Package ‘geonet’

August 6, 2021

Title  Intensity Estimation on Geometric Networks with Penalized Splines
Version  0.6.0
Date  2021-08-05

Language  en-US
License  MIT + file LICENSE
Encoding  UTF-8
LazyData  true
RoxygenNote  7.1.1
Imports  spatstat.linnet, spatstat.geom, spatstat.core, mgcv, Matrix, splines, igraph, dplyr (>= 1.0.2), ggplot2 (>= 3.3.2)
Depends  R (>= 4.0.0),
Suggests  testthat, spatstat, spatstat.data
NeedsCompilation  no
Author  Marc Schneble [aut, cre]
Maintainer  Marc Schneble <marc.schneble@stat.uni-muenchen.de>
Repository  CRAN
Date/Publication  2021-08-06 12:10:02 UTC

R topics documented:

as.linnet.gn ....................................................... 2
as_gn ................................................................. 3
as_gnpp ............................................................... 4
as_lpp ................................................................. 5
as.linnet.gn

**Transmute to Linear Network**

**Description**

as.linnet.gn is a method for the generic function as.linnet which transmutes a geometric network (object of class gn) to a linear network (object of class linnet).

**Usage**

```r
## S3 method for class 'gn'
as.linnet(X, ...)
```

**Arguments**

- `X` An object of class gn.
- `...` Other arguments.
**as_gn**

Description

as_gn transmutes an existing object into a geometric network, an object of class gn.

Usage

```r
as_gn(x, ...)
```

## S3 method for class 'linnet'

```r
as_gn(x, ..., spatstat = FALSE)
```

## S3 method for class 'gnpp'

```r
as_gn(x, ...)
```

## S3 method for class 'gnppfit'

```r
as_gn(x, ...)
```

## S3 method for class 'lpp'

```r
as_gn(x, ..., spatstat = FALSE)
```

## S3 method for class 'lppfit'

```r
as_gn(x, ...)
```
Arguments

- **x**: An object that could reasonably be transmuted to an object of class `gn`.
- **...**: Other arguments.
- **spatstat**: Set to `TRUE` if retransformation to an object of any `spatstat` class is desired. Some elements of these objects (such as the window) are otherwise discarded when being transmuted to an object of class `gn`.

Value

An object of class `gn`.

Author(s)

Marc Schneble <marc.schneble@stat.uni-muenchen.de>

Examples

```r
library(spatstat.data)
G <- as_gn(simplenet)
summary(G)
```

---

**as_gnpp**

Transmute to Point Pattern on a Geometric Network

Description

`as_gnpp` transmutes an existing object into a point pattern on a geometric network, an object of class `gnpp`.

Usage

```r
as_gnpp(x, ...)
```

### S3 method for class 'gnppfit'

```r
as_gnpp(x, ...)
```

### S3 method for class 'lpp'

```r
as_gnpp(x, ..., spatstat = FALSE)
```

Arguments

- **x**: An object that could reasonably be transmuted to an object of class `gnpp`.
- **...**: Other arguments.
- **spatstat**: Set to `TRUE` if retransformation to an object of any `spatstat` class is desired. Some elements of these objects (such as the window) are otherwise discarded when being transmuted to an object of class `gn`.
as_lpp

Value
An object of class gnpp.

Author(s)
Marc Schneble <marc.schneble@stat.uni-muenchen.de>

Examples

```r
library(spatstat.data)
X <- as_gnpp(chicago)
summary(X)
```

Transmute to Point Pattern on a Linear Network

Description

as_lpp transmutes an existing object into a point pattern on a geometric network, an object of class lpp.

Usage

```r
as_lpp(x, ...)
```

## S3 method for class 'gnpp'
as_lpp(x, ...)

Arguments

x An object of class gnpp.
...
Other arguments.

Value

A point pattern on a linear network, an object of class lpp.

Author(s)

Marc Schneble <marc.schneble@stat.uni-muenchen.de>
Examples

library(spatstat.data)
library(spatstat.linnet)

x <- as_lpp(montgomery)
plot(x)

L <- chicago
X <- as_gnpp(chicago, spatstat = TRUE)
x <- as_lpp(X)
# TRUE
all.equal(x, L)

bin_data

Bin Point Pattern on a Geometric Network

Description

bin_data bins the data on the supplied point pattern according to all possible combination of covariates.

