Data Visualization using matplotlib

Bruno Gonçalves

www.bgoncalves.com

https://bmtgoncalves.github.io/DataVisualization/

Data V using matplotlib

Bruno G

www.bgo

https://bm .io/DataVisualization/

Data Visualization using matplotlib

Bruno Gonçalves

www.bgoncalves.com

https://bmtgoncalves.github.io/DataVisualization/

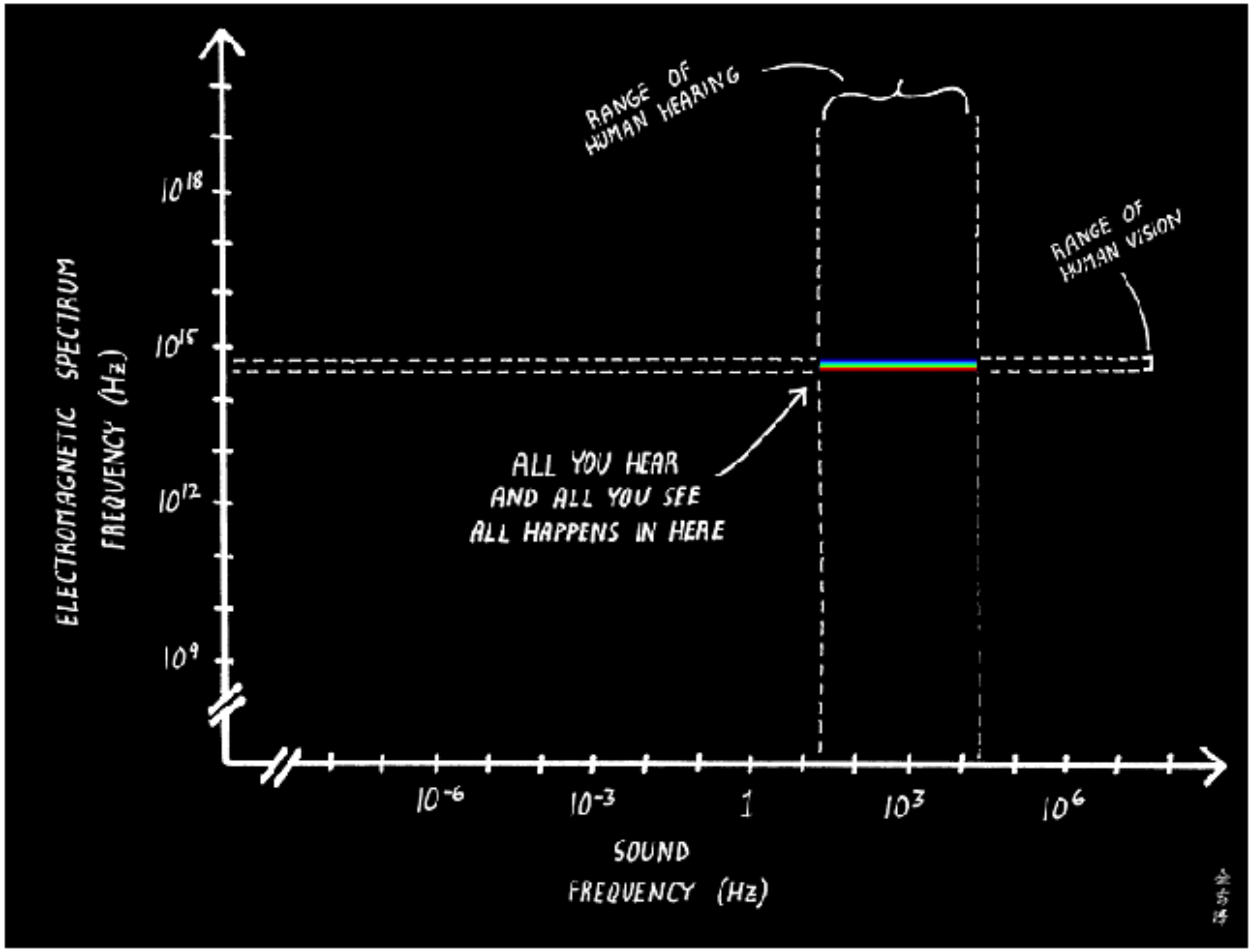
Disclaimer

The views and opinions expressed in this article are those of the authors and do not necessarily reflect the official policy or position of my employer. The examples provided with this tutorial were chosen for their didactic value and are not mean to be representative of my day to day work.

Human Perception

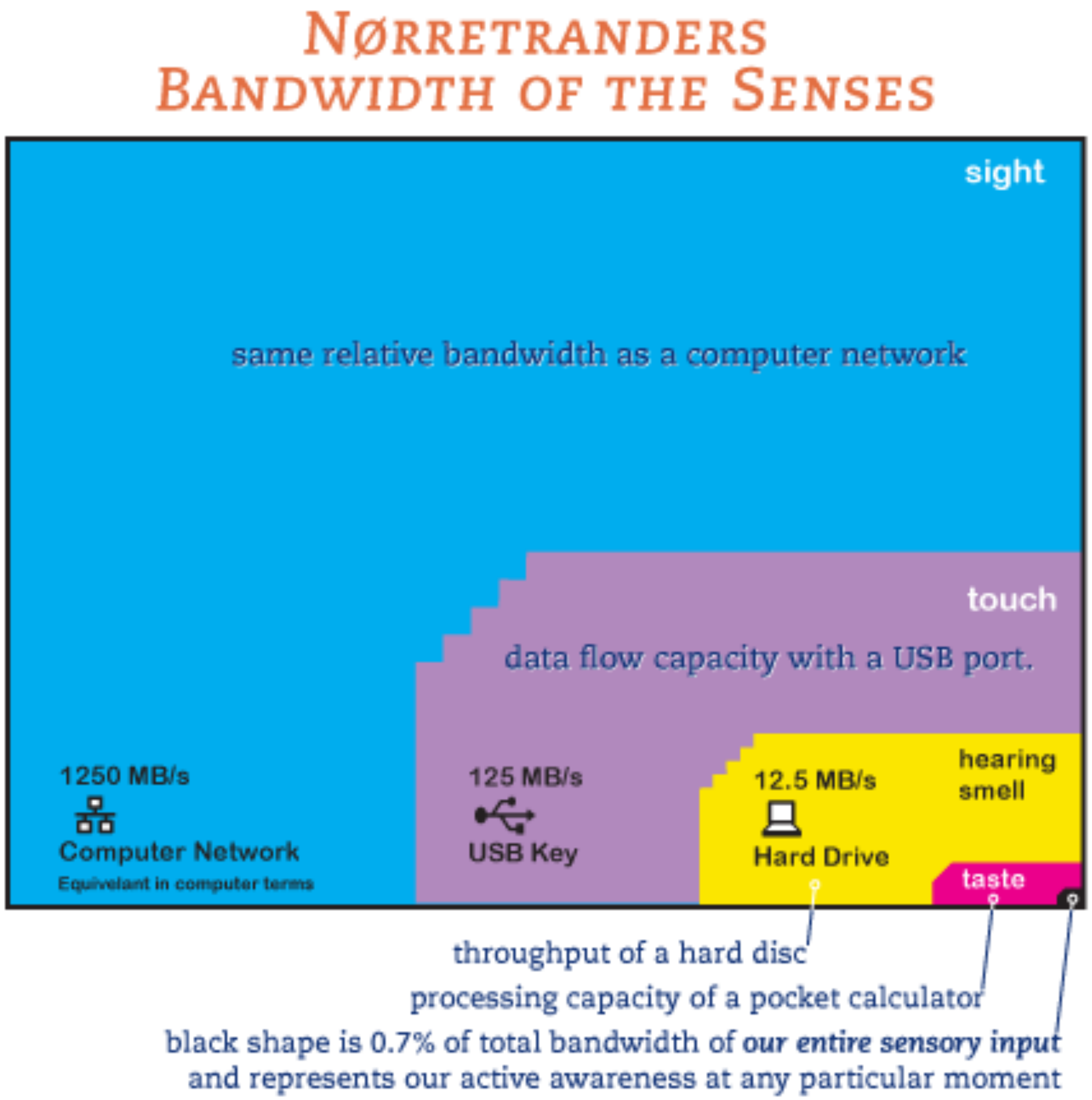
Human Perception

http://abstrusegoose.com/421



In the grand scheme of things,
we're all pretty much blind and deaf.

