`nupoint` (Version 1.0.45) package example: environmental preference

Martin J. Cox, David L. Borchers and Natalie Kelly

March 21, 2013

1 Introduction

The purpose of this vignette is to provide an example to enable users to estimate an environmental preference function from point transect survey data. To do this, we estimate the distribution of beaked whales cues as a function of depth, while simultaneously accounting for decreasing detectability of whale cues with range. We fit a variety of functional forms for cue distribution as a function of depth, using a half-normal form for for the detection function, and select the best model on the basis of AIC. Model adequacy is tested using a χ^2 goodness-of-fit test. We illustrate interval and variance estimation using a non-parametric bootstrap.

We start by loading the `nupoint` package:

```
> library(nupoint)
```

2 Example dataset description

The survey region is specified via a grid of Cartesian coordinate pairs (x, y) spanning the region in which we want to draw inference. The coordinates are oriented so that y runs perpendicularly from the observer offshore and x is perpendicular to y. Beaked whale cue prevalence is parameterised as a function of depth, $z(x, y)$, at (x, y) using a probability density function (pdf) $\pi_z(z)$. In order to convert a pdf specified in terms of z into one specified in terms of (x, y), we need the derivative $\frac{\partial z(x,y)}{\partial y}$ throughout the survey region. This is therefore one of the required inputs to the estimation function.

The estimation function `nupoint.env.fit( )` needs two kinds of data:

- Sightings data This is a data frame with a row for each sighting and columns `r`, `z` and `dzdy`, being the radial distance of the sighting from the observer, the depth at the location of the sightings, and $\frac{\partial z(x,y)}{\partial y}$, the derviative of depth with respect to offshore distance y, at the location of the sighting.

- Survey region data The data required here are matrices specifying the depth at every grid point in the survey region (z), the derivative $\frac{\partial z(x,y)}{\partial y}$ at every grid point, and the radial distance from the observer (r) at every grid point. In addition, the bounds of the survey region and the range of depths in it are required. These are specified via the maximum absolute values of x and y and the lower and upper limits of depth, z.

The data named `sightings` included in the `nupoint` package is a `list` object containing all of the above information. (see Table 1 for `sightings` structure). In addition to example whale cue observations ($n = 191$), the data set also describes the survey region. See Arranz et al. [submitted] for details of data acquisition.

	Variable name	description
1	sighting.mat	Sightings data frame with columns x, y, r, z, dxdy and obs.period (see below)
2	x.mat	Matrix containing x-coordinates of every grid point in the survey region
3	y.mat	Matrix containing y-coordinates of every grid point in the survey region
4	rd.mat	Matrix with radial distance to every grid point in the survey region
5	z.mat	Matrix containing depths of every grid point in the survey region
6	zGradmat	Matrix containing dzdy of every grid point in the survey region
7	x	Vector of unique x-values in x.mat
8	y	Vector of unique y-values in y.mat
9	obsx	Observer x-coordinate
10	obsy	Observer y-coordinate
11	wx	Absolute value of maximum x
12	wy	Maximum y (distance offshore)
13	wz	Maximum depth
14	minz	Minimum depth

Table 1: Description of the sightings data set.

2.1 Beaked whale observations

Beaked whale observations (Table 2) were collected by a shore based observer (at coordinates `(obsx,obsy)` equal to (4850, 0). Sighting position data (`x, y, r`), stored in the `sightings$sight.mat` were all observed from this location.

2.2 Survey region description

The survey region is defined using matrices (Table 1). `z.mat` is a matrix of seabed depths, with `NA` values defining regions which either cannot be surveyed (e.g., in this example land), or areas that exceed the truncation distances (Figure 1). When defining a survey region, all matrices (`x.mat, y.mat, rd.mat, z.mat, zGradmat`; see Table 1) must have the same dimensions, and each element in a given matrix must share common spatial locations with other matrices

	x	y	r	z	dzdy	obs.period
1	Min. : 450	Min. : 950	Min. :1408	Min. : 103.2	Min. :0.02079	Length:191
2	1st Qu.: 4350	1st Qu.:2100	1st Qu.:2742	1st Qu.: 664.0	1st Qu.:0.15042	Class :character
3	Median : 5950	Median :2950	Median :3855	Median : 916.9	Median :0.23940	Mode :character
4	Mean : 5914	Mean :3131	Mean :3928	Mean : 896.0	Mean :0.29743	
5	3rd Qu.: 7450	3rd Qu.:4050	3rd Qu.:5073	3rd Qu.:1132.9	3rd Qu.:0.42892	
6	Max. :11150	Max. :7750	Max. :8280	Max. :1903.3	Max. :1.26226	

Table 2: Beaked whale sightings summary statistics.

on a regular grid. The matrix x-axis must be aligned along the coastline with the y-axis positioned perpendicularly to the x-axis.

