Introduction to R for distance sampling

3. Covariates in the detection function

In this practical, we illustrate fitting multiple covariate distance sampling (MCDS) models to point transect data using a bird survey in Hawaii; data on an abundant species, the Hawaii amakihi (*Hemignathus virens*) is used. This practical is based on the case study shown in Section 5.3.2 of Buckland *et al.* (2015) which duplicates the analysis presented in Marques *et al.* (2007). This set of data is included in the distribution of Distance as one of the Sample Projects. You can open this project (entitled amakihi.zip) in the Sample project directory underneath 'My Distance projects' directory residing under 'My Documents'. This document describes the analysis of these data using R.

Objectives of this practical

1. Introduce different types of plots to explore covariates
2. Add covariates to the detection function
3. Plot the detection functions.

Importing the data

Analysis begins by importing the data from a comma-delimited file (this file was created by copying the data from the amakihi Distance project and removing a few columns). Remember to copy the datafile to your R workspace or specify the directory where the data file is stored in the file name

```
# Import Amakihi data
amakihi <- read.csv(file="amakihi.csv")

# Check that it has been imported correctly
head(amakihi)
```

```
##   Study.Area Region.Label Sample.Label Effort distance OBS MAS HAS
## 1       Kana       Jul-92            1      1       40 TJS  50   1
## 2       Kana       Jul-92            1      1       60 TJS  50   1
## 3       Kana       Jul-92            1      1       45 TJS  50   1
## 4       Kana       Jul-92            1      1      100 TJS  50   1
## 5       Kana       Jul-92            1      1      125 TJS  50   1
## 6       Kana       Jul-92            1      1      120 TJS  50   1
```

These data are made of of eight columns:

- Study.Area - name of the study area
- Region.Label - survey dates which are used as 'strata'
- Sample.Label - point transect identifier
- Effort - survey effort (which is always 1 because point transects used)
- distance - perpendicular distances
- OBS - initials of the observer
- MAS - minutes after sunrise

- HAS - hour after sunrise

The latter three columns are the covariates to be considered for possible inclusion into the detection function.

There a couple of records with missing distances and so can be deleted with the following command:

```
# Get rid of missing values
amakihi <- amakihi[!is.na(amakihi$distance), ]
```

In this command,

- records in `amakihi` are selected using the square brackets `[]`
- `amakihi` is a data frame and so selection can be performed on either rows or columns i.e. `[rows, columns]`. In this case, the selection is performed on the rows (because the selection criteria is before the comma) and all columns will be retained
- the rows selected as those where the distances (stored in `amakihi$distance`) are not missing. The function `is.na` selects elements that are missing; the symbol `!` means 'not', and so `!is.na` selects elements that are not missing.

Exploratory data analysis

It is important to gain an understanding of the data prior to fitting detection functions (Buckland *et al.* 2015). With this in mind, preliminary analysis of distance sampling data involves:

- assessing the shape of the collected data,
- considering the level of truncation of distances, and
- exploring patterns in potential covariates.

Using the `summary` function provides a quick way to summarise of all the columns in the data set.

```
summary(amakihi)
```

```
##  Study.Area  Region.Label  Sample.Label       Effort     distance
##  Kana:1485   Apr-93:274   Min.   : 1.00   Min.   :1   Min.   :  1.00
##              Apr-94:148   1st Qu.: 9.00   1st Qu.:1   1st Qu.: 27.00
##              Apr-95:250   Median :19.00   Median :1   Median : 45.00
##              Dec-92:158   Mean   :19.61   Mean   :1   Mean   : 50.67
##              Jan-94:262   3rd Qu.:30.00   3rd Qu.:1   3rd Qu.: 70.00
##              Jul-92:183   Max.   :41.00   Max.   :1   Max.   :250.00
##              Jul-93:210
##   OBS            MAS             HAS
##     :   0   Min.   :-18.0   Min.   :0.000
##  SGF: 229   1st Qu.: 78.0   1st Qu.:1.000
##  TJS:1183   Median :137.0   Median :2.000
##  TKP:  73   Mean   :140.9   Mean   :2.344
##             3rd Qu.:199.0   3rd Qu.:3.000
##             Max.   :307.0   Max.   :5.000
##
```

We begin by assessing the distribution of distances by plotting histograms with different number of bins and different truncation.

