Package ‘scanstatistics’

January 26, 2023

Type Package
Title Space-Time Anomaly Detection using Scan Statistics

Version 1.1.1
Date 2023-01-25
Encoding UTF-8
License GPL (>= 3)

URL https://github.com/promerpr/scanstatistics
BugReports https://github.com/promerpr/scanstatistics/issues
Depends R (>= 3.4)
Imports dplyr, ismev, magrittr, plyr, Rcpp, stats, sets, tibble, tidyr
Suggests purrr, doParallel, foreach, ggplot2, knitr, MASS, pscl, reshape2, rmarkdown, sp, testthat, gamlss.dist

VignetteBuilder knitr
RoxygenNote 7.2.1
ByteCompile true
SystemRequirements C++11
LinkingTo Rcpp, RcppArmadillo
LazyData true
NeedsCompilation yes
Author Benjamin Allévius [aut], Paul Romer Present [ctb, cre]
Maintainer Paul Romer Present <paul.romerpresent@fastmail.fm>
Repository CRAN
Date/Publication 2023-01-26 12:40:02 UTC
 coords_to_knn

Get the k nearest neighbors for each location, given its coordinates.

Description

Get the k nearest neighbors for each location, including the location itself. This function calls dist, so the options for the distance measure used is the same as for that one. Distances are calculated between rows.

Usage

coords_to_knn(x, k = min(10, nrow(x)), method = "euclidean", p = 2)

Arguments

x a numeric matrix, data frame or "dist" object.
k The number of nearest neighbors, counting the location itself.
method the distance measure to be used. This must be one of "euclidean", "maximum", "manhattan", "canberra", "binary" or "minkowski". Any unambiguous substring can be given.
p The power of the Minkowski distance.
Value

An integer matrix of the \( k \) nearest neighbors for each location. Each row corresponds to a location, with the first element of each row being the location itself. Locations are encoded as integers.

Examples

```r
x <- matrix(c(0, 0,
1, 0,
2, 1,
0, 4,
1, 3),
ncol = 2, byrow = TRUE)
plot(x)
coords_to_knn(x)
```

---

**df_to_matrix**

*Convert a long data frame to a wide matrix.*

**Description**

Convert a long data frame to a wide matrix, with time along the row dimension and locations along the column dimension. Values in the matrix could be e.g. the observed counts or the population.

**Usage**

```r
df_to_matrix(df, time_col = 1, location_col = 2, value_col = 3)
```

**Arguments**

- `df` A data frame with at least 3 columns.
- `time_col` Integer or string that specifies the time column.
- `location_col` Integer or string that specifies the location column.
- `value_col` Integer or string that specifies the value column.

**Value**

A matrix with time on rows and locations on columns.
Description

Given a distance matrix, calculate the $k$ nearest neighbors of each location, including the location itself. The matrix should contain only zeros on the diagonal, and all other elements should be positive.

Usage

```r
dist_to_knn(x, k = min(10, nrow(x)))
```

Arguments

- `x` A (square) distance matrix. Elements should be non-negative and the diagonal zeros, but this is not checked.
- `k` The number of nearest neighbors, counting the location itself.

Value

A matrix of integers, row $i$ containing the $k$ nearest neighbors of location $i$, including itself.

Examples

```r
x <- matrix(c(0, 0,
              1, 0,
              2, 1,
              0, 4,
              1, 3),
             ncol = 2, byrow = TRUE)
d <- dist(x, diag = TRUE, upper = TRUE)
dist_to_knn(d, k = 3)
```

flexible_zones

Computes the flexibly shaped zones as in Tango (2005).

Description

Given a matrix of $k$ nearest neighbors and an adjacency matrix for the locations involved, produces the set of flexibly shaped zones as a list of integer vectors. The locations in these zones are all connected, in the sense that any location in the zone can be reached from another by traveling through adjacent locations within the zone.

Usage

```r
flexible_zones(k_nearest, adjacency_matrix)
```
get_zone

Arguments

k_nearest  
An integer matrix of the k nearest neighbors for each location. Each row corresponds to a location, with the first element of each row being the location itself. Locations should be encoded as integers.

adjacency_matrix  
A boolean matrix, with element (i, j) set to TRUE if location j is adjacent to location i.

Value  
A list of integer vectors.

References  

Examples

A <- matrix(c(0,1,0,0,0,0,  
              1,0,1,0,0,0,  
              0,1,0,0,0,0,  
              0,0,0,1,0,0,  
              0,0,1,0,0,0,  
              0,0,0,0,0,0),  
nrow = 6, byrow = TRUE) == 1

nn <- matrix(as.integer(c(1,2,3,4,5,6,  
                          2,1,3,4,5,6,  
                          3,2,1,4,5,6,  
                          4,5,1,6,3,2,  
                          5,4,6,1,3,2,  
                          6,5,4,1,3,2)),  
nrow = 6, byrow = TRUE)

flexible_zones(nn, A)

get_zone  
Extract a zone from the set of all zones.

Description

Extract zone number n from the set of all zones.

