

```

#### Begin ####
# Draw care points from a normal distribution with mean = 2 and SD=
  1
spr.set <- data.frame("ego" = NA, "mom" = NA, "age" = NA, "mom_giv"
  = 0, "mom_get" = 0, "sib_giv" = 0, "sib_get" = 0, "mgn_giv" =
  0, "mgn_get" = 0, "ant_giv" = 0, "ant_get" = 0, "cuz_giv" = 0, "
  cuz_get" = 0)
age.efs <- data.frame("age"=0:99, "need"=c((5:1)/5, rep(0, 95)), "
  care"=c(rep(0, 5), (1:20)*0.05, rep(1, 75)))
baby.drw <- c(rep(1, 25), rep(0, 75)) # gives the odds of birth
  each year

for(r in 1:100){
fam <- data.frame(matrix(nrow=3, ncol=6)) # build the base
  dataset with mom and her three older daughters then columns to
  be filled
colnames(fam) <- c("ego", "mom", "mgn", "age", "need", "care")
fam[1,] <- c(1, 98, 99, 39, 0, 0)
fam[2,] <- c(2, 1, 98, 19, 0, 0)
fam[3,] <- c(3, 1, 98, 19, 0, 0)
#fam[4,] <- c(4, 1, 98, 19, 0, 0)

for(k in 1:21){ # this loop is responsible for generating each year
  fam$age <- fam$age + 1 # captures the aging across time
  for(i in 1:nrow(fam)){ # Here is where we add new kids
    which are born to women btw 20 n 40 every three years
    if(fam$ego[i] == 2 | fam$ego[i] == 3 & sample(baby.
      drw, 1) == 1){ # Need to adjust
      IDs if +/- sisters.
      fam <- rbind(fam, c(nrow(fam) + 1, fam$ego[
        i], fam$mom[i], 0, 0, 0))
    } else {
    }
  }
  # set the distribution for how much care the women have to
  offer and draw a sample for this time set
  for(i in 1:nrow(fam)){
    fam$care[i] <- age.efs$care[age.efs$age == fam$age[
      i]]
  }
  for(i in 1:nrow(fam)){
    fam$need[i] <- age.efs$need[age.efs$age == fam$age[
      i]]
  }
  rcrds <- data.frame("ego" = fam$ego, "mom" = fam$mom, "age"
    = fam$age, "mom_giv" = 0, "mom_get" = 0, "sib_giv" =
    0, "sib_get" = 0, "mgn_giv" = 0, "mgn_get" = 0, "ant_
    giv" = 0, "ant_get" = 0, "cuz_giv" = 0, "cuz_get" = 0)
  if(sum(fam$need) > 0){
    cg <- subset(fam, care > 0) # Isolate care
    givers
    cr <- subset(fam, need > 0) # Isolate care
    recievers
    cmat <- data.frame(matrix(nrow = 0, ncol = 5)) #
    Build the ranking matrix
    colnames(cmat) <- c("cg", "cr", "care", "need", "

