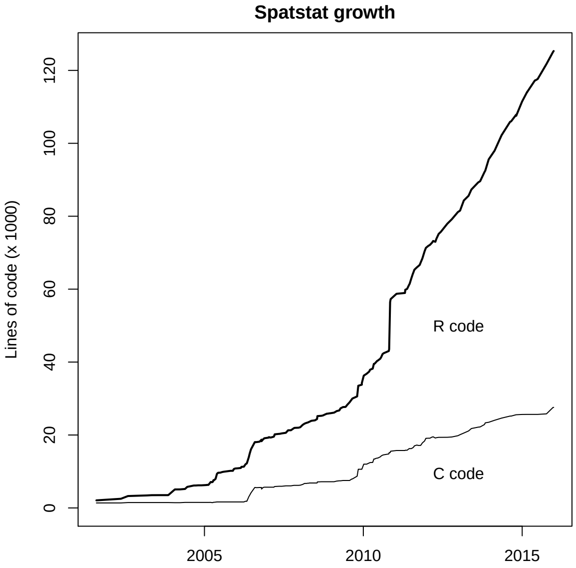
Summary of recent updates to `spatstat`

Adrian Baddeley, Rolf Turner and Ege Rubak

For `spatstat` version `1.44-1`

This is a summary of changes that have been made to the `spatstat` package since the publication of the accompanying book [2]. The book, published in December 2015, covers everything in `spatstat` up to version `1.42-0`, released in May 2015.

The current version of `spatstat` is `1.44-1`. It contains 163 new functions and 1 new dataset. This document summarises the most important changes. It also lists all important bugs detected since 2010.



Contents

1 Precis of all changes

Here is the text from the ‘overview’ sections of the News and Release Notes for each update.

- `spatstat` now depends on the package `nlme`.
- More support for replicated point patterns.
- More support for tessellations.
- More support for multidimensional point patterns and point processes.
- More options for one-sided envelopes.
- More support for model comparison.
- Convexifying operation.
- Subdivide a linear network.
- Penttinen process can be simulated (by Metropolis-Hastings or CFTP).
- Calculate the predicted variance of number of points.
- Accelerated algorithms for linear networks.
- Quadrat counting accelerated, in some cases.
- Simulation algorithms have been accelerated; simulation outcomes are *not* identical to those obtained from previous versions of `spatstat`.
- Determinantal point process models.
- Random-effects and mixed-effects models for replicated patterns.
- Dao-Genton test, and corresponding simulation envelopes.
- Simulated annealing and simulated tempering.
- spatstat colour tools now handle transparent colours.
- Improvements to `[` and `subset` methods
- Extensions to kernel smoothing on a linear network.
- Support for one-dimensional smoothing kernels.
- Mark correlation function may include weights.
- Cross-correlation version of the mark correlation function.
- Penttinen pairwise interaction model.
- Improvements to simulation of Neyman-Scott processes.
- Improvements to fitting of Neyman-Scott models.
- Extended functionality for pixel images.

- Fitted intensity on linear network
- Triangulation of windows.
- Corrected an edge correction.

2 New datasets

The following datasets have been added to the package.

- `austates`: The states and large mainland territories of Australia represented as polygonal regions forming a tessellation.

3 New Functions

Following is a list of all the functions that have been added.

- `as.function.tess`: Convert a tessellation to a `function(x,y)`. The function value indicates which tile of the tessellation contains the point (x, y).
- `tileindex`: Determine which tile of a tessellation contains a given point (x, y).
- `persp.leverage.ppm`: Method for persp plots for objects of class `leverage.ppm`
- `AIC.mppm`, `extractAIC.mppm`: AIC for point process models fitted to replicated point patterns.
- `nobs.mppm`, `terms.mppm`, `getCall.mppm`: Methods for point process models fitted to replicated point patterns.
- `rPenttinen`: Simulate the Penttinen process using perfect simulation.
- `varcount`: Given a point process model, compute the predicted variance of the number of points falling in a window.
- `inside.boxx`: Test whether multidimensional points lie inside a specified multidimensional box.
- `lixellate`: Divide each segment of a linear network into smaller segments.
- `nsegments.linnet`, `nsegments.lpp`: Count the number of line segments in a linear network.
- `grow.boxx`: Expand a multidimensional box.
- `deviance.ppm`, `deviance.lppm`: Deviance for a fitted point process model.
- `pseudoR2`: Pseudo-R-squared for a fitted point process model.
- `tiles.empty` Checks whether each tile of a tessellation is empty or nonempty.
- `summary.linim`: Summary for a pixel image on a linear network.
- Determinantal Point Process models:
  - `dppm`: Fit a determinantal point process model.
  - `fitted.dppm`, `predict.dppm`, `intensity.dppm`: prediction for a fitted determinantal point process model.

