MCDS.exe Testing

Laura Marshall

2023-05-03

Things outstanding in mrds

In addition to the notes throughout this document the following also needs further work or investigations:

- Issue 83 in mrds relating to factor ordering differences between mrds and MCDS (https://github.com/DistanceDevelopment/mrds/issues/83)
- Reading in and processing the warnings and / or errors in the log file generated by MCDS - What do we do about cases where mcds.exe fits with negative pdf? E.g. https://github.com/DistanceDevelopment/Distance/issues/160
- Do the monotonicity constraints get passed to MCDS correctly?
- Check the passing of parameter starting values to MCDS - also potential issues here regarding factor ordering!
- Check the passing of parameter bound to MCDS - again potential for factor ordering issues. Note this can only be done via mrds NOT via Distance.

Capercaillie Data

Things that might want further investigation:

- When the R optimiser is selected both the lnl_R and lnl_MCDS values appear to be the same in all cases. (After moving on to other datasets this is not found to be the case.)

```
data("capercaillie")

model.compare <- test.models(capercaillie,
                             truncation = max(capercaillie$distance),
                             transect = "line",
                             convert.units = 10)

save(model.compare, file = "results/capercaillie.ROBJ")
```

Table 1: Comparison of R and MCDS model fits for Capercaillie data. key - key function, adj - type of adjustment term, nadj - number of adjustments, lnl_R - likelihood value for R optimiser, lnl_MCDS - likelihood value for MCDS optimiser, optimizer - the selected optimiser, p_R the estimated average probability of detection for the R optimised model, p_MCDS the estimated average probability of detection for the MCDS optimised model, Nhat_R - estimated abundance in covered region from R optimised model, Nhat_MCDS - estimated abundance in covered region from MCDS optimised model.

key	adj	nadj	lnl_R	lnl_MCDS	optimizer	p_R	p_MCDS	Nhat_R	Nhat_MCDS
unif	cos	0	-477.9525	-477.9525	mrds (nlminb)	0.61	0.61	182.76	182.76
unif	cos	1	-477.2076	-477.2057	MCDS.exe	0.66	0.66	169.09	169.04
unif	cos	2	-477.0673	-477.0672	MCDS.exe	0.69	0.69	162.03	162.04
unif	cos	3	-476.9576	-476.9459	MCDS.exe	0.69	0.68	163.20	164.18
hn	cos	0	-477.9525	-477.9525	mrds (nlminb)	0.61	0.61	182.76	182.76
hn	cos	1	-477.2076	-477.2057	MCDS.exe	0.66	0.66	169.09	169.04
hn	cos	2	-477.0673	-477.0672	MCDS.exe	0.69	0.69	162.03	162.04
hn	herm	0	-477.9525	-477.9525	mrds (nlminb)	0.61	0.61	182.76	182.76
hn	herm	1	-477.2173	-477.2162	MCDS.exe	0.67	0.67	166.20	166.15
hn	herm	2	-477.2205	-477.2162	MCDS.exe	0.67	0.67	166.04	166.14
hr	poly	0	-477.9525	-477.9525	mrds (nlminb)	0.61	0.61	182.76	182.76
hr	poly	1	-477.6612	-477.6612	mrds (nlminb)	0.64	0.64	175.11	175.12
hr	poly	2	-477.5669	-477.0932	MCDS.exe	0.64	0.68	174.65	164.64

Cue Counting Data

Things that might want further investigation:

- Nhat for the hn herm 1 model is ~14% higher for the MCDS optimised model than the R optimised model
- Why is the lnl_R value for the hr poly 2 model negative?

```
data("CueCountingExample")

model.compare <- test.models(CueCountingExample,
                             truncation = max(CueCountingExample$distance),
                             transect = "point")

save(model.compare, file = "results/cue_counting.ROBJ")
```

Table 2: Comparison of R and MCDS model fits for cue counting data. key - key function, adj - type of adjustment term, nadj - number of adjustments, lnl_R - likelihood value for R optimiser, lnl_MCDS - likelihood value for MCDS optimiser, optimizer - the selected optimiser, p_R the estimated average probability of detection for the R optimised model, p_MCDS the estimated average probability of detection for the MCDS optimised model, Nhat_R - estimated abundance in covered region from R optimised model, Nhat_MCDS - estimated abundance in covered region from MCDS optimised model.

key	adj	nadj	lnl_R	lnl_MCDS	optimizer	p_R	p_MCDS	Nhat_R	Nhat_MCDS
unif	cos	0	1.916333	1.916332	MCDS.exe	0.24	0.24	167.61	167.61
unif	cos	1	2.109706	2.109790	mrds (nlminb)	0.30	0.30	132.40	132.27
unif	cos	2	2.132097	2.132096	MCDS.exe	0.28	0.28	144.42	144.34
unif	cos	3	2.509622	2.512145	mrds (nlminb)	0.25	0.25	162.05	161.83
hn	cos	0	1.916333	1.916332	MCDS.exe	0.24	0.24	167.61	167.61
hn	cos	1	2.109706	2.109790	mrds (nlminb)	0.30	0.30	132.40	132.27
hn	cos	2	2.132097	2.132096	MCDS.exe	0.28	0.28	144.42	144.34
hn	herm	0	1.916333	1.916332	MCDS.exe	0.24	0.24	167.61	167.61
hn	herm	1	2.016928	1.925329	MCDS.exe	0.28	0.24	145.05	165.77
hn	herm	2	2.016971	3.876089	mrds (nlminb)	0.28	0.28	145.04	142.45
hr	poly	0	1.916333	1.916332	MCDS.exe	0.24	0.24	167.61	167.61
hr	poly	1	1.930192	1.930192	MCDS.exe	0.24	0.24	164.59	164.58
hr	poly	2	-7.983148	3.772227	MCDS.exe	0.31	0.32	130.88	126.38

Ducknest Data

Things that might want further investigation:

- Unhelpful error "Error in array(x, c(length(x), 1L), if (!is.null(names(x))) list(names(x), : 'data' must be of a vector type, was 'NULL' Error in t(partial) %*% vcov : requires numeric/complex matrix/vector arguments"
- What should happen when you have a uniform with no adjustments??? Shouldn't P always be 1?
- p_MCDS is 1 for a few of these models and when it is, it is estimating Nhat much lower than the R optimiser.

```
data("ducknest")

model.compare <- test.models(ducknest,
                             truncation = max(ducknest$distance),
                             transect = "line")

save(model.compare, file = "results/ducknest.ROBJ")
```