Usage

get_zone(n, zones)
Arguments

\( n \)
A integer; the number of the zone you wish to retrieve.

\( \text{zones} \)
A list of integer vectors, representing the set of all zones.

Value

An integer vector.

Examples

\[
\text{zones} \leftarrow \text{list}(1L, 2L, 3L, 1:2, c(1L, 3L), c(2L, 3L))
\]
\[
\text{get_zone}(4, \text{zones})
\]

---

**gumbel_pvalue**

*Calculate the Gumbel \( p \)-value for a scan statistic.*

Description

Given an observed scan statistic \( \lambda^* \) and a vector of replicate scan statistics \( \lambda_i, i = 1, \ldots, R \), fit a Gumbel distribution to the replicates and calculate a \( p \)-value for the observed statistic based on the fitted distribution.

\[
\frac{1 + \sum_{i=1}^{R} I(\lambda_i > \lambda^*)}{1 + R}
\]

The function is vectorized, so multiple \( p \)-values can be calculated if several scan statistics (e.g. statistics from secondary clusters) are supplied.

Usage

\[
gumbel_pvalue(\text{observed, replicates, method = "ML", ...})
\]

Arguments

\( \text{observed} \)
A scalar containing the observed value of the scan statistic, or a vector of observed values from secondary clusters.

\( \text{replicates} \)
A vector of Monte Carlo replicates of the scan statistic.

\( \text{method} \)
Either "ML", for maximum likelihood, or "MoM", for method of moments.

\( ... \)
Additional arguments passed to ismev::gum.fit, which may include arguments passed along further to optim.

Value

The \( p \)-value or \( p \)-values corresponding to the observed scan statistic(s).
**knn_zones**

Find the increasing subsets of $k$ nearest neighbors for all locations.

**Description**

Returns the set of increasing nearest neighbor sets for all locations, as a list of integer vectors. That is, for each location the list returned contains one vector containing the location itself, another containing the location and its nearest neighbor, and so on, up to the vector containing the location and its $k - 1$ nearest neighbors.

**Usage**

```r
knn_zones(k_nearest)
```

**Arguments**

- `k_nearest` An integer matrix of with $k$ columns and as many rows as locations. The first element of each row is the integer encoding the location (and equal to the row number); the following elements are the $k - 1$ nearest neighbors in ascending order of distance.

**Value**

A list of integer vectors.

**Examples**

```r
nn <- matrix(c(1L, 2L, 4L, 3L, 5L,  2L, 1L, 3L, 4L, 5L,  3L, 2L, 4L, 1L, 5L,  4L, 1L, 2L, 3L, 5L,  5L, 3L, 4L, 2L, 1L),
ncol = 5, byrow = TRUE)

knn_zones(nn[, 1:3])
```

**mc_pvalue**

Calculate the Monte Carlo $p$-value for a scan statistic.

**Description**

Given an observed scan statistic $\lambda^*$ and a vector of replicate scan statistics $\lambda_i, i = 1, \ldots, R$, calculate the Monte Carlo $p$-value as

$$
1 + \frac{\sum_{i=1}^{R} I(\lambda_i > \lambda^*)}{1 + R}
$$

The function is vectorized, so multiple $p$-values can be calculated if several scan statistics (e.g. statistics from secondary clusters) are supplied.
Usage

\texttt{mc_pvalue(\texttt{observed}, \texttt{replicates})}

Arguments

\begin{itemize}
  \item \texttt{observed} \hspace{1cm} A scalar containing the observed value of the scan statistic, or a vector of observed values from secondary clusters.
  \item \texttt{replicates} \hspace{1cm} A vector of Monte Carlo replicates of the scan statistic.
\end{itemize}

Value

The $p$-value or $p$-values corresponding to the observed scan statistic(s).

\hspace{1cm}

\textbf{NM\_geo} \hspace{1cm} \textit{Longitude and latitude of New Mexico county seats.}

Description

A dataset containing the longitude and latitude of the county seats of New Mexico, except for Cibola county.

Usage

\texttt{NM\_geo}

Format

A data frame with 33 rows and 7 variables:

\begin{itemize}
  \item \texttt{county} \hspace{1cm} Factor; the counties of New Mexico (no spaces).
  \item \texttt{seat} \hspace{1cm} Character; the name of the county seat, i.e. the administrative center or seat of government.
  \item \texttt{area(km2)} \hspace{1cm} Numeric; the area in square kilometers of each county.
  \item \texttt{seat\_long} \hspace{1cm} Numeric; the longitude of the county seat.
  \item \texttt{seat\_lat} \hspace{1cm} Numeric; the latitude of the county seat.
  \item \texttt{center\_long} \hspace{1cm} Numeric; the longitude of the geographical center of the county.
  \item \texttt{center\_lat} \hspace{1cm} Numeric; the latitude of the geographical center of the county.
\end{itemize}

Source

**NM_map**

Data to plot the counties of New Mexico.

**Description**

Map data for New Mexico. Was created using `ggplot2::map_data`.

