Transformed Cartesian coordinate system

`coord_trans` is different to scale transformations in that it occurs after statistical transformation and will affect the visual appearance of geoms - there is no guarantee that straight lines will continue to be straight.

```
coord_trans(x = "identity", y = "identity", limx = NULL, limy = NULL,
  xtrans, ytrans)
```

Arguments

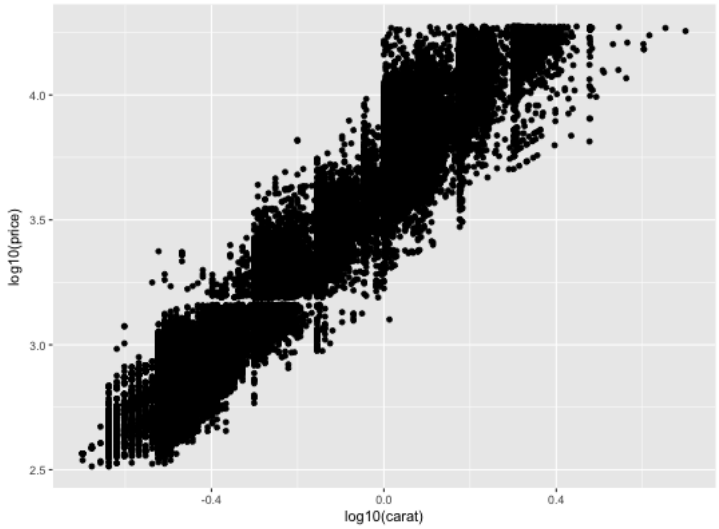
x, y	transformers for x and y axes
limx, limy	limits for x and y axes. (Named so for backward compatibility)
xtrans, ytrans	Deprecated; use `x` and `y` instead.

Details

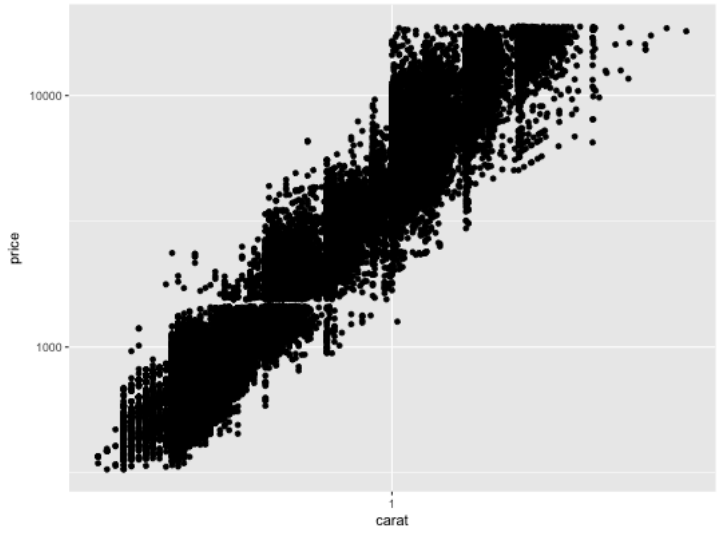
Transformations only work with continuous values: see **`trans_new (http://www.rdocumentation.org/packages/scales/topics/trans_new)`** for list of transformations, and instructions on how to create your own.

Examples

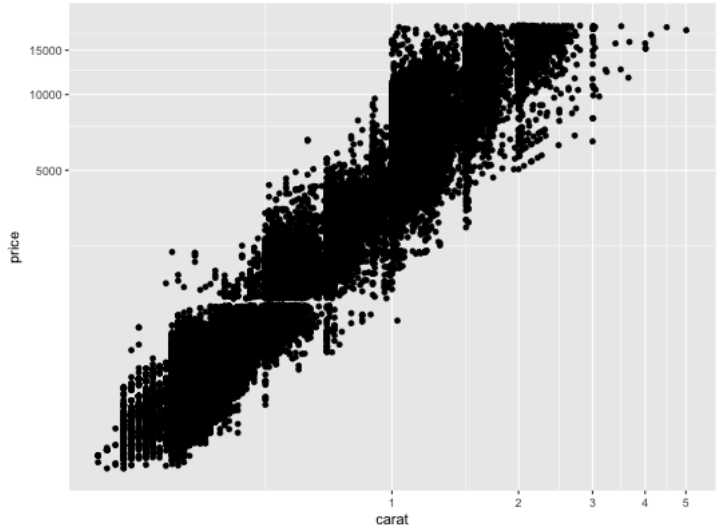
```
# See ?geom_boxplot for other examples

# Three ways of doing transformation in ggplot:
#  * by transforming the data
ggplot (ggplot.html)(diamonds, aes (aes.html)(log10(carat), log10(price))) +
  geom_point (geom_point.html)()
```