```

```

    rel")
  for(j in 1:nrow(cg)){
    hld <- data.frame(cg$ego[j], cr$ego, cg$
      care[j], cr$need, NA)
    colnames(hld) <- c("cg", "cr", "care", "
      need", "rel")
    for(h in 1:nrow(hld)){
      if(cg$ego[j] == cr$mom[h] | cg$mom[
        j] == cr$mom[h]){
        hld$rel[h] <- 0.5
      } else if((cg$mom[j] == cr$mgm[h] |
        cg$ego[j] == cr$mgm[h]) & cg$
        ego[j] != cr$mom[h]){
        hld$rel[h] <- 0.25
      } else{
        hld$rel[h] <- 0.125
      }
    }
    cmat <- rbind(cmat, hld)
  }
  while(sum(cmat$need) > 0 & sum(cmat$care) > 0){
    cmat <- subset(cmat, need > 0 & care > 0)
    rand <- sample(nrow(cmat))
    cmat <- cmat[rand,]
    cmat <- cmat[order(cmat$care, decreasing=T)
      ,]
    cmat <- cmat[order(cmat$need, decreasing=T)
      ,]
    cmat <- cmat[order(cmat$rel, decreasing=T)
      ,]
    cr.gv <- ifelse(cmat$need[1] > cmat$care
      [1], cmat$care[1], cmat$need[1]) # how
      much more is given?
    cmat$need[cmat$cr == cmat$cr[1]] <- cmat$
      need[cmat$cr == cmat$cr[1]] - cr.gv
    cmat$care[cmat$cg == cmat$cg[1]] <- cmat$
      care[cmat$cg == cmat$cg[1]] - cr.gv
    if(cmat$cg[1] == cr$mom[cr$ego == cmat$cr
      [1]]){
      # Is mom and kid?
      rcrds$mom.giv[rcrds$ego == cmat$cg
        [1]] <- rcrds$mom.giv[rcrds$ego
          == cmat$cg[1]] + cr.gv
      rcrds$mom.get[rcrds$ego == cmat$cr
        [1]] <- rcrds$mom.get[rcrds$ego
          == cmat$cr[1]] + cr.gv
    } else if(cg$mom[cg$ego == cmat$cg[1]] == cr
      $mom[cr$ego == cmat$cr[1]]){
      # is sibs?
      rcrds$sib.giv[rcrds$ego == cmat$cg
        [1]] <- rcrds$sib.giv[rcrds$ego
          == cmat$cg[1]] + cr.gv
      rcrds$sib.get[rcrds$ego == cmat$cr
        [1]] <- rcrds$sib.get[rcrds$ego
          == cmat$cr[1]] + cr.gv
    } else if(cmat$cg[1] == cr$mgm[cr$ego ==
      cmat$cr[1]]){

```

```

# is granny
and gkids?
  rcrds$mgm_get[rcrds$ego == cmat$cr
    [1]] <- rcrds$mgm_get[rcrds$ego
    == cmat$cr[1]] + cr.gv
  rcrds$mgm_giv[rcrds$ego == cmat$cg
    [1]] <- rcrds$mgm_giv[rcrds$ego
    == cmat$cg[1]] + cr.gv
} else if (cg$mom[cg$ego == cmat$cg[1]] == cr
  $mgm[cr$ego == cmat$cr[1]] & cmat$cg[1]
  != cr$mom[cr$ego == cmat$cr[1]]) { #
  aunty and nenes?
  rcrds$ant_get[rcrds$ego == cmat$cr
    [1]] <- rcrds$ant_get[rcrds$ego
    == cmat$cr[1]] + cr.gv
  rcrds$ant_giv[rcrds$ego == cmat$cg
    [1]] <- rcrds$ant_giv[rcrds$ego
    == cmat$cg[1]] + cr.gv
} else {

  # is cuzs?
  rcrds$cuz_giv[rcrds$ego == cmat$cg
    [1]] <- rcrds$cuz_giv[rcrds$ego
    == cmat$cg[1]] + cr.gv
  rcrds$cuz_get[rcrds$ego == cmat$cr
    [1]] <- rcrds$cuz_get[rcrds$ego
    == cmat$cr[1]] + cr.gv
}
}
for (g in 4:ncol(rcrds)){
  rcrds[2,g] <- rcrds[2,g] + sum(rcrds[,g][rcrds$mom
    == rcrds$ego[2]])
  # rcrds[3,g] <- rcrds[3,g] + sum(rcrds[,g][rcrds$mom
  # == rcrds$ego[3]])
  # rcrds[4,g] <- rcrds[4,g] + sum(rcrds[,g][rcrds$mom
  # == rcrds$ego[4]])
}
for (g in 2:3){
  if (length(rcrds$ego[rcrds$ego[g] == rcrds$
    mom]) > 0){
    spr.set <- rbind(spr.set, rcrds[g
      ,])
  }
}
} else {
}
}
spr.set <- spr.set[-1,]

#### END #####

```