3L, 4L, 3L), Atr22 = c(3L,
2L, 3L, 3L, 0L, 0L, 3L, 4L, 0L, 4L, 0L, 3L, 0L, 2L, 3L, 0L,
4L, 0L, 0L, 4L, 0L, 0L, 0L, 0L, 3L, 2L, 0L, 1L, 0L, 3L, 0L,
0L, 0L, 1L, 3L, 0L, 0L, 2L, 1L, 3L, 0L, 0L, 0L, 0L, 0L, 0L,
4L, 3L, 0L, 0L, 3L, 2L, 1L, 4L, 0L, 0L, 0L, 0L, 1L, 4L, 4L,
0L, 0L, 0L, 2L, 2L, 1L, 0L, 3L, 0L, 0L, 3L, 0L, 2L, 0L, 2L,
4L, 4L, 2L, 0L, 2L, 1L, 0L, 1L, 0L, 3L, 0L, 0L, 0L, 0L, 0L,
0L, 0L, 3L, 1L, 0L, 3L, 0L, 0L, 0L, 0L, 4L, 0L, 3L, 0L, 3L,
0L, 3L, 4L, 3L, 0L, 0L, 3L, 2L, 2L, 0L, 4L, 4L, 2L), Atr23 = c(3L,
4L, 3L, 0L, 0L, 0L, 3L, 3L, 0L, 3L, 0L, 3L, 0L, 3L, 4L, 0L,
3L, 0L, 0L, 3L, 0L, 0L, 0L, 0L, 3L, 3L, 0L, 0L, 0L, 4L, 0L,
0L, 0L, 1L, 4L, 0L, 0L, 3L, 2L, 3L, 0L, 0L, 0L, 0L, 2L, 0L,

```

```

3L, 3L, 0L, 0L, 4L, 3L, 1L, 3L, 0L, 0L, 0L, 0L, 0L, 4L, 4L,
0L, 0L, 0L, 3L, 4L, 0L, 0L, 4L, 0L, 0L, 4L, 0L, 2L, 0L, 3L,
3L, 3L, 2L, 0L, 3L, 0L, 2L, 0L, 0L, 3L, 0L, 0L, 0L, 0L, 0L,
0L, 0L, 3L, 1L, 0L, 3L, 0L, 0L, 1L, 0L, 3L, 0L, 4L, 0L, 4L,
0L, 4L, 3L, 4L, 0L, 0L, 3L, 3L, 3L, 2L, 3L, 3L, 4L), Atr24 = c(2L,
2L, 4L, 4L, 0L, 0L, 4L, 4L, 0L, 4L, 0L, 4L, 0L, 2L, 3L, 0L,
3L, 1L, 1L, 3L, 0L, 0L, 0L, 1L, 3L, 3L, 0L, 2L, 0L, 4L, 0L,
0L, 0L, 3L, 3L, 0L, 0L, 3L, 0L, 3L, 0L, 0L, 0L, 0L, 4L, 1L,
4L, 3L, 2L, 0L, 3L, 3L, 1L, 4L, 0L, 1L, 1L, 0L, 2L, 3L, 2L,
2L, 1L, 0L, 2L, 2L, 1L, 0L, 4L, 0L, 0L, 4L, 0L, 3L, 0L, 3L,
4L, 3L, 2L, 0L, 4L, 0L, 2L, 2L, 0L, 3L, 0L, 1L, 0L, 0L, 0L,
0L, 0L, 3L, 0L, 0L, 3L, 0L, 2L, 1L, 0L, 3L, 0L, 3L, 0L, 3L,
0L, 3L, 2L, 4L, 0L, 0L, 3L, 3L, 2L, 2L, 4L, 3L, 4L), Atr25 = c(3L,
4L, 3L, 0L, 0L, 0L, 3L, 3L, 0L, 3L, 0L, 3L, 0L, 3L, 4L, 1L,
3L, 1L, 0L, 3L, 0L, 0L, 0L, 0L, 3L, 3L, 1L, 2L, 0L, 4L, 0L,
0L, 0L, 3L, 4L, 1L, 0L, 3L, 0L, 3L, 0L, 0L, 0L, 0L, 4L, 0L,
3L, 3L, 0L, 0L, 4L, 3L, 1L, 3L, 1L, 1L, 1L, 1L, 2L, 4L, 4L,
0L, 1L, 0L, 3L, 4L, 1L, 0L, 4L, 1L, 0L, 4L, 1L, 2L, 1L, 3L,
3L, 3L, 2L, 0L, 3L, 0L, 0L, 2L, 0L, 2L, 2L, 1L, 0L, 0L, 0L,
0L, 1L, 3L, 1L, 0L, 3L, 0L, 0L, 0L, 1L, 3L, 1L, 4L, 0L, 4L,
0L, 4L, 3L, 4L, 1L, 2L, 3L, 3L, 3L, 1L, 3L, 3L, 4L), Atr26 = c(2L,
2L, 4L, 4L, 0L, 0L, 3L, 4L, 0L, 4L, 0L, 4L, 0L, 2L, 3L, 0L,
3L, 0L, 0L, 3L, 0L, 0L, 0L, 0L, 3L, 3L, 0L, 2L, 0L, 4L, 0L,
0L, 0L, 3L, 3L, 1L, 0L, 3L, 0L, 3L, 0L, 0L, 0L, 0L, 4L, 1L,
4L, 3L, 0L, 0L, 3L, 3L, 1L, 4L, 0L, 0L, 1L, 0L, 2L, 3L, 2L,
0L, 1L, 0L, 2L, 2L, 0L, 0L, 4L, 1L, 0L, 4L, 1L, 3L, 1L, 3L,
4L, 3L, 2L, 0L, 4L, 0L, 0L, 1L, 0L, 3L, 1L, 0L, 0L, 0L, 0L,
0L, 1L, 3L, 0L, 0L, 3L, 1L, 0L, 0L, 0L, 3L, 1L, 3L, 0L, 3L,
0L, 3L, 2L, 4L, 1L, 2L, 3L, 3L, 2L, 1L, 4L, 3L, 4L), Atr27 = c(2L,
3L, 2L, 4L, 0L, 0L, 2L, 3L, 0L, 3L, 0L, 3L, 0L, 3L, 2L, 0L,
4L, 0L, 0L, 3L, 0L, 0L, 0L, 1L, 2L, 3L, 1L, 2L, 0L, 4L, 0L,
0L, 0L, 2L, 2L, 1L, 0L, 3L, 0L, 2L, 0L, 0L, 0L, 0L, 4L, 0L,
4L, 2L, 0L, 0L, 3L, 3L, 0L, 3L, 0L, 0L, 0L, 0L, 2L, 3L, 3L,
0L, 0L, 0L, 3L, 3L, 1L, 0L, 4L, 0L, 0L, 4L, 0L, 2L, 0L, 3L,
3L, 3L, 3L, 0L, 3L, 0L, 1L, 2L, 0L, 3L, 1L, 0L, 0L, 0L, 0L,
0L, 0L, 3L, 0L, 0L, 4L, 1L, 0L, 0L, 0L, 3L, 1L, 3L, 0L, 3L,
0L, 4L, 3L, 4L, 1L, 0L, 3L, 3L, 3L, 1L, 3L, 4L, 3L), Atr28 = c(3L,
2L, 3L, 0L, 0L, 0L, 4L, 4L, 0L, 4L, 0L, 3L, 0L, 2L, 3L, 1L,
4L, 0L, 0L, 4L, 0L, 0L, 0L, 0L, 3L, 2L, 1L, 2L, 0L, 3L, 0L,
0L, 0L, 3L, 3L, 0L, 0L, 2L, 0L, 3L, 0L, 0L, 0L, 0L, 1L, 0L,
4L, 3L, 0L, 0L, 3L, 2L, 0L, 4L, 0L, 0L, 0L, 0L, 2L, 4L, 4L,
0L, 0L, 0L, 2L, 2L, 0L, 0L, 3L, 0L, 0L, 3L, 0L, 3L, 0L, 2L,
4L, 4L, 2L, 0L, 2L, 1L, 1L, 0L, 0L, 2L, 0L, 0L, 0L, 0L, 0L,
0L, 0L, 3L, 0L, 0L, 3L, 1L, 0L, 0L, 0L, 4L, 1L, 3L, 0L, 3L,
0L, 3L, 4L, 3L, 0L, 0L, 3L, 2L, 2L, 0L, 4L, 4L, 2L), Atr29 = c(3L,

```