**Usage**

`NM_map`

**Format**

A data frame with 867 rows and 7 variables:

- `long` Numeric; longitude of county polygon corner.
- `lat` Numeric; latitude of county polygon corner.
- `group` Numeric; grouping by county.
- `order` Numeric; order of the polygon corners.
- `region` Character; region is "new mexico" for all rows.
- `subregion` Character; the county name (with spaces).
- `county` Factor; the county name (no spaces).

**NM_popcas**


**Description**

A dataset containing the population count and number of brain cancer cases in the counties of New Mexico during the years 1973–1991. The population numbers are interpolations from the censuses conducted in 1973, 1982, and 1991. Interpolations were done using a quadratic function of time. Thus the year-to-year changes are overly smooth but match the census numbers in the three years mentioned.

**Usage**

`NM_popcas`

**Format**

A data frame with 608 rows and 4 variables:

- `year` Integer; the year the cases were recorded.
- `county` Character; the name of the county (no spaces).
- `population` Integer; the population in that county and year.
- `count` Integer; the number of brain cancer cases in that county and year.
scanstatistics

scanstatistics: Space-time anomaly detection using scan statistics.

Description

The scanstatistics package provides two categories of important functions: data preparation functions, and the scan statistics themselves.

Data preparation functions

These functions prepare your data for use. In particular, it helps you define the zones which will be considered by the scan statistics.

Scan statistics

These are the functions used for space-time anomaly detection. Scan statistic functions for univariate space-time data have a name that begins with `scan_` and functions for multivariate space-time data have a name that begins with `mscan_`.

scan_bayes_negbin

Calculate the negative binomial bayesian scan statistic.

Description

Calculate the “Bayesian Spatial Scan Statistic” by Neill et al. (2006), adapted to a spatio-temporal setting. The scan statistic assumes that, given the relative risk, the data follows a Poisson distribution. The relative risk is in turn assigned a Gamma distribution prior, yielding a negative binomial marginal distribution for the counts under the null hypothesis. Under the alternative hypothesis, the

Usage

```
scan_bayes_negbin(
  counts,  # the observed counts
  zones,  # the zones
  baselines = NULL,  # the baselines for the scan statistic
  population = NULL,  # the population
  outbreak_prob = 0.05,  # the outbreak probability
  alpha_null = 1,  # the null prior for the relative risk
  beta_null = 1,  # the null prior for the relative risk
  alpha_alt = alpha_null,  # the alternative prior for the relative risk
  beta_alt = beta_null,  # the alternative prior for the relative risk
  inc_values = seq(1, 3, by = 0.1),  # the incidence values
  inc_probs = 1  # the incidence probabilities
)
```
Arguments

counts Either:
  • A matrix of observed counts. Rows indicate time and are ordered from least recent (row 1) to most recent (row nrow(counts)). Columns indicate locations, numbered from 1 and up. If counts is a matrix, the optional matrix argument baselines should also be specified.
  • A data frame with columns "time", "location", "count", "baseline". Alternatively, the column "baseline" can be replaced by a column "population". The baselines are the expected values of the counts.

zones A list of integer vectors. Each vector corresponds to a single zone; its elements are the numbers of the locations in that zone.

baselines Optional. A matrix of the same dimensions as counts. Not needed if counts is a data frame. Holds the Poisson mean parameter for each observed count. Will be estimated if not supplied (requires the population argument). These parameters are typically estimated from past data using e.g. Poisson (GLM) regression.

population Optional. A matrix or vector of populations for each location. Not needed if counts is a data frame. If counts is a matrix, population is only needed if baselines are to be estimated and you want to account for the different populations in each location (and time). If a matrix, should be of the same dimensions as counts. If a vector, should be of the same length as the number of columns in counts.

outbreak_prob A scalar; the probability of an outbreak (at any time, any place). Defaults to 0.05.

alpha_null A scalar; the shape parameter for the gamma distribution under the null hypothesis of no anomaly. Defaults to 1.

beta_null A scalar; the scale parameter for the gamma distribution under the null hypothesis of no anomaly. Defaults to 1.

alpha_alt A scalar; the shape parameter for the gamma distribution under the alternative hypothesis of an anomaly. Defaults to the same value as alpha_null.

beta_alt A scalar; the scale parameter for the gamma distribution under the alternative hypothesis of an anomaly. Defaults to the same value as beta_null.

inc_values A vector of possible values for the increase in the mean (and variance) of an anomalous count. Defaults to evenly spaced values between 1 and 3, with a difference of 0.1 between consecutive values.

inc_probs A vector of the prior probabilities of each value in inc_values. Defaults to 1, implying a discrete uniform distribution.

Value

A list which, in addition to the information about the type of scan statistic, has the following components: priors (list), posteriors (list), MLC (list) and marginal_data_prob (scalar). The list MLC has elements

zone The number of the spatial zone of the most likely cluster (MLC).
duration  The most likely event duration.
log_posterior  The posterior log probability that an event is ongoing in the MLC.
log_bayes_factor  The logarithm of the Bayes factor for the MLC.
posterior  The posterior probability that an event is ongoing in the MLC.
locations  The locations involved in the MLC.