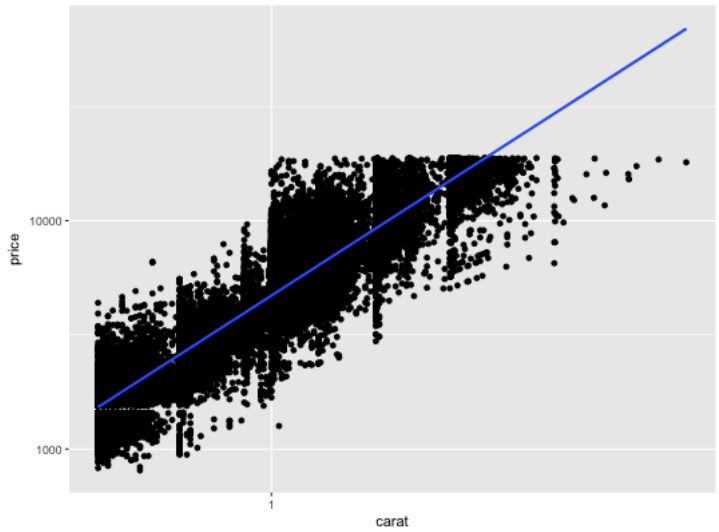

```
#  * by transforming the scales
ggplot (ggplot.html)(diamonds, aes (aes.html)(carat, price)) +
  geom_point (geom_point.html)() +
  scale_x_log10 (scale_continuous.html)() +
  scale_y_log10 (scale_continuous.html)()
```



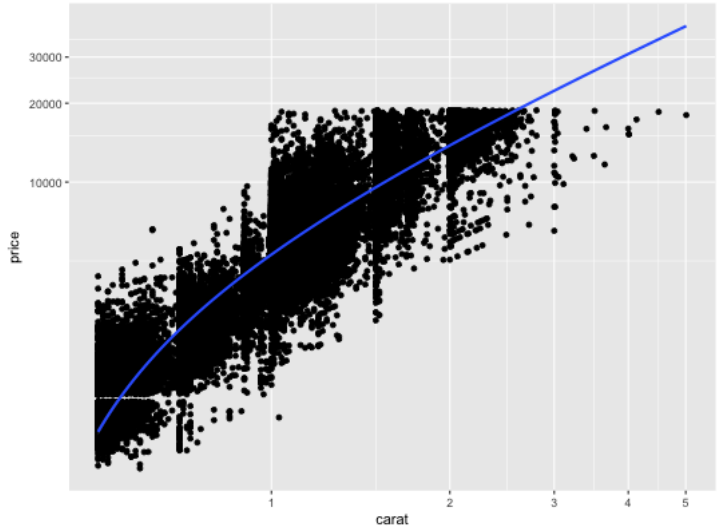

```
#  * by transforming the coordinate system:
ggplot (ggplot.html)(diamonds, aes (aes.html)(carat, price)) +
  geom_point (geom_point.html)() +
  coord_trans(x = "log10", y = "log10")
```



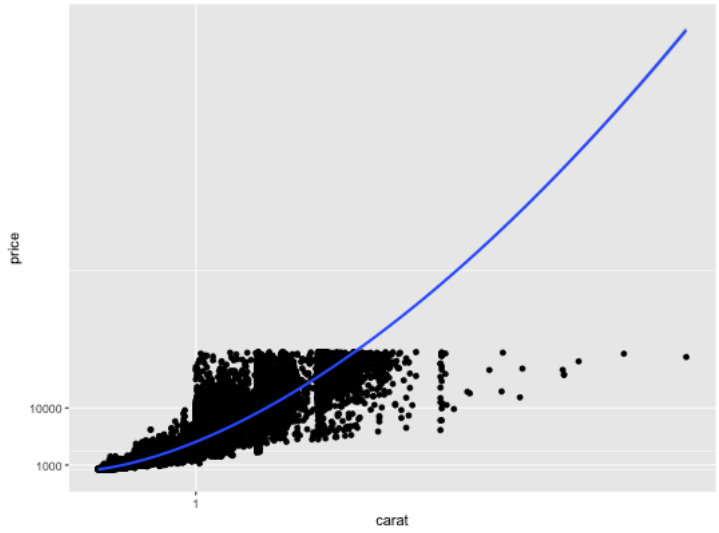

```
# The difference between transforming the scales and
# transforming the coordinate system is that scale
# transformation occurs BEFORE statistics, and coordinate
# transformation afterwards.  Coordinate transformation also
# changes the shape of geoms:

d <- subset(diamonds, carat > 0.5)

ggplot (ggplot.html)(d, aes (aes.html)(carat, price)) +
  geom_point (geom_point.html)() +
  geom_smooth (geom_smooth.html)(method = "lm") +
  scale_x_log10 (scale_continuous.html)() +
  scale_y_log10 (scale_continuous.html)()
```



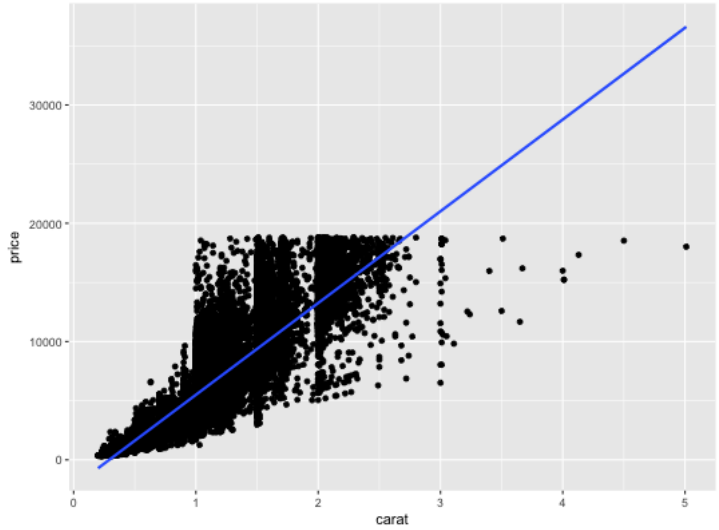

```
ggplot (ggplot.html)(d, aes (aes.html)(carat, price)) +
  geom_point (geom_point.html)() +
  geom_smooth (geom_smooth.html)(method = "lm") +
  coord_trans(x = "log10", y = "log10")
```



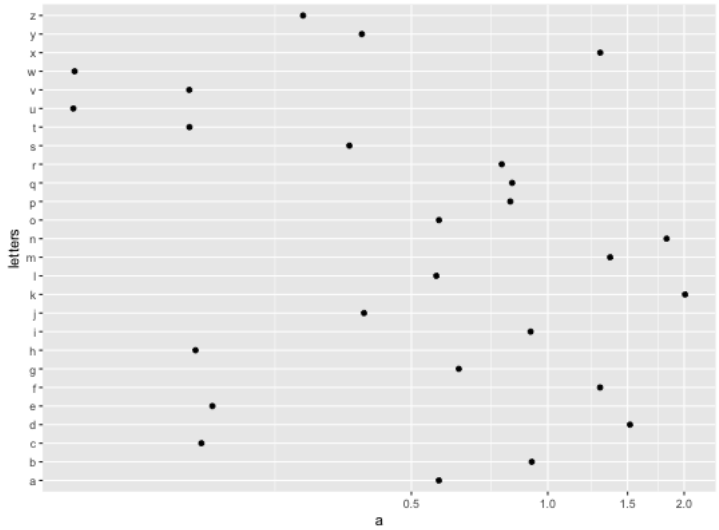

```
# Here I used a subset of diamonds so that the smoothed line didn't
# drop below zero, which obviously causes problems on the log-transformed
# scale

# With a combination of scale and coordinate transformation, it's
# possible to do back-transformations:
ggplot (ggplot.html)(diamonds, aes (aes.html)(carat, price)) +
  geom_point (geom_point.html)() +
  geom_smooth (geom_smooth.html)(method = "lm") +
  scale_x_log10 (scale_continuous.html)() +
  scale_y_log10 (scale_continuous.html)() +
  coord_trans(x = scales::exp_trans(10), y = scales::exp_trans(10))
```




```
# cf.
ggplot (ggplot.html)(diamonds, aes (aes.html)(carat, price)) +
  geom_point (geom_point.html)() +
  geom_smooth (geom_smooth.html)(method = "lm")
```




```
# Also works with discrete scales
df <- data.frame(a = abs(rnorm(26)),letters)
plot <- ggplot (ggplot.html)(df,aes (aes.html)(a,letters)) + geom_point (geom_point.html)()

plot + coord_trans(x = "log10")
```




```
plot + coord_trans(x = "sqrt")
```