Human Senses



Source: http://chitownmediapsych.blogspot.ie/2010/09/context-will-be-king-working-title.html
[2]David McCandless : Information Is Beautiful

Perception

Perception

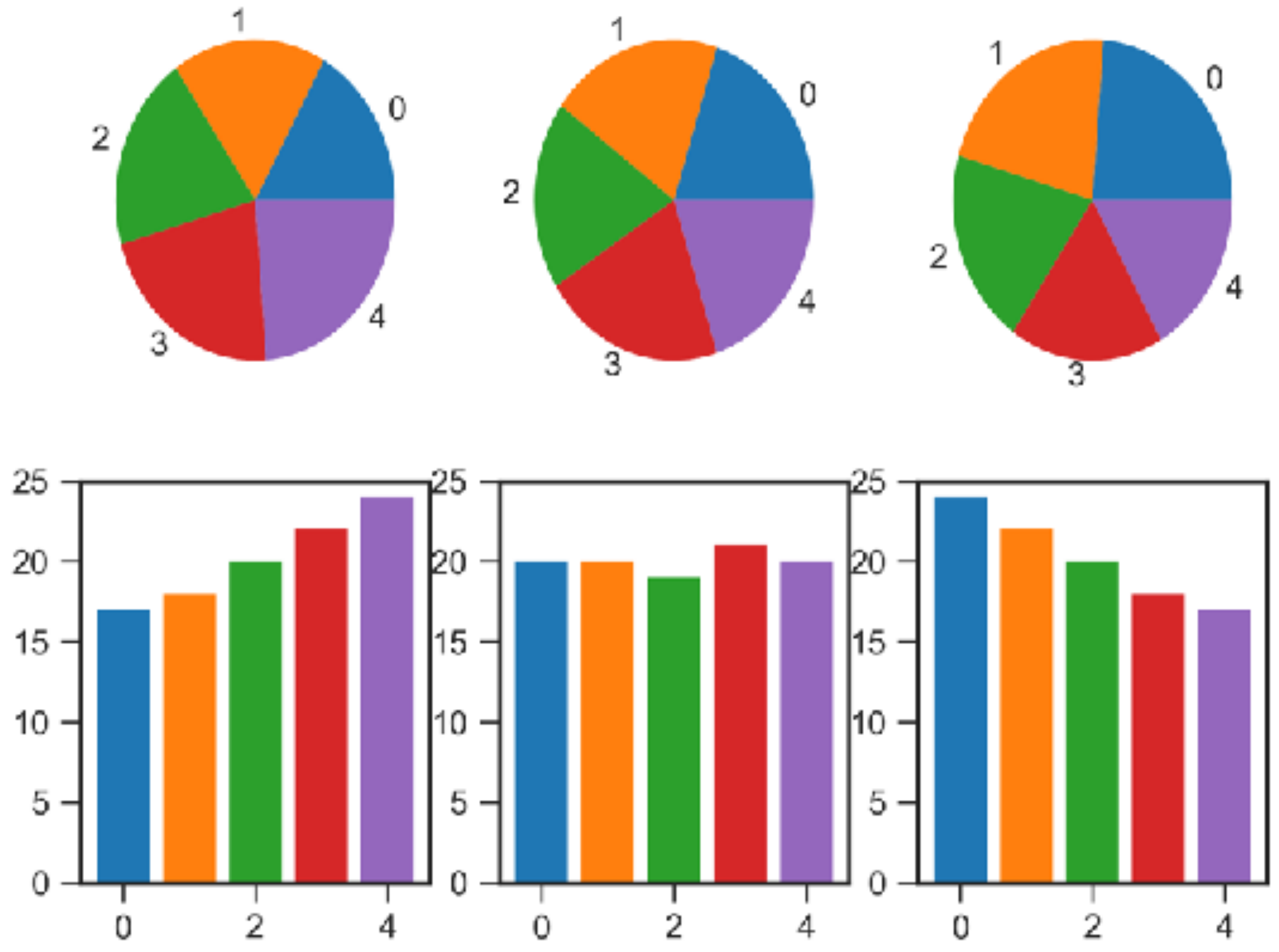
- Some cognitive tasks are significantly easier than others. In order, we are good a distinguishing:

Perception

- Some cognitive tasks are significantly easier than others. In order, we are good a distinguishing:
    - Position, length

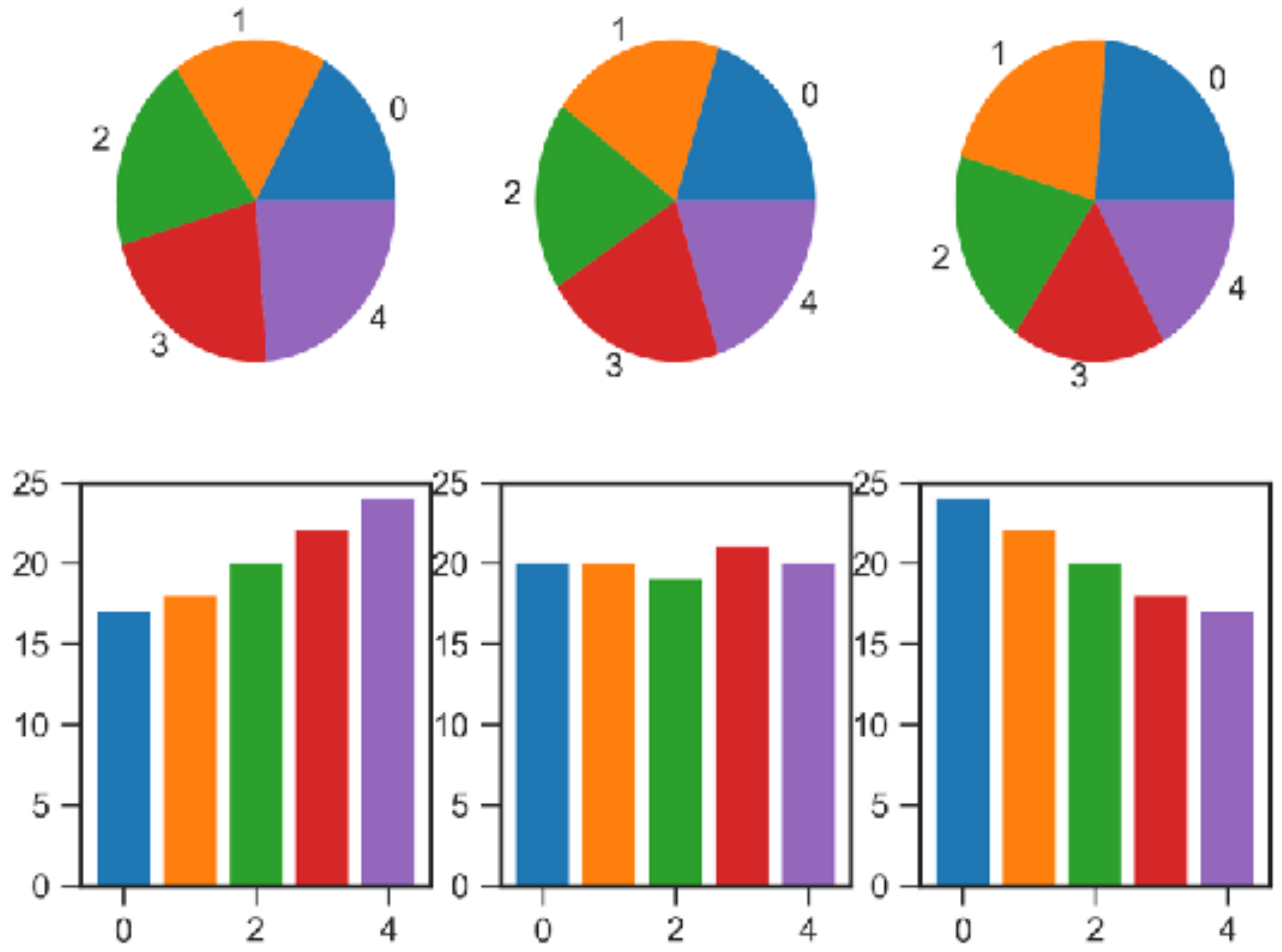
Perception

- Some cognitive tasks are significantly easier than others. In order, we are good a distinguishing:
    - Position, length
    - Direction, Angle, Area



