# Package: gglorenz (via r-universe)

October 30, 2026

**Title** Plotting Lorenz Curve with the Blessing of 'ggplot2'

**Version** 0.0.2.9000

**Description** Provides statistical transformations for plotting empirical ordinary Lorenz curve (Lorenz 1905) <doi:10.2307/2276207> and generalized Lorenz curve (Shorrocks 1983) <doi:10.2307/2554117>.

**Depends** R (>= 3.2.0), ggplot2 (>= 2.2.1)

**License** MIT + file LICENSE

**Encoding** UTF-8

**LazyData** true

**URL** https://github.com/jjchern/gglorenz

**BugReports** https://github.com/jjchern/gglorenz/issues

**RoxygenNote** 7.1.1

**Imports** ineq

**Suggests** spelling

**Language** en-US

**Repository** https://jjchern.r-universe.dev

**Date/Publication** 2020-07-13 00:58:49 UTC

**RemoteUrl** https://github.com/jjchern/gglorenz

**RemoteRef** HEAD

**RemoteSha** 305ba873ceb13dbf1fdb8ac08e049147e77bae84

## Contents


annotate_ineq . . . . . 2
billionaires . . . . . 3
gglorenz . . . . . 3
stat_lorenz . . . . . 4
stat_lorenz_generalized . . . . . 5

**Index** **7**

annotate_ineq *Annotate ggplot2 chart with inequality metric*

## Description

Adds text annotation to chart with any inequality measure from ineq::ineq Inequality measures include Gini, RS, Atkinson, Theil, Kol, var, square.var, entropy

## Usage

```
annotate_ineq(
  data_ineq,
  freq_ineq = rep(1, length(data_ineq)),
  x = 0.1,
  y = 0.95,
  decimals = 2,
  measure_ineq = "Gini",
  sep_ineq = ": ",
  ...
)
```

## Arguments

| | |
|---|---|
| data_ineq | Data to calculate the inequality metric on. |
| freq_ineq | (optional) Frequency of each observation, defaults to 1, should be same length as data |
| x | annotation x-axis position, defaults to 0.1. |
| y | annotation y-axis position, defaults to 0.95. |
| decimals | number of decimals to show, defaults to 2. |
| measure_ineq | Name of measure to use; defaults to Gini. |
| sep_ineq | text separator between annotation label and value. |
| ... | any additional parameters to ggplot2::annotate(). |

## References

Gini coefficient from Wikipedia

## Examples

```
library(gglorenz)

ggplot(billionaires, aes(TNW)) +
    stat_lorenz() +
    annotate_ineq(billionaires$TNW)

ggplot(billionaires, aes(TNW)) +
```

```
    stat_lorenz(desc = TRUE) +
    geom_abline(linetype = "dashed") +
    theme_bw() +
    annotate_ineq(billionaires$TNW, measure_ineq = "RS", color = "red",
                  family = theme_get()$text[["family"]],
                  size = theme_get()$text[["size"]] / 2,
                  fontface = "italic")

## Not run:
ggplot(freqdata, aes(x=value, n=freq)) +
    stat_lorenz() +annotate_ineq(freqdata$val, freqdata$freq)
## End(Not run)
```

---

`billionaires` *Billionaires data*

---

## Description

Contains 500 billionaires' name, country, industry, and total net worth. The data is collected in Feb. 8, 2018.

## Usage

```
billionaires
```

## Format

An object of class `tbl_df` (inherits from `tbl`, `data.frame`) with 500 rows and 6 columns.

## Source

https://www.bloomberg.com/billionaires/

---

`gglorenz` *gglorenz: Plotting Lorenz Curve with ggplot2*

---

## Description

The package provides statistical transformations for plotting empirical ordinary Lorenz curve and generalized Lorenz curve.