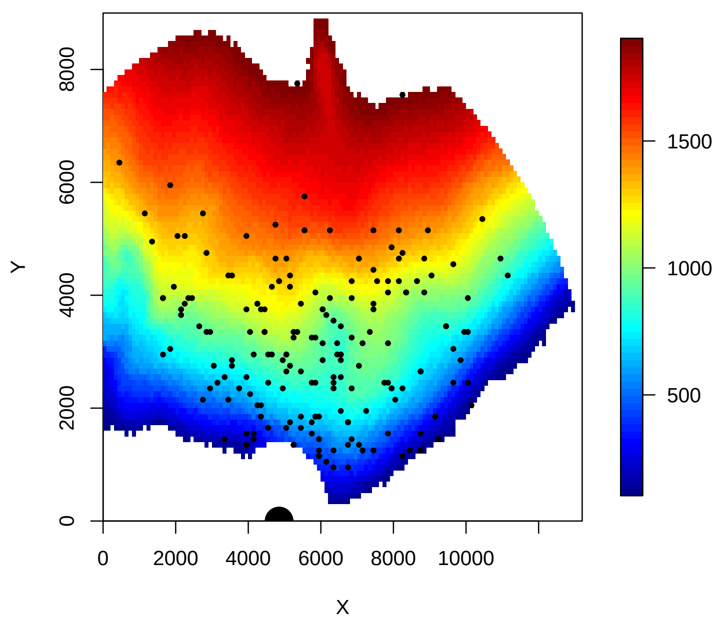

Figure 1: Plan view of seabed depth in survey region (units: m). Sightings of whale cues are shown as black points, and the shore-based observer as a solid black semicircle on the x-axis. Data were taken from a subset of those used by Arranz et al. [submitted].

3 Maximum likelihood estimation

We now use maximum likelihood to simultaneously estimate parameters, $\boldsymbol{\phi}$, for the depth preference function, $\pi_z(z;\boldsymbol{\phi})$ and parameters, $\boldsymbol{\Theta}$, for the detection function, $g(r;\Theta)$. By way of example, we will use a half-normal form for the detection function (specified via `nupoint.env.fit` argument `det.form`='HNORM'), and select a depth preference from five candidate forms (specified via `nupoint.env.fit` argument `grad.form`): 'NORM' for normal; 'LOGNORM' for log-normal; 'BETA' for beta; 'UNIFORM' for uniform, 'MNORM' (with additional `nupoint.env.fit` argument `n`= 2) for a mixture of two normal distributions.

The following code snippet is used to fit the normal depth preference form. Before we fit, we need to set the depth preference and detection functions (`norm.grad.type` and `norm.det.type`), and starting values for the parameters of both these functions (`norm.pars`), together with lower and upper bounds for the parameters (`norm.LB`, `norm.UB`):

```
> norm.grad.type <- "NORM" #seabed depth to cue relationship distribution shape.
> norm.det.type <- "HNORM"
> norm.pars <- c(1000,200,3000) #starting values for parameter vector to be estimated
>   #parameter vector to be estmated
>   #[1] = normal depth preference function mean;
>   #[2] = normal depth preference function standard error;
>   #[3] = half-normal detection function standard error.
> norm.LB <- c(-2000,1,1) #lower parameter space bound.
> norm.UB <- c(10000,10000,10000) #upper parameter space bound.
```

We then attach the example sighting data (see section 2.1):

```
> attach(sightings)
```

and finally call the `nupoint.env.fit` function and simultaneously estimate the depth preference and detection function parameters.

```
> norm.fit<-nupoint.env.fit(pars=norm.pars,
+                           z=sighting.mat$z,
+                           rd=sighting.mat$r,
+                           dzdy=sighting.mat$dzdy,
+                           z.mat=z.mat,
+                           dzdy.mat=zGradmat,
+                           rd.mat=rd.mat,
+                           minz=minz,
+                           wx=wx,
+                           wy=wy,
+                           wz=wz,
+                           grad.type=norm.grad.type,
+                           det.type=norm.det.type,
+                           n=nDist,lower.b=norm.LB,upper.b=norm.UB)
```

```
-----------------------------------------------------
Environmental gradient likelihood settings
-----------------------------------------------------
Environment preference parametric form: NORM
range detection function, g(r), parametric form: HNORM
Parameter starting values = 1000 200 3000
Truncation distances
x= 6450  ; y= 8850  ; z= 1903.951
------------------------ ----------------------------
Estimating parameters
-----------------------------------------------------
Maximum likelihood results
-----------------------------------------------------
parameter point estimates = 1171.479 503.87 3020.442
AIC = 3893.23
-----------------------------------------------------
```

3.1 AIC model selection

We use AIC to chose a model from a candidate set of models. In the case of beaked whale cues, our candidate set of models for the prevalence of whale cues with depth, $\pi_z(z)$, was: normal, log-normal, beta, uniform and a two mixture distribution comprising of two normals. We estimate using each of the candidate models and then compare AICs.