  - `Kmodel.dppm`, `pcfmodel.dppm`: Second moments of a determinantal point process model.
  - `rdpp`, `simulate.dppm`: Simulation of a determinantal point process model.
  - `logLik.dppm`, `AIC.dppm`, `extractAIC.dppm`, `nobs.dppm`: Likelihood and AIC for a fitted determinantal point process model.
  - `print.dppm`, `reach.dppm`, `valid.dppm`: Basic information about a `dpp` model.
  - `coef.dppm`, `formula.dppm`, `print.dppm`, `terms.dppm`, `labels.dppm`, `model.frame.dppm`, `model.matrix.dppm`, `model.images.dppm`, `is.stationary.dppm`, `reach.dppm`, `unitname.dppm`, `unitname<-.dppm`, `Window.dppm`: Various methods for `dppm` objects.
  - `parameters.dppm`: Extract meaningful list of model parameters.
  - `objsurf.dppm`: Objective function surface of a `dppm` object.
  - `residuals.dppm`: Residual measure for a `dppm` object.

- Determinantal Point Process model families:

  - `dppBessel`, `dppCauchy`, `dppGauss`, `dppMatern`, `dppPowerExp`: Determinantal Point Process family functions.
  - `detpointprocfamilyfun`: Create a family function.
  - `update.detpointprocfamily`: Set parameter values in a determinantal point process model family.
  - `simulate.dppm`: Simulation.
  - `is.stationary.detpointprocfamily`, `intensity.detpointprocfamily`, `Kmodel.detpointprocfamily`, `pcfmodel.detpointprocfamily`: Moments.
  - `dim.detpointprocfamily`, `dppapproxkernel`, `dppapproxpcf`, `dppeigen`, `dppkernel`, `dppparbounds`, `dppspecdenrange`, `dppspecden`: Helper functions.

- `dg.envelope`: Simulation envelopes corresponding to Dao-Genton test.

- `dg.progress`: Progress plot (envelope representation) for the Dao-Genton test.

- `dg.sigtrace`: significance trace for the Dao-Genton test.

- `markcrosscorr`: Mark cross-correlation function for point patterns with several columns of marks.

- `rtemper`: Simulated annealing or simulated tempering.

- `rgb2hsva`: Convert RGB to HSV data, like `rgb2hsv`, but preserving transparency.

- `superimpose.ppplist`, `superimpose.splitppp`: New methods for 'superimpose' for lists of point patterns.

- `dkernel`, `pkernel`, `qkernel`, `rkernel`: Probability density, cumulative probability, quantiles and random generation from distributions used in basic one-dimensional kernel smoothing.

- `kernel.factor`: Auxiliary calculations for one-dimensional kernel smoothing.

- `spatdim`: Spatial dimension of any object in the `spatstat` package.

- `as.boxx`: Convert data to a multi-dimensional box.

- intensity.ppx: Method for intensity for multi-dimensional space-time point patterns.
- fourierbasis: Evaluate Fourier basis functions in any number of dimensions.
- valid: New generic function, with methods valid.ppm, valid.lppm, valid.dppm.
- emend, emend.ppm, emend.lppm: New generic function with methods for ppm and lppm. emend.ppm is equivalent to project.ppm.
- Penttinen: New pairwise interaction model.
- quantile.density: Calculates quantiles from kernel density estimates.
- CDF.density: Calculates cumulative distribution function from kernel density estimates.
- triangulate.owin: decompose a spatial window into triangles.
- fitted.lppm: fitted intensity values for a point process on a linear network.
- parameters: Extract all parameters from a fitted model.

4 Alphabetical list of changes

Here is a list of all changes made to existing functions, listed alphabetically.

