

Extended `dfuncEstim` examples

Trent McDonald

2023-06-12

Following is a series of `dfuncEstim` calls that show the calling parameters for popular distance sampling options.

```
library(Rdistance)
```

```
## Rdistance (version 3.0.0)
```

```
data("sparrowDetectionData")
```

```
data("sparrowSiteData")
```

Half-normal function with right truncation

```
dfunc <- dfuncEstim(formula = dist ~ 1  
                    , detectionData = sparrowDetectionData  
                    , w.hi = units::set_units(100, "m"))
```

```
dfunc
```

```
## Call: dfuncEstim(formula = dist ~ 1, detectionData =  
##   sparrowDetectionData, w.hi = units::set_units(100, "m"))
```

```
## Coefficients:
```

```
##      Estimate SE      z      p(>|z|)  
## Sigma 46.3587 2.549913 18.1805 7.365789e-74
```

```
##
```

```
## Convergence: Success
```

```
## Function: HALFNORM
```

```
## Strip: 0 [m] to 100 [m]
```

```
## Effective strip width (ESW): 56.30084 [m]
```

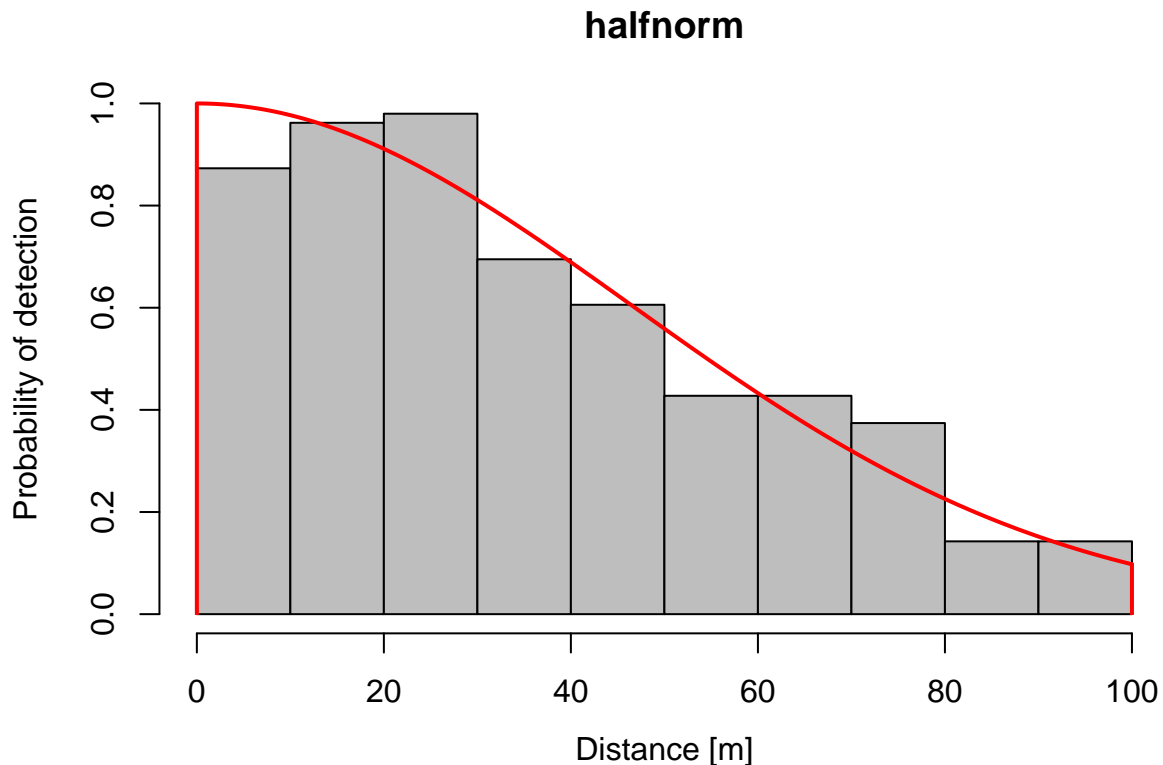
```
## Probability of detection: 0.5630084
```

```
## Scaling: g(0 [m]) = 1
```

```
## Negative log likelihood: 1484.297
```

```
## AICc: 2970.605
```

```
plot(dfunc, col="grey")
```