3.1.1 Beta

We follow the same procedure of specifying the depth preference and detection functions, as well as specifying the starting parameters and upper and lower parameter estimate bounds.

```
> beta.grad.type<-"BETA" #seabed depth to cue relationship distribution shape.
> beta.det.type<-"HNORM"
> beta.pars<-c(2,2,2200) #starting values for
>   #parameter vector to be estmated
>   #[1] = beta depth preference function log(mean);
>   #[2] = beta depth preference function log(standard error);
>   #[3] = half-normal detection function standard error.
> beta.LB<-c(0.1,0.1,1) #lower parameter space bound.
> beta.UB<-c(20,20,10000) #upper parameter space bound.
> beta.fit<-nupoint.env.fit(pars=beta.pars,
+                        z=sighting.mat$z,
+                        rd=sighting.mat$r,
+                        dzdy=sighting.mat$dzdy,
+                        z.mat=z.mat,
+                        dzdy.mat=zGradmat,
+                        rd.mat=rd.mat,
```

```
+                         minz=minz,
+                         wx=wx,
+                         wy=wy,
+                         wz=wz,
+                         grad.type=beta.grad.type,
+                         det.type=beta.det.type,
+                         n=nDist,lower.b=beta.LB,upper.b=beta.UB)

------------------------------------------------------
Environmental gradient likelihood settings
------------------------------------------------------
Environment preference parametric form: BETA
range detection function, g(r), parametric form: HNORM
Parameter starting values = 2 2 2200
Truncation distances
x= 6450  ; y= 8850  ; z= 1903.951
------------------------ ------------------------------
Estimating parameters
------------------------------------------------------
Maximum likelihood results
------------------------------------------------------
parameter point estimates = 1.961 1.06 2836.474
AIC = 3903.61
------------------------------------------------------
```

3.1.2 Uniform

```
> unif.grad.type <- "UNIFORM" #seabed depth to cue relationship distribution shape.
> unif.det.type <- "HNORM"
> unif.pars <- 2200 # starting values for parameter vector to be estmated
>   # *** I don't follow what this is - is the uniform constrained to 0 and and this paramet
> unif.LB <- 100 #lower parameter space bound.
> unif.UB <- 6000 #upper parameter space bound.
> unif.fit <- nupoint.env.fit(pars=unif.pars,
+                         z=sighting.mat$z,
+                         rd=sighting.mat$r,
+                         dzdy=sighting.mat$dzdy,
+                         z.mat=z.mat,
+                         dzdy.mat=zGradmat,
+                         rd.mat=rd.mat,
+                         minz=minz,
+                         wx=wx,
+                         wy=wy,
+                         wz=wz,
+                         grad.type=unif.grad.type,
+                         det.type=unif.det.type,
```

```
+                         n=nDist,lower.b=unif.LB,upper.b=unif.UB)
```

```
-----------------------------------------------------
Environmental gradient likelihood settings
-----------------------------------------------------
Environment preference parametric form: UNIFORM
range detection function, g(r), parametric form: HNORM
Parameter starting values = 2200
Truncation distances
x= 6450  ; y= 8850  ; z= 1903.951
------------------------ -----------------------------
Estimating parameters
-----------------------------------------------------
Maximum likelihood results
-----------------------------------------------------
parameter point estimates = 3387.024
AIC = 3937.5
-----------------------------------------------------
```

3.1.3 Log-normal

```
> lnorm.grad.type <- "LOGNORM" #seabed depth to cue relationship distribution shape.
> lnorm.det.type <- "HNORM"
> lnorm.pars=c(7,0.5,2500) #starting values for
>   #parameter vector to be estmated
>   #[1] = log-normal depth preference function log(mean);
>   #[2] = log-normal depth preference function log(standard deviation);
>   #[3] half-normal detection function standard deviation.
> lnorm.LB <- c(1,0.1,10) #lower parameter space bound.
> lnorm.UB <- c(20,10,10000) #upper parameter space bound.
> lnorm.fit <- nupoint.env.fit(pars=lnorm.pars,
+                         z=sighting.mat$z,
+                         rd=sighting.mat$r,
+                         dzdy=sighting.mat$dzdy,
+                         z.mat=z.mat,
+                         dzdy.mat=zGradmat,
+                         rd.mat=rd.mat,
+                         minz=minz,
+                         wx=wx,
+                         wy=wy,
+                         wz=wz,
+                         grad.type=lnorm.grad.type,
+                         det.type=lnorm.det.type,
+                         n=nDist,lower.b=lnorm.LB,upper.b=lnorm.UB)
```

```
-----------------------------------------------------
Environmental gradient likelihood settings
```

```
--------------------------------------------------------
Environment preference parametric form: LOGNORM
range detection function, g(r), parametric form: HNORM
Parameter starting values = 7 0.5 2500
Truncation distances
x= 6450  ; y= 8850  ; z= 1903.951
------------------------ ------------------------------
Estimating parameters
--------------------------------------------------------
Maximum likelihood results
--------------------------------------------------------
parameter point estimates = 20 2.668 2842.204
AIC = 3903.84
--------------------------------------------------------
```

