R Colors

Austin Pesina

2021-08-21

Contents

Introduction

The default color palettes used in computer graphics often leave much to to be desired. Besides being aesthetically vapid, they often have poor contrast and colors that are difficult to distinguish, especially for the color-blind.

Fortunately, cartographic researchers identified this problem quite some time ago. One of the results of this research was the ColorBrewer set of palettes. And lucky for us, these palettes are available to R programmers in the `rColorBrewer` package.

This demonstration script compares the default palette and a selected ColorBrewer palette.

Data

Plot some data from a regression example given in Ott and Longneckers *Statistical Methods and Data Analysis*:

```
FER.data <- read.table(header=TRUE, text="
additive    FER
0   1.30
0   1.35
0   1.44
0   1.52
0   1.56
0   1.61
```

```
0   1.48
0   1.56
0   1.45
0   1.14
20  2.17
20  2.11
20  2.08
20  2.13
20  2.22
20  2.29
20  2.33
20  2.24
20  2.16
20  2.21
40  2.30
40  2.34
40  2.20
40  2.38
40  2.48
40  2.44
40  2.37
40  2.43
40  2.37
40  2.41
60  2.47
60  2.51
60  2.79
60  2.40
60  2.55
60  2.67
60  2.50
60  2.55
60  2.60
60  2.49
80  3.31
80  3.17
80  3.24
80  3.21
80  3.35
80  3.38
80  3.42
80  3.36
80  3.25
80  3.51
100 4.92
100 3.87
100 4.81
100 4.88
100 5.06
100 5.09
100 4.97
100 4.95
100 4.59
```