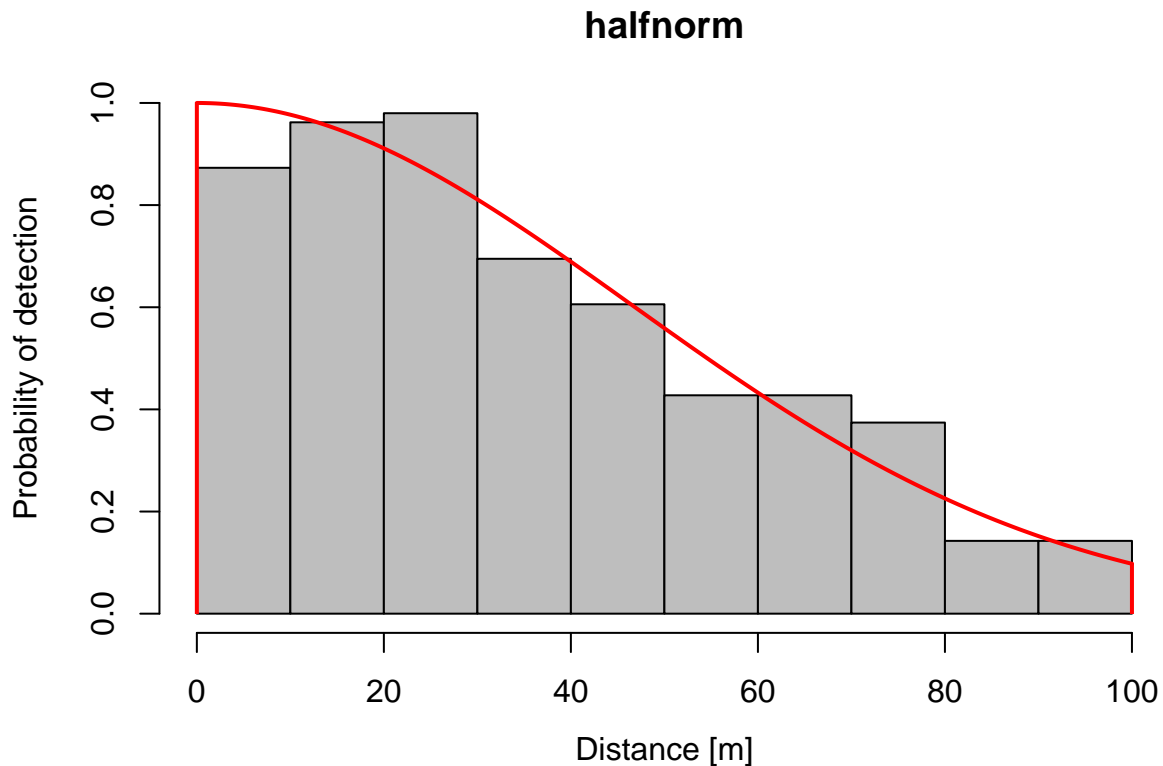
Half-normal function, truncation, group sizes

```
dfunc <- dfuncEstim(formula = dist ~ groupsize(groupsize)
  , detectionData = sparrowDetectionData
  , w.hi = units::set_units(100, "m"))
```

```
dfunc
```

```
## Call: dfuncEstim(formula = dist ~ groupsize(groupsize), detectionData =
##   sparrowDetectionData, w.hi = units::set_units(100, "m"))
## Coefficients:
##      Estimate SE          z      p(>|z|)
## Sigma 46.3587 2.549913 18.1805 7.365789e-74
##
## Convergence: Success
## Function: HALF NORM
## Strip: 0 [m] to 100 [m]
## Effective strip width (ESW): 56.30084 [m]
## Probability of detection: 0.5630084
## Scaling: g(0 [m]) = 1
## Negative log likelihood: 1484.297
## AICc: 2970.605
```

```
plot(dfunc, col="grey")
```