```

4L, 3L, 4L, 0L, 0L, 3L, 3L, 0L, 3L, 0L, 3L, 0L, 3L, 4L, 1L,
3L, 1L, 0L, 3L, 0L, 0L, 0L, 0L, 3L, 3L, 1L, 2L, 0L, 4L, 0L,
0L, 0L, 2L, 4L, 0L, 0L, 3L, 2L, 3L, 0L, 0L, 0L, 0L, 1L, 0L,
3L, 3L, 0L, 0L, 4L, 3L, 0L, 3L, 0L, 0L, 0L, 0L, 2L, 4L, 4L,
0L, 0L, 0L, 3L, 4L, 0L, 0L, 4L, 0L, 0L, 4L, 1L, 2L, 0L, 3L,
3L, 3L, 2L, 0L, 3L, 0L, 0L, 1L, 0L, 2L, 0L, 0L, 0L, 0L, 0L,
0L, 1L, 3L, 0L, 0L, 3L, 0L, 0L, 0L, 0L, 3L, 1L, 4L, 0L, 4L,
0L, 4L, 3L, 4L, 0L, 0L, 3L, 3L, 3L, 1L, 3L, 3L, 4L), Atr30 = c(2L,
2L, 4L, 4L, 0L, 0L, 4L, 4L, 1L, 4L, 0L, 4L, 0L, 2L, 3L, 1L,
3L, 0L, 0L, 3L, 0L, 1L, 0L, 0L, 3L, 3L, 1L, 3L, 0L, 4L, 0L,
0L, 0L, 2L, 3L, 0L, 0L, 3L, 2L, 3L, 0L, 0L, 0L, 0L, 0L, 0L,
4L, 3L, 1L, 0L, 3L, 3L, 0L, 4L, 1L, 0L, 0L, 0L, 3L, 3L, 2L,
0L, 0L, 0L, 2L, 2L, 0L, 0L, 4L, 2L, 0L, 4L, 0L, 3L, 1L, 3L,
4L, 3L, 2L, 0L, 4L, 0L, 0L, 1L, 0L, 2L, 0L, 1L, 0L, 0L, 0L,
0L, 0L, 3L, 1L, 0L, 3L, 0L, 0L, 0L, 1L, 3L, 1L, 3L, 0L, 3L,
0L, 3L, 2L, 4L, 0L, 0L, 3L, 3L, 2L, 0L, 4L, 3L, 4L), Atr31 = c(4L,
4L, 4L, 0L, 0L, 2L, 4L, 3L, 1L, 4L, 0L, 3L, 0L, 4L, 3L, 1L,
4L, 2L, 0L, 3L, 1L, 0L, 0L, 0L, 4L, 4L, 0L, 3L, 4L, 4L, 1L,
0L, 2L, 3L, 4L, 1L, 0L, 4L, 3L, 4L, 2L, 2L, 2L, 0L, 4L, 2L,
4L, 4L, 0L, 0L, 4L, 4L, 2L, 4L, 2L, 0L, 0L, 1L, 3L, 4L, 4L,
0L, 2L, 0L, 4L, 4L, 0L, 0L, 4L, 0L, 0L, 4L, 0L, 1L, 1L, 3L,
4L, 4L, 4L, 0L, 4L, 0L, 1L, 0L, 2L, 1L, 0L, 1L, 0L, 2L, 1L,
0L, 0L, 4L, 1L, 1L, 4L, 2L, 0L, 1L, 0L, 4L, 1L, 4L, 0L, 4L,
2L, 4L, 4L, 4L, 0L, 4L, 3L, 4L, 4L, 0L, 3L, 3L, 4L), Atr32 = c(4L,
4L, 4L, 4L, 0L, 0L, 4L, 3L, 2L, 4L, 0L, 4L, 2L, 4L, 4L, 0L,
4L, 0L, 1L, 4L, 0L, 1L, 0L, 0L, 3L, 4L, 0L, 3L, 4L, 4L, 1L,
0L, 2L, 3L, 3L, 0L, 0L, 3L, 3L, 4L, 1L, 1L, 0L, 1L, 2L, 0L,
3L, 4L, 2L, 0L, 3L, 4L, 1L, 4L, 1L, 0L, 0L, 3L, 3L, 3L, 4L,
0L, 1L, 0L, 4L, 4L, 0L, 0L, 3L, 0L, 0L, 3L, 0L, 1L, 1L, 4L,
4L, 4L, 4L, 0L, 4L, 0L, 0L, 4L, 0L, 2L, 0L, 0L, 2L, 1L, 0L,
0L, 0L, 3L, 1L, 1L, 4L, 1L, 0L, 3L, 2L, 4L, 0L, 4L, 0L, 3L,
0L, 3L, 4L, 4L, 0L, 1L, 3L, 4L, 4L, 0L, 3L, 3L, 4L), Atr33 = c(4L,
4L, 4L, 4L, 0L, 0L, 4L, 3L, 1L, 4L, 0L, 3L, 0L, 4L, 3L, 0L,
4L, 0L, 1L, 3L, 0L, 0L, 0L, 0L, 4L, 4L, 0L, 3L, 3L, 4L, 1L,
0L, 0L, 3L, 4L, 0L, 1L, 4L, 3L, 4L, 2L, 1L, 0L, 0L, 0L, 0L,
4L, 4L, 0L, 0L, 4L, 4L, 0L, 4L, 0L, 0L, 1L, 0L, 3L, 4L, 4L,
0L, 1L, 0L, 4L, 4L, 0L, 0L, 4L, 0L, 0L, 4L, 0L, 1L, 2L, 3L,
4L, 4L, 4L, 0L, 4L, 0L, 0L, 2L, 0L, 2L, 0L, 0L, 0L, 1L, 0L,
0L, 0L, 4L, 1L, 0L, 4L, 0L, 0L, 0L, 0L, 4L, 0L, 4L, 0L, 4L,
0L, 4L, 4L, 4L, 0L, 1L, 3L, 4L, 4L, 0L, 3L, 3L, 4L), Atr34 = c(4L,
4L, 4L, 0L, 0L, 0L, 4L, 3L, 2L, 4L, 0L, 4L, 1L, 4L, 4L, 0L,
4L, 1L, 1L, 4L, 0L, 1L, 0L, 0L, 3L, 4L, 0L, 3L, 4L, 4L, 1L,
0L, 0L, 3L, 3L, 0L, 1L, 3L, 3L, 4L, 0L, 1L, 1L, 0L, 0L, 0L,
3L, 4L, 1L, 0L, 3L, 4L, 1L, 4L, 2L, 0L, 0L, 0L, 3L, 3L, 4L,
0L, 1L, 1L, 4L, 4L, 1L, 0L, 3L, 1L, 0L, 3L, 0L, 1L, 2L, 4L,

```

