

The PairPlot addon

Artur Tarassow and Allin Cottrell

October 29, 2026

This package produces grids of subplots depicting pairwise relationships in a dataset. This package maps each variable in a list onto a column and row in a grid of multiple axes.

It can also represent an additional level of conditionalization with the ‘factor’ argument (a series) which plots different subsets of data in different colors and/or point/circle types. This is used to resolve elements on a third dimension.

Please report bugs or comments on the gretl mailing list, report an issue on github (<https://github.com/atecon/PairPlot/issues>) or write to atecon@posteo.de.

1 Usage via the GUI

The function `PairPlot()` can be accessed through the GUI. The menu entry is “View->Graph specified vars->Pair Plot”. However, via scripting the user can control many more details.

2 Public function

```
PairPlot(const list L, const series factor[null], bundle opts[null])
```

Main function for creating pair-plots.

Parameters

- **L**: list, Series to plot against each other
- **factor**: series, Categorical (numeric or string-valued) variable for mapping plot aspects to different colors and/or points/ circles (optional, default: null). The factor series should have at most 16 distinct values. This series must be of type ‘discrete’. To set a series to discrete, use the command `setinfo <series name> --discrete`.
- **opts**: bundle, Optional for passing additional parameters. See below for explanation (default: null)

Returns

An integer of value ‘0’ (FALSE) if no error occurs, otherwise 1 (TRUE).

Options passed to the `PairPlot()` function by means of the ‘opts’ bundle

The user can control the following aspects by adding the respective parameter to the ‘opts’ bundle.

- **centroid**: string, Draw a symbol highlighting the centroid of the 2-dimensional data. Possible options are `mean`, `median` and `none` (default: “none”). If a **factor** series is passed, the centroid for each distinct factor is shown.
- **centroid_label**: bool, Show the label of the centroid statistics if TRUE, otherwise do not show if FALSE (default: FALSE).

- **centroid_linewidth**: scalar, Width of the line (default: 2).
- **centroid_pointsize**: scalar, Size of the point (default: 2).
- **filename**: string, Path plus filename plus file type. (optional, default: **display** which draws the resulting plot directly at the screen). Currently figures can only be stored in **png** format.
- **fontsize**: int, Control the font size of the labels (default: 16) .
- **grid**: bool, Draw a grid in the background if TRUE (=1) (default: FALSE)
- **height**: scalar, Height of the canvas plot (default: 600, min. size: 100).
- **title**: string: set an overall title for the plot(s).
- **key**: bool, If the **factor** series is provided, a legend shows the color and point pattern for each distinct value of the **factor** variable. Default: 1 (TRUE).
- **key_fontsize** int, Control the font size for the key. Default: 14
- **key_position**: string, Controls the position of the legend in each subplot (use standard gnuplot options). default: “top left”.
- **offset_level**: scalar, Provides a mechanism to put a boundary around the data inside of an auto-scaled graph. For details see: https://gnuplot.sourceforge.net/docs_6.0/loc14594.html , Default: 0.02.
- **tmargin**: scalar, Top margin of the plot in lines. Positive values move the
- **bmarg**: scalar, Bottom margin of the plot in lines.
- **lmarg**: scalar, Left margin of the plot in lines.
- **rmarg**: scalar, Right margin of the plot in lines.
- **points**: int, Control the size of points (default: 2). No support for circles.
- **pointtype**: matrix, Control the type of points. Can be either an integer entry such that points associated to different levels of **factor** are of the same type. One may also provide a vector of integers for each distinct value of the **factor** series. Default: 7. See also: <https://i.imgur.com/xNZa7Rz.png>
- **ticks**: bool, Print x-ticks and y-ticks if 1 (TRUE, default), or not if 0 (FALSE).
- **ticks_fontsize** int, Control the font size of the x-y ticks. Default: 12
- **transparency_level**: int, Controls the transparency level of points and circles (between 0 (no transparency) and 255 (full transparency)). Default: 90.
- **type**: string, Select the type of plot (default: “triangle”). The following types are supported:
- (i) “matrix”: Scatterplot matrix which draws a scatter plot between all combinations of variables included in list L.
- (ii) “triangle”: Lower triangular version of the scatterplot “matrix” type.
- (iii) “row”: Plot the first series element in list L in a single row against all remaining list members.
- (iv) “column”: Works like type “row” but subplots are arranged in a single column instead.
- **use_circles**: bool, Draw circles instead of points if set to 1 (TRUE), default: 0 (FALSE).
- **width**: scalar, Width of the canvas plot (default: 900, min. size: 100).