Half-normal function with factor covariate

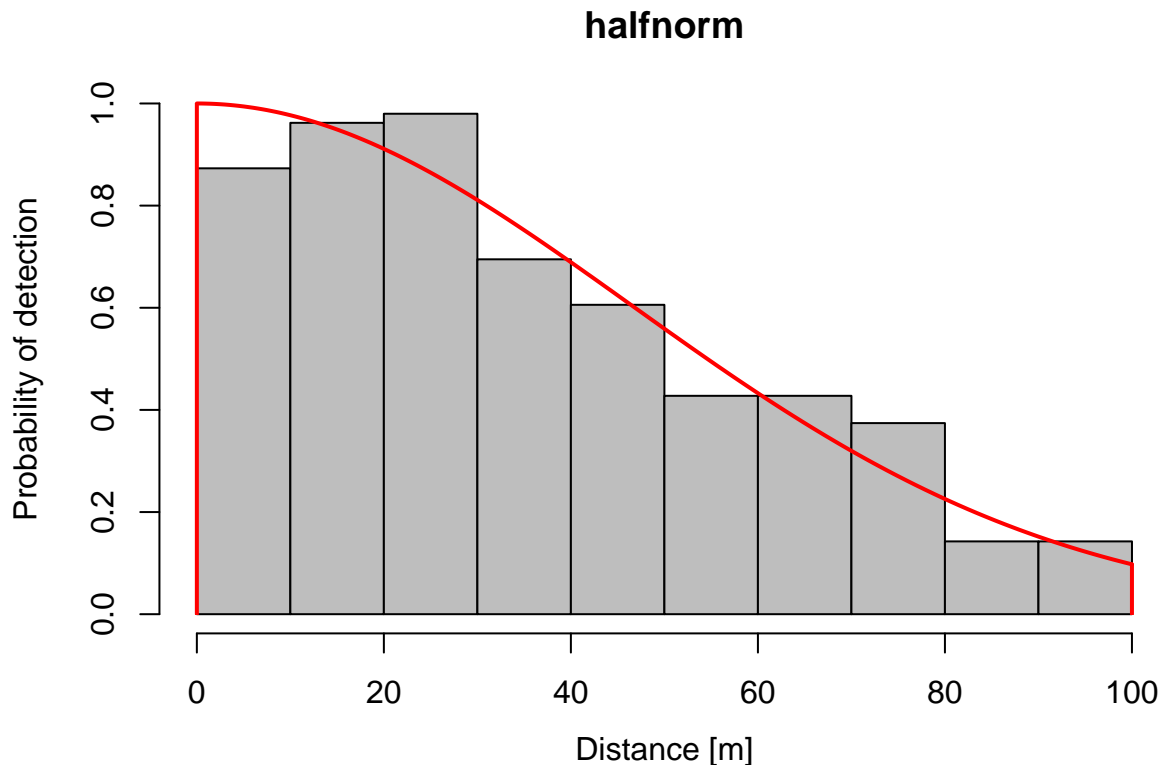
Increase the maximum number of iterations if distance function convergence is an issue. The `observer` covariate is constant within transects and appears in the `site` data frame (`sparrowSiteData`), so the site data frame must be included in the call to `dfuncEstim`. Otherwise, the site data frame is not needed until abundance is estimated (in `abundEstim`).

```
dfuncObs <- dfuncEstim(formula = dist ~ observer
                        , detectionData = sparrowDetectionData
                        , siteData = sparrowSiteData
                        , w.hi = units::set_units(100, "m")
                        , control=RdistanceControls(maxIter=1000))
dfunc
```

```
## Call: dfuncEstim(formula = dist ~ groupsize(groupsize), detectionData =
##   sparrowDetectionData, w.hi = units::set_units(100, "m"))
## Coefficients:
##      Estimate SE      z      p(>|z|)
## Sigma  46.3587  2.549913  18.1805  7.365789e-74
##
## Convergence: Success
## Function: HALFNORM
## Strip: 0 [m] to 100 [m]
## Effective strip width (ESW): 56.30084 [m]
```

```
## Probability of detection: 0.5630084
## Scaling: g(0 [m]) = 1
## Negative log likelihood: 1484.297
## AICc: 2970.605
```

```
plot(dfunc, col="grey")
```