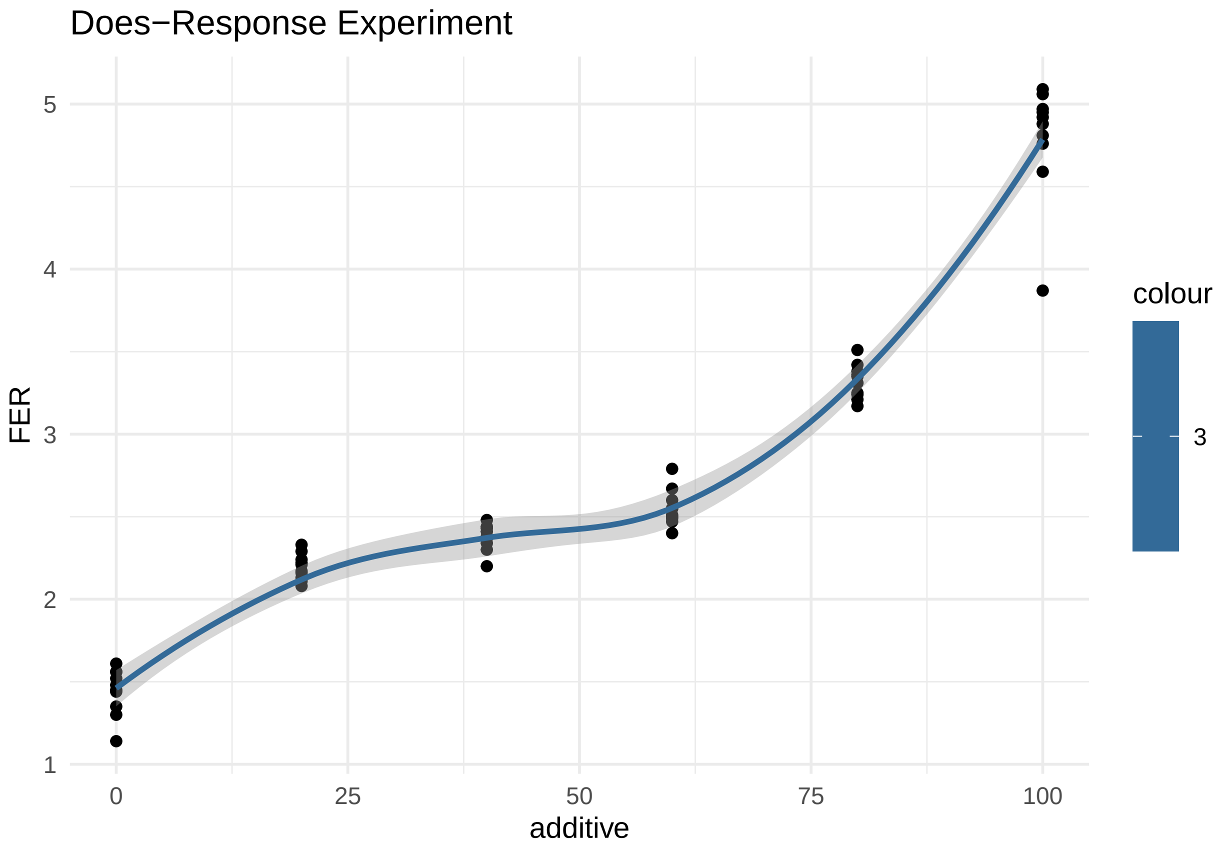
```
100 4.76
")
```

Exploratory Plot

Use `tidyverse::ggplot` to draw a *loess* plot of the data. Use a color from the default color palette. (Colors can be identified by a non-zero integer.)

```
ggplot(data=FER.data, aes(x=additive, y=FER)) +
  geom_point() +
  geom_smooth(method="loess", aes(color=3) ) +
  labs(title="Does-Response Experiment") +
  theme_minimal()
```

```
## ‘geom_smooth()‘ using formula ’y ~ x’
```

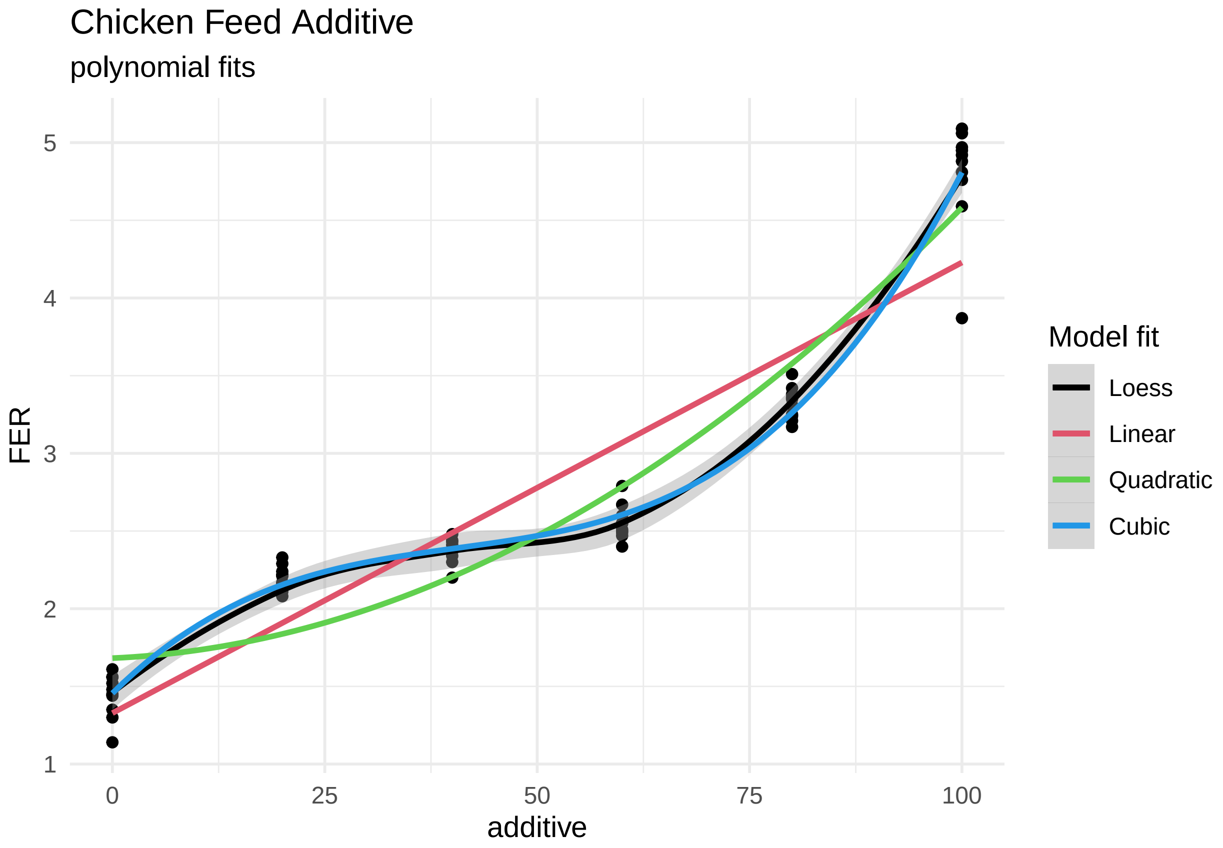


Some Fitted Models

Plot a more complicated graph, showing various polynomial fits to the data overlaid on one another. Here the default colors are identified by character versions of the integers to support the`scale_color_identity_name()` function, used to build a legend for the plot.

```
ggplot(data=FER.data, aes(x=additive, y=FER)) +
  geom_point() +
  geom_smooth(method="loess", show.legend=TRUE, aes(color="1")) +
  geom_smooth(method="lm", formula='y~x', se=FALSE, show.legend=TRUE, aes(color="2") ) +
  geom_smooth(method="lm", formula='y~poly(x,2)', se=FALSE, show.legend=TRUE, aes(color="3") ) +
  geom_smooth(method="lm", formula='y~poly(x,3)', se=FALSE, show.legend=TRUE, aes(color="4") ) +
  scale_color_identity(name = "Model fit",
                        breaks = c("1", "2", "3", "4"),
                        labels = c("Loess", "Linear", "Quadratic", "Cubic"),
                        guide = "legend") +
  labs(title="Chicken Feed Additive", subtitle="polynomial fits") +
  theme_minimal()
```

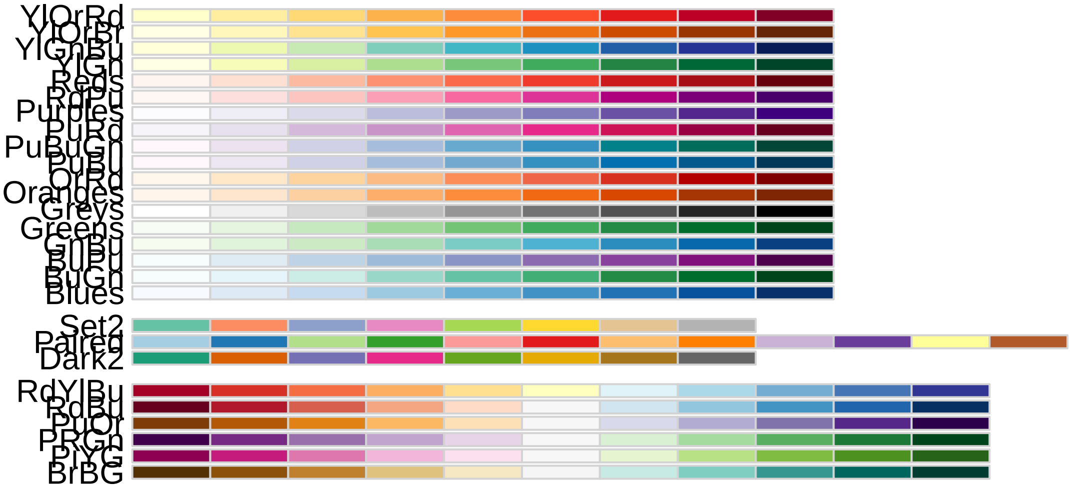
```
## `geom_smooth()` using formula 'y ~ x'
```



Change Color Palette

Hello, ColorBrewer!