```

4L, 4L, 4L, 0L, 4L, 1L, 0L, 3L, 0L, 1L, 0L, 0L, 0L, 1L, 1L,
1L, 0L, 3L, 1L, 0L, 4L, 0L, 0L, 0L, 0L, 4L, 0L, 4L, 0L, 3L,
0L, 3L, 4L, 4L, 0L, 1L, 3L, 4L, 4L, 0L, 3L, 3L, 4L), Atr35 = c(4L,
4L, 4L, 1L, 0L, 0L, 4L, 3L, 0L, 4L, 0L, 3L, 0L, 4L, 3L, 0L,
4L, 0L, 0L, 3L, 0L, 0L, 0L, 0L, 4L, 4L, 0L, 3L, 0L, 4L, 0L,
0L, 0L, 3L, 4L, 0L, 0L, 4L, 3L, 4L, 0L, 0L, 0L, 0L, 0L, 0L,
4L, 4L, 0L, 0L, 4L, 4L, 0L, 4L, 0L, 0L, 1L, 0L, 3L, 4L, 4L,
0L, 0L, 0L, 4L, 4L, 0L, 0L, 4L, 0L, 0L, 4L, 0L, 1L, 0L, 3L,
4L, 4L, 4L, 0L, 4L, 0L, 0L, 0L, 0L, 1L, 0L, 0L, 0L, 0L, 0L,
0L, 0L, 4L, 0L, 0L, 4L, 0L, 0L, 0L, 0L, 4L, 0L, 4L, 0L, 4L,
0L, 4L, 4L, 4L, 0L, 1L, 3L, 4L, 4L, 0L, 3L, 3L, 4L), Atr36 = c(4L,
4L, 4L, 0L, 0L, 0L, 4L, 3L, 1L, 4L, 0L, 4L, 0L, 4L, 4L, 0L,
4L, 0L, 0L, 4L, 0L, 0L, 0L, 0L, 3L, 4L, 0L, 3L, 0L, 4L, 0L,
0L, 0L, 3L, 3L, 0L, 0L, 3L, 3L, 4L, 0L, 0L, 0L, 0L, 0L, 0L,
3L, 4L, 0L, 0L, 3L, 4L, 0L, 0L, 0L, 0L, 0L, 3L, 3L, 4L,
0L, 0L, 0L, 4L, 4L, 0L, 0L, 3L, 0L, 0L, 3L, 0L, 1L, 0L, 4L,
4L, 4L, 4L, 0L, 4L, 0L, 0L, 2L, 0L, 2L, 0L, 0L, 0L, 0L, 0L,
0L, 0L, 3L, 0L, 0L, 4L, 0L, 0L, 0L, 0L, 4L, 0L, 4L, 0L, 3L,
0L, 3L, 4L, 4L, 0L, 1L, 3L, 4L, 4L, 0L, 3L, 3L, 4L), Atr37 = c(4L,
4L, 4L, 2L, 0L, 0L, 4L, 3L, 2L, 4L, 0L, 3L, 0L, 4L, 3L, 0L,
4L, 1L, 0L, 3L, 0L, 1L, 0L, 0L, 4L, 4L, 0L, 4L, 4L, 4L, 1L,
1L, 0L, 3L, 4L, 0L, 0L, 4L, 3L, 4L, 2L, 1L, 1L, 0L, 3L, 2L,
4L, 4L, 0L, 0L, 4L, 4L, 1L, 4L, 1L, 0L, 1L, 2L, 4L, 4L, 4L,
0L, 2L, 0L, 4L, 4L, 0L, 0L, 4L, 1L, 0L, 4L, 0L, 1L, 0L, 3L,
4L, 4L, 4L, 0L, 4L, 1L, 0L, 3L, 4L, 3L, 0L, 0L, 0L, 0L, 0L,
1L, 0L, 4L, 1L, 0L, 4L, 1L, 1L, 0L, 0L, 4L, 0L, 4L, 2L, 4L,
0L, 4L, 4L, 4L, 0L, 1L, 3L, 4L, 4L, 0L, 3L, 3L, 4L), Atr38 = c(4L,
4L, 4L, 0L, 0L, 1L, 4L, 3L, 1L, 4L, 3L, 4L, 2L, 4L, 4L, 0L,
4L, 0L, 2L, 4L, 0L, 1L, 0L, 0L, 3L, 4L, 0L, 4L, 0L, 4L, 0L,
0L, 0L, 3L, 3L, 0L, 1L, 3L, 3L, 4L, 0L, 1L, 0L, 0L, 0L, 2L,
3L, 4L, 0L, 0L, 4L, 4L, 0L, 4L, 1L, 0L, 0L, 1L, 4L, 3L, 4L,
0L, 0L, 0L, 4L, 4L, 0L, 0L, 3L, 1L, 0L, 3L, 0L, 2L, 0L, 4L,
4L, 4L, 4L, 0L, 4L, 0L, 0L, 4L, 0L, 2L, 0L, 0L, 0L, 1L, 0L,
0L, 0L, 3L, 0L, 0L, 4L, 1L, 0L, 0L, 0L, 4L, 0L, 4L, 1L, 3L,
0L, 3L, 4L, 4L, 0L, 2L, 3L, 4L, 4L, 0L, 3L, 3L, 4L), Atr39 = c(4L,
4L, 4L, 4L, 0L, 0L, 4L, 3L, 3L, 4L, 0L, 3L, 0L, 4L, 3L, 0L,
4L, 2L, 1L, 3L, 0L, 1L, 0L, 1L, 4L, 4L, 0L, 4L, 1L, 4L, 1L,
0L, 0L, 4L, 4L, 0L, 1L, 4L, 1L, 4L, 1L, 2L, 0L, 0L, 4L, 1L,
4L, 4L, 0L, 0L, 3L, 4L, 0L, 4L, 1L, 0L, 1L, 0L, 4L, 4L, 4L,
0L, 1L, 0L, 4L, 4L, 0L, 0L, 4L, 2L, 1L, 4L, 0L, 2L, 0L, 3L,
4L, 4L, 4L, 0L, 4L, 1L, 0L, 2L, 0L, 2L, 0L, 0L, 0L, 1L, 0L,
4L, 0L, 4L, 0L, 0L, 4L, 2L, 0L, 0L, 0L, 4L, 0L, 4L, 1L, 4L,
0L, 4L, 4L, 4L, 0L, 0L, 3L, 4L, 4L, 0L, 3L, 3L, 4L), Atr40 = c(4L,
4L, 4L, 0L, 0L, 0L, 4L, 3L, 3L, 4L, 0L, 4L, 0L, 4L, 4L, 0L,
4L, 0L, 0L, 4L, 0L, 0L, 0L, 1L, 3L, 4L, 0L, 4L, 0L, 4L, 1L,

```

