



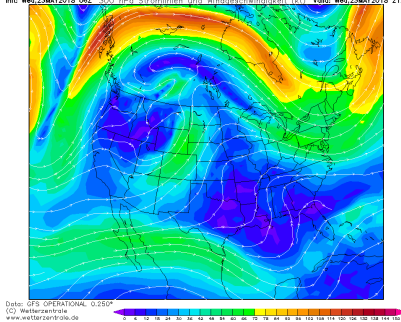
Somewhere Over the Rainbow: How to Make Effective Use of Colors in Scientific Visualizations

Reto Stauffer

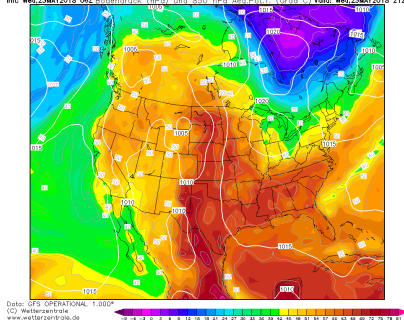
<http://hclwizard.org>

Introduction

Init: Wed, 23MAY2018 06Z, 300 hPa, Strömlinien und Windschwindigkeit (kt), Valid: Wed, 23MAY2018 21Z



Init: Wed, 23MAY2018 06Z, Bodendruck (hPa) und 850 hPa Avg. Pct.T. (Grad C), Valid: Wed, 23MAY2018 21Z



Two 'random' forecast maps from wetterzentrale.de.

Introduction

Color:

- Integral element in graphical displays.
- Easily available in (statistical) software.
- Omnipresent in (electronic) publications: Technical reports, electronic journal articles, presentation slides.

Problem: Little guidance about how to choose appropriate colors for a particular visualization task.

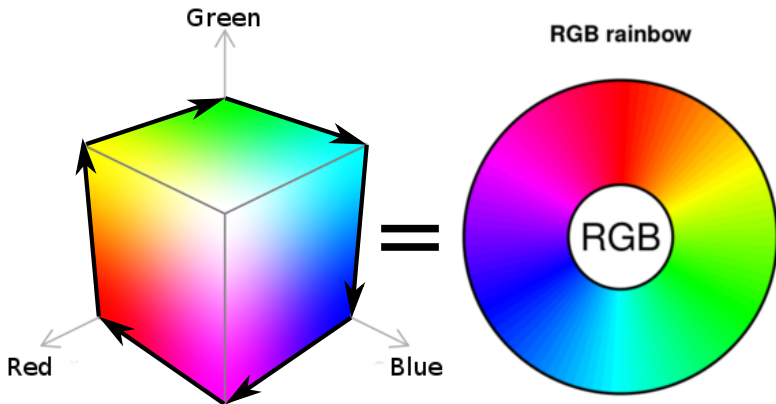
Question: What are useful color palettes for coding qualitative and quantitative variables?

Introduction

Main goal of our work:

- Raise awareness of the issue.
- Introduce Hue-Chroma-Luminance (HCL) model.
 - Based on human perception.
 - Better control for choosing color palettes.
- Provide convenient software for exploring and assessing HCL-based palettes.

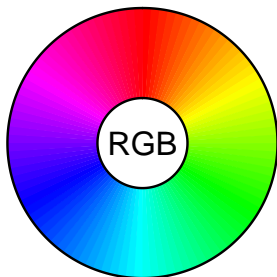
RGB Rainbow



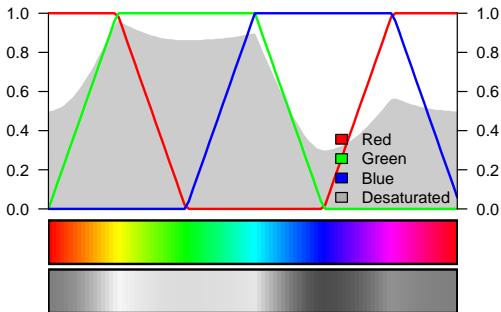
RGB color space: And the (in)famous rainbow color palette.

RGB Rainbow

RGB rainbow



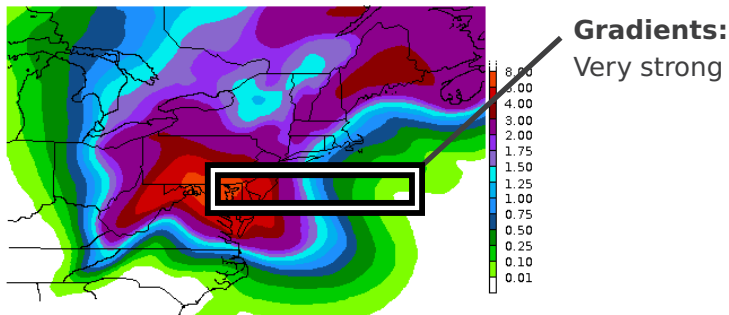
RGB rainbow spectrum



- The default color in many software packages.
- Conveniently used by many practitioners.
- Defaults only change slowly (if at all).

Question: Everybody does it – why should it be wrong?

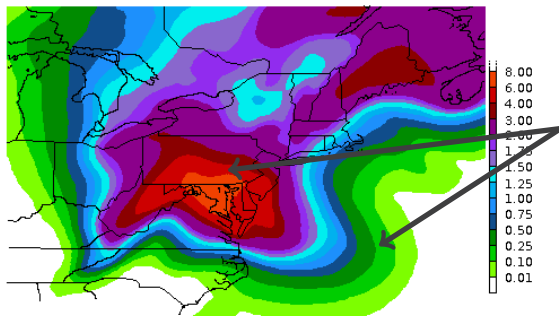
What's Wrong?



Hurricane Sandy
120-hour Day 1-5 Rainfall Forecast

Original figure as published by the NOAA.

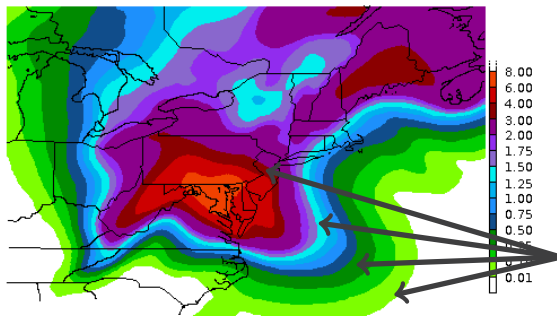
What's Wrong?



Hurricane Sandy
120-hour Day 1-5 Rainfall Forecast

Original figure as published by the NOAA.

What's Wrong?



Hurricane Sandy
120-hour Day 1-5 Rainfall Forecast

Gradients:

Very strong

Saturation

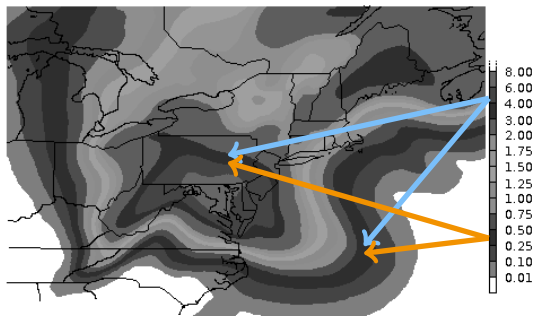
Highly-saturated
colors

Discontinuous

Bright, dark,
bright, dark, ...

Original figure as published by the NOAA.

What's Wrong?

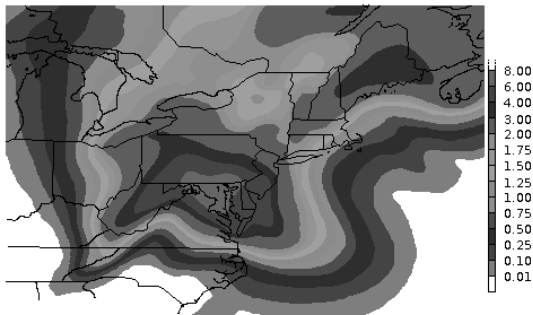


Hurricane Sandy
120-hour Day 1-5 Rainfall Forecast

Desaturated version of the original figure.

Assignment
No longer unique

What's Wrong?



Hurricane Sandy
120-hour Day 1-5 Rainfall Forecast

Desaturated version of the original figure.

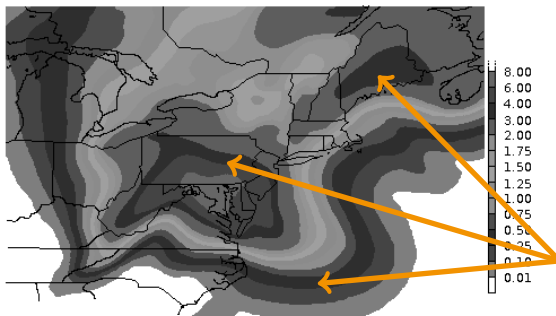
Assignment

No longer unique

Interpretation

Where is the maximum?

What's Wrong?



Hurricane Sandy
120-hour Day 1-5 Rainfall Forecast

Desaturated version of the original figure.

Assignment

No longer unique

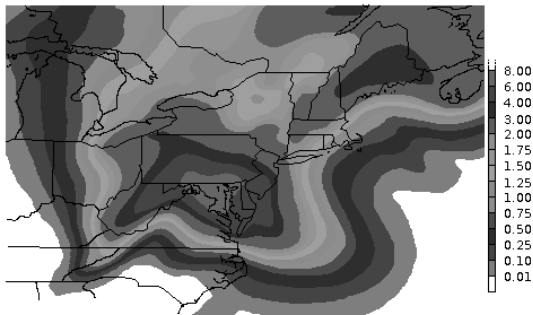
Interpretation

Where is the maximum?

Focus

On dark artefacts

What's Wrong?

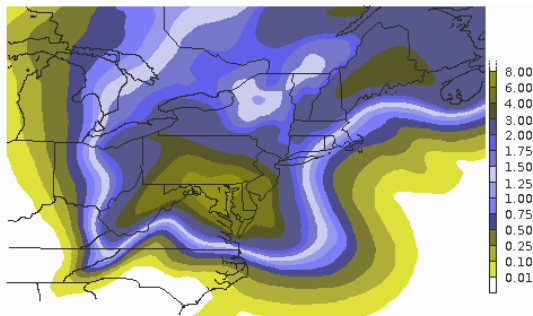


Hurricane Sandy
120-hour Day 1-5 Rainfall Forecast

Desaturated version of the original figure.



What's Wrong?



Hurricane Sandy
120-hour Day 1-5 Rainfall Forecast

End-user

Who is it?

Consider

Visual

constraints?

What color-blind people see (red-green weakness).

About **5% of all Europeans** are affected.

Challenges

Summary: The colors in a palette should

- be simple and natural,
- not be unappealing,
- highlight the important information,
- not mislead the reader,
- work everywhere and for everyone.

In practice:

- People often do not think about it at all.
- ... and simply use default colors.

Potential problems:

- For end users – reviewers, supervisor, colleague, customer.
- For your own day-to-day work.



The Hue-Chroma-Luminance Color Space

A Perception-Based Color Space

Perception-Based Way: HCL

Advantages:

- Hue: Type of color.
- Chroma: Colorfulness.
- Luminance: Brightness.

Hue



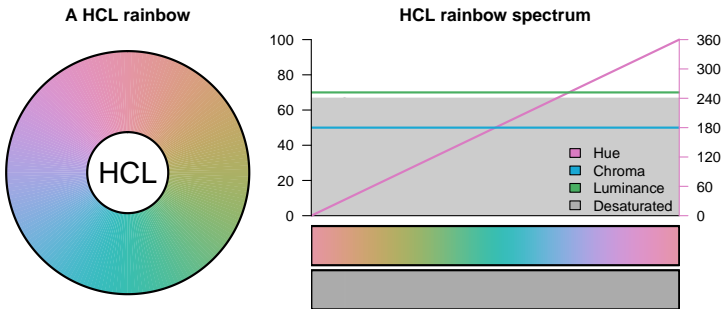
Chroma



Luminance

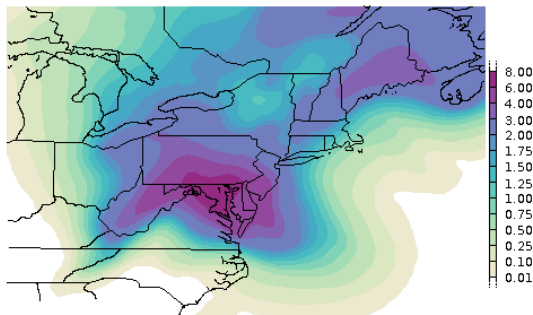


Perception-Based Way: HCL



- **H**ue (*defines the color*)
- **C**hroma (*defines the coloriness*) and
- **L**uminance (*defines the brightness*)

HCL Version



Hurricane Sandy
120-hour Day 1-5 Rainfall Forecast

Same information, changed color scheme.

Colors:

Smooth
gradients.

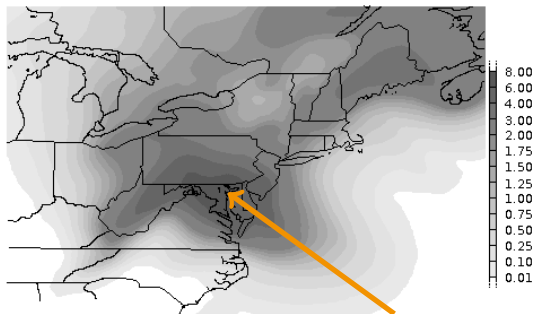
Information:

Guiding, no hidden
information.

Works:

Screen,
projector,
gray-scaled
device.

HCL Version



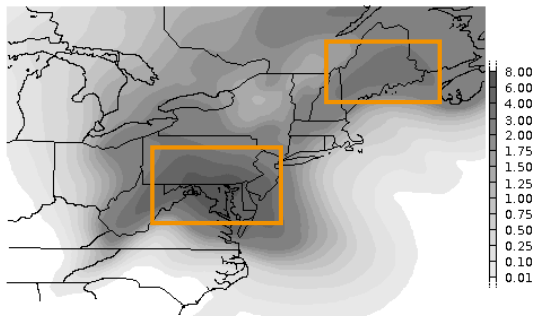
Hurricane Sandy
120-hour Day 1-5 Rainfall Forecast

Assignment:

Higher values \Rightarrow
lower luminance.

Desaturated representation of the HCL-version.

HCL Version



Hurricane Sandy
120-hour Day 1-5 Rainfall Forecast

Desaturated representation of the HCL-version.

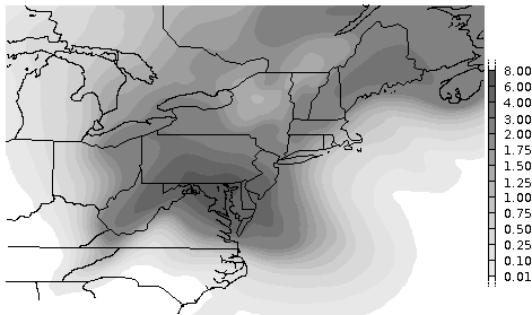
Assignment:

Higher values \Rightarrow
lower luminance.

Focus:

leads readers to
most important ar-
eas.

HCL Version



Hurricane Sandy
120-hour Day 1-5 Rainfall Forecast

Assignment:

Higher values \Rightarrow
lower luminance.

Focus:

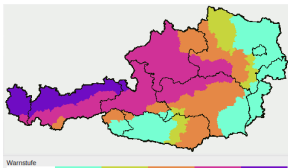
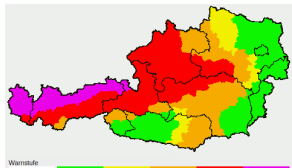
leads readers to
most important ar-
eas.

Desaturated representation of the HCL-version.

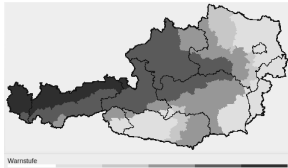
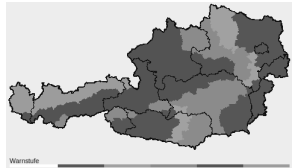
Summary:

Solved a lot of problems by changing the color palette.

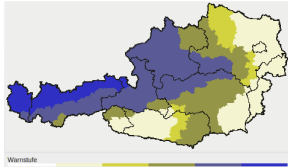
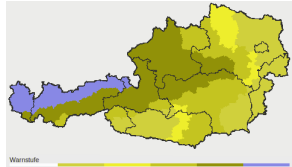
Warning Map Example



Colorized
Original (left)
HCL idea (right)



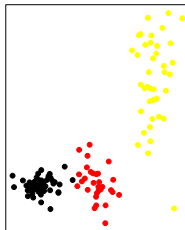
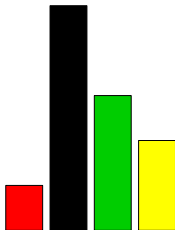
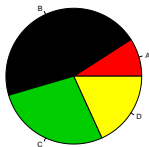
Gray-scale



Deuteranopia
Red-Green
weakness

Color Palettes: Qualitative

Goal: Code quantitative data.



Color Palettes: Qualitative

Goal: Code quantitative data.

dynamic [30, 300]



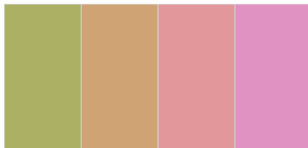
harmonic [60, 240]



cold [270, 150]

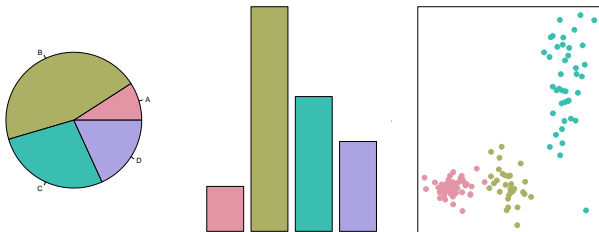


warm [90, -30]



Color Palettes: Qualitative

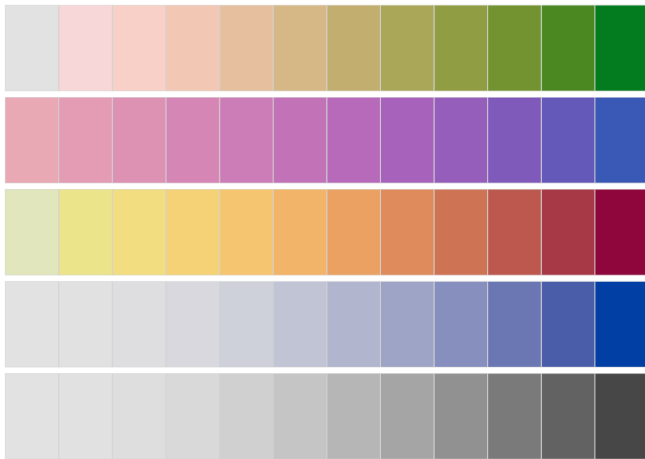
Goal: Code quantitative data.



Solution: Take colors with different hues, but keep chroma and luminance constant. E.g.: $(H, 50, 70)$

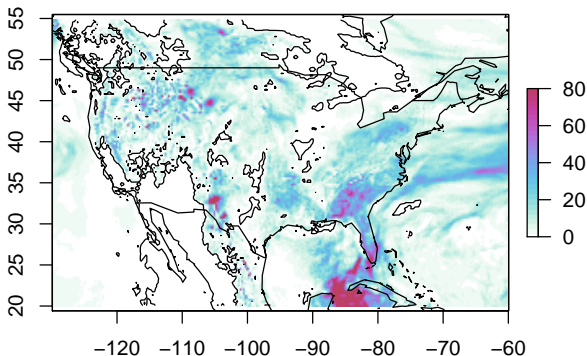
Color Palettes: Sequential

Goal: Code quantitative data (e.g., probabilities) where one side is of main interest.



Color Palettes: Sequential

Goal: Code quantitative data (e.g., probabilities) where one side is of main interest.



Solution: Constant hue and changing chroma/luminance.
E.g., (90 – 0, 30 – 100, 90 – 50).

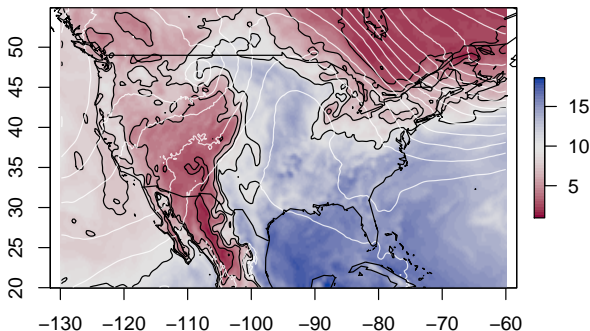
Color Palettes: Diverging

Goal: Code quantitative data and highlight both ends of the spectrum (e.g., anomalies, wet/dry, probabilities, ...).



Color Palettes: Diverging

Goal: Code quantitative data and highlight both ends of the spectrum (e.g., anomalies, wet/dry, probabilities, ...).

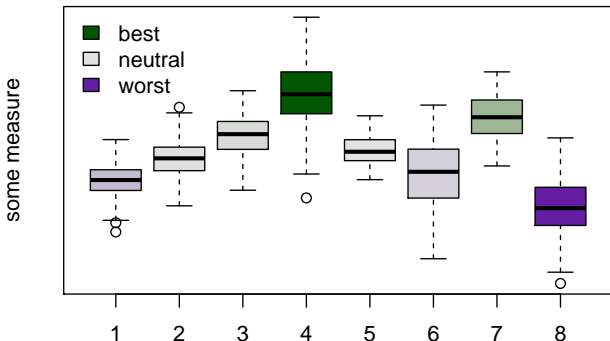


Solution: Diverging color schemes; combine sequential schemes with smooth transition.

Color Palettes: Diverging

Goal: Code quantitative data and highlight both ends of the spectrum (e.g., anomalies, wet/dry, probabilities, ...).

Model Comparison



Solution: Diverging color schemes; combine sequential schemes with smooth transition.

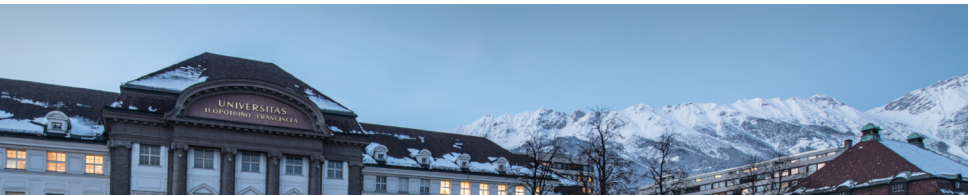
Experiences With Practitioners

In the beginning

- Hesitation of colleagues.
- “Not necessary!”
- “Why should we change existing products?”
- “Everybody does it like this”

A few days later

- Mainly positive feedback.
- Decrease of misinterpretations in classroom (“Weather & Forecast”).
- “Much easier to interpret”
- “How can I make use of those palettes (in my software)?”



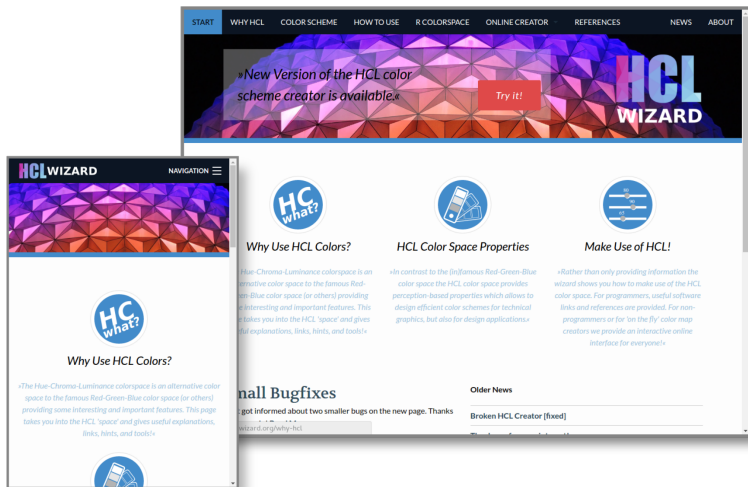
hclwizard.org

A Perception-Based Color Space

The R colorspace package

- Available on c-ran.
- Provides methods to explore, choose, and assess HCL based color maps.

For non-R users: We set up a website to provide the same interface to everyone. Just visit: <http://hclwizard.org>.



START WHY HCL COLOR SCHEME HOW TO USE R COLORSPACE ONLINE CREATOR REFERENCES NEWS ABOUT

»New Version of the HCL color scheme creator is available.« Try it!

HCLWIZARD

NAVIGATION

R colorspace Package

Using the colorspace package in R

The open source software R provides a package called (Ihaka, Murrell, Hornik, Fisher, & Ellis, 2016) which uses Ihaka's colorspace library. The package offers some preset color palettes (rainbow_hcl, terrain_hcl, heat_hcl, ...) to compare to the default color palettes. Furthermore there is a graphical user interface (GUI) where you can design your own color palette. The function therefore is called choose_palette() (needs some dependent packages) and use them for your own work.

The first thing you need is an R installation on your computer. The installation packages for all available operating systems (Windows, OSX, Linux) can be found on the <http://cran.r-hcl.org>. The installation is no adventure and the R base version just needs a few MB of space. For beginners the R-Studio GUI is recommended (or a similar R editor).

R is successfully installed on your system you can install optional packages like the colorspace package. You can simply do that over your GUI interface or install them by using following code line:

```
install.packages('colorspace')
```

colorspace Package

Using the colorspace package in R

The open source software R provides a package called (Ihaka, Murrell, Hornik, Fisher, & Ellis, 2016) which uses Ihaka's colorspace library. The package offers some preset color palettes (rainbow_hcl, terrain_hcl, heat_hcl, ...) to compare to the default color palettes. Furthermore there is a graphical user interface (GUI) where you can design your own color palette. The function therefore is called choose_palette() (needs some dependent packages) and use them for your own work.

The first thing you need is an R installation on your computer. The installation packages for all available operating systems (Windows, OSX, Linux) can be found on the <http://cran.r-hcl.org>. The installation is no adventure and the R base version just needs a few MB of space. For beginners the R-Studio GUI is recommended (or a similar R editor).

R is successfully installed on your system you can install optional packages like the colorspace package. You can simply do that over your GUI interface or install them by using following code line:

```
install.packages('colorspace')
```

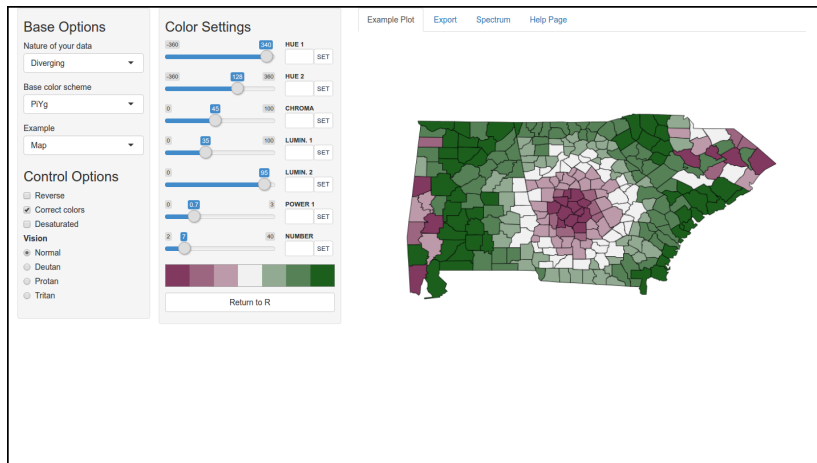
If I'm an advanced R-user, but I never heard about the colorspace package. How can I use this awesome feature?

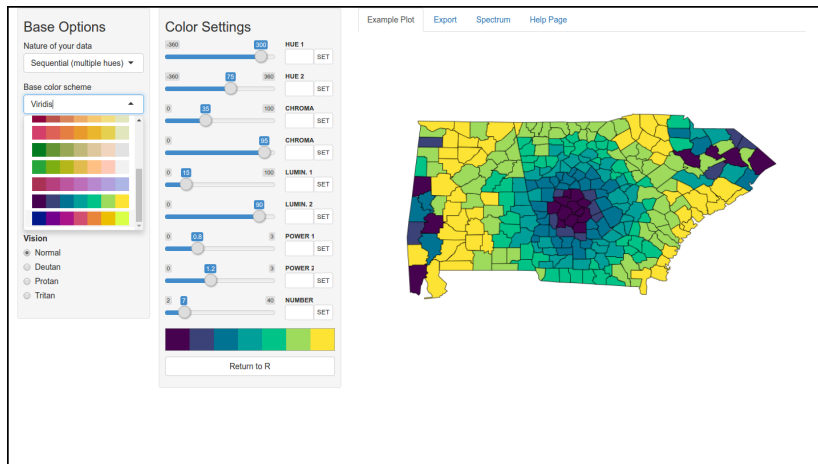
Download R Package

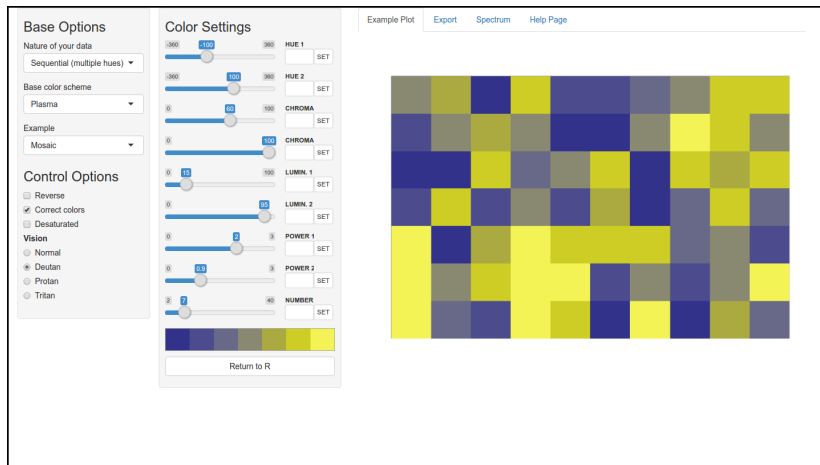
The current version of the R colorspace package can be downloaded from the cran website: Visit: [cran package colorspace](http://cran.r-hcl.org)

R-Forge

R-forge Repository







Base Options

Nature of your data

Sequential (multiple hues) ▾

Base color scheme

Plasma ▾

Example

Mosaic ▾

Control Options

☐ Reverse

☒ Correct colors

☐ Desaturated

Vision

☒ Normal

☐ Deutan

☐ Protan

☐ Tritan

Color Settings

HUE 1: -360 100 360 SET

HUE 2: -360 100 360 SET

CHROMA: 0 80 100 SET

CHROMA: 0 100 100 SET

LUMIN. 1: 0 15 100 SET

LUMIN. 2: 0 55 100 SET

POWER 1: 0 3 3 SET

POWER 2: 0 0.8 3 SET

NUMBER: 2 22 40 SET

Return to R

Example Plot Export Spectrum Help Page

RAW GrADS Python matlab

RGB values [0-1]	RGB values [0-255]	HEX colors, no alpha
0.000 0.094 0.537	0 24 137	#001889
0.173 0.047 0.537	44 12 137	#2C0C89
0.310 0.000 0.541	79 0 138	#4F008A
0.404 0.000 0.549	103 0 140	#67008C
0.482 0.000 0.553	123 0 141	#7B008D
0.553 0.000 0.553	141 0 141	#8D008D
0.612 0.000 0.545	156 0 139	#9C008B
0.671 0.078 0.533	171 20 136	#AB1488
0.718 0.149 0.518	183 38 132	#872684
0.765 0.212 0.494	195 54 126	#C3367E
0.804 0.275 0.463	205 70 118	#CD4676
0.839 0.333 0.420	214 85 107	#DE5568
0.871 0.396 0.373	222 101 95	#DE655F
0.894 0.459 0.310	228 117 79	#E4754F
0.910 0.522 0.227	232 133 58	#E8853A
0.922 0.588 0.098	235 150 25	#EB9619
0.929 0.651 0.000	237 166 0	#EDA600
0.925 0.718 0.000	236 183 0	#ECB700
0.918 0.784 0.000	234 200 0	#EACB00
0.902 0.855 0.000	230 218 0	#EGDA00
0.878 0.925 0.000	224 236 0	#E0EC00
0.855 1.000 0.278	218 255 71	#DAFF47

Color Map

Download Download Download

Base Options

Nature of your data

Sequential (multiple hues) ▾

Base color scheme

Plasma ▾

Example

Mosaic ▾

Control Options

☐ Reverse

☒ Correct colors

☐ Desaturated

Vision

☒ Normal

☐ Deutan

☐ Protan

☐ Tritan

Color Settings

HUE 1: -360 to 360, value: -100, SET

HUE 2: -360 to 360, value: 100, SET

CHROMA: 0 to 100, value: 80, SET

CHROMA: 0 to 100, value: 100, SET


LUMIN. 1: 0 to 100, value: 15, SET

LUMIN. 2: 0 to 100, value: 95, SET

POWER 1: 0 to 3, value: 2, SET

POWER 2: 0 to 3, value: 0.5, SET

NUMBER: 2 to 40, value: 22, SET



Return to R

Example Plot | Export | Spectrum | Help Page

RAW | GrADS | Python | matlab

```

** Define colors palette
'set rgb 20 0 24 137'
'set rgb 21 44 12 137'
'set rgb 22 79 0 138'
'set rgb 23 103 0 140'
'set rgb 24 123 0 141'
'set rgb 25 141 0 141'
'set rgb 26 156 0 139'
'set rgb 27 171 20 136'
'set rgb 28 183 38 132'
'set rgb 29 195 54 126'
'set rgb 30 205 70 118'
'set rgb 31 214 85 107'
'set rgb 32 222 101 95'
'set rgb 33 228 117 79'
'set rgb 34 232 133 58'
'set rgb 35 235 150 25'
'set rgb 36 237 166 0'
'set rgb 37 236 183 0'
'set rgb 38 234 200 0'
'set rgb 39 230 218 0'
'set rgb 40 224 236 0'
'set rgb 41 218 255 71'
'set ccols 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41'
'set clevs 0 5 10 15 20 25 30 35 40 45 50 55 60 65 70 75 80 85 90 95 100'

** Open data set via DOOS
** Open data set via DOOS
'sdlopen http://nomads.ncep.noaa.gov/9090/dods/gfs_lp00/gfs20170710/gfs_lp00_00z_an1'

** - Setting region to load
'set lon -20 40'
'set lat 30 85'

** - Map projection
'set mpdset hiroa'

```

127.0.0.1:3055/#tab-4440-2



The R colorspace Package

A Perception-Based Color Space

R colorspace

```
> library('colorspace')  
> # Interactively choosing color palettes  
> #  
> # Variant A:  
> # pal <- choose_palette()  
> #  
> # Variant B (requires shiny and shinyjs):  
> # pal <- hclwizard()
```

R colorspace

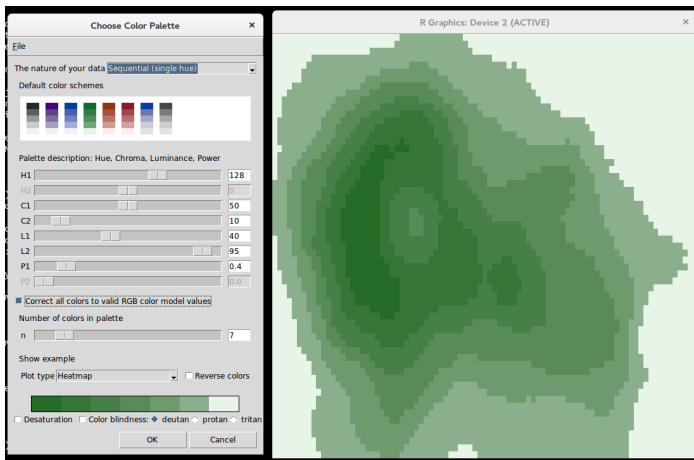


Figure: Screenshot of the tikz choose_palette interface.

R colorspace

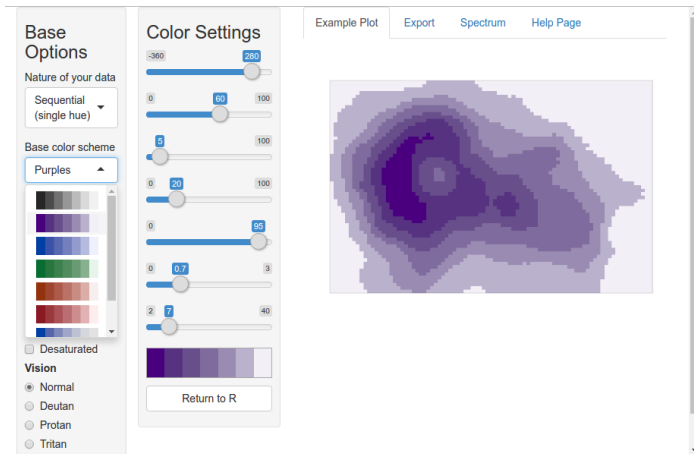


Figure: Screenshot of the hclwizard interface.

R colorspace

Use colorspace package on command-line level

```
> # choose_palette and hclwizard return a colormap function
> class(pal)
[1] "function"
> # function (n, h = c(12, 265), c = 80, l = c(25, 95), power = 0.7,
> #          fixup = TRUE, gamma = NULL, alpha = 1, ...)
```

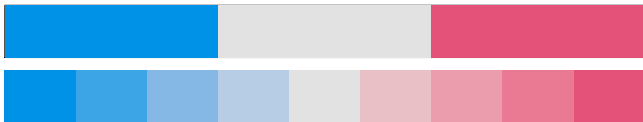
R colorspace

Use colorspace package on command-line level

```
> # choose_palette and hclwizard return a colormap function
> class(pal)
[1] "function"
> # function (n, h = c(12, 265), c = 80, l = c(25, 95), power = 0.7,
> #          fixup = TRUE, gamma = NULL, alpha = 1, ...)
```

Draw a color map with N colors:

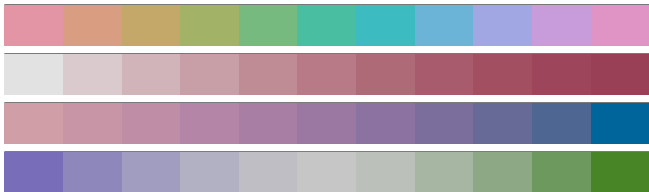
```
> pal(3)
[1] "#0092E7" "#E2E2E2" "#E45279"
> pal(9)
[1] "#0092E7" "#3BA5E6" "#86B8E5" "#B7CDE5" "#E2E2E2" "#EAC0C7" "#EC9D9D"
[8] "#EA7A93" "#E45279"
```



R colorspace

Basic colorspace wrapper methods:

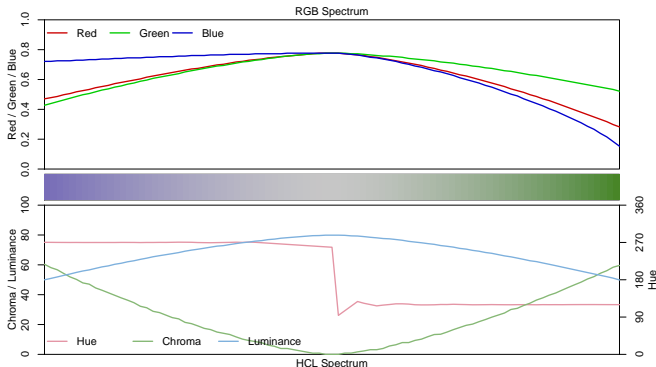
```
> qual <- rainbow_hcl(n=11)
> seq  <- sequential_hcl(n=11, h=0, l=c(90,40), c.=c(0,60))
> heat <- heat_hcl(n=11, h=c(0,-120), l=c(70,40), c.=c(30,60))
> div  <- diverge_hcl(n=11, h=c(270,120), c=60, l=c(50,80))
```