Plot a histogram of the distances with lots of bins and no truncation. In the command below, `seq(0, 260)` creates a sequence of numbers from 0 to 260, inclusive, at unit intervals (e.g. 0, 1, 2, 3, . . . , 260):

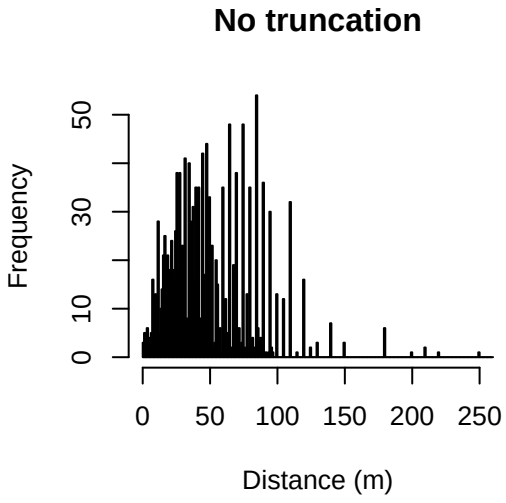
```
hist(amakihi$distance, breaks=seq(0,260), main="",xlab = "Distance (m)")
```

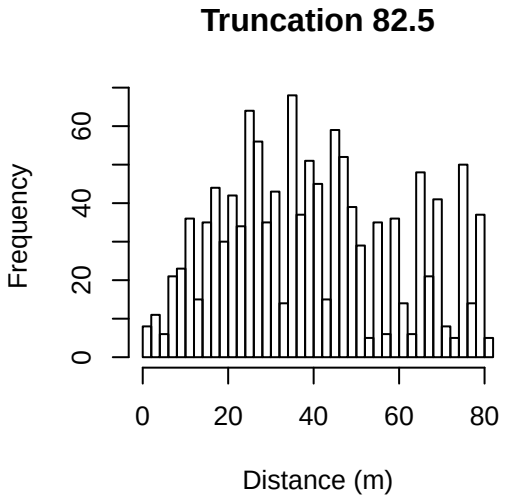
Plot a histogram with lots of bins and truncation at 82.5 - this is the truncation distance that was used in Marques *et al.* (2007):

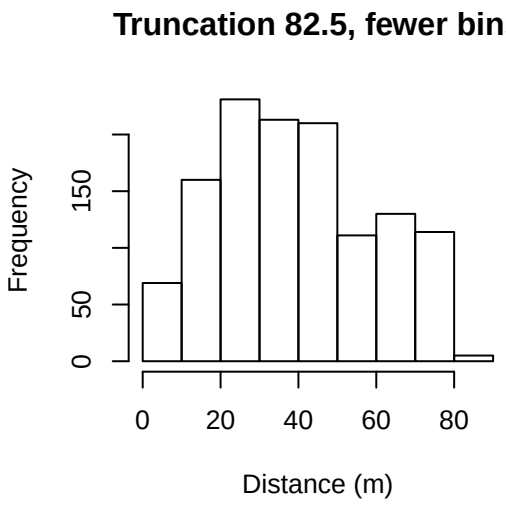
```
hist(amakihi$distance[amakihi$distance<82.5], breaks=33, main="",xlab = "Distance (m)")
```

Plot a histogram with truncation at 82.5 but with fewer bins:

```
hist(amakihi$distance[amakihi$distance<82.5], breaks=10, main="",xlab = "Distance (m)")
```



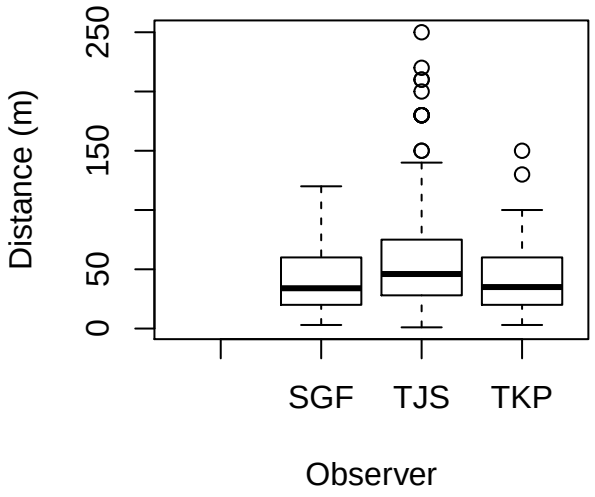




Next we look at the distribution of distances for each of the potential covariates. In this case, we use boxplots to show the distribution of distances recorded by each observer and for each hour after sunrise.

