

# A multistate habitat-driven Langevin diffusion for inferring behaviour-specific utilization distributions

Brett T. McClintock and Michelle Lander

NOAA Marine Mammal Lab, Seattle, USA

25 June 2022

# Langevin diffusion

$$d\boldsymbol{\mu}_t = \frac{\sigma^2}{2} \nabla \log \pi(\boldsymbol{\mu}_t | \boldsymbol{\beta}) dt + \sigma d\mathbf{B}_t$$

$$\pi(\boldsymbol{\mu} | \boldsymbol{\beta}) = \frac{\exp\left(\sum_{k=1}^K x_k(\boldsymbol{\mu}) \beta_k\right)}{\int_{\mathcal{M}} \exp\left(\sum_{k=1}^K x_k(\mathbf{z}) \beta_k\right) d\mathbf{z}}$$



Michelot et al. (2019). The Langevin diffusion as a continuous-time model of animal movement and habitat selection, *Methods Ecol Evol.*

## Langevin diffusion (discrete approximation of)

$$\boldsymbol{\mu}_{t+1} = \boldsymbol{\mu}_t + \frac{\sigma^2 \Delta_t}{2} \nabla \log \pi(\boldsymbol{\mu}_t | \boldsymbol{\beta}) + \boldsymbol{\epsilon}_{t+1}$$

$$\nabla \log \pi(\boldsymbol{\mu} | \boldsymbol{\beta}) = \sum_{k=1}^K \beta_k \nabla x_k(\boldsymbol{\mu})$$

$\boldsymbol{\mu}_t$  = location at time  $t$

$\sigma^2$  = speed parameter

$\boldsymbol{\epsilon}_{t+1} \sim \mathcal{N}(\mathbf{0}, \sigma^2 \Delta_t \mathbf{I})$

$\beta_k$  = habitat selection coefficient for covariate  $k$

$x_k(\boldsymbol{\mu})$  = covariate  $k$  evaluated at  $\boldsymbol{\mu}$

$\Delta_t$  = interval duration between times  $t$  and  $t + 1$

# Multistate Langevin diffusion

$$[\boldsymbol{\mu}_{t+1} \mid \boldsymbol{\mu}_t, \boldsymbol{\sigma}, \boldsymbol{\beta}, s_t] \equiv \mathcal{N} \left( \boldsymbol{\mu}_t + \frac{\sigma_{s_t}^2 \Delta_t}{2} \nabla \log \pi_{s_t}(\boldsymbol{\mu}_t \mid \boldsymbol{\beta}_{s_t}), \sigma_{s_t}^2 \Delta_t \mathbf{I} \right)$$

$s_t$  = state from time  $t$  to  $t+1$  ( $s_t \in \{1, \dots, S\}$ )

$$\pi_s(\boldsymbol{\mu} \mid \boldsymbol{\beta}_s) = \frac{\exp \left( \sum_{k=1}^{K_s} x_k(\boldsymbol{\mu}) \beta_{s,k} \right)}{\int_{\mathcal{M}} \exp \left( \sum_{k=1}^{K_s} x_k(\mathbf{z}) \beta_{s,k} \right) d\mathbf{z}}$$

$$\nabla \log \pi_s(\boldsymbol{\mu} \mid \boldsymbol{\beta}_s) = \sum_{k=1}^{K_s} \beta_{s,k} \nabla x_k(\boldsymbol{\mu})$$

→ This can simply be the observation distribution ( $\mathbf{P}_t$ ) of a (continuous-time) hidden Markov model!

# Continuous-time HMM

$$\mathcal{L}(\theta | \mu) = \delta \mathbf{P}_1 \left[ \prod_{t=2}^T \Gamma_t \mathbf{P}_t \right] \mathbf{1}$$

$$\mathbf{P}_t = \begin{bmatrix} [\mu_{t+1} | \mu_t, \theta, s_t = 1] & 0 & \dots & 0 \\ 0 & [\mu_{t+1} | \mu_t, \theta, s_t = 2] & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & [\mu_{t+1} | \mu_t, \theta, s_t = S] \end{bmatrix}$$

$$\Gamma_t = \exp(\mathbf{Q} \Delta_t)$$

$$\mathbf{Q} = \begin{bmatrix} s_{t+1} = 1 & s_{t+1} = 2 & \dots & s_{t+1} = S \\ -q_{1,1} & q_{1,2} & \dots & q_{1,S} \\ q_{2,1} & -q_{2,2} & \dots & q_{2,S} \\ \vdots & \vdots & \ddots & \vdots \\ q_{S,1} & q_{S,2} & \dots & -q_{S,S} \end{bmatrix} \quad \begin{array}{l} s_t = 1 \\ s_t = 2 \\ \vdots \\ s_t = S \end{array}$$