The list priors has elements

null_prior  The prior probability of no anomaly.
alt_prior  The prior probability of an anomaly.
inc_prior  A vector of prior probabilities of each value in the argument inc_values.
window_prior  The prior probability of an outbreak in any of the space-time windows.

The list posteriors has elements

null_posterior  The posterior probability of no anomaly.
alt_posterior  The posterior probability of an anomaly.
inc_posterior  A data frame with columns inc_values and inc_posterior.
window_posteriors  A data frame with columns zone, duration, log_posterior and log_bayes_factor, each row corresponding to a space-time window.

space_time_posteriors  A matrix with the posterior anomaly probability of each location-time combination.
location_posteriors  A vector with the posterior probability of an anomaly at each location.

References


Examples

set.seed(1)
# Create location coordinates, calculate nearest neighbors, and create zones
n_locs <- 50
max_duration <- 5
n_total <- n_locs * max_duration
geo <- matrix(rnorm(n_locs * 2), n_locs, 2)
knn_mat <- coords_to_knn(geo, 15)
zones <- knn_zones(knn_mat)

# Simulate data
baselines <- matrix(rexp(n_total, 1/5), max_duration, n_locs)
counts <- matrix(rpois(n_total, as.vector(baselines)), max_duration, n_locs)

# Inject outbreak/event/anomaly
ob_dur <- 3
ob_cols <- zones[[10]]
ob_rows <- max_duration + 1 - seq_len(ob_dur)
counts[ob_rows, ob_cols] <- matrix(
  rpois(ob_dur * length(ob_cols), 2 * baselines[ob_rows, ob_cols]),
  length(ob_rows), length(ob_cols))
res <- scan_bayes_negbin(counts = counts,
                         zones = zones,
                         baselines = baselines)

 scan_eb_negbin  Calculate the expectation-based negative binomial scan statistic.

Description

Calculate the expectation-based negative binomial scan statistic devised by Tango et al. (2011).

Usage

scan_eb_negbin(
  counts,
  zones,
  baselines = NULL,
  thetas = 1,
  type = c("hotspot", "emerging"),
  n_mcsim = 0,
  gumbel = FALSE,
  max_only = FALSE
)

Arguments

counts  Either:

  • A matrix of observed counts. Rows indicate time and are ordered from
    least recent (row 1) to most recent (row nrow(counts)). Columns indicate
    locations, numbered from 1 and up. If counts is a matrix, the optional
    matrix arguments baselines and thetas should also be specified.

  • A data frame with columns "time", "location", "count", "baseline", "theta".
    See the description of the optional arguments baselines and thetas below
    to see their definition.

zones  A list of integer vectors. Each vector corresponds to a single zone; its elements
        are the numbers of the locations in that zone.

baselines  Optional. A matrix of the same dimensions as counts. Holds the expected value
           parameter for each observed count. These parameters are typically estimated
           from past data using e.g. GLM.

thetas  Optional. A matrix of the same dimensions as counts, or a scalar. Holds the
         dispersion parameter of the distribution, which is such that if \( \mu \)
         is the expected value, the variance is \( \mu + \mu^2/\theta \). These parameters
         are typically estimated from past data using e.g. GLM. If a scalar is supplied, the
         dispersion parameter is assumed to be the same for all locations and time points.
type A string, either "hotspot" or "emerging". If "hotspot", the relative risk is assumed to be fixed over time. If "emerging", the relative risk is assumed to increase with the duration of the outbreak.
n_mcsim A non-negative integer; the number of replicate scan statistics to generate in order to calculate a $P$-value.
gumbel Logical: should a Gumbel $P$-value be calculated? Default is FALSE.
max_only Boolean. If FALSE (default) the statistic calculated for each zone and duration is returned. If TRUE, only the largest such statistic (i.e. the scan statistic) is returned, along with the corresponding zone and duration.

Value

A list which, in addition to the information about the type of scan statistic, has the following components:

MLC A list containing the number of the zone of the most likely cluster (MLC), the locations in that zone, the duration of the MLC, and the calculated score. In order, the elements of this list are named zone_number, locations, duration, score.

observed A data frame containing, for each combination of zone and duration investigated, the zone number, duration, and score. The table is sorted by score with the top-scoring location on top. If max_only = TRUE, only contains a single row corresponding to the MLC.

replicates A data frame of the Monte Carlo replicates of the scan statistic (if any), and the corresponding zones and durations.

MC_pvalue The Monte Carlo $P$-value.

Gumbel_pvalue A $P$-value obtained by fitting a Gumbel distribution to the replicate scan statistics.

n_zones The number of zones scanned.

n_locations The number of locations.

max_duration The maximum duration considered.

n_mcsim The number of Monte Carlo replicates made.