Table 3: Comparison of R and MCDS model fits for Ducknest data. key - key function, adj - type of adjustment term, nadj - number of adjustments, lnl_R - likelihood value for R optimiser, lnl_MCDS - likelihood value for MCDS optimiser, optimizer - the selected optimiser, p_R the estimated average probability of detection for the R optimised model, p_MCDS the estimated average probability of detection for the MCDS optimised model, Nhat_R - estimated abundance in covered region from R optimised model, Nhat_MCDS - estimated abundance in covered region from MCDS optimised model.

key	adj	nadj	lnl_R	lnl_MCDS	optimizer	p_R	p_MCDS	Nhat_R	Nhat_MCDS
unif	cos	0	-12539.80	-12539.80	mrds (nlminb)	0.26	0.26	22381.29	22380.94
unif	cos	1	-12476.04	-12475.87	MCDS.exe	0.33	0.33	17813.30	17796.84
unif	cos	2	-12458.57	-12458.56	MCDS.exe	0.37	0.37	15798.09	15799.97
unif	cos	3	-12437.63	-12434.56	MCDS.exe	0.34	0.37	17412.16	16068.38
hn	cos	0	-12539.80	-12539.80	mrds (nlminb)	0.26	0.26	22381.29	22380.94
hn	cos	1	-12476.04	-12475.87	MCDS.exe	0.33	0.33	17813.30	17796.84
hn	cos	2	-12458.57	-12458.56	MCDS.exe	0.37	0.37	15798.09	15799.97
hn	herm	0	-12539.80	-12539.80	mrds (nlminb)	0.26	0.26	22381.29	22380.94
hn	herm	1	-12472.93	-12510.95	mrds (nlminb)	0.34	0.29	17320.90	20043.80
hn	herm	2	-12472.49	NA	mrds (nlminb)	0.34	NA	17224.44	NA
hr	poly	0	-12539.80	-12539.80	mrds (nlminb)	0.26	0.26	22381.29	22380.94
hr	poly	1	-12506.68	-12506.68	MCDS.exe	0.30	0.30	19709.11	19708.10
hr	poly	2	-12505.64	-12468.78	MCDS.exe	0.30	0.33	19791.86	17927.86

DuikerCameraTraps

Things that might want further investigation: - - Unhelpful error "Error in array(x, c(length(x), 1L), if (!is.null(names(x))) list(names(x), : 'data' must be of a vector type, was 'NULL' Error in t(partial) %*% vcov : requires numeric/complex matrix/vector arguments" - Some of the Nhat values look to differ by around 20% between the two optimisers

```
data("DuikerCameraTraps")

model.compare <- test.models(DuikerCameraTraps,
                             truncation = list(left = 2, right = 15),
                             cutpoints = c(seq(2,8,1), 10, 12, 15),
                             transect = "point")

save(model.compare, file = "results/DuikerCameraTraps.ROBJ")
```

Table 4: Comparison of R and MCDS model fits for Duiker camera trap data. key - key function, adj - type of adjustment term, nadj - number of adjustments, lnl_R - likelihood value for R optimiser, lnl_MCDS - likelihood value for MCDS optimiser, optimizer - the selected optimiser, p_R the estimated average probability of detection for the R optimised model, p_MCDS the estimated average probability of detection for the MCDS optimised model, Nhat_R - estimated abundance in covered region from R optimised model, Nhat_MCDS - estimated abundance in covered region from MCDS optimised model.

key	adj	nadj	lnl_R	lnl_MCDS	optimizer	p_R	p_MCDS	Nhat_R	Nhat_MCDS
unif	cos	0	-12539.80	-12539.80	mrds (nlminb)	0.26	0.26	22381.29	22380.94
unif	cos	1	-12476.04	-12475.87	MCDS.exe	0.33	0.33	17813.30	17796.84
unif	cos	2	-12458.57	-12458.56	MCDS.exe	0.37	0.37	15798.09	15799.97
unif	cos	3	-12437.63	-12434.56	MCDS.exe	0.34	0.37	17412.16	16068.38
hn	cos	0	-12539.80	-12539.80	mrds (nlminb)	0.26	0.26	22381.29	22380.94
hn	cos	1	-12476.04	-12475.87	MCDS.exe	0.33	0.33	17813.30	17796.84
hn	cos	2	-12458.57	-12458.56	MCDS.exe	0.37	0.37	15798.09	15799.97
hn	herm	0	-12539.80	-12539.80	mrds (nlminb)	0.26	0.26	22381.29	22380.94
hn	herm	1	-12472.93	-12510.95	mrds (nlminb)	0.34	0.29	17320.90	20043.80
hn	herm	2	-12472.49	NA	mrds (nlminb)	0.34	NA	17224.44	NA
hr	poly	0	-12539.80	-12539.80	mrds (nlminb)	0.26	0.26	22381.29	22380.94
hr	poly	1	-12506.68	-12506.68	MCDS.exe	0.30	0.30	19709.11	19708.10
hr	poly	2	-12505.64	-12468.78	MCDS.exe	0.30	0.33	19791.86	17927.86

LTExercise Data

Things that might want further investigation: - - Why is is warning about "Warning in process.data(data, meta.data, check = FALSE) : no truncation distance specified; using largest observed distance"?

```
data("LTExercise")

model.compare <- test.models(LTExercise,
                             truncation = max(LTExercise$distance),
                             transect = "line")

save(model.compare, file = "results/LTExercise.ROBJ")
```

Table 5: Comparison of R and MCDS model fits for LTExercise data. key - key function, adj - type of adjustment term, nadj - number of adjustments, lnl_R - likelihood value for R optimiser, lnl_MCDS - likelihood value for MCDS optimiser, optimizer - the selected optimiser, p_R the estimated average probability of detection for the R optimised model, p_MCDS the estimated average probability of detection for the MCDS optimised model, Nhat_R - estimated abundance in covered region from R optimised model, Nhat_MCDS - estimated abundance in covered region from MCDS optimised model.

key	adj	nadj	lnl_R	lnl_MCDS	optimizer	p_R	p_MCDS	Nhat_R	Nhat_MCDS
unif	cos	0	-317.4681	-317.4681	mrds (nlminb)	0.35	0.35	300.70	300.70
unif	cos	1	-317.4680	-317.4680	MCDS.exe	0.35	0.35	300.21	300.26
unif	cos	2	-317.2043	-317.2043	MCDS.exe	0.37	0.37	281.75	281.75
unif	cos	3	-315.6816	-315.6505	MCDS.exe	0.33	0.33	319.76	319.21
hn	cos	0	-317.4681	-317.4681	mrds (nlminb)	0.35	0.35	300.70	300.70
hn	cos	1	-317.4680	-317.4680	MCDS.exe	0.35	0.35	300.21	300.26
hn	cos	2	-317.2043	-317.2043	MCDS.exe	0.37	0.37	281.75	281.75
hn	herm	0	-317.4681	-317.4681	mrds (nlminb)	0.35	0.35	300.70	300.70
hn	herm	1	-317.4610	-317.4610	mrds (nlminb)	0.35	0.35	300.77	300.78
hn	herm	2	-315.4484	-315.4070	MCDS.exe	0.34	0.34	311.17	306.79
hr	poly	0	-317.4681	-317.4681	mrds (nlminb)	0.35	0.35	300.70	300.70
hr	poly	1	-317.3553	-317.3553	MCDS.exe	0.34	0.34	304.63	304.63
hr	poly	2	-317.1466	-319.4356	mrds (nlminb)	0.35	0.35	301.95	300.71