- affine.owin: Allows transformation matrix to be singular, if the window is polygonal.
- anova.mppm: Now handles Gibbs models, and performs the adjusted composite likelihood ratio test.
- as.im.function: New argument strict.
- as.polygonal: Can now repair errors in polygon data, if repair=TRUE.
- bw.ppl: New argument weights.
- clusterset: Improved behaviour.
- clusterfit: New argument algorithm specifies the choice of optimisation algorithm.
- collapse.fv: This is now treated as a method for the nlme generic collapse. Its syntax has been adjusted slightly.
- contour.im: New argument col specifies the colour of the contour lines. If col is a colour map, then the contours are drawn in different colours.
- dclf.test, mad.test, dclf.progress, mad.progress, dclf.sigtrace, mad.sigtrace, dg.progress, dg.sigtrace:
  - New argument clamp determines the test statistic for one-sided tests.
  - New argument rmin determines the left endpoint of the test interval.
  - New argument leaveout specifies how to calculate discrepancy between observed and simulated function values.
  - New argument scale allows summary function values to be rescaled before the comparison is performed.

  - New argument `interpolate` supports interpolation of p-value.
  - New argument `interpolate` supports interpolation of critical value of test.

- `default.rmhcontrol, default.rmhexpand`: New argument `w`.

- `density.lpp`:

  - New argument `kernel` specifies the smoothing kernel. Any of the standard one-dimensional smoothing kernels can be used.
  - Now supports both the 'equal-split continuous' and 'equal-split discontinuous' smoothers. New argument `continuous` determines the choice of smoother.

- `diagnose.ppm, plot.diagppm`: New arguments `col.neg, col.smooth` control the colour maps.

- `envelope`:

  - New argument `clamp` gives greater control over one-sided envelopes.
  - New argument `funargs`
  - New argument `scale` allows global envelopes to have width proportional to a specified function of r, rather than constant width.

- `Kest.fft`: Now has `...` arguments allowing control of spatial resolution.

- `kppm`:

  - Fitting a model with `clusters="LGCP"` no longer requires the package `RandomFields` to be loaded explicitly.
  - New argument `algorithm` specifies the choice of optimisation algorithm.

- `kppm`: Left hand side of formula can now involve entries in the list `data`.

- `Hest`: Argument `X` can now be a pixel image with logical values. New argument `W`. [Based on code by Kassel Hingee.]

- `logLik.ppm`: New argument `absolute`.

- `logLik.mppm`: new argument `warn`.

- `lppm`: Computation accelerated.

- `markcorr`: New argument `weights` allows computation of the weighted version of the mark correlation function.

- `mppm`: Now handles models with a random effect component. New argument `random` is a formula specifying the random effect.

- `nndist.lpp, nnwhich.lpp, nncross.lpp, distfun.lpp`: New argument `k` allows computation of k-th nearest point. Computation accelerated.

- `padimage`: New argument `W` allows an image to be padded out to fill any window.

- `plot.kppm`: New arguments `pause` and `xname`.

- `plot.mppm` New argument `se`.

- `plot.ppp` The default colour for the points is now a transparent grey, if this is supported by the plot device.

- `ppm.ppp, ppm.quad` New argument `emend`, equivalent to `project`.

- `predict.kppm, residuals.kppm` Now issues a warning when the calculation ignores the cluster/Cox component and treats the model as if it were Poisson. (This currently happens in predict.kppm when se=TRUE or interval != "none", and in residuals.kppm when type != "raw").

- `progressreport` New argument `state` New option: `style="tk"`

- `quadratcount.ppp`: Computation accelerated in some cases.

- `rgbim, hsvim` New argument `A` controls the alpha (transparency) channel.

- `rgb2hex, col2hex, paletteindex, is.colour, samecolour, complementarycolour, is.grey, to.grey` These colour tools now handle transparent colours.

- `rgb2hex` New argument `maxColorValue`

- `rLGCP` This function no longer requires the package `RandomFields` to be loaded explicitly.

- `rMaternI, rMaternII`: These functions can now generate random patterns in three dimensions and higher dimensions, when the argument `win` is of class `box3` or `boxx`.

- `rmh.ppm, rmhmodel.ppm, simulate.ppm`: A model fitted using the `Penttinen` interaction can now be simulated.

- `rmh.default, rmhmodel.default`: These functions now recognise `cif='penttinen'` for the Penttinen interaction.

- `rose.default` New argument `weights`.

- `rose` New arguments `start` and `clockwise` specify the convention for measuring and plotting angles.

- `rotmean`: New argument `padzero`. Default behaviour has changed.

- `rpoispp` Accelerated, when `lambda` is a pixel image.

- `rStrauss, rHardcore, rStraussHard, rDiggleGratton, rDGS, rPenttinen:` New argument `drop`.

- `rthin` Accelerated, when `P` is a single number.