# Fitting multistate Langevin models in momentuHMM

```
library(remote)
install_github("bmcclintock/momentuHMM@develop")
```

- ▶ Relies on the pseudo-design matrix (`DM`) argument and `langevin()` special function
- ▶ Pseudo-design matrix is analogous to  $\mathbf{X}$  in GLMs:

$$g(\boldsymbol{\theta}) = \mathbf{X}\boldsymbol{\beta},$$

where  $g()$  is a link function (e.g. log, logit)

- ▶ Rows of  $\mathbf{X}$  correspond to (real-scale) data stream distribution parameters ( $\boldsymbol{\theta}$ )
- ▶ Columns of  $\mathbf{X}$  correspond to (working-scale) data stream distribution parameters ( $\boldsymbol{\beta}$ )

# The pseudo-design matrix

```
# m is a momentuHMM object (as returned by fitHMM), automatically loaded with the package
m <- example$m

# default identity matrix for step ~ gamma(mean, sd) with no covariates
X <- m$conditions$fullDM$step
X

##      mean_1:(Intercept) mean_2:(Intercept) sd_1:(Intercept) sd_2:(Intercept)
## mean_1                  1                 0                 0                 0
## mean_2                  0                 1                 0                 0
## sd_1                   0                 0                 1                 0
## sd_2                   0                 0                 0                 1

# working-scale parameters (on log scale)
beta <- m$CIbeta$step$est
beta

##      mean_1:(Intercept) mean_2:(Intercept) sd_1:(Intercept) sd_2:(Intercept)
## [1,]        2.680676       5.014916       2.149273       3.034212
```

# The pseudo-design matrix

```
# mean and sd use log link function
exp(X %*% t(beta))

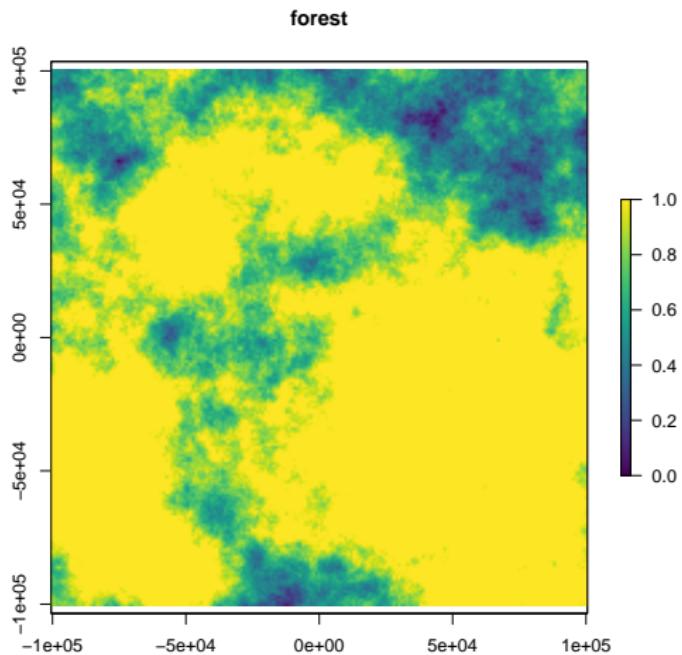
## [1,]
## mean_1 14.594959
## mean_2 150.643519
## sd_1    8.578616
## sd_2    20.784598

# compare to real-scale parameter estimates
m$CIreal$step$est

## state 1 state 2
## mean 14.594959 150.6435
## sd    8.578616 20.7846
```

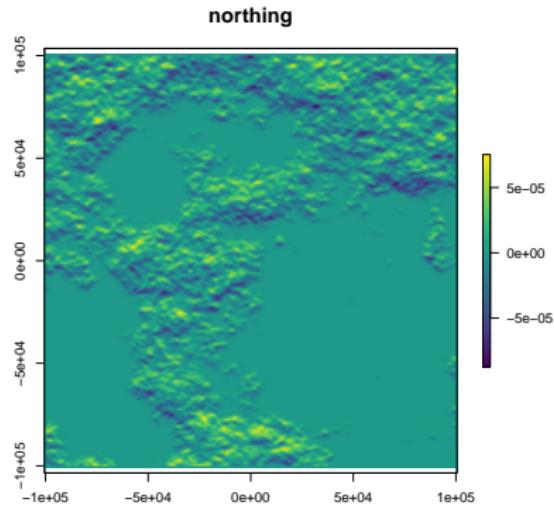
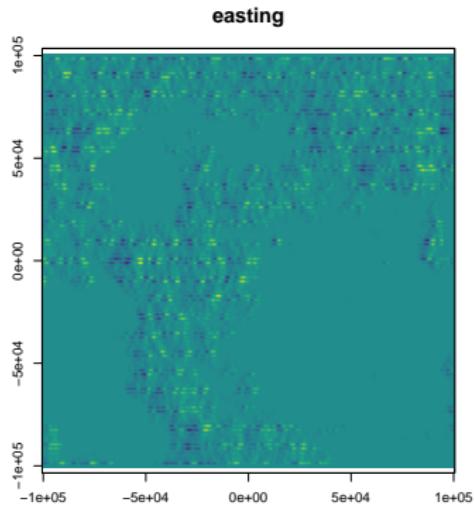
# Fitting multistate Langevin models in momentuHMM

prepData can calculate gradients for raster covariate data



# Fitting multistate Langevin models in momentuHMM

prepData can calculate gradients for raster covariate data



# Fitting multistate Langevin models in momentuHMM

prepData can calculate gradients for raster covariate data

```
# prepare data and calculate gradients
tracks <- prepData(data, altCoordNames = "mu",
                     spatialCovs = covlist,
                     gradient = TRUE)
head(tracks)

##     ID      step      angle      mu.x      mu.y    forest    forest.x
## 1  1 1066.1577        NA  0.0000  0.0000 0.7108917 1.140799e-04
## 2  1 386.0116  0.5715145 108.6667 -1060.6054 0.7155679 1.071387e-04
## 3  1 560.5277  1.7486863 349.4669 -1362.3010 0.7155679 1.157587e-04
## 4  1 295.9300 -0.8186218 718.7713 -940.6311 0.8209750 1.059219e-04
## 5  1 5802.6805 -1.1914430 1014.5414 -930.9015 0.8209750 -1.351290e-05
## 6  1 8165.3330 -0.2604089 3339.4429 -6247.4732 0.9614867 2.727722e-05
##           forest.y
## 1 -2.585472e-05
## 2  5.085680e-05
## 3  4.397668e-05
## 4  1.557643e-06
## 5  4.423903e-06
## 6  3.605136e-05
```

## Fitting multistate Langevin models in `momentuHMM`

`langevin()` special function in DM argument of `fitCTHMM`

```
# define observation distribution
dist <- list(mu="rw_mvnorm2") # bivariate normal random walk
```

```
# define single-state pseudo-design matrix
DM1 <- list(mu=matrix(c("mu.x_tm1","langevin(cov1.x)","langevin(cov2.x)",0,0,
                      "mu.y_tm1","langevin(cov1.y)","langevin(cov2.y)",0,0,
                      0,0,0,1,0,
                      0,0,0,0,1,
                      0,0,0,1,0),5,5,byrow=TRUE,
dimnames=list(c("mean.x","mean.y","sigma.x","sigma.xy","sigma.y"),
              c("mean","cov1","cov2","sigma","sigma.xy"))))
```

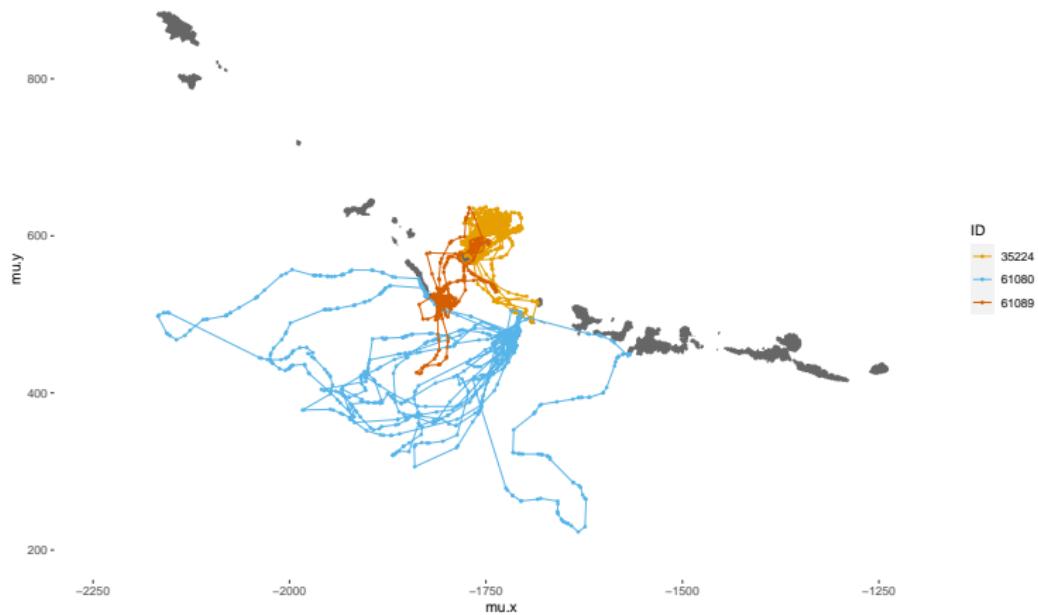
DM1

```
## $mu
##      mean      cov1      cov2      sigma sigma.xy
## mean.x "mu.x_tm1" "langevin(cov1.x)" "langevin(cov2.x)" "0"    "0"
## mean.y "mu.y_tm1" "langevin(cov1.y)" "langevin(cov2.y)" "0"    "0"
## sigma.x "0"        "0"          "0"          "1"    "0"
## sigma.xy "0"        "0"          "0"          "0"    "1"
## sigma.y "0"        "0"          "0"          "1"    "0"
```

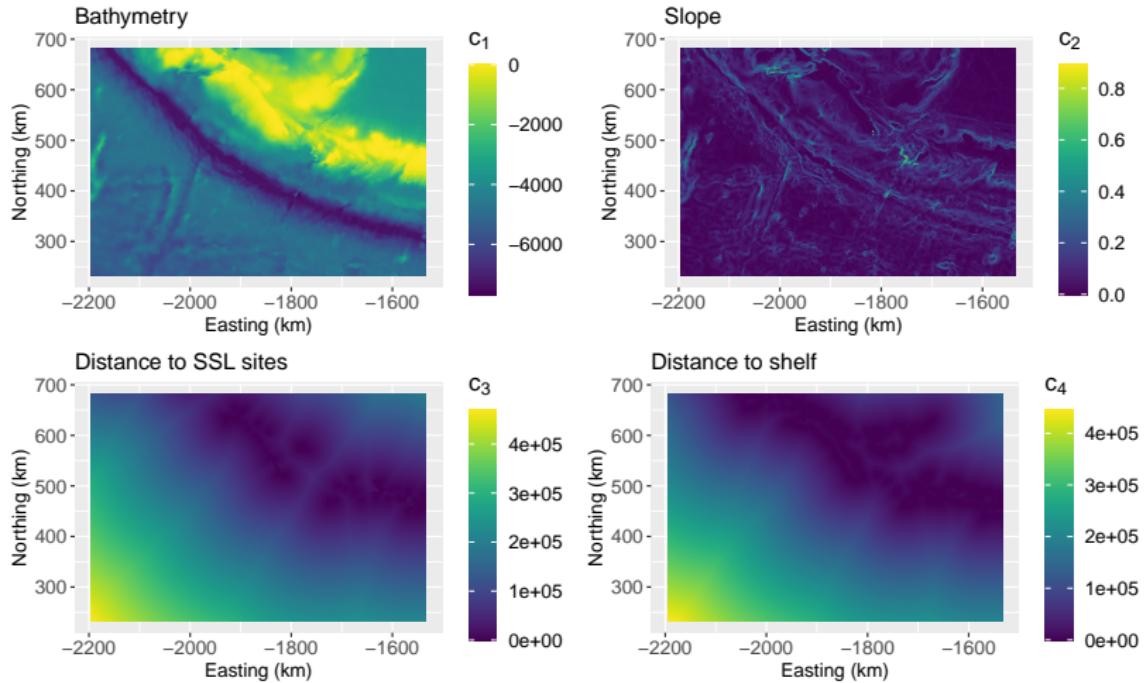
```
# fit model
```

```
fitLangevin <- fitCTHMM(tracks, nbStates=1, dist=dist, DM=DM1, Par0=Par0)
```

# Illustration: Steller sea lions

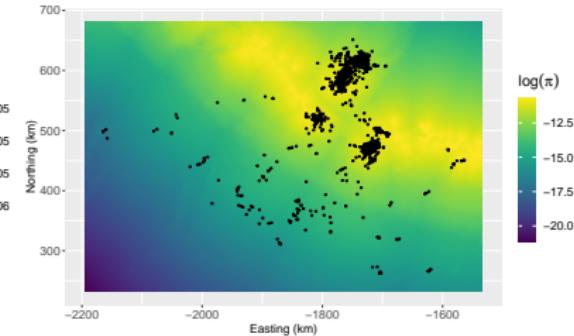
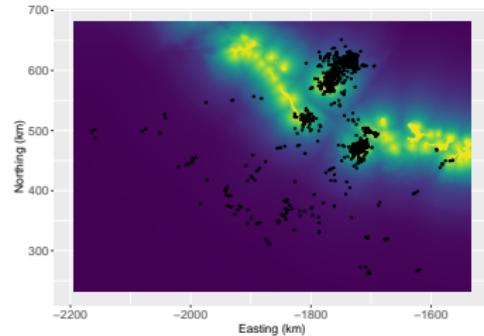


# Illustration: Steller sea lions



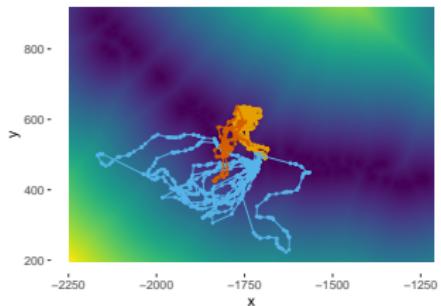
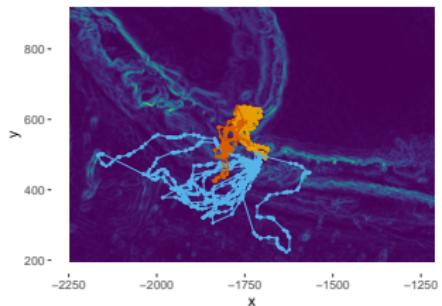
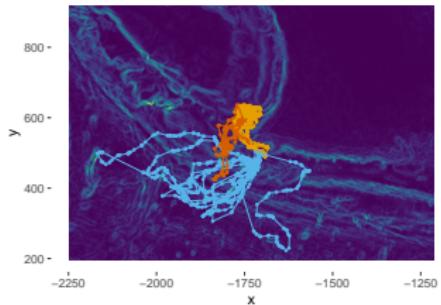
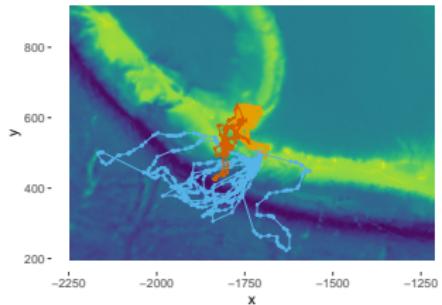
Michelot et al. (2019). The Langevin diffusion as a continuous-time model of animal movement and habitat selection, *Methods Ecol Evol.*

# Illustration: Steller sea lions



Michelot et al. (2019). The Langevin diffusion as a continuous-time model of animal movement and habitat selection, *Methods Ecol Evol.*

# Illustration: Steller sea lions



## Illustration: Steller sea lions

$$[\mu_{n,t+1} \mid \mu_{n,t}, \theta, s_{n,t}] \equiv$$
$$\begin{cases} \mathcal{N}\left(\mu_{n,t} + \frac{\sigma_{n,AS}^2 \Delta_t}{2} \nabla \log \pi_{s_{n,t}}(\mu_{n,t} \mid \beta_{s_{n,t}}), \sigma_{n,AS}^2 \Delta_t \mathbf{I}\right) & \text{if } s_{n,t} \in \{1, 2, 3\} \\ \mathcal{N}(\mu_{n,t}, \sigma_{HO}^2 \Delta_t \mathbf{I}) & \text{if } s_{n,t} = 4 \end{cases}$$

$$[\mu_{n,t+1} \mid \mu_{n,t}, \theta, s_{n,t}] \equiv$$
$$\begin{cases} \mathcal{N}\left(\mu_{n,t} + \frac{\sigma_{n,AS}^2 \Delta_{n,t}}{2} \nabla \log \pi_{s_{n,t}}(\mu_{n,t} \mid \beta_{s_{n,t}}), \sigma_{n,AS}^2 \Delta_{n,t} \mathbf{I}\right) & \text{if } s_{n,t} \in \{1, 3, 4\} \\ \mathcal{N}\left(\mu_{n,t} + \frac{\sigma_{n,2}^2 \Delta_{n,t}}{2} \nabla \log \pi_2(\mu_{n,t} \mid \beta_2), \sigma_{n,2}^2 \Delta_{n,t} \mathbf{I}\right) & \text{if } s_{n,t} = 2 \\ \mathcal{N}(\mu_{n,t}, \sigma_{HO}^2 \Delta_{n,t} \mathbf{I}) & \text{if } s_{n,t} = 5 \end{cases}$$

$$\beta_s = (\beta_{depth,s}, \beta_{slope,s}, \beta_{depth:slope,s}, \beta_{d2site,s}, \beta_{d2site^2,s})$$

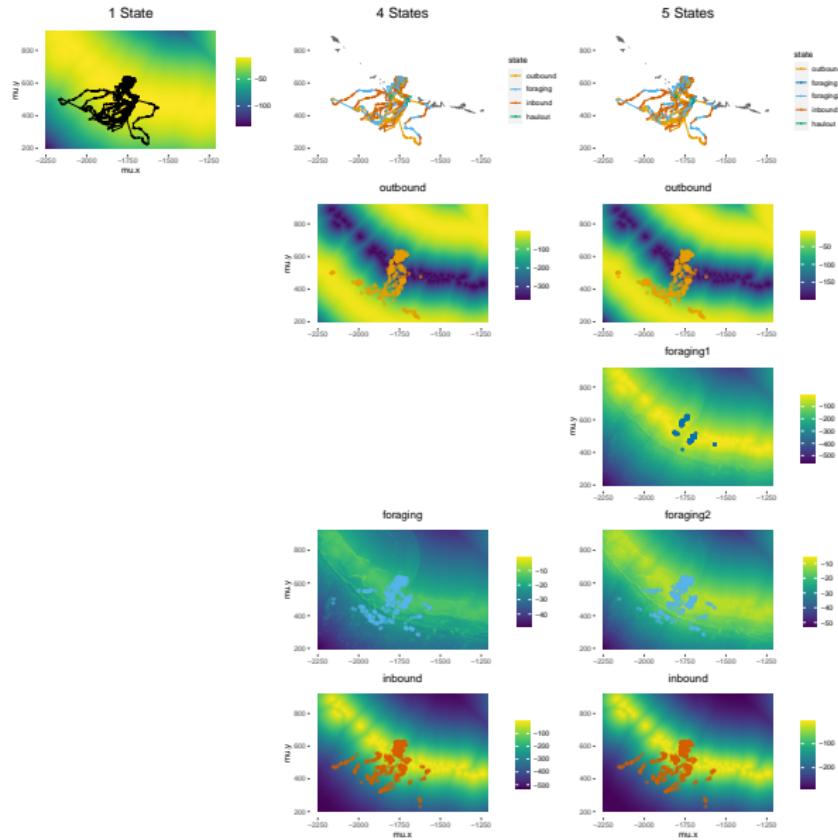
## Illustration: Steller sea lions

$$\Gamma_t = \exp(\mathbf{Q}_t \Delta_t)$$

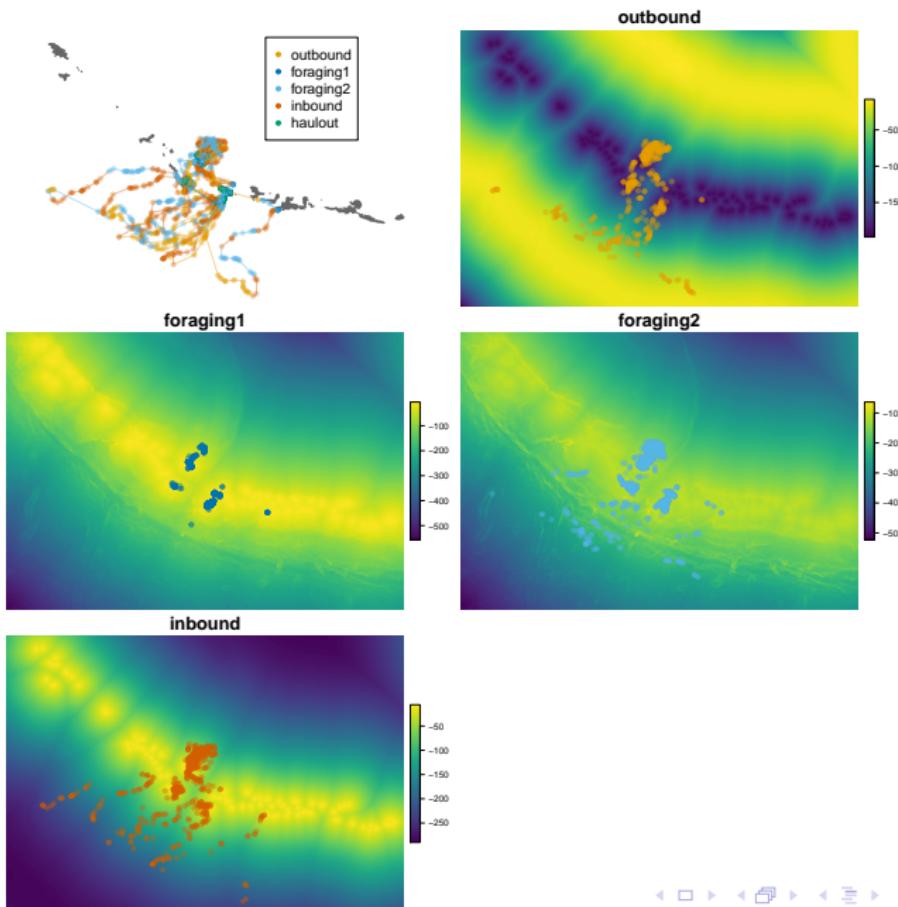
$$\mathbf{Q}_t = \begin{bmatrix} s_{t+1}=1 & s_{t+1}=2 & \dots & s_{t+1}=S \\ -q_{t,1,1} & q_{t,1,2} & \dots & q_{t,1,S} \\ q_{t,2,1} & -q_{t,2,2} & \dots & q_{t,2,S} \\ \vdots & \vdots & \ddots & \vdots \\ q_{t,S,1} & q_{t,S,2} & \dots & -q_{t,S,S} \end{bmatrix} \begin{array}{l} s_t=1 \\ s_t=2 \\ \vdots \\ s_t=S \end{array}$$

$$\log(q_{t,i,j}) = \alpha_{0,i,j} + \alpha_{1,i,j} \text{d2site}(\boldsymbol{\mu}_t) \text{ for } i \neq j$$

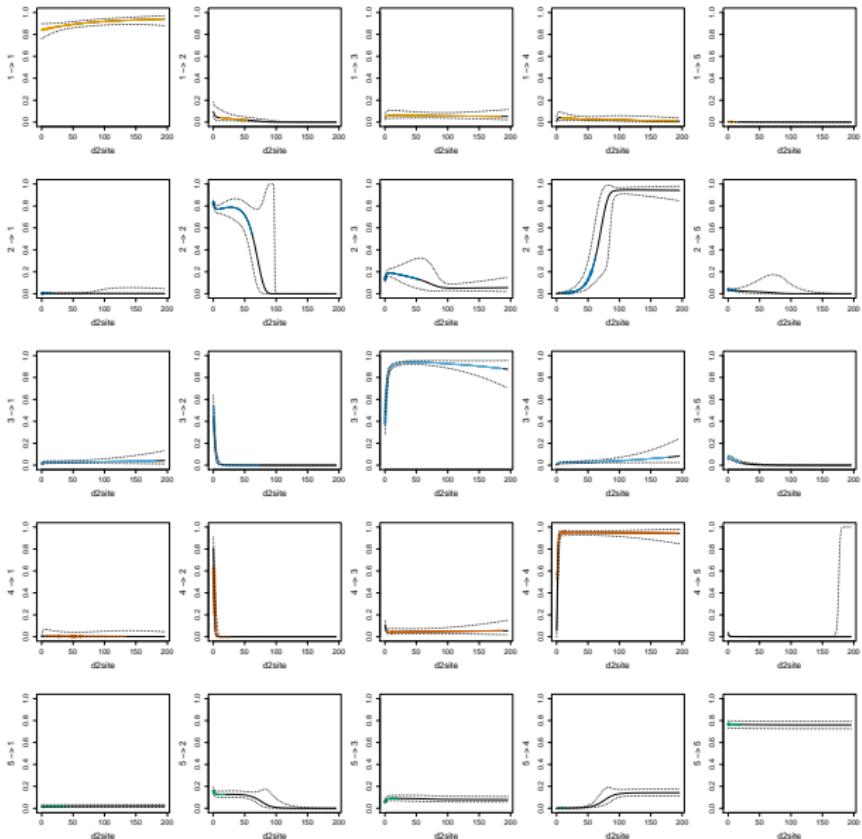
## Illustration: Steller sea lions



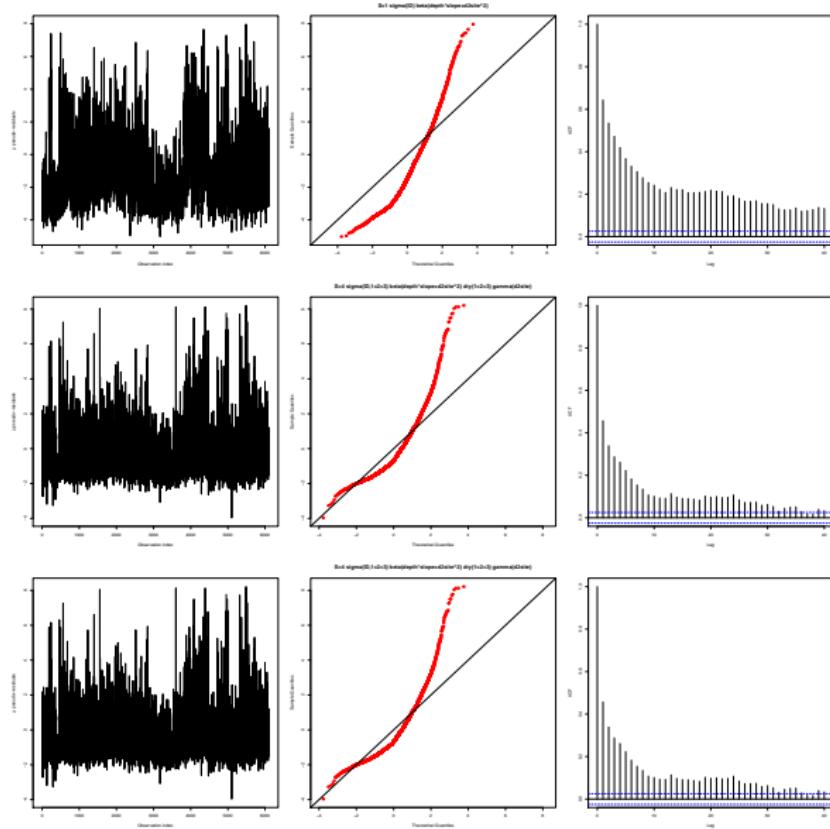
# Illustration: Steller sea lions



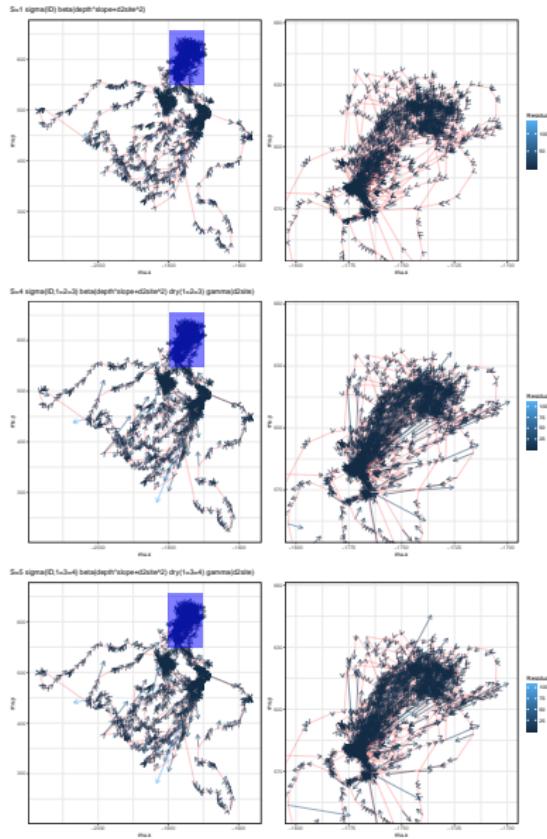
# State transition probabilities



# Pseudo-residuals



# Predicted steps and residuals



# Simulation

```
# simulate from fitted model  
simLangevin <- simCTHMM(model=fitLangevin, spatialCovs=covlist)
```

