

R color cheatsheet

Finding a good color scheme for presenting data can be challenging. This color cheatsheet will help!

R uses hexadecimal to represent colors

Hexadecimal is a base-16 number system used to describe color. Red, green, and blue are each represented by two characters (#rrggbb). Each character has 16 possible symbols: 0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F:

“00” can be interpreted as 0.0 and “FF” as 1.0
i.e., red = #FF0000, black = #000000, white = #FFFFFF

Two additional characters (with the same scale) can be added to the end to describe transparency (#rrggbbaa)

R has 657 built in color names

To see a list of names:

```
colors()
```

These colors are displayed on P. 3.

Example:

```
peachpuff4
```

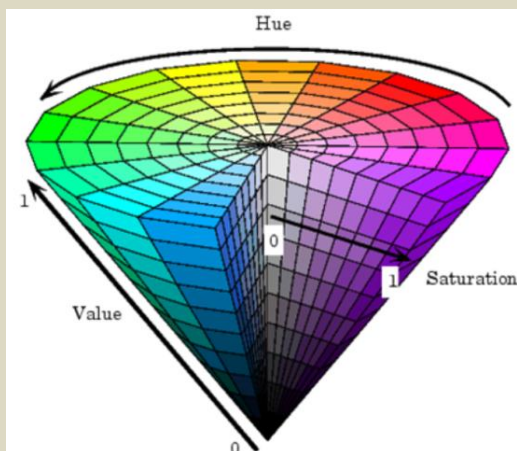
R translates various color models to hex, e.g.:

- RGB (red, green, blue): The default intensity scale in R ranges from 0-1; but another commonly used scale is 0-255. This is obtained in R using `maxColorValue=255`. *alpha* is an optional argument for transparency, with the same intensity scale.
`rgb(r, g, b, maxColorValue=255, alpha=255)`
- HSV (hue, saturation, value): values range from 0-1, with optional alpha argument
`hsv(h, s, v, alpha)`
- HCL (hue, chroma, luminance): hue describes the color and ranges from 0-360; 0 = red, 120 = green, blue = 240, etc. Range of chroma and luminance depend on hue and each other
`hcl(h, c, l, alpha)`

A few notes on HSV/HCL

HSV is a better model for how humans perceive color. HCL can be thought of as a perceptually based version of the HSV model....blah blah blah...

Without delving into color theory: color schemes based on HSV/HCL models generally just look good.



R can translate colors to rgb (this is handy for matching colors in other programs)

```
col2rgb(c("#FF0000", "blue"))
```

R Color Palettes

This is for all of you who don't know anything about color theory, and don't care but want some nice colors on your map or figure....NOW!

TIP: When it comes to selecting a color palette, **DO NOT** try to handpick individual colors! You will waste a lot of time and the result will probably not be all that great. R has some good packages for color palettes. Here are some of the options

Packages: grDevices and colorRamps

grDevices comes with the base installation and colorRamps must be installed. Each palette's function has an argument for the number of colors and transparency (*alpha*):

```
heat.colors(4, alpha=1)  
> #FF0000FF" "#FF8000FF" "#FFFF00FF" "#FFFF80FF"
```

grDevices
palettes
cm.colors
topo.colors
terrain.colors
heat.colors
rainbow
see P. 4 for
options

For the `rainbow` palette you can also select start/end color (red = 0, yellow = 1/6, green = 2/6, cyan = 3/6, blue = 4/6 and magenta = 5/6) and saturation (*s*) and value (*v*):
`rainbow(n, s = 1, v = 1, start = 0, end = max(1, n - 1)/n, alpha = 1)`

Package: RcolorBrewer

This function has an argument for the number of colors and the color palette (see P. 4 for options).
`brewer.pal(4, "Set3")`

```
> "#8DD3C7" "#FFFFB3" "#BEBADA" "#FB8072"
```

To view colorbrewer palettes in R: `display.brewer.all(5)`

There is also a very nice interactive viewer:

<http://colorbrewer2.org/>

My Recommendation

Package: colorspace

These color palettes are based on HCL and HSV color models. The results can be very aesthetically pleasing. There are some default palettes:

```
rainbow_hcl(4)  
"#E495A5" "#ABB065" "#39BEB1" "#ACA4E2"
```

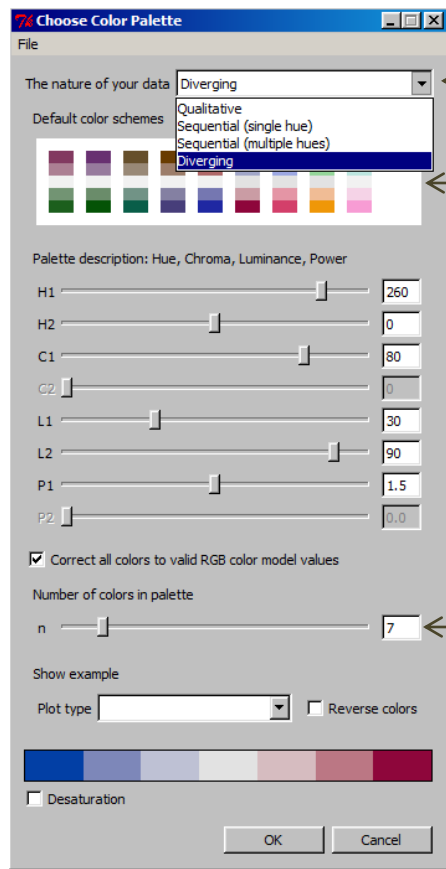
colorspace
default palettes
diverge_hcl
diverge_hsl
terrain_hcl
sequential_hcl
rainbow_hcl

However, all palettes are fully customizable:
`diverge_hcl(7, h = c(246, 40), c = 96, l = c(65, 90))`
Choosing the values *would* be daunting. But there are some recommended palettes in the colorspace documentation. There is also an interactive tool that can be used to obtain a customized palette. To start the tool:
`pal <- choose_palette()`

R color cheatsheet

Overview of colorspace palette selector

```
library("colorspace")
pal <- choose_palette()
```

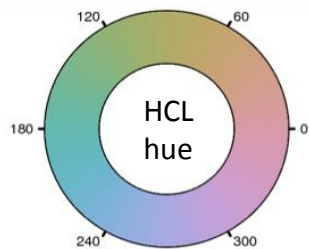


Select the type of color scheme based on the type of data

Default color schemes – can be used “as is” or as a starting point for modification

Interactively select:

- hue: color
- chroma: low chroma = gray
- luminance: high luminance = pastel
- power: how the color changes along a gradient



Select # of colors in palette

Save palette for future R sessions:

- txt file with hex codes
- .R file with a function describing how to generate the palette.

`source` can be used to import the function into R; but one complication is that you have to open the .R file and name the function to use it.

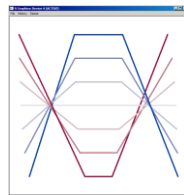
- Copy values into relevant colorspace functions.

Diverging color schemes:
`diverge_hcl(7, h = c(260, 0), c = 100, l = c(28, 90), power = 1.5)`

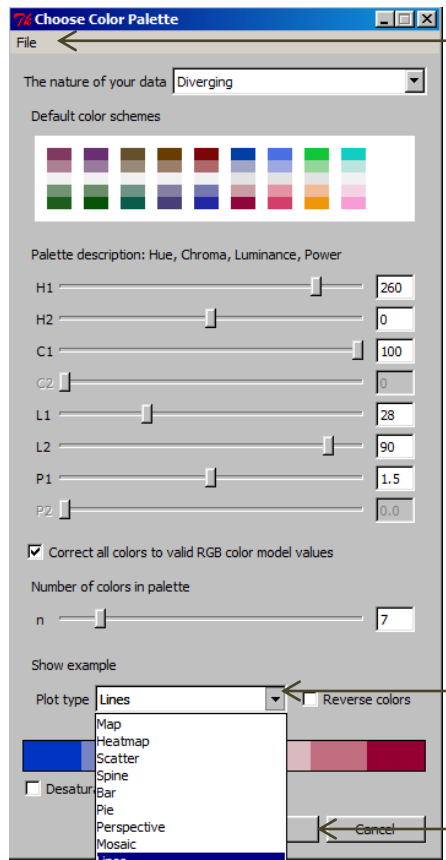
Sequential color schemes:
`sequential_hcl(n, h, c, l, power)`

Qualitative color schemes:
`rainbow_hcl(n, c, l, start, end)` (for qualitative schemes; start/ end refer to the H1/H2 hue values)

Display color scheme with different plot types



When “OK” is selected, the color palette will be saved in the R session. To return 7 hex color codes from the selected palette:
`pal <- choose_palette()`
`pal(7)`
 [NOTE: These values are not saved if you don't save the session]