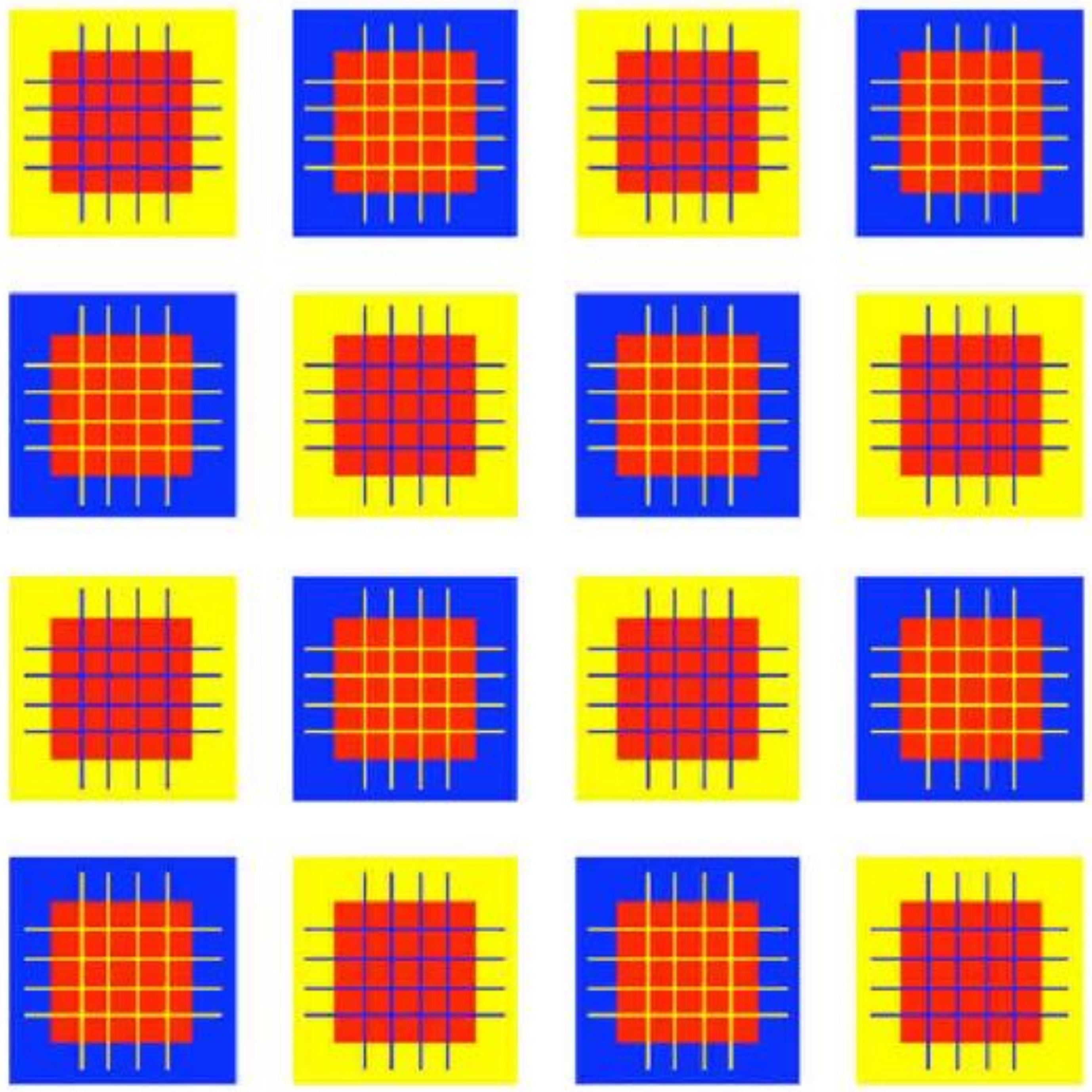
Perception

- Some cognitive tasks are significantly easier than others. In order, we are good a distinguishing:
    - Position, length
    - Direction, Angle, Area
    - Volume, Curvature, Shade



Perception

- Some cognitive tasks are significantly easier than others. In order, we are good a distinguishing:
    - Position, length
    - Direction, Angle, Area
    - Volume, Curvature, Shade
    - Color Saturation.

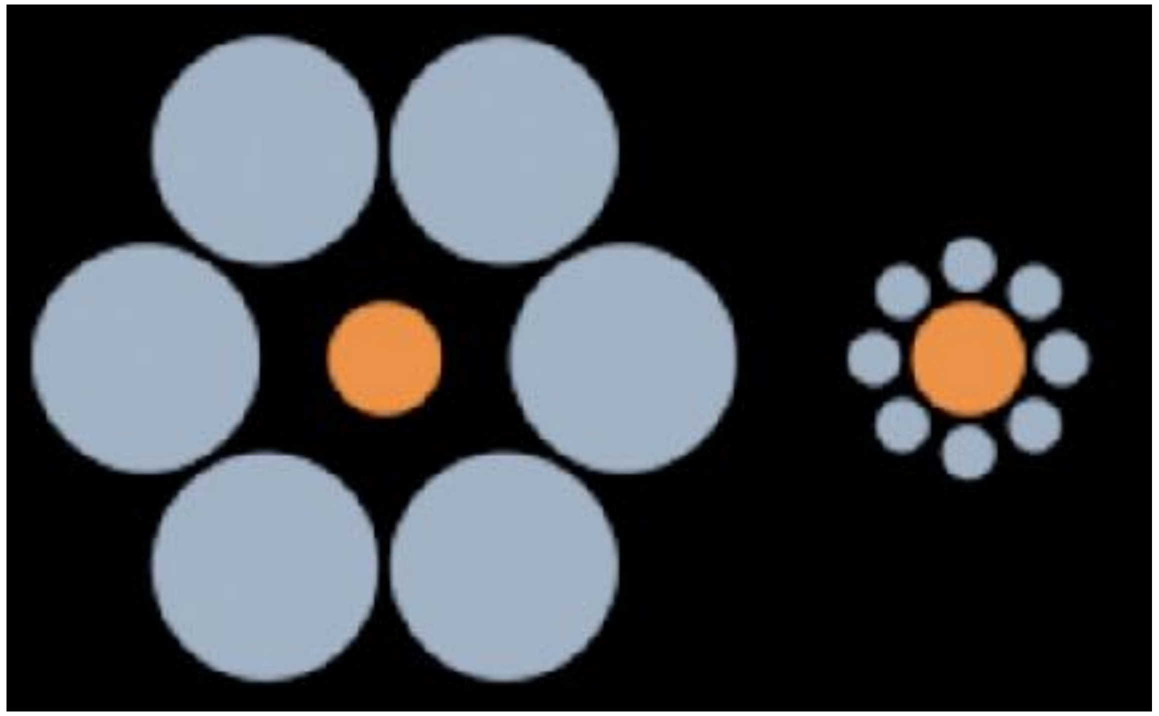


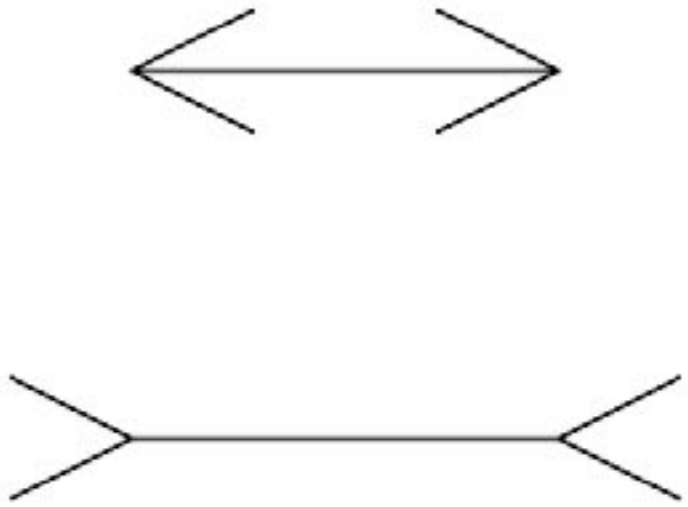
Perception

- Some cognitive tasks are significantly easier than others. In order, we are good a distinguishing:
    - Position, length
    - Direction, Angle, Area
    - Volume, Curvature, Shade
    - Color Saturation.

Perception

- Some cognitive tasks are significantly easier than others. In order, we are good a distinguishing:
  - Position, length
  - Direction, Angle, Area
  - Volume, Curvature, Shade
  - Color Saturation.
- Context also matters!
  - An object seen in the context of larger objects will appear smaller, while in the content of smaller objects it will appear larger.