References


Examples

set.seed(1)
# Create location coordinates, calculate nearest neighbors, and create zones
n_locs <- 50
max_duration <- 5
n_total <- n_locs * max_duration
geo <- matrix(rnorm(n_locs * 2), n_locs, 2)
knn_mat <- coords_to_knn(geo, 15)
zones <- knn_zones(knn_mat)
# Simulate data
baselines <- matrix(rexp(n_total, 1/5), max_duration, n_locs)
thetas <- matrix(runif(n_total, 0.05, 1), max_duration, n_locs)
counts <- matrix(rnbinom(n_total, mu = baselines, size = thetas),
                 max_duration, n_locs)

# Inject outbreak/event/anomaly
ob_dur <- 3
ob_cols <- zones[[10]]
ob_rows <- max_duration + 1 - seq_len(ob_dur)
counts[ob_rows, ob_cols] <- matrix(
    rbinom(ob_dur * length(ob_cols),
           mu = 2 * baselines[ob_rows, ob_cols],
           size = thetas[ob_rows, ob_cols]),
    length(ob_rows), length(ob_cols))
res <- scan_eb_negbin(counts = counts,
                      zones = zones,
                      baselines = baselines,
                      thetas = thetas,
                      type = "hotspot",
                      n_mcsim = 99,
                      max_only = FALSE)

----------

scan_eb_poisson  Calculate the expectation-based Poisson scan statistic.

Description
Calculate the expectation-based Poisson scan statistic devised by Neill et al. (2005).

Usage
scan_eb_poisson(
    counts, zones,
    baselines = NULL,
    population = NULL,
    n_mcsim = 0,
    gumbel = FALSE,
    max_only = FALSE
)

Arguments
counts  Either:
  - A matrix of observed counts. Rows indicate time and are ordered from
    least recent (row 1) to most recent (row nrow(counts)). Columns indicate
    locations, numbered from 1 and up. If counts is a matrix, the optional
    matrix argument baselines should also be specified.
• A data frame with columns "time", "location", "count", "baseline". Alternatively, the column "baseline" can be replaced by a column "population". The baselines are the expected values of the counts.

zones A list of integer vectors. Each vector corresponds to a single zone; its elements are the numbers of the locations in that zone.

baselines Optional. A matrix of the same dimensions as counts. Not needed if counts is a data frame. Holds the Poisson mean parameter for each observed count. Will be estimated if not supplied (requires the population argument). These parameters are typically estimated from past data using e.g. Poisson (GLM) regression.

population Optional. A matrix or vector of populations for each location. Not needed if counts is a data frame. If counts is a matrix, population is only needed if baselines are to be estimated and you want to account for the different populations in each location (and time). If a matrix, should be of the same dimensions as counts. If a vector, should be of the same length as the number of columns in counts.

n_mcsim A non-negative integer; the number of replicate scan statistics to generate in order to calculate a $P$-value.

gumbel Logical: should a Gumbel $P$-value be calculated? Default is FALSE.

max_only Boolean. If FALSE (default) the log-likelihood ratio statistic for each zone and duration is returned. If TRUE, only the largest such statistic (i.e. the scan statistic) is returned, along with the corresponding zone and duration.

Value

A list which, in addition to the information about the type of scan statistic, has the following components:

MLC A list containing the number of the zone of the most likely cluster (MLC), the locations in that zone, the duration of the MLC, the calculated score, and the relative risk. In order, the elements of this list are named zone_number, locations, duration, score, relative_risk.

observed A data frame containing, for each combination of zone and duration investigated, the zone number, duration, score, relative risk. The table is sorted by score with the top-scoring location on top. If max_only = TRUE, only contains a single row corresponding to the MLC.

replicates A data frame of the Monte Carlo replicates of the scan statistic (if any), and the corresponding zones and durations.

MC_pvalue The Monte Carlo $P$-value.

Gumbel_pvalue A $P$-value obtained by fitting a Gumbel distribution to the replicate scan statistics.

n_zones The number of zones scanned.

n_locations The number of locations.

max_duration The maximum duration considered.

n_mcsim The number of Monte Carlo replicates made.
scan_eb_zip

References

Examples

```r
set.seed(1)
# Create location coordinates, calculate nearest neighbors, and create zones
n_locs <- 50
max_duration <- 5
n_total <- n_locs * max_duration
geo <- matrix(rnorm(n_locs * 2), n_locs, 2)
knn_mat <- coords_to_knn(geo, 15)
zones <- knn_zones(knn_mat)

# Simulate data
baselines <- matrix(rexp(n_total, 1/5), max_duration, n_locs)
counts <- matrix(rpois(n_total, as.vector(baselines)), max_duration, n_locs)

# Inject outbreak/event/anomaly
ob_dur <- 3
ob_cols <- zones[[10]]
ob_rows <- max_duration + 1 - seq_len(ob_dur)
counts[ob_rows, ob_cols] <- matrix(rpois(ob_dur * length(ob_cols), 2 * baselines[ob_rows, ob_cols]), length(ob_rows), length(ob_cols))
res <- scan_eb_poisson(counts = counts,
                      zones = zones,
                      baselines = baselines,
                      n_mcsim = 99,
                      max_only = FALSE)
```

scan_eb_zip

*Calculate the expectation-based ZIP scan statistic.*

Description
Calculates the expectation-based scan statistic. See details below.