Hazard-rate function with covariate, truncation, and variable group sizes

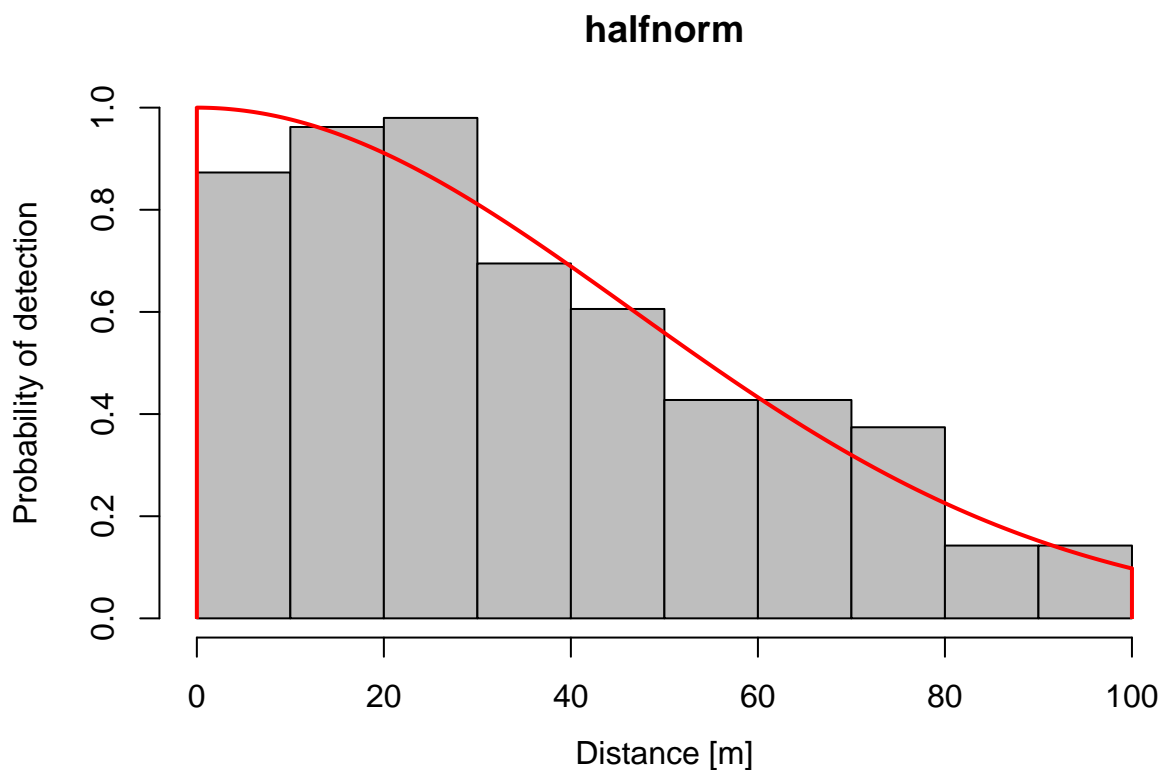
Group sizes do not influence the estimated distance function. Only distance to the group is used. But, group sizes are associated with individual detections and are used to estimate abundance in function `abundEstim`. If abundance will be estimate and group sizes vary, `Rdistance` requires specification of a group size variable in the call to `dfuncEstim`. Here, `groupsize` is a column in the *detection* data frame and group sizes are specified using `groupsize()` in the formula.

```
dfuncObs <- dfuncEstim(formula = dist ~ observer + groupsize(groupsize)
  , likelihood = "hazrate"
  , detectionData = sparrowDetectionData
  , siteData = sparrowSiteData
  , w.hi = units::set_units(100, "m"))
dfunc
```

```
## Call: dfuncEstim(formula = dist ~ groupsize(groupsize), detectionData =
##   sparrowDetectionData, w.hi = units::set_units(100, "m"))
## Coefficients:
```

```
##           Estimate SE           z      p(>|z|)
## Sigma  46.3587   2.549913  18.1805  7.365789e-74
##
## Convergence: Success
## Function: HALFNORM
## Strip: 0 [m] to 100 [m]
## Effective strip width (ESW): 56.30084 [m]
## Probability of detection: 0.5630084
## Scaling: g(0 [m]) = 1
## Negative log likelihood: 1484.297
## AICc: 2970.605
```

```
plot(dfunc, col="grey")
```