```

0L, 0L, 4L, 3L, 0L, 0L, 3L, 3L, 4L, 0L, 1L, 0L, 0L, 4L, 0L,
3L, 4L, 0L, 0L, 4L, 4L, 0L, 4L, 1L, 0L, 0L, 2L, 4L, 3L, 4L,
0L, 0L, 0L, 4L, 4L, 0L, 0L, 3L, 0L, 0L, 3L, 0L, 2L, 0L, 4L,
4L, 4L, 4L, 0L, 4L, 0L, 0L, 4L, 0L, 3L, 0L, 0L, 0L, 0L, 0L,
1L, 0L, 3L, 0L, 0L, 4L, 1L, 0L, 0L, 0L, 4L, 0L, 4L, 0L, 3L,
0L, 3L, 4L, 4L, 0L, 2L, 3L, 4L, 4L, 0L, 3L, 3L, 4L), Atr41 = c(4L,
4L, 4L, 2L, 0L, 0L, 4L, 3L, 2L, 4L, 1L, 3L, 2L, 4L, 3L, 0L,
4L, 1L, 0L, 3L, 0L, 0L, 0L, 0L, 4L, 4L, 0L, 3L, 1L, 4L, 1L,
0L, 0L, 4L, 4L, 0L, 0L, 4L, 2L, 4L, 0L, 1L, 1L, 0L, 4L, 0L,
4L, 4L, 0L, 1L, 4L, 4L, 1L, 4L, 0L, 0L, 0L, 0L, 4L, 4L, 4L,
0L, 1L, 0L, 4L, 4L, 0L, 1L, 4L, 1L, 0L, 4L, 1L, 2L, 1L, 3L,
4L, 4L, 4L, 0L, 4L, 0L, 3L, 2L, 0L, 3L, 0L, 0L, 0L, 0L, 0L,
2L, 0L, 4L, 0L, 0L, 4L, 0L, 0L, 0L, 0L, 4L, 2L, 4L, 1L, 4L,
0L, 4L, 4L, 4L, 0L, 2L, 3L, 4L, 4L, 0L, 3L, 3L, 4L), Atr42 = c(4L,
4L, 4L, 4L, 2L, 0L, 3L, 3L, 1L, 4L, 1L, 4L, 2L, 4L, 4L, 1L,
4L, 3L, 0L, 4L, 0L, 4L, 1L, 0L, 3L, 4L, 1L, 0L, 0L, 4L, 1L,
1L, 4L, 4L, 3L, 0L, 3L, 3L, 3L, 4L, 0L, 3L, 1L, 0L, 4L, 2L,
3L, 4L, 0L, 0L, 4L, 4L, 0L, 4L, 2L, 0L, 0L, 0L, 3L, 3L, 4L,
0L, 0L, 0L, 4L, 4L, 0L, 2L, 3L, 2L, 2L, 3L, 0L, 2L, 0L, 4L,
4L, 4L, 3L, 0L, 4L, 0L, 1L, 2L, 0L, 3L, 0L, 2L, 0L, 3L, 2L,
1L, 2L, 3L, 0L, 0L, 4L, 2L, 0L, 2L, 2L, 4L, 0L, 4L, 3L, 3L,
0L, 3L, 4L, 4L, 0L, 1L, 3L, 4L, 4L, 0L, 3L, 3L, 4L), Atr43 = c(4L,
4L, 4L, 4L, 4L, 2L, 4L, 3L, 1L, 4L, 1L, 3L, 3L, 4L, 3L, 3L,
4L, 2L, 2L, 3L, 0L, 2L, 1L, 3L, 4L, 4L, 2L, 0L, 0L, 4L, 1L,
2L, 4L, 3L, 4L, 2L, 2L, 4L, 3L, 4L, 2L, 2L, 2L, 4L, 4L, 2L,
4L, 4L, 3L, 1L, 4L, 4L, 2L, 4L, 2L, 1L, 1L, 0L, 0L, 4L, 4L,
0L, 0L, 3L, 4L, 4L, 0L, 4L, 4L, 3L, 2L, 4L, 2L, 2L, 1L, 3L,
4L, 4L, 4L, 3L, 4L, 2L, 1L, 3L, 0L, 3L, 1L, 4L, 0L, 2L, 2L,
1L, 3L, 4L, 0L, 2L, 4L, 2L, 2L, 2L, 2L, 4L, 3L, 4L, 4L, 4L,
2L, 4L, 4L, 4L, 3L, 2L, 3L, 4L, 4L, 4L, 3L, 3L, 4L), Atr44 = c(4L,
4L, 4L, 1L, 4L, 0L, 3L, 3L, 2L, 4L, 0L, 4L, 0L, 4L, 4L, 2L,
4L, 0L, 1L, 4L, 2L, 0L, 1L, 2L, 3L, 4L, 0L, 3L, 0L, 4L, 1L,
0L, 0L, 4L, 3L, 1L, 1L, 3L, 4L, 4L, 0L, 0L, 0L, 0L, 4L, 1L,
3L, 4L, 0L, 0L, 3L, 4L, 0L, 4L, 0L, 0L, 0L, 0L, 0L, 3L, 4L,
0L, 0L, 0L, 4L, 4L, 0L, 2L, 3L, 0L, 0L, 3L, 0L, 2L, 2L, 4L,
3L, 4L, 4L, 0L, 4L, 2L, 0L, 4L, 0L, 4L, 1L, 1L, 0L, 1L, 0L,
0L, 1L, 3L, 0L, 0L, 4L, 2L, 2L, 0L, 2L, 4L, 0L, 4L, 0L, 3L,
0L, 3L, 4L, 4L, 1L, 3L, 3L, 4L, 4L, 4L, 3L, 3L, 4L), Atr45 = c(4L,
4L, 3L, 4L, 4L, 2L, 3L, 3L, 3L, 4L, 3L, 3L, 3L, 4L, 3L, 1L,
4L, 1L, 1L, 3L, 1L, 1L, 2L, 4L, 4L, 3L, 2L, 0L, 0L, 4L, 1L,
1L, 4L, 2L, 4L, 0L, 3L, 3L, 4L, 4L, 1L, 2L, 1L, 4L, 4L, 2L,
3L, 4L, 0L, 0L, 4L, 4L, 0L, 4L, 0L, 3L, 0L, 0L, 0L, 4L, 4L,
0L, 0L, 0L, 4L, 4L, 1L, 3L, 4L, 1L, 1L, 4L, 1L, 2L, 0L, 3L,
4L, 3L, 4L, 3L, 4L, 2L, 1L, 2L, 0L, 3L, 0L, 0L, 0L, 2L, 3L,
4L, 3L, 4L, 0L, 0L, 3L, 1L, 4L, 1L, 2L, 4L, 3L, 4L, 4L, 4L,

```



```

2L, 4L, 3L, 4L, 0L, 0L, 3L, 4L, 4L, 4L, 3L, 3L, 4L), Atr46 = c(4L,
4L, 4L, 0L, 4L, 0L, 3L, 3L, 2L, 4L, 3L, 4L, 2L, 4L, 4L, 3L,
4L, 3L, 1L, 4L, 4L, 2L, 3L, 4L, 3L, 4L, 1L, 0L, 1L, 4L, 0L,
4L, 1L, 2L, 3L, 3L, 3L, 2L, 4L, 2L, 1L, 2L, 3L, 2L, 3L,
3L, 4L, 2L, 0L, 4L, 4L, 2L, 4L, 2L, 2L, 3L, 0L, 0L, 3L, 4L,
0L, 0L, 0L, 4L, 4L, 1L, 1L, 3L, 1L, 0L, 3L, 1L, 1L, 1L, 4L,
4L, 4L, 4L, 4L, 4L, 0L, 3L, 2L, 3L, 3L, 4L, 2L, 2L, 2L, 3L,
3L, 3L, 3L, 0L, 2L, 4L, 0L, 4L, 3L, 2L, 4L, 3L, 4L, 3L, 3L,
2L, 3L, 4L, 4L, 3L, 2L, 3L, 4L, 4L, 4L, 3L, 3L, 4L), Atr47 = c(4L,
4L, 4L, 4L, 4L, 2L, 4L, 3L, 1L, 4L, 2L, 3L, 0L, 4L, 3L, 0L,
4L, 3L, 1L, 3L, 2L, 1L, 1L, 0L, 4L, 4L, 0L, 0L, 0L, 4L, 0L,
4L, 4L, 3L, 4L, 0L, 0L, 4L, 3L, 4L, 1L, 2L, 2L, 1L, 4L, 0L,
4L, 4L, 2L, 1L, 3L, 4L, 1L, 4L, 1L, 0L, 0L, 0L, 0L, 4L, 4L,
0L, 0L, 0L, 4L, 4L, 1L, 1L, 4L, 2L, 1L, 4L, 1L, 1L, 0L, 3L,
4L, 4L, 4L, 3L, 4L, 0L, 3L, 2L, 0L, 2L, 1L, 1L, 2L, 1L, 3L,
3L, 3L, 4L, 0L, 0L, 4L, 1L, 4L, 4L, 2L, 4L, 1L, 4L, 4L, 4L,
1L, 4L, 4L, 4L, 0L, 2L, 3L, 4L, 4L, 4L, 3L, 3L, 4L), Atr48 = c(4L,
4L, 4L, 4L, 4L, 3L, 4L, 3L, 3L, 4L, 2L, 4L, 2L, 4L, 4L, 2L,
4L, 2L, 2L, 4L, 3L, 3L, 0L, 1L, 3L, 4L, 0L, 3L, 4L, 4L, 2L,
3L, 2L, 3L, 3L, 2L, 2L, 3L, 2L, 4L, 3L, 3L, 2L, 3L, 4L, 2L,
3L, 4L, 0L, 0L, 4L, 4L, 2L, 4L, 2L, 2L, 2L, 3L, 3L, 3L, 4L,
4L, 2L, 2L, 4L, 4L, 0L, 2L, 3L, 2L, 2L, 3L, 1L, 1L, 3L, 4L,
4L, 4L, 4L, 0L, 4L, 2L, 2L, 3L, 2L, 3L, 1L, 2L, 3L, 2L, 3L,
2L, 2L, 3L, 2L, 2L, 4L, 3L, 2L, 4L, 2L, 4L, 2L, 4L, 3L, 3L,
1L, 3L, 4L, 4L, 1L, 1L, 3L, 4L, 4L, 0L, 3L, 3L, 4L), Atr49 = c(4L,
4L, 4L, 4L, 4L, 0L, 4L, 3L, 3L, 4L, 0L, 3L, 4L, 4L, 3L, 3L,
4L, 2L, 1L, 3L, 0L, 1L, 0L, 0L, 4L, 4L, 1L, 3L, 1L, 4L, 1L,
3L, 3L, 2L, 4L, 3L, 3L, 4L, 2L, 4L, 3L, 2L, 2L, 1L, 3L, 0L,
4L, 4L, 0L, 0L, 4L, 4L, 2L, 4L, 1L, 2L, 2L, 1L, 3L, 4L, 4L,
4L, 2L, 0L, 4L, 4L, 1L, 0L, 4L, 1L, 0L, 4L, 0L, 1L, 1L, 3L,
4L, 4L, 4L, 2L, 4L, 2L, 0L, 4L, 2L, 2L, 4L, 1L, 3L, 3L, 0L,
2L, 1L, 4L, 1L, 3L, 4L, 2L, 2L, 0L, 0L, 4L, 1L, 4L, 1L, 4L,
1L, 4L, 4L, 4L, 0L, 2L, 3L, 4L, 4L, 4L, 3L, 3L, 4L), Atr50 = c(4L,
4L, 4L, 3L, 2L, 3L, 4L, 3L, 3L, 4L, 2L, 3L, 0L, 4L, 3L, 3L,
4L, 1L, 1L, 3L, 0L, 1L, 1L, 0L, 4L, 4L, 1L, 3L, 1L, 4L, 2L,
1L, 2L, 1L, 4L, 4L, 2L, 4L, 3L, 4L, 2L, 2L, 2L, 3L, 4L, 0L,
4L, 4L, 1L, 0L, 3L, 4L, 2L, 4L, 2L, 2L, 2L, 1L, 3L, 4L, 4L,
4L, 2L, 0L, 4L, 4L, 2L, 0L, 4L, 0L, 0L, 4L, 0L, 1L, 1L, 3L,
4L, 4L, 4L, 2L, 4L, 2L, 0L, 4L, 2L, 3L, 4L, 2L, 3L, 2L, 1L,
3L, 1L, 4L, 2L, 2L, 4L, 2L, 2L, 1L, 2L, 4L, 1L, 4L, 3L, 4L,
1L, 4L, 4L, 4L, 0L, 1L, 3L, 4L, 4L, 0L, 3L, 3L, 4L), Atr51 = c(4L,
4L, 4L, 4L, 2L, 3L, 3L, 3L, 2L, 4L, 2L, 4L, 0L, 4L, 4L, 2L,
4L, 1L, 1L, 4L, 0L, 1L, 1L, 1L, 3L, 4L, 1L, 3L, 4L, 4L, 2L,
2L, 2L, 1L, 3L, 3L, 2L, 3L, 3L, 4L, 2L, 2L, 2L, 3L, 4L, 2L,
3L, 4L, 1L, 1L, 3L, 4L, 2L, 4L, 2L, 2L, 2L, 1L, 3L, 3L, 4L,

```

```

4L, 2L, 1L, 4L, 4L, 2L, 1L, 3L, 1L, 0L, 3L, 0L, 1L, 3L, 4L,
4L, 4L, 4L, 2L, 4L, 2L, 0L, 4L, 2L, 3L, 2L, 2L, 3L, 2L, 3L,
2L, 1L, 3L, 2L, 2L, 4L, 2L, 2L, 3L, 1L, 4L, 1L, 4L, 3L, 3L,
1L, 3L, 4L, 4L, 0L, 1L, 3L, 4L, 4L, 0L, 3L, 3L, 4L), Atr52 = c(4L,
4L, 4L, 4L, 0L, 3L, 3L, 3L, 3L, 4L, 0L, 3L, 1L, 4L, 3L, 1L,
4L, 2L, 0L, 3L, 2L, 1L, 0L, 4L, 4L, 4L, 1L, 3L, 2L, 4L, 2L,
3L, 0L, 2L, 4L, 3L, 4L, 4L, 4L, 4L, 2L, 1L, 2L, 3L, 1L, 1L,
4L, 4L, 3L, 0L, 4L, 4L, 2L, 4L, 1L, 2L, 2L, 0L, 3L, 4L, 4L,
0L, 1L, 2L, 4L, 4L, 0L, 4L, 4L, 3L, 3L, 4L, 1L, 1L, 1L, 3L,
4L, 4L, 4L, 0L, 4L, 1L, 4L, 4L, 4L, 2L, 2L, 4L, 1L, 3L, 3L,
4L, 1L, 4L, 3L, 1L, 4L, 1L, 4L, 3L, 1L, 4L, 0L, 4L, 0L, 4L,
1L, 4L, 4L, 4L, 2L, 1L, 3L, 4L, 4L, 0L, 3L, 3L, 4L), Atr53 = c(4L,
4L, 4L, 4L, 0L, 2L, 4L, 3L, 2L, 4L, 0L, 4L, 0L, 4L, 4L, 0L,
4L, 0L, 0L, 4L, 4L, 1L, 0L, 1L, 3L, 4L, 1L, 3L, 2L, 4L, 2L,
1L, 0L, 2L, 3L, 0L, 1L, 3L, 2L, 4L, 2L, 1L, 2L, 1L, 1L, 0L,
3L, 4L, 4L, 0L, 4L, 4L, 1L, 4L, 0L, 2L, 2L, 3L, 3L, 3L, 4L,
0L, 0L, 2L, 4L, 4L, 1L, 1L, 3L, 2L, 2L, 3L, 1L, 1L, 1L, 4L,
4L, 4L, 4L, 0L, 4L, 1L, 4L, 2L, 4L, 2L, 0L, 2L, 1L, 1L, 3L,
3L, 1L, 3L, 3L, 1L, 4L, 0L, 2L, 3L, 1L, 4L, 1L, 4L, 1L, 3L,
1L, 3L, 4L, 4L, 0L, 2L, 3L, 4L, 4L, 0L, 3L, 3L, 4L), Atr54 = c(4L,
4L, 4L, 4L, 0L, 2L, 4L, 3L, 1L, 4L, 4L, 3L, 0L, 4L, 3L, 0L,
4L, 0L, 0L, 3L, 1L, 0L, 0L, 0L, 4L, 4L, 1L, 3L, 2L, 4L, 3L,
1L, 0L, 2L, 4L, 2L, 1L, 4L, 2L, 4L, 0L, 0L, 2L, 0L, 0L, 0L,
4L, 4L, 0L, 1L, 3L, 4L, 0L, 4L, 0L, 2L, 1L, 0L, 3L, 4L, 4L,
0L, 0L, 1L, 4L, 4L, 1L, 2L, 4L, 2L, 1L, 4L, 1L, 1L, 1L, 3L,
4L, 4L, 4L, 0L, 4L, 0L, 2L, 2L, 0L, 2L, 1L, 0L, 0L, 1L, 1L,
1L, 1L, 4L, 2L, 0L, 4L, 0L, 2L, 1L, 0L, 4L, 0L, 4L, 0L, 4L,
1L, 4L, 4L, 4L, 0L, 0L, 3L, 4L, 4L, 0L, 3L, 3L, 4L), Class = c(1L,
1L, 1L, 1L, 0L, 0L, 1L, 1L, 1L, 1L, 0L, 1L, 0L, 1L, 1L, 0L,
1L, 0L, 0L, 1L, 0L, 0L, 0L, 0L, 1L, 1L, 0L, 1L, 0L, 1L, 0L,
0L, 0L, 1L, 1L, 0L, 0L, 1L, 1L, 1L, 0L, 0L, 0L, 0L, 1L, 0L,
1L, 1L, 0L, 0L, 1L, 1L, 0L, 1L, 0L, 0L, 0L, 0L, 1L, 1L, 1L,
0L, 0L, 0L, 1L, 1L, 0L, 0L, 1L, 0L, 0L, 1L, 0L, 1L, 0L, 1L,
1L, 1L, 1L, 0L, 1L, 0L, 0L, 1L, 0L, 1L, 0L, 0L, 0L, 0L,
0L, 0L, 1L, 0L, 0L, 1L, 0L, 0L, 0L, 0L, 1L, 0L, 1L, 0L, 1L,
0L, 1L, 1L, 1L, 0L, 1L, 1L, 1L, 1L, 0L, 1L, 1L, 1L)), n.trees = 1000L,
interaction.depth = 1L, n.minobsinnode = 10L, shrinkage = 0.3,
bag.fraction = 0.65, train.fraction = 0.6)

```

A gradient boosted model with bernoulli loss function.

1000 iterations were performed.

The best test-set iteration was 291.

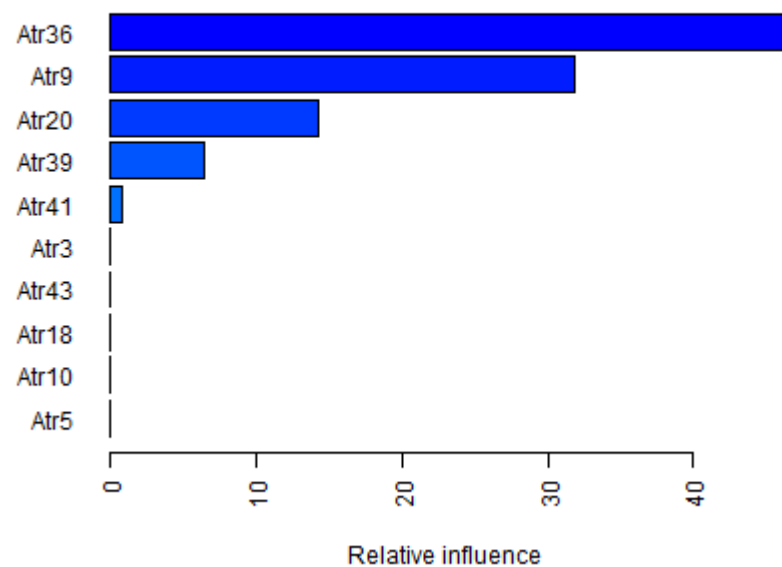
There were 54 predictors of which 42 had non-zero influence.

In [124...