Usage
```
scan_eb_zip(
  counts,
  zones,
  baselines = NULL,
  probs = NULL,
  population = NULL,
  n_mcsim = 0,
```
gumbel = FALSE,
max_only = FALSE,
rel_tol = 0.001
)

Arguments

Counts

Either:

- A matrix of observed counts. Rows indicate time and are ordered from least recent (row 1) to most recent (row nrow(counts)). Columns indicate locations, numbered from 1 and up. If counts is a matrix, the optional matrix arguments baselines and probs should also be specified.

- A data frame with columns "time", "location", "count", "baseline", "prob". The baselines are the expected values of the counts, and "prob" are the structural zero probabilities of the counts. If "baseline" and "prob" are not found as columns, their values are estimated in a very heuristic fashion (not recommended). If population numbers are available, they can be included in a column "population" to help with the estimation.

zones

A list of integer vectors. Each vector corresponds to a single zone; its elements are the numbers of the locations in that zone.

baselines

Optional. A matrix of the same dimensions as counts. Holds the Poisson mean parameter of the ZIP distribution for each observed count. These parameters are typically estimated from past data using e.g. ZIP regression.

probs

Optional. A matrix of the same dimensions as counts. Holds the structural zero probability of the ZIP distribution for each observed count. These parameters are typically estimated from past data using e.g. ZIP regression.

population

Optional. A matrix or vector of populations for each location. Only needed if baselines and probs are to be estimated and you want to account for the different populations in each location (and time). If a matrix, should be of the same dimensions as counts. If a vector, should be of the same length as the number of columns in counts.

n_mcsim

A non-negative integer; the number of replicate scan statistics to generate in order to calculate a P-value.

gumbel

Logical: should a Gumbel P-value be calculated? Default is FALSE.

max_only

Boolean. If FALSE (default) the log-likelihood ratio statistic for each zone and duration is returned. If TRUE, only the largest such statistic (i.e. the scan statistic) is returned, along with the corresponding zone and duration.

rel_tol

A positive scalar. If the relative change in the incomplete information likelihood is less than this value, then the EM algorithm is deemed to have converged.

Details

For the expectation-based zero-inflated Poisson scan statistic (Allévius & Höhle 2017), the null hypothesis of no anomaly holds that the count observed at each location \(i\) and duration \(t\) (the number of
time periods before present) has a zero-inflated Poisson distribution with expected value parameter \( \mu_{it} \) and structural zero probability \( p_{it} \):

\[
H_0 : Y_{it} \sim \text{ZIP}(\mu_{it}, p_{it}).
\]

This holds for all locations \( i = 1, \ldots, m \) and all durations \( t = 1, \ldots, T \), with \( T \) being the maximum duration considered. Under the alternative hypothesis, there is a space-time window \( W \) consisting of a spatial zone \( Z \subset \{1, \ldots, m\} \) and a time window \( D \subseteq \{1, \ldots, T\} \) such that the counts in that window have their Poisson expected value parameters inflated by a factor \( q_W > 1 \) compared to the null hypothesis:

\[
H_1 : Y_{it} \sim \text{ZIP}(q_W \mu_{it}, p_{it}), \quad (i, t) \in W.
\]

For locations and durations outside of this window, counts are assumed to be distributed as under the null hypothesis. The sets \( Z \) considered are those specified in the argument zones, while the maximum duration \( T \) is taken as the maximum value in the column duration of the input table.

For each space-time window \( W \) considered, (the log of) a likelihood ratio is computed using the distributions under the alternative and null hypotheses, and the expectation-based Poisson scan statistic is calculated as the maximum of these quantities over all space-time windows. The expectation-maximization (EM) algorithm is used to obtain maximum likelihood estimates.

Value

A list which, in addition to the information about the type of scan statistic, has the following components:

**MLC** A list containing the number of the zone of the most likely cluster (MLC), the locations in that zone, the duration of the MLC, the calculated score, the relative risk, and the number of iterations until convergence for the EM algorithm. In order, the elements of this list are named zone_number, locations, duration, score, relative_risk, n_iter.

**observed** A data frame containing, for each combination of zone and duration investigated, the zone number, duration, score, relative risk, number of EM iterations. The table is sorted by score with the top-scoring location on top. If max_only = TRUE, only contains a single row corresponding to the MLC.

**replicates** A data frame of the Monte Carlo replicates of the scan statistic (if any), and the corresponding zones and durations.