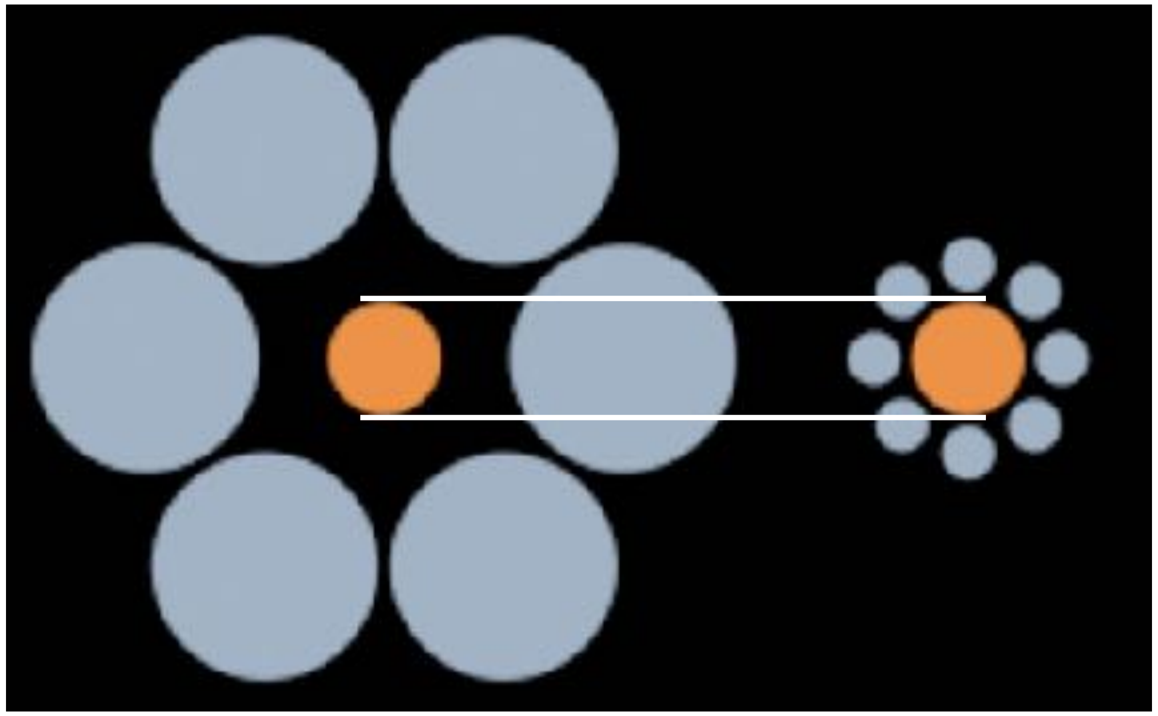


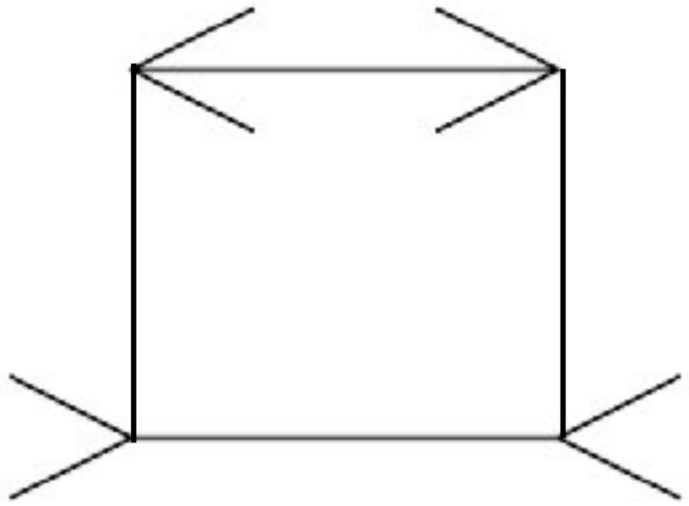


@bgoncalves

Perception

- Some cognitive tasks are significantly easier than others. In order, we are good a distinguishing:
  - Position, length
  - Direction, Angle, Area
  - Volume, Curvature, Shade
  - Color Saturation.
- Context also matters!
  - An object seen in the context of larger objects will appear smaller, while in the content of smaller objects it will appear larger.

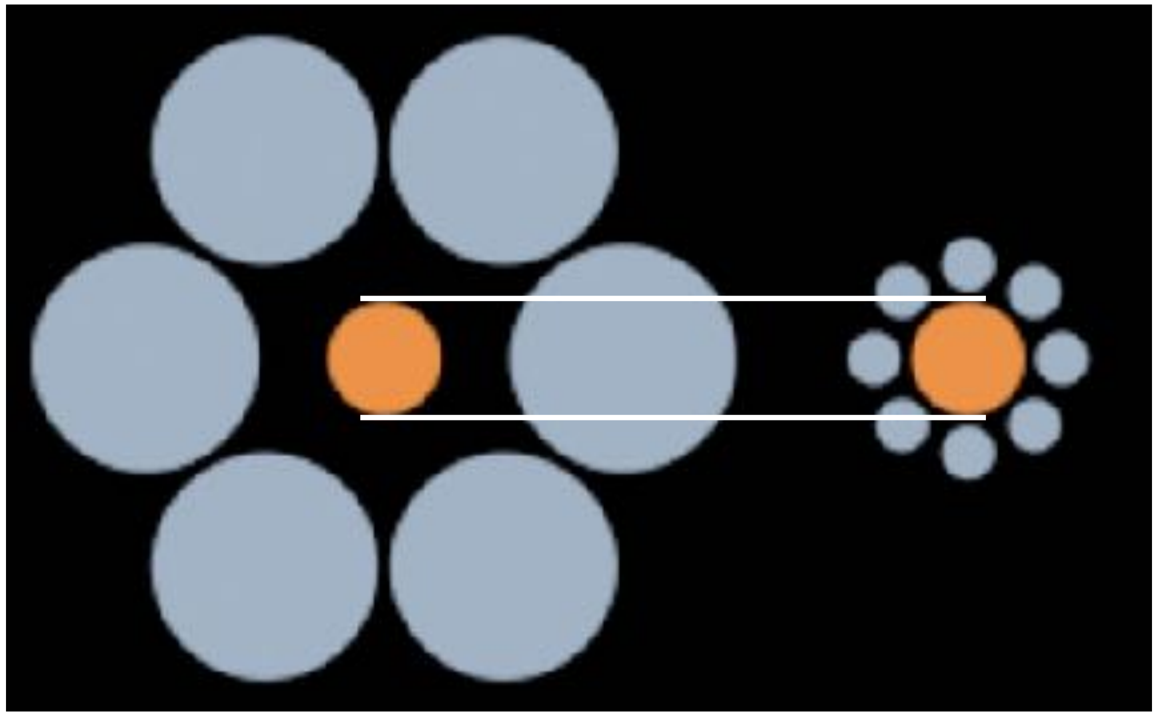


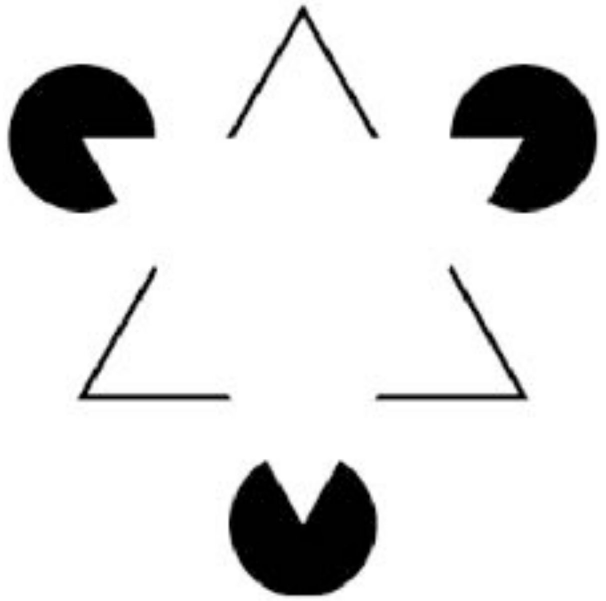


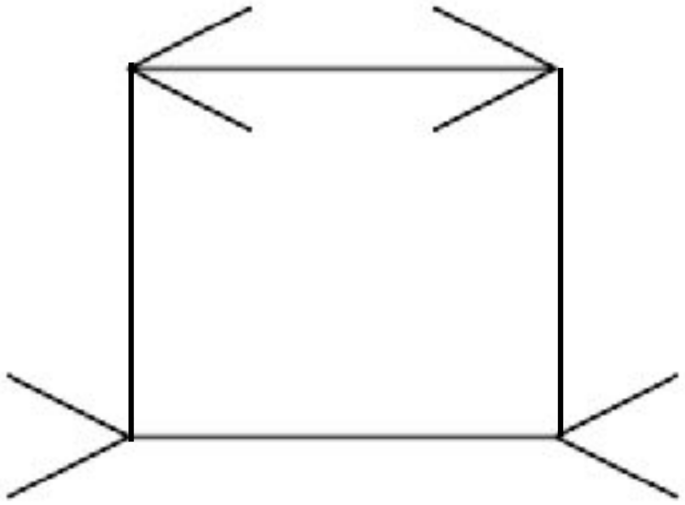
@bgoncalves

Perception

- Some cognitive tasks are significantly easier than others. In order, we are good a distinguishing:
  - Position, length
  - Direction, Angle, Area
  - Volume, Curvature, Shade
  - Color Saturation.
- Context also matters!
  - An object seen in the context of larger objects will appear smaller, while in the content of smaller objects it will appear larger.
  - And we "fill in the gaps"