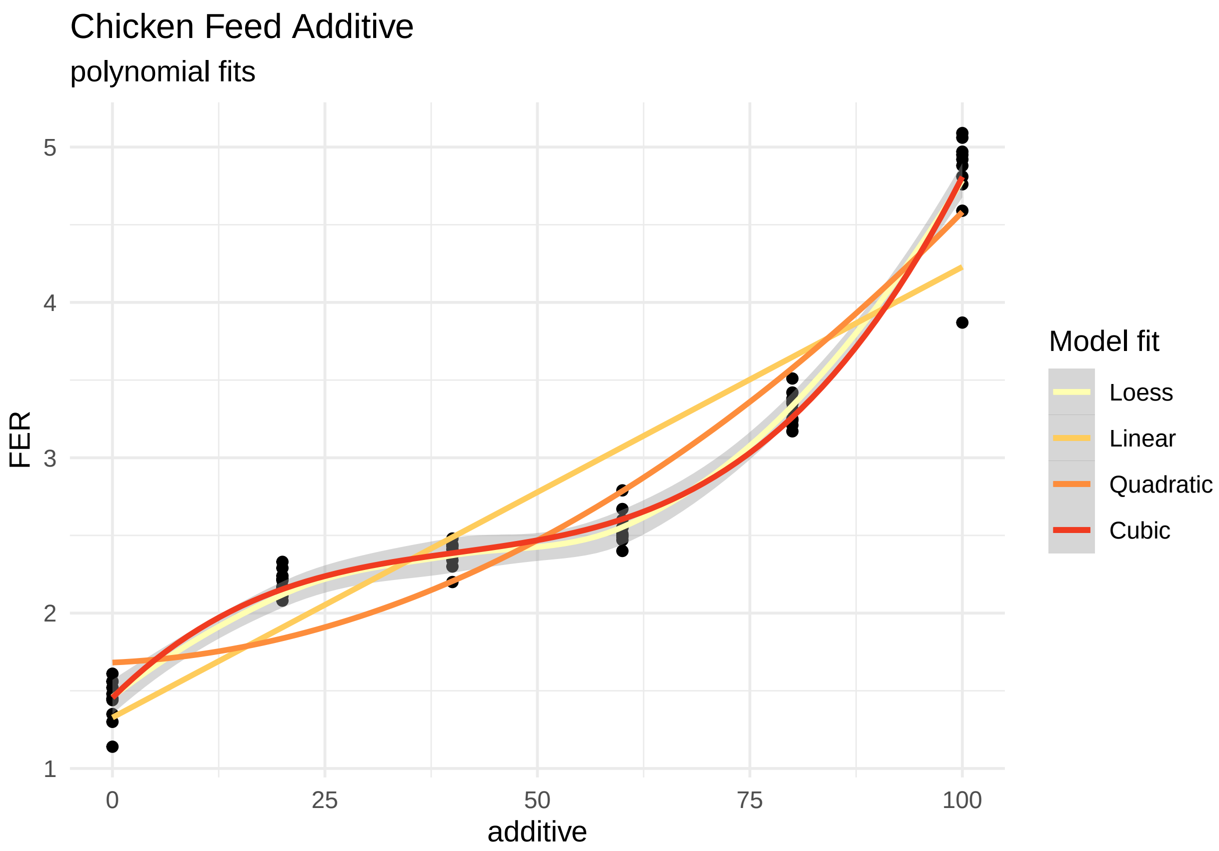
```
library("RColorBrewer")
display.brewer.all(colorblindFriendly=TRUE)
```



Redraw the previous graphic, using your favorite ColorBrewer palette:

```
cbPalette <- brewer.pal(5, "YlOrRd")
ggplot(data=FER.data, aes(x=additive, y=FER)) +
  geom_point() +
  geom_smooth(method="loess", show.legend=TRUE, aes(color=cbPalette[1])) +
  geom_smooth(method="lm", formula='y~x', se=FALSE, show.legend=TRUE, aes(color=cbPalette[2]) ) +
  geom_smooth(method="lm", formula='y~poly(x,2)', se=FALSE, show.legend=TRUE, aes(color=cbPalette[3]) )
  geom_smooth(method="lm", formula='y~poly(x,3)', se=FALSE, show.legend=TRUE, aes(color=cbPalette[4]) )
  scale_color_identity(name = "Model fit",
                       breaks = cbPalette[1:4],
                       labels = c("Loess", "Linear", "Quadratic", "Cubic"),
                       guide = "legend") +
  labs(title="Chicken Feed Additive", subtitle="polynomial fits") +
  theme_minimal()
```

```
## ‘geom_smooth()‘ using formula ’y ~ x’
```


Chicken Feed Additive
polynomial fits
FER
5
4
3
2
1
0
25
50
75
100
additive
Model fit
Loess
Linear
Quadratic
Cubic