**MC_pvalue** The Monte Carlo \( P \)-value.

**Gumbel_pvalue** A \( P \)-value obtained by fitting a Gumbel distribution to the replicate scan statistics.

**n_zones** The number of zones scanned.

**n_locations** The number of locations.

**max_duration** The maximum duration considered.

**n_mcsim** The number of Monte Carlo replicates made.

References

Allévius, B. and Höhle, M, *An expectation-based space-time scan statistic for ZIP-distributed data*, (Technical report), Link to PDF.
Examples

```r
if (require("gamlss.dist")) {
  set.seed(1)
  # Create location coordinates, calculate nearest neighbors, and create zones
  n_locs <- 50
  max_duration <- 5
  n_total <- n_locs * max_duration
  geo <- matrix(rnorm(n_locs * 2), n_locs, 2)
  knn_mat <- coords_to_knn(geo, 15)
  zones <- knn_zones(knn_mat)
  # Simulate data
  baselines <- matrix(rexp(n_total, 1/5), max_duration, n_locs)
  probs <- matrix(runif(n_total) / 4, max_duration, n_locs)
  counts <- matrix(gamlss.dist::rZIP(n_total, baselines, probs),
                   max_duration, n_locs)
  # Inject outbreak/event/anomaly
  ob_dur <- 3
  ob_cols <- zones[[10]]
  ob_rows <- max_duration + 1 - seq_len(ob_dur)
  counts[ob_rows, ob_cols] <- gamlss.dist::rZIP(
    ob_dur * length(ob_cols), 2 * baselines[ob_rows, ob_cols],
    2 * length(ob_cols)
  )
  res <- scan_eb_zip(counts = counts,
                     zones = zones,
                     baselines = baselines,
                     probs = probs,
                     n_mcsim = 9,
                     max_only = FALSE,
                     rel_tol = 1e-3)
}
```

scan_pb_poisson

Calculate the population-based Poisson scan statistic.

Description


Usage

```r
scan_pb_poisson(
  counts,
  zones,
  population = NULL,
  n_mcsim = 0,
  gumbel = FALSE,
  max_only = FALSE
)
```
Arguments

- **counts**: Either:
  - A matrix of observed counts. Rows indicate time and are ordered from least recent (row 1) to most recent (row `nrow(counts)`). Columns indicate locations, numbered from 1 and up. If `counts` is a matrix, the optional argument `population` should also be specified.
  - A data frame with columns "time", "location", "count", "population".

- **zones**: A list of integer vectors. Each vector corresponds to a single zone; its elements are the numbers of the locations in that zone.

- **population**: Optional. A matrix or vector of populations for each location and time point. Only needed if baselines are to be estimated and you want to account for the different populations in each location (and time). If a matrix, should be of the same dimensions as `counts`. If a vector, should be of the same length as the number of columns in `counts` (the number of locations).

- **n_mcsim**: A non-negative integer; the number of replicate scan statistics to generate in order to calculate a P-value.

- **gumbel**: Logical: should a Gumbel P-value be calculated? Default is `FALSE`.

- **max_only**: Boolean. If `FALSE` (default) the log-likelihood ratio statistic for each zone and duration is returned. If `TRUE`, only the largest such statistic (i.e. the scan statistic) is returned, along with the corresponding zone and duration.

Value

A list which, in addition to the information about the type of scan statistic, has the following components:

- **MLC**: A list containing the number of the zone of the most likely cluster (MLC), the locations in that zone, the duration of the MLC, the calculated score, and the relative risk inside and outside the cluster. In order, the elements of this list are named `zone_number`, `locations`, `duration`, `score`, `relrisk_in`, `relrisk_out`.

- **observed**: A data frame containing, for each combination of zone and duration investigated, the zone number, duration, score, relative risks. The table is sorted by score with the top-scoring location on top. If `max_only = TRUE`, only contains a single row corresponding to the MLC.

- **replicates**: A data frame of the Monte Carlo replicates of the scan statistic (if any), and the corresponding zones and durations.

- **MC_pvalue**: The Monte Carlo P-value.

- **Gumbel_pvalue**: A P-value obtained by fitting a Gumbel distribution to the replicate scan statistics.

- **n_zones**: The number of zones scanned.

- **n_locations**: The number of locations.

- **max_duration**: The maximum duration considered.

- **n_mcsim**: The number of Monte Carlo replicates made.
References


Examples

```r
set.seed(1)
# Create location coordinates, calculate nearest neighbors, and create zones
n_locs <- 50
max_duration <- 5
n_total <- n_locs * max_duration
geo <- matrix(rnorm(n_locs * 2), n_locs, 2)
knn_mat <- coords_to_knn(geo, 15)
zones <- knn_zones(knn_mat)

# Simulate data
population <- matrix(rnorm(n_total, 100, 10), max_duration, n_locs)
counts <- matrix(rpois(n_total, as.vector(population) / 20),
                 max_duration, n_locs)

# Inject outbreak/event/anomaly
ob_dur <- 3
ob_cols <- zones[[10]]
ob_rows <- max_duration + 1 - seq_len(ob_dur)
counts[ob_rows, ob_cols] <- matrix(
rpois(ob_dur * length(ob_cols), 2 * population[ob_rows, ob_cols] / 20),
length(ob_rows), length(ob_cols))
res <- scan_pb_poisson(counts = counts,
                      zones = zones,
population = population,
n_mcsim = 99,
max_only = FALSE)
```

scan_permutation

*Calculate the space-time permutation scan statistic.*

Description

Calculate the space-time permutation scan statistic devised by Kulldorff (2005).

