How can I create a manual legend in ggplot2 with examples?

Authored by

stats writer

May 12, 2024

RECOMMENDED CITATION

stats writer (2024). *How can I create a manual legend in ggplot2 with examples?*. PSYCHOLOGICAL SCALES. Retrieved from https://scales.arabpsychology.com/?p=144037

ARABPSYCHOLOGY.COM

Creating a manual legend in ggplot2 allows for the customization and organization of the legend in a graph. This can be useful when representing multiple variables or data points on a single plot. To create a manual legend in ggplot2, one must specify the labels and colors for each factor or variable in the legend using the "scale_color_manual" or "scale_fill_manual" functions. This allows for greater control over the appearance of the legend and can be particularly helpful when creating complex or customized plots. For example, if a scatter plot is created with multiple data points, a manual legend can be used to label each point by its color or shape. Overall, using a manual legend in ggplot2 can enhance the clarity and visual appeal of a graph.

Create a Manual Legend in ggplot2 (With Examples)

Often you may want to add a manual legend to a plot in ggplot2 with custom colors, labels, title, etc.

Fortunately this is simple to do using the scale_color_manual() function and the following example shows how to do so.

Example: Create Manual Legend in ggplot2

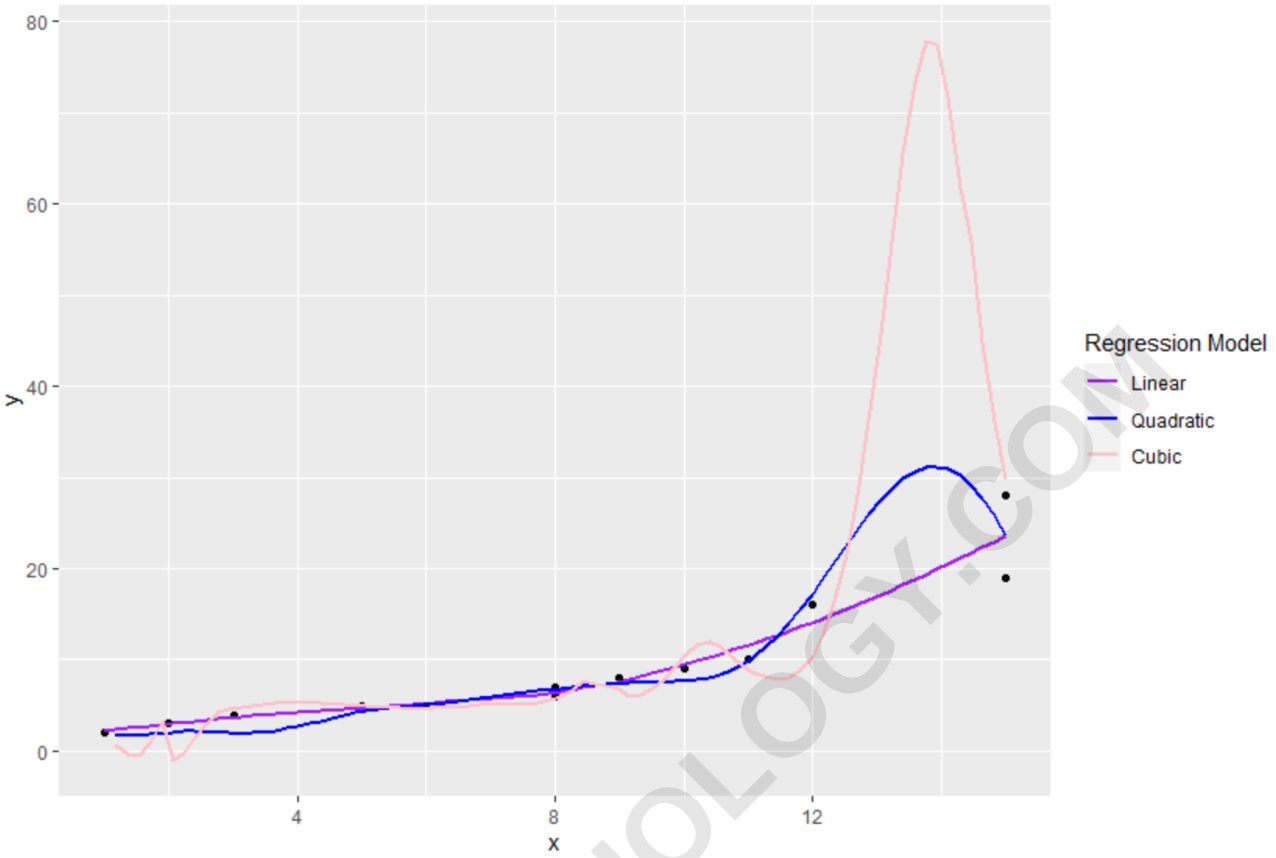
The following code shows how to plot three fitted regression lines in a plot in ggplot2 with a custom manual legend:

```
library(ggplot2)

#create data frame
df <- data.frame(x=c(1, 2, 2, 3, 5, 6, 8, 8, 9, 9, 10, 11, 12, 15, 15),
```