## Author(s)

JJ Chen

stat_lorenz *Values of Ordinary Lorenz Curve*

## Description

Provides ordinary Lorenz curve values for line plots

## Usage

```
stat_lorenz(
  mapping = NULL,
  data = NULL,
  geom = "path",
  position = "identity",
  ...,
  desc = FALSE,
  show.legend = NA,
  inherit.aes = TRUE
)
```

## Arguments

| | |
|---|---|
| mapping | Set of aesthetic mappings created by aes() or aes_(). If specified and inherit.aes = TRUE (the default), it is combined with the default mapping at the top level of the plot. You must supply mapping if there is no plot mapping. |
| data | The data to be displayed in this layer. There are three options:<br>If NULL, the default, the data is inherited from the plot data as specified in the call to ggplot().<br>A data.frame, or other object, will override the plot data. All objects will be fortified to produce a data frame. See fortify() for which variables will be created.<br>A function will be called with a single argument, the plot data. The return value must be a data.frame, and will be used as the layer data. A function can be created from a formula (e.g. ~ head(.x, 10)). |
| geom | which geom to use; defaults to "path". |
| position | Position adjustment, either as a string, or the result of a call to a position adjustment function. |
| ... | Other arguments passed on to layer(). These are often aesthetics, used to set an aesthetic to a fixed value, like colour = "red" or size = 3. They may also be parameters to the paired geom/stat. |
| desc | If FALSE, the default, the population is arranged in ascending order along the x-axis. If TRUE, the population is arranged in descending order. |
| show.legend | logical. Should this layer be included in the legends? NA, the default, includes if any aesthetics are mapped. FALSE never includes, and TRUE always includes. It can also be a named logical vector to finely select the aesthetics to display. |

inherit.aes — If FALSE, overrides the default aesthetics, rather than combining with them. This is most useful for helper functions that define both data and aesthetics and shouldn't inherit behaviour from the default plot specification, e.g. borders().

## Aesthetics

stat_lorenz() understands the following aesthetics:

- x (required: wealth of individual, or set of individuals if n aesthetic used)
- n (optional: frequency of given observation x)

## References

Lorenz curve from Wikipedia

## Examples

```
library(gglorenz)

ggplot(billionaires, aes(TNW)) +
    stat_lorenz()

ggplot(billionaires, aes(TNW)) +
    stat_lorenz(desc = TRUE) +
    coord_fixed() +
    geom_abline(linetype = "dashed") +
    theme_minimal()

## Not run:
ggplot(freqdata, aes(x=value, n=freq)) +
    stat_lorenz()
## End(Not run)
```

---

# stat_lorenz_generalized

*Values of Generalized Lorenz Curve*

---

## Description

Provides generalized Lorenz curve values for line plots

## Usage

```
stat_lorenz_generalized(
  mapping = NULL,
  data = NULL,
  geom = "path",
  position = "identity",
```

```
  ...,
  show.legend = NA,
  inherit.aes = TRUE
)
```

## Arguments

| | |
|---|---|
| mapping | Set of aesthetic mappings created by aes() or aes_(). If specified and inherit.aes = TRUE (the default), it is combined with the default mapping at the top level of the plot. You must supply mapping if there is no plot mapping. |
| data | The data to be displayed in this layer. There are three options:<br>If NULL, the default, the data is inherited from the plot data as specified in the call to ggplot().<br>A data.frame, or other object, will override the plot data. All objects will be fortified to produce a data frame. See fortify() for which variables will be created.<br>A function will be called with a single argument, the plot data. The return value must be a data.frame, and will be used as the layer data. A function can be created from a formula (e.g. ~ head(.x, 10)). |
| geom | which geom to use; defaults to "path". |
| position | Position adjustment, either as a string, or the result of a call to a position adjustment function. |
| ... | Other arguments passed on to layer(). These are often aesthetics, used to set an aesthetic to a fixed value, like colour = "red" or size = 3. They may also be parameters to the paired geom/stat. |
| show.legend | logical. Should this layer be included in the legends? NA, the default, includes if any aesthetics are mapped. FALSE never includes, and TRUE always includes. It can also be a named logical vector to finely select the aesthetics to display. |
| inherit.aes | If FALSE, overrides the default aesthetics, rather than combining with them. This is most useful for helper functions that define both data and aesthetics and shouldn't inherit behaviour from the default plot specification, e.g. borders(). |

## References

Lorenz curve from Wikipedia

## Examples

```
library(gglorenz)

ggplot(billionaires, aes(TNW)) +
    stat_lorenz_generalized()
```

# Index

∗ **datasets**
billionaires, 3
gglorenz, 3

aes(), *4*, *6*
aes_(), *4*, *6*
annotate_ineq, 2

billionaires, 3
borders(), *5*, *6*

fortify(), *4*, *6*

gglorenz, 3
ggplot(), *4*, *6*

layer(), *4*, *6*

stat_lorenz, 4
stat_lorenz_generalized, 5
StatLorenz *(*gglorenz*)*, 3
StatLorenzGeneralized *(*gglorenz*)*, 3