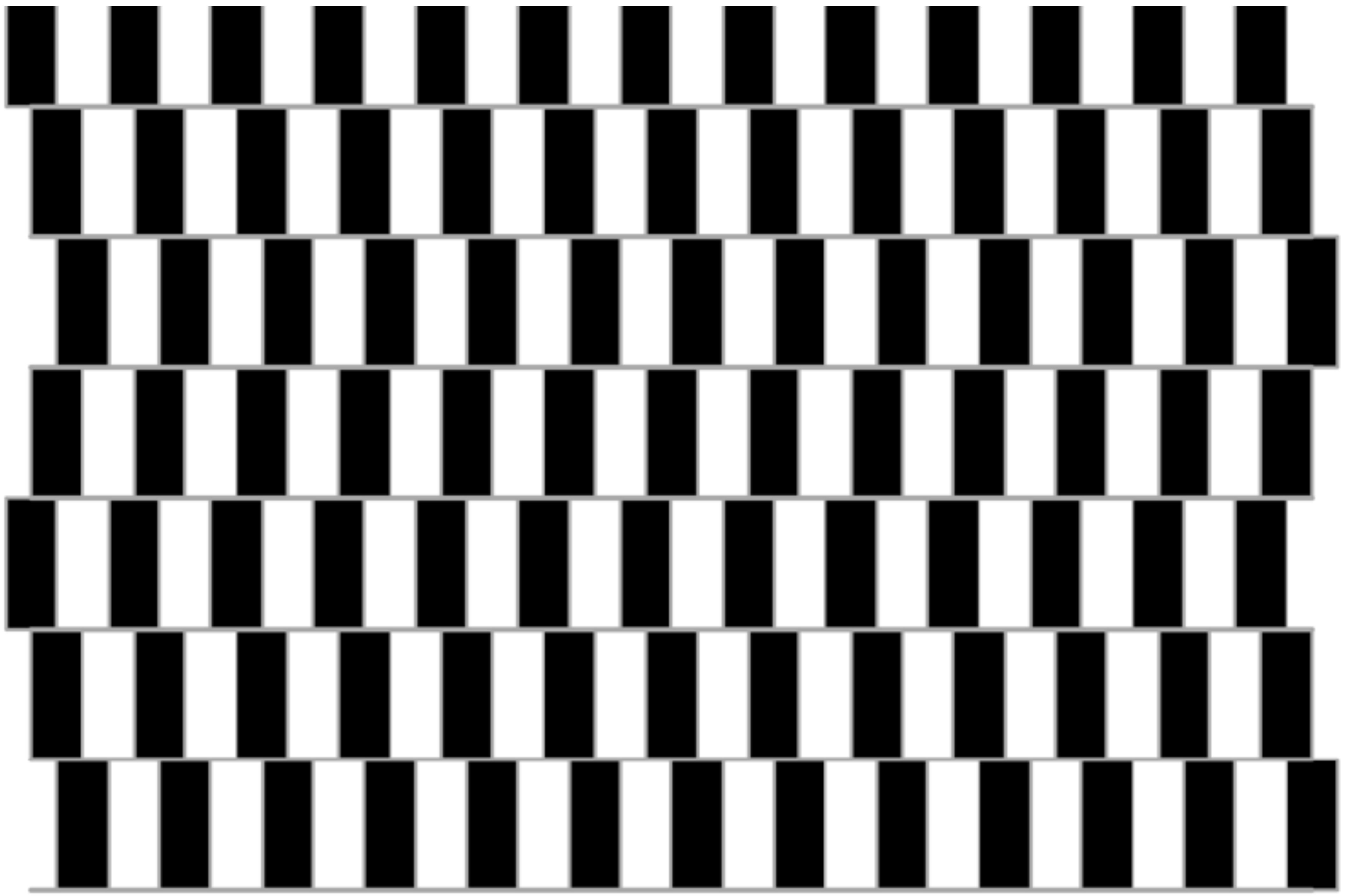






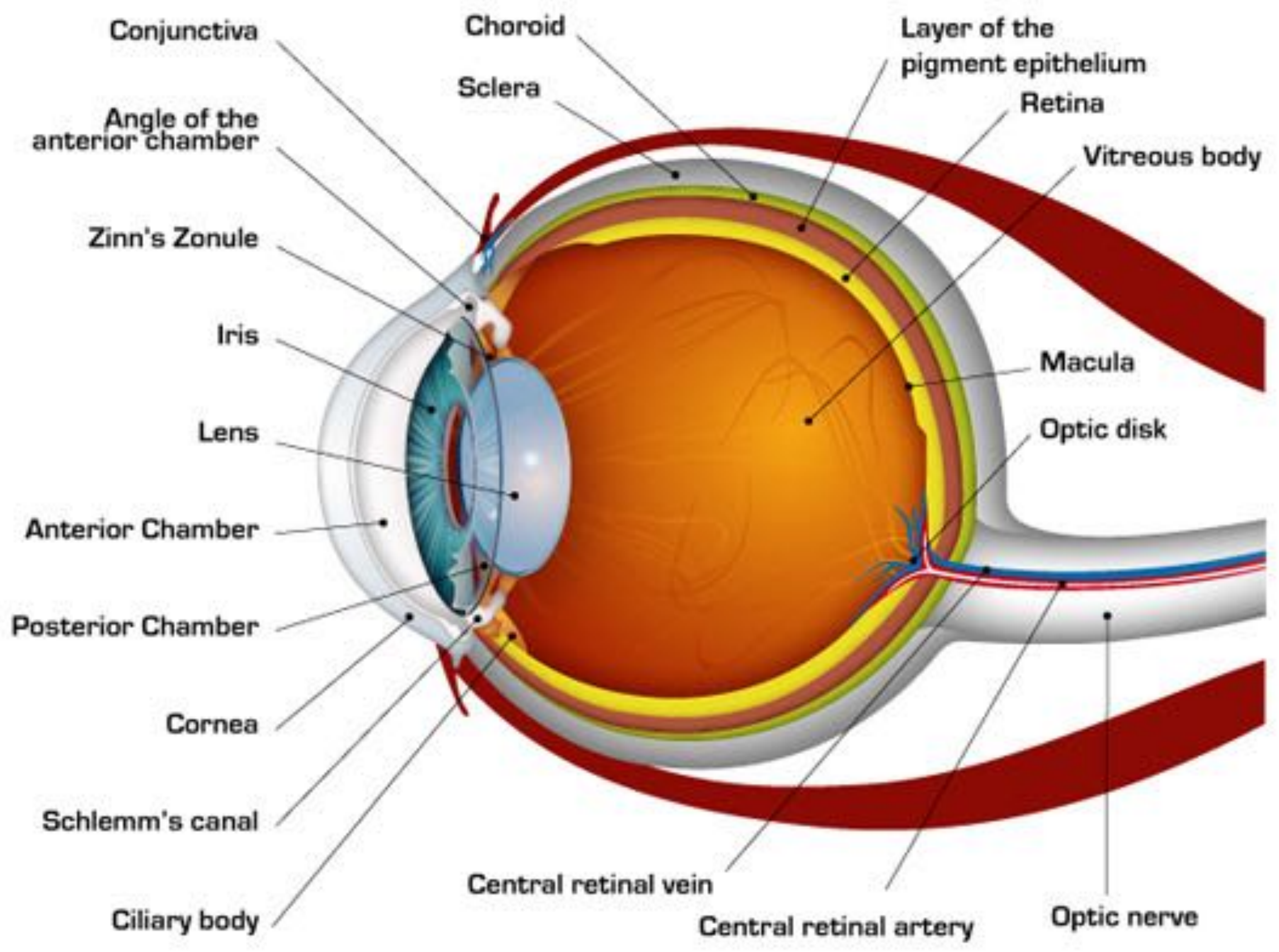
@bgoncalves

Perception Biases



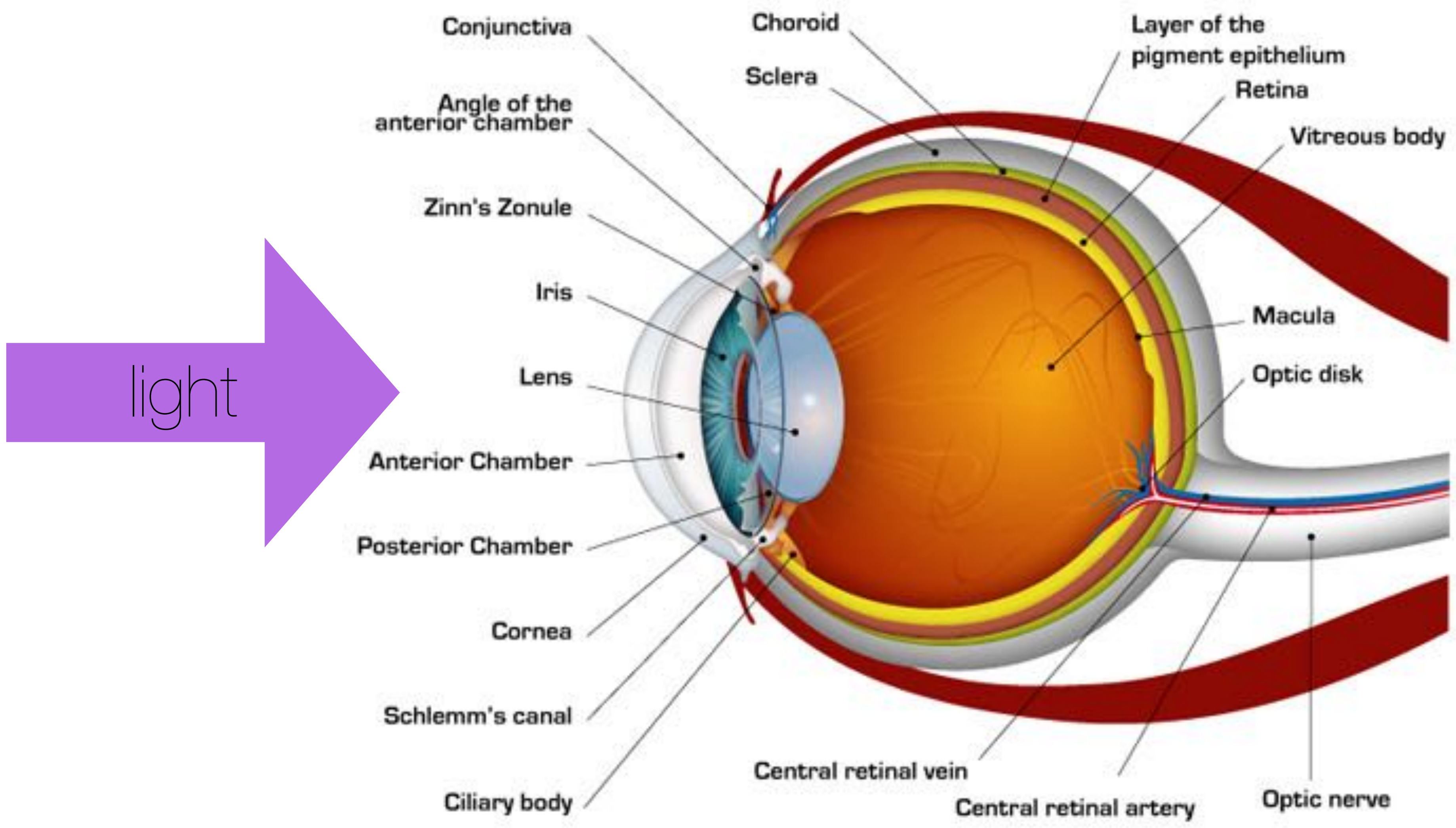
Human Vision

https://en.wikipedia.org/wiki/Photoreceptor_cell



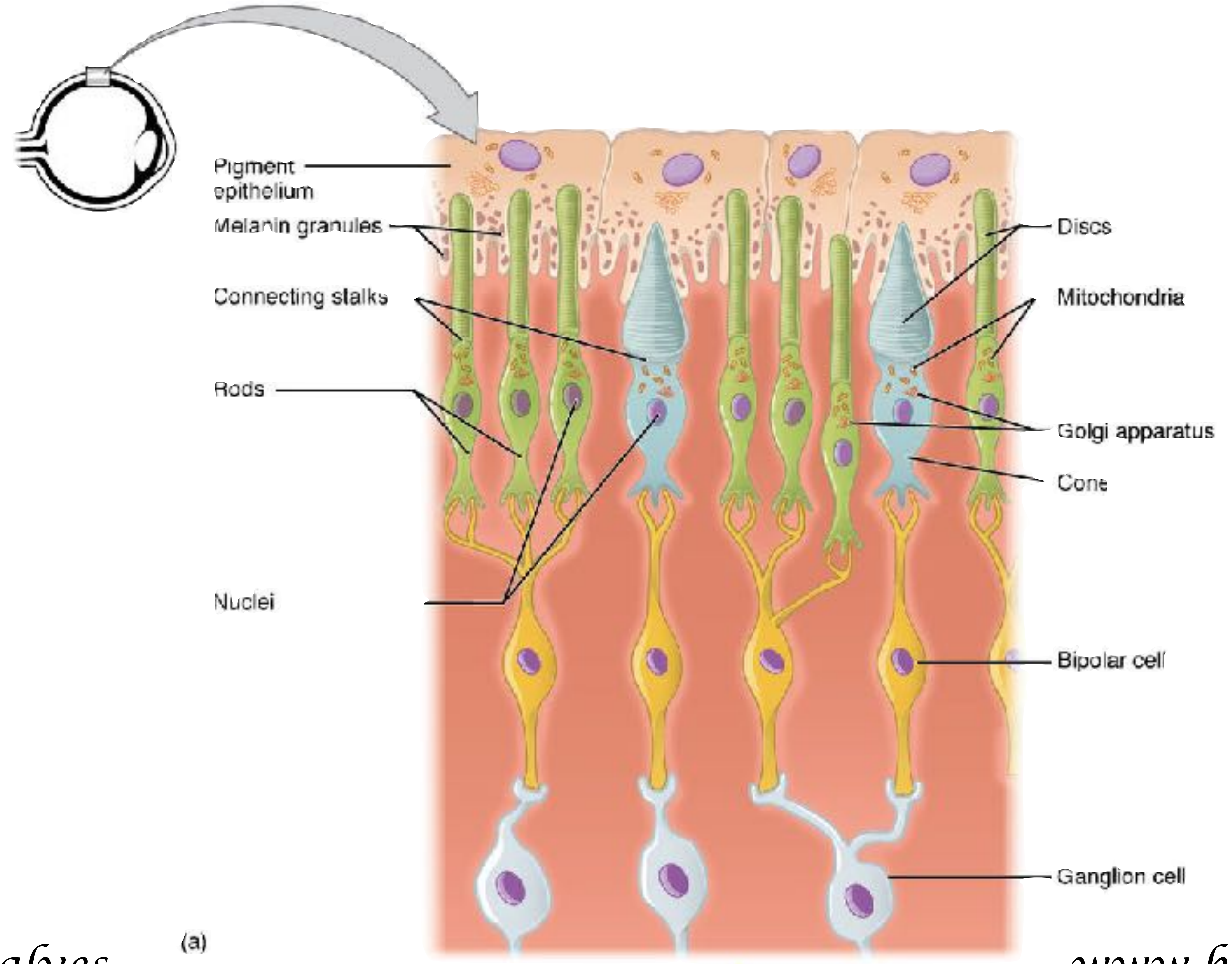
Human Vision

https://en.wikipedia.org/wiki/Photoreceptor_cell



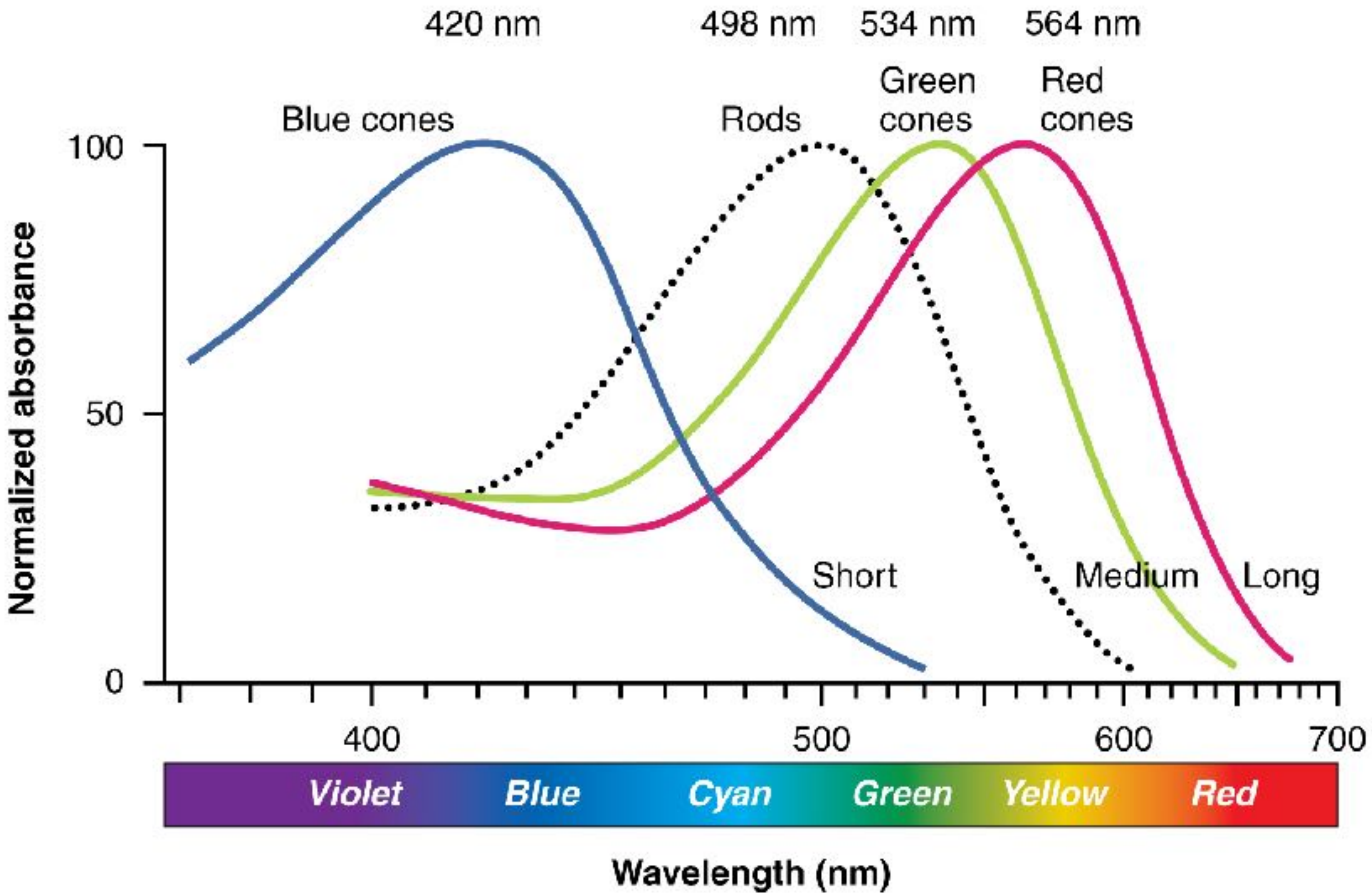
Human Vision

https://en.wikipedia.org/wiki/Photoreceptor_cell



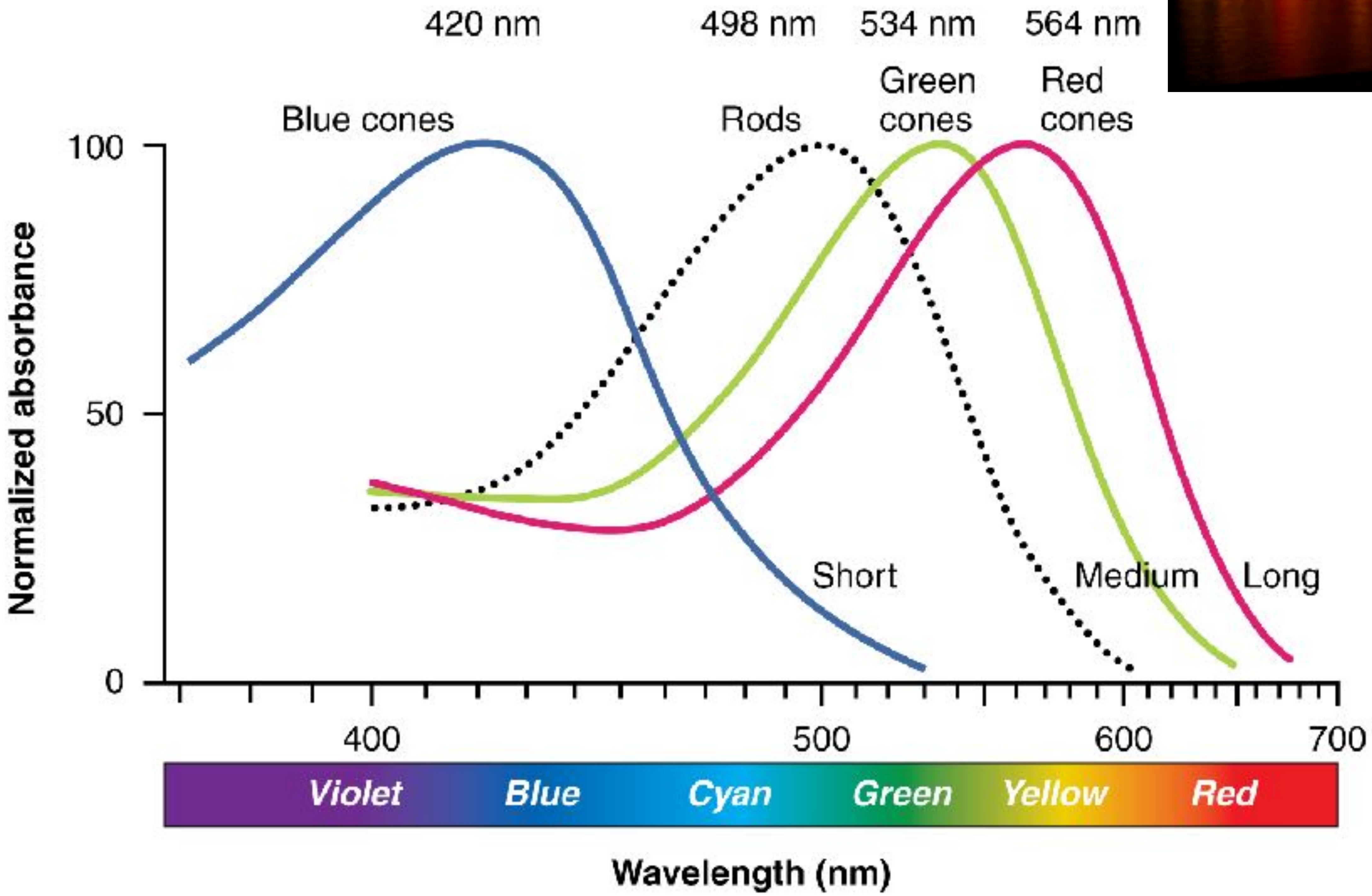
Human Vision

https://en.wikipedia.org/wiki/Photoreceptor_cell



Human Vision

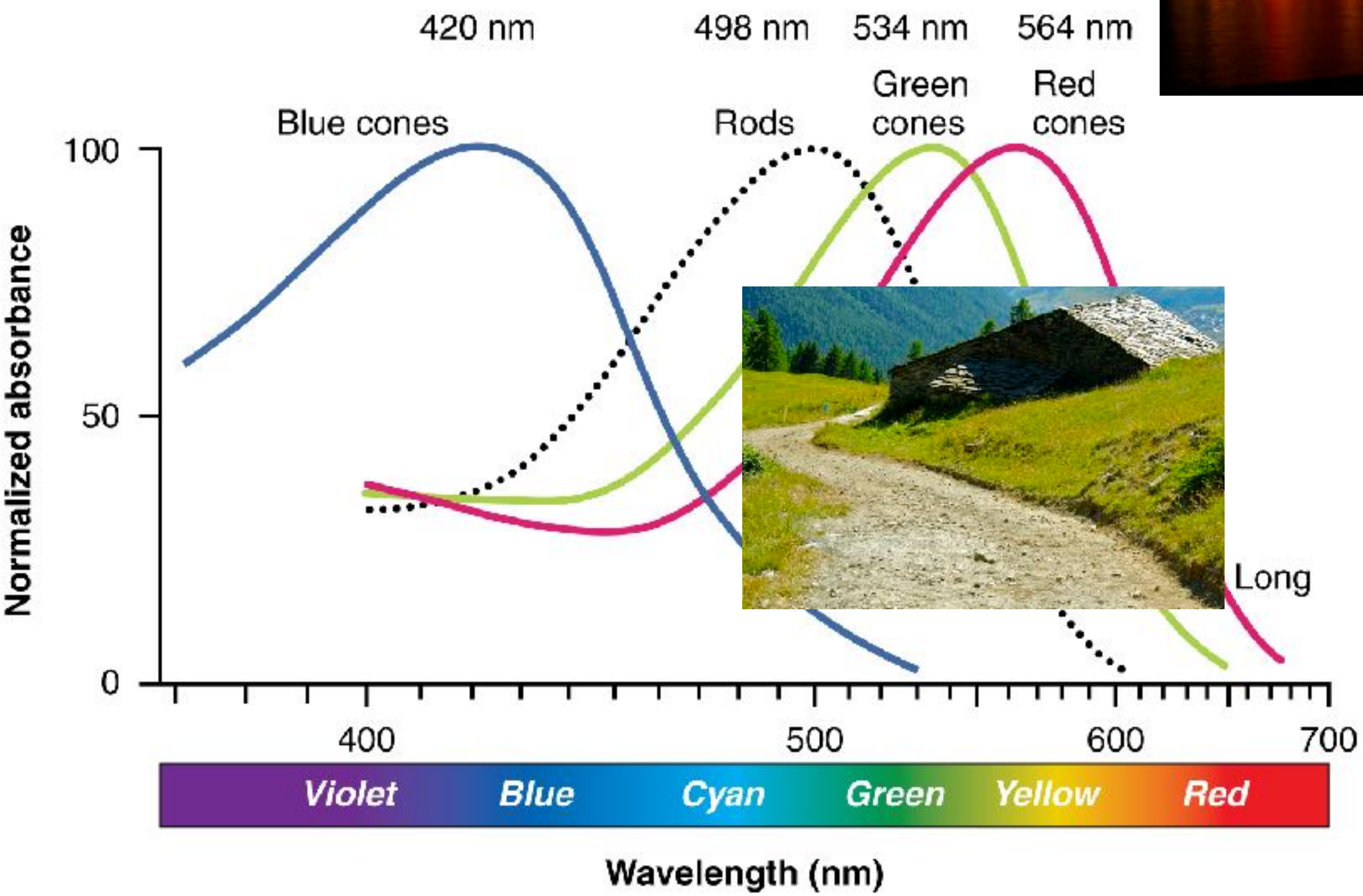
https://en.wikipedia.o ell



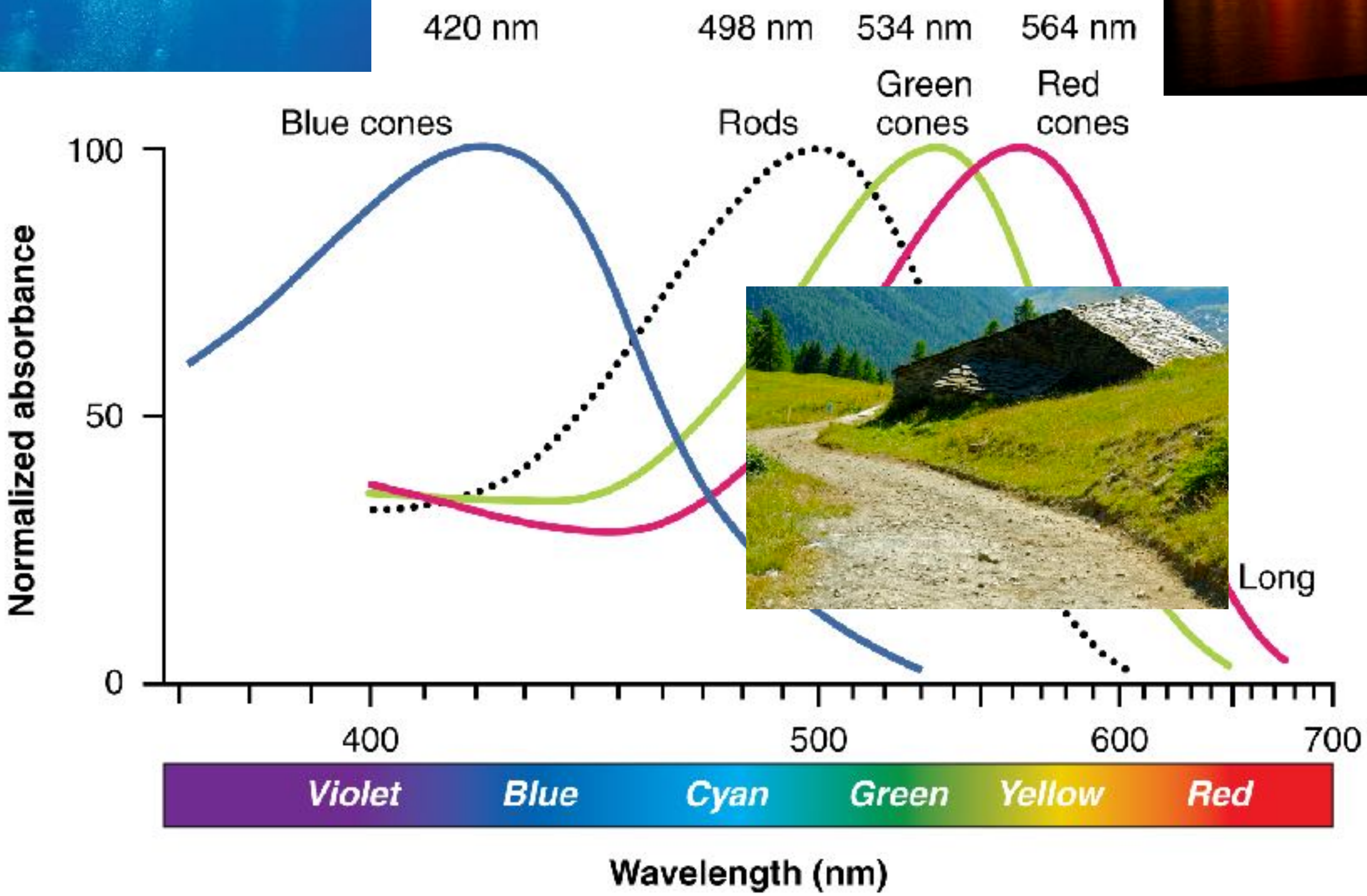
@bgoncalves www.bgoncalves.com

Human Vision

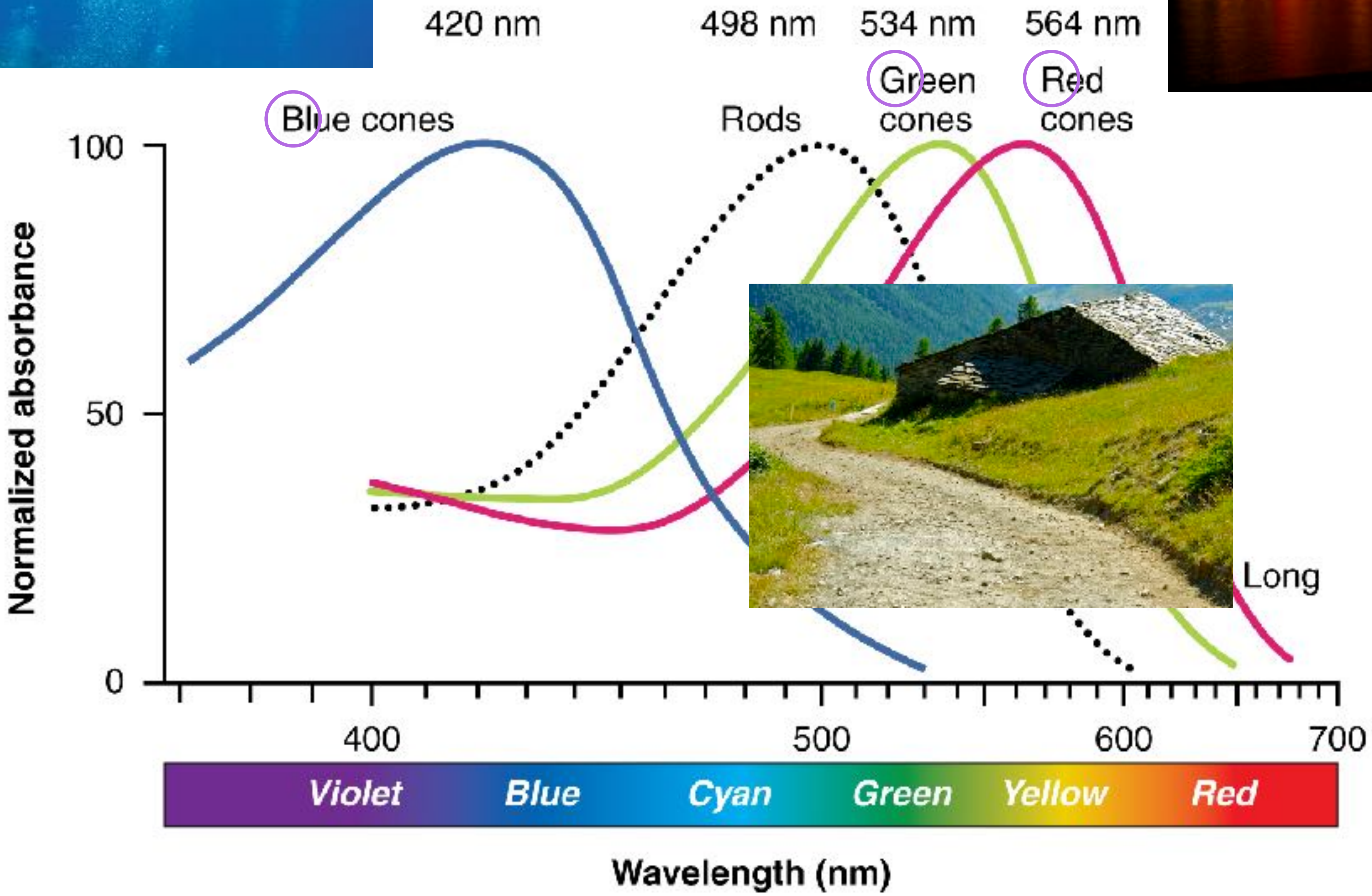