Usage

bin_data(X, bins = NULL, vars = NULL, vars_internal = NULL, scale = NULL)

Arguments

X Point pattern on a geometric network (object of class gnpp)
bins A list containing the bins of the geometric network.
vars A character vector containing the name of all covariates in the model.
vars_Internal A character vector containing the name of all interval covariates in the model.
scale A named list which specifies the rescaling of network related covariates. Currently, internal covariates "x", "y", and "dist2V" can be scaled.

Value

The binned data.

Author(s)

Marc Schneble <marc.schneble@stat.uni-muenchen.de>
bspline_design

**Description**

bspline_design constructs the design matrix which represents the (log-)baseline intensity on the geometric network.

**Usage**

```r
design_matrix <- bspline_design(G, knots, bins)
```

**Arguments**

- `G`: An object of class `gn`.
- `knots`: A list which contains the knots on which the B-splines are defined.
- `bins`: List of which contains the mid points of the bins.

**Value**

A sparse matrix design matrix of dimension N x J.

**Author(s)**

Marc Schneble <marc.schneble@stat.uni-muenchen.de>

---

bspline_design_plot

**Description**

B-Spline Design Matrix for Plotting

**Usage**

```r
design_matrix <- bspline_design_plot(X, df)
```

**Arguments**

- `X`: A point pattern on a geometric network (object of class `gnpp`).
- `df`: A data frame with points at which the fitted intensity should be plotted.

**Value**

A sparse design matrix.
**Author(s)**

Marc Schneble <marc.schneble@stat.uni-muenchen.de>

---

**confidence_band**  
*Confidence Bands of Smooth Terms*

**Description**

`confidence_band` computes the lower and upper limits of smooth terms fitted with `intensity_pspline`.

**Usage**

```r
confidence_band(theta, V, X, q = 0.05, R = 1000)
```

**Arguments**

- `theta`: The estimated coefficients which corresponds to the smooth term.
- `V`: The covariance matrix of the estimated coefficients `theta`.
- `X`: The design matrix of the model which corresponds to the smooth term.
- `q`: The quantile. Default to `q = 0.05` which corresponds to 95% confidence bands.
- `R`: The number of replications in the simulation process.

**Value**

A list of two vectors which contain the lower and the upper limits of the confidence band.

---

**delta_h_global**  
*Computes a Global Knot Distance from the Input*

**Description**

`delta_h_global` computes a Global Knot Distance from the Input.

**Usage**

```r
delta_h_global(G, delta = NULL, h = NULL)
```
Arguments

G An object of class gn) or a point pattern on a geometric network (object of class gnpp).

delta A numeric vector of length one which already defines the global knot distance. Alternatively, delta can be supplied in terms of a quantile of the curve lengths of the network, i.e. a number in the unit interval. In the latter case, delta must be supplied as a character vector of length one, see the examples. By default, delta is chosen to be half of the minimal curve length.

h A numeric vector of length one which already defines the global knot distance h. Alternatively, h can be supplied in terms of a fraction of delta, i.e. a number in the interval (0,1]. In the latter case, h must be supplied as character vector of length one, see the examples. By default, h is chosen to be half of the global knot distance.

Value

The global knot distance delta.

Author(s)

Marc Schneble <marc.schneble@stat.uni-muenchen.de>

Examples

G <- as_gn(montgomery)
# use default arguments
setup <- delta_h_global(G)
setup
# set numeric value for delta and fraction for h
setup <- delta_h_global(G, delta = 0.1, h = "0.25")
setup
# set quantile for delta
setup <- delta_h_global(G, delta = "0.05")
setup

fit_poisson_model

Fit a Penalized Spline Poisson Model on a Geometric Network

Description

fit_poisson_model is called from intensity_pspline and performs the iterative algorithm to estimate the model parameters and the smoothing parameters rho in the penalized Poisson model.

Usage

fit_poisson_model(data, Z, K, ind, verbose = FALSE, control = list())
Arguments

data  The binned data.
Z     The (sparse) model matrix where the number of columns must correspond to the
      length of the vector of model coefficients theta.
K     A (sparse) square penalty matrix of with the same dimension as theta.
ind   A list which contains the indices belonging to each smooth term and the linear
      terms.
verbose If TRUE, prints information on the process of the fitting algorithm.
control A list of optional arguments which control the convergence of the fitting algo-
         rithm. See "Details".