- `rThomas, rMatClust, rCauchy, rVarGamma` When the model is approximately Poisson, it is simulated using rpoispp. This avoids computations which would require huge amounts of memory. New argument `poisthresh` controls this behaviour.

- Simulation: Several basic simulation algorithms have been accelerated. Consequently, simulation outcomes are not identical to those obtained with previous versions of `spatstat`, even when the same random seed is used. To ensure compatibility with previous versions of spatstat, revert to the slower code by setting `spatstat.options(fastthin=FALSE, fastpois=FALSE)`.

- `simulate.ppm` New argument `w` controls the window of the simulated patterns.

- spatstat.options New options fastthin and fastpois enable fast simulation algorithms. Set these options to FALSE to reproduce results obtained with previous versions of spatstat.

- split.ppp The splitting variable f can now be a logical vector.

- step: now works for models of class "mppm".

- tess: Argument window is ignored when xgrid, ygrid are given.

- textureplot: Argument x can now be something acceptable to as.im.

- to.grey New argument transparent.

- union.owin: Improved behaviour when there are more than 2 windows.

- update: now works for models of class "mppm".

- update.kppm: Now handles additional arguments in any order, with or without names. Changed arguments. Improved behaviour.

- valid.ppm This is now a method for the generic function valid.

- vcov.mppm: Now handles models with Gibbs interactions.

- [<-.im Accepts an array for value.

- [.ppx: The subset index i may now be a spatial domain of class boxx or box3.

- [.ppp New argument clip determines whether the window is clipped.

- [.ppp The previously-unused argument drop now determines whether to remove unused levels of a factor.

- [.pp3, [.lpp, [.ppx, subset.ppp, subset.pp3, subset.lpp, subset.ppx: These methods now have an argument drop which determines whether to remove unused levels of a factor.

5 Serious Bugs Fixed

Hundreds of bugs have been detected and fixed in spatstat. Bugs that may have affected the user are listed in the package NEWS file. To read all these bug reports, type

```
> news(grepl("^BUG", Category), package="spatstat")
```

which currently produces a list of 512 bugs, of which 44 were detected after publication of the book [2].

Following is a list of the **most serious bugs** only, in order of potential impact.

- nncross.ppp:

  Results were completely incorrect if $k > 1$.
  (Bug introduced in spatstat 1.31-2, april 2013; fixed in spatstat 1.35-0, december 2013)

- nncross.pp3: Results were completely incorrect in some cases.
  (Bug introduced in spatstat 1.32-0, august 2013; fixed in spatstat 1.34-0, october 2013)

- rmh:

- Simulation was completely incorrect in the case of a multitype point process with an interaction that does not depend on the marks, such as `ppm(betacells, ~marks, Strauss(60))` due to a coding error in the `C` interface.
  (Bug introduced in `spatstat 1.22-3`, march 2010; fixed in `spatstat 1.22-3`, june 2011)
- Simulation of the Area-Interaction model was completely incorrect.
  (Bug introduced in `spatstat 1.23-6`, october 2011; fixed in `spatstat 1.31-0`, january 2013)
- Simulation of the Geyer saturation process was completely incorrect.
  (Bug introduced in `spatstat 1.31-0`, january 2013; fixed in `spatstat 1.31-1`, march 2013)
- Simulation of the Strauss-Hard Core process was partially incorrect, giving point patterns with a slightly lower intensity.
  (Bug introduced in `spatstat 1.31-0`, january 2013; fixed in `spatstat 1.37-0`, may 2014)
- The result of simulating a model with a hard core did not necessarily respect the hard core constraint, and simulation of a model with strong inhibition did not necessarily converge. This only happened if the first order trend was large, the starting state (`n.start` or `x.start`) was not given, and the number of iterations `nrep` was not very large. It occurred because of a poor choice for the default starting state. (Bug was present since about 2010. Fixed in `spatstat 1.40-0`, december 2014)
- Simulation was incorrect in the case of an inhomogeneous multitype model with `fixall=TRUE` (i.e. with a fixed number of points of each type) if the model was segregated (i.e. if different types of points had different first order trend). The effect of the error was that all types of points had the same first order trend. (Bug was present since about 2010. Fixed in `spatstat 1.43-0`, september 2015)
- Simulation of the Geyer saturation process was incorrectly initialised, so that the results of a short run (i.e. small value of `nrep`) were incorrect, while long runs were correct.
  (Bug introduced in `spatstat 1.17-0`, october 2009; fixed in `spatstat 1.31-1`, march 2013)

- `nncross`, `distfun`, `AreaInter`: Results of `nncross` were possibly incorrect when `X` and `Y` did not have the same window. This bug affected values of `distfun` and may also have affected ppm objects with interaction `AreaInter`.
  (Bug introduced in `spatstat 1.9-4`, june 2006; fixed in `spatstat 1.25-2`, january 2012)

- `update.ppm`: If the argument `Q` was given, the results were usually incorrect, or an error was generated.
  (Bug introduced in `spatstat 1.38-0`, august 2014; fixed in `spatstat 1.38-1`, august 2014)

- `envelope.ppm`: If the model was an inhomogeneous Poisson process, the resulting envelope object was incorrect (the simulations were correct, but the envelopes were calculated assuming the model was CSR).
  (Bug introduced in `spatstat 1.23-5`, september 2011; fixed in `spatstat 1.23-6`, october 2011)

- `leverage.ppm`, `influence.ppm`, `dfbetas.ppm`: Results were incorrect for non-Poisson processes.
  (Bug introduced in `spatstat 1.25-0`, december 2011; fixed in `spatstat 1.34-0`, october 2013)

- `rVarGamma`: Simulations were incorrect; they were generated using the wrong value of the parameter `nu.ker`.
  (Bug introduced in `spatstat 1.25-0`, december 2011; fixed in `spatstat 1.35-0`, december 2013)

- `rCauchy`: Simulations were incorrect; they were generated using the wrong value of the parameter `omega`.
  (Bug introduced in `spatstat 1.25-0`, december 2011; fixed in `spatstat 1.25-2`, january 2012)

- kppm, matclust.estpcf, pcfmodel: The pair correlation function of the Mátern Cluster Process was evaluated incorrectly at distances close to 0. This could have affected the fitted parameters in matclust.estpcf() or kppm(clusters="MatClust").
(Bug introduced in spatstat 1.20-2, august 2010; fixed in spatstat 1.33-0, september 2013)
- ppm: Results were incorrect for the Geyer saturation model with a non-integer value of the saturation parameter sat.
(Bug introduced in spatstat 1.20-0, july 2010; fixed in spatstat 1.31-2, april 2013)
- lppm: For multitype patterns, the fitted model was completely incorrect due to an error in constructing the quadrature scheme.
(Bug introduced in spatstat 1.23-0, july 2011; fixed in spatstat 1.30-0, december 2012)
- Geyer: For point process models with the Geyer interaction, vcov.ppm and suffstat sometimes gave incorrect answers.
(Bug introduced in spatstat 1.27-0, may 2012; fixed in spatstat 1.30-0, december 2012)
- vcov.ppm, suffstat: These functions sometimes gave incorrect values for marked point process models.
(Bug introduced in spatstat 1.27-0, may 2012; fixed in spatstat 1.29-0, october 2012)
- linearK, linearKinhom: If any data points were located exactly at a vertex of the linear network, the weights for Ang's correction were incorrect, due to numerical error. This sometimes produced infinite or NA values of the linear K function.
(Bug introduced in spatstat 1.23-0, july 2011; fixed in spatstat 1.27-0, may 2012)
- Kinhom, Linhom: the results were not renormalised (even if renormalise=TRUE) in some cases.
(Bug introduced in spatstat 1.21-0, december 2010; fixed in spatstat 1.37-0, may 2014)
- Kinhom, Linhom: Ignored argument reciplambda2 in some cases.
(Bug introduced in spatstat 1.39-0, october 2014; fixed in spatstat 1.40-0, december 2014)
- markcorrint: Results were completely incorrect.
(Bug introduced in spatstat 1.39-0, october 2014; fixed in spatstat 1.40-0, december 2014)
- Kinhom, Linhom: Calculations were incorrect if lambda was a fitted point process model.
(Bug introduced in spatstat 1.38-0, august 2014; fixed in spatstat 1.38-1, august 2014)
- predict.ppm: Calculation of the conditional intensity omitted the edge correction if correction='translate' or correction='periodic'.
(Bug introduced in spatstat 1.17-0, october 2009; fixed in spatstat 1.31-3, may 2013)
- varblock: Calculations were incorrect if more than one column of edge corrections was computed.
(Bug introduced in spatstat 1.21-1, november 2010; fixed in spatstat 1.39-0, october 2014)
- scan.test Results were sometimes incorrect due to numerical instability (a 'Gibbs phenomenon').
(Bug introduced in spatstat 1.24-1, october 2011; fixed in spatstat 1.26-1, april 2012)
- relrisk: When at="pixels", a small fraction of pixel values were sometimes wildly inaccurate, due to numerical errors. This affected the range of values in the result, and therefore the appearance of plots. (Bug fixed in spatstat 1.40-0, december 2014)