For the log-normal parametric form, `nupoint.env.fit` returns a warning message:

```
Warning message:
In nupoint.env.fit(pars = lnorm.pars, z = sighting.mat$z, rd = sighting.mat$r,  :
  nupoint.env.fit: one or more parameter estimates at upper bound.  Check convergence
```

Examining the parameter estimates, $\hat{\boldsymbol{\phi}}$ (`parameter point estimates` in the above output), shows that $\log \mu$ is estimated at the upper bound of the parameter space, making the fit suspect, so the log-normal parametric form is excluded from further analysis.

3.1.4 Mixture of two normal distributions

Specifying arguments for the `nupoint.env.fit` is a little more complex when using the normal mixture `grad.type='MNORM'` environmental preference function, but there is only one more argument, `n`, which specifies the number of mixture distributions. We start by specifying the depth preference and detection functions:

```
> det.type <- "HNORM"
> sigma.r <- 3000 #half-normal standard deviation (sigma) detection function parameter
> grad.type <- "MNORM" #seabed depth cue distribution shape.
> nDist <- 2 #number of normal distributions in the normal mixture distribution.
```

We then assign a starting weight to each distribution in the mixture, $\boldsymbol{\alpha}$:

```
> wt <- rep(1/nDist,nDist) #distribution weights (final element removed later)
```

We distribute mixture distribution means, $\boldsymbol{\mu}$, evenly along the environmental gradient:

```
> mu <- seq(minz,wz,length=nDist)
```

and assign an arbitrary $\boldsymbol{\sigma}$= 300 m to each distribution.

```
> sigma <- rep(300,nDist)
> pars <- as.vector(matrix(c(mu,sigma,wt),ncol=nDist,byrow=TRUE))
```

The final element of the vector, $\boldsymbol{\alpha}$, the weight of the nth distribution in the mixture, is removed since $\alpha_n = 1 - \sum_{i=1}^{(n-1)} \alpha_i$.

```
> pars <- c(pars[-length(pars)],sigma.r)
```

Finally, we assign lower and upper parameter space bounds. We start by specifying minima:

```
> mumin <- rep(-2000,nDist) #minimum mean
> sigmamin <- rep(1,nDist) #minimum sigma
> alphamin <- rep(-15,nDist) #minimum alpha
> sigma.rmin <- 1 #minimum detection function parameter
```

then combine the parameter space minima in a single vector:

```
> lower.b <- as.vector(matrix(c(mumin,sigmamin,alphamin),ncol=nDist,byrow=TRUE))
> lower.b <- c(lower.b[-length(lower.b)],sigma.rmin) #remove nth weight
```

Next we specify maxima:

```
> mumax <- rep(5e+3,nDist) #maximum mean
> sigmamax <- rep(1e+8,nDist) #maximum sigma
> alphamax <- rep(15,nDist) #maximum alpha
> sigma.rmax <- 1e+5 #maximum detection function parameter.
```

then combine the parameter space maxima in a single vector:

```
> upper.b <- as.vector(matrix(c(mumax,sigmamax,alphamax),ncol=nDist,byrow=TRUE))
> upper.b <- c(upper.b[-length(upper.b)],sigma.rmax) #remove nth weight
```

We can now fit the two normal mixture distribution using the above settings:

```
> mn.environ.fit <- nupoint.env.fit(pars=pars,
+              z=sighting.mat$z,
+                       rd=sighting.mat$r,
+                        dzdy=sighting.mat$dzdy,
+                       z.mat=z.mat,
+                       dzdy.mat=zGradmat,
+                       rd.mat=rd.mat,
+                       minz=minz,
+                       wx=wx,
+                       wy=wy,
+                       wz=wz,
+                       grad.type=grad.type,
+                       det.type=det.type,
+                       n=nDist,lower.b=lower.b,upper.b=upper.b)
```

```
------------------------------------------------------
Environmental gradient likelihood settings
------------------------------------------------------
Environment preference parametric form: MNORM
range detection function, g(r), parametric form: HNORM
Mixture of normal distributions with starting values =
                 mu sigma weight
mixture-1  103.2158   300    0.5
mixture-2 1903.9510   300    0.5
Detection function starting values  = 3000
Truncation distances
x= 6450  ; y= 8850  ; z= 1903.951
------------------------ ------------------------------
Estimating parameters
------------------------------------------------------
Maximum likelihood results
------------------------------------------------------
parameter point estimates = -349.812 226.903 0.29 1145.626 429.287 3023.626
AIC = 3894.54
------------------------------------------------------
```

Results of the AIC model selection(Table 3) show that the Normal distribution was optimum for explaining whale cue habitat preference.

	AIC	dAIC	AIC weight
Normal	3893.23	0.00	0.65
2-normal mixture	3894.54	1.31	0.34
Beta	3903.61	10.38	0.00
Lognormal	3903.84	10.61	0.00
Uniform	3937.50	44.27	0.00

Table 3: AIC model selection results for whale cue prevelance with depth. From the four candidate models, the Normal distribution was optimum for describing whale cue preference with depth (AIC weight = 0.65)

3.2 Goodness-of-fit test

The `nupoint.env.gof` function is used to calculate a one-dimensional χ^2 goodness-of-fit test. The `nupoint.env.gof` function is available for all the environmental gradient functional forms. Here we calculate the one-dimensional goodness-of-fit statistic for the Normal form of the whale cue prevalence model. We selected the Normal form because this form was selected under AIC (Table 3):

```
> GoF.norm <- nupoint.env.gof(pars=norm.fit$par,
+         r.mat=rd.mat,
```

```
+          z.mat=z.mat,
+          minz=minz,
+          wz=wz,
+          z.obs=sighting.mat$z,
+          grad.type=norm.grad.type,
+          det.type=norm.det.type,
+          intervals=13,plot=FALSE,
+          dzdy.mat=zGradmat)

-------------------------------------------------------
1D Chi-squared Goodness-of-Fit results
-------------------------------------------------------
   bin.min bin.max    mids expected observed Chisq
1   103.22  241.73  172.47     5.08        7  0.73
2   241.73  380.25  310.99     8.81        6  0.89
3   380.25  518.77  449.51    15.43       16  0.02
4   518.77  657.29  588.03    22.63       17  1.40
5   657.29  795.81  726.55    24.77       24  0.02
6   795.81  934.32  865.07    28.88       29  0.00
7   934.32 1072.84 1003.58    25.65       30  0.74
8  1072.84 1211.36 1142.10    21.33       24  0.33
9  1211.36 1349.88 1280.62    16.85       19  0.28
10 1349.88 1488.40 1419.14    11.22        9  0.44
11 1488.40 1626.91 1557.66     6.46        7  0.04
12 1626.91 1765.43 1696.17     2.93        0  2.93
13 1765.43 1903.95 1834.69     0.97        2  1.10
-------------------------------------------------------
Chi-squ. statistic   =  8.92607
Number of parameters =  3
Chi-squ. df          =  9
Chi-squ. GoF p-value =  0.4441264
-------------------------------------------------------
```

A graphical output of the goodness-of-fit results is available by changing the argument `plot=TRUE` in `nupoint.env.gof` function (Figure 2). The `intervals` argument `nupoint.env.gof()` can also be used to manually specify break points for the goodness-of-fit bin intervals, allowing for varying bin widths.

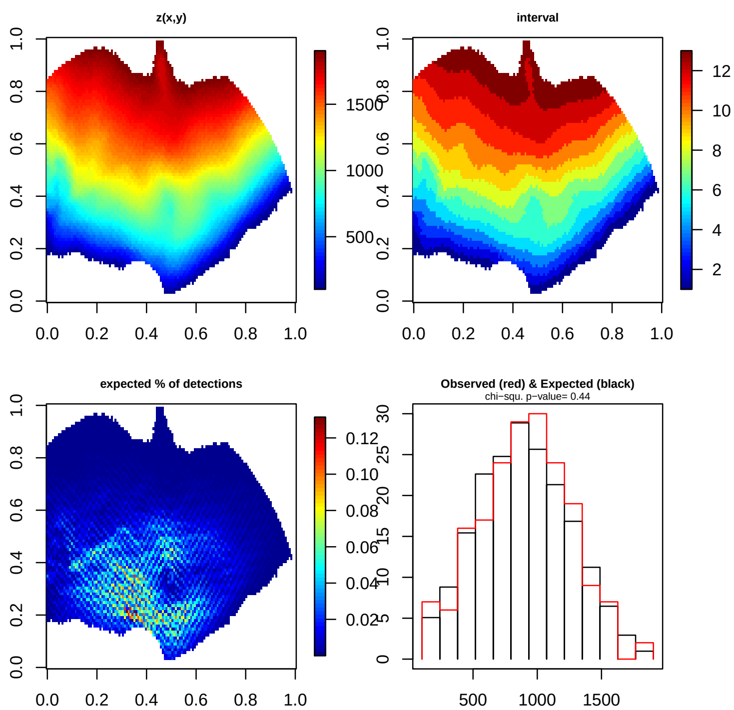

Figure 2: Graphical display of one-dimensional χ^2 goodness-of-fit test. The upper left panel is seabed depth, m. Upper right panel is the intervals over which observed whale cues are aggregated and expected sightings, under the normal model are calculated. The lower right panel is the expected percentage of detections, and lower right a histogram of observed sightings and expected sightings.