PTExercise Data

Things that might want further investigation: - - Why is is warning about "Warning in process.data(data, meta.data, check = FALSE) : no truncation distance specified; using largest observed distance"?

```
data("PTExercise")

model.compare <- test.models(PTExercise,
                             truncation = max(PTExercise$distance),
                             transect = "point")

save(model.compare, file = "results/PTExercise.ROBJ")
```

Table 6: Comparison of R and MCDS model fits for PTExercise data. key - key function, adj - type of adjustment term, nadj - number of adjustments, lnl_R - likelihood value for R optimiser, lnl_MCDS - likelihood value for MCDS optimiser, optimizer - the selected optimiser, p_R the estimated average probability of detection for the R optimised model, p_MCDS the estimated average probability of detection for the MCDS optimised model, Nhat_R - estimated abundance in covered region from R optimised model, Nhat_MCDS - estimated abundance in covered region from MCDS optimised model.

key	adj	nadj	lnl_R	lnl_MCDS	optimizer	p_R	p_MCDS	Nhat_R	Nhat_MCDS
unif	cos	0	-458.5701	-458.5701	mrds (nlminb)	0.16	0.16	875.75	875.75
unif	cos	1	-457.8850	-457.8850	mrds (nlminb)	0.20	0.20	726.49	726.28
unif	cos	2	-457.8646	-457.8646	mrds (nlminb)	0.21	0.21	700.66	700.37
unif	cos	3	-457.3805	-457.3805	mrds (nlminb)	0.17	0.17	841.14	840.58
hn	cos	0	-458.5701	-458.5701	mrds (nlminb)	0.16	0.16	875.75	875.75
hn	cos	1	-457.8850	-457.8850	mrds (nlminb)	0.20	0.20	726.49	726.28
hn	cos	2	-457.8646	-457.8646	mrds (nlminb)	0.21	0.21	700.66	700.37
hn	herm	0	-458.5701	-458.5701	mrds (nlminb)	0.16	0.16	875.75	875.75
hn	herm	1	-458.1114	-458.5393	mrds (nlminb)	0.19	0.17	750.89	866.98
hn	herm	2	-458.1115	-458.1112	MCDS.exe	0.19	0.19	753.01	750.65
hr	poly	0	-458.5701	-458.5701	mrds (nlminb)	0.16	0.16	875.75	875.75
hr	poly	1	-458.5174	-458.5174	mrds (nlminb)	0.17	0.17	861.26	861.19
hr	poly	2	-458.3287	-456.7638	MCDS.exe	0.17	0.19	857.51	750.54

Savannah Sparrow 1980

Everything looks to run fine.

```
data("Savannah_sparrow_1980")

model.compare <- test.models(Savannah_sparrow_1980,
                             truncation = max(Savannah_sparrow_1980$distance, na.rm = TRUE),
                             transect = "point")

save(model.compare, file = "results/Savannah_sparrow_1980.ROBJ")
```

Table 7: Comparison of R and MCDS model fits for Savannah sparrow 1980 data. key - key function, adj - type of adjustment term, nadj - number of adjustments, lnl_R - likelihood value for R optimiser, lnl_MCDS - likelihood value for MCDS optimiser, optimizer - the selected optimiser, p_R the estimated average probability of detection for the R optimised model, p_MCDS the estimated average probability of detection for the MCDS optimised model, Nhat_R - estimated abundance in covered region from R optimised model, Nhat_MCDS - estimated abundance in covered region from MCDS optimised model.

key	adj	nadj	lnl_R	lnl_MCDS	optimizer	p_R	p_MCDS	Nhat_R	Nhat_MCDS
unif	cos	0	-1103.866	-1103.866	mrds (nlminb)	0.15	0.15	1827.90	1827.80
unif	cos	1	-1103.810	-1103.810	MCDS.exe	0.15	0.15	1886.75	1887.79
unif	cos	2	-1102.887	-1102.887	MCDS.exe	0.13	0.13	2153.27	2152.34
unif	cos	3	-1102.863	-1102.863	mrds (nlminb)	0.13	0.13	2198.21	2196.94
hn	cos	0	-1103.866	-1103.866	mrds (nlminb)	0.15	0.15	1827.90	1827.80
hn	cos	1	-1103.810	-1103.810	MCDS.exe	0.15	0.15	1886.75	1887.79
hn	cos	2	-1102.887	-1102.887	MCDS.exe	0.13	0.13	2153.27	2152.34
hn	herm	0	-1103.866	-1103.866	mrds (nlminb)	0.15	0.15	1827.90	1827.80
hn	herm	1	-1103.864	-1103.864	MCDS.exe	0.15	0.15	1828.57	1828.97
hn	herm	2	-1102.912	-1102.593	MCDS.exe	0.16	0.16	1734.66	1694.40
hr	poly	0	-1103.866	-1103.866	mrds (nlminb)	0.15	0.15	1827.90	1827.80
hr	poly	1	-1103.856	-1103.856	mrds (nlminb)	0.15	0.15	1844.13	1843.92
hr	poly	2	-1103.855	-1103.354	MCDS.exe	0.15	0.17	1839.73	1613.36

Savannah Sparrow 1981

Everything looks to run fine.

```
data("Savannah_sparrow_1981")

model.compare <- test.models(Savannah_sparrow_1981,
                             truncation = max(Savannah_sparrow_1981$distance, na.rm = TRUE),
                             transect = "point")

save(model.compare, file = "results/Savannah_sparrow_1981.ROBJ")
```

Table 8: Comparison of R and MCDS model fits for Savannah sparrow 1981 data. key - key function, adj - type of adjustment term, nadj - number of adjustments, lnl_R - likelihood value for R optimiser, lnl_MCDS - likelihood value for MCDS optimiser, optimizer - the selected optimiser, p_R the estimated average probability of detection for the R optimised model, p_MCDS the estimated average probability of detection for the MCDS optimised model, Nhat_R - estimated abundance in covered region from R optimised model, Nhat_MCDS - estimated abundance in covered region from MCDS optimised model.

key	adj	nadj	lnl_R	lnl_MCDS	optimizer	p_R	p_MCDS	Nhat_R	Nhat_MCDS
unif	cos	0	-790.8846	-790.8846	mrds (nlminb)	0.16	0.16	1153.99	1153.93
unif	cos	1	-790.1360	-790.1360	mrds (nlminb)	0.14	0.14	1336.80	1336.55
unif	cos	2	-788.3169	-788.3169	mrds (nlminb)	0.17	0.17	1083.33	1083.00
unif	cos	3	-788.3125	-788.3124	MCDS.exe	0.17	0.17	1072.20	1069.79
hn	cos	0	-790.8846	-790.8846	mrds (nlminb)	0.16	0.16	1153.99	1153.93
hn	cos	1	-790.1360	-790.1360	mrds (nlminb)	0.14	0.14	1336.80	1336.55
hn	cos	2	-788.3169	-788.3169	mrds (nlminb)	0.17	0.17	1083.33	1083.00
hn	herm	0	-790.8846	-790.8846	mrds (nlminb)	0.16	0.16	1153.99	1153.93
hn	herm	1	-790.8620	-790.8620	mrds (nlminb)	0.16	0.16	1154.83	1154.77
hn	herm	2	-788.4557	-788.4557	mrds (nlminb)	0.15	0.15	1258.17	1257.86
hr	poly	0	-790.8846	-790.8846	mrds (nlminb)	0.16	0.16	1153.99	1153.93
hr	poly	1	-790.2692	-790.1383	MCDS.exe	0.16	0.15	1194.24	1225.36
hr	poly	2	-789.8253	-788.4404	MCDS.exe	0.15	0.15	1218.63	1240.13

Sika Deer

Everything looks to run fine.

```
data("sikadeer")

model.compare <- test.models(sikadeer,
                             truncation = max(sikadeer$distance, na.rm = TRUE),
                             transect = "line")

save(model.compare, file = "results/sikadeer.ROBJ")
```

Table 9: Comparison of R and MCDS model fits for sikadeer data. key - key function, adj - type of adjustment term, nadj - number of adjustments, lnl_R - likelihood value for R optimiser, lnl_MCDS - likelihood value for MCDS optimiser, optimizer - the selected optimiser, p_R the estimated average probability of detection for the R optimised model, p_MCDS the estimated average probability of detection for the MCDS optimised model, Nhat_R - estimated abundance in covered region from R optimised model, Nhat_MCDS - estimated abundance in covered region from MCDS optimised model.

key	adj	nadj	lnl_R	lnl_MCDS	optimizer	p_R	p_MCDS	Nhat_R	Nhat_MCDS
unif	cos	0	-10117.72	-10117.72	mrds (nlminb)	0.77	0.77	2506.17	2506.17
unif	cos	1	-10104.13	-10104.13	MCDS.exe	0.65	0.65	2939.25	2939.16
unif	cos	2	-10101.39	-10101.15	MCDS.exe	0.69	0.69	2792.91	2796.22
unif	cos	3	-10097.53	-10096.31	MCDS.exe	0.64	0.63	3011.35	3048.37
hn	cos	0	-10117.72	-10117.72	mrds (nlminb)	0.77	0.77	2506.17	2506.17
hn	cos	1	-10104.13	-10104.13	MCDS.exe	0.65	0.65	2939.25	2939.16
hn	cos	2	-10101.39	-10101.15	MCDS.exe	0.69	0.69	2792.91	2796.22
hn	herm	0	-10117.72	-10117.72	mrds (nlminb)	0.77	0.77	2506.17	2506.17
hn	herm	1	-10117.29	-10117.29	MCDS.exe	0.77	0.77	2509.27	2509.21
hn	herm	2	-10096.92	-10096.82	MCDS.exe	0.67	0.66	2887.09	2905.58
hr	poly	0	-10117.72	-10117.72	mrds (nlminb)	0.77	0.77	2506.17	2506.17
hr	poly	1	-10107.26	-10105.23	MCDS.exe	0.73	0.71	2623.99	2695.62
hr	poly	2	-10100.85	-10096.62	MCDS.exe	0.71	0.66	2708.49	2891.20

Wren 5 minute count

This one brings up the warning 'Detection function is less than 0 at some distances' for quite a few of the models. This warning occured for the last model where there is a substantial difference in abundance estimates.

```
data("wren_5min")

model.compare <- test.models(wren_5min,
                             truncation = max(wren_5min$distance, na.rm = TRUE),
                             transect = "point")

save(model.compare, file = "results/wren_5min.ROBJ")
```

Table 10: Comparison of R and MCDS model fits for wren_5min data. key - key function, adj - type of adjustment term, nadj - number of adjustments, lnl_R - likelihood value for R optimiser, lnl_MCDS - likelihood value for MCDS optimiser, optimizer - the selected optimiser, p_R the estimated average probability of detection for the R optimised model, p_MCDS the estimated average probability of detection for the MCDS optimised model, Nhat_R - estimated abundance in covered region from R optimised model, Nhat_MCDS - estimated abundance in covered region from MCDS optimised model.

key	adj	nadj	lnl_R	lnl_MCDS	optimizer	p_R	p_MCDS	Nhat_R	Nhat_MCDS
unif	cos	0	-604.6857	-604.6857	mrds (nlminb)	0.26	0.26	524.24	524.22
unif	cos	1	-600.3846	-600.3704	MCDS.exe	0.31	0.31	428.89	428.51
unif	cos	2	-597.6245	-597.5602	MCDS.exe	0.37	0.38	360.38	355.28
unif	cos	3	-597.3345	-597.0976	MCDS.exe	0.36	0.36	374.92	372.17
hn	cos	0	-604.6857	-604.6857	mrds (nlminb)	0.26	0.26	524.24	524.22
hn	cos	1	-600.3846	-600.3704	MCDS.exe	0.31	0.31	428.89	428.51
hn	cos	2	-597.6245	-597.5602	MCDS.exe	0.37	0.38	360.38	355.28
hn	herm	0	-604.6857	-604.6857	mrds (nlminb)	0.26	0.26	524.24	524.22
hn	herm	1	-600.1696	-603.7420	mrds (nlminb)	0.32	0.27	418.55	493.43
hn	herm	2	-600.1204	-600.0568	MCDS.exe	0.32	0.32	418.34	413.04
hr	poly	0	-604.6857	-604.6857	mrds (nlminb)	0.26	0.26	524.24	524.22
hr	poly	1	-603.4410	-603.4410	mrds (nlminb)	0.28	0.28	485.25	485.24
hr	poly	2	-603.1873	-595.2352	MCDS.exe	0.28	0.36	484.34	372.71