Details

Smoothing parameters are estimated using the generalized Fellner-Schall method (Wood and Fasi-
olo, 2017).

Value

Model fit.

Author(s)

Marc Schneble <marc.schneble@stat.uni-muenchen.de>

References

optimization with application to Tweedie location, scale and shape models. Biometrics 73 1071-
1081.

Description

incidence constructs the incidence matrix of a geometric network from the vertices and the curve
segments.

Usage

incidence(vertices, lins)

Arguments

vertices  A data frame containing the vertices of the geometric network.
lins      A data frame containing the curve segments of the geometric network.
intensity_kernel

Value

The incidence matrix of dimension $W$ by $M$.

Author(s)

Marc Schneble <marc.schneble@stat.uni-muenchen.de>

Description

Intensity Estimation on Geometric Networks based on Kernel Smoothing

Usage

intensity_kernel(X, kernel = "heat")

Arguments

X A point pattern on a geometric network (object of class gnpp).

kernel If kernel = "heat", a heat kernel is used. If kernel = "Euclidean", a two-dimensional kernel smoother is used.

Value

A fitted point process on a linear network, an object of class lppfit.

Examples

X <- runifgn(n = 50, G = small_gn)
fit <- intensity_kernel(X)
plot(fit)
Description

This is the main function of the geonet package. `intensity_pspline` estimates the intensity of a point pattern on a geometric network employing penalized splines as outlined in Schneble and Kauermann (2020). In distinction to `density.lpp` from the spatstat.linnet package, which provides kernel based tools for intensity estimation of point patterns on linear networks, `intensity_pspline` allows to incorporate covariates while also estimating the baseline intensity. Covariates can be either internal or external. External covariates can also be incorporated as a smooth term using penalized splines with the same syntax as in `gam`.

Usage

```r
intensity_pspline(
  X,
  ..., 
  formula = ~1,
  delta = "0",
  h = "0.5",
  r = 2,
  scale = NULL,
  density = FALSE,
  verbose = FALSE,
  control = list()
)
```

Arguments

- `X` A point pattern on a geometric network (object of class `gnpp`). The data (X$data) must contain information on all covariates included in `formula`.
- `...` Other arguments. The following arguments must match exactly.
- `formula` A one-sided formula (if a two-sided formula is supplied, the left hand side of the formula is ignored). The formula can consist of either linear terms as in linear models (`lm`) or smooth terms as in `gam` formulae, where the usage is restricted to smooth terms constructed with `s` and the argument `bs` is set to `bs = "ps"` by default, i.e. `intensity_pspline` can handle penalized spline based smooth terms.
- `delta` The global knot distance δ, a numerical vector of length one. If not supplied, delta will be chosen properly according to the geometric network `X` which is supplied.
- `h` The global bin width h, a numerical vector of length one. If not supplied, h will be chosen properly according to the geometric network `X` which is supplied.
- `r` The order of the penalty of the baseline intensity on the geometric network, default to a penalty of order r = 2.
**internal**

A named list which specifies the rescaling of network related covariates. Currently, only x- and y-coordinates can be scaled.

**density**

TRUE if the intensity should be normalized such that it can be interpreted as a density, i.e. the integral over the estimated density is equal to one.

**verbose**

If TRUE, prints information on the process of the fitting algorithm.

**control**

A list of optional arguments which control the convergence of the fitting algorithm. See "Details".

**Value**

A fitted geometric network (object of class gnppfit).

**Author(s)**

Marc Schneble <marc.schneble@stat.uni-muenchen.de>

**References**


**Examples**

```r
library(geonet)
X <- runifgn(50, small_gn)
delta <- 0.2
model <- intensity_pspline(X, delta = delta)
summary(model)
plot(model)
```

---

**internal**

**Internal Covariates**

**Description**

internal computes the values of internal covariates at the midpoints of the bins of the network. Internal covariates can either be supplied via the point pattern or they are a function of the network. Currently, x- and y- coordinates are supported for the latter.

**Usage**

```r
internal(vars, X, bins, scale)
```
Arguments

vars  The name of the covariates which should go into the model as linear internal covariates.

X    Point pattern on a geometric network (object of class gnpp)

bins A list containing the bins of the geometric network.

scale A named list which specifies the rescaling of network related covariates. Currently, only x- and y-coordinates can be scaled.

Value

A data frame with the number of rows equal to the number of bins of the geometric network (sum(bins$N)) and the number of columns equal to the length of vars.

Author(s)

Marc Schneble <marc.schneble@stat.uni-muenchen.de>

Description

A point pattern which represents 18,263 car crashes on a network of highways in Montgomery county, Maryland. The temporal dimension is represented through the covariate "hour".

Usage

data(montgomery)

Format

montgomery is an object of class "gnpp".

Source

Created by Marc Schneble <marc.schneble@lmu.de>.
network_bins  

**Defining bins on a Geometric Network**

**Description**

network_bins subdivides each curve segment into several bins.

**Usage**

```r
network_bins(G, h = NULL)
```

**Arguments**

- **G**: An object of class `gn` or a point pattern on a geometric network (object of class `gnpp`).
- **h**: The global bin width h.

**Value**

A list which contains the bins of every curve of the geometric network.

**Author(s)**

Marc Schneble <marc.schneble@stat.uni-muenchen.de>

network_integral  

**Integral of a fitted intensity**

**Description**

Integral of a fitted intensity

**Usage**

```r
network_integral(fit)
```

**Arguments**

- **fit**: A fitted point process on a geometric network.

**Value**

A numeric vector of the length one, the integral.
network_intensity  
*Fitted Intensity on a Geometric Network*

**Description**

Fitted Intensity on a Geometric Network

**Usage**

```r
network_intensity(z, m, fit1, fit2 = NULL, scale = NULL)
```

**Arguments**

- `z`  
  The shortest path distance from the beginning of the network segment.

- `m`  
  The network segment index.

- `fit1`  
  A fitted geometric network.

- `fit2`  
  A second fitted geometric network. If specified, the function returns the squared difference of the intensity fits at the specified point of the network.

- `scale`  
  A numeric vector of length two which determines the scaling of the two intensity functions.

**Value**

A numeric vector of length one, indicating the intensity (or the squared difference of two intensities) at the specified point.

---

network_ISE  
*Computation of the Integrated Squared Error*

**Description**

Computes the integrated squared error between a true intensity `fit1` and an estimate `fit2`.

**Usage**

```r
network_ISE(fit1, fit2)
```

**Arguments**

- `fit1`  
  The true intensity.

- `fit2`  
  The estimated intensity.

**Value**

The (normalized) integrated squared error.
**network_knots**

*Defining knots on a Geometric Network*

**Description**

`network_knots` defines knots on a geometric network (object of class `gn`) which can be used to construct linear B-splines on it.

**Usage**

```r
network_knots(G, delta)
```

**Arguments**

- **G**: An object of class `gn` or a point pattern on a geometric network (object of class `gnpp`).
- **delta**: The global knot distance `delta`.

**Value**

A list which contains the knot sequence of every curve of the geometric network.

**Author(s)**

Marc Schneble <marc.schneble@stat.uni-muenchen.de>

---

**network_location**

*Find Location of a Point on a Geometric Network*

**Description**

Find Location of a Point on a Geometric Network

**Usage**

```r
network_location(G, m, z)
```

**Arguments**

- **G**: A geometric network.
- **m**: The segment index.
- **z**: The shortest path distance from the beginning of the network segment.

**Value**

A list with names `l`, `tp_l`, `x` and `y`. 
network_penalty

Penalty Matrix of a Geometric Network

Description

network_penalty constructs the penalty matrix which relates to the B-Splines created by bspline_design.

Usage

network_penalty(G, knots, r)

Arguments

G A geometric network (object of class gnpp).

knots A list which contains the knots on which the B-splines are defined

r The order of the penalty, default to first-order penalty (r = 1).

Value

A sparse and square penalty matrix.

Author(s)

Marc Schneble <marc.schneble@stat.uni-muenchen.de>

plot.gn

Plot Methods for Geometric Network related Objects

Description

plot method for geometric networks, point patterns on geometric networks, or a fitted point process.

Usage

## S3 method for class 'gn'
plot(x, ..., title = list(), size = list(), color = list(), frame = FALSE)

## S3 method for class 'gnpp'
plot(
  x,
  ...,
  title = list(),
  size = list(),
  color = list(),
  shape = 1,
## S3 method for class 'gnppfit'
```r
plot(
  x,
  ...,
  title = list(),
  size = list(),
  color = list(),
  shape = 1,
  frame = FALSE,
  data = FALSE,
  trans = "identity",
  select = NULL,
  sol = 100
)
```

## S3 method for class 'lppfit'
```r
plot(
  x,
  ...,
  title = list(),
  size = list(),
  color = list(),
  shape = 1,
  frame = FALSE,
  data = FALSE,
  trans = "identity",
  sol = 100
)
```

### Arguments

- **x**
  An object which is related to a geometric network (object of class `gn`, `gnpp` or `gnppfit`).

- **...**
  Other arguments.

- **title**
  A named list with names "x", "y" and "plot" which specify the arguments x, y and title of the `labs` function. Each list entry must be a character vector which has length equal to the number of plots. The list entries can remain unspecified in which case the respective titles are left blank.

- **size**
  A named list with names "lines" and "points" which specify the size argument of `geom_segment` and `geom_point`, respectively. Each list entry must be a numeric vector of length one with positive values. If the whole list or one entry remains unspecified, default values are used.

- **color**
  A named list with names "lines" and "points" which specify the color argument of `geom_segment` and `geom_point`. Each list entry must specify a valid color.
By default, lines and points are plotted in black.

frame
If set to TRUE, draws a frame around the network and adds tick marks and axis labeling.

shape
The shape used for plotting the points. An integer between 0 and 25. Default to shape = 1 which shows the points as a circle.

covariate
Character vector of length one which is name of the covariate to be plotted. Must be an external categorical covariate with at most ten different values.

data
Set to TRUE if the data shall be plotted on top of the fitted intensity.

trans
The transformation applied to the color bar of the intensity fit. Specifies the trans argument of scale_color_gradient.

select
Allows the plot for a single model term to be selected for printing. e.g. if just the plot for the second smooth should be printed to the console, set select = 2.

sol
Solution of the color network plot.

Value
Invisibly returns an object of class ggplot or a list of ggplot objects.

---

print.gn

Print Method for Geometric Networks

Description
Prints basic information of a geometric network related object.

Usage

## S3 method for class 'gn'
print(x, ...)

## S3 method for class 'gnpp'
print(x, ...)

## S3 method for class 'gnppfit'
print(x, ...)

Arguments

x
A geometric network (object of class gn), a point pattern on a geometric network (object of class gnpp) or a fitted point process on a geometric network (object of class gnppfit).

...
Other arguments.

Value
Invisibly returns the NULL object.
**print.summary.gn**  
*Print Method for Summaries*

**Description**
Prints basic information of a geonet summary object.

**Usage**
```r
## S3 method for class 'summary.gn'
print(x, ...)

## S3 method for class 'summary.gnpp'
print(x, ...)

## S3 method for class 'summary.gnppfit'
print(x, ...)
```

**Arguments**
- `x` A geonet summary object of class `summary.gn`, `codesummary.gnpp` or `summary.gnppfit`.
- `...` Other arguments.

**Value**
Invisibly returns the NULL object.

---

**rgnpp**  
*Random Points on a Geometric Network*

**Description**
Random Points on a Geometric Network

**Usage**
```r
rgnpp(n, fit)
```

**Arguments**
- `n` Number of random points. A nonnegative integer.
- `fit` A fitted point process on a geometric network (object of class `gnpp`).

**Value**
A point pattern on a geometric network, an object of class `gnpp`. 
Examples

```r
X <- runifgn(n = 50, G = small_gn)
fit <- intensity_pspline(X)
X2 <- rgnpp(n = 50, fit = fit)
plot(X2)
```

---

**runifgn**

*Simulate Uniform Points on a Geometric Network*

**Description**

`runifgn` simulates points on a geometric network according to a uniform density.

**Usage**

```r
runifgn(n, G)
```

**Arguments**

- `n`: Number of random points to generate. A nonnegative integer.
- `G`: A geometric network (object of class `gn`).

**Value**

A point pattern on a geometric network, an object of class `gnpp`.

**Author(s)**

Marc Schneble <marc.schneble@stat.uni-muenchen.de>

---

**scoring**

*Maximum-Likelihood Estimation*

**Description**

Scoring algorithm for maximum-likelihood estimation of a penalized Poisson model while treating the smoothing parameters as fixed. Since the model matrix $Z$ when fitting a point process model on a geometric network is very large with usually several millions of entries, `scoring` builds an sparse representations of matrices in `R`.

**Usage**

```r
scoring(theta, rho, data, Z, K, ind, eps_theta = 1e-05)

score(theta, rho, data, Z, K, ind)

fisher(theta, rho, data, Z, K, ind)
```
Arguments

- **theta**: An initial vector of model coefficients.
- **rho**: The current vector of smoothing parameters. For each smooth term, including the baseline intensity of the network, one smoothing parameter must be supplied.
- **data**: A data frame containing the data.
- **Z**: The (sparse) model matrix where the number of column must correspond to the length of the vector of model coefficients \( \theta \).
- **K**: A (sparse) square penalty matrix of with the same dimension as \( \theta \).
- **ind**: A list which contains the indices belonging to each smooth term and the linear terms.
- **eps_theta**: The termination condition. If the relative change of the norm of the model parameters is less than \( \text{eps}_\theta \), the scoring algorithm terminates and returns the current vector of model parameters.

Details

scoring performs the scoring algorithm for maximum-likelihood estimation according to Fahrmeir et al. (2013). This algorithm is based on the score-function and the Fisher-information of the log-likelihood. score returns the score-function (the gradient of the log-likelihood) and fisher returns the Fisher-information (negative Hessian of the log-likelihood).

Value

The maximum likelihood estimate for fixed smoothing parameters.

References


---

**small_gn**

*Simple example of a Geometric Network*

Description

A simple, artificially created, example of a geometric network.

Usage

```
data(small_gn)
```

Format

*small_gn* is an object of class "gn".
summary.gn

Summary for a (point pattern on a) geometric network

Description

Takes an object of class gn or gnpp and computes a summary from it.

Usage

## S3 method for class 'gn'
summary(object, ...)

## S3 method for class 'gnpp'
summary(object, ...)

Arguments

object A geometric network (object of class gn or a point pattern on a geometric network (object of class gnpp).

... Other arguments.

Value

A summary of a (point pattern on a) geometric network. This is an object of class summary.gn or summary.gnpp, respectively.

Author(s)

Marc Schneble <marc.schneble@stat.uni-muenchen.de>

summary.gnppfit

Summary for fitted point process on a geometric network

Description

Takes a fitted gnppfit object produced by intensity_spline and computes a summary from it.

Usage

## S3 method for class 'gnppfit'
summary(object, ...)

Source

Created by Marc Schneble <marc.schneble@lmu.de>.
Arguments

- **object**: A fitted point process on a geometric network.
- **...**: Other arguments.

Value

A summary of a fitted point process on a geometric network. This is an object of class `summary.gnppfit`.

Author(s)

Marc Schneble <marc.schneble@stat.uni-muenchen.de>
Index

* datasets
  - montgomery, 14
  - small_gn, 23
  - as.linnet, 2
  - as.linnet.gn, 2
  - as_gn, 3
  - as_gnpp, 4
  - as_lpp, 5
  - bin_data, 6
  - bspline_design, 7, 18
  - bspline_design_plot, 7
  - confidence_band, 8
  - delta_h_global, 8
  - density.lpp, 12
  - fisher (scoring), 22
  - fit_poisson_model, 9
  - gam, 12
  - geom_point, 19
  - geom_segment, 19
  - incidence, 10
  - intensity_kernel, 11
  - intensity_pspline, 9, 12
  - internal, 13
  - labs, 19
  - linnet, 2
  - lm, 12
  - montgomery, 14
  - network_bins, 15
  - network_integral, 15
  - network_intensity, 16
  - network_ISE, 16
  - network_knots, 17
  - network_location, 17
  - network_penalty, 18
  - plot.gn, 18
  - plot.gnpp (plot.gn), 18
  - plot.gnppfit (plot.gn), 18
  - plot.lppfit (plot.gn), 18
  - print.gn, 20
  - print.gnpp (print.gn), 20
  - print.gnppfit (print.gn), 20
  - print.summary.gn, 21
  - print.summary.gnpp (print.summary.gn), 21
  - print.summary.gnppfit (print.summary.gn), 21
  - rgnpp, 21
  - runifgn, 22
  - s, 12
  - scale_color_gradient, 20
  - score (scoring), 22
  - scoring, 22
  - small_gn, 23
  - summary.gn, 24
  - summary.gnpp (summary.gn), 24
  - summary.gnppfit, 24

26