4 Non-parametric bootstrap

To estimate the variance of parameter estimates for the whale cue prevalence with depth and detection functions, we use the `nupoint.env.boot` function. This function requires the observation data, `sight.mat`, and matrices describing the survey region, e.g. seabed depth `sightings$z.mat` to be specified in a single list object (see Table 1 for object structure). The `sightings` list is passed into the `nupoint.env.boot` via the `sightings` argument. During each bootstrap iteration (number of iterations specified in `nboot` argument), the `nupoint.env.boot` function samples from `sightings`. The sampling unit is specified in the blocking variable (`bloackVar`) and may be passed to the `nupoint.env.boot` function as either the grouping variable column name or column number. In this case the grouping variable is specified by observation day `blockVar='obs.period'`.

In the following example we use the `nupoint.env.boot` function to calculate variance for the normal form of depth preference `grad.type='NORM'`. The normal form was selected as optimum from the candidate depth preference models by AIC (Table 3):

```
> boot.res <- nupoint.env.boot(sightings=sightings,
+                 nboot=100,
+                 blockVar='obs.period',
+                 initial.pars=c(1171.479,503.87,3020.442),
+                 grad.type='NORM',
+                 det.type='HNORM',
+                 lower.b=c(-2000,1,1),
+                 upper.b=c(10000,10000,10000))
```

We obtain variance estimates for each of the environmental preference gradient and half-normal detection function parameters (Table 4.)

	point.est	CV.percent	min	max
gradient mean	1171.48	68.11	940.19	10000.00
gradient sigma	503.87	44.34	364.53	2712.40
detection sigma	3020.44	8.15	2447.30	4008.33

Table 4: Bootstrap results for the normal density gradient and half-normal detection function.

5 A simulation example

The `nupoint.env.simulator` function enables one to simulate a survey region and observations within the simulated region. The survey region boundary is rectangular area, specified by the `xlim` and `ylim` arguments in the `nupoint.env.simulator` function. Within the survey region an environment feature, e.g., seabed depth, can either be passed into the simulator as a two-dimensional array using the `z.mat` argument, or can be simulated. In the case of an actual (real) survey region, the simulator function can be used to explore a range of plausible distributions of sightings with respect to the environmental feature of interest.

Environmental preference can be specified using one of the parametric distributions available in the `nupoint.env.fit` i.e. normal, log-normal, beta, uniform and mixture-normal, with detectability specified as per the `detectF` function. The true number of animals is specified via the `nbr.targets` argument.

5.1 Simulating the survey region

In the first example we have a predefined survey region. For the purpose of this example, we will create the environmental variable (seabed depth) outside of the `nupoint.env.simulator` function:

```
> z.mat <- matrix(rep(1:200,each=100),nrow=100,byrow=TRUE)
>
```

and display the `z.mat` object (Figure 3).

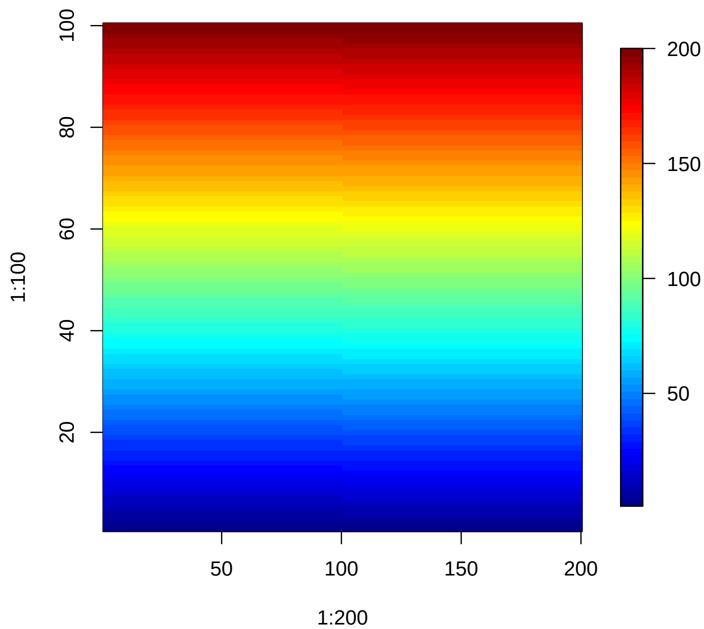

Figure 3: An environmental feature, seabed depth, within a predefined survey region. The coastline runs along the x-axis, with seabed depth increasing linearly with distance from the coastline.