R colorspace

Assess the spectrum of a color map:

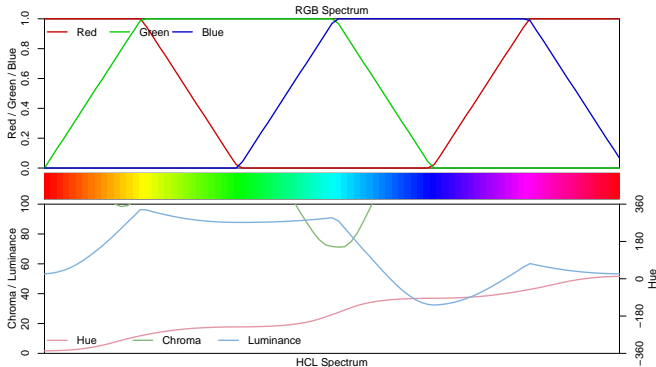
```
> div <- diverge_hcl(n=91, h=c(270,120), c=60, l=c(50,80))  
> specplot( div )
```



R colorspace

Assess the spectrum of a color map:

```
> rainbow <- rainbow(91)  
> specplot( rainbow )
```



R colorspace

Use colorspace to convert colors:

```
> div <- diverge_hcl(n=5, h=c(270,120), c=60, l=c(50,80))  
> RGB <- hex2RGB( div );      RGB
```

	R	G	B
[1,]	0.4705882	0.4274510	0.7215686
[2,]	0.6666667	0.6549020	0.7568627
[3,]	0.7764706	0.7764706	0.7764706
[4,]	0.6039216	0.6862745	0.5803922
[5,]	0.2823529	0.5215686	0.1529412

```
> # Convert to HCL  
> HCL <- as(RGB,"polarLUV");    HCL
```

	L	C	H
[1,]	49.96609	60.378199003	270.31045
[2,]	69.49331	20.598091195	270.26623
[3,]	79.88122	0.006140369	94.09931
[4,]	69.23223	21.080046408	119.96933
[5,]	49.85643	59.608000199	119.81271

R colorspace

One of the “core functions” is `polarLUV`:

```
> L <- seq(100, 30, length=12)
> C <- seq(40, 80, length=12)
> H <- rep( c(0,120,240), c(4,4,4) )
> HCL <- polarLUV(H=H, C=C, L=L)
```

R colorspace

One of the “core functions” is `polarLUV`:

```
> L <- seq(100, 30, length=12)
> C <- seq(40, 80, length=12)
> H <- rep( c(0,120,240), c(4,4,4) )
> HCL <- polarLUV(H=H, C=C, L=L)
```

Convert colors to hexadecimal representation:

```
> hexT <- hex( as(HCL,"RGB"), fixup=TRUE)
> hexF <- hex( as(HCL,"RGB"), fixup=FALSE)
```



R colorspace

One of the “core functions” is polarLUV:

```
> L <- seq(100, 30, length=12)
> C <- seq(40, 80, length=12)
> H <- rep( c(0,120,240), c(4,4,4) )
> HCL <- polarLUV(H=H, C=C, L=L)
```

What does the fixup=TRUE:

```
> as(HCL,"RGB")
```

	R	G	B
[1,]	1.50358973	0.85466492	0.95620852
[2,]	1.33963389	0.70133126	0.80120053
[3,]	1.18684461	0.56676651	0.66377437
[4,]	1.04495144	0.44977227	0.54287660
[5,]	0.27519179	0.56279146	0.20191278
[6,]	0.19728039	0.46266385	0.12965811
[7,]	0.13385835	0.37503736	0.07240051
[8,]	0.08354332	0.29909188	0.02861425
[9,]	-0.01705634	0.20360461	0.48078327
[10,]	-0.05184201	0.15500028	0.41481690
[11,]	-0.07965869	0.11583956	0.36140338
[12,]	-0.10459109	0.08556191	0.32440858

Summary

Choice of colors:

- Use color with care!
- Think about who the readers/users are.
- Avoid large areas of flashy, highly-saturated colors.
- Employ monotonic luminance scale for numerical data.

Try it yourself:

- <http://hclwizard.org>
- *colorspace* in R.

References

Zeileis, A., K. Hornik, and P. Murrell (2009): Escaping RGBland: Selecting colors for statistical graphics. *Computational Statistics & Data Analysis*, **53**(9), 3259–3270. <http://doi.org/10.1016/j.csda.2008.11.033>.

Stauffer, R., G.J. Mayr, M. Dabernig, and A. Zeileis (2015): Somewhere Over the Rainbow: How to Make Effective Use of Colors in Meteorological Visualizations. *Bulletin of the American Meteorological Society*, **96**(2), 203–216. <http://doi.org/10.1175/BAMS-D-13-00155.1>.

Ihaka R., P. Murrell, K. Hornik, J.C. Fisher, and A. Zeileis (2016): colorspace: Color Space Manipulation. R package version 1.3-2. URL <https://CRAN.R-project.org/package=colorspace>

Lumley T. (2006): Color Coding and Color Blindness in Statistical Graphics, *ASA Statistical Computing & Graphics Newsletter*, **17**(2), 4–7.

Ihaka R. (2003): Colour for Presentation Graphics, *Proceedings of the 3rd International Workshop on Distributed Statistical Computing*, Vienna (A), ISSN 1609-395X, www.ci.tuwien.ac.at/Conferences/DSC-2003/Proceedings.



Thank you for your attention!