```

## variable importance
summary_gbm(gbmfitfinal, cBars=10, method=relative_influence, las=2)

```



```

Out[124...] RObject{VecSxp}
              var      rel.inf
Atr36 Atr36 4.656368e+01
Atr9  Atr9 3.178883e+01
Atr20 Atr20 1.426894e+01
Atr39 Atr39 6.469082e+00
Atr41 Atr41 7.751315e-01
Atr3  Atr3 4.969773e-02
Atr43 Atr43 4.513574e-02
Atr18 Atr18 1.132426e-02
Atr10 Atr10 8.659461e-03
Atr5  Atr5 8.158494e-03
Atr19 Atr19 6.677647e-03
Atr40 Atr40 2.939721e-03
Atr24 Atr24 1.611443e-03
Atr47 Atr47 8.306560e-05
Atr46 Atr46 1.641042e-05
Atr16 Atr16 1.228826e-05
Atr42 Atr42 1.052843e-05
Atr31 Atr31 8.029381e-06
Atr38 Atr38 5.428242e-06
Atr37 Atr37 2.894578e-07

```

```
Atr45 Atr45 2.130625e-07
Atr53 Atr53 1.219763e-07
Atr49 Atr49 4.979980e-08
Atr4   Atr4 1.355115e-08
Atr29 Atr29 2.513691e-09
Atr12 Atr12 3.599763e-10
Atr7   Atr7 1.564867e-10
Atr6   Atr6 7.295320e-11
Atr25 Atr25 1.396280e-11
Atr13 Atr13 5.409304e-12
Atr34 Atr34 4.720206e-16
Atr54 Atr54 3.737765e-16
Atr32 Atr32 1.402412e-16
Atr17 Atr17 1.179978e-18
Atr52 Atr52 3.792935e-21
Atr27 Atr27 3.563406e-21
Atr11 Atr11 1.220153e-21
Atr30 Atr30 3.888702e-22
Atr44 Atr44 1.413635e-24
Atr8   Atr8 2.245950e-25
Atr50 Atr50 3.020823e-26
Atr22 Atr22 8.758835e-31
Atr2   Atr2 6.828315e-31
Atr28 Atr28 6.490013e-31
Atr1   Atr1 4.046873e-31
Atr26 Atr26 1.815913e-31
Atr51 Atr51 1.464523e-31
Atr14 Atr14 6.004580e-32
Atr21 Atr21 3.594215e-32
Atr15 Atr15 0.000000e+00
Atr23 Atr23 0.000000e+00
Atr33 Atr33 0.000000e+00
Atr35 Atr35 0.000000e+00
Atr48 Atr48 0.000000e+00
```

Significant Variables: Atr5, Atr10, Atr18, Atr43, Atr3, Atr41, Atr39, Atr20, Atr9, Atr36

In [125...

```
Base.names(gbmfitfinal)
```

Out[125...

```
27-element Vector{Symbol}:
 :initF
 :fit
```

```

Symbol("train.error")
Symbol("valid.error")
Symbol("oobag.improve")
:trees
Symbol("c.splits")
Symbol("bag.fraction")
:distribution
Symbol("interaction.depth")
Symbol("n.minobsinnode")
Symbol("num.classes")
Symbol("n.trees")
:
Symbol("response.name")
:shrinkage
Symbol("var.levels")
Symbol("var.monotone")
Symbol("var.names")
Symbol("var.type")
:verbose
:data
:Terms
Symbol("cv.folds")
:call
:m

```

In [126... `pretty_gbm_tree(gbmfitfinal,var"i.tree"=4)`

Out[126... RObject{VecSxp}

	SplitVar	SplitCodePred	LeftNode	RightNode	MissingNode	ErrorReduction	Weight
0	8	1.50000000	1	2	3	1.799597	46
1	-1	-0.37324026	-1	-1	-1	0.000000	26
2	-1	0.39139673	-1	-1	-1	0.000000	20
3	-1	-0.04078939	-1	-1	-1	0.000000	46

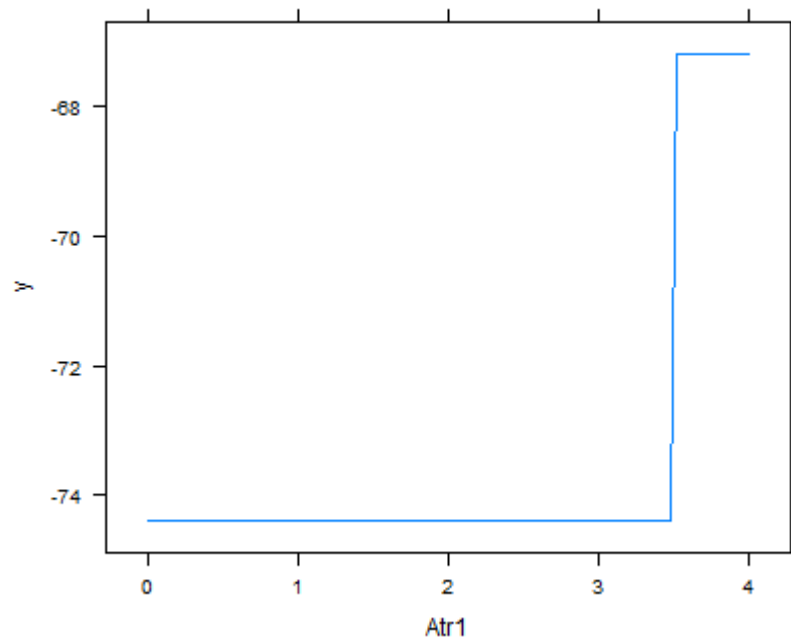
Prediction