- `selfcrossing.psp`: y coordinate values were incorrect.
(Bug introduced in `spatstat 1.23-2`, august 2011; fixed in `spatstat 1.25-3`, february 2012)

- `predict.slrm`: Results of `predict(object, newdata)` were incorrect if the spatial domain of `newdata` was larger than the original domain.
(Bug introduced in `spatstat 1.21-0`, november 2010; fixed in `spatstat 1.25-3`, february 2012)

- `Lest`: The variance approximations (Lotwick-Silverman and Ripley) obtained with `var.approx=TRUE` were incorrect for `Lest` (although they were correct for `Kest`) due to a coding error.
(Bug introduced in `spatstat 1.24-1`, october 2011; fixed in `spatstat 1.24-2`, november 2011)

- `bw.diggle`: Bandwidth was too large by a factor of 2.
(Bug introduced in `spatstat 1.23-4`, september 2011; fixed in `spatstat 1.23-5`, september 2011)

- pair correlation functions (`pcf.ppp`, `pcfdot`, `pcfcross` etc:) The result had a negative bias at the maximum r value, because contributions to the pcf estimate from interpoint distances greater than `max(r)` were mistakenly omitted. (Bugs fixed in `spatstat 1.35-0`, december 2013)

- `Kest`, `Lest`: Gave incorrect values in very large datasets, due to numerical overflow. 'Very large' typically means about 1 million points in a random pattern, or 100,000 points in a tightly clustered pattern. [Overflow cannot occur unless there are at least 46,341 points.]

- `bw.relrisk`: Implementation of `method="weightedleastsquares"` was incorrect and was equivalent to `method="leastsquares"`.
(Bug introduced in `spatstat 1.21-0`, november 2010; fixed in `spatstat 1.23-4`, september 2011)

- `triangulate.owin`: Results were incorrect in some special cases.
(Bug introduced in `spatstat 1.42-2`, june 2015; fixed in `spatstat 1.44-0`, december 2015)

- `crosspairs`: If `X` and `Y` were identical point patterns, the result was not necessarily symmetric (on some machines) due to numerical artifacts.
(Bug introduced in `spatstat 1.35-0`, december 2013; fixed in `spatstat 1.44-0`, december 2015)

- `bdist.tiles`: Values were incorrect in some cases due to numerical error. (Bug fixed in `spatstat 1.29-0`, october 2012)

- `Kest.fft`: Result was incorrectly normalised.
(Bug introduced in `spatstat 1.21-2`, january 2011; fixed in `spatstat 1.44-0`, december 2015)

- `crossdist.ppp`: Ignored argument `squared` if `periodic=FALSE`. (Bug fixed in `spatstat 1.38-0`, july 2014)

- polygon geometry: The point-in-polygon test gave the wrong answer in some boundary cases.
(Bug fixed in `spatstat 1.23-2`, august 2011)

- `MultiStraussHard`: If a fitted model with `MultiStraussHard` interaction was invalid, `project.ppm` sometimes yielded a model that was still invalid. (Bug fixed in `spatstat 1.42-0`, may 2015)

- `pool.envelope`: Did not always respect the value of `use.theory`.
(Bug introduced in `spatstat 1.23-5`, september 2011; fixed in `spatstat 1.43-0`, september 2015)

- `nncross.lpp`, `nnwhich.lpp`, `distfun.lpp`: Sometimes caused a segmentation fault.
(Bug introduced in `spatstat 1.44-0`, december 2015; fixed in `spatstat 1.44-1`, december 2015)

- `anova.ppm`: If a single `object` was given, and it was a Gibbs model, then `adjust` was effectively set to `FALSE`.
  (Bug introduced in `spatstat 1.39-0`, october 2014; fixed in `spatstat 1.44-1`, december 2015)

References

[1] A. Baddeley. Analysing spatial point patterns in R. Technical report, CSIRO, 2010. Version 4. URL `https://research.csiro.au/software/r-workshop-notes/`

[2] A. Baddeley, E. Rubak, and R. Turner. *Spatial Point Patterns: Methodology and Applications with R*. Chapman & Hall/CRC Press, 2015.