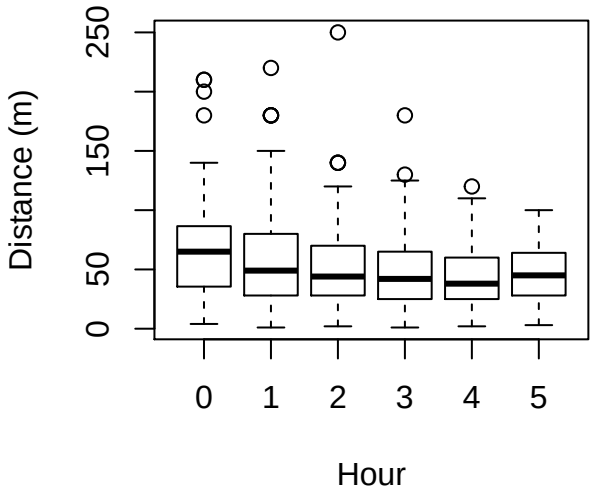
Boxplots of distances by observer:

```
boxplot(amakihi$distance~amakihi$OBS, xlab="Observer", ylab="Distance (m)")
```



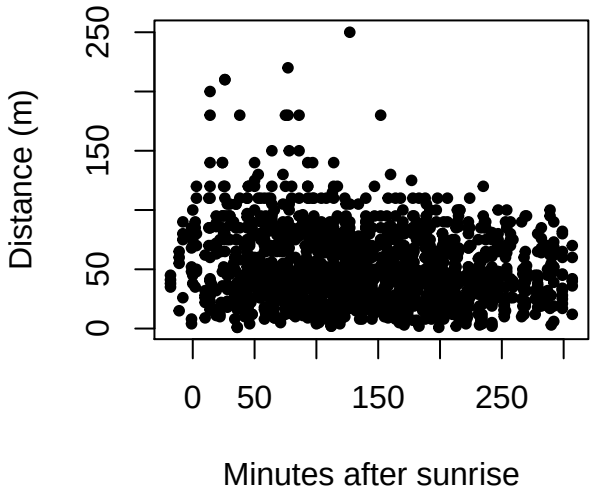
Boxplot of distances for each hour after sunrise:

```
boxplot(amakihi$distance~amakihi$HAS, xlab="Hour", ylab="Distance (m)")
```



For minutes after sunrise, we create a scatterplot of MAS (on the x-axis) against distances (on the y-axis). The plotting symbol (or character) selected is a dot with the argument `pch=20`:

```
plot(amakihi$MAS, amakihi$distance, xlab="Minutes after sunrise", ylab="Distance (m)",
pch=19, cex=0.5)
```



Adjusting the raw covariates

We would like to treat OBS and HAS as factor variables as in the original analysis; OBS is, by default, treated as a factor variable because it consists of characters rather than numbers. `HAS`, on the other hand, consists of numbers and so by default would be treated as a continuous variable (i.e. non-factor). That is fine if we want the effect of `HAS` to be monotonic (i.e. detectability either increases or decreases as a function of `HAS`). If we want `HAS` to have a non-linear effect on detectability, then we need to indicate to R to treat it as a factor as shown below.

```
# Convert HAS to a factor
amakihi$HAS <- as.factor(amakihi$HAS)
```

The next adjustment is to change the *reference* level of the *observer* and *hour* factor covariates - the only reason to do this is to get the estimated parameters in the detection function to match the parameters estimated by `Distance`. By default R uses the first factor level but by using the `relevel` function, this can be changed:

```
# Set the reference level
amakihi$OBS <- relevel(amakihi$OBS, ref="TKP")
amakihi$HAS <- relevel(amakihi$HAS, ref="5")
```

One final adjustment, and more subtle, is a transformation of the continuous covariate, `MAS`. We are entertaining three possible covariates in our detection function: `OBS`, `HAS` and `MAS`. The first two variables, `OBS` and `HAS`, are both factor variables, and so, essentially, we can think of them as taking on values between 1 and 3 in the case of `OBS`, and 1 to 6 in the case of `HAS`. However, `MAS` can take on values from -18 (detections before sunrise) to >300 and the disparity in scales of measure between `MAS` and the other candidate covariates can lead to difficulties in the performance of the optimizer fitting the detection functions in R. The solution to the difficulty is to scale `MAS` such that it is on a scale (approx. 1 to 5) comparable with the other covariates.