```

0 -0.04078939
1 -0.37324026
2 0.39139673
3 -0.04078939

```

In [127... `plot_gbm(gbmfitfinal)`



Out[127... RObject{VecSxp}

```
In [128... pred=zeros(rbase.dim(traindf)[1])
@. pred=Base.exp(gbmfitfinal[:fit]) / (1 + Base.exp(gbmfitfinal[:fit]))
```

Out[128... 119-element Vector{Float64}:

```
1.0
1.0
1.0
1.0
4.788251549851476e-81
4.8958255980253826e-89
1.0
1.0
0.9999999999999999
1.0
3.8129785633342425e-82
1.0
6.822416596671893e-81
⋮
1.0
```

```

1.0
1.0
6.976792381878964e-99
1.9997196066040062e-26
1.0
1.0
1.0
2.7017353594745792e-81
1.0
1.0
1.0

```

```
In [129... testpred = predict_gbm(gbmfitfinal, testdf, type="response")
```

```

Warning: RCall.jl: Using 291 trees...
@ RCall C:\Users\ASUS\.julia\packages\RCall\6kphM\src\io.jl:172
RObject{RealSxp}
Out[129... [1] 1.000000e+00 1.000000e+00 1.469274e-20 1.584378e-20 6.850998e-20
[6] 1.000000e+00 1.000000e+00 3.533149e-18 4.085305e-18 8.890953e-11
[11] 1.015183e-17 1.000000e+00 1.000000e+00 1.000000e+00 1.000000e+00
[16] 6.440071e-11 1.000000e+00 1.328913e-16 9.080681e-19 1.000000e+00
[21] 1.000000e+00 1.000000e+00 1.000000e+00 1.728944e-06 1.000000e+00
[26] 2.994805e-11 1.906533e-18 1.000000e+00 1.451178e-07 1.826004e-14
[31] 1.000000e+00 6.760925e-18 1.058344e-20 1.000000e+00 1.000000e+00
[36] 4.085092e-14 5.029092e-17 1.000000e+00 1.000000e+00 1.000000e+00
[41] 7.900524e-12 1.503554e-18 1.000000e+00 1.000000e+00 1.044455e-15
[46] 1.000000e+00 2.017992e-08 1.799879e-17 4.529867e-18 1.906533e-18
[51] 3.734590e-18

```

(d) ROC Curves

```

In [130... @rput testdf treeModel rfmodel gbmfitfinal

R"treetestpred=predict(treeModel,testdf)"

R"rfetestpred=predict(rfmodel,testdf)"

R"bsttestpred=predict(gbmfitfinal,testdf)"

@rget treetestpred
@rget rfetestpred

```

```
@rget bsttestpred

tree_predbin=ifelse.(treetestpred .>= 0.5 , 1 , 0);
rf_predbin=ifelse.(rftestpred .>= 0.5 , 1 , 0);
bst_predbin=ifelse.(bsttestpred .>= 0.5 , 1 , 0);
```

```
└ Warning: RCall.jl: Using 291 trees...
└ @ RCall C:\Users\ASUS\.julia\packages\RCall\6kphM\src\io.jl:172
```

In [131...

```
R"library(ROCR)"

target=testdf.Class

@rput target
@rput tree_predbin rf_predbin bst_predbin

R"pred_tree=prediction(treetestpred,target)"
R"pred_rf=prediction(rf_predbin,target)"
R"pred_bst=prediction(bst_predbin,target)"
```

Out[131...

```
RObject{S4Sxp}
A prediction instance
with 51 data points
```

In [132...

```
R""perf_tree=performance(pred_tree,"tpr","fpr")""
```

Out[132...

```
RObject{S4Sxp}
A performance instance
'False positive rate' vs. 'True positive rate' (alpha: 'Cutoff')
with 4 data points
```

In [133...

```
R""perf_tree=performance(pred_tree,"tpr","fpr")""
R""perf_rf=performance(pred_rf,"tpr","fpr")""
R""perf_bst=performance(pred_bst,"tpr","fpr")""
```

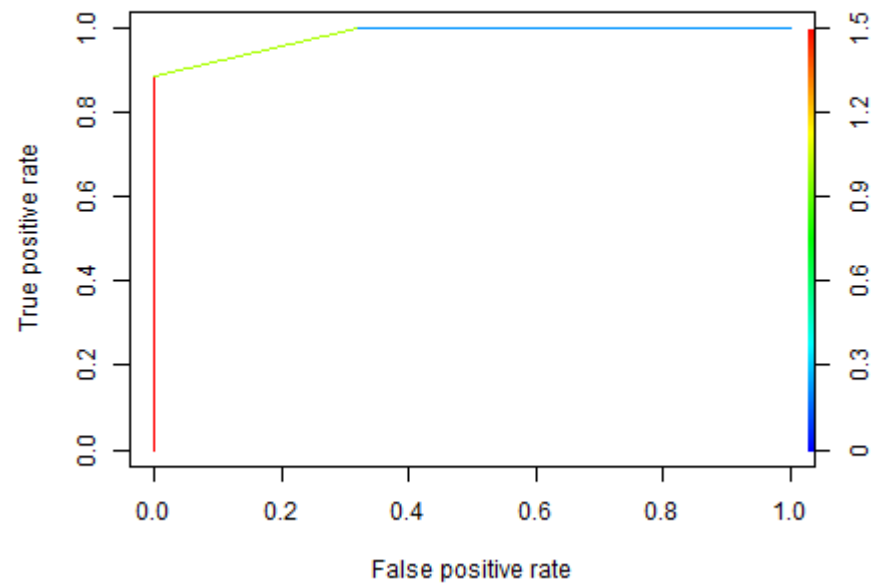
Out[133...

```
RObject{S4Sxp}
A performance instance
'False positive rate' vs. 'True positive rate' (alpha: 'Cutoff')
with 3 data points
```

Tree ROC

In [134...