The predefined survey region, in the `z.mat` object, can now be passed into the `nupoint.env.simulator` and a true underlying spatial distribution of animals simulated. The spatial distribution of animals is simulated using the environmental gradient functional form specified in the `grad.type` argument, with parameters `pars` (Figure 4). Sightings of the true distribution of animals are simulated based on the detection function `det.type`:

```
environ.sim.pre.dat <-nupoint.env.simulator(pars=c(100,10,50),
                                  z.mat=z.mat,
                      xlim=c(0,200),ylim=c(0,100),
                      grid.resolution=1,grad.type='NORM',det.type='HNORM',
                      observer.coords=c(100,0),nbr.targets=1000,
                      environment.simulator.control=NULL,
                      mask.mat=NULL,mask.ang=0,plot=TRUE,
                      perp.lines=NULL,n=NULL)
```

Simulator output has been assigned to `environ.sim.pre.dat`, and has the following structure: The `sightings` data frame contains detected simulated

	Length	Class	Mode
sightings	6	data.frame	list
rd.mat	20000	-none-	numeric
z.mat	20000	-none-	numeric
zGradmat	20000	-none-	numeric
settings	8	-none-	list

Table 5: Simulator data structure output

sightings, with each simulated sighting having data described as per Table 1, but without the `obs.period` variable. The `rd.mat`, `z.mat` and `zGradmat` objects are matrices of radial distance from observer to each cell in the survey area, seabed depth, and seabed depth gradient with respect the y-dimension. The `settings` object describes the simulation parameters (the `environ.sim.dat` function arguments).

In addition to using an existing survey region, we can use the `nupoint.env.simulator` function to simulate the environmental feature of interest. This is specified by setting the `z.mat=NULL` and using the `environment.simulator.control` argument, which is a list object, with each element comprising of a vector specifying the centre coordinates (`X, Y`) and standard deviation (`sd`) of a radial basis function. The radial basis functions are combined additively to create the environmental feature of interest:

```
environ.sim.dat<-nupoint.env.simulator(pars=c(60,10,50),
                                  z.mat=NULL,
                      xlim=c(0,200),ylim=c(0,100),
                      grid.resolution=1,grad.type='NORM',det.type='HNORM',
```

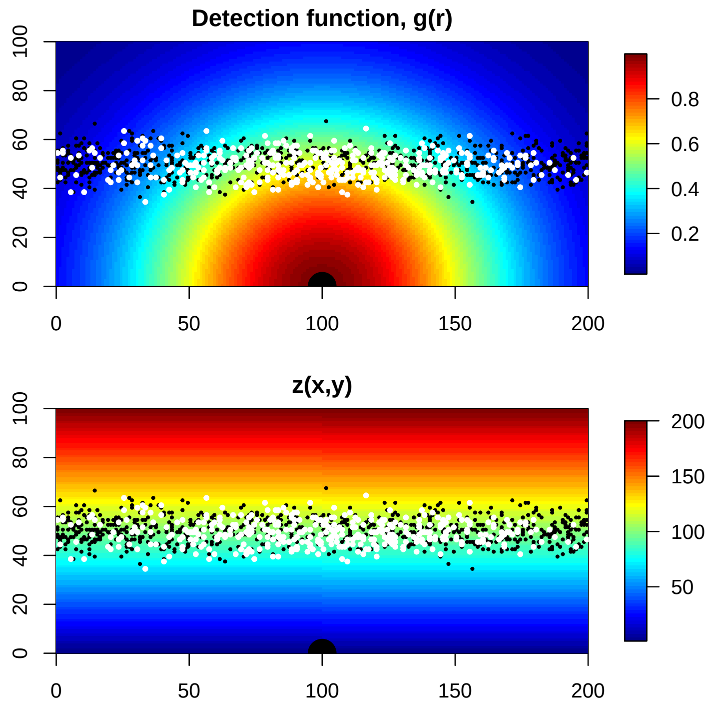

Figure 4: Simulation output based on an existing survey area. The upper panel shows the radial detection function, with higher detection probabilities in dark red, lower in dark blue. The lower panel shows seabed depth, the environmental feature of interest. All simulated animals are shown is both panels: detected animals as white dots, present, but undetected as black dots.

```
                              observer.coords=c(100,0),nbr.targets=1000,
                              environment.simulator.control=
                                                  list(c(X=50,Y=10,sd=60),c(X=90,Y=0,sd=30)),
                              mask.mat=NULL,mask.ang=0,plot=TRUE,
                              perp.lines=NULL,n=NULL)
```

The above `environment.simulator.control` specification results in a slightly perturbed seabed (Figure 5), with a distribution of animals specified by `grad.type='NORM'` and `pars=c(60,10,50)` that follows the perturbation.

