Package: spatialHMRF (via r-universe)

November 29, 2026

Title Spatial Domains with a Hidden Markov Random Field

Version 0.1.1

Description Finds spatial domains by fitting the hidden Markov random field (HMRF) model of Zhu et al. (2018) <doi:10.1038/nbt.4260>: each cell's z-scored expression of spatial genes follows a multivariate Gaussian per domain, and a Potts prior favours domains shared with spatial neighbours. The model is fitted by expectation-maximization with iterated conditional modes over the colour classes of the spatial network, from a k-means start, as in the 'smfishHmrf' package. Colour classes are independent sets, so each update is a sparse matrix product; results equal 'smfishHmrf' on the same inputs.

License MIT + file LICENSE

Encoding UTF-8

URL https://github.com/drieslab/spatialHMRF

BugReports https://github.com/drieslab/spatialHMRF/issues

Depends R (>= 4.1.0)

Imports igraph (>= 1.5.0), Matrix, stats

Suggests testthat (>= 3.0.0)

Config/testthat/edition 3

Roxygen list(markdown = TRUE)

Config/roxygen2/version 8.0.0

Config/pak/sysreqs libglpk-dev libxml2-dev

Repository https://drieslab.r-universe.dev

Date/Publication 2026-11-24 21:51:40 UTC

RemoteUrl https://github.com/drieslab/spatialHMRF

RemoteRef HEAD

RemoteSha 01723b3e2c6c40ca4a8708c539658f1ce01acedb

Contents

hmrf 2
hmrf_adjacency 4
hmrf_init 4

Index **6**

`hmrf` *Spatial domains with a hidden Markov random field*

Description

Fits the hidden Markov random field (HMRF) model of Zhu et al. (2018), as in the **smfishHmrf** package: each cell's values follow a multivariate Gaussian per domain, and a Potts prior of strength `beta` favours domains shared with spatial neighbours. Fitted by expectation-maximization with iterated conditional modes (ICM) over the colour classes of the network, from a k-means start. Each colour class is an independent set, so its ICM update and the neighbour counts are sparse matrix products; the results equal **smfishHmrf**'s on the same inputs.

If you use it, please cite the original paper (see References).

Usage

```
hmrf(
  y,
  network,
  k,
  beta = 2,
  zscore = c("cell", "feature", "none"),
  init = NULL,
  blocks = NULL,
  nstart = 200L,
  seed = 100L,
  subsample = 20000L,
  maxit = 50L,
  err = 1e-07,
  label_tol = 1e-04,
  n_icm = 10L,
  damp_tolerance = 1e-05,
  damp_factor = 1.05
)
```

Arguments

`y`	cells x features numeric matrix, cell IDs as row names (e.g. the scaled expression of spatial genes)
`network`	spatial network: a `data.frame` with columns `from` and `to` (cell IDs), each pair listed once

k	integer, number of domains
beta	numeric, strength of the spatial prior (0: no prior)
zscore	"cell" (default, as Giotto's initHMRF_V2()): z-score each cell across the features; "feature": z-score each feature across the cells (for a few features on different scales, such as module scores, where "cell" leaves the covariances singular); "none": use y as given
init	optional start from hmrf_init(), e.g. to reuse one start for several beta; computed from y when NULL
blocks	optional colour classes from hmrf_blocks(); computed from network when NULL
nstart, seed, subsample	k-means starts, seed and subsample size (used when init is NULL, see hmrf_init())
maxit	integer, maximum EM iterations
err	numeric, relative change of means and covariances at which EM stops
label_tol	numeric, share of cells changing domain at which EM stops (labels can flip between two states without the parameters converging); 0 stops on err or maxit only, as **smfishHmrf**
n_icm	integer, ICM sweeps per EM iteration
damp_tolerance, damp_factor	see hmrf_init()

Value

A list: class (integer domain per cell), prob (cells x k posterior probabilities), mu (features x k), sigma (features x features x k), likelihood, iterations, init and blocks.

References

Zhu Q, Shah S, Dries R, Cai L, Yuan GC. Identification of spatially associated subpopulations by combining scRNAseq and sequential fluorescence in situ hybridization data. *Nature Biotechnology* 36, 1183–1190 (2018). doi:10.1038/nbt.4260

Examples

```
set.seed(1)
xy <- matrix(runif(400), ncol = 2) * 10
dom <- 1L + (xy[, 1] > 5)
y <- matrix(rnorm(200 * 4), 200) + c(-1.5, 1.5)[dom]
rownames(y) <- paste0("c", 1:200)
nn <- function(xy, k) {
    d <- as.matrix(stats::dist(xy)); diag(d) <- Inf
    t(apply(d, 1, order))[, seq_len(k)]
}
knn <- nn(xy, 5)
e <- unique(t(apply(cbind(rep(1:200, 5), as.vector(knn)), 1, sort)))
net <- data.frame(from = rownames(y)[e[, 1]], to = rownames(y)[e[, 2]])
fit <- hmrf(y, net, k = 2, beta = 2, zscore = "none", nstart = 10)
table(fit$class, dom)
```

hmrf_adjacency *Network helpers*

Description

hmrf_adjacency() turns an edge list of cell IDs into a symmetric sparse matrix of neighbour counts (self edges dropped); hmrf_blocks() colours it with igraph's DSatur heuristic and returns the colour classes, which are independent sets.

Usage

```
hmrf_adjacency(network, ids)

hmrf_blocks(adj)
```

Arguments

network	data.frame with columns from and to (cell IDs)
ids	cell IDs, the row order of y
adj	a matrix from hmrf_adjacency()

Value

hmrf_adjacency(): a sparse n x n matrix. hmrf_blocks(): a list of integer vectors (row positions) partitioning the cells.

hmrf_init *HMRF start*

Description

k-means start, per-domain covariances and damping, as in **smfishHmrf**: a domain whose covariance determinant is below damp_tolerance gets the smallest diagonal term (searched in steps of damp_factor from 1e-4) that brings it above. With more than subsample cells, the nstart restarts run on a random subsample of cells, k-means on all cells refines the 10 best, and the best refined solution is kept, so the cost of the restarts does not grow with the number of cells (**smfishHmrf** runs them on all cells).

Usage

```
hmrf_init(
  y,
  k,
  nstart = 200L,
  seed = 100L,
  subsample = 20000L,
  damp_tolerance = 1e-05,
  damp_factor = 1.05
)
```

Arguments

`y`	cells x features matrix (already z-scored if wanted)
`k`	integer, number of domains
`nstart`	integer, k-means starts (**smfishHmrf**: 1000)
`seed`	integer, seed (0 or negative leaves the RNG alone)
`subsample`	integer, number of cells for the restarts; `Inf` uses all cells
`damp_tolerance, damp_factor`	damping search settings

Value

A list with `mu` (features x k), `sigma` (features x features x k) and `damp` (length k).

Index

hmrf, 2
hmrf_adjacency, 4
hmrf_blocks *(*hmrf_adjacency*)*, 4
hmrf_blocks(), *3*
hmrf_init, 4
hmrf_init(), *3*