Hazard rate, covariate, group sizes, left and right truncation

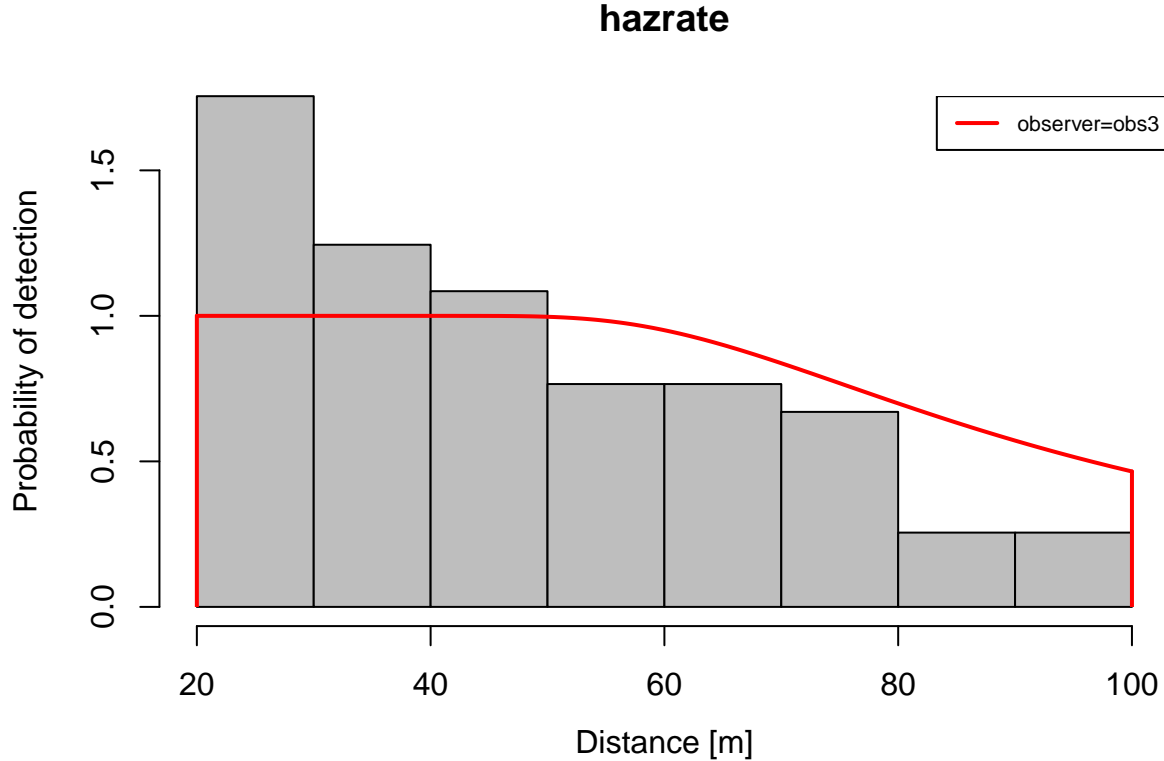
Right truncation at 100 meters, left truncation at 20 meters. If `x.scl` is not specified as greater than `w.lo`, a warning is issued.

```
dfunc <- dfuncEstim(formula = dist ~ observer + groupsize(groupsize)
  , likelihood = "hazrate"
  , detectionData = sparrowDetectionData
  , siteData = sparrowSiteData
  , w.lo = units::set_units(20, "m")
  , x.scl = units::set_units(20, "m")
  , w.hi = units::set_units(100, "m"))
```

```
dfunc
```

```
## Call: dfuncEstim(formula = dist ~ observer + groupsize(groupsize),
##   detectionData = sparrowDetectionData, siteData = sparrowSiteData,
##   likelihood = "hazrate", w.lo = units::set_units(20, "m"), w.hi =
##   units::set_units(100, "m"), x.scl = units::set_units(20, "m"))
## Coefficients:
##           Estimate      SE          z          p(>|z|)
## (Intercept)  3.8931772  0.2295966 16.95659731 1.720222e-64
## observerobs2  0.4785896  0.3884430  1.23207161 2.179223e-01
## observerobs3  0.2821896  0.2923237  0.96533254 3.343783e-01
## observerobs4 -3.1636957 47.3789797 -0.06677425 9.467614e-01
## observerobs5 -0.3483488  0.3284466 -1.06059501 2.888740e-01
## k            2.2649670  0.3463938  6.53870471 6.205391e-11
##
## Convergence: Success
## Function: HAZRATE
## Strip: 20 [m] to 100 [m]
## Average effective strip width (ESW): 50.49281 [m]
## Average probability of detection: 0.6311601
## Scaling: g(20 [m]) = 1
## Negative log likelihood: 901.4591
## AICc: 1815.322
```

```
plot(dfunc, col="grey")
```