Currently, the simulated environmental feature of interested is based on a linearly increasing feature, where the simulated environmental feature at Cartesian coordinates x, y, is the y coordinate perturbed by random noise and any radial basis functions specified in the `environment.simulator.control`. In the whale example, this means before perturbation, $z(x, y) = y$. The user may control use the use of the y coordinate as the simulated environmen-

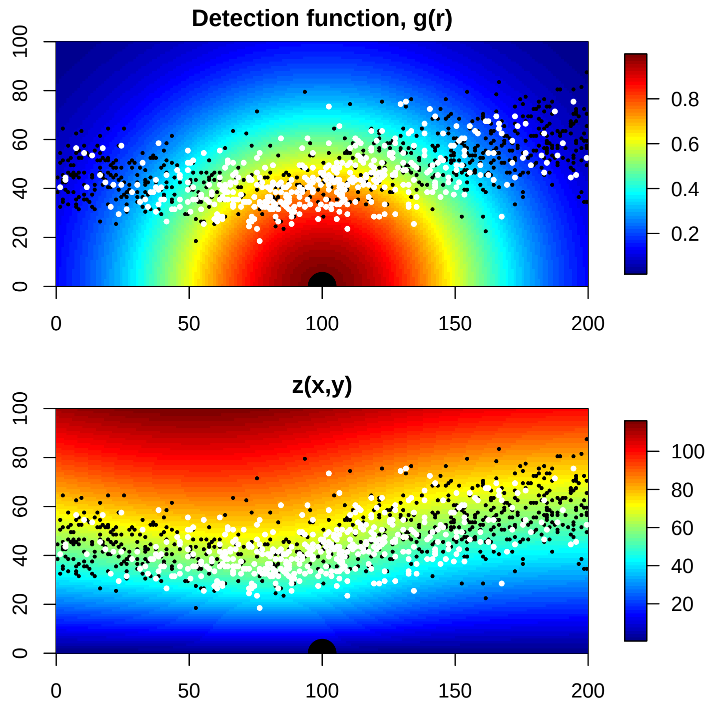

Figure 5: Simulation output based on a survey area generated using `environment.simulator.control=list(c(X=50,Y=10,sd=60),c(X=90,Y=0,sd=30))` in the `nupoint.env.simulator` function. The upper panel shows the radial detection function, with higher detection probabilities in dark red, lower in dark blue. The lower panel shows seabed depth, the environmental feature of interest. All simulated animals are shown is both panels: detected animals as white dots, undetected animals as black dots.

tal feature through the `slope.control` function. The `slope.control` argument is either `slope.control=NULL`, in which case $z(x, y) = y$ holds. Alternatively, `slope.control` can be specified as a two element numeric vector so that $z(x, y) =$ `slope.control[1]` $+y\times$ `slope.control[2]`, which is then perturbed by any radial basis functions specified in the `environment.simulator.control` argument. The user also has the option of passing in a a simulated environmental feature of interest to `environment.simulator.control` via the `z.mat` argument (see Section 5.1).

5.2 Maximum likelihood estimation using simulated data

Our objective in this section is to use the `nupoint.env.fit` function to estimate parameters, $\hat{\boldsymbol{\phi}}$, and compare this estimate to the parameters used to simulate the data. In this case, the parameters used in the simulation were $\boldsymbol{\phi} =$ `c(60,10,50)`, where $\boldsymbol{\phi}_{1,2}$ describe the normal form of the environmental gradient $\pi_z(z(x, y)) \sim norm(\mu = 50, \sigma = 10)$ and the detection function, $g(r)$, a half-normal form, $\sigma_r = \boldsymbol{\phi}_3 = 50$.

We then use `nupoint.env.fit` to estimate $\hat{\boldsymbol{\phi}}$:

```
> sim.norm.fit<-nupoint.env.fit(pars=c(60,10,50),
+                         z=environ.sim.dat$sightings$z,
+                         rd=environ.sim.dat$sightings$d,
+                         dzdy=environ.sim.dat$sightings$dzdy,
+                         z.mat=environ.sim.dat$z.mat,
+                         dzdy.mat=environ.sim.dat$zGradmat,
+                         rd.mat=environ.sim.dat$rd.mat,
+                         minz=min(environ.sim.dat$z.mat),
+                         wx=environ.sim.dat$settings$xlim[2],
+                         wy=environ.sim.dat$settings$ylim[2],
+                         wz=max(environ.sim.dat$z.mat),
+                         grad.type=environ.sim.dat$settings$grad.type,
+                         det.type=environ.sim.dat$settings$det.type,
+                         n=NULL,lower.b=rep(1,3),upper.b=rep(100,3))

------------------------------------------------------
Environmental gradient likelihood settings
------------------------------------------------------
Environment preference parametric form: NORM
range detection function, g(r), parametric form: HNORM
Parameter starting values = 60 10 50
Truncation distances
x= 200  ; y= 100  ; z= 115.9389
------------------------ ------------------------------
Estimating parameters
------------------------------------------------------
Maximum likelihood results
------------------------------------------------------
parameter point estimates = 61.36 9.773 47.248
AIC = 3931.15
------------------------------------------------------
```

The relative bias of our parameter estimates, $\hat{\boldsymbol{\phi}} =$ [61.36,9.77,47.25] was [0.02, -0.02, -0.06].

References

P. Arranz, D.L. Borchers, N. Aguilar Soto, M.P. Johnson, and M.J. Cox. A new method to study inshore whale cue distribution from land-based observations. *Marine Mammal Science*, submitted.