https://en.wikipedia.o ell



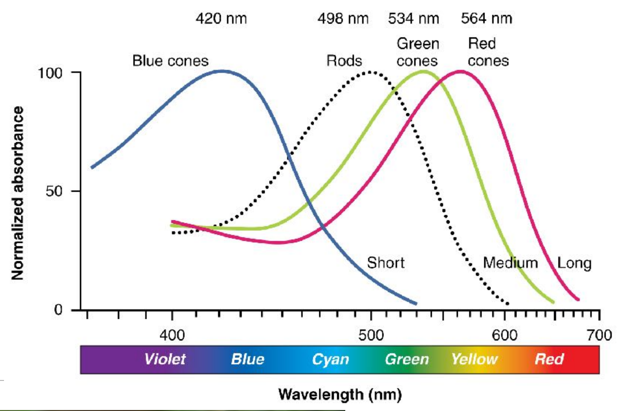
https://en.wikipedia.o ell



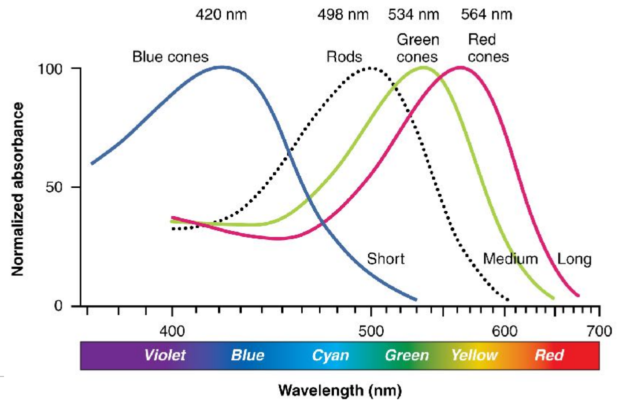
https://en.wikipedia.o ell



Colors galore!



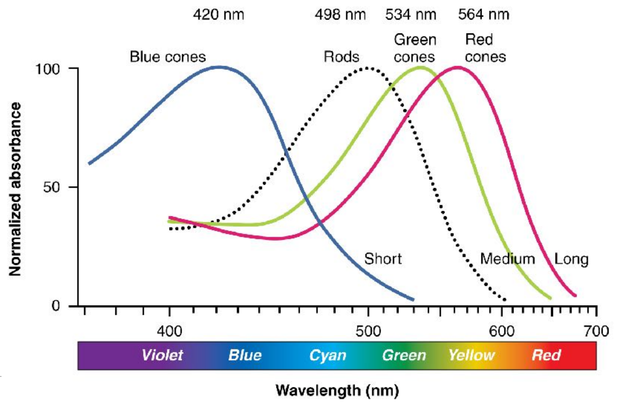
Color Perception