Hazard rate, covariate, group sizes, specify $g(0)$

Specify $g(0)$ at the intercept by setting `g.x.scl`. This scales the entire distance function. Here, probability of detection on the transect is known to be 0.8.

```
dfunc <- dfuncEstim(formula = dist ~ observer + groupsize(groupsize)
                    , likelihood = "hazrate"
                    , detectionData = sparrowDetectionData
                    , siteData = sparrowSiteData
                    , g.x.scl = 0.8)
dfunc

## Call: dfuncEstim(formula = dist ~ observer + groupsize(groupsize),
##   detectionData = sparrowDetectionData, siteData = sparrowSiteData,
##   likelihood = "hazrate", g.x.scl = 0.8)
## Coefficients:
##           Estimate      SE          z          p(>|z|)
## (Intercept)  3.999741296  0.1316417  30.38354648  9.061662e-203
## observerobs2  0.140084708  0.1737911   0.80605228  4.202127e-01
## observerobs3  0.004448875  0.1452423   0.03063071  9.755640e-01
## observerobs4 -0.423433553     0.1612352  -2.62618559  8.634772e-03
## observerobs5 -0.151863036     0.1508740  -1.00655543  3.141485e-01
## k            3.117839108  0.3313693   9.40895702  5.010843e-21
##
## Convergence: Success
## Function: HAZRATE
## Strip: 0 [m] to 207 [m]
## Average effective strip width (ESW): 53.55757 [m]
## Average probability of detection: 0.2587322
## Scaling:  $g(0 \text{ [m]}) = 0.8$ 
## Negative log likelihood: 1659.513
## AICc: 3331.266
plot(dfunc, col="grey")
```

hazrate