How to use hex codes to define color using the plot function

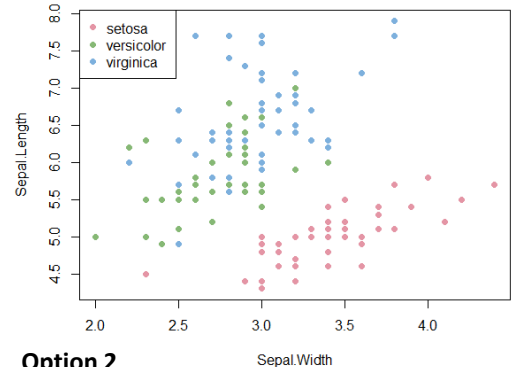
Discrete variables

Option 1

If you don't need to control which colors are associated with each level of a variable:

```
plot(Sepal.Length ~ Sepal.Width,
     col=rainbow_hcl(3)[c(Species)],
     data=iris, pch=16)
```

```
legend("topleft", pch=16, col=rainbow_hcl(3),
      legend=unique(iris$Species))
```



Option 2

If you want to control which colors are associated with the levels of a variable, I find it easiest to create a variable in the data:

```
iris$color <- factor(iris$Species,
                    levels=c("virginica", "versicolor", "setosa"),
                    labels=rainbow_hcl(3))
```

```
plot(Sepal.Length ~ Sepal.Width,
     col=as.character(color), pch=16, data=iris)
```

Continuous variables

Option 1

Break into categories and assign colors:

```
iris2 <- subset(iris, Species=="setosa")
```

```
color <- cut(iris2$Petal.Length,
            breaks=c(0,1.3,1.5,2), labels=sequential_hcl(3))
```

Or, break by quantiles (be sure to include 0 & 1):

```
color <- cut(iris2$Petal.Length,
            breaks=quantile(iris$Petal.Length, c(0, 0.25, 0.5, 0.75, 1)),
            labels=sequential_hcl(3))
```

```
plot(Sepal.Width ~ Sepal.Length, pch=16,
     col=color, data=iris2)
```

Option 2

Fully continuous gradient:

```
data <- data.frame("a"=runif(10000),
                  "b"=runif(10000))
```

```
color=diverge_hcl(length(data$a))[rank(data$a)]
plot(a~b, col=color, pch=16, data=data)
```

For ggplot2, I think the most flexible color scales are:

- `scale_colour_manual`
- `scale_colour_gradient`

for discrete and continuous variables, respectively

grDevices::colors

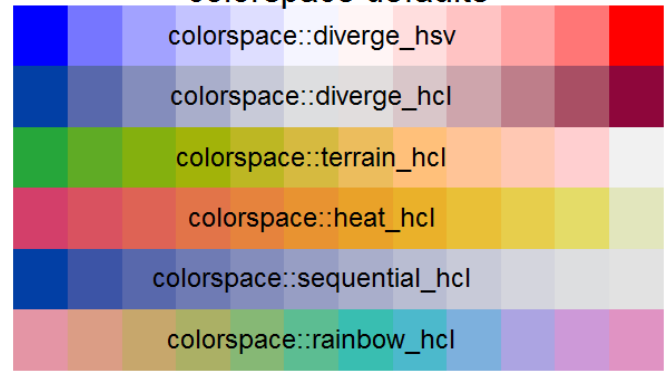
coral3	gray27	gray99	lightpink1	mistyrose1	pink4	slategray1	yellowgreen
coral2	gray26	gray98	lightpink2	mistyrose2	pink3	slategray2	yellow4
coral1	gray25	gray97	lightpink3	mintcream	pink2	slategray3	yellow3
coral	gray24	gray96	lightgray	midnightblue	pink1	slateblue3	yellow2
chocolate4	gray23	gray95	lightgreen	mediumvioletred	peru	slateblue2	yellow1
chocolate3	gray22	gray94	lightgoldenrodyellow	mediumslateblue	peachpuff4	slateblue1	whitesmoke
chocolate2	gray21	gray93	lightgoldenrod3	mediumspringgreen	peachpuff3	skyblue4	wheat4
chocolate1	gray20	gray92	lightgoldenrod2	mediumslateblue	peachpuff2	skyblue3	wheat3
chartreuse4	gray19	gray91	lightgoldenrod1	mediumseagreen	peachpuff1	skyblue2	wheat2
chartreuse3	gray18	gray90	lightgoldenrod	mediumpurple4	peachpuff	skyblue1	wheat1
chartreuse2	gray17	gray89	lightgoldenrod	mediumpurple3	papaya Whip	sienna4	violate4
chartreuse1	gray16	gray88	lightcyan4	mediumpurple2	palevioletred4	sienna3	violate3
chartreuse	gray15	gray87	lightcyan3	mediumpurple1	palevioletred3	sienna2	violate2
cadetblue4	gray14	gray86	lightcyan2	mediumpurple	palevioletred2	sienna1	violate1
cadetblue3	gray13	gray85	lightcyan1	mediumorchid4	palevioletred1	sienna	turquoise4
cadetblue2	gray12	gray84	lightcyan	mediumorchid3	paleturquoise4	seashell4	turquoise3
cadetblue1	gray11	gray83	lightcoral	mediumorchid2	paleturquoise3	seashell3	turquoise2
burlywood4	gray10	gray82	lightblue4	mediumorchid1	paleturquoise2	seashell2	turquoise1
burlywood3	gray9	gray81	lightblue3	mediumorchid	paleturquoise1	seashell1	tomato4
burlywood2	gray8	gray80	lightblue2	mediumblue	paleturquoise	seashell	tomato3
burlywood1	gray7	gray79	lightblue1	mediumaquamarine1	paleturquoise	sandybrown	tomato2
brown4	gray6	gray78	lightblue	maroon4	palegreen4	salmon4	tomato1
brown3	gray6	gray77	lemonchiffon4	maroon3	palegreen3	salmon3	thistle4
brown2	gray5	gray76	lemonchiffon3	maroon2	palegreen2	salmon2	thistle3
brown1	gray4	gray75	lemonchiffon2	maroon1	palegreen1	salmon1	thistle2
brown	gray3	gray74	lemonchiffon1	maroon	palegreen	salmon	thistle1
blueviolet	gray2	gray73	lemonchiffon	magenta4	palegreen	seagreen	thistle
blue3	gray1	gray72	lavenderblush4	magenta3	palegreen	seagreen	tan4
blue2	gray	gray71	lavenderblush3	magenta2	orchid4	seagreen	tan3
blue1	gray	gray70	lavenderblush2	magenta1	orchid3	seagreen	tan2
blanchedalmond	gray	gray69	lavenderblush1	limegreen	orchid2	seagreen	tan1
black	gray	gray68	lavenderblush	lightyellow4	orchid1	seagreen	tan
bisque4	goldenrod4	gray67	lavender	lightyellow3	orange4	saddlebrown	steelblue4
bisque3	goldenrod3	gray66	khaki4	lightyellow2	orange3	royalblue4	steelblue3
bisque2	goldenrod2	gray65	khaki3	lightyellow1	orange2	royalblue3	steelblue2
bisque1	goldenrod1	gray64	khaki2	lightyellow	orange1	royalblue2	steelblue1
bisque	gold4	gray63	khaki1	lightsteelblue4	orange	royalblue1	steelblue
beige	ghostwhite	gray62	ivory4	lightsteelblue3	orange	rosybrown4	springgreen4
azure3	gainsboro	gray61	ivory3	lightsteelblue2	orange	rosybrown3	springgreen3
azure2	floralwhite	gray60	ivory2	lightsteelblue1	orange	rosybrown2	springgreen2
azure1	firebrick4	gray59	ivory1	lightslategray	orange	rosybrown1	springgreen1
azure	firebrick3	gray58	indianred4	lightslateblue	olivedrab4	red4	snow4
aquamarine4	firebrick2	gray57	indianred3	lightslateblue	olivedrab3	red3	snow3
aquamarine3	firebrick1	gray56	indianred2	lightskyblue4	olivedrab2	red2	snow2
aquamarine2	firebrick	gray55	indianred1	lightskyblue3	olivedrab1	red1	snow1
aquamarine1	dodgerblue4	gray54	indianred	lightskyblue2	oldlace	red	snow
aquamarine	dodgerblue3	gray53	hotpink4	lightskyblue1	navy	purple4	slategray
antiquewhite4	dodgerblue2	gray52	hotpink3	lightseagreen	navajowhite4	purple3	slategray4
antiquewhite3	dodgerblue1	gray51	hotpink2	lightsalmon4	navajowhite3	purple2	slategray3
antiquewhite2	dimgray	gray50	hotpink1	lightsalmon3	navajowhite2	purple1	slategray2
antiquewhite1	dimgray	gray49	honeydew4	lightsalmon2	navajowhite1	purple	
aliceblue	deeppink4	gray48	honeydew3	lightsalmon1	navajowhite	powderblue	
white	deeppink3	gray47	honeydew2	lightsalmon	moccasin	plum4	
	deeppink2	gray46	honeydew1	lightpink4	mistyrose4	plum3	
	deeppink1	gray45	gray100	lightpink3	mistyrose3	plum2	
	darkturquoise	gray44	greenyellow	lightpink2	mistyrose2	plum1	
	darkslategray	gray43	green4	lightpink1	gray100		
	darkslategray3	gray42	green3	gray99			
	darkslategray2	gray41	green2	gray98			
	darkslategray1	gray40	green1	gray97			
	darkcyan	gray39	green	gray96			
	darkblue	gray38	gray99	gray95			
	cyan4	gray37	gray98	gray94			
	cyan3	gray36	gray97	gray93			
	cyan2	gray35	gray96	gray92			
	cyan	gray34	gray95	gray91			
	cornsilk4	gray33	gray94	gray90			
	cornsilk3	gray32	gray93	gray89			
	cornsilk2	gray31	gray92	gray88			
	cornsilk1	gray30	gray91				
	cornflowerblue	gray29	gray90				
	coral4	gray28	gray89				

