R-Script Hidden Markov Models

This is run with an R kernel instead of a python kernel in Jupyter Notebook.

More reading:

- https://cran.r-project.org/web/packages/moveHMM/vignettes/moveHMM-guide.pdf
- https://stackoverflow.com/questions/57870575/install-and-run-r-kernel-for-jupyter-notebook

The goal for this notebook is to take the prepped GPS data from the 01-GPS PreProcessing notebook and then to use that to fit multiple different HMM that represent the deer movement model. Different numbers of behavioural states, and different landscape rasters are used to fit the HMM.

The different models are tested using Akaike's Information Criteriont (AIC) and Bayesian Information Criterion (BIC) to see which models are "Best".

The models tested are:

- Simple 2-state HMM: No landscape information included. Only 2 behaviour states modelled
- 2-state Landscape HMM: 2 behaviour states, raster info as covariate
- 3-state Landscape HMM: 2 behaviour states, raster info as covariate
- 2-state Home-Return HMM: 2 Behaviour states, no raster info, turn angles replaced with angle to centroid value

NOTE: This whole notebook takes hours to run. This isn't a very fast lib.

```
Loading required package: CircStats

Loading required package: MASS

Loading required package: boot


Attaching package: ‘arrow’


The following object is masked from ‘package:utils’:

    timestamp
```

ID	Deer ID	timestamp	sex	lat	lon	time_group	geometry	utm eastin
<chr>	<dbl>	<dttm>	<chr>	<dbl>	<dbl>	<int>	<arrw_bnr>	<db
399253139925_gap_1	31	2017-02-02 00:00:12	f	39.22963	-76.88448	565	01, 01, 00, 00, 00, 0c, 38, f4, 1b, a0, c7, 38, 41, e7, da, 15, 9e, ef, f9, 3d, 41	337342
399253139925_gap_1	31	2017-02-02 01:00:28	f	39.23195	-76.88437	565	01, 01, 00, 00, 00, a4, ac, 86, 13, 75, c7, 38, 41, 36, 37, 8e, de, ef, fa, 3d, 41	337357

Take the GPS data and "prepare it", which calculates some variables and converts to data types required by the moveHMM package. Some different dataframes are created:

- **gps_data_simple**: This has deer ID's, step and turns, position info but not landscape raster info.
- **gps_data**: This has deer ID's, step and turns, position info and landscape raster info.
- **gps_data_home**: This has all the same fields as the "simple" dataframe, and also the distance and angles to the centroid

This takes a while to run... this isn't a multithreaded library.

The "prepData" command can take UTM or Lat/Lon position data, and the LLangle param can be TRUE, for great circle step and turn calculations, or FALSE for trigonometric calculations.

'ID' · 'step' · 'angle' · 'x' · 'y'

'ID' · 'step' · 'angle' · 'x' · 'y' · 'raster_value'

Step and Turn comparison

Let's double check that the step and turn calculated in the previous notebooks is the same, or pretty close, to the step and turns calculated by the MoveHMM library

A moveData: 6 × 5

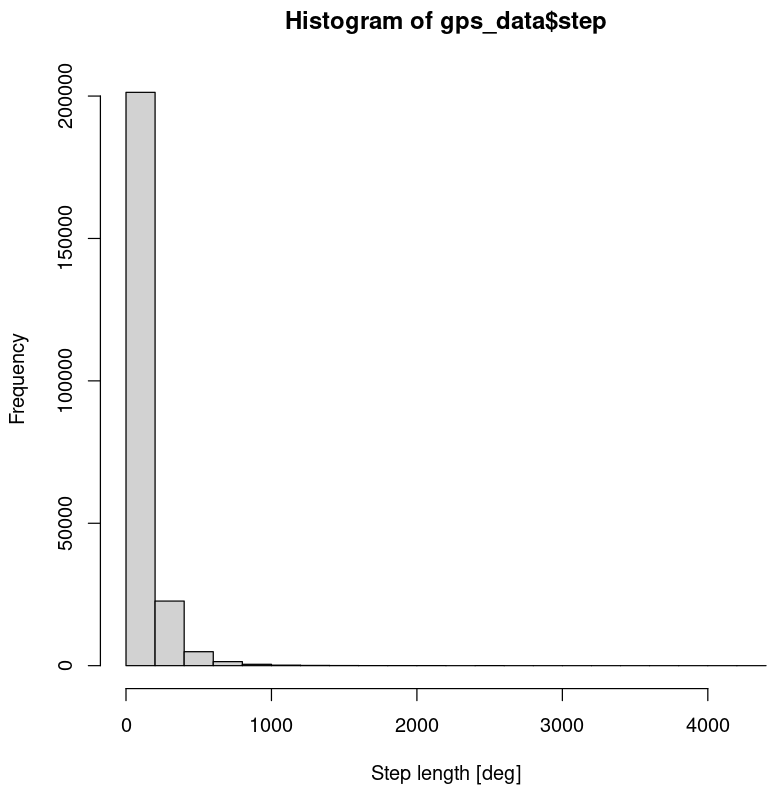
	ID	step	step_distance	angle	turn_angle
	<chr>	<dbl>	<dbl>	<dbl>	<dbl>
1	399253139925_gap_1	257.37552	259.84002	NA	-0.9321692
2	399253139925_gap_1	130.19103	129.57798	2.0515044	2.0603458
3	399253139925_gap_1	406.56052	408.93218	1.5184729	1.5027547
4	399253139925_gap_1	19.70712	19.60042	-2.5065172	-2.5066943
5	399253139925_gap_1	56.52751	56.32712	1.0567136	1.0731928
6	399253139925_gap_1	111.18051	110.40422	-0.2242944	-0.2267500

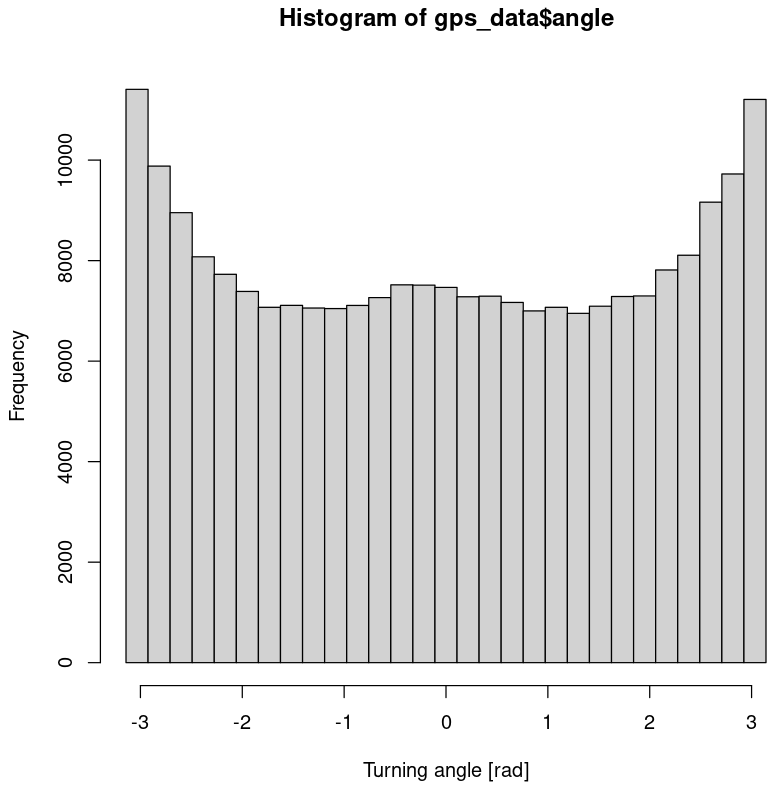
A moveData: 6 × 6

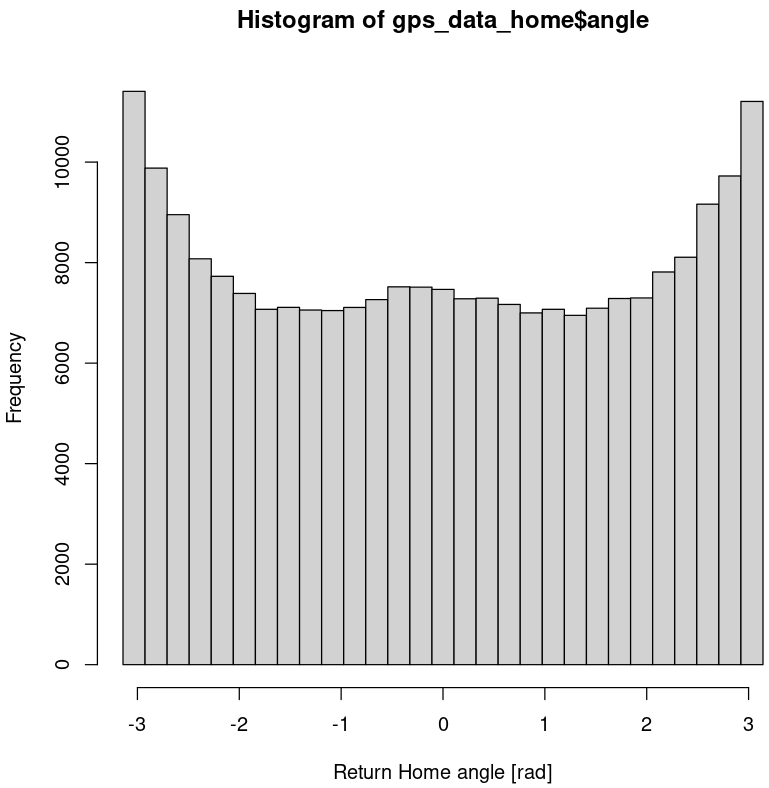
	ID	step	angle	x	y	raster_value
	<chr>	<dbl>	<dbl>	<dbl>	<dbl>	<int>
1	399253139925_gap_1	257.37552	NA	4343952	337342.8	95
2	399253139925_gap_1	130.19103	2.0515044	4344209	337357.2	53
3	399253139925_gap_1	406.56052	1.5184729	4344142	337469.1	87
4	399253139925_gap_1	19.70712	-2.5065172	4343783	337279.7	86
5	399253139925_gap_1	56.52751	1.0567136	4343791	337297.4	95
6	399253139925_gap_1	111.18051	-0.2242944	4343759	337343.9	65

A moveData: 6 × 7

	ID	x	y	step	angle	return_home_distance	raster_valu
	<chr>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<int
1	399253139925_gap_1	4343952	337342.8	257.37552	NA	230300.9	9
2	399253139925_gap_1	4344209	337357.2	130.19103	2.0515044	237949.5	5
3	399253139925_gap_1	4344142	337469.1	406.56052	1.5184729	236835.4	8
4	399253139925_gap_1	4343783	337279.7	19.70712	-2.5065172	224864.5	8
5	399253139925_gap_1	4343791	337297.4	56.52751	1.0567136	225248.0	9
6	399253139925_gap_1	4343759	337343.9	111.18051	-0.2242944	224655.5	6


Histogram of gps_data$step
Frequency
0
50000
100000
150000
200000
0
1000
2000
3000
4000
Step length [deg]


Histogram of gps_data$angle
Frequency
0
2000
4000
6000
8000
10000
-3
-2
-1
0
1
2
3
Turning angle [rad]



Some Different HMM Models:

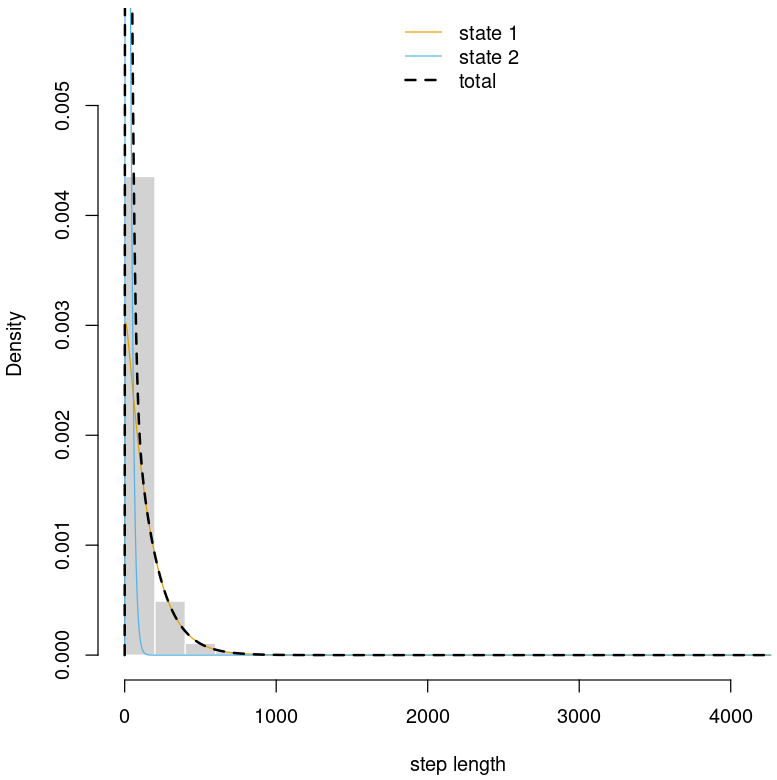
Simple 2 State Model

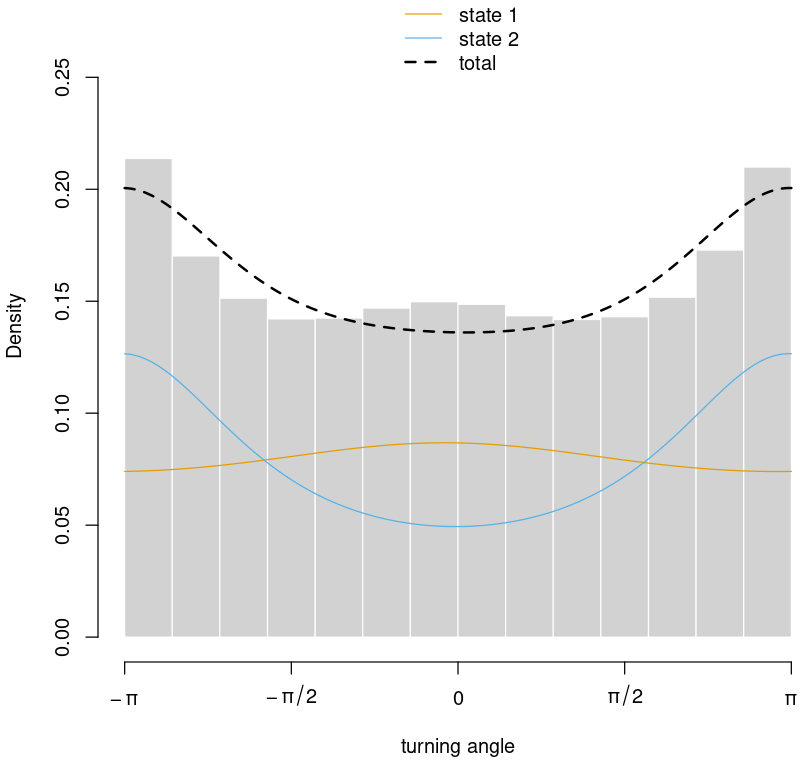
No covariates, 2 states.

When fitting a new model the initial parameters are VERY important. Values not close to the end results may prevent the model from converging or settling on a local minima.

```
Warning message in rbind(parts$upper, chars$ellip_v, parts$lower, deparse.level = 0
L):
“number of columns of result is not a multiple of vector length (arg 2)”
Warning message in rbind(parts$upper, chars$ellip_v, parts$lower, deparse.level = 0
L):
“number of columns of result is not a multiple of vector length (arg 2)”
Warning message in rbind(parts$upper, chars$ellip_v, parts$lower, deparse.level = 0
L):
“number of columns of result is not a multiple of vector length (arg 2)”
```

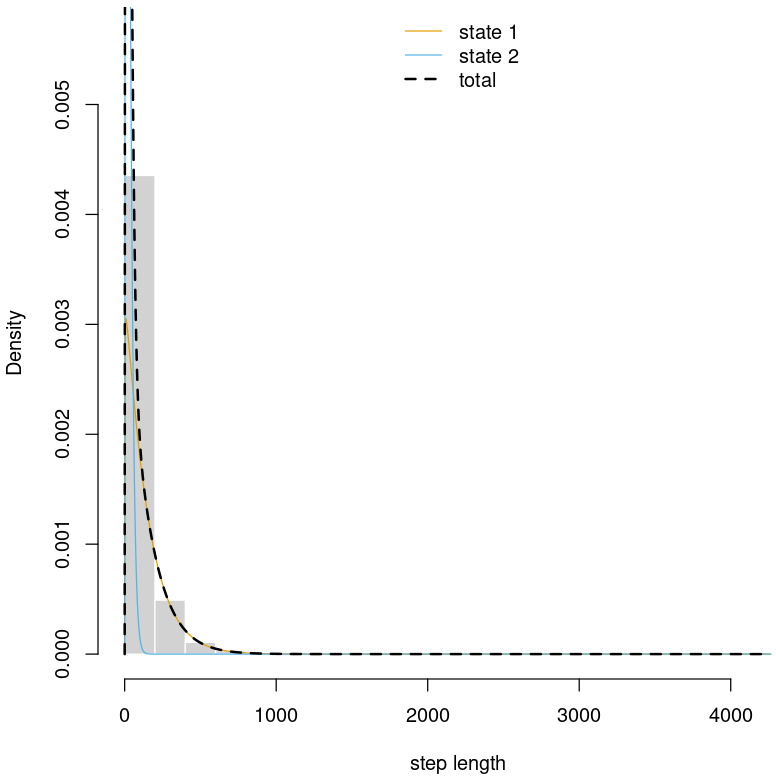
```
Value of the maximum log-likelihood: -1671793

Step length parameters:
-----------------------
                state 1      state 2
shape     1.041134e+00 1.357803e+00
scale     1.465244e+02 2.900054e+01
zero-mass 1.239168e-06 6.682292e-05

Turning angle parameters:
-------------------------
                    state 1   state 2
mean          -0.12704320 3.1176960
concentration  0.03997021 0.2311387

Regression coeffs for the transition probabilities:
---------------------------------------------------
             1 -> 2    2 -> 1
intercept -1.138212 -0.829216

Transition probability matrix:
------------------------------
          [,1]      [,2]
[1,] 0.7573513 0.2426487
[2,] 0.3038109 0.6961891

Initial distribution:
---------------------
[1] 0.5677461 0.4322539

Decoding states sequence... DONE
```

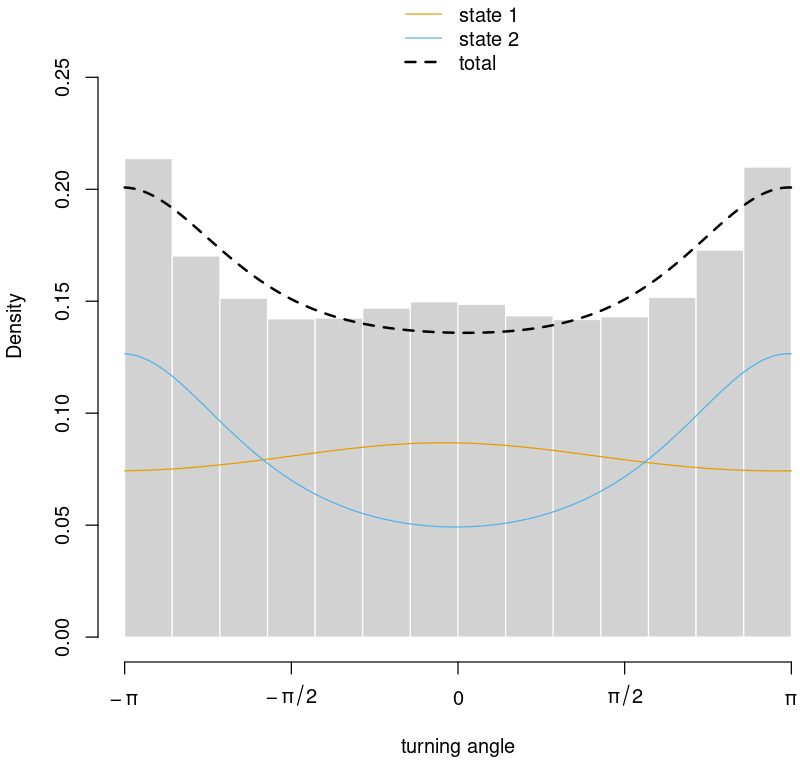


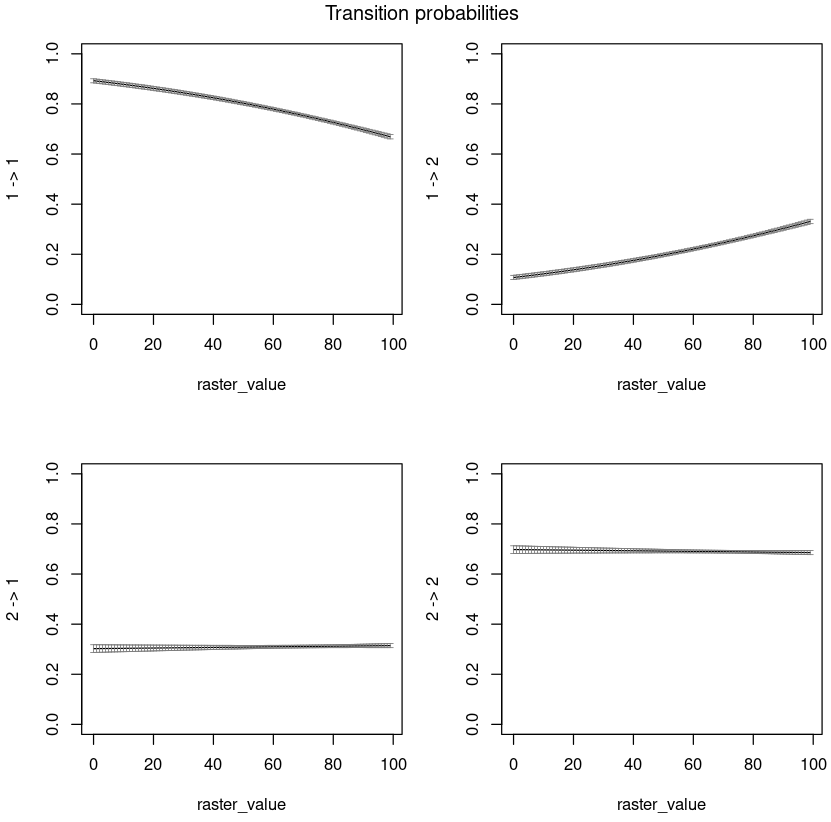


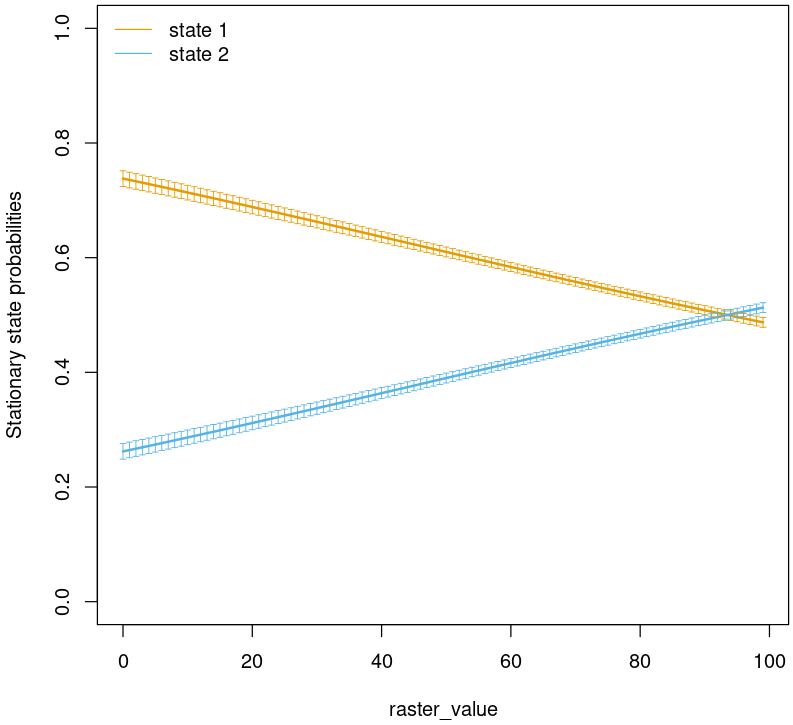
2 State Model, Landscape included

```
Decoding states sequence... DONE
```




state 1
state 2
total
Density
0.00
0.05
0.10
0.15
0.20
0.25
−π
−π/2
0
π/2
π
turning angle


Transition probabilities
1 -> 1
1 -> 2
2 -> 1
2 -> 2
0.0
0.2
0.4
0.6
0.8
1.0
0
20
40
60
80
100
raster_value




```
Value of the maximum log-likelihood: -1671251

Step length parameters:
-----------------------
               state 1      state 2
shape     1.038574e+00 1.370233e+00
scale     1.458696e+02 2.882389e+01
zero-mass 2.170312e-05 4.096078e-05

Turning angle parameters:
-------------------------
                  state 1   state 2
mean          -0.13072763 3.1172712
concentration  0.03912245 0.2319933

Regression coeffs for the transition probabilities:
---------------------------------------------------
                  1 -> 2        2 -> 1
intercept    -2.11832409 -0.8364456422
raster_value  0.01428307  0.0005707251

Initial distribution:
---------------------
[1] 0.57357 0.42643
```

HMM numbers to stick into Repast Model

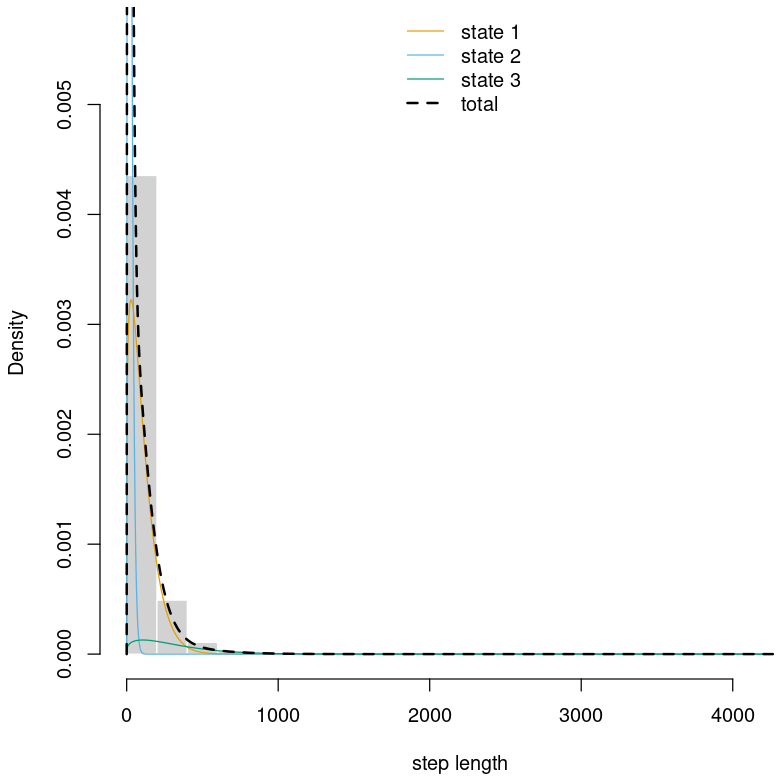
The HMM numbers that are needed for the Repast model. This gets stuck into the python model:

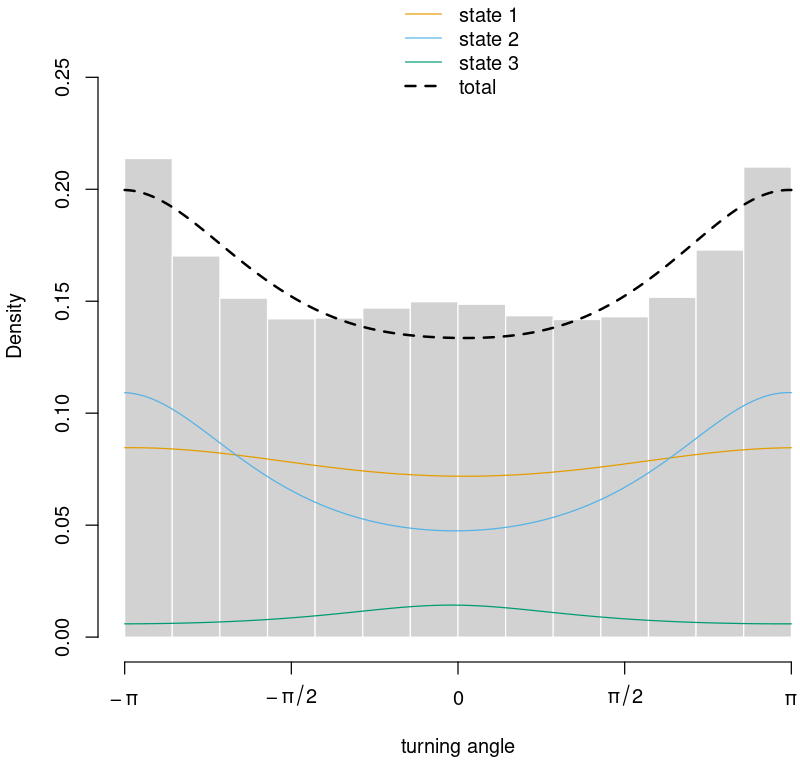
```python
class HMM_MoveModel_2_States(BaseMoveModel):
    '''
    Simple random movement model. Agents move around with a weibull/cauchy step and turn model.
    Not influenced by environmental, time or behaviour states. Just a pure random walk.
    '''
    def __init__(self, *args, **kwargs):
        self.movement_n_states = 2
        self.movement_params = [{'state': 0,
                                 'state_name': 'HMM State 0',
                                 'step_params':{'c': 1.38666,
                                                'loc': 1,
                                                'scale': 29.142},
                                 'turn_params':{'c': 0.23,
                                                'loc': -3.123,
                                                'scale': 1}},
                                {'state': 1,
                                 'state_name': 'HMM State 1',
                                 'step_params':{'c': 1.0378,
                                                'loc': 1,
                                                'scale': 148},
                                 'turn_params':{'c': 0.04,
                                                'loc': 0.1,
                                                'scale': 1}}
                                ] # Params for each different move state. List of N dictiona

        assert len(self.movement_params) == self.movement_n_states, "Too few movement params for number of m

        # HMM Covariate Intercept matrix
        self.hmm_covariate_intercept = np.asarray([-8.359713e-01, -2.08067466])
        # HMM Covariate Raster Coeff matrix
        self.hmm_covariate_coeff = np.asarray([5.790579e-05, 0.01360621])
```

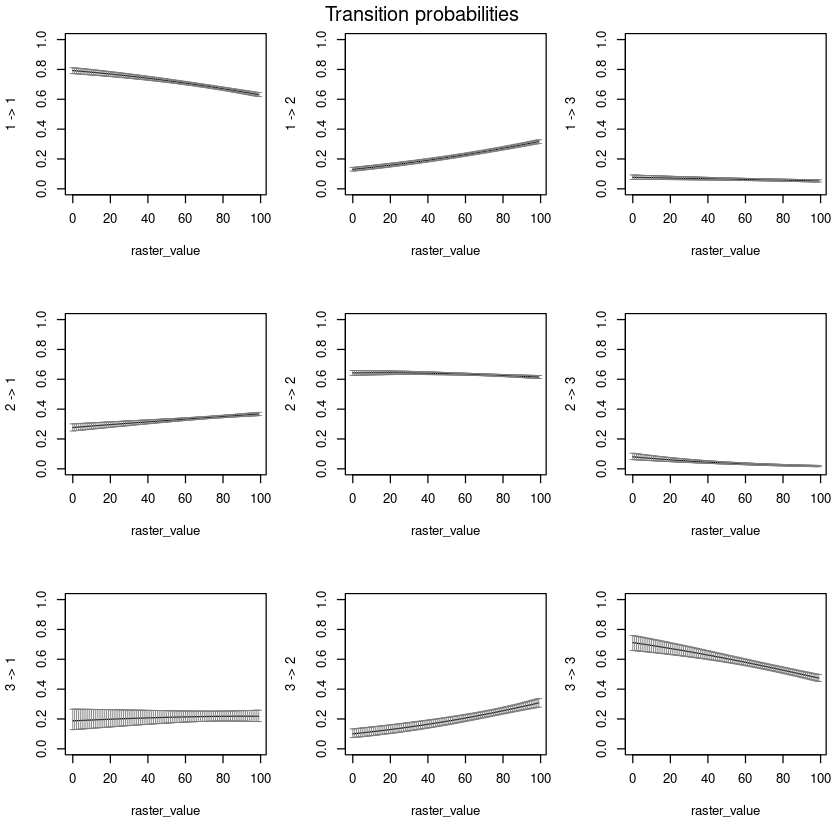
```
Value of the maximum log-likelihood: -1671251

Step length parameters:
-----------------------
                state 1      state 2
shape     1.038574e+00 1.370233e+00
scale     1.458696e+02 2.882389e+01
zero-mass 2.170312e-05 4.096078e-05

Turning angle parameters:
-------------------------
                   state 1   state 2
mean          -0.13072763 3.1172712
concentration  0.03912245 0.2319933

Regression coeffs for the transition probabilities:
---------------------------------------------------
                  1 -> 2        2 -> 1
intercept    -2.11832409 -0.8364456422
raster_value  0.01428307  0.0005707251

Initial distribution:
---------------------
[1] 0.57357 0.42643
```

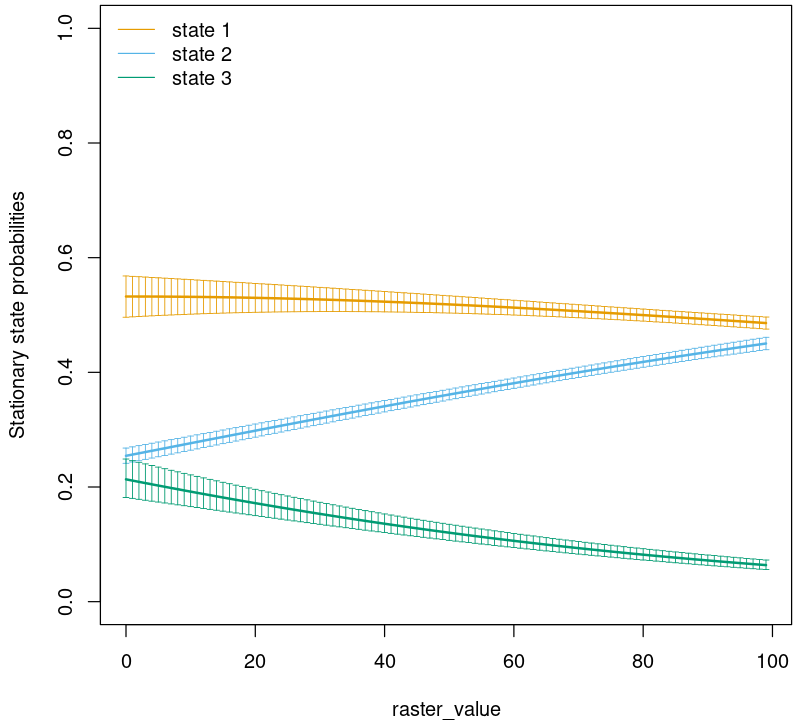
3 State Model, Landscape included

```
Value of the maximum log-likelihood: -1666956

Step length parameters:
-----------------------
               state 1      state 2      state 3
shape     1.228546e+00 1.476773e+00 1.297833e+00
scale     1.133292e+02 2.405681e+01 3.318213e+02
zero-mass 1.004137e-08 7.622952e-05 7.083518e-15

Turning angle parameters:
-------------------------
                  state 1   state 2    state 3
mean          -3.08108420 3.1140039 -0.0631689
concentration  0.04079163 0.2053319  0.2165741

Regression coeffs for the transition probabilities:
---------------------------------------------------
                  1 -> 2       1 -> 3       2 -> 1      2 -> 3       3 -> 1
intercept    -1.80329097 -2.332902740 -0.842190949 -2.07295724 -1.333573276
raster_value  0.01124452 -0.001542683  0.003304142 -0.01484277  0.005627201
                  3 -> 2
intercept    -1.95697222
raster_value  0.01540365

Initial distribution:
---------------------
[1] 0.4933514 0.3867857 0.1198629

Decoding states sequence... DONE
```




state 1
state 2
state 3
total
Density
0.00
0.05
0.10
0.15
0.20
0.25
−π
−π/2
0
π/2
π
turning angle


Transition probabilities
1 -> 1
1 -> 2
1 -> 3
2 -> 1
2 -> 2
2 -> 3
3 -> 1
3 -> 2
3 -> 3
0.0
0.2
0.4
0.6
0.8
1.0
0
20
40
60
80
100
raster_value



$stepPar

A matrix: 3 × 3 of type dbl

	state 1	state 2	state 3
shape	1.228546e+00	1.476773e+00	1.297833e+00
scale	1.133292e+02	2.405681e+01	3.318213e+02
zero-mass	1.004137e-08	7.622952e-05	7.083518e-15

$anglePar

A matrix: 2 × 3 of type dbl

	state 1	state 2	state 3
mean	-3.08108420	3.1140039	-0.0631689
concentration	0.04079163	0.2053319	0.2165741

$beta

A matrix: 2 × 6 of type dbl

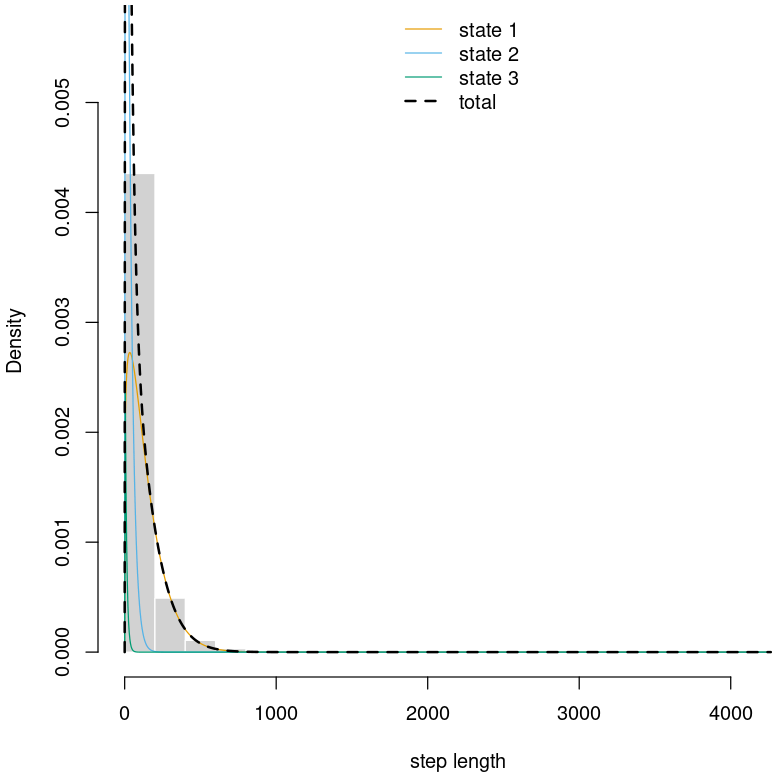
	1 -> 2	1 -> 3	2 -> 1	2 -> 3	3 ->
intercept	-1.80329097	-2.332902740	-0.842190949	-2.07295724	-1.3335732
raster_value	0.01124452	-0.001542683	0.003304142	-0.01484277	0.0056272

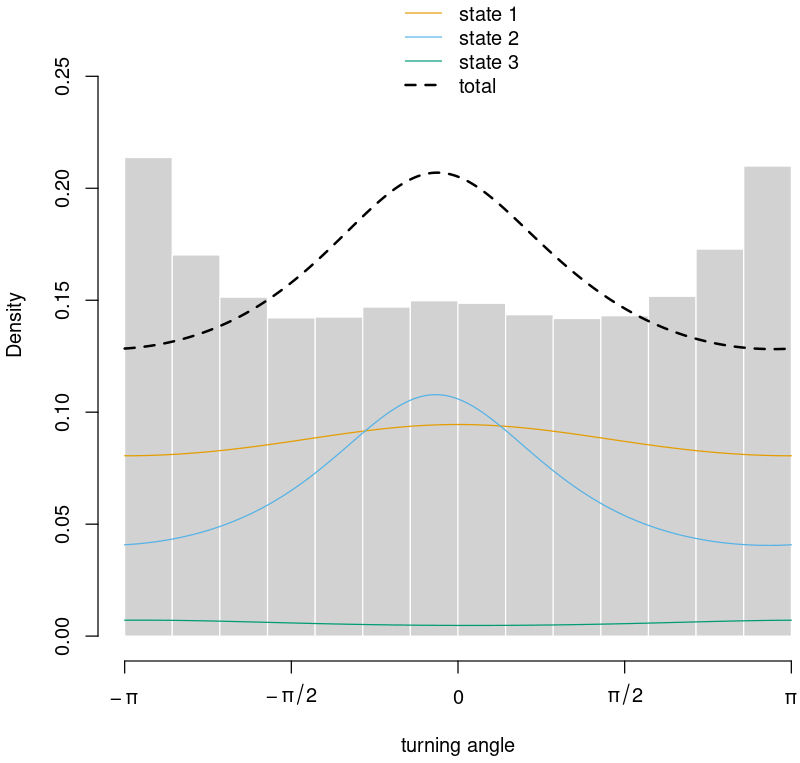
$delta

0.493351376918209 · 0.386785739343407 · 0.119862883738384

3-States Return Home Model

```
Value of the maximum log-likelihood: -1693788

Step length parameters:
-----------------------
               state 1      state 2      state 3
shape     1.199954e+00  1.100021218  1.000004954
scale     1.500012e+02 30.000096939 10.000060157
zero-mass 9.999986e-03  0.009999991  0.009999998

Turning angle parameters:
-------------------------
                    state 1    state 2   state 3
mean          4.215788e-05 -0.2100128 -3.000021
concentration 3.997471e-02  0.2399684  0.100006

Regression coeffs for the transition probabilities:
---------------------------------------------------
                            1 -> 2        1 -> 3        2 -> 1        2 -> 3
intercept             -1.499999e+00 -1.500000e+00 -1.499999e+00 -1.500000e+00
raster_value           9.321153e-05 -1.319901e-05  8.017867e-05 -3.413783e-05
return_home_distance   4.707659e-06 -2.871356e-04  1.493262e-05 -2.486314e-05
                            3 -> 1        3 -> 2
intercept             -1.500000e+00 -1.500000e+00
raster_value           2.574714e-05  6.771150e-06
return_home_distance   3.381874e-04  1.998063e-04

Initial distribution:
---------------------
[1] 0.3333334 0.3333334 0.3333333

Decoding states sequence... DONE
```


state 1
state 2
state 3
total
Density
0.000
0.001
0.002
0.003
0.004
0.005
0
1000
2000
3000
4000
step length

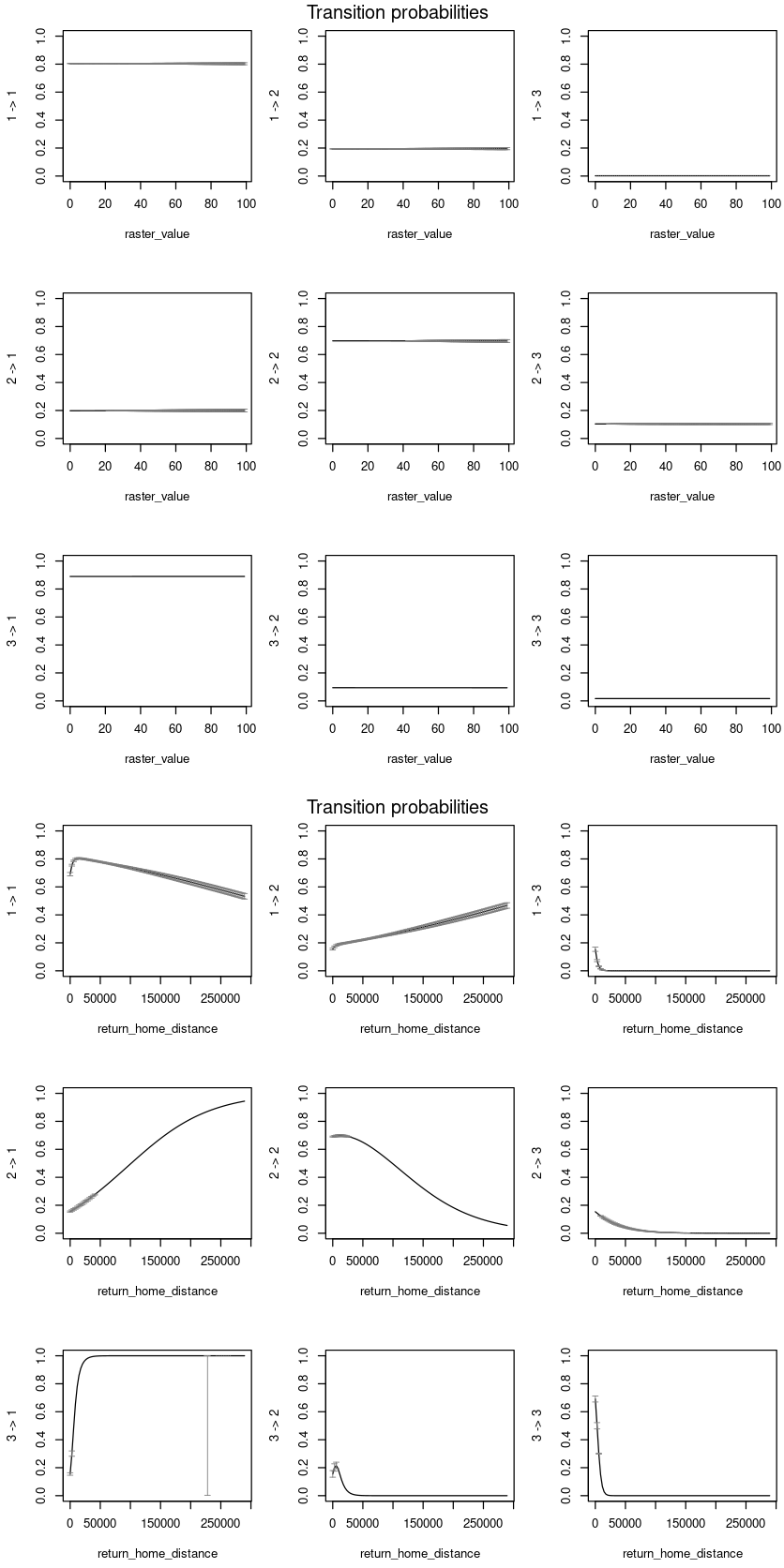

state 1
state 2
state 3
total
Density
0.00
0.05
0.10
0.15
0.20
0.25
−π
−π/2
0
π/2
π
turning angle

```
Warning message in doTryCatch(return(expr), name, parentenv, handler):
"zero-length arrow is of indeterminate angle and so skipped"
Warning message in doTryCatch(return(expr), name, parentenv, handler):
"zero-length arrow is of indeterminate angle and so skipped"
Warning message in doTryCatch(return(expr), name, parentenv, handler):
"zero-length arrow is of indeterminate angle and so skipped"
Warning message in doTryCatch(return(expr), name, parentenv, handler):
"zero-length arrow is of indeterminate angle and so skipped"
Warning message in doTryCatch(return(expr), name, parentenv, handler):
"zero-length arrow is of indeterminate angle and so skipped"
Warning message in doTryCatch(return(expr), name, parentenv, handler):
"zero-length arrow is of indeterminate angle and so skipped"
Warning message in doTryCatch(return(expr), name, parentenv, handler):
"zero-length arrow is of indeterminate angle and so skipped"
Warning message in doTryCatch(return(expr), name, parentenv, handler):
"zero-length arrow is of indeterminate angle and so skipped"
Warning message in doTryCatch(return(expr), name, parentenv, handler):
"zero-length arrow is of indeterminate angle and so skipped"
Warning message in doTryCatch(return(expr), name, parentenv, handler):
"zero-length arrow is of indeterminate angle and so skipped"
Warning message in doTryCatch(return(expr), name, parentenv, handler):
"zero-length arrow is of indeterminate angle and so skipped"
Warning message in doTryCatch(return(expr), name, parentenv, handler):
"zero-length arrow is of indeterminate angle and so skipped"
Warning message in doTryCatch(return(expr), name, parentenv, handler):
"zero-length arrow is of indeterminate angle and so skipped"
Warning message in doTryCatch(return(expr), name, parentenv, handler):
"zero-length arrow is of indeterminate angle and so skipped"
Warning message in doTryCatch(return(expr), name, parentenv, handler):
"zero-length arrow is of indeterminate angle and so skipped"
Warning message in doTryCatch(return(expr), name, parentenv, handler):
"zero-length arrow is of indeterminate angle and so skipped"
Warning message in doTryCatch(return(expr), name, parentenv, handler):
"zero-length arrow is of indeterminate angle and so skipped"
Warning message in doTryCatch(return(expr), name, parentenv, handler):
"zero-length arrow is of indeterminate angle and so skipped"
Warning message in doTryCatch(return(expr), name, parentenv, handler):
"zero-length arrow is of indeterminate angle and so skipped"
Warning message in doTryCatch(return(expr), name, parentenv, handler):
"zero-length arrow is of indeterminate angle and so skipped"
Warning message in doTryCatch(return(expr), name, parentenv, handler):
"zero-length arrow is of indeterminate angle and so skipped"
Warning message in doTryCatch(return(expr), name, parentenv, handler):
"zero-length arrow is of indeterminate angle and so skipped"
Warning message in doTryCatch(return(expr), name, parentenv, handler):
"zero-length arrow is of indeterminate angle and so skipped"
Warning message in doTryCatch(return(expr), name, parentenv, handler):
"zero-length arrow is of indeterminate angle and so skipped"
Warning message in doTryCatch(return(expr), name, parentenv, handler):
"zero-length arrow is of indeterminate angle and so skipped"
Warning message in doTryCatch(return(expr), name, parentenv, handler):
"zero-length arrow is of indeterminate angle and so skipped"
Warning message in doTryCatch(return(expr), name, parentenv, handler):
"zero-length arrow is of indeterminate angle and so skipped"
Warning message in doTryCatch(return(expr), name, parentenv, handler):
"zero-length arrow is of indeterminate angle and so skipped"
```

```
Warning message in doTryCatch(return(expr), name, parentenv, handler):
“zero-length arrow is of indeterminate angle and so skipped”
Warning message in doTryCatch(return(expr), name, parentenv, handler):
“zero-length arrow is of indeterminate angle and so skipped”
Warning message in doTryCatch(return(expr), name, parentenv, handler):
“zero-length arrow is of indeterminate angle and so skipped”
Warning message in doTryCatch(return(expr), name, parentenv, handler):
“zero-length arrow is of indeterminate angle and so skipped”
Warning message in doTryCatch(return(expr), name, parentenv, handler):
“zero-length arrow is of indeterminate angle and so skipped”
Warning message in doTryCatch(return(expr), name, parentenv, handler):
“zero-length arrow is of indeterminate angle and so skipped”
Warning message in doTryCatch(return(expr), name, parentenv, handler):
“zero-length arrow is of indeterminate angle and so skipped”
Warning message in doTryCatch(return(expr), name, parentenv, handler):
“zero-length arrow is of indeterminate angle and so skipped”
Warning message in doTryCatch(return(expr), name, parentenv, handler):
“zero-length arrow is of indeterminate angle and so skipped”
Warning message in doTryCatch(return(expr), name, parentenv, handler):
“zero-length arrow is of indeterminate angle and so skipped”
Warning message in doTryCatch(return(expr), name, parentenv, handler):
“zero-length arrow is of indeterminate angle and so skipped”
Warning message in doTryCatch(return(expr), name, parentenv, handler):
“zero-length arrow is of indeterminate angle and so skipped”
Warning message in doTryCatch(return(expr), name, parentenv, handler):
“zero-length arrow is of indeterminate angle and so skipped”
Warning message in doTryCatch(return(expr), name, parentenv, handler):
“zero-length arrow is of indeterminate angle and so skipped”
Warning message in doTryCatch(return(expr), name, parentenv, handler):
“zero-length arrow is of indeterminate angle and so skipped”
Warning message in doTryCatch(return(expr), name, parentenv, handler):
“zero-length arrow is of indeterminate angle and so skipped”
Warning message in doTryCatch(return(expr), name, parentenv, handler):
“zero-length arrow is of indeterminate angle and so skipped”
Warning message in doTryCatch(return(expr), name, parentenv, handler):
“zero-length arrow is of indeterminate angle and so skipped”
Warning message in doTryCatch(return(expr), name, parentenv, handler):
“zero-length arrow is of indeterminate angle and so skipped”
Warning message in doTryCatch(return(expr), name, parentenv, handler):
“zero-length arrow is of indeterminate angle and so skipped”
Warning message in doTryCatch(return(expr), name, parentenv, handler):
“zero-length arrow is of indeterminate angle and so skipped”
Warning message in doTryCatch(return(expr), name, parentenv, handler):
“zero-length arrow is of indeterminate angle and so skipped”
Warning message in doTryCatch(return(expr), name, parentenv, handler):
“zero-length arrow is of indeterminate angle and so skipped”
Warning message in doTryCatch(return(expr), name, parentenv, handler):
“zero-length arrow is of indeterminate angle and so skipped”
Warning message in doTryCatch(return(expr), name, parentenv, handler):
“zero-length arrow is of indeterminate angle and so skipped”
Warning message in doTryCatch(return(expr), name, parentenv, handler):
“zero-length arrow is of indeterminate angle and so skipped”
Warning message in doTryCatch(return(expr), name, parentenv, handler):
“zero-length arrow is of indeterminate angle and so skipped”
Warning message in doTryCatch(return(expr), name, parentenv, handler):
“zero-length arrow is of indeterminate angle and so skipped”
```

```
Warning message in doTryCatch(return(expr), name, parentenv, handler):
“zero-length arrow is of indeterminate angle and so skipped”
Warning message in doTryCatch(return(expr), name, parentenv, handler):
“zero-length arrow is of indeterminate angle and so skipped”
Warning message in doTryCatch(return(expr), name, parentenv, handler):
“zero-length arrow is of indeterminate angle and so skipped”
Warning message in doTryCatch(return(expr), name, parentenv, handler):
“zero-length arrow is of indeterminate angle and so skipped”
Warning message in doTryCatch(return(expr), name, parentenv, handler):
“zero-length arrow is of indeterminate angle and so skipped”
Warning message in doTryCatch(return(expr), name, parentenv, handler):
“zero-length arrow is of indeterminate angle and so skipped”
Warning message in doTryCatch(return(expr), name, parentenv, handler):
“zero-length arrow is of indeterminate angle and so skipped”
Warning message in doTryCatch(return(expr), name, parentenv, handler):
“zero-length arrow is of indeterminate angle and so skipped”
Warning message in doTryCatch(return(expr), name, parentenv, handler):
“zero-length arrow is of indeterminate angle and so skipped”
Warning message in doTryCatch(return(expr), name, parentenv, handler):
“zero-length arrow is of indeterminate angle and so skipped”
Warning message in doTryCatch(return(expr), name, parentenv, handler):
“zero-length arrow is of indeterminate angle and so skipped”
Warning message in doTryCatch(return(expr), name, parentenv, handler):
“zero-length arrow is of indeterminate angle and so skipped”
Warning message in doTryCatch(return(expr), name, parentenv, handler):
“zero-length arrow is of indeterminate angle and so skipped”
Warning message in doTryCatch(return(expr), name, parentenv, handler):
“zero-length arrow is of indeterminate angle and so skipped”
Warning message in doTryCatch(return(expr), name, parentenv, handler):
“zero-length arrow is of indeterminate angle and so skipped”
Warning message in doTryCatch(return(expr), name, parentenv, handler):
“zero-length arrow is of indeterminate angle and so skipped”
Warning message in doTryCatch(return(expr), name, parentenv, handler):
“zero-length arrow is of indeterminate angle and so skipped”
Warning message in doTryCatch(return(expr), name, parentenv, handler):
“zero-length arrow is of indeterminate angle and so skipped”
Warning message in doTryCatch(return(expr), name, parentenv, handler):
“zero-length arrow is of indeterminate angle and so skipped”
Warning message in doTryCatch(return(expr), name, parentenv, handler):
“zero-length arrow is of indeterminate angle and so skipped”
Warning message in doTryCatch(return(expr), name, parentenv, handler):
“zero-length arrow is of indeterminate angle and so skipped”
Warning message in doTryCatch(return(expr), name, parentenv, handler):
“zero-length arrow is of indeterminate angle and so skipped”
Warning message in doTryCatch(return(expr), name, parentenv, handler):
“zero-length arrow is of indeterminate angle and so skipped”
Warning message in doTryCatch(return(expr), name, parentenv, handler):
“zero-length arrow is of indeterminate angle and so skipped”
Warning message in doTryCatch(return(expr), name, parentenv, handler):
“zero-length arrow is of indeterminate angle and so skipped”
Warning message in doTryCatch(return(expr), name, parentenv, handler):
“zero-length arrow is of indeterminate angle and so skipped”
Warning message in doTryCatch(return(expr), name, parentenv, handler):
“zero-length arrow is of indeterminate angle and so skipped”
Warning message in doTryCatch(return(expr), name, parentenv, handler):
“zero-length arrow is of indeterminate angle and so skipped”
```

```
Warning message in doTryCatch(return(expr), name, parentenv, handler):
“zero-length arrow is of indeterminate angle and so skipped”
Warning message in doTryCatch(return(expr), name, parentenv, handler):
“zero-length arrow is of indeterminate angle and so skipped”
Warning message in doTryCatch(return(expr), name, parentenv, handler):
“zero-length arrow is of indeterminate angle and so skipped”
Warning message in doTryCatch(return(expr), name, parentenv, handler):
“zero-length arrow is of indeterminate angle and so skipped”
Warning message in doTryCatch(return(expr), name, parentenv, handler):
“zero-length arrow is of indeterminate angle and so skipped”
Warning message in doTryCatch(return(expr), name, parentenv, handler):
“zero-length arrow is of indeterminate angle and so skipped”
Warning message in doTryCatch(return(expr), name, parentenv, handler):
“zero-length arrow is of indeterminate angle and so skipped”
Warning message in doTryCatch(return(expr), name, parentenv, handler):
“zero-length arrow is of indeterminate angle and so skipped”
Warning message in doTryCatch(return(expr), name, parentenv, handler):
“zero-length arrow is of indeterminate angle and so skipped”
Warning message in doTryCatch(return(expr), name, parentenv, handler):
“zero-length arrow is of indeterminate angle and so skipped”
Warning message in doTryCatch(return(expr), name, parentenv, handler):
“zero-length arrow is of indeterminate angle and so skipped”
Warning message in doTryCatch(return(expr), name, parentenv, handler):
“zero-length arrow is of indeterminate angle and so skipped”
Warning message in doTryCatch(return(expr), name, parentenv, handler):
“zero-length arrow is of indeterminate angle and so skipped”
Warning message in doTryCatch(return(expr), name, parentenv, handler):
“zero-length arrow is of indeterminate angle and so skipped”
Warning message in doTryCatch(return(expr), name, parentenv, handler):
“zero-length arrow is of indeterminate angle and so skipped”
Warning message in doTryCatch(return(expr), name, parentenv, handler):
“zero-length arrow is of indeterminate angle and so skipped”
Warning message in doTryCatch(return(expr), name, parentenv, handler):
“zero-length arrow is of indeterminate angle and so skipped”
```



But which is best?

Use the AIC values to decide which model is best.

A data.frame: 4 × 2

Model	AIC
<chr>	<dbl>
mod_3states	3333970
mod_simple_landscape	3342531
model_simple	3343613
mod_3states_rh	3387646

Looks like the 3 state model performs the best, while the "return home" model is the worst.

```
Error in parse(text = input): <text>:2:11: unexpected symbol
1: ## Export this notebook to HTML and then to pdf
2: ! jupyter nbconvert
              ^
Traceback:
```