Dividing all the `MAS` measurements by the standard deviation (function `sd`) of those measurements accomplishes the desired compaction in the range of the `MAS` covariate without changing the shape of the distribution of `MAS` values. The `na.rm=TRUE` argument ensures that any missing values are ignored.

```
# Rescale MAS by dividing by standard deviation
amakihi$MAS <- amakihi$MAS/sd(amakihi$MAS, na.rm=TRUE)
```

Check what it has done by looking at a summary of the adjusted MAS:

```
summary(amakihi$MAS)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## -0.2365  1.0250  1.8000  1.8510  2.6150  4.0330
```

Candidate models

With three potential covariates, there are 8 possible combinations for including them in the detection function:

- No covariates
- OBS
- HAS
- MAS
- OBS + HAS
- OBS + MAS
- HAS + MAS
- OBS + HAS + MAS

There are three possible key functions (although note that if the uniform function is selected, covariates cannot be included) and if adjustment terms are included, the total number of formula/key function/adjustment combinations is large. It is not best practice to take a scatter gun approach to detection function model fitting and so in Buckland *et al.* (2015) 13 combinations were considered. Here, we look at a subset of these as illustration of including covariates but first a hazard rate key function with no covariates and no adjustment terms is fitted.

If it is not already loaded, then first load the `Distance` package.

```
library(Distance)
```

Fit a hazard rate model with no covariates or adjustment terms. By default, line transects are assumed and because we want point transects, the argument `transect="point"` is specified:

```
hr.model0 <- ds(amakihi, transect="point", key="hr", truncation=82.5, adjustment=NULL,
order=0)
```

The fitted model can be investigated using the `summary` function:

```
summary(hr.model0)
```

Make a note of the AIC for this model.

```
##
## Summary for distance analysis
## Number of observations :  1243
## Distance range         :  0  -  82.5
##
## Model : Hazard-rate key function
## AIC   : 10807.55
##
## Detection function parameters
```

```
## Scale coefficient(s):
##             estimate         se
## (Intercept) 3.454538 0.06310866
##
## Shape coefficient(s):
##              estimate         se
## (Intercept) 0.8342899 0.06533116
##
##                          Estimate           SE         CV
## Average p              0.3285785   0.02013101 0.06126697
## N in covered region 3782.9619603 247.88659914 0.06552712
```

Now fit a hazard rate model with OBS as a covariate in the detection function and make a note of the AIC. Has it reduced by including a covariate?

```
hr.model1 <- ds(amakihi, transect="point", key="hr", formula=~OBS, truncation=82.5,
adjustment=NULL, order=0)
```

```
##
## Summary for distance analysis
## Number of observations :  1243
## Distance range         :  0  -  82.5
##
## Model : Hazard-rate key function
## AIC   : 10778.45
##
## Detection function parameters
## Scale coefficient(s):
##                estimate        se
## (Intercept)  3.15326620 0.1688433
## OBSSGF      -0.08884974 0.1807228
## OBSTJS       0.44132377 0.1649255
##
## Shape coefficient(s):
##              estimate         se
## (Intercept) 0.8689995 0.06262048
##
##                          Estimate           SE         CV
## Average p              0.3142721   0.02044163 0.06504436
## N in covered region 3955.1711340 274.23266878 0.06933522
```

Fit a hazard rate model with OBS and HAS in the detection function:

```
hr.model2 <- ds(amakihi, transect="point", key="hr", formula=~OBS+HAS, truncation=82.5,
adjustment=NULL, order=0)
```

```
##
## Summary for distance analysis
## Number of observations :  1243
## Distance range         :  0  -  82.5
##
## Model : Hazard-rate key function
## AIC   : 10783.14
##
## Detection function parameters
## Scale coefficient(s):
##                estimate        se
```

```
## (Intercept)  3.22920350 0.2320232
## OBSSGF      -0.09132230 0.1739767
## OBSTJS       0.40401895 0.1589206
## HAS0         0.23507785 0.2157231
## HAS1         0.01157636 0.1803708
## HAS2         0.01858175 0.1760987
## HAS3        -0.09531145 0.1773469
## HAS4        -0.11981839 0.1830174
##
## Shape coefficient(s):
##              estimate         se
## (Intercept) 0.8907441 0.06358871
##
##                          Estimate           SE         CV
## Average p              0.3197378   0.02004631 0.06269610
## N in covered region 3887.5607304 260.92915995 0.06711899
```

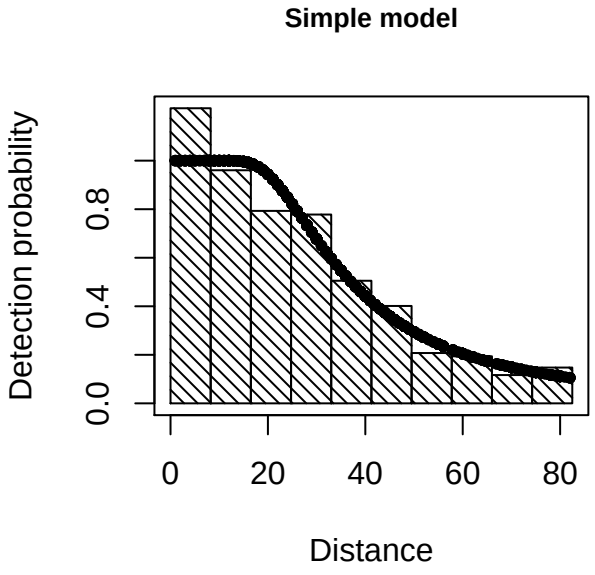
Try fitting other possible formula and decide which model is best in terms of AIC.

```
## The models, sorted in order of smallest AIC, is shown below.
##
##       formula key npar      aic  deltaaic
## 5     OBS+MAS  hr    5 10777.38  0.000000
## 2         OBS  hr    4 10778.45  1.072908
## 6     OBS+HAS  hr    9 10783.14  5.759905
## 8 OBS+MAS+HAS  hr   10 10785.12  7.743570
## 4         MAS  hr    3 10805.63 28.253419
## 1        None  hr    2 10807.55 30.173452
## 3         HAS  hr    7 10808.22 30.843087
## 7     MAS+HAS  hr    8 10810.22 32.841742
##
##  The model with the lowest AIC was the model with OBS+MAS.
```

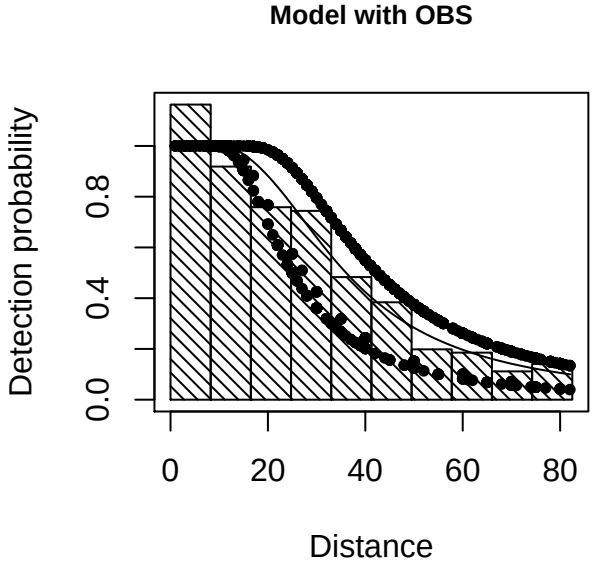
Plotting the detection functions

The detection functions can be investigated using the `plot` function as shown below:

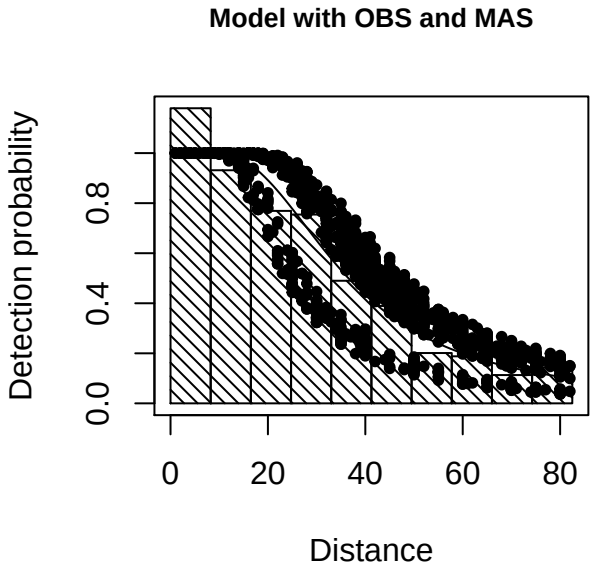
```
plot(hr.model0$ddf, nc=10, main="Simple model", pch=20)
```




```
plot(hr.model1$ddf, nc=10, main="Model with OBS", pch=20)
```



What does the detection function look like for your selected model?



To see more sophisticated examples of plotting the detection function for the selected model see the code accompanying Buckland *et al.* (2015) Hawaiian Amakihi case study.

Possible extensions

- Provide example of plotting detection function for each OBS and different levels of MAS.

References

Buckland ST, Rexstad EA, Marques TA and Oedekoven CS (2015) Distance Sampling: Methods and Applications. Springer 277 pp. ISBN: 978-3-319-19218-5 (Print) 978-3-319-19219-2 (Online)

Marques TA, Thomas L, Fancy SG and Buckland ST (2007) Improving estimates of bird density using multiple covariate distance sampling. The Auk 124:1229-1243