colorRamps and grDevices



colorRamps and grDevices color palette, display from: <http://bc.bojanorama.pl/2013/04/r-color-reference-sheet/>

colorspace defaults

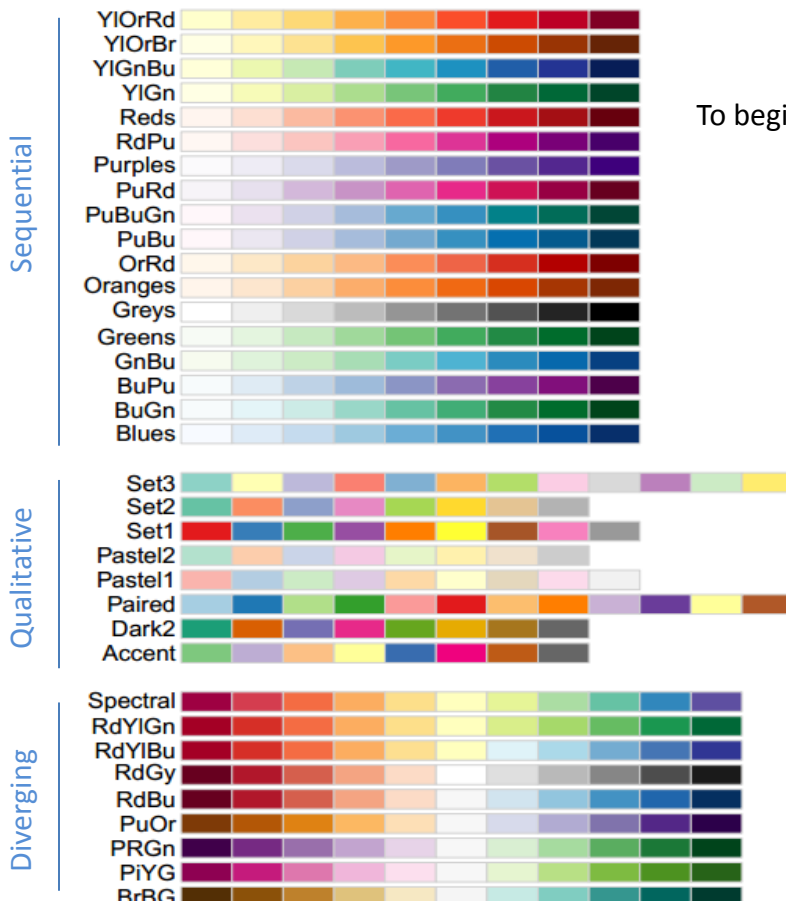


colorspace useful palette examples



To begin interactive color selector: `pal <- choose_palette()`

RColorBrewer



To display RColorBrewer palette: `display.brewer.all()`

For interactive color selector: <http://colorbrewer2.org/>

Useful Resources:

A larger color chart of R named colors:

<http://research.stowers-institute.org/efg/R/Color/Chart/ColorChart.pdf>

Nice overview of color in R:

<http://research.stowers-institute.org/efg/Report/UsingColorInR.pdf>

http://students.washington.edu/mclarkso/documents/colors_Ver2.pdf

A color theory reference:

Zeileis, A. K. Hornik, P. Murrell. 2009. Escaping RGBland: selecting colors for statistical graphics. *Computational and Statistics & Data Analysis* 53:3259-3270