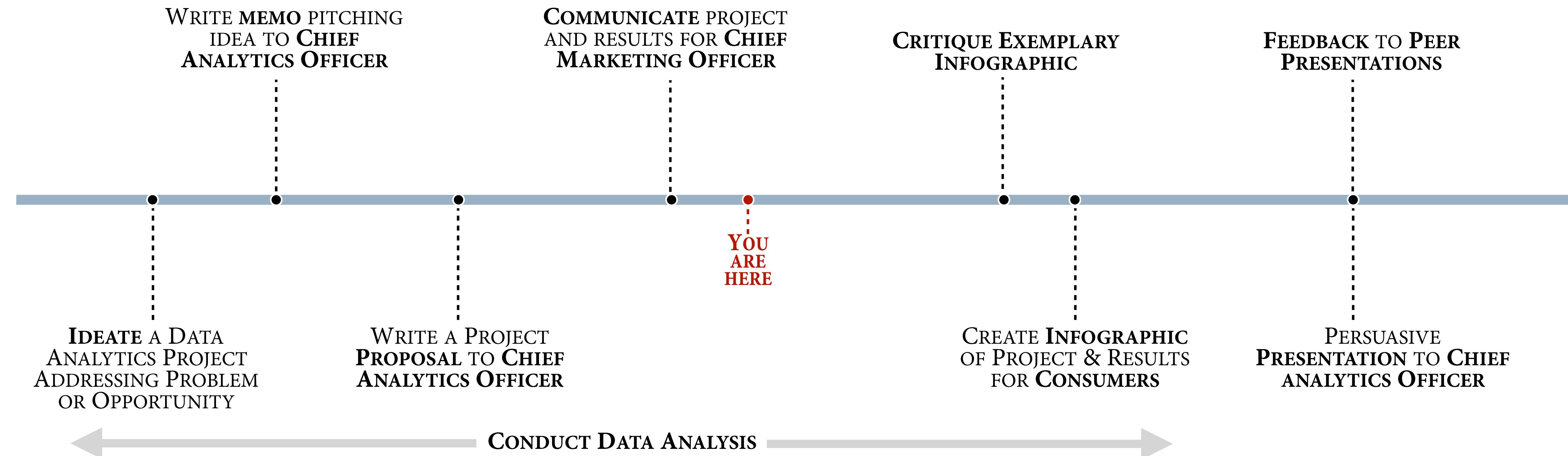


Storytelling With Data

**Effective visuals — encoding data  
graphically: grammar, layers, color**

# Conceptual project timeline



# **the grammar of graphics**

**DATA** : a set of data operations that create variables from datasets

**TRANSFORMATIONS** : variable transformations (*e.g., rank*)

**SCALES** : scale transformations (*e.g., log*)

**COORDINATES** : a coordinate system (*e.g., cartesian, polar*)

**ELEMENTS** : graphs (*e.g., points, lines*) and their aesthetic attributes (*e.g., color, opacity, shape, size, orientation*)

**GUIDES** : one or more guides (*axes, legends, etc.*)



# the grammar of graphics | *ggplot2 is the grammar of graphics implemented in R*

```
# load grammar of graphics
library(ggplot2)

p <-

# functions for data ink

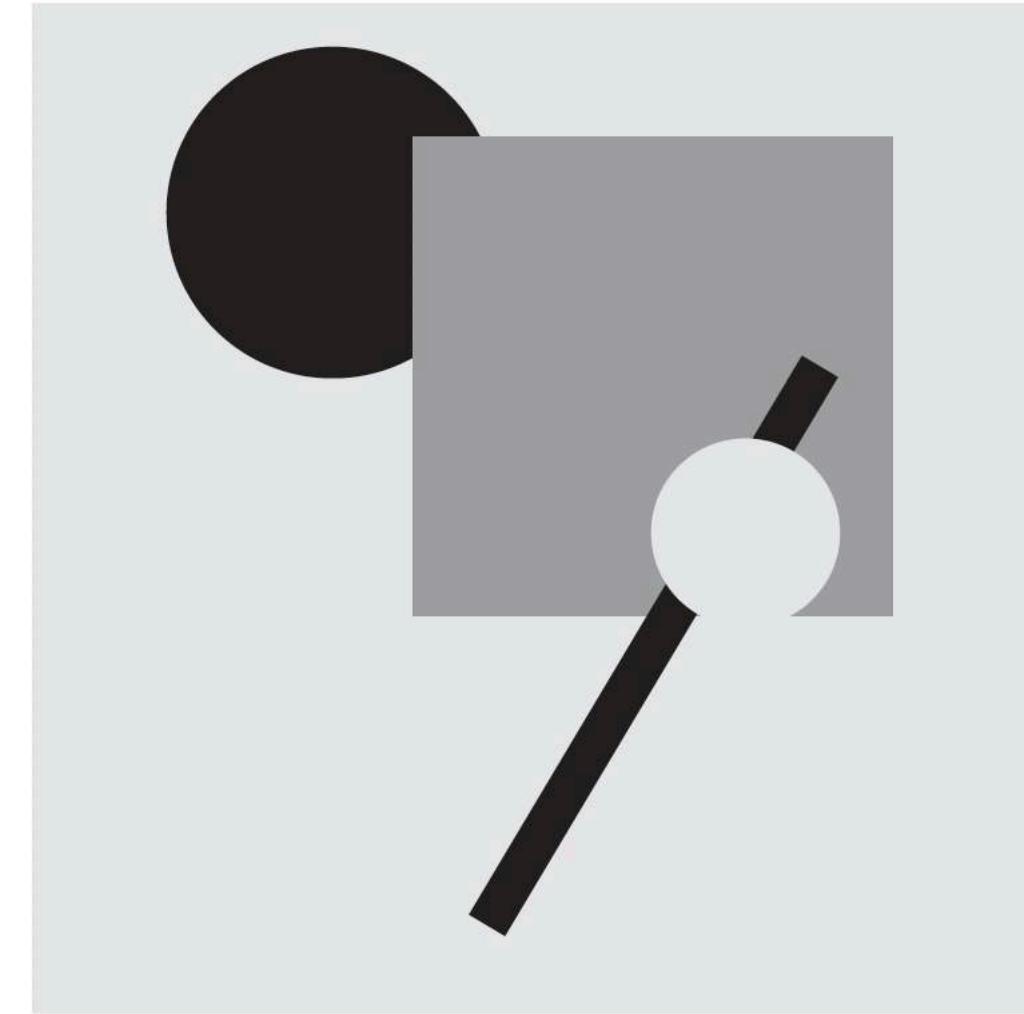
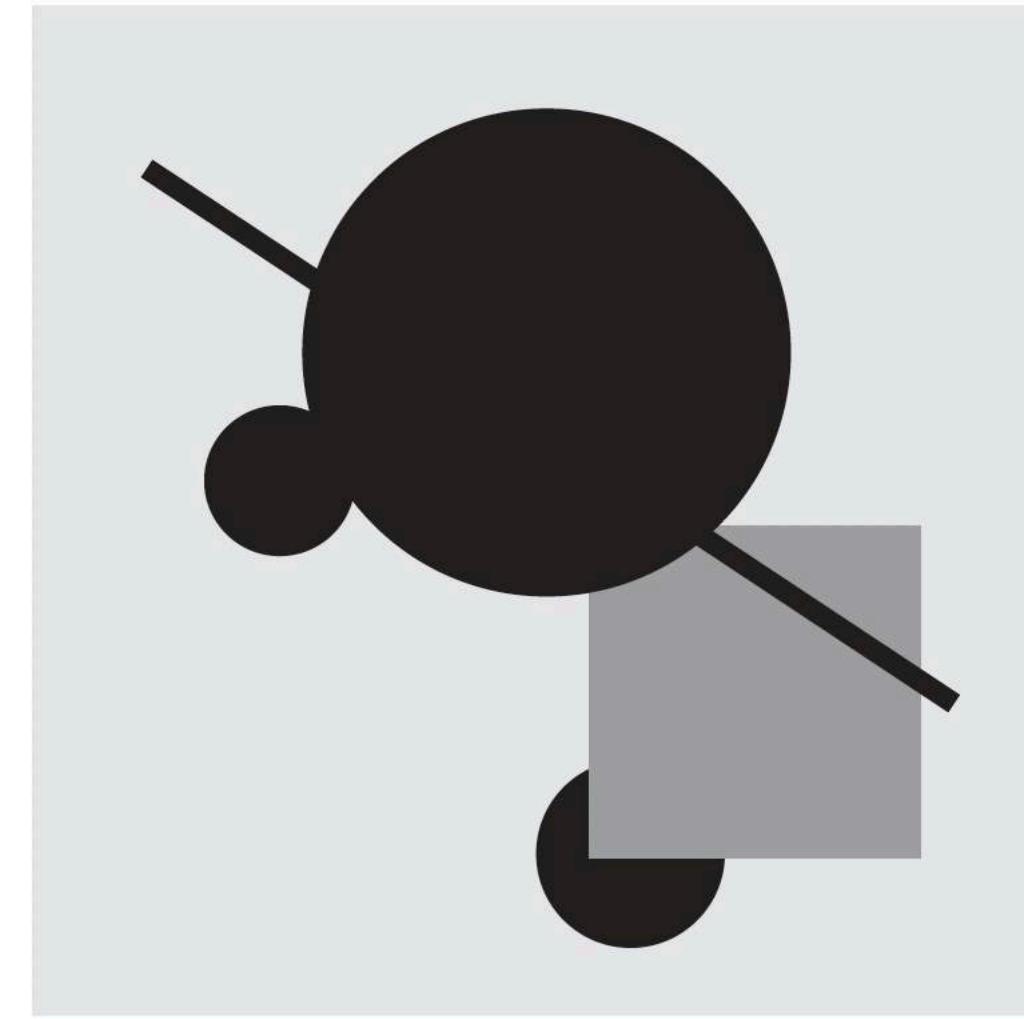
DATA      ggplot(data = <data>,
                mapping = aes(<aesthetic> = <variable>,
                            <aesthetic> = <variable>,
                            <...> = <...>)) +
TRANSFORMATIONS
ELEMENTS    geom_<type>(<...>) +
SCALES & GUIDES   scale_<mapping>_<type>(<...>) +
COORDINATES   coord_<type>(<...>) +
               facet_<type>(<...>) +
               <...> +
GUIDES       # functions for non-data ink
               labs(<...>) +
               theme(<...> = <...>) +
               annotate(<...>) +
               <...>
```

{ element\_blank()  
element\_line(<...> = <...>)  
element\_rect(<...> = <...>)  
element\_text(<...> = <...>)



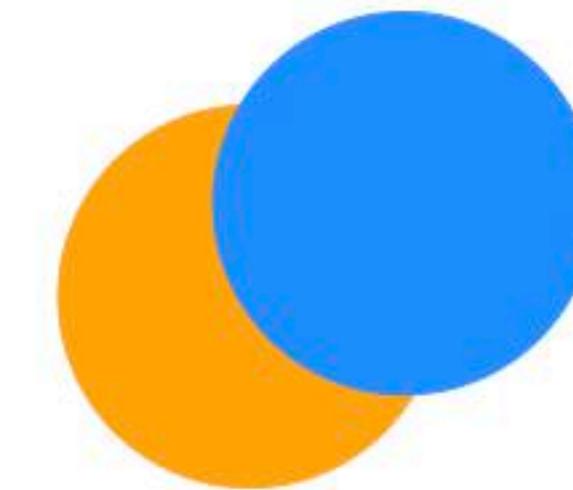
# **thinking about graphics as layers**

graphics as layers | *when one form or shape overlaps another, we see an illusion of near and far.*

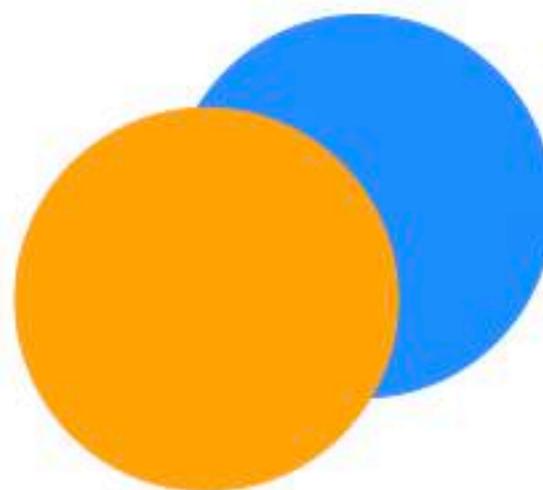


graphics as layers | *order of elements determines position towards reader and when overlapping, occlude.*

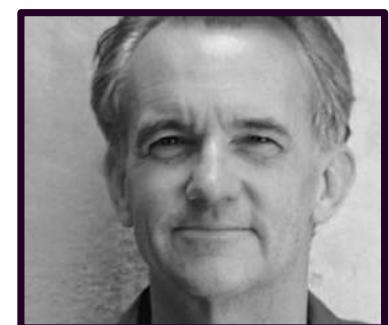
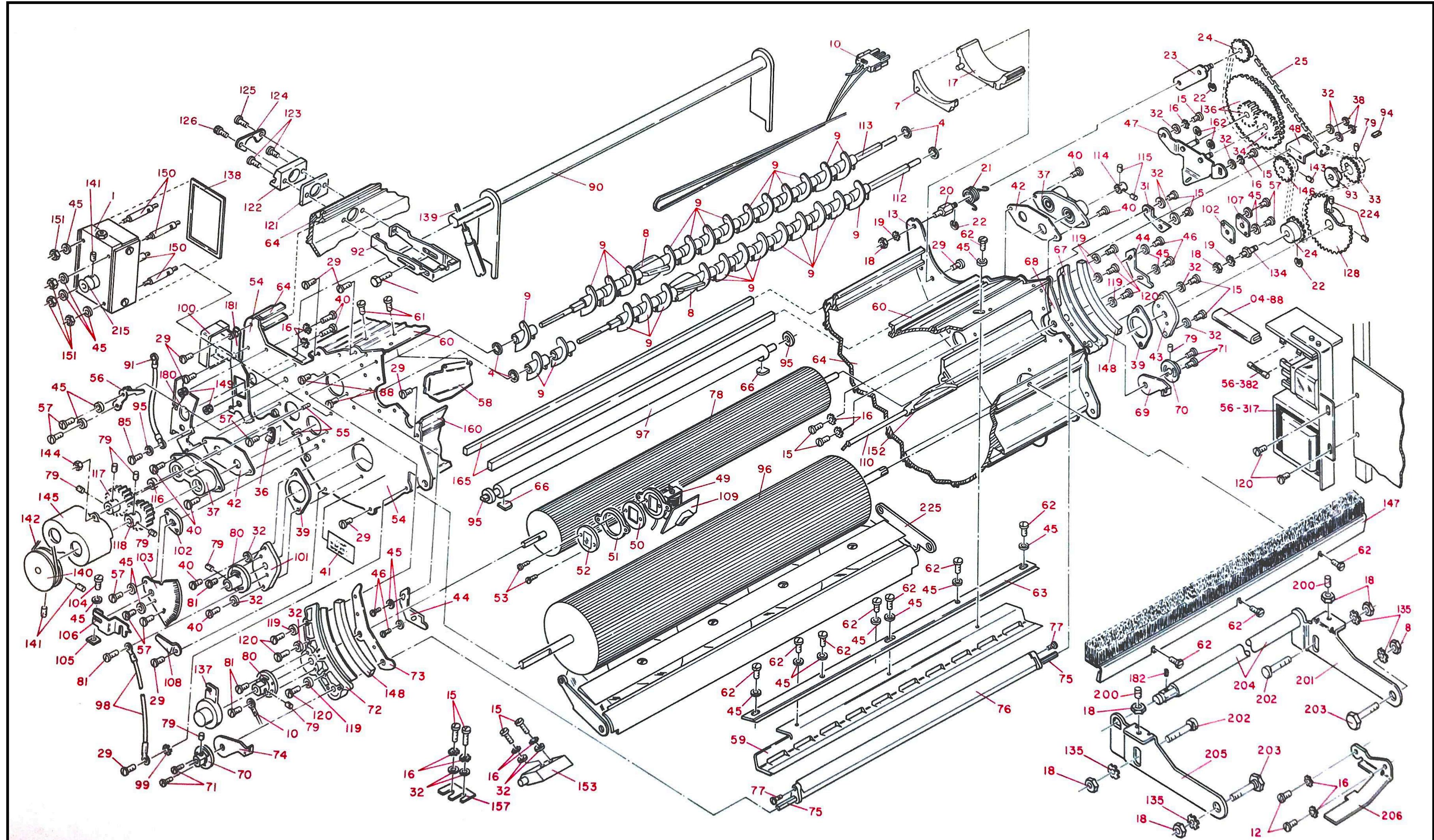
```
ggplot() +  
  theme_void() +  
  scale_x_continuous(limits = c(-5, 5)) +  
  scale_y_continuous(limits = c(-5, 5)) +  
  geom_point(aes(x = 0, y = 0),  
             size = 50, color = "orange") +  
  geom_point(aes(x = 1, y = 1),  
             size = 50, color = "dodgerblue")
```



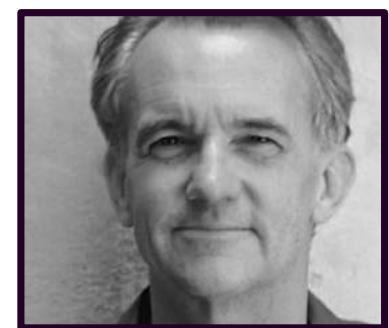
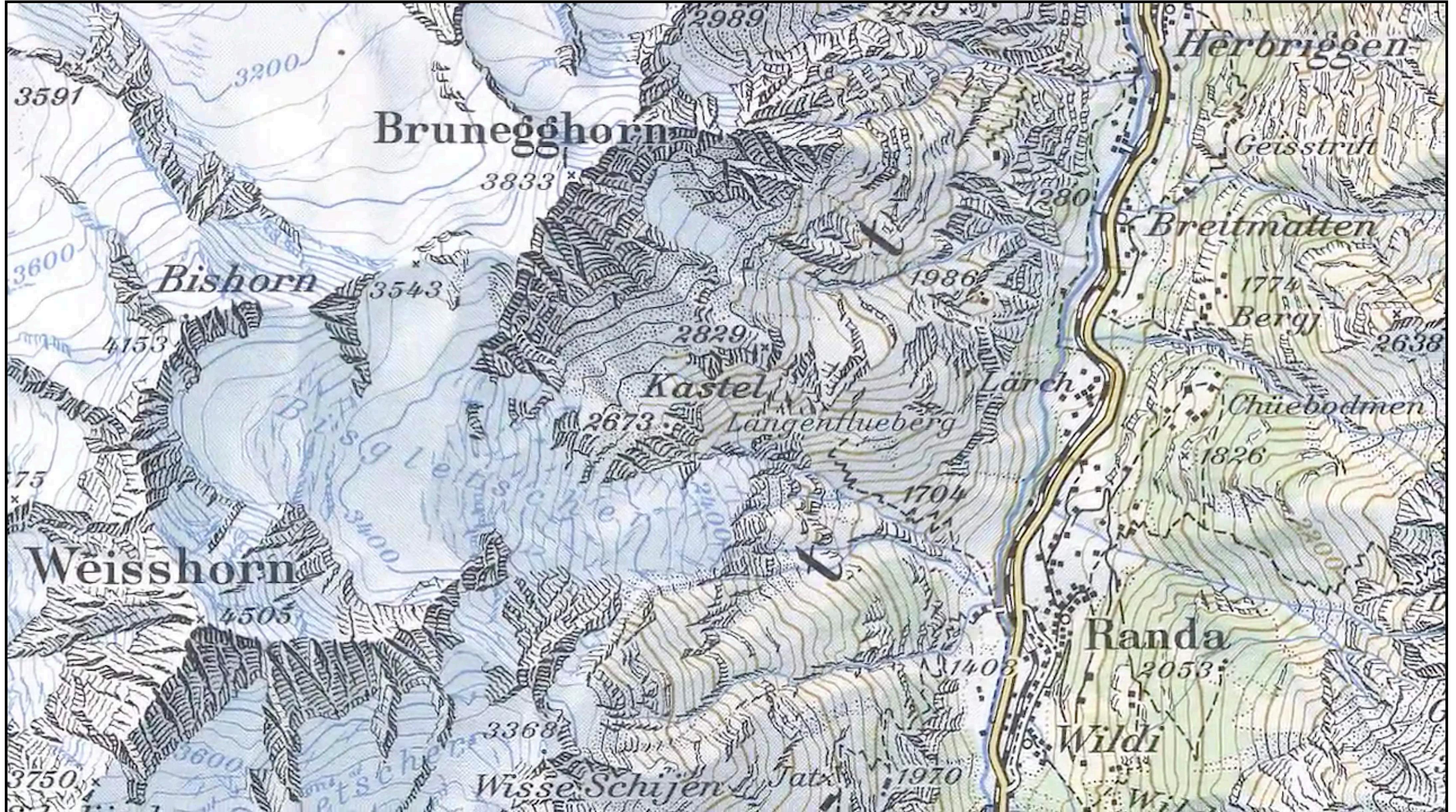
```
ggplot() +  
  theme_void() +  
  scale_x_continuous(limits = c(-5, 5)) +  
  scale_y_continuous(limits = c(-5, 5)) +  
  geom_point(aes(x = 1, y = 1),  
             size = 50, color = "dodgerblue") +  
  geom_point(aes(x = 0, y = 0),  
             size = 50, color = "orange")
```



graphics as layers | *layering by type of encoding, color in this case, helps separate information types.*



graphics as layers | *layering by type of encoding helps separate information types. Maps tend to be exemplary.*



graphics as layers | *in this example, observed data are separately layered “behind” scale of residuals.*

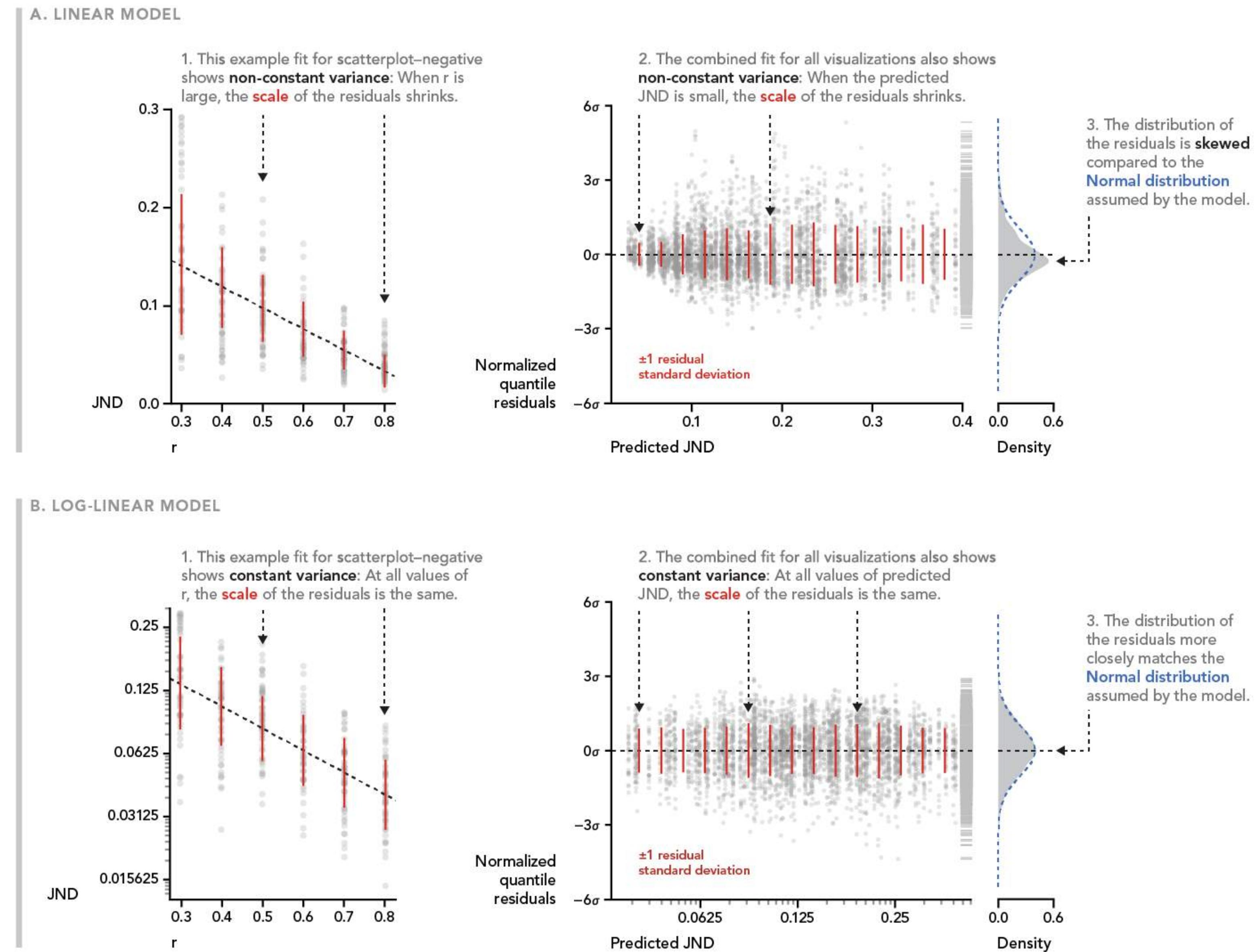
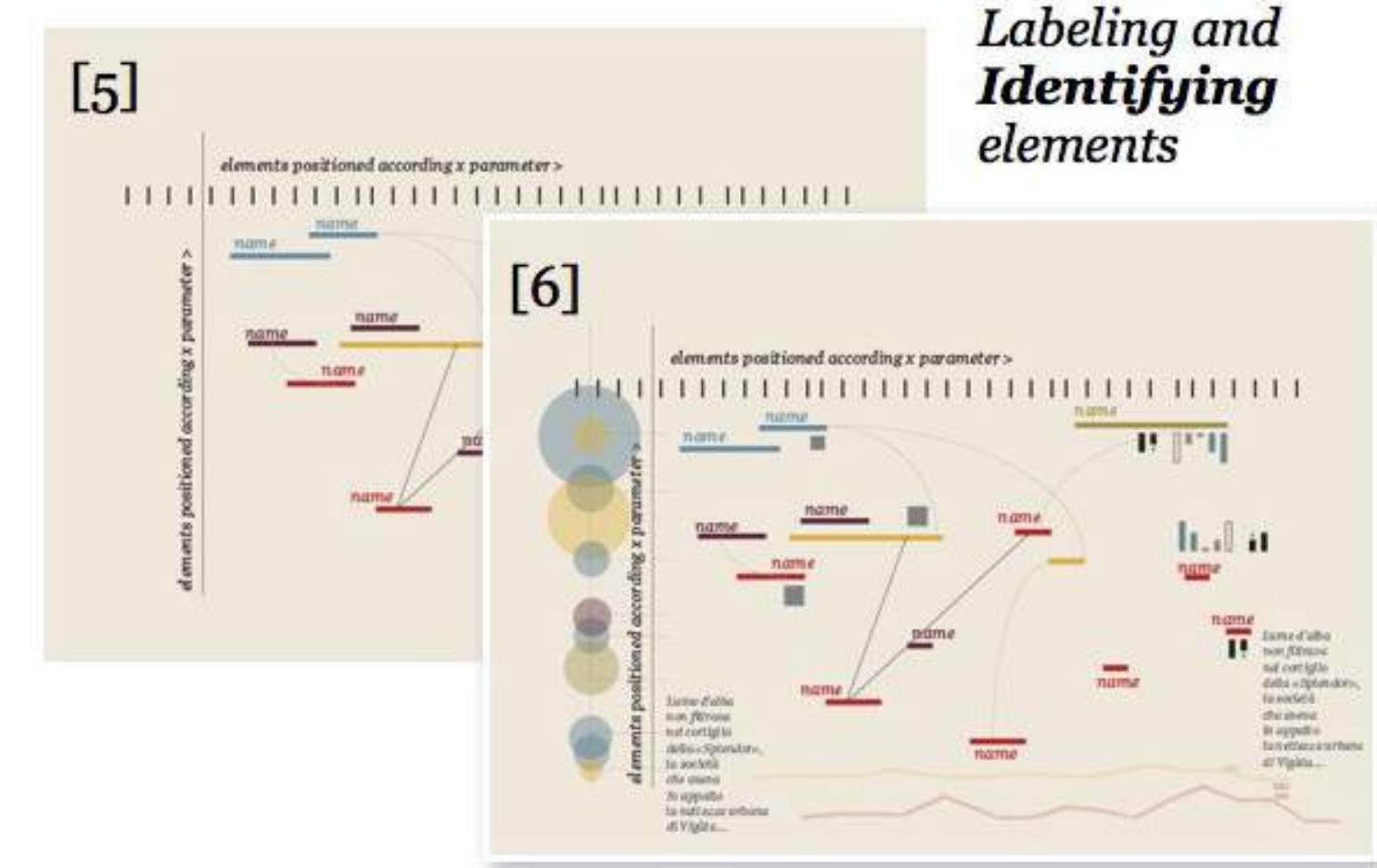
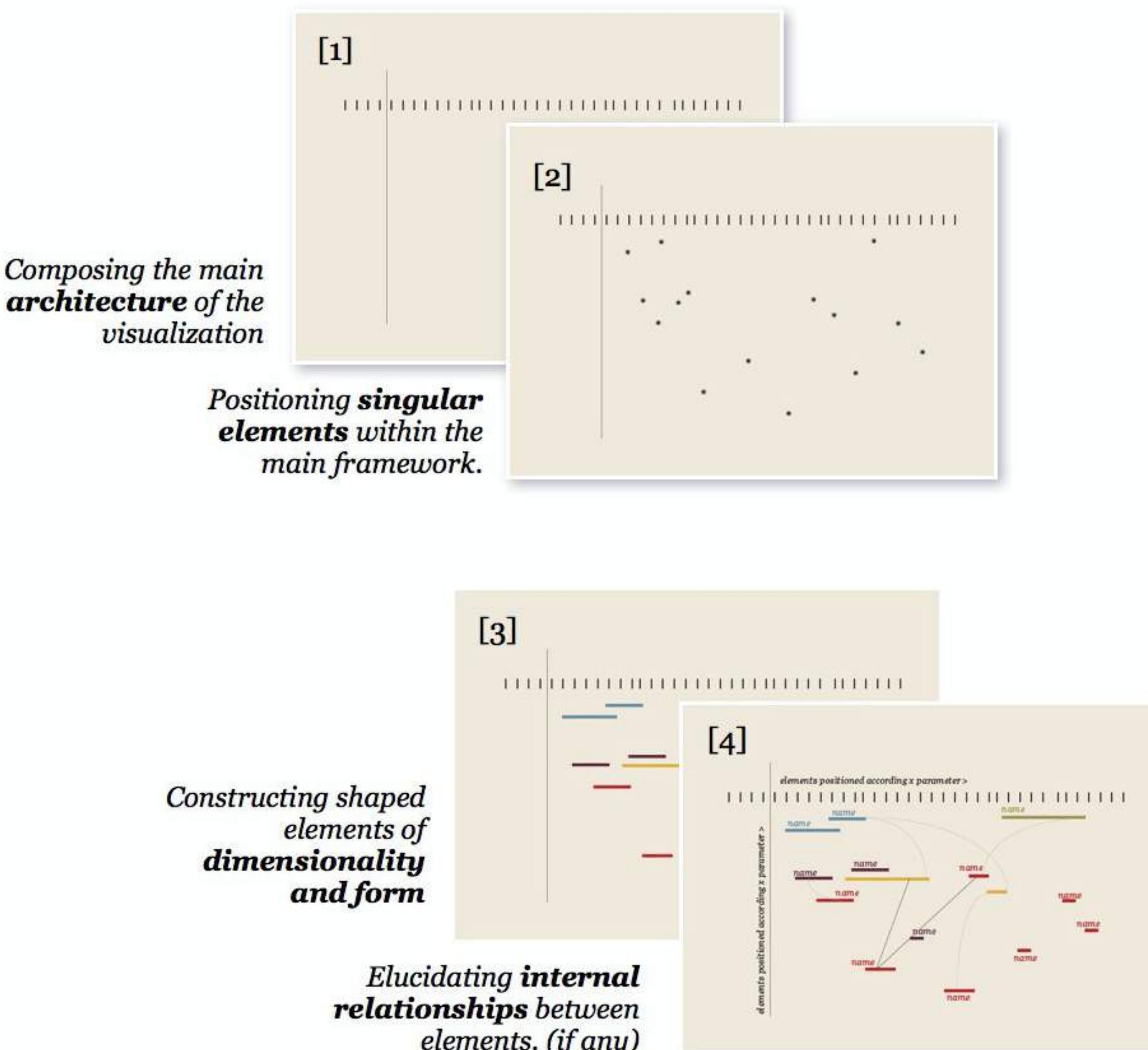
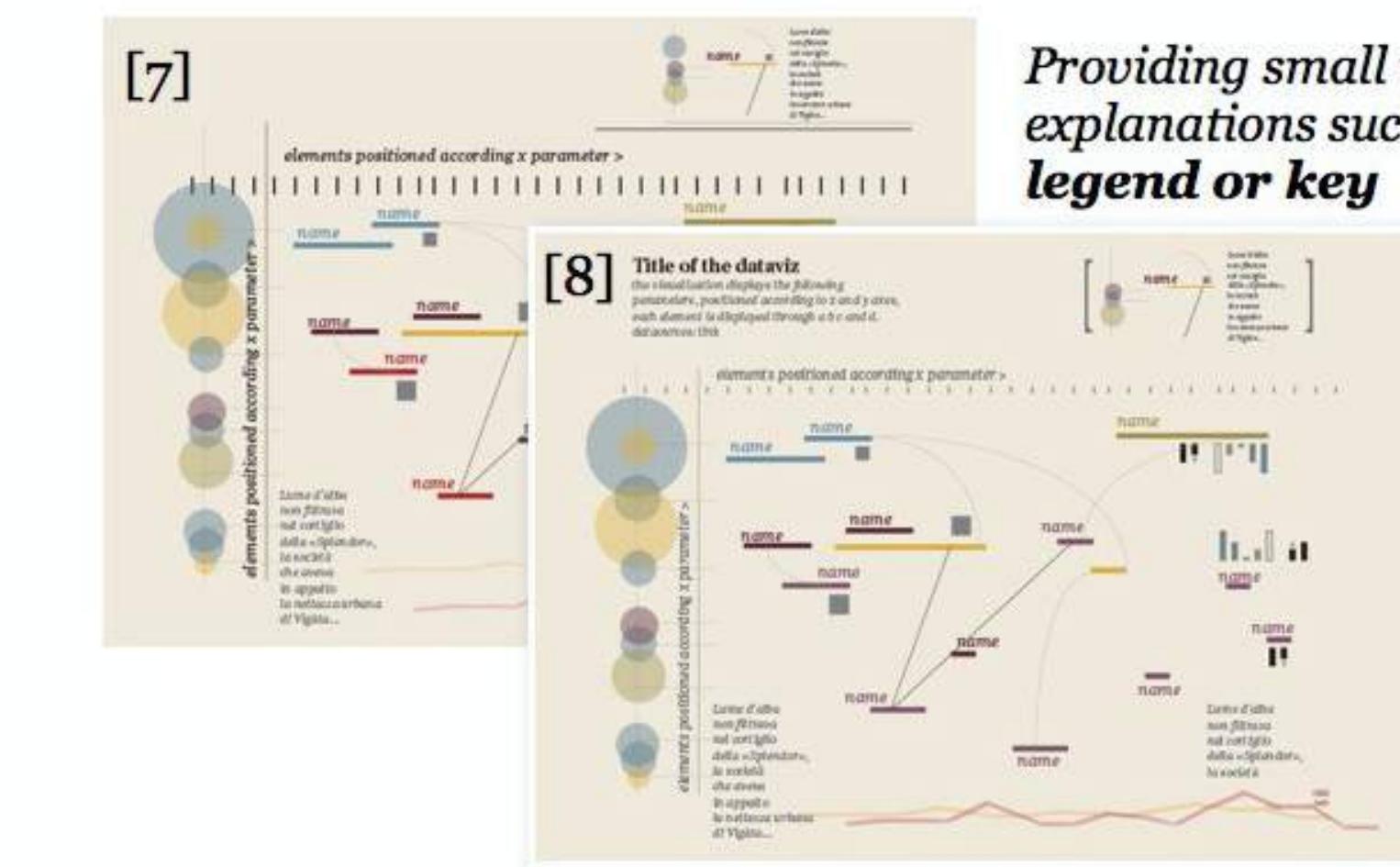


Fig. 3 Comparison of fits of the linear model (Section 3) and the log-linear model (Section 4). Example fits of each model to scatterplot-negative are shown in A.1 and B.1. Plots of normalized residuals for all visualization × direction pairs are shown in A.2 and B.2. Density plots of normalized residuals with comparison to the standard normal distribution are shown in A.3 and B.3.

# graphics as layers | *layering can create hierarchy and clarity in graphics narratives*

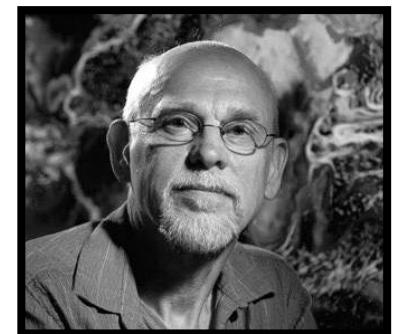
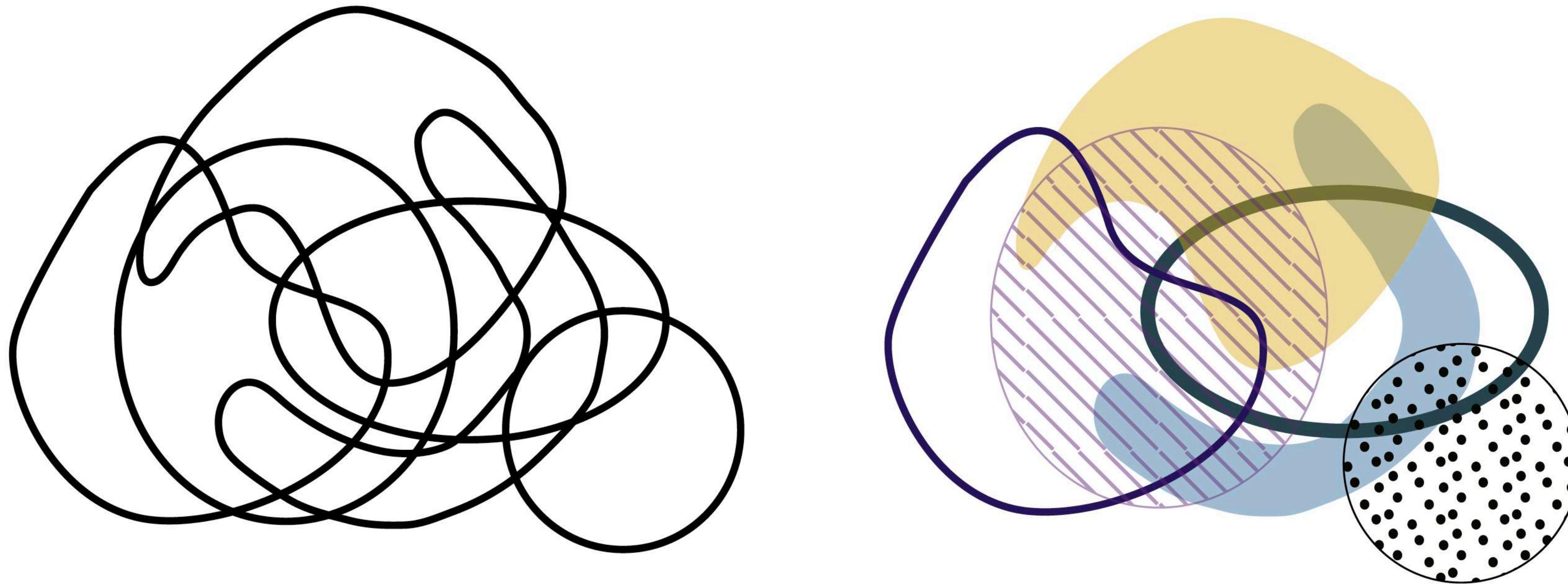


*Supplementing the greater story through the addition of “minor or tangential tales” elements*



*Fine-tuning and stylizing of elements shapes, colors, and weights to make hierarchies pop out.*

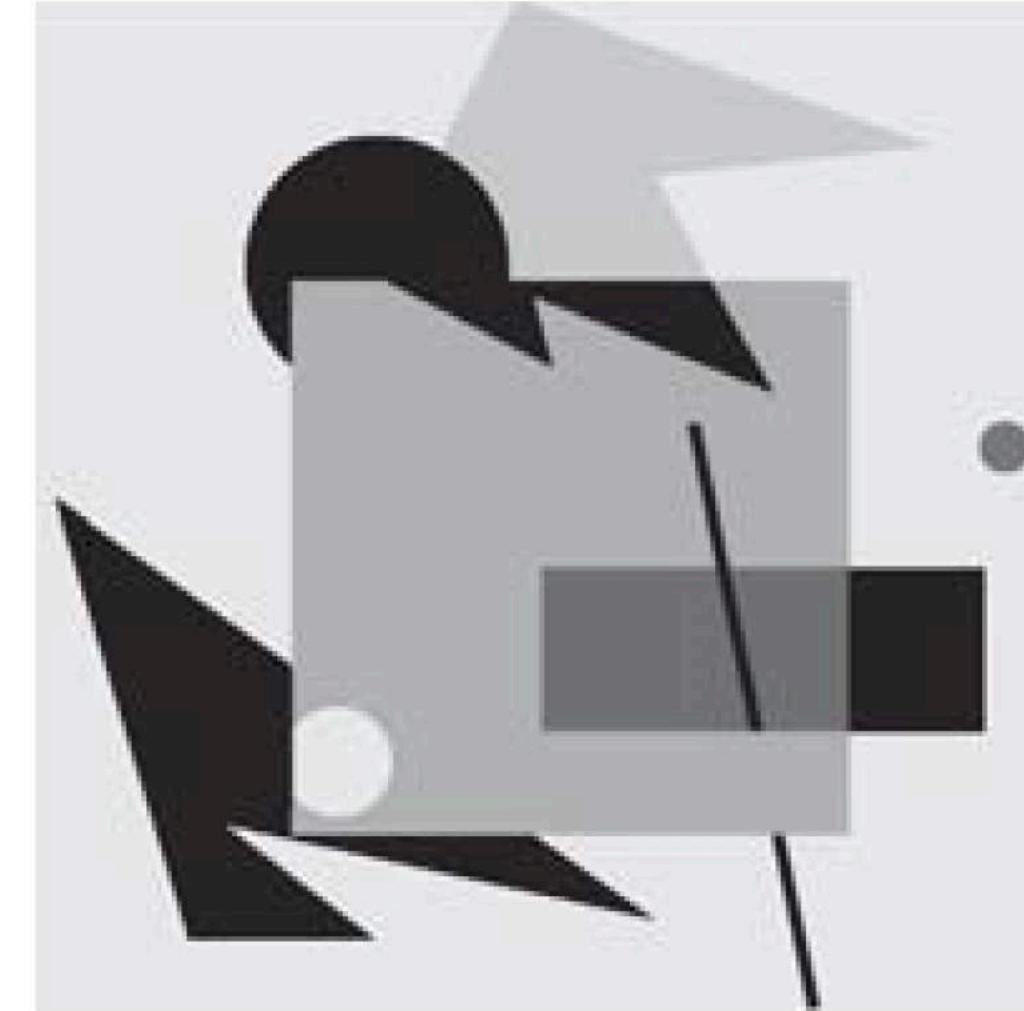
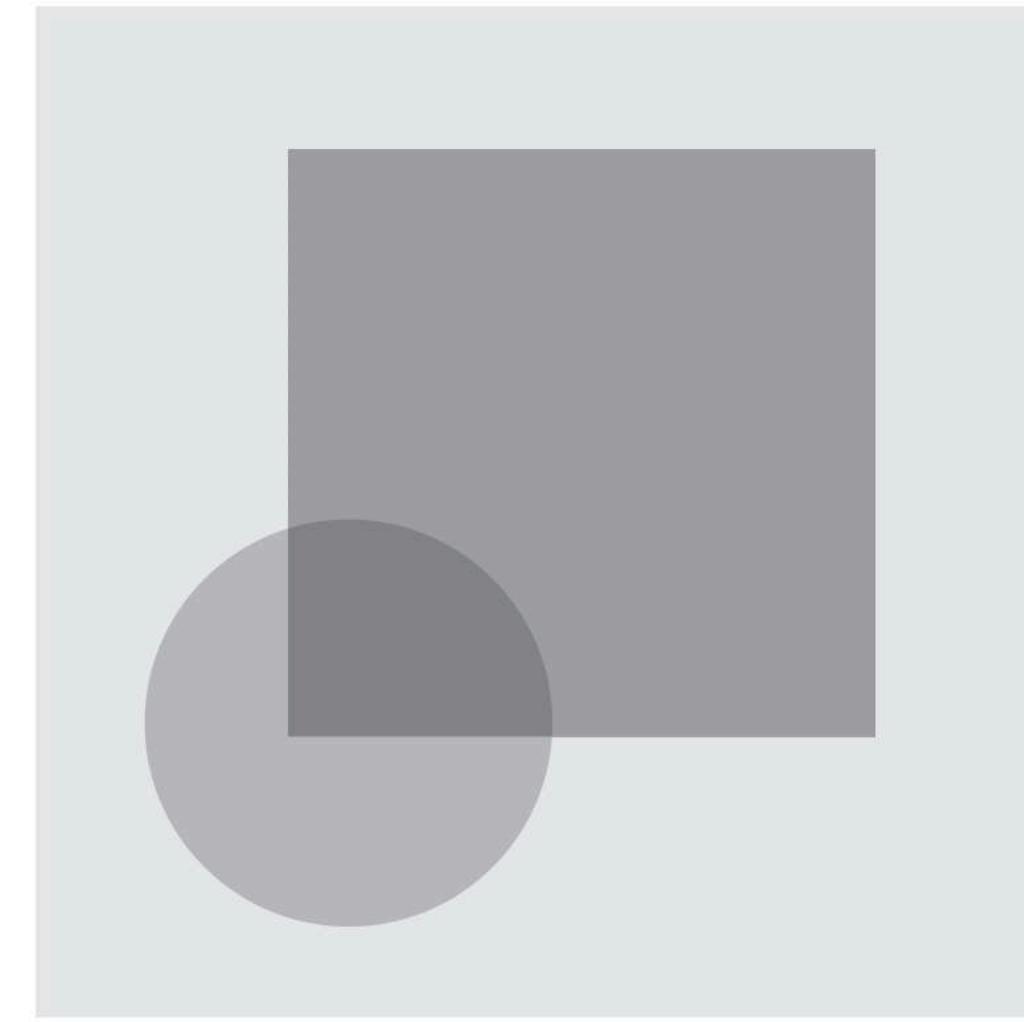
graphics as layers | *in this Euler diagram, color and texture help to distinguish layers of information*



---

Ware, Colin

graphics as layers | *transparency can enhance, or create conflicts with, the illusion of spatial arrangement.*



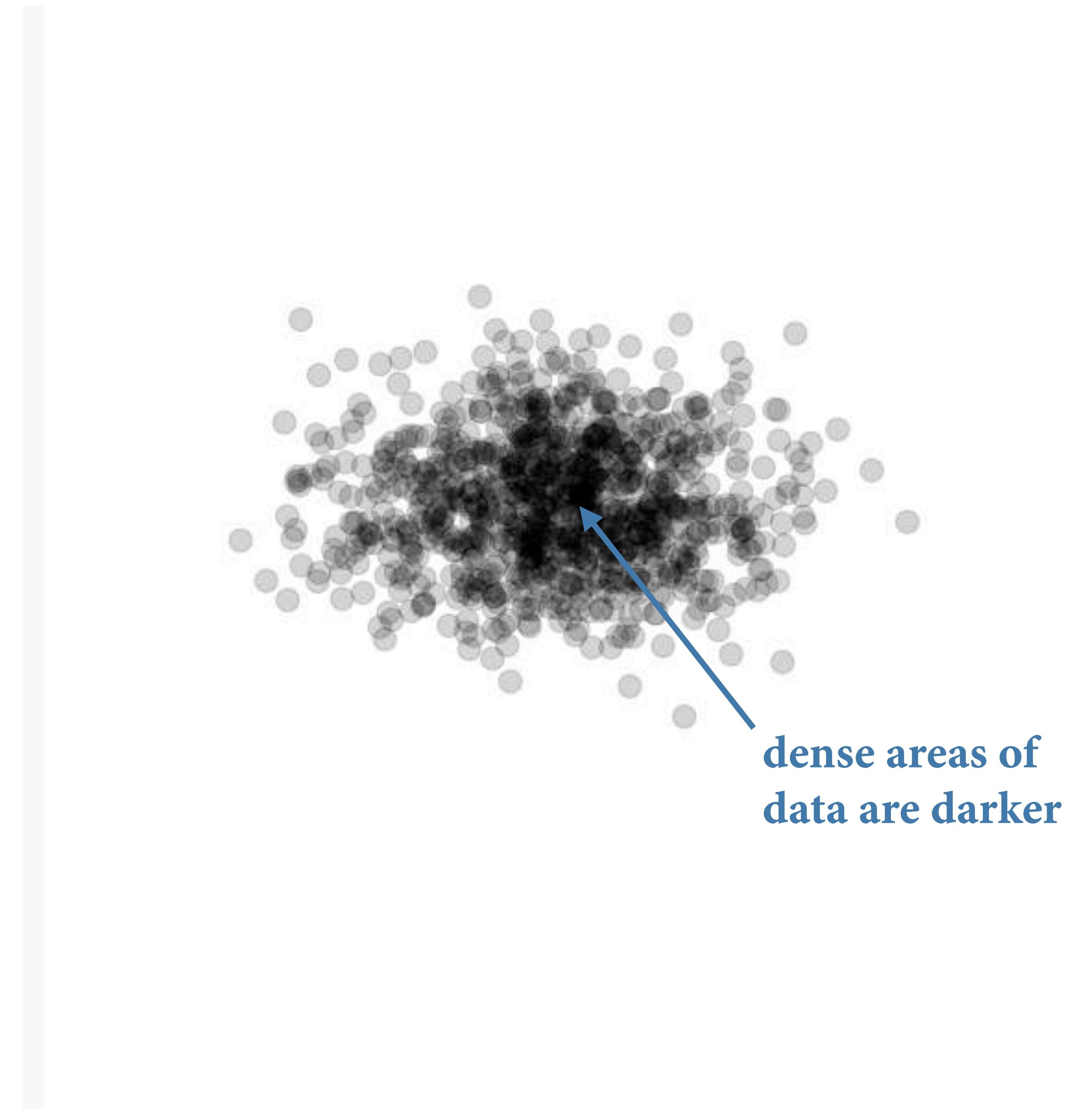
---

Samara, Timothy

graphics as layers | *for layered data encoded in monochrome, transparency reveals density.*

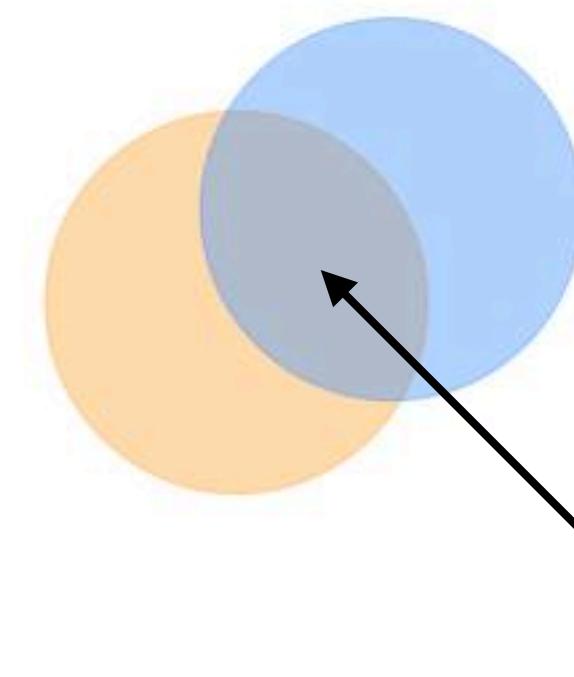
```
x <- rnorm(1000)
y <- rnorm(1000)

ggplot() +
  theme_void() +
  scale_x_continuous(limits = c(-5, 5)) +
  scale_y_continuous(limits = c(-5, 5)) +
  geom_point(aes(x = x, y = y),
             size = 4, color = "black",
             alpha = 0.2)
```



# graphics as layers | *layered data encoded in color, if overlapping, is affected by transparency!*

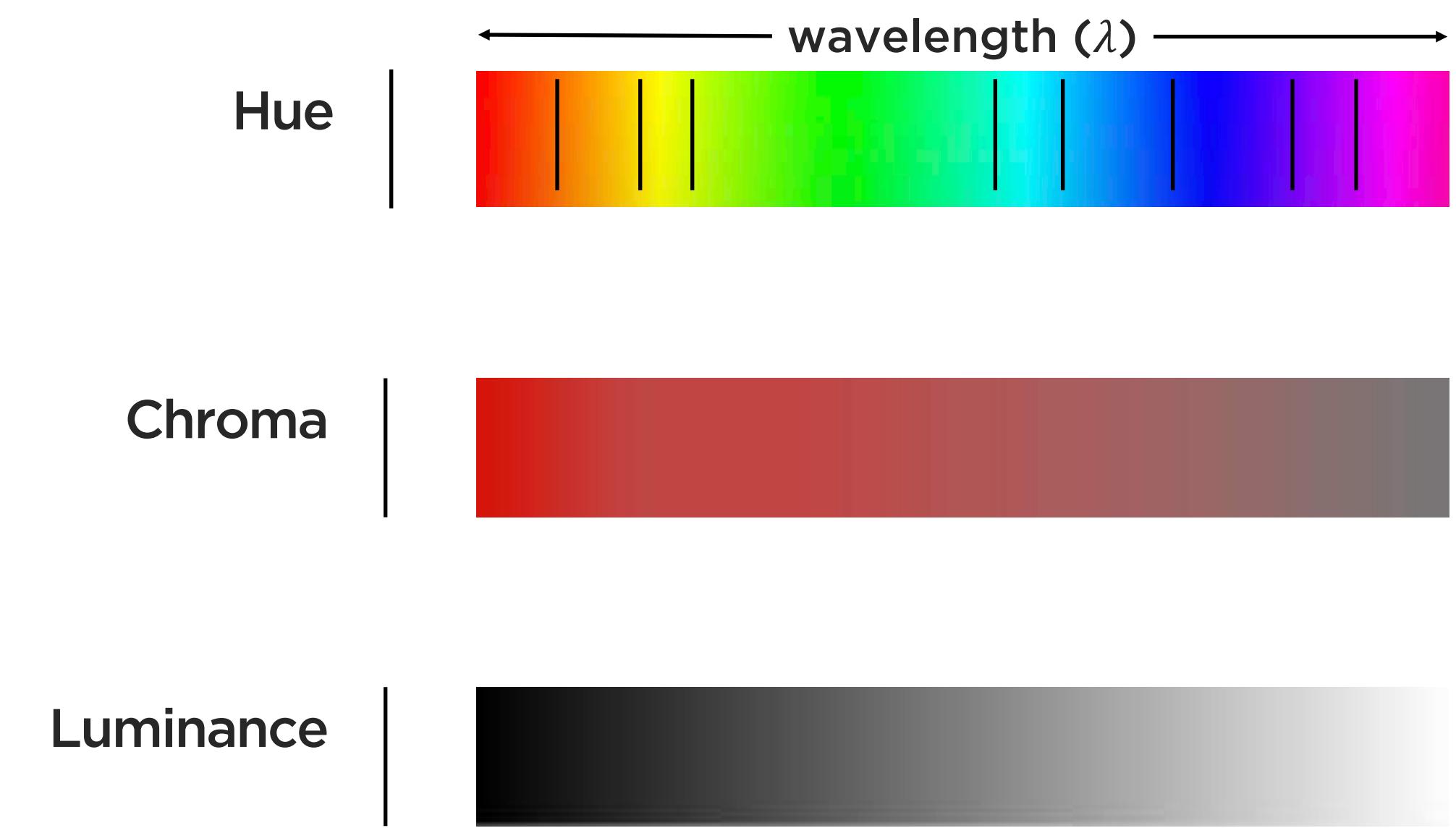
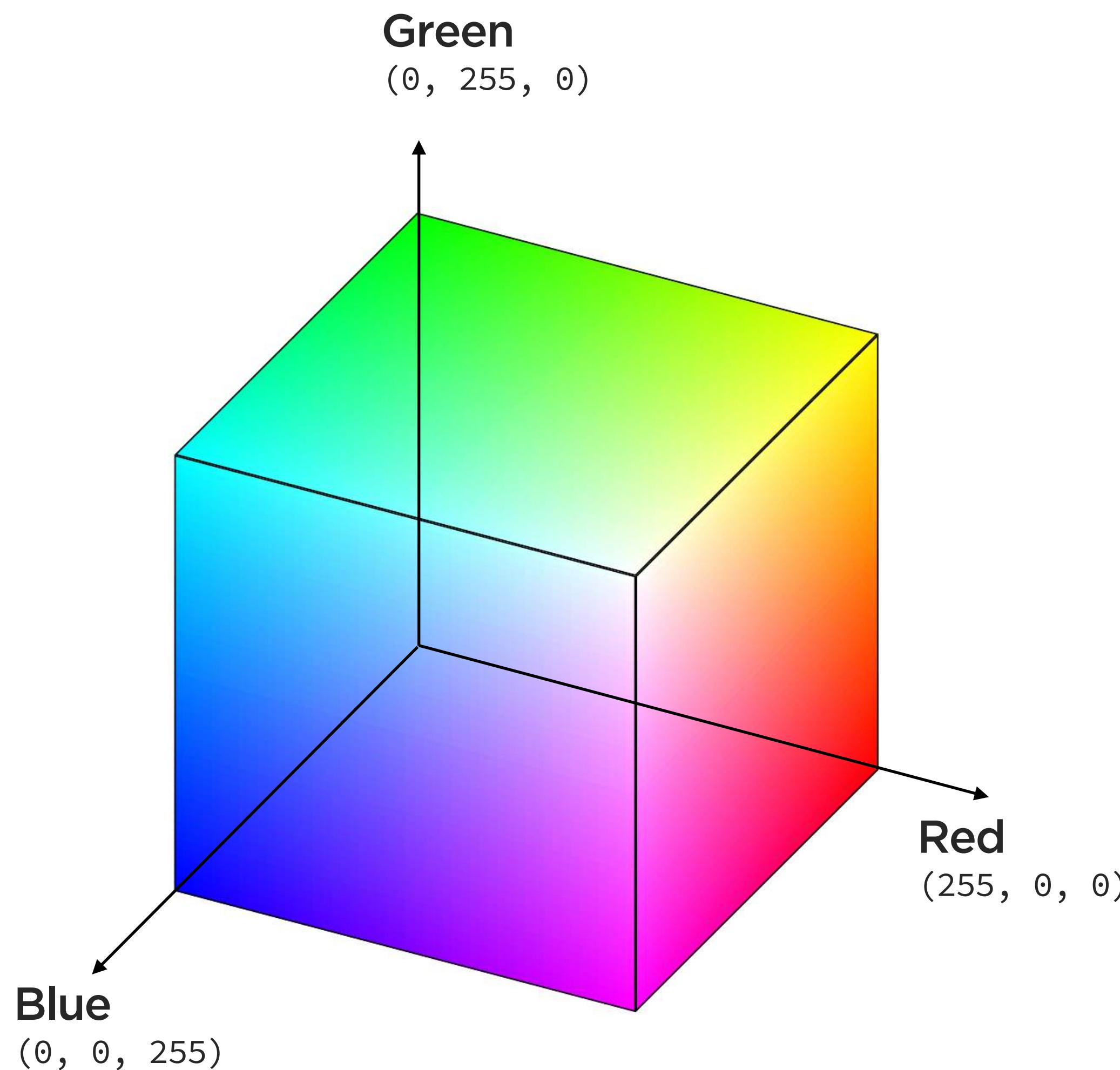
```
ggplot() +  
  theme_void() +  
  scale_x_continuous(limits = c(-5, 5)) +  
  scale_y_continuous(limits = c(-5, 5)) +  
  geom_point(aes(x = 0, y = 0),  
             size = 50, color = "orange",  
             alpha = 0.4) +  
  geom_point(aes(x = 1, y = 1),  
             size = 50, color = "dodgerblue",  
             alpha = 0.4)
```



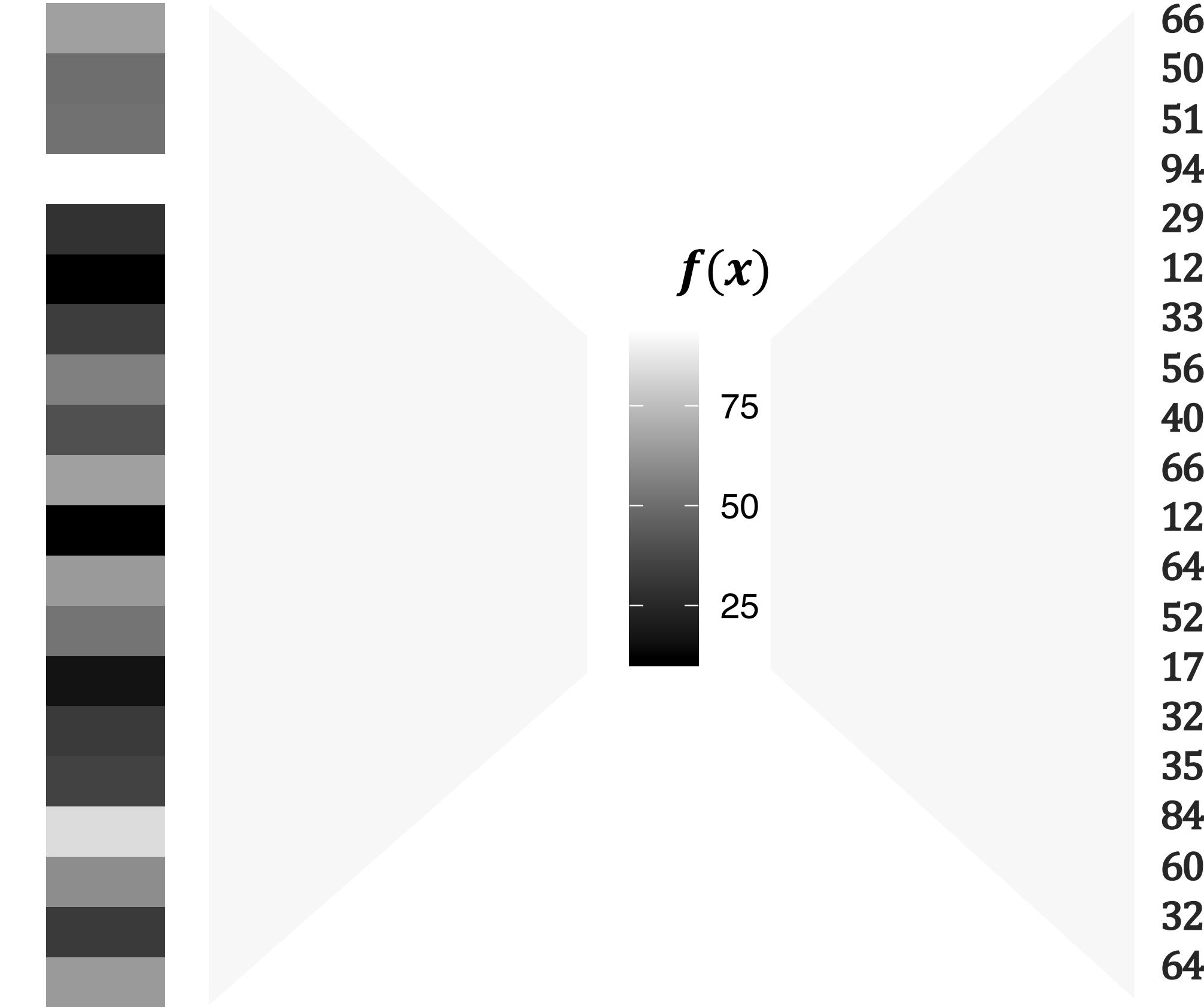
**I didn't encode  
data with *this* color!?**

# **encoding data as color**

color | encode data using color spaces, which are mathematical models



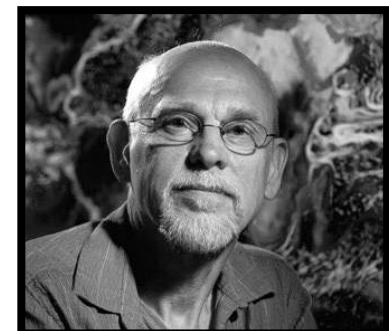
# color | how can we map data to light, whether using its hue, chroma, or luminance?



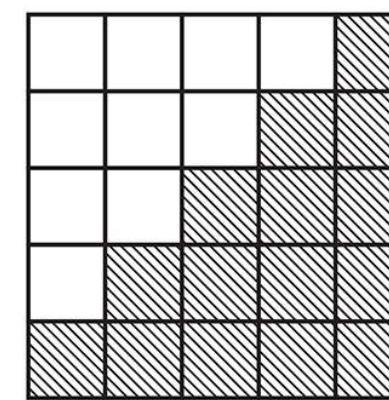
# color | *perceived brightness is nonlinear function of luminance*

**LUMINANCE** : the *measured* amount of light coming from some region of space.

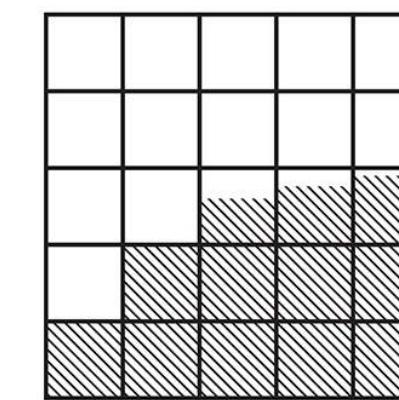
**BRIGHTNESS** : the *perceived* amount of light coming from that region of space.



# color | visual perception of an arithmetical progression depends on a physical geometric progression

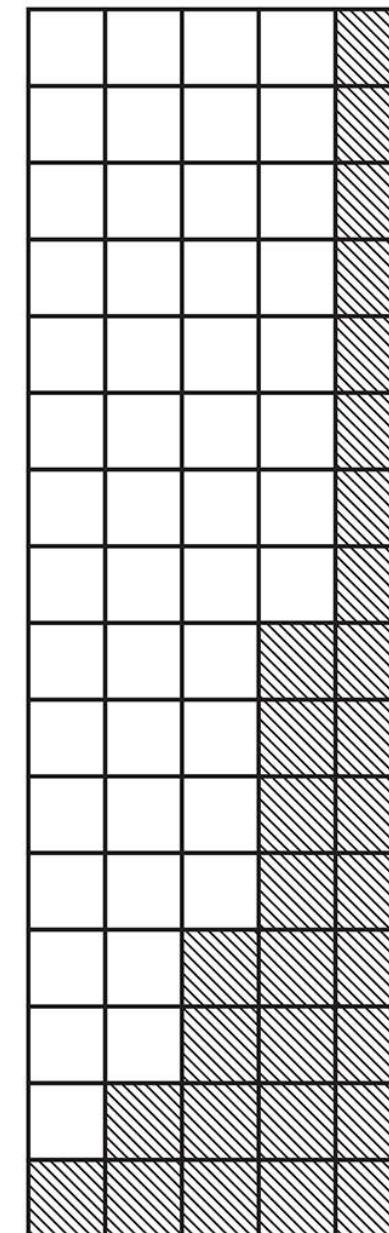


THIS PHYSICAL FACT



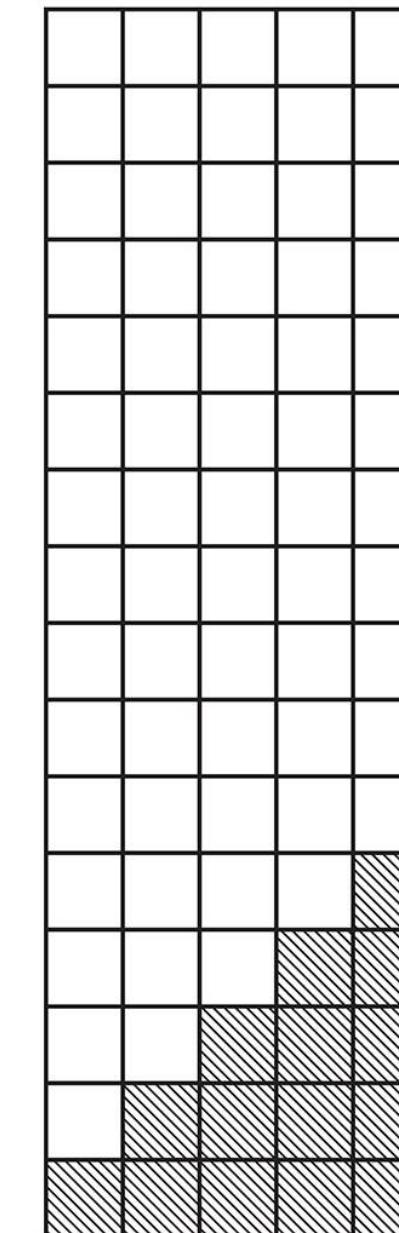
REDUCES TO

THIS PSYCHOLOGICAL EFFECT

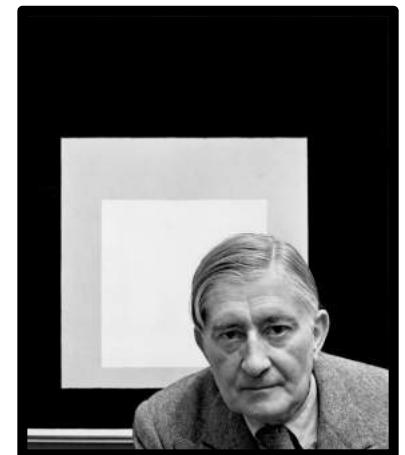


THIS PHYSICAL FACT

PRODUCES

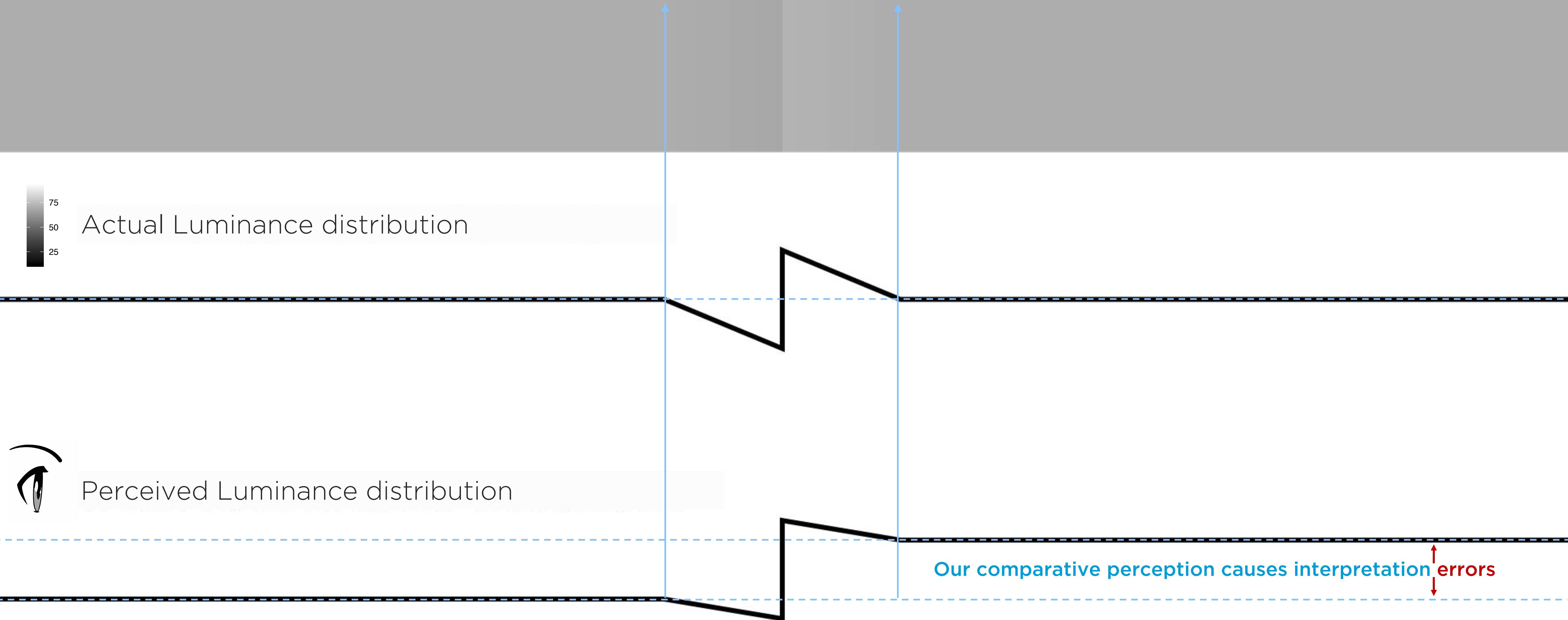


THIS PSYCHOLOGICAL EFFECT

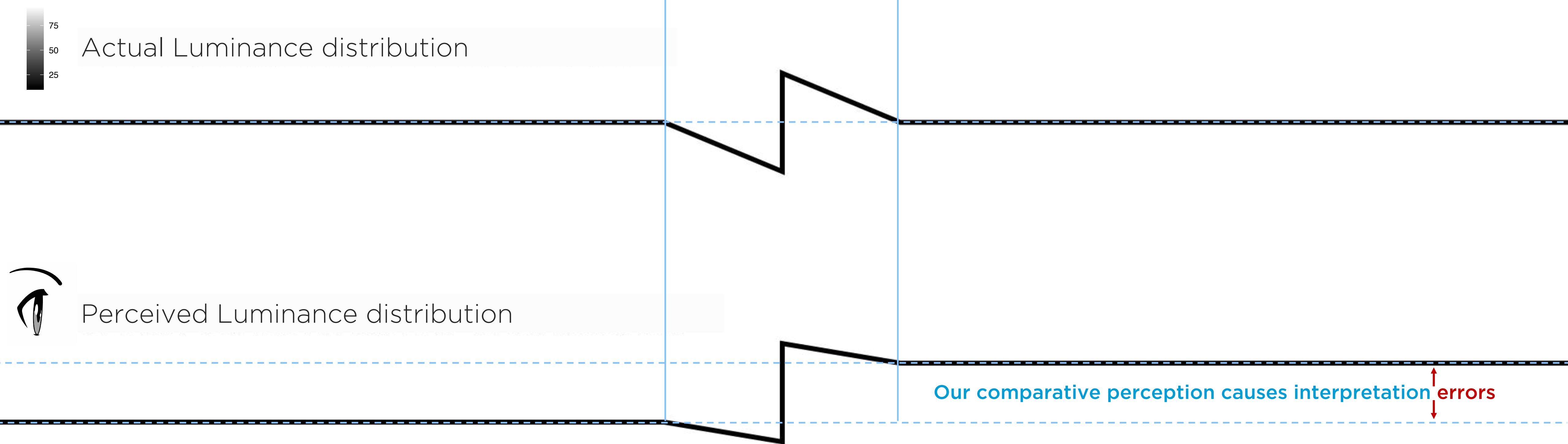


Albers, Josef

color | humans have evolved to see edge contrasts. We see comparative — not absolute — luminance value.



color | humans have evolved to see edge contrasts. We see comparative — not absolute — luminance value.



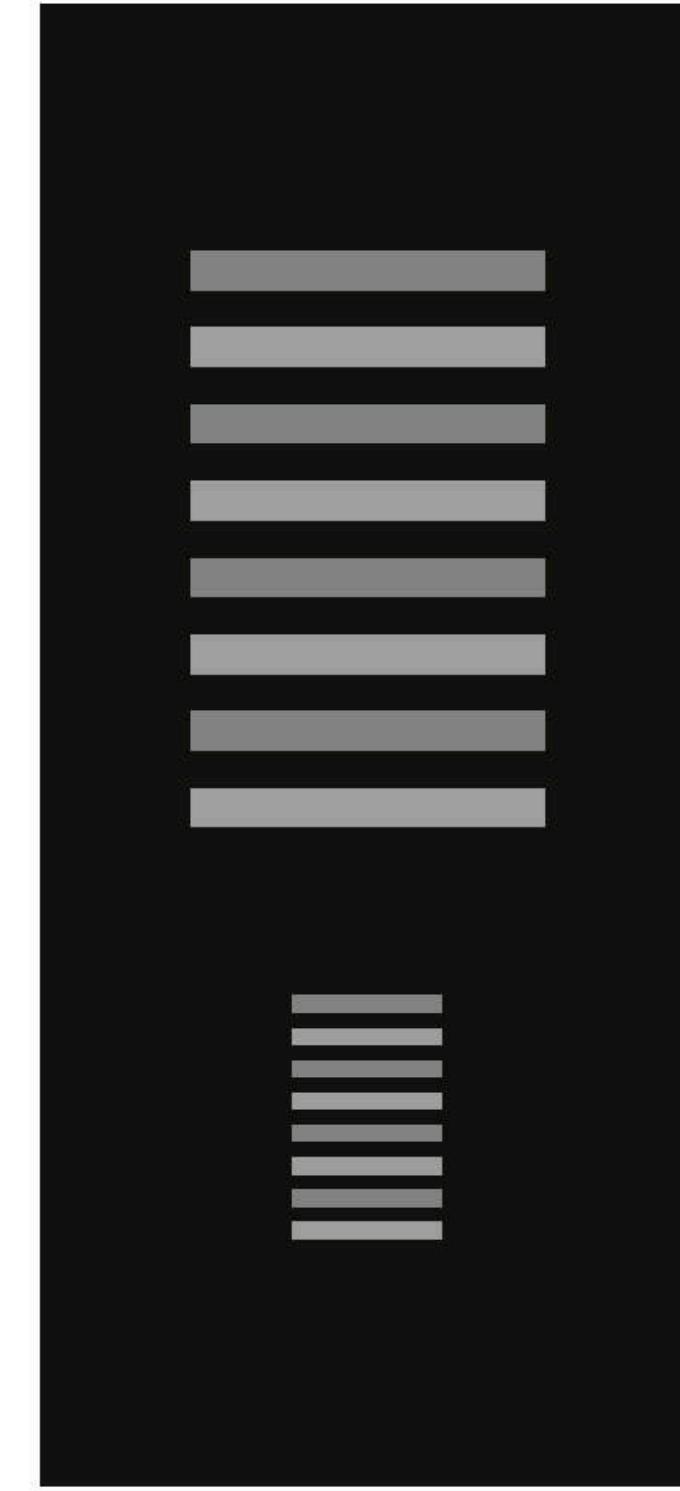
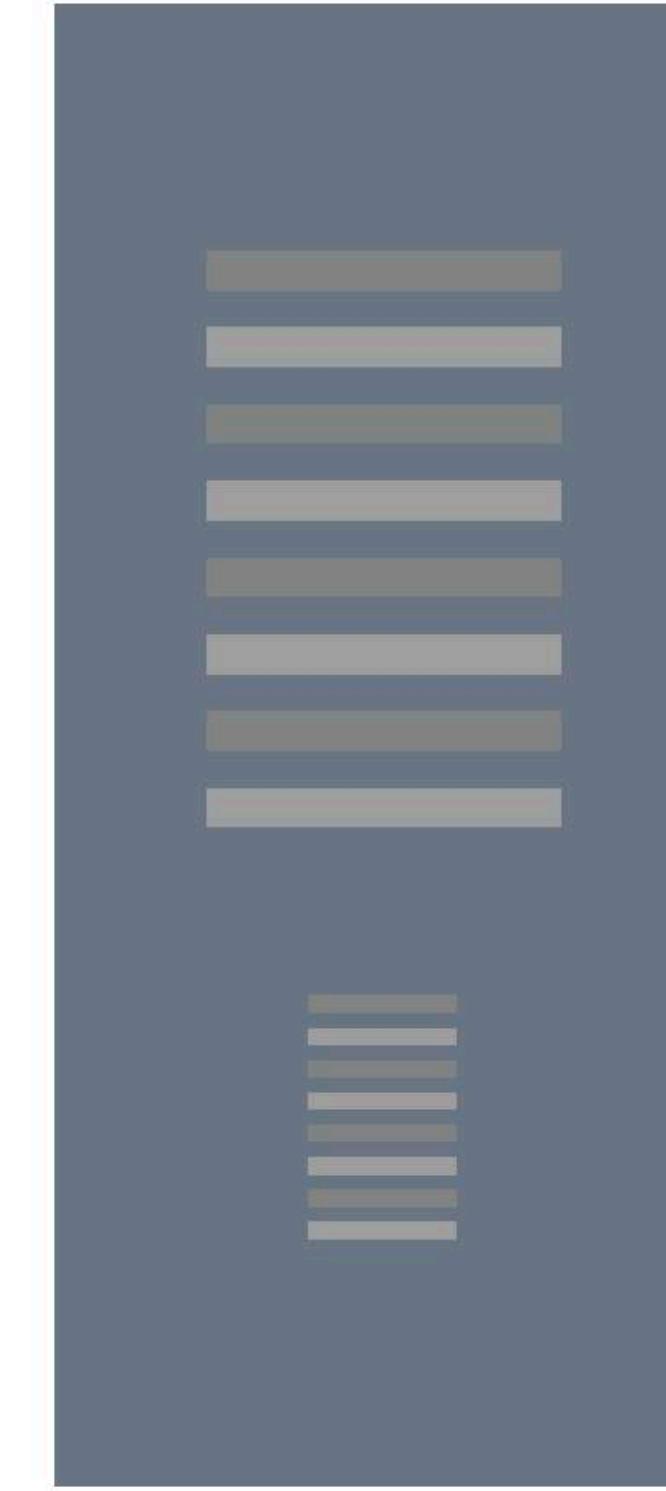
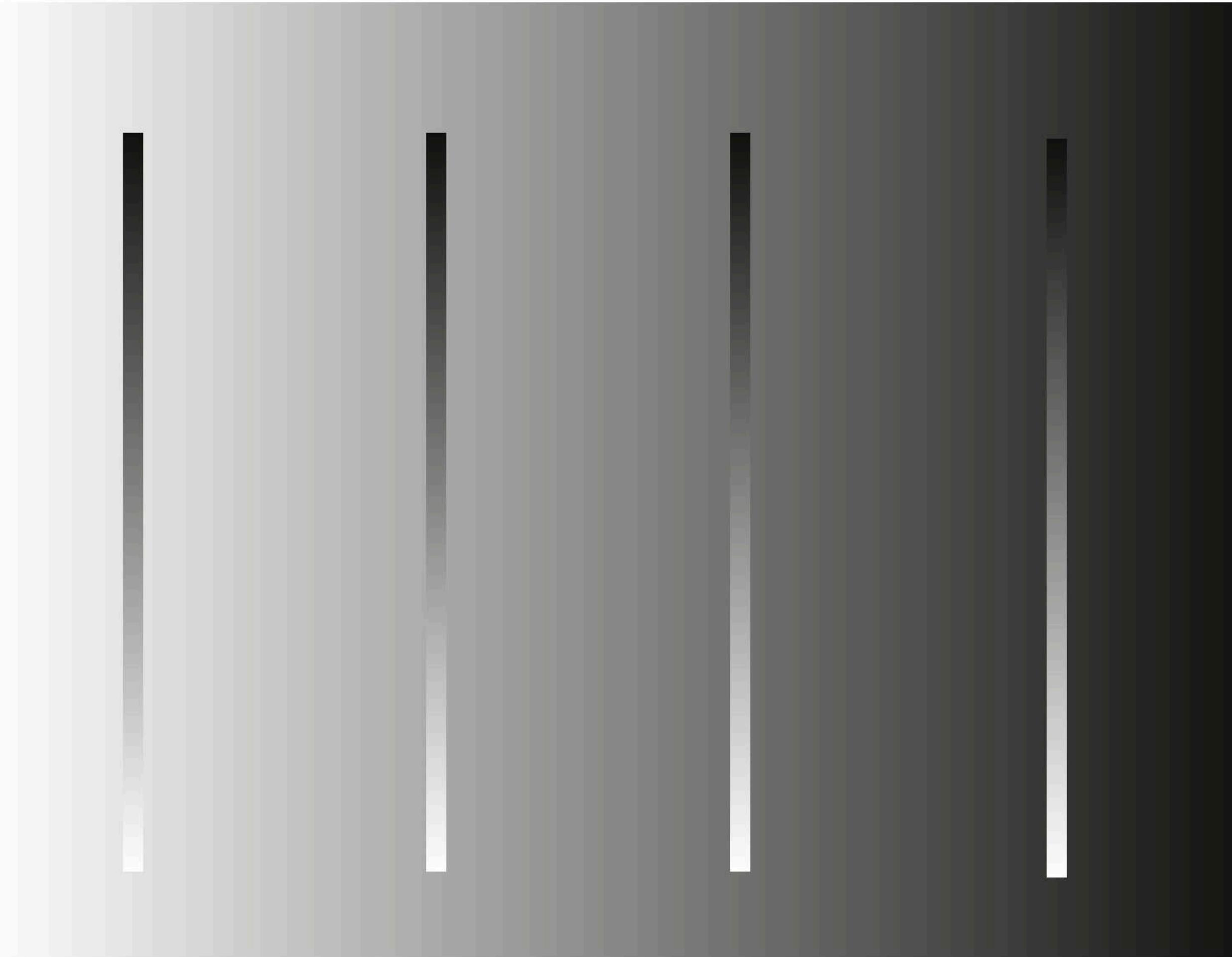
color | *background and adjacent luminance can interfere with our perception*



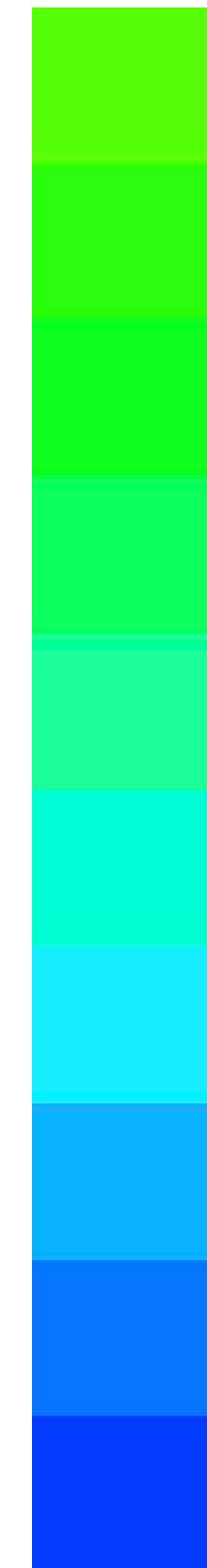
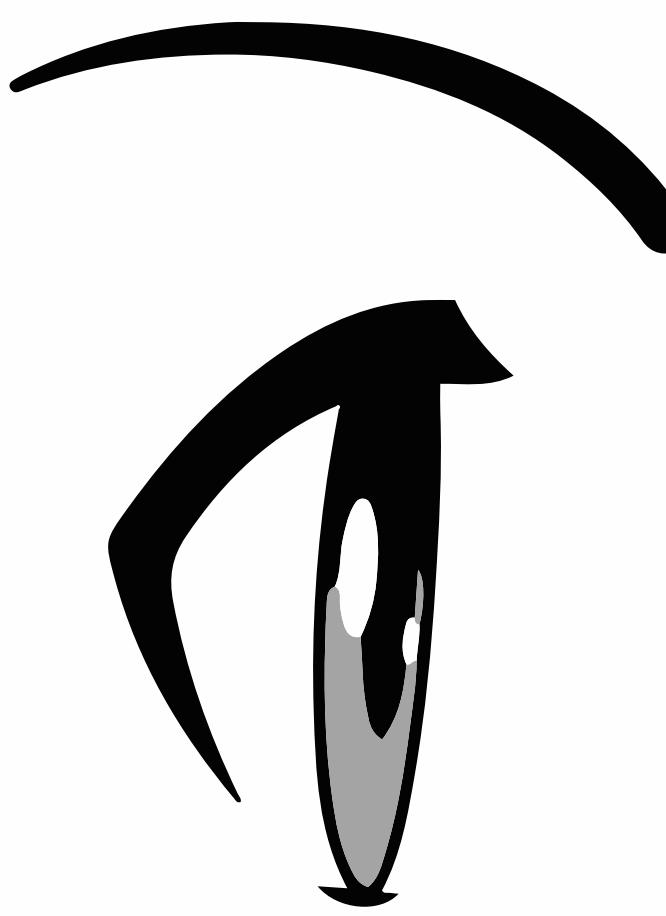
color | background and adjacent luminance can interfere with our perception



*color | high foreground to background luminance contrast enhances shape, lower contrast enhances grayscale*



**color | as with luminance, hue values in the RGB color space fail to uniformly scale across values.**



color | *HSL colorspace is intuitive, but not perceptually uniform in each attribute.*

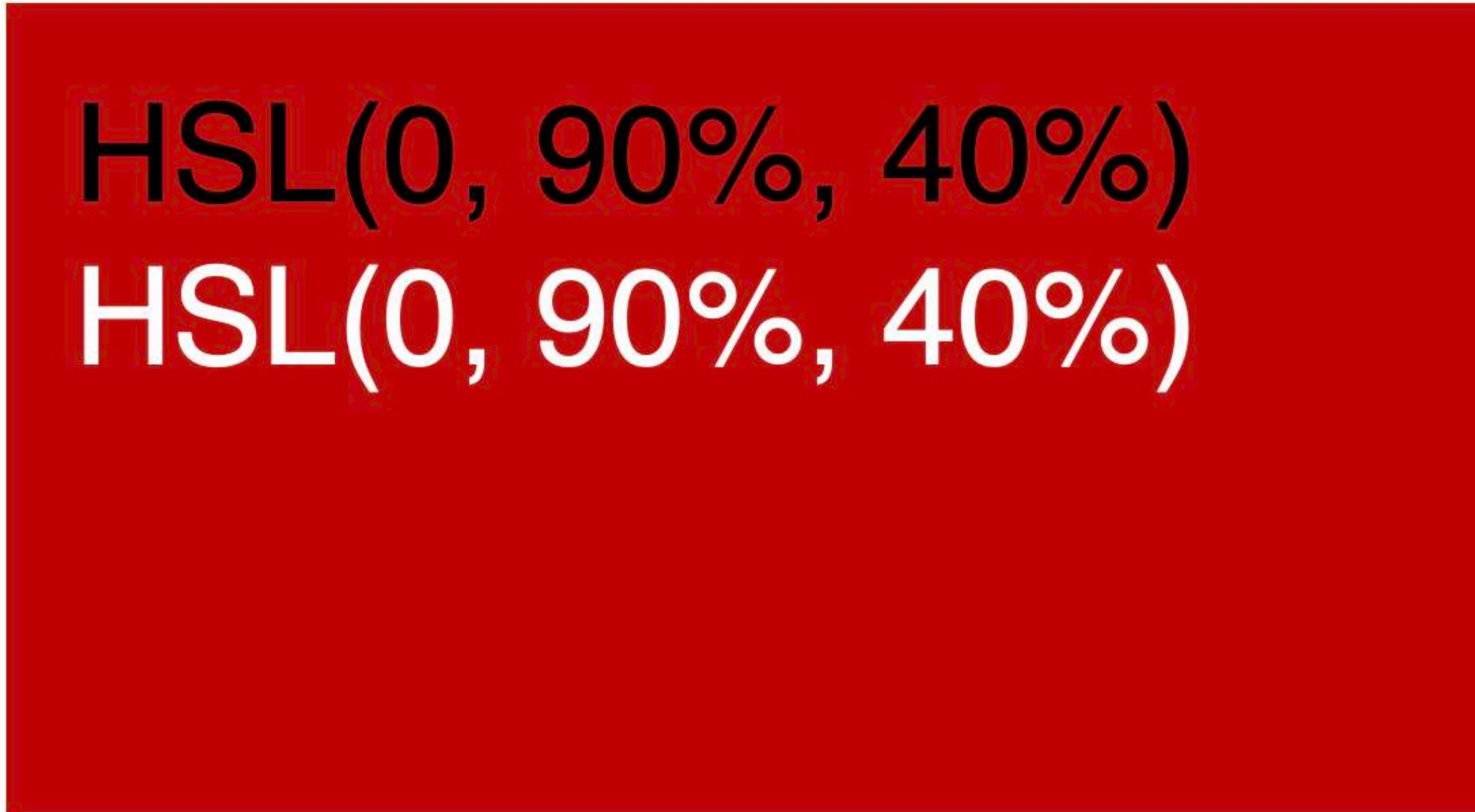
Same luminance or lightness?

**HSL(250, 100%, 50%)**  
**HSL(250, 100%, 50%)**

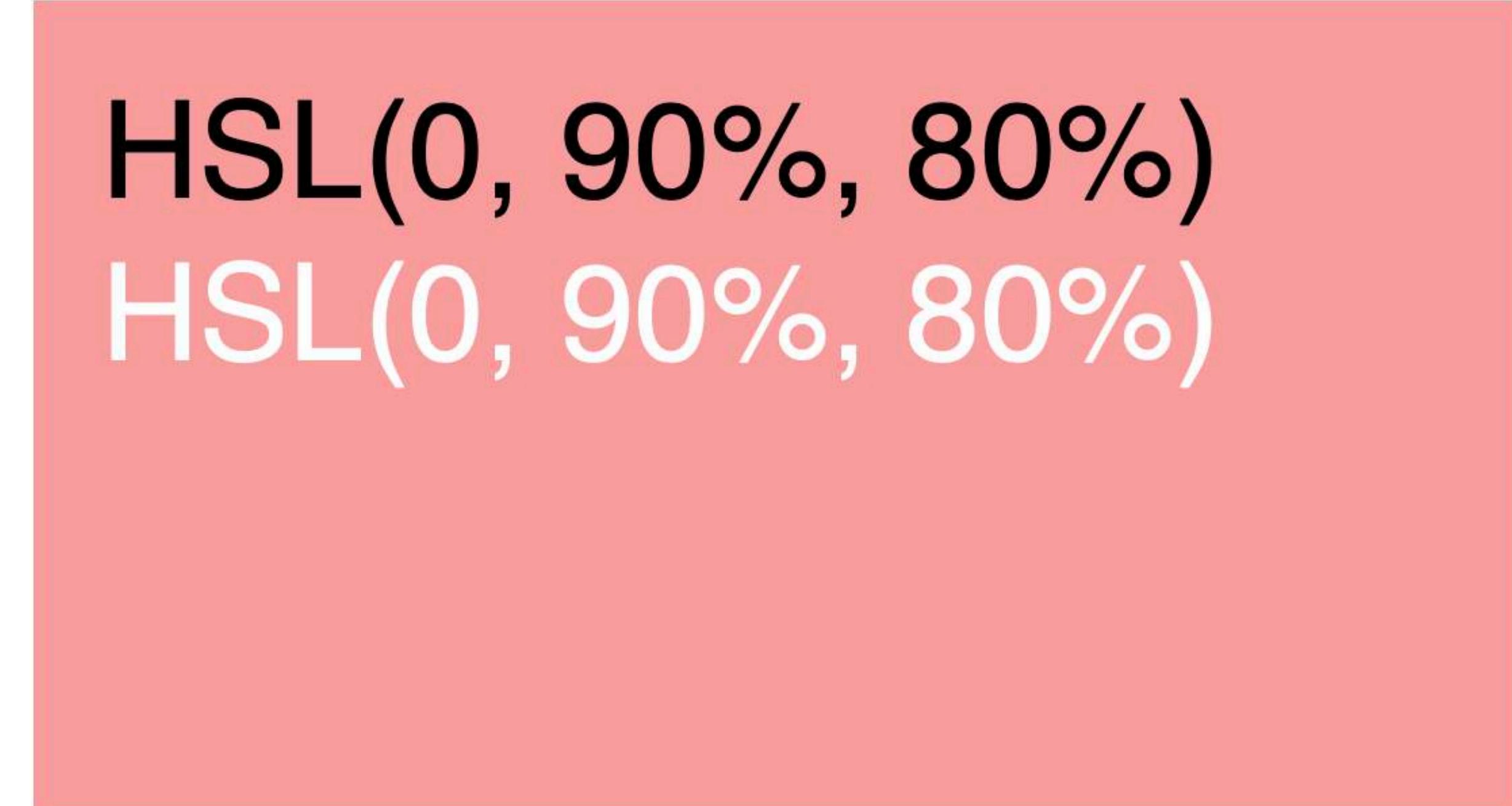
**HSL(60, 100%, 50%)**  
**HSL(60, 100%, 50%)**

color | *HSL colorspace is intuitive, but not perceptually uniform in each attribute.*

Same saturation?



**HSL(0, 90%, 40%)**  
**HSL(0, 90%, 40%)**



**HSL(0, 90%, 80%)**  
**HSL(0, 90%, 80%)**

color | *HSL colorspace is intuitive, but not perceptually uniform in each attribute.*

## Equal difference between hues?

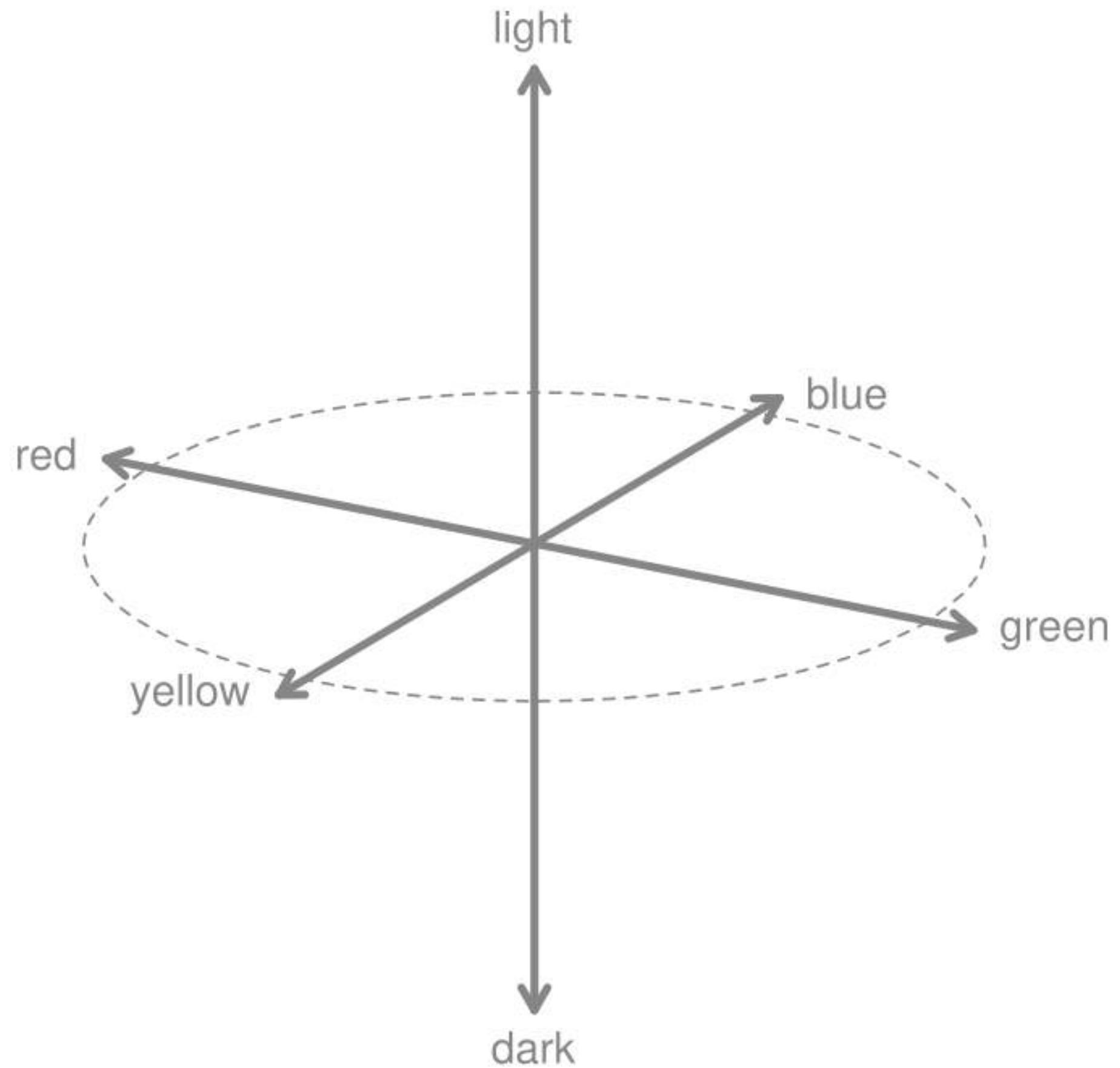
HSL(30, 100%, 50%)  
HSL(30, 100%, 50%)

HSL(50, 100%, 50%)  
HSL(50, 100%, 50%)

HSL(230, 100%, 50%)  
HSL(230, 100%, 50%)

HSL(250, 100%, 50%)  
HSL(250, 100%, 50%)

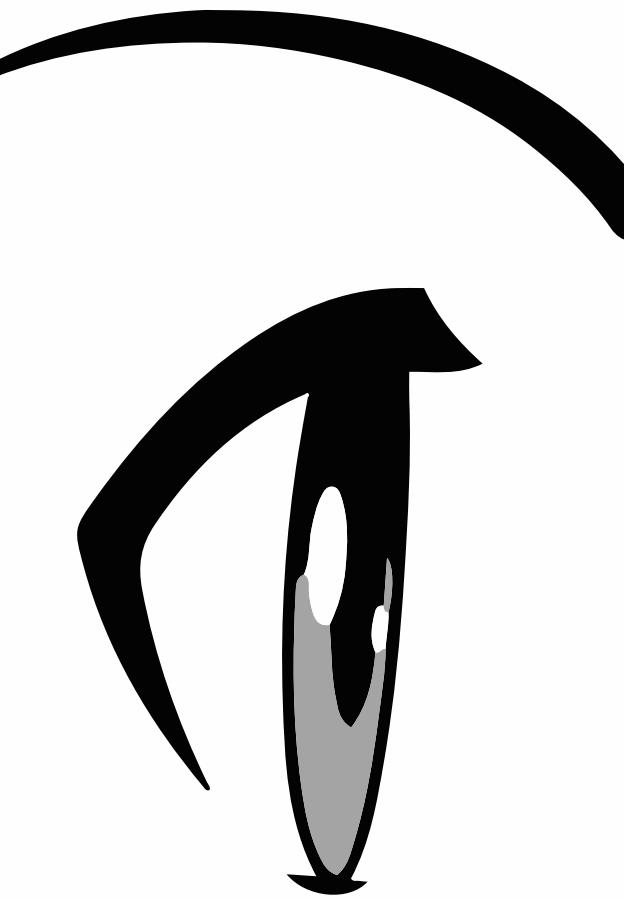
## color | other color spaces show changes in color we perceive as uniform.



The International Commission on Illumination (CIE) studied human perception and re-mapped color into a space where we perceive color changes uniformly.

Their **CIELuv** color model has two dimensions — u and v — that represent color scales from red to green and yellow to blue.

color | *perceptually uniform color spaces better represent quantity.*



# color | example encoding data as perceptually uniform color attributes in R/ggplot2

Load functions for mapping color  
<https://www.hsluv.org/implementations/>

```
url_hsluv_py <-  
  paste0('https://raw.githubusercontent.com/','  
    'hsluv/hsluv-python/master/hsluv.py')
```

```
reticulate::source_python(url_hsluv_py)
```

Create sample data encoded as hue, saturation, luminance

```
df <- expand.grid(H = c(20, 290),  
                  S = seq(0, 100, by = 10),  
                  L = seq(0, 100, by = 10))
```

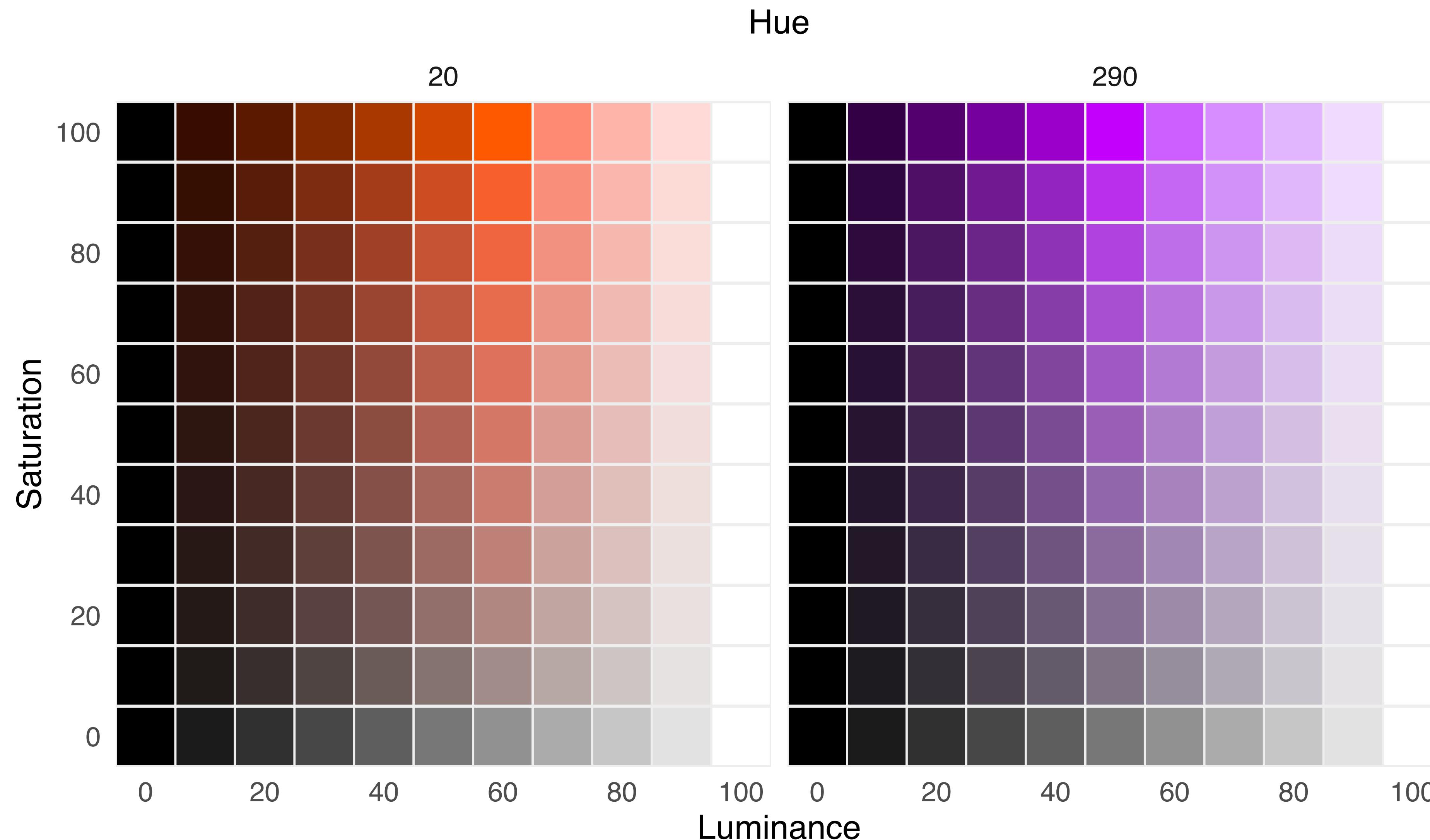
Convert HSLuv scaled values to RGB  
color space as hex code #RRGGBB

```
for(i in seq(NROW(df)))  
  df$colors <- with(df[i,], hsluv_to_hex(c(H, S, L)) )
```

Plot data encoded as colors

```
library(ggplot2)  
  
ggplot(df) +  
  theme_minimal() +  
  theme(panel.grid = element_blank(),  
        axis.text.x.top = element_blank()) +  
  geom_point(aes(L, S),  
             color = '#eeeeee',  
             fill = df$colors,  
             size = 10,  
             shape = 22) +  
  scale_x_continuous(breaks = seq(0, 100, by = 20),  
                     sec.axis = sec_axis(~., name = 'Hue')) +  
  scale_y_continuous(breaks = seq(0, 100, by = 20)) +  
  facet_wrap(~H) +  
  labs(x = 'Luminance',  
       y = 'Saturation')
```

# color | example encoding data as perceptually uniform color attributes in R/ggplot2

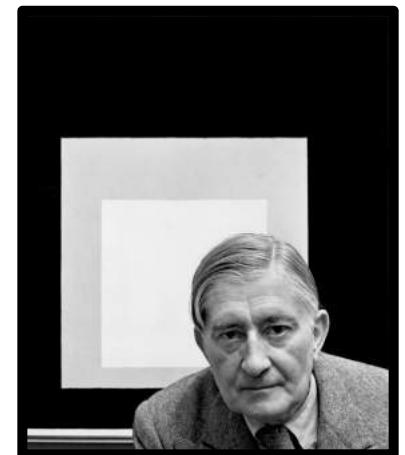
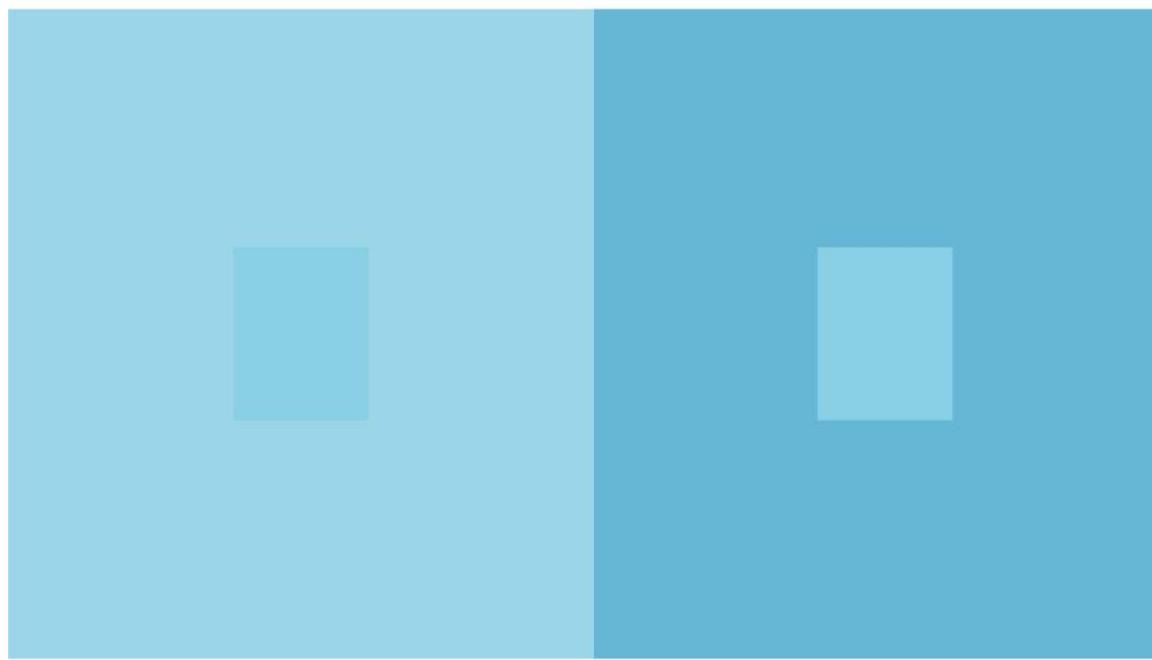
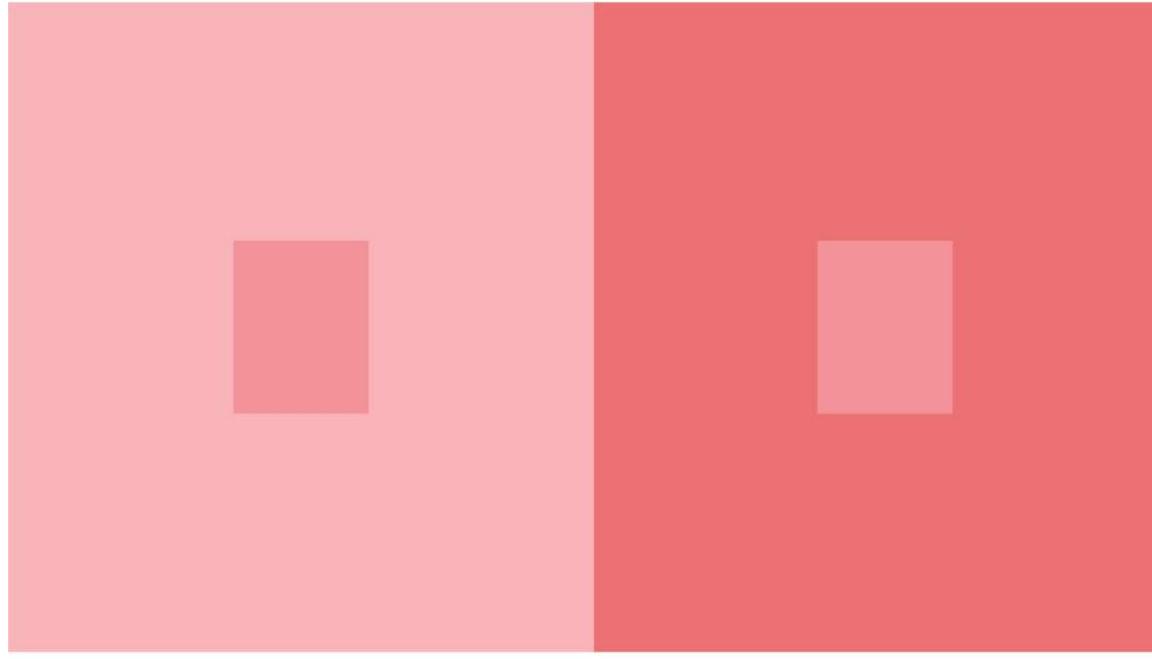


color | perceptually uniform color spaces also help in distinguishing categorical data.



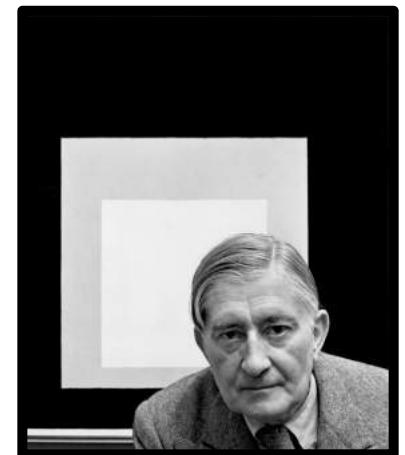
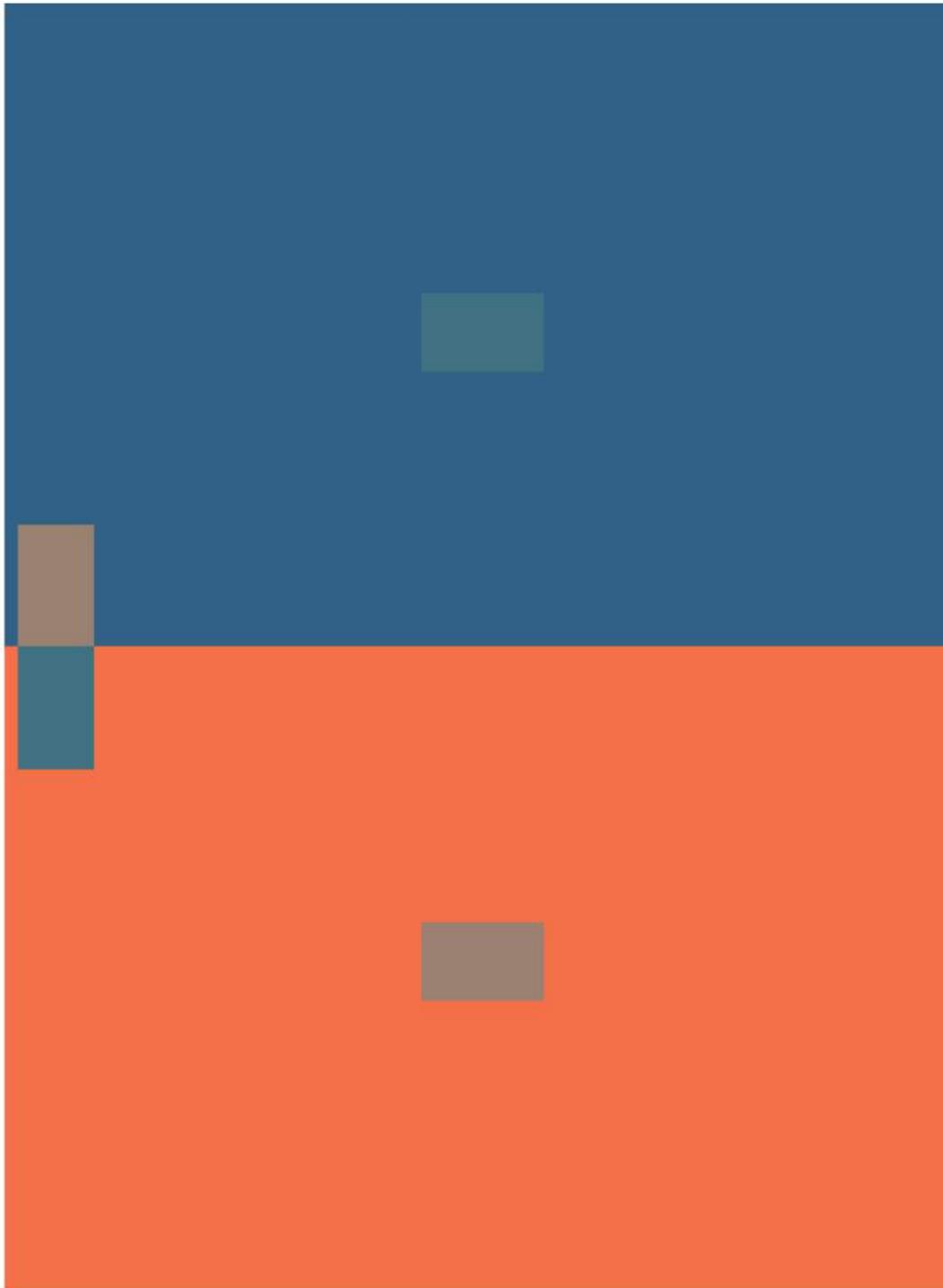
# **interaction of color**

# interaction of color | *one color appearing as two*



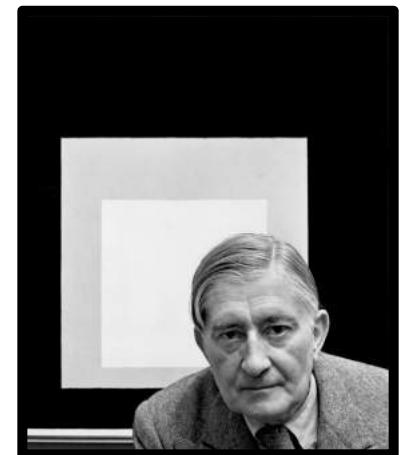
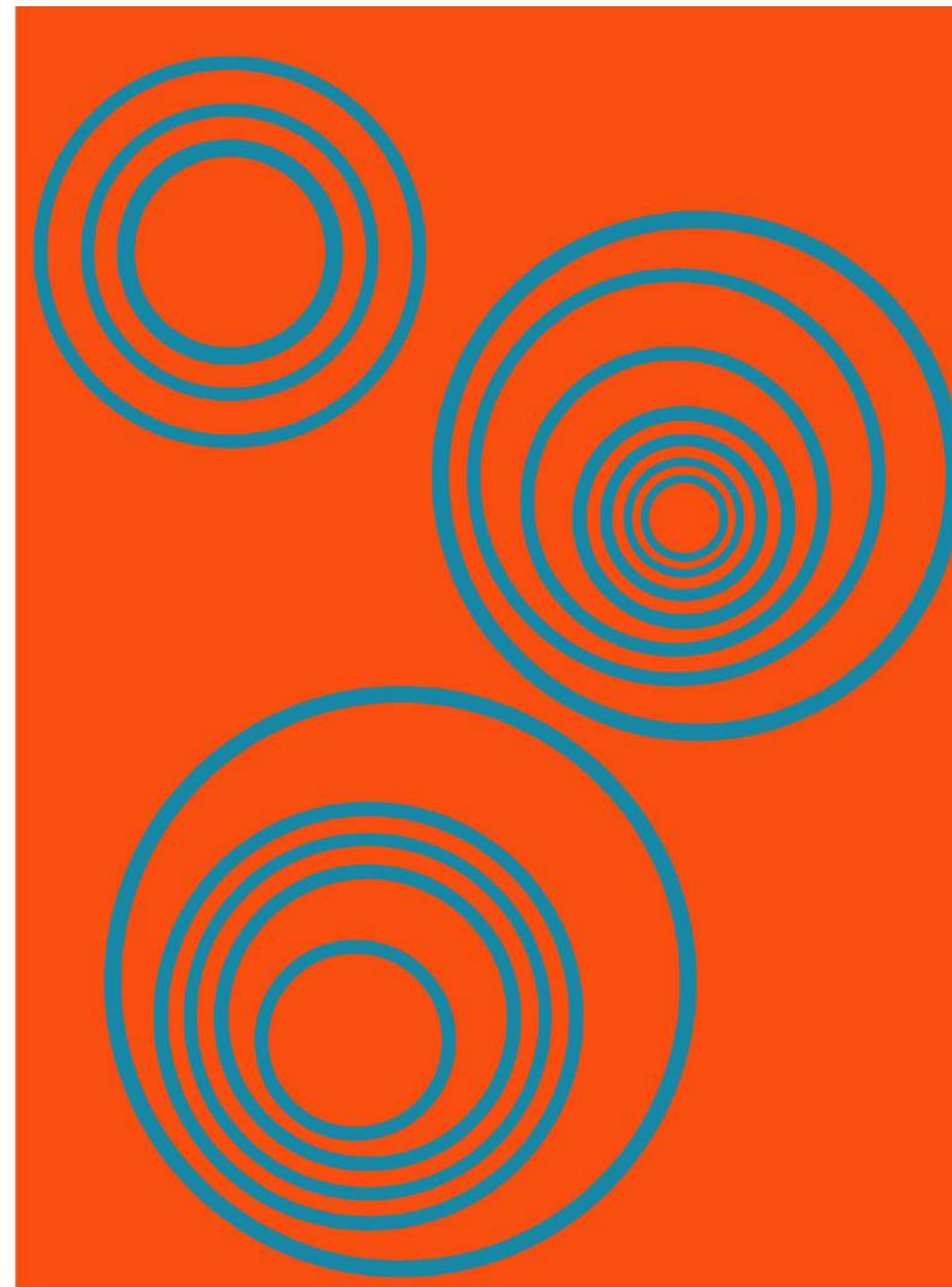
Albers, Josef

interaction of color | *two different colors look alike*



Albers, Josef

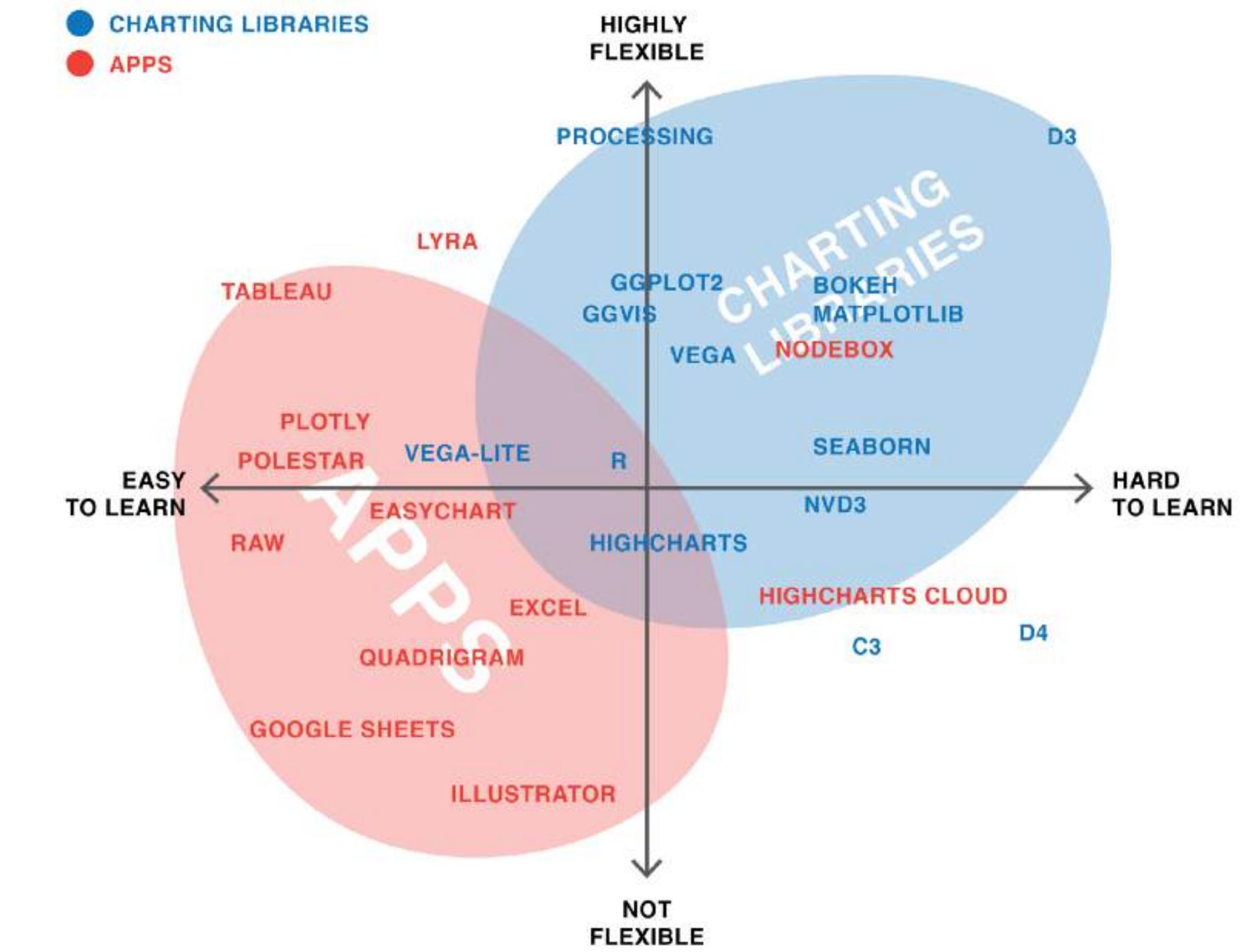
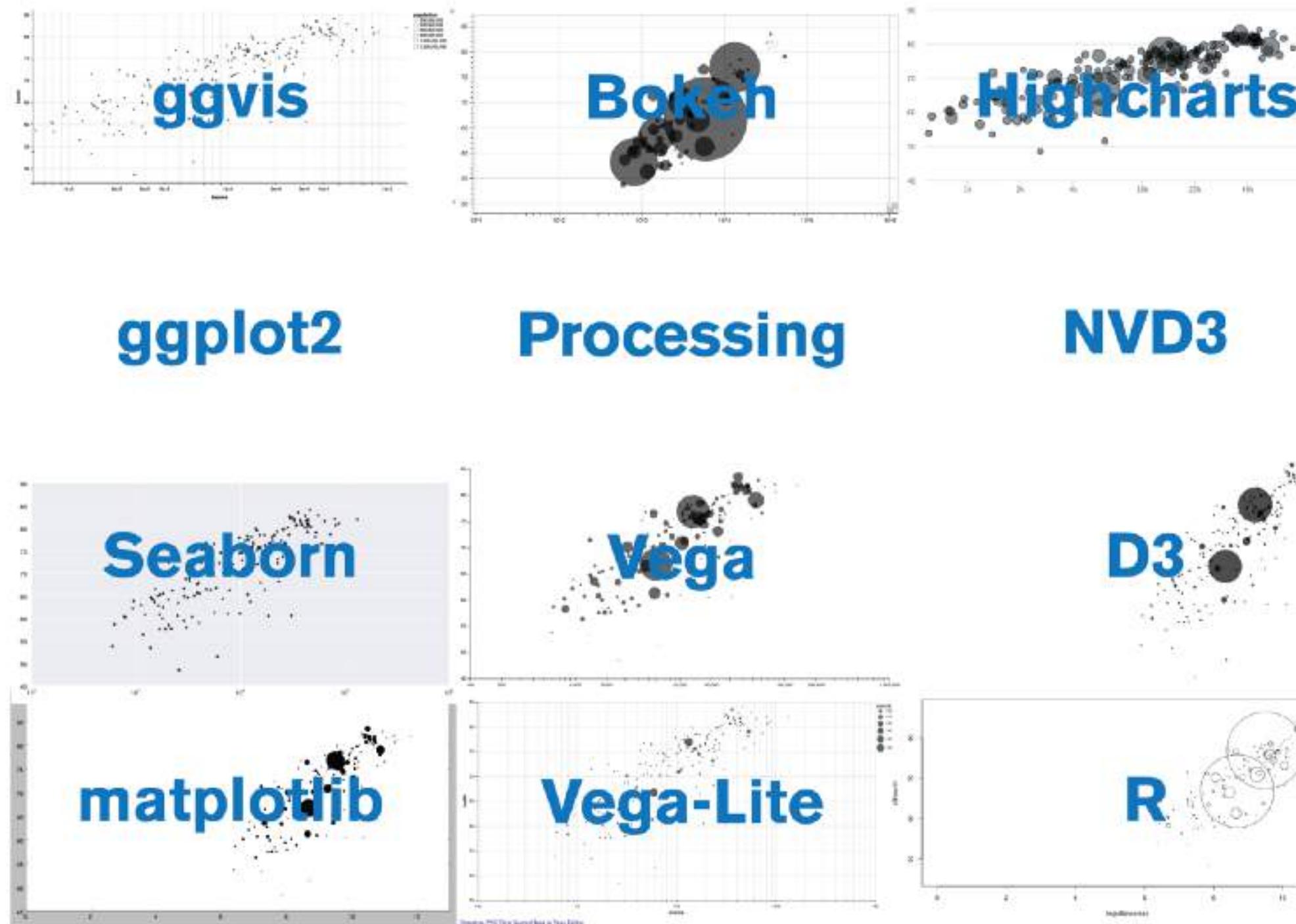
interaction of color | *vibrating boundaries, occurs with contrasting hues of similar luminance*



Albers, Josef

# **perspectives on tools**

thoughts about tools | *a designer's experiments. One chart created with numerous tools (though more exist)*



“There are no perfect tools, just good tools for people with certain goals.”



Rost, Lisa Charlotte

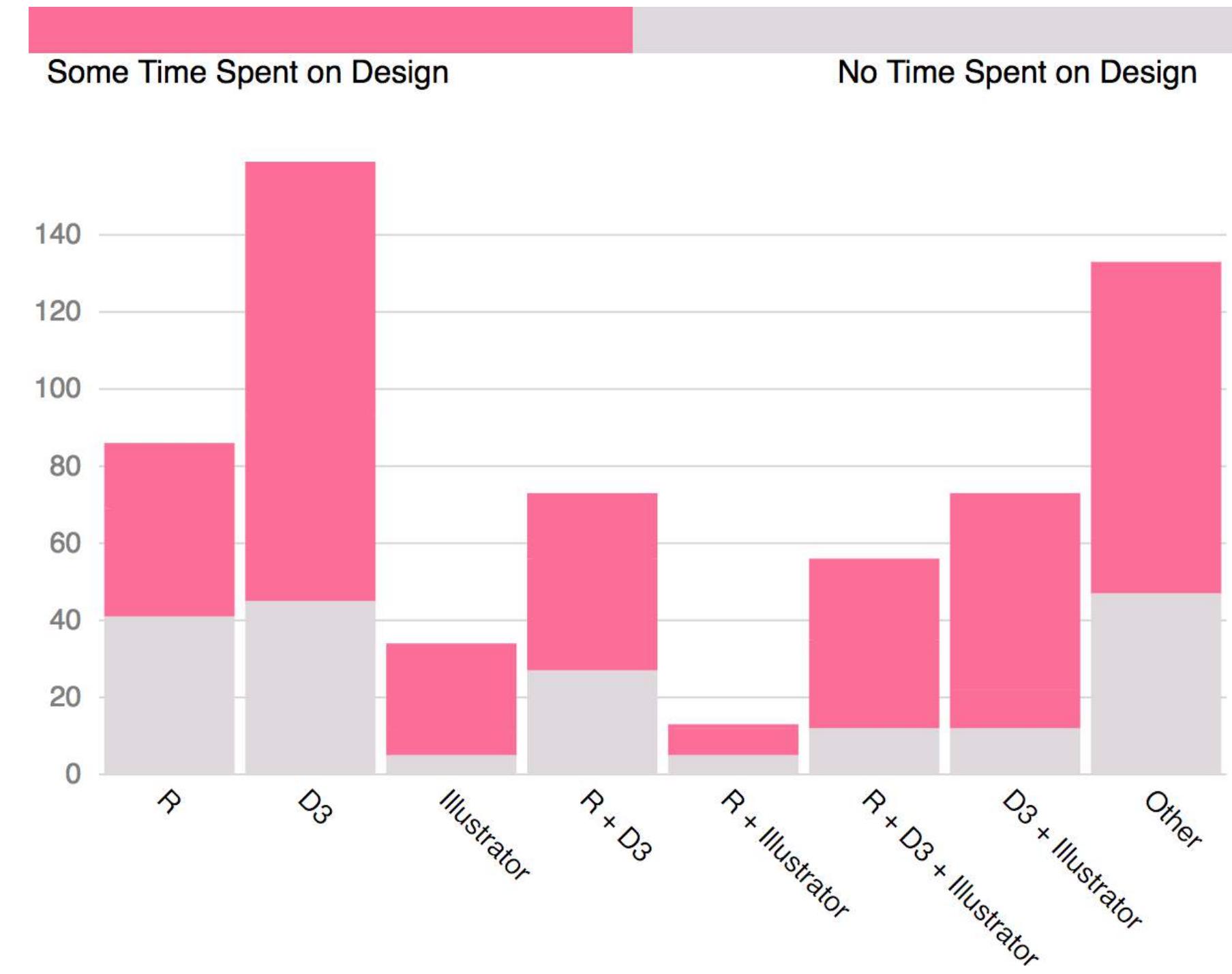
thoughts about tools | *whatever your tools, invest time learning—and applying—design. Do it for your audience.*

 **Edward Tufte**  
@EdwardTufte

#Rstats coders and users just can't do words on graphics and typography. Proof: 40 years of clunky, even recent Stanford Statistics textbooks. Publication-quality work requires: R + Adobe Illustrator + reasoning about words on graphics + respect for audience/readers/viewers

1 4:04 PM - Jun 26, 2018

[See ./philip.sh's other Tweets](#)



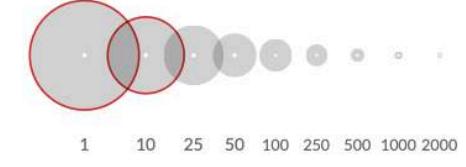
## **practice in the studio**

# practice in the studio

## Let The Music Play

All songs from the **TOP 2000** of 2017 according to their release years

Position in the Top 2000



1 10 25 50 100 250 500 1000 2000

2000

1950

1900

1850

1800

1750

1700

1650

1600

1550

1500

1450

1400

1350

1300

1250

1200

1150

1100

1050

1000

950

900

850

800

750

700

650

600

550

500

450

400

350

300

250

200

150

100

50

10

1

2015

2010

2005

2000

1995

1990

1985

1980

1975

1970

1965

1960

1955

1950

1945

1940

1935

1930

1925

1920

1915

1910

1905

1900

1895

1890

1885

1880

1875

1870

1865

1860

1855

1850

1845

1840

1835

1830

1825

1820

1815

1810

1805

1800

1795

1790

1785

1780

1775

1770

1765

1760

1755

1750

1745

1740

1735

1730

1725

1720

1715

1710

1705

1700

1695

1690

1685

1680

1675

1670

1665

1660

1655

1650

1645

1640

1635

1630

1625

1620

1615

1610

1605

1600

1595

1590

1585

1580

1575

1570

1565

1560

1555

1550

1545

1540

1535

1530

1525

1520

1515

1510

1505

1500

1495

1490

1485

1480

1475

1470

1465

1460

1455

1450

1445

1440

1435

1430

1425

1420

1415

1410

1405

1400

1395

1390

1385

1380

1375

1370

1365

1360

1355

1350

1345

1340

1335

1330

1325

1320

1315

1310

1305

1300

1295

1290

1285

1280

1275

1270

1265

1260

1255

1250

1245

1240

1235

1230

1225

# References

**Albers**, Josef. *Interaction of Color*. Yale University Press, 2006, and interactive app on iPad: <http://interactionofcolor.com>

**Bremer**, Nadieh. “*Let the Music Play*.” Github pages. Accessed March 26, 2020. <https://nbremer.github.io/top2000vinyl/>.

**Boronine**, Alexei. “*Color Spaces for Human Beings*.” HSLuv.org (blog), March 26, 2012. <https://www.hsluv.org> and <https://www.boronine.com/2012/03/26/Color-Spaces-for-Human-Beings/>.

**Kay**, Matthew, and Jeffrey Heer. “*Beyond Weber’s Law: A Second Look at Ranking Visualizations of Correlation*.” IEEE Transactions on Visualization and Computer Graphics 22, no. 1 (January 31, 2016): 469–78.

**Lupi**, Giorgia. “*The Architecture of a Data Visualization: Multilayered Storytelling through Info-Spatial Compositions*,” February 2015. <https://medium.com/accurat-studio/the-architecture-of-a-data-visualization-470b807799b4>.

**Meeks**, Elijah. “*You Can Design a Good Chart with R: But Do R Users Invest in Design?*,” July 2, 2018. <https://medium.com/nightingale/you-can-design-a-good-chart-with-r-5d00ed7dd18e>.

———. “*Non-Linear Storytelling: Journalism through Info-Spatial Compositions*.” Parsons Journal for Information Mapping IV, no. 4 (October 2012): 1–11.

**Rost**, Lisa Charlotte. “*One Chart, Nine Tools – Revisited*.” Lisa Charlotte Rost (blog), October 18, 2018. <https://lisacharlotterost.de/datavistools-revisited>.

———. “*What I Learned Recreating One Chart Using 24 Tools*.” Code in Journalism. Source (blog), December 8, 2016. <https://source.opennews.org/articles/what-i-learned-recreating-one-chart-using-24-tools/>.

**Samara**, Timothy. *Design Elements: A Graphic Style Manual. Understanding the Rules and Knowing When to Break Them*. Rockport, 2014.

**Spencer**, Scott. “*Demonstration of Layers in Graphics*.” Columbia University., March 6, 2020.

**Tufte**, Edward R. “*Layers and Separation*.” In *Envisioning Information*. Graphics Press, 1990.

———. “*Color and Information*.” In *Envisioning Information*. Graphics Press, 1990.

**Ware**, Colin. *Information Visualization: Perception for Design*. Fourth. Philadelphia: Elsevier, Inc, 2020.

**Wickham**, Hadley. “*A Layered Grammar of Graphics*.” Journal of Computational and Graphical Statistics 19, no. 1 (January 2010): 3–28.

**Wilkinson**, Leland. *The Grammar of Graphics*. Second. Springer, 2005.