- `xlabel_offset_horizontal`: scalar, Horizontal offset for the X-axis label (default: 0). Negative value move the label to the left, and vice versa.
- `xlabel_offset_vertical`: scalar, Vertical offset for the X-axis label (default: 0). Negative values move the label downwards, and vice versa.
- `ylabel_offset_horizontal`: scalar, Horizontal offset for the Y-axis label (default: 0). Negative value move the label to the left, and vice versa.
- `ylabel_offset_vertical`: scalar, Vertical offset for the Y-axis label (default: 0). Negative values move the label downwards, and vice versa.

3 Changelog

- **October 2026**

Switch help to L^AT_EX (pdf) format.

- **v1.1 (July 2025)**

- For triangle-type of plot: improved x-/y-label positioning
- Make margins of the plot configurable via `tmargin`, `bmarg`, `lmarg` and `rmarg` parameters.
- Added `xlabel_offset_horizontal`, `xlabel_offset_vertical`, `ylabel_offset_horizontal` and `ylabel_offset_vertical` parameter to control X-axis and Y-axis label offset.

- **v1.0 (February 2025)**

- Bugfix: Catch error in case size of width or height is less than 100.
- Bugfix: Sample 2 not working before as reference to dataset `iris.gdt` was missing.
- Ensure that the `factor` series is of type ‘discrete’. If not, an error message is thrown.
- Throw an error if the `factor` series has more than 16 distinct values.

- **v0.99 (July 2024)**

- Add new parameter `title` for setting an overall title
- Internal refactoring: Switch to gretl’s built-in gridplot apparatus which means that all the graphics file formats supported by gretl can be used.
- It also means that usage via gretlcli in “display” mode works.
- In case the user’s specification calls for just a single plot, no “multiplot” is created.
- The graphical interface is enhanced, with more options.

- **v0.98 (April 2024)**

- Make width, height, font size and point size a function of the number of variables for better readability in case of many variables.
- Set parameter `transparency` per default to 90.
- Minor update of the sample script

- **v0.97 (February 2024)**

- Error in case of CLI-mode if a plot is expected to appear on the screen (filename=“display”)

- **v0.96 (February 2024)**

- Set all terminals to ‘noenhanced’ mode to respect eventual underscores in variable names

- Allow the parameters ‘centroid_pointsize’, ‘centroid_linewidth’, ‘width’ and ‘height’ to be scalar values instead of integers.
- **v0.95 (December 2023)**
 - Bugfix: Path with blanks failed under Windows sometimes.
 - Bugfix: An error occurred with negative values in some cases
 - Bugfix: An error occurred of the name of the last variable in the list starts with ‘e’ (or ‘E’)
 - Fix handling of transparency
 - Add `grid` option for (optionally) drawing a grid in the background
 - Add options `key_fontsize` and `ticks_fontsize`.
 - Add support for drawing centroids of 2-dimensional space.
 - Remove the `font` option for setting the font (controlled by gretl now)
 - Increase default font size to 12
 - Switch to markdown-based help file
 - Increase minimum gretl version to 2022d
- **v0.9 (October 2022)**
 - Fix bug with `const` inheritance: its illegal for a function which receives an argument A that is declared as `const` to pass A on in pointer form to another function which does not promise to treat it as `const`.
- **v0.8 (January 2022)**
 - Handle eventual missing values in passed series.
 - Bugfix of gnuplot’s `offset` option: More than at maximum 4 were passed in case of more than 4 features to plot.
 - Bugfix: Works now with more than 8 features.
- **v0.7 (October 2021)**
 - Enforce full sample instead of using `--replace` option
 - Changed ordering of the colors when plotting distinct factors
 - Change the color palette to “dark2”
- **v0.6 (March 2021)**
 - New API for `PairPlot()` – backward-incompatible
 - update help text
 - enforce decimal point when print numerical data to gnuplot file
 - substantial internal refactoring and clean up
 - gain of speed
 - new minimum gretl version is 2020e
- **v0.5 (Oct. 2019)**
 - Initial version