```
R"plot(perf_tree,colorize=TRUE)"
```

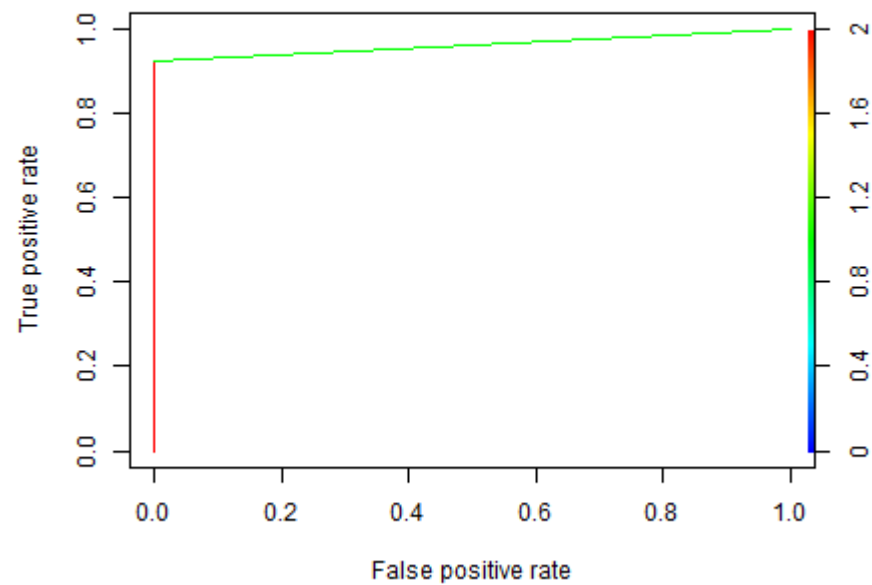


Out[134... RObject{NilSxp}
NULL

Random Forest ROC

In [135...

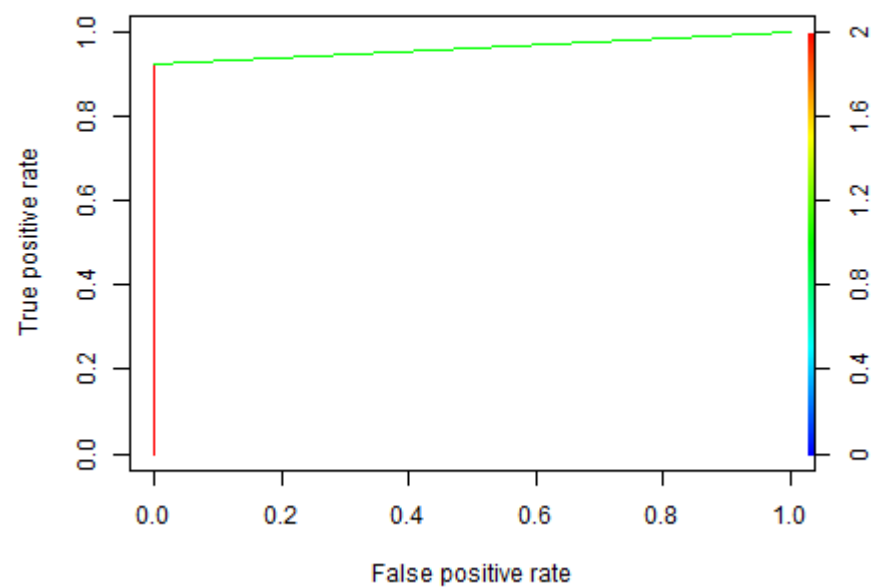
```
R"plot(perf_rf, colorize=TRUE)"
```



Out[135... RObject{NilSxp}
NULL

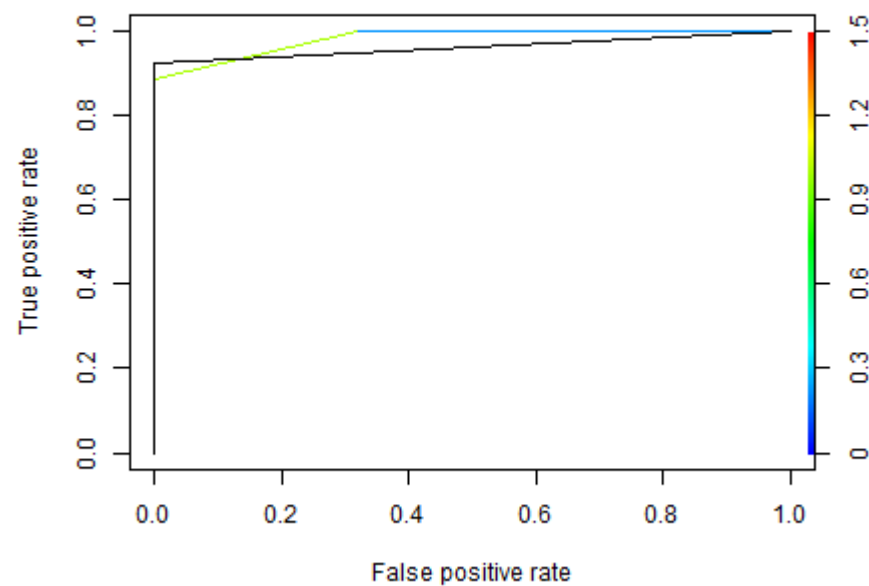
Boosting Model ROC

In [136... `R"plot(perf_bst,colorize=TRUE)"`



Out[136... RObject{NilSxp}
NULL

```
In [137... R"plot(perf_tree,colorize=TRUE)"  
R"plot(perf_rf, add=TRUE, colorize=FALSE)"  
R"plot(perf_bst, add=TRUE, colorize=FALSE)"
```



Out[137... RObject{NilSxp}
NULL

```
In [138... R""" aucout=performance(pred_tree, measure = "auc")"""
R"auc = aucout@y.values[[1]]"
```

Out[138... RObject{RealSxp}
[1] 0.9815385

```
In [139... R""" aucout=performance(pred_rf, measure = "auc")"""
R"auc = aucout@y.values[[1]]"
```

Out[139... RObject{RealSxp}
[1] 0.9615385

```
In [140... R""" aucout=performance(pred_bst, measure = "auc")"""
R"auc = aucout@y.values[[1]]"
```

Out[140... RObject{RealSxp}
[1] 0.9615385

(e) Evaluation Metrics

```
In [142... @rget treetestpred  
@rget rftestpred  
@rget bsttestpred
```

```
Out[142... 51-element Vector{Float64}:  
 52.079341680640916  
 33.15128067536675  
-45.66693368630261  
-45.591509960503686  
-44.12730745800129  
 55.05023342732032  
 53.73054485084614  
-40.18434204587633  
-40.03913521691153  
-23.14340178738558  
-39.12887810365402  
 51.611612676002196  
 53.73054485084614  
  ⋮  
 41.4303289148677  
-25.56409198385277  
-41.03870013714935  
 54.85696124989908  
 54.19827385548486  
-34.49528156271359  
 50.97567434171983  
-17.718577777658997  
-38.556227152668136  
-39.93583913331837  
-40.80124504686618  
-40.128893612293325
```

```
In [143... function Eval(thres, target, pred)  
    bin_pred =ifelse.(pred .>= thres , 1 , 0);  
    acc = EvalMetrics.accuracy(target, bin_pred)  
    confusion_matrix1 = MLBase.roc(target, bin_pred)  
    TP = confusion_matrix1.tp  
    TN = confusion_matrix1.tn  
    FP = confusion_matrix1.fp  
    FN = confusion_matrix1.fn;
```

```

specificity = (TN) / (FP + TN)
sensitivity = (TP) / (TP + FN)
return acc, specificity, sensitivity
end

```

Out[143...] Eval (generic function with 1 method)

For Tree Model

```

In [71]: Threshold = 0.5
acc_tree, spec_tree, sens_tree = Eval(Threshold, target, treetestpred)

```

Out[71]: (0.9607843137254902, 1.0, 0.9259259259259259)

```

In [144...] println("Tree Model Accuracy::", acc_tree)
println("Tree Model Specificity::", spec_tree)
println("Tree Model Sensivity::", sens_tree)

```

Tree Model Accuracy::0.9803921568627451
Tree Model Specificity::1.0
Tree Model Sensivity::0.9629629629629629

For Random Forest

Threshold = 0.5 acc_rf, spec_rf, sens_rf = Eval(Threshold, target, rftestpred)

```

In [145...] println("Random Forest Model Accuracy::", acc_rf)
println("Random Forest Model Specificity::", spec_rf)
println("Random Forest Model Sensivity::", sens_rf)

```

Random Forest Model Accuracy::0.9803921568627451
Random Forest Model Specificity::1.0
Random Forest Model Sensivity::0.9629629629629629

For Boosting Model

```

In [146...] Threshold = 0.5
acc_bst, spec_bst, sens_bst = Eval(Threshold, target, bsttestpred)

```

Out[146... (0.9607843137254902, 1.0, 0.9230769230769231)

```
In [147...  
println("Boosting Model Accuracy::", acc_bst)  
println("Boosting Model Specificity::", spec_bst)  
println("Boosting Model Sensivity::", sens_bst)
```

```
Boosting Model Accuracy::0.9607843137254902  
Boosting Model Specificity::1.0  
Boosting Model Sensivity::0.9230769230769231
```

Overall Performance of Tree Different Models are pretty good according to the different Evaluation Metrics

In []: