Package ‘MATHPOP’

July 15, 2024

Title Point process models for inferring properties of globular cluster system in Ultra-Diffuse Galaxies

Version 0.0.0.9

Description MArked-dependently THinned POint Process (MATHPOP) model for inferring the globular cluster counts and their luminosity functions in Ultra-Diffuse Galaxies.

License GNU GPL-v3.0

Imports Rcpp (>= 1.0.11), tidyverse, mixtools, sf, sp, raster, spatstat, VGAM, progress, tikzDevice, ggridges, wesanderson, ggnewscale, modeest, reshape2, HDInterval, parallel, loo, priorsense, ggstar

LinkingTo Rcpp, RcppArmadillo

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.2.9000

R topics documented:

`MATHPOP-package`	*Point process models for inferring properties of globular cluster system in Ultra-Diffuse Galaxies*

Description

MArked-dependently THinned POint Process (MATHPOP) model for inferring the globular cluster counts and their luminosity functions in Ultra-Diffuse Galaxies.

Details

The DESCRIPTION file: This package was not yet installed at build time.

Index: This package was not yet installed at build time.
For an overview and introduction of the package, see the MATHPOP webpage. See here for a quick start on how to use the package.

Author(s)

Dayi Li [aut, cre] (<https://orcid.org/0000-0002-5478-3966>)

Maintainer: Dayi Li <dayi.li@mail.utoronto.ca>

References

Li et al. (2024) "Discovery of Two Ultra-Diffuse Galaxies with Unusually Bright Globular Cluster Luminosity Functions via a Mark-Dependently Thinned Point Process (MATHPOP)", submitted to the Astrophysical Journal.

`err_M_cpp`	*C++ implementation of the photometric measurement uncertainty.*

Description

Compute the photometric measurement uncertainty based on

$$\sigma(M) = \beta_0 \exp(\beta_1 (M - m_0)).$$

Usage

```
err_M_cpp(M, m0 = 25.5, b0 = 0.08836, b1 = 0.645)
```

Arguments

`M`	Numeric with length 1. The true GC magnitude
`m0`	Numeric with length 1. The offset magnitude. Default to `m0 = 25.5`.
`b0`	Numeric with length 1. The uncertainty when M = m0. Default to `b0 = 0.08836`.
`b1`	Numeric with length 1. Rate at which the uncertainty increases with M. Default to `b1 = 0.645`.

Value

The measurement uncertainty at the true magnitude M.

---

fit_MATHPOP *Function that fits the MATHPOP model*

Description

Function that fits the MATHPOP model

Usage

```
fit_MATHPOP(
  Data,
  spat_dom,
  fixed_Theta,
  prior,
  p = 1,
  cf_error,
  M,
  Theta = NULL,
  tune,
  n = 1000,
  prob_model = TRUE,
  seed = 12345,
  burnin = 0.1
)
```

Arguments

Data	A data.frame: Each row is an observed GC or point source in the data. Requires at least the columns named x, y for the spatial coordinates of GCs (in physical coordinates), and the magnitudes named M. If there are more columns, they need to be the probabilities a point source is a GC.
spat_dom	A List. A list object containing a list called vertices that gives the vertices of the spatial domain, and a list of the number of integration grid called n_grid.
fixed_Theta	A List. A list that specifies the known parameters of GC system, which contains two list objects gal and UDG that specify the respective known parameters of normal galaxies and UDGs. If there are no normal galaxies in the data, gal does not need to be specified.
prior	A List. A list that specifies the parameter values of the prior distributions of the model parameters, specified in a similar fashion as fixed_Theta.
p	A numeric value or vector. Crowding effect. Either a numeric value between $(0, 1)$, or a numeric vector whose entries are all in $(0, 1)$ and length equal to n_grid in spat_dom with each numeric element being the crowding effect at the location of the spatial grid. In the current implementation, it is default to 1 (no crowding).
cf_error	A List. List of parameters for completeness fraction and measurement uncertainties.

M	An integer. Total number of iteration to run the MCMC algorithm.
Theta	A List. Starting values of the MCMC chain. Default to NULL, and specified internally.
tune	A List. Tuning parameters for initial MCMC pilot run.
n	An integer. Initial MCMC pilot run iteration. Default to 1000.
prob_model	Logical. Whether the GC data used is a probabilistic catalog or a binary catalog. Default to TRUE.
seed	An integer. Random seed value. Default to 12345.
burnin	A numeric value. A real number between 0 and 1. Percentage of the sample to be discarded as burn-in for MCMC. Default to 0.1

Value

A data frame with `floor(burn_in*M)` number of rows that gives the posterior sample of the fitted MATHPOP model.

---

f_cpp *C++ implementation of the completeness fraction.*

Description

Compute the completeness fraction based on

$$f(M) = \frac{1}{\exp(\alpha(M - M_{\text{Lim}}))}.$$

Usage

```
f_cpp(M, Lim, a = 1.5)
```

Arguments

M	A numeric vector. The true GC magnitude
Lim	Numeric with length 1. 50% completeness fraction.
a	Numeric with length 1. Rate at which the completeness fraction drops with M. Default to `a = 1.5`.

Value

The completeness fraction evaluated at the true magnitude `M`.

`meas_uncertain_mix_func` *Fitting the mixture model with noisy point source data*

Description

Function to obtain probabilistic GC catalog based on the parametric finite-mixture model in Li et al. (2024).

Usage

```
meas_uncertain_mix_func(dat, n_iter, seed = 12345)
```

Arguments

`dat`	A data frame that contains the point source data used to obtain the probabilistic GC catalog. Should contain the columns `C` for GC color; `F814W` for GC magnitudes; `C_err` for measurement uncertainty in color; `M_err` for measurement uncertainty in magnitudes.
`n_iter`	An integer for number of iteration to jitter the sources with their measurement uncertainty.
`seed`	An integer for the random seed. Default to 12345.

Value

A list with three objects `prob`, `par`, and `sim`. `prob` contains the computed probability that a source is a GC. `par` contains the inferred parameter values for the finite-mixture model. `sim` contains the jittered point source data.

`phi_eM_cpp` *C++ implementation to calculate the value of the noisy GCLF given the observed noisy magnitude* `M`*.*

Description

Compute the noisy GCLF evaluated at `M` based on

$$\varphi(m; \mu_{\text{TO}}, \sigma^2) = \int_{-\infty}^{\infty} \phi(m; m_t, \sigma_M^2(m_t))\phi(m_t; \mu_{\text{TO}}, \sigma^2) dm_t,$$

where $\phi(m; m_t, \sigma_M^2(m_t))$ is a Gaussian density that gives the probability density of the noisy magnitude m given the true magnitude m_t. $\phi(m_t; \mu_{\text{TO}}, \sigma^2)$ is the true GCLF with TO point μ_{TO} and dispersion σ, evaluated at the true magnitude m_t.

Usage

```
phi_eM_cpp(M, mu, sigma, m0, b0, b1, n = 20L)
```

Arguments

M	A numeric vector. The observed GC magnitude
mu	Numeric with length 1. True GCLF TO point
sigma	Numeric with length 1. True GCLF dispersion
m0	Numeric with length 1. The offset magnitude
b0	Numeric with length 1. The uncertainty when M = m0
b1	Numeric with length 1. Rate at which the uncertainty increases with M
n	Integer. Number of integration points used to marginalize out the true magnitude. Default to 20.

Value

A numeric vector that gives the value of the noisy GCLF given the observed noisy magnitude `M`.

---

psi_f *Function to compute the (unnormalized) truncated noisy GCLF*

Description

Compute the (unnormalized) truncated noisy GCLF:

$$\varphi(m; \mu_{\text{TO}}, \sigma^2) f(m),$$

where $\varphi(m; \mu_{\text{TO}}, \sigma^2)$ is obtained from `phi_eM_cpp()`.

Usage

```
psi_f(M, mu, sigma, Lim, m0, b0, b1, a)
```

Arguments

M	A numeric vector. Observed GC magnitudes.
mu	A numeric value. True GCLF TO point.
sigma	A numeric value. True GCLF dispersion.
Lim	A numeric value. 50% completeness limit.
m0	A numeric value. Offset magnitude for measurement uncertainty.
b0	A numeric value. Measurement uncertainty when `M` = `m0`.
b1	A numeric value. Rate at which the measurement uncertainty increases as `M` increases.
a	A numeric value. Rate at which the completeness fraction decreases as `M` increases.

Value

A numeric vector giving (unnormalized) truncated noisy GCLF evaluated at `M`.

`p_eM_cpp`	*C++ implementation to calculate the proportion of remaining GCs after removal of faint GCs.*

Description

Compute the proportion of the remaining GCs after removing the faint GCs based on their noisy magnitudes and the completeness fraction $f(M)$. It computes the following integral

$$\int_{-\infty}^{\infty} \varphi(m; \mu_{\text{TO}}, \sigma^2) f(m) dm,$$

where the noisy GCLF $\varphi(m; \mu_{\text{TO}}, \sigma^2)$ is obtained from `phi_eM_cpp()`.

Usage

```
p_eM_cpp(Lim, mu, sigma, m0, b0, b1, a, n = 50L)
```

Arguments

`Lim`	Numeric with length 1. 50% completeness fraction
`mu`	Numeric with length 1. True GCLF TO point
`sigma`	Numeric with length 1. True GCLF dispersion
`m0`	Numeric with length 1. The offset magnitude
`b0`	Numeric with length 1. The uncertainty when M = m0
`b1`	Numeric with length 1. Rate at which the uncertainty increases with M
`a`	Numeric with length 1. Rate at which the completeness fraction drops with M
`n`	Integer. Number of integration points used to marginalize out the true magnitude. Default to 50.

Value

A numeric value between $(0, 1)$: the proportion of the GCs remaining after removing the faint GCs.

`Sersic_ints`	*Function to compute the intensity at the locations of a point pattern* `X` *given a Sersic intensity profile*

Description

Compute the intensity of a point pattern `X` based on the Sersic profile given by

$$\Lambda(\mathbf{X}; N, R_h, n) = \frac{N b_n^{2n}}{2\pi R_h^2 n \Gamma(2n) e} \exp\left(-b_n \left(\frac{r(\mathbf{X})}{R_h}\right)^{1/n}\right),$$

where

$$r^2(\mathbf{X}) = ((\mathbf{X}_x - c_x)\cos(\vartheta) - (\mathbf{X}_y - c_y)\sin(\vartheta))^2 + ((\mathbf{X}_x - c_x)\sin(\vartheta) + (\mathbf{X}_y - c_y)\cos(\vartheta))^2/e^2.$$

.

Usage

```
Sersic_ints(X, c, N, R_eff, e = 1, n = 0.5, theta = 0)
```

Arguments

X	A two-column matrix of the GC point pattern locations. The first column is the x coordinate, the second column is the y coordinate.
c	A vector of length two for the central location of a galaxy. The first entry is the x coordinate of the galactic center, the second entry is the y coordinate.
N	A non-negative numeric value. Mean number of GC in a galaxy.
R_eff	A non-negative numeric value. Half-number radius of the GC system of a galaxy.
e	A non-negative numeric value. Aspect ratio of the GC system of a galaxy.
n	A non-negative numeric value. Sersic index of the GC system of a galaxy.
theta	A numeric value in $(0, 2\pi)$. Orientation angle (in radian) of the GC system of a galaxy.

Value

A numeric vector with the length equal to the number of row of X that gives the Sersic intensity at the locations specified by X.

`simulate_Sersic` *Simulate point pattern from a given Sersic profile*

Description

Simulate point pattern from a given Sersic profile given by

$$\Lambda(s; N, R_h, n) = \frac{N b_n^{2n}}{2\pi R_h^2 n \Gamma(2n) e} \exp\left(-b_n \left(\frac{r(s)}{R_h}\right)^{1/n}\right),$$

where

$$r^2(s) = ((s_x - c_x)\cos(\vartheta) - (s_y - c_y)\sin(\vartheta))^2 + ((s_x - c_x)\sin(\vartheta) + (s_y - c_y)\cos(\vartheta))^2/e^2.$$

.

Usage

```
simulate_Sersic(N, c, R_eff, e, n, theta)
```

Arguments

N	A non-negative numeric value. Number of GCs to simulate.
c	A vector of length two for the central location of a galaxy. The first entry is the x coordinate of the galactic center, the second entry is the y coordinate.
R_eff	A non-negative numeric value. Half-number radius of the GC system of a galaxy.
e	A non-negative numeric value. Aspect ratio of the GC system of a galaxy.
n	A non-negative numeric value. Sersic index of the GC system of a galaxy.
theta	A numeric value in $(0, 2\pi)$. Orientation angle (in radian) of the GC system of a galaxy.

Value

A data frame with `N` number of row and two columns that gives the locations of the simulated point pattern.

`simulate_Y` *Simulate GC locations and magnitudes (without measurement uncertainty) based on a Sersic profile and Gaussian GCLF.*

Description

Simulate GC locations and magnitudes (without measurement uncertainty) based on a Sersic profile and Gaussian GCLF.

Usage

```
simulate_Y(S, l0, c, N, R_eff, e, n, theta, mu, sigma)
```

Arguments

`S`	A `SpatialPolygonsDataFrame` that gives the spatial domain on which the point pattern resides.
`l0`	A positive numeric value. The intensity (/kpc^2) of GCs in the IGM.
`c`	A matrix with two columns for the central locations of galaxies in `S`. The first column is the x coordinate of the galactic centers, the second column is the y coordinate.
`N`	A (non-negative) numeric vector with length `nrow(c)`. Each element is the mean number of GCs in a galaxy.
`R_eff`	A (non-negative) numeric vector with length `nrow(c)`. Each element is the half-number radius of the GC system of a galaxy.
`e`	A (non-negative) numeric vector with length `nrow(c)`. Each element is the aspect ratio of the GC system of a galaxy.
`n`	A (non-negative) numeric vector with length `nrow(c)`. Each element is the Sersic index of the GC system of a galaxy.
`theta`	A numeric vector with length `nrow(c)`. Each element is within (0, 2*pi) and is the orientation angle of the GC system of a galaxy.
`mu`	A numeric vector with length `nrow(c) + 1`. Each element is the GCLF TO point of each of the `nrow(c) + 1` GC sub-populations.
`sigma`	A numeric vector with length `nrow(c) + 1`. Each element is the GCLF dispersion of each of the `nrow(c) + 1` GC sub-populations.

Value

A data frame with four columns. `x`, `y` columns give the locations of the simulated GCs. `M` gives the GC magnitudes. `id` is the identifier for GC sub-population; `id = 0` means GCs from the IGM, while others are from galaxies.

`simulate_Y_noisy` *Simulate GC locations and magnitudes with noisy magnitudes based on a Sersic profile and Gaussian GCLF.*

Description

Simulate GC locations and magnitudes with noisy magnitudes based on a Sersic profile and Gaussian GCLF.

Usage

```
simulate_Y_noisy(
  S,
  l0,
  c,
  N,
  R_eff,
  e,
  n,
  theta,
  mu,
  sigma,
  m0 = 25.5,
  b0 = 0.0884,
  b1 = 0.645
)
```

Arguments

`S`	A `SpatialPolygonsDataFrame` that gives the spatial domain on which the point pattern resides.
`l0`	A positive numeric value. The intensity (/kpc^2) of GCs in the IGM.
`c`	A matrix with two columns for the central locations of galaxies in `S`. The first column is the x coordinate of the galactic centers, the second column is the y coordinate.
`N`	A (non-negative) numeric vector with length `nrow(c)`. Each element is the mean number of GCs in a galaxy.
`R_eff`	A (non-negative) numeric vector with length `nrow(c)`. Each element is the half-number radius of the GC system of a galaxy.
`e`	A (non-negative) numeric vector with length `nrow(c)`. Each element is the aspect ratio of the GC system of a galaxy.
`n`	A (non-negative) numeric vector with length `nrow(c)`. Each element is the Sersic index of the GC system of a galaxy.
`theta`	A numeric vector with length `nrow(c)`. Each element is within (0, 2*pi) and is the orientation angle of the GC system of a galaxy.
`mu`	A numeric vector with length `nrow(c) + 1`. Each element is the GCLF TO point of each of the `nrow(c) + 1` GC sub-populations.
`sigma`	A numeric vector with length `nrow(c) + 1`. Each element is the GCLF dispersion of each of the `nrow(c) + 1` GC sub-populations.

`m0`	A numeric value. The offset magnitude of the photometric measurement uncertainty.
`b0`	A numeric value. The uncertainty when `M = m0`.
`b1`	Rate at which the uncertainty increases with `M`.

Value

A data frame with six columns. `x`, `y` columns give the locations of the simulated GCs. `M` gives the noisy GC magnitudes. `Mt` gives the true GC magnitudes. `id` is the identifier for GC sub-population; `id = 0` means GCs from the IGM, while others are from galaxies. `e_M` gives the measurement uncertainty.

Index