Wren cue counting

This one brings up the warning 'Detection function is less than 0 at some distances' for quite a few of the models. This warning occured for the last model where there is a substantial difference in abundance estimates.

```
data("wren_cuecount")

model.compare <- test.models(wren_cuecount,
                             truncation = max(wren_cuecount$distance, na.rm = TRUE),
                             transect = "point")

save(model.compare, file = "results/wren_cuecount.ROBJ")
```

Table 11: Comparison of R and MCDS model fits for wren_cuecount data. key - key function, adj - type of adjustment term, nadj - number of adjustments, lnl_R - likelihood value for R optimiser, lnl_MCDS - likelihood value for MCDS optimiser, optimizer - the selected optimiser, p_R the estimated average probability of detection for the R optimised model, p_MCDS the estimated average probability of detection for the MCDS optimised model, Nhat_R - estimated abundance in covered region from R optimised model, Nhat_MCDS - estimated abundance in covered region from MCDS optimised model.

key	adj	nadj	lnl_R	lnl_MCDS	optimizer	p_R	p_MCDS	Nhat_R	Nhat_MCDS
unif	cos	0	-3446.447	-3446.447	mrds (nlminb)	0.13	0.13	5883.48	5883.13
unif	cos	1	-3415.952	-3415.825	MCDS.exe	0.17	0.17	4512.30	4504.66
unif	cos	2	-3410.772	-3410.746	MCDS.exe	0.19	0.19	4074.67	4072.11
unif	cos	3	-3402.460	-3401.425	MCDS.exe	0.20	0.21	3837.94	3742.83
hn	cos	0	-3446.447	-3446.447	mrds (nlminb)	0.13	0.13	5883.48	5883.13
hn	cos	1	-3415.952	-3415.825	MCDS.exe	0.17	0.17	4512.30	4504.66
hn	cos	2	-3410.772	-3410.746	MCDS.exe	0.19	0.19	4074.67	4072.11
hn	herm	0	-3446.447	-3446.447	mrds (nlminb)	0.13	0.13	5883.48	5883.13
hn	herm	1	-3416.716	-3497.836	mrds (nlminb)	0.17	0.15	4518.43	5224.07
hn	herm	2	-3416.781	-3435.901	mrds (nlminb)	0.17	0.15	4507.43	5128.45
hr	poly	0	-3446.447	-3446.447	mrds (nlminb)	0.13	0.13	5883.48	5883.13
hr	poly	1	-3443.161	-3443.161	MCDS.exe	0.14	0.14	5705.77	5703.63
hr	poly	2	-3442.462	-3424.355	MCDS.exe	0.14	0.15	5702.68	5258.93

Wren line transect

Some errors:

```
# Fitting half-normal key function with Hermite(4,6) adjustments
# AIC= 1417.081
# Error in array(x, c(length(x), 1L), if (!is.null(names(x))) list(names(x),  :
#   'data' must be of a vector type, was 'NULL'
# Error in t(partial) %*% vcov :
#   requires numeric/complex matrix/vector arguments
# Fitting half-normal key function with Hermite(4,6) adjustments
# AIC= 1417.081
# Error in array(x, c(length(x), 1L), if (!is.null(names(x))) list(names(x),  :
#   'data' must be of a vector type, was 'NULL'
# Error in t(partial) %*% vcov :
#   requires numeric/complex matrix/vector arguments
```

```
data("wren_lt")

model.compare <- test.models(wren_lt,
                             truncation = max(wren_lt$distance, na.rm = TRUE),
                             transect = "line")

save(model.compare, file = "results/wren_lt.ROBJ")
```

Table 12: Comparison of R and MCDS model fits for wren_lt data. key - key function, adj - type of adjustment term, nadj - number of adjustments, lnl_R - likelihood value for R optimiser, lnl_MCDS - likelihood value for MCDS optimiser, optimizer - the selected optimiser, p_R the estimated average probability of detection for the R optimised model, p_MCDS the estimated average probability of detection for the MCDS optimised model, Nhat_R - estimated abundance in covered region from R optimised model, Nhat_MCDS - estimated abundance in covered region from MCDS optimised model.

key	adj	nadj	lnl_R	lnl_MCDS	optimizer	p_R	p_MCDS	Nhat_R	Nhat_MCDS
unif	cos	0	-708.0940	-708.0940	mrds (nlminb)	0.69	0.69	227.72	227.73
unif	cos	1	-706.8055	-706.8016	MCDS.exe	0.72	0.72	216.36	216.31
unif	cos	2	-705.8964	-705.8591	MCDS.exe	0.76	0.76	206.22	205.41
unif	cos	3	-705.3362	-705.3220	MCDS.exe	0.78	0.78	201.03	199.28
hn	cos	0	-708.0940	-708.0940	mrds (nlminb)	0.69	0.69	227.72	227.73
hn	cos	1	-706.8055	-706.8016	MCDS.exe	0.72	0.72	216.36	216.31
hn	cos	2	-705.8964	-705.8591	MCDS.exe	0.76	0.76	206.22	205.41
hn	herm	0	-708.0940	-708.0940	mrds (nlminb)	0.69	0.69	227.72	227.73
hn	herm	1	-706.1314	-706.1162	MCDS.exe	0.73	0.73	213.46	212.84
hn	herm	2	-706.0681	NA	NA	0.73	NA	212.99	NA
hr	poly	0	-708.0940	-708.0940	mrds (nlminb)	0.69	0.69	227.72	227.73
hr	poly	1	-705.3389	-705.3389	mrds (nlminb)	0.76	0.76	206.54	206.56
hr	poly	2	-705.3002	-704.8722	MCDS.exe	0.76	0.75	206.20	207.07

Wren snapshot

Some errors / warnings:

```
# Warning in check.mono(result, n.pts = control$mono.points) :
#   Detection function is less than 0 at some distances
# AIC= 2e+24
# Warning in mrds::check.mono(model, n.pts = 20) :
#   Detection function is less than 0 at some distances
# Some variance-covariance matrix elements were NA, possible numerical problems; only estimating detect
# ...
# Fitting half-normal key function with Hermite(4,6) adjustments
# AIC= 1069.234
# Error in array(x, c(length(x), 1L), if (!is.null(names(x))) list(names(x),  :
#   'data' must be of a vector type, was 'NULL'
# Error in t(partial) %*% vcov :
#   requires numeric/complex matrix/vector arguments
```

```
data("wren_snapshot")

model.compare <- test.models(wren_snapshot,
                             truncation = max(wren_snapshot$distance, na.rm = TRUE),
                             transect = "point")

save(model.compare, file = "results/wren_snapshot.ROBJ")
```

Table 13: Comparison of R and MCDS model fits for wren_snapshot data. key - key function, adj - type of adjustment term, nadj - number of adjustments, lnl_R - likelihood value for R optimiser, lnl_MCDS - likelihood value for MCDS optimiser, optimizer - the selected optimiser, p_R the estimated average probability of detection for the R optimised model, p_MCDS the estimated average probability of detection for the MCDS optimised model, Nhat_R - estimated abundance in covered region from R optimised model, Nhat_MCDS - estimated abundance in covered region from MCDS optimised model.

key	adj	nadj	lnl_R	lnl_MCDS	optimizer	p_R	p_MCDS	Nhat_R	Nhat_MCDS
unif	cos	0	-5.339206e+02	-533.9206	mrds (nlminb)	0.28	0.28	427.98	427.97
unif	cos	1	-5.304114e+02	-530.4001	MCDS.exe	0.33	0.33	353.86	353.57
unif	cos	2	-5.285742e+02	-528.5358	MCDS.exe	0.39	0.40	299.20	296.36
unif	cos	3	-5.279025e+02	-527.5610	MCDS.exe	0.36	0.36	329.14	327.97
hn	cos	0	-5.339206e+02	-533.9206	mrds (nlminb)	0.28	0.28	427.98	427.97
hn	cos	1	-5.304114e+02	-530.4001	MCDS.exe	0.33	0.33	353.86	353.57
hn	cos	2	-5.285742e+02	-528.5358	MCDS.exe	0.39	0.40	299.20	296.36
hn	herm	0	-5.339206e+02	-533.9206	mrds (nlminb)	0.28	0.28	427.98	427.97
hn	herm	1	-1.000000e+24	-532.7566	MCDS.exe	0.00	0.30	53499.62	393.40
hn	herm	2	-5.301191e+02	NA	mrds (nlminb)	0.34	NA	343.11	NA
hr	poly	0	-5.339206e+02	-533.9206	mrds (nlminb)	0.28	0.28	427.98	427.97
hr	poly	1	-5.324896e+02	-532.4896	MCDS.exe	0.31	0.31	386.28	386.29
hr	poly	2	-5.323749e+02	-529.5301	MCDS.exe	0.31	0.40	385.89	297.72

dathr1

```
load(file = "data/dathr1.RData")

model.compare <- test.models(dat,
                             truncation = 30,
                             transect = "point")

save(model.compare, file = "results/dathr1.ROBJ")
```

Table 14: Comparison of R and MCDS model fits for dathr1 data. key - key function, adj - type of adjustment term, nadj - number of adjustments, lnl_R - likelihood value for R optimiser, lnl_MCDS - likelihood value for MCDS optimiser, optimizer - the selected optimiser, p_R the estimated average probability of detection for the R optimised model, p_MCDS the estimated average probability of detection for the MCDS optimised model, Nhat_R - estimated abundance in covered region from R optimised model, Nhat_MCDS - estimated abundance in covered region from MCDS optimised model.

key	adj	nadj	lnl_R	lnl_MCDS	optimizer	p_R	p_MCDS	Nhat_R	Nhat_MCDS
unif	cos	0	-198.1089	-198.1089	mrds (nlminb)	0.31	0.31	190.77	190.76
unif	cos	1	-197.9999	-197.9999	mrds (nlminb)	0.28	0.28	215.86	215.80
unif	cos	2	-197.9895	-197.9895	mrds (nlminb)	0.27	0.27	225.83	225.75
unif	cos	3	-196.3096	-196.1618	MCDS.exe	0.16	0.16	368.54	385.80
hn	cos	0	-198.1089	-198.1089	mrds (nlminb)	0.31	0.31	190.77	190.76
hn	cos	1	-197.9999	-197.9999	mrds (nlminb)	0.28	0.28	215.86	215.80
hn	cos	2	-197.9895	-197.9895	mrds (nlminb)	0.27	0.27	225.83	225.75
hn	herm	0	-198.1089	-198.1089	mrds (nlminb)	0.31	0.31	190.77	190.76
hn	herm	1	-198.1012	-198.1012	mrds (nlminb)	0.31	0.31	191.06	191.06
hn	herm	2	-197.6629	-197.6522	MCDS.exe	0.29	0.29	208.12	206.06
hr	poly	0	-198.1089	-198.1089	mrds (nlminb)	0.31	0.31	190.77	190.76
hr	poly	1	-197.9458	-197.9361	MCDS.exe	0.30	0.29	202.88	206.83
hr	poly	2	-197.8401	-197.6433	MCDS.exe	0.29	0.30	203.84	198.10

dathr2

```
load(file = "data/dathr2.RData")

model.compare <- test.models(dat,
                             truncation = 30,
                             transect = "point")

save(model.compare, file = "results/dathr2.ROBJ")
```

Table 15: Comparison of R and MCDS model fits for dathr2 data. key - key function, adj - type of adjustment term, nadj - number of adjustments, lnl_R - likelihood value for R optimiser, lnl_MCDS - likelihood value for MCDS optimiser, optimizer - the selected optimiser, p_R the estimated average probability of detection for the R optimised model, p_MCDS the estimated average probability of detection for the MCDS optimised model, Nhat_R - estimated abundance in covered region from R optimised model, Nhat_MCDS - estimated abundance in covered region from MCDS optimised model.

key	adj	nadj	lnl_R	lnl_MCDS	optimizer	p_R	p_MCDS	Nhat_R	Nhat_MCDS
unif	cos	0	-198.1089	-198.1089	mrds (nlminb)	0.31	0.31	190.77	190.76
unif	cos	1	-197.9999	-197.9999	mrds (nlminb)	0.28	0.28	215.86	215.80
unif	cos	2	-197.9895	-197.9895	mrds (nlminb)	0.27	0.27	225.83	225.75
unif	cos	3	-196.3096	-196.1618	MCDS.exe	0.16	0.16	368.54	385.80
hn	cos	0	-198.1089	-198.1089	mrds (nlminb)	0.31	0.31	190.77	190.76
hn	cos	1	-197.9999	-197.9999	mrds (nlminb)	0.28	0.28	215.86	215.80
hn	cos	2	-197.9895	-197.9895	mrds (nlminb)	0.27	0.27	225.83	225.75
hn	herm	0	-198.1089	-198.1089	mrds (nlminb)	0.31	0.31	190.77	190.76
hn	herm	1	-198.1012	-198.1012	mrds (nlminb)	0.31	0.31	191.06	191.06
hn	herm	2	-197.6629	-197.6522	MCDS.exe	0.29	0.29	208.12	206.06
hr	poly	0	-198.1089	-198.1089	mrds (nlminb)	0.31	0.31	190.77	190.76
hr	poly	1	-197.9458	-197.9361	MCDS.exe	0.30	0.29	202.88	206.83
hr	poly	2	-197.8401	-197.6433	MCDS.exe	0.29	0.30	203.84	198.10

dathr3

Some errors / warnings:

```
# Fitting half-normal key function with Hermite(4,6) adjustments
# Error in -lt$value : invalid argument to unary operator
# In addition: Warning message:
# In system(paste0(path.to.MCDS.dot.exe, " 0, ", test.file$command.file.name),  :
#   running command 'C:/Users/lhm/AppData/Local/R/win-library/4.2/mrds/MCDS.exe 0, C:\Users\lhm\AppData
# Error in if (lt$message == "FALSE CONVERGENCE") { :
#   argument is of length zero
#
#
# All models failed to fit!
#
# Error in ds(dist.data, truncation = truncation, transect = transect, formula = ~1,  :
#   No models could be fitted.
# Fitting half-normal key function with Hermite(4,6) adjustments
# Error in -lt$value : invalid argument to unary operator
# In addition: Warning message:
# In system(paste0(path.to.MCDS.dot.exe, " 0, ", test.file$command.file.name),  :
#   running command 'C:/Users/lhm/AppData/Local/R/win-library/4.2/mrds/MCDS.exe 0, C:\Users\lhm\AppData
```

```
load(file = "data/dathr3.RData")

model.compare <- test.models(dat,
                             truncation = 30,
                             transect = "point")

save(model.compare, file = "results/dathr3.ROBJ")
```

Table 16: Comparison of R and MCDS model fits for dathr3 data. key - key function, adj - type of adjustment term, nadj - number of adjustments, lnl_R - likelihood value for R optimiser, lnl_MCDS - likelihood value for MCDS optimiser, optimizer - the selected optimiser, p_R the estimated average probability of detection for the R optimised model, p_MCDS the estimated average probability of detection for the MCDS optimised model, Nhat_R - estimated abundance in covered region from R optimised model, Nhat_MCDS - estimated abundance in covered region from MCDS optimised model.

key	adj	nadj	lnl_R	lnl_MCDS	optimizer	p_R	p_MCDS	Nhat_R	Nhat_MCDS
unif	cos	0	-198.4751	-198.4751	mrds (nlminb)	0.31	0.31	192.29	192.29
unif	cos	1	-197.8558	-197.8558	mrds (nlminb)	0.23	0.23	258.19	258.15
unif	cos	2	-197.8472	-197.8472	mrds (nlminb)	0.24	0.24	249.59	249.52
unif	cos	3	-197.8323	-197.8323	mrds (nlminb)	0.23	0.23	266.19	266.00
hn	cos	0	-198.4751	-198.4751	mrds (nlminb)	0.31	0.31	192.29	192.29
hn	cos	1	-197.8558	-197.8558	mrds (nlminb)	0.23	0.23	258.19	258.15
hn	cos	2	-197.8472	-197.8472	mrds (nlminb)	0.24	0.24	249.59	249.52
hn	herm	0	-198.4751	-198.4751	mrds (nlminb)	0.31	0.31	192.29	192.29
hn	herm	1	-198.4644	-198.4644	mrds (nlminb)	0.31	0.31	192.59	192.58
hn	herm	2	-197.7853	NA	mrds (nlminb)	0.25	NA	244.31	NA
hr	poly	0	-198.4751	-198.4751	mrds (nlminb)	0.31	0.31	192.29	192.29
hr	poly	1	-198.1751	-198.0700	MCDS.exe	0.29	0.28	204.41	215.30
hr	poly	2	-197.9932	-197.8039	MCDS.exe	0.28	0.25	213.18	238.67

dathr4

```
load(file = "data/dathr4.RData")

model.compare <- test.models(dat,
                             truncation = 30,
                             transect = "point")

save(model.compare, file = "results/dathr4.ROBJ")
```

Table 17: Comparison of R and MCDS model fits for dathr4 data. key - key function, adj - type of adjustment term, nadj - number of adjustments, lnl_R - likelihood value for R optimiser, lnl_MCDS - likelihood value for MCDS optimiser, optimizer - the selected optimiser, p_R the estimated average probability of detection for the R optimised model, p_MCDS the estimated average probability of detection for the MCDS optimised model, Nhat_R - estimated abundance in covered region from R optimised model, Nhat_MCDS - estimated abundance in covered region from MCDS optimised model.

key	adj	nadj	lnl_R	lnl_MCDS	optimizer	p_R	p_MCDS	Nhat_R	Nhat_MCDS
unif	cos	0	-195.4085	-195.4085	mrds (nlminb)	0.29	0.29	205.79	205.79
unif	cos	1	-195.4086	-195.4083	MCDS.exe	0.29	0.29	208.55	206.82
unif	cos	2	-195.2546	-195.2546	mrds (nlminb)	0.25	0.25	242.96	242.89
unif	cos	3	-195.2449	-195.2449	mrds (nlminb)	0.26	0.26	229.47	229.33
hn	cos	0	-195.4085	-195.4085	mrds (nlminb)	0.29	0.29	205.79	205.79
hn	cos	1	-195.4086	-195.4083	MCDS.exe	0.29	0.29	208.55	206.82
hn	cos	2	-195.2546	-195.2546	mrds (nlminb)	0.25	0.25	242.96	242.89
hn	herm	0	-195.4085	-195.4085	mrds (nlminb)	0.29	0.29	205.79	205.79
hn	herm	1	-195.3781	-195.3252	MCDS.exe	0.30	0.31	196.85	194.22
hn	herm	2	-195.3267	-195.3251	MCDS.exe	0.31	0.31	193.86	194.25
hr	poly	0	-195.4085	-195.4085	mrds (nlminb)	0.29	0.29	205.79	205.79
hr	poly	1	-195.3338	-195.3338	MCDS.exe	0.31	0.31	193.12	193.12
hr	poly	2	-195.3027	-195.2179	MCDS.exe	0.30	0.25	198.52	235.76

rocio_dat

```
load(file = "data/rocio_dat.RData")

model.compare <- test.models(dat,
                             truncation = 30,
                             transect = "point")

save(model.compare, file = "results/rocio_dat.ROBJ")
```

Table 18: Comparison of R and MCDS model fits for rocio_dat data. key - key function, adj - type of adjustment term, nadj - number of adjustments, lnl_R - likelihood value for R optimiser, lnl_MCDS - likelihood value for MCDS optimiser, optimizer - the selected optimiser, p_R the estimated average probability of detection for the R optimised model, p_MCDS the estimated average probability of detection for the MCDS optimised model, Nhat_R - estimated abundance in covered region from R optimised model, Nhat_MCDS - estimated abundance in covered region from MCDS optimised model.

key	adj	nadj	lnl_R	lnl_MCDS	optimizer	p_R	p_MCDS	Nhat_R	Nhat_MCDS
unif	cos	0	-1695.395	-1695.395	mrds (nlminb)	0.38	0.38	1318.76	1318.69
unif	cos	1	-1679.401	-1679.401	MCDS.exe	0.23	0.23	2196.20	2196.70
unif	cos	2	-1678.248	-1678.248	MCDS.exe	0.25	0.25	1964.17	1963.41
unif	cos	3	-1676.808	-1676.808	mrds (nlminb)	0.21	0.21	2362.77	2361.44
hn	cos	0	-1695.395	-1695.395	mrds (nlminb)	0.38	0.38	1318.76	1318.69
hn	cos	1	-1679.401	-1679.401	MCDS.exe	0.23	0.23	2196.20	2196.70
hn	cos	2	-1678.248	-1678.248	MCDS.exe	0.25	0.25	1964.17	1963.41
hn	herm	0	-1695.395	-1695.395	mrds (nlminb)	0.38	0.38	1318.76	1318.69
hn	herm	1	-1695.216	-1695.216	mrds (nlminb)	0.38	0.38	1321.14	1321.11
hn	herm	2	-1676.157	-1676.157	mrds (nlminb)	0.25	0.25	2014.21	2013.56
hr	poly	0	-1695.395	-1695.395	mrds (nlminb)	0.38	0.38	1318.76	1318.69
hr	poly	1	-1690.317	-1688.612	MCDS.exe	0.35	0.34	1412.32	1486.21
hr	poly	2	-1686.672	-1676.828	MCDS.exe	0.34	0.26	1481.87	1897.57

rocio_dat2

```
load(file = "data/rocio_dat_2.RData")

model.compare <- test.models(dat,
                             truncation = 30,
                             transect = "point")

save(model.compare, file = "results/rocio_dat2.ROBJ")
```

Table 19: Comparison of R and MCDS model fits for rocio_dat2 data. key - key function, adj - type of adjustment term, nadj - number of adjustments, lnl_R - likelihood value for R optimiser, lnl_MCDS - likelihood value for MCDS optimiser, optimizer - the selected optimiser, p_R the estimated average probability of detection for the R optimised model, p_MCDS the estimated average probability of detection for the MCDS optimised model, Nhat_R - estimated abundance in covered region from R optimised model, Nhat_MCDS - estimated abundance in covered region from MCDS optimised model.

key	adj	nadj	lnl_R	lnl_MCDS	optimizer	p_R	p_MCDS	Nhat_R	Nhat_MCDS
unif	cos	0	-127.7456	-127.7456	mrds (nlminb)	0.21	0.21	201.75	201.74
unif	cos	1	-125.6836	-125.6764	MCDS.exe	0.26	0.26	162.02	161.83
unif	cos	2	-122.3967	-122.0822	MCDS.exe	0.30	0.23	138.45	180.95
unif	cos	3	-122.2946	-120.8187	MCDS.exe	0.25	0.29	166.70	146.77
hn	cos	0	-127.7456	-127.7456	mrds (nlminb)	0.21	0.21	201.75	201.74
hn	cos	1	-125.6836	-125.6764	MCDS.exe	0.26	0.26	162.02	161.83
hn	cos	2	-122.3967	-122.0822	MCDS.exe	0.30	0.23	138.45	180.95
hn	herm	0	-127.7456	-127.7456	mrds (nlminb)	0.21	0.21	201.75	201.74
hn	herm	1	-125.4281	-121.2811	MCDS.exe	0.26	0.27	159.26	157.90
hn	herm	2	-125.3682	-121.1492	MCDS.exe	0.26	0.25	158.70	166.01
hr	poly	0	-127.7456	-127.7456	mrds (nlminb)	0.21	0.21	201.75	201.74
hr	poly	1	-126.3195	-119.2618	MCDS.exe	0.23	0.32	183.31	130.74
hr	poly	2	-185.5514	-118.3607	MCDS.exe	0.06	0.36	755.19	116.78

amakihi

There is a very big difference in estimated abundance for the first model here!

```
data("amakihi")

model.compare <- test.cov.models(amakihi,
                                 truncation = 82.5,
                                 transect = "point",
                                 models = list(~as.factor(OBs)+HAS+MAS, ~HAS))

save(model.compare, file = "results/amakihi.ROBJ")
```

Table 20: Comparison of R and MCDS model fits for amakihi data. key - key function, adj - type of adjustment term, nadj - number of adjustments, lnl_R - likelihood value for R optimiser, lnl_MCDS - likelihood value for MCDS optimiser, optimizer - the selected optimiser, p_R the estimated average probability of detection for the R optimised model, p_MCDS the estimated average probability of detection for the MCDS optimised model, Nhat_R - estimated abundance in covered region from R optimised model, Nhat_MCDS - estimated abundance in covered region from MCDS optimised model.

models	lnl_R	lnl_MCDS	optimizer	p_R	p_MCDS	Nhat_R	Nhat_MCDS
~as.factor(OBs) + HAS + MAS	-5396.835	-5479.755	mrds (nlminb)	0.34	0.25	3629.41	4999.25
~HAS	-5411.118	-5411.118	mrds (nlminb)	0.35	0.35	3559.08	3559.57

amakihi

```
load(file = "data/akepa.RData")

model.compare <- test.cov.models(dat,
                                 truncation = 58,
                                 transect = "line",
                                 models = list(~as.factor(Obs)))

save(model.compare, file = "results/akepa.ROBJ")
```

Table 21: Comparison of R and MCDS model fits for akepa data. key - key function, adj - type of adjustment term, nadj - number of adjustments, lnl_R - likelihood value for R optimiser, lnl_MCDS - likelihood value for MCDS optimiser, optimizer - the selected optimiser, p_R the estimated average probability of detection for the R optimised model, p_MCDS the estimated average probability of detection for the MCDS optimised model, Nhat_R - estimated abundance in covered region from R optimised model, Nhat_MCDS - estimated abundance in covered region from MCDS optimised model.

models	lnl_R	lnl_MCDS	optimizer	p_R	p_MCDS	Nhat_R	Nhat_MCDS
~as.factor(Obs)	-373.0925	-373.646	mrds (nlminb)	0.72	0.78	128.84	118.85