Usage

```r
scan_permutation(
  counts,
  zones,
  population = NULL,
```
n_mcsim = 0,  
gumbel = FALSE,  
max_only = FALSE 
)

Arguments

counts Either:
• A matrix of observed counts. Rows indicate time and are ordered from least recent (row 1) to most recent (row nrow(counts)). Columns indicate locations, numbered from 1 and up. If counts is a matrix, the optional argument population should also be specified.
• A data frame with columns "time", "location", "count", "population".

zones A list of integer vectors. Each vector corresponds to a single zone; its elements are the numbers of the locations in that zone.

population Optional. A matrix or vector of populations for each location and time point. Only needed if baselines are to be estimated and you want to account for the different populations in each location (and time). If a matrix, should be of the same dimensions as counts. If a vector, should be of the same length as the number of columns in counts (the number of locations).

n_mcsim A non-negative integer; the number of replicate scan statistics to generate in order to calculate a P-value.

gumbel Logical: should a Gumbel P-value be calculated? Default is FALSE.

max_only Boolean. If FALSE (default) the log-likelihood ratio statistic for each zone and duration is returned. If TRUE, only the largest such statistic (i.e. the scan statistic) is returned, along with the corresponding zone and duration.

Value

A list which, in addition to the information about the type of scan statistic, has the following components:

MLC A list containing the number of the zone of the most likely cluster (MLC), the locations in that zone, the duration of the MLC, the calculated score, and the relative risk inside and outside the cluster. In order, the elements of this list are named zone_number, locations, duration, score, relrisk_in, relrisk_out.

observed A data frame containing, for each combination of zone and duration investigated, the zone number, duration, score, relative risks. The table is sorted by score with the top-scoring location on top. If max_only = TRUE, only contains a single row corresponding to the MLC.

replicates A data frame of the Monte Carlo replicates of the scan statistic (if any), and the corresponding zones and durations.

MC_pvalue The Monte Carlo P-value.

Gumbel_pvalue A P-value obtained by fitting a Gumbel distribution to the replicate scan statistics.

n_zones The number of zones scanned.

n_locations The number of locations.
**score_locations**  
Score each location over zones and duration.

### Description

For each location, compute the average of the statistic calculated for each space-time window that the location is included in, i.e. average the statistic over both zones and the maximum duration.

### Usage

score_locations(x, zones)
**top_clusters**

**Arguments**

- **x**: An object of class `scanstatistic`.
- **zones**: A list of integer vectors.

**Value**

A `data.table` with the following columns:

- **location**: The locations (as integers).
- **total_score**: For each location, the sum of all window statistics that the location appears in.
- **n_zones**: The number of spatial zones that the location appears in.
- **score**: The total score divided by the number of zones and the maximum duration.
- **relative_score**: The score divided by the maximum score.

**Examples**

```r
# Simple example
set.seed(1)
table <- data.frame(zone = 1:5, duration = 1, score = 5:1)
zones <- list(1:2, 1:3, 2:5, 4:5, c(1, 5))
x <- list(observed = table, n_locations = 5, max_duration = 1, n_zones = 5)
score_locations(x, zones)
```

**Description**

Get the top \( k \) space-time clusters according to the statistic calculated for each cluster (the maximum being the scan statistic). The default is to return the spatially non-overlapping clusters, i.e. those that do not have any locations in common.

**Usage**

```r
top_clusters(  
  x,  
  zones,  
  k = 5,  
  overlapping = FALSE,  
  gumbel = FALSE,  
  alpha = NULL,  
  ...  
)
```
Arguments

- x: An object of class scanstatistics.
- zones: A list of integer vectors.
- k: An integer, the number of clusters to return.
- overlapping: Logical; should the top clusters be allowed to overlap in the spatial dimension? The default is FALSE.
- gumbel: Logical; should a Gumbel P-value be calculated? The default is FALSE.
- alpha: A significance level, which if not NULL will be used to calculate a critical value for the statistics in the table.
- ...: Parameters passed to quantile.

Value

A data frame with at most \( k \) rows, with columns `zone`, `duration`, `score` and possibly `MC_pvalue`, `Gumbel_pvalue` and `critical_value`.

Examples

```r
set.seed(1)
counts <- matrix(rpois(15, 3), 3, 5)
zones <- list(1:2, 1:3, 2:5, c(1, 3), 4:5, c(1, 5))
scanres <- scan_permutation(counts, zones, n_mcsim = 5)
top_clusters(scanres, zones, k = 4, overlapping = FALSE)
```
Index

* datasets
  NM_geo, 8
  NM_map, 9
  NM_popcas, 9

coords_to_knn, 2

df_to_matrix, 3
dist, 2
dist_to_knn, 4

flexible_zones, 4

get_zone, 5
gumbel_pvalue, 6

knn_zones, 7

mc_pvalue, 7

NM_geo, 8
NM_map, 9
NM_popcas, 9

quantile, 26

scan_bayes_negbin, 10
scan_eb_negbin, 13
scan_eb_poisson, 15
scan_eb_zip, 17
scan_pb_poisson, 20
scan_permutation, 22
scanstatistics, 10
score_locations, 24

top_clusters, 25