ARABPSYCHOLOGY.COM

```
y=c(2, 3, 3, 4, 5, 5, 6, 7, 8, 8, 9, 10, 16, 19, 28))

#create plot with three fitted regression models
ggplot(df, aes(x, y)) +
geom_point() +
geom_smooth(se=FALSE, aes(color='Linear')) +
geom_smooth(formula=y~poly(x, 2), se=FALSE,
aes(color='Quadratic')) +
geom_smooth(formula=y~poly(x, 3), se=FALSE,
aes(color='Cubic')) +
scale_color_manual(name='Regression Model',
breaks=c('Linear', 'Quadratic', 'Cubic'),
values=c('Cubic'='pink', 'Quadratic'='blue',
'Linear'='purple'))
```

ARABPSYCHOLOGY.COM



ARABPSYCHOLOGY.COM

Using the scale_color_manual() function, we were able to specify the following aspects of the legend:

name: The title of the legendbreaks: The labels in the legendvalues: The colors in the legend

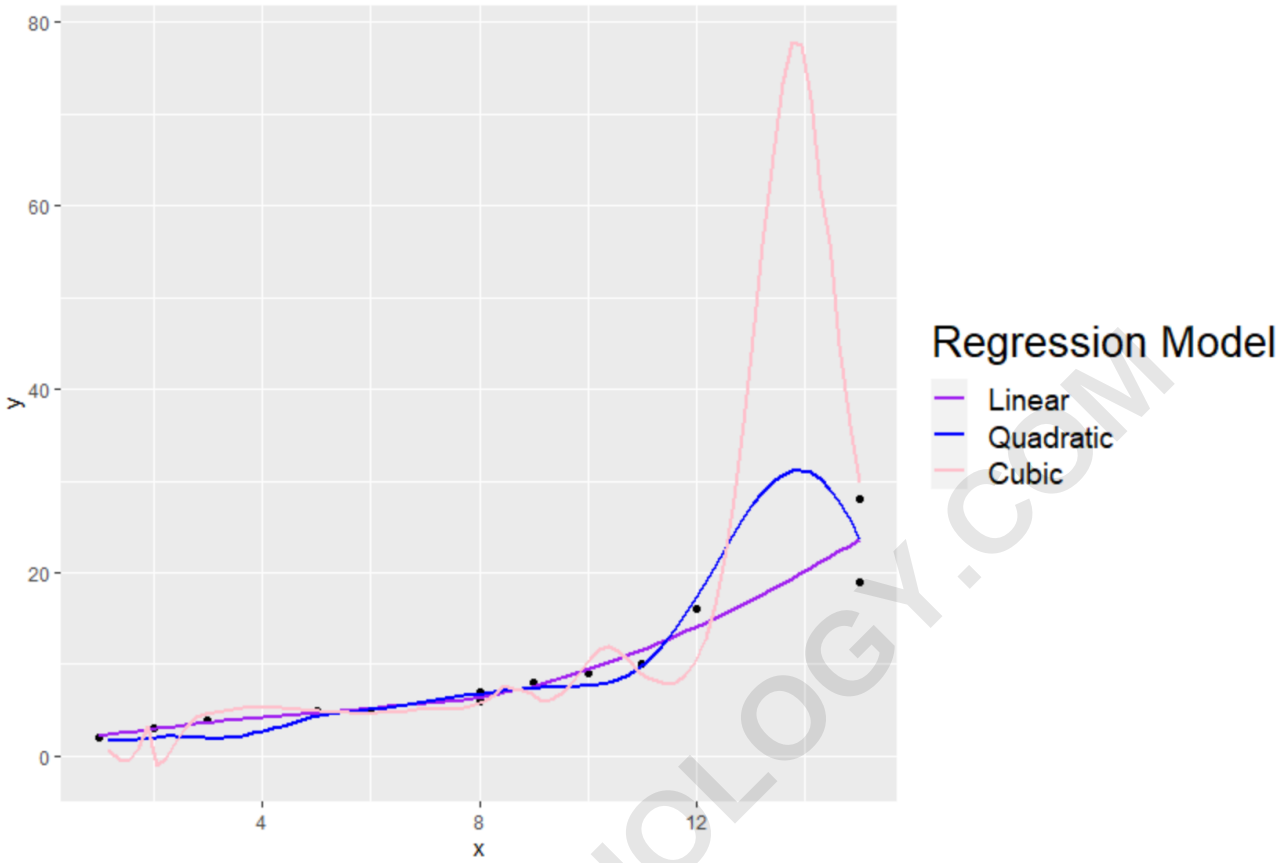
Note that we can also use the theme() function to modify the font size of the elements in the legend:

library(ggplot2)

#create data frame

```
df <- data.frame(x=c(1, 2, 2, 3, 5, 6, 8, 8, 9, 9, 10, 11, 12, 15, 15),
y=c(2, 3, 3, 4, 5, 5, 6, 7, 8, 8, 9, 10, 16, 19, 28))

#create plot with three fitted regression models
ggplot(df, aes(x, y)) +
geom_point() +
geom_smooth(se=FALSE, aes(color='Linear')) +
geom_smooth(formula=y~poly(x, 2), se=FALSE, aes(color='Quadratic')) +
geom_smooth(formula=y~poly(x, 3), se=FALSE, aes(color='Cubic')) +
scale_color_manual(name='Regression Model',
breaks=c('Linear', 'Quadratic', 'Cubic'),
values=c('Cubic'='pink', 'Quadratic'='blue', 'Linear'='purple'))+
theme(legend.title=element_text(size=20),
legend.text=element_text(size=14))
```

ARABPSYCHOLOGY.COM



ARABPSYCHOLOGY.COM

Notice that the font size of both the title and the labels in the legend were increased.

The following tutorials explain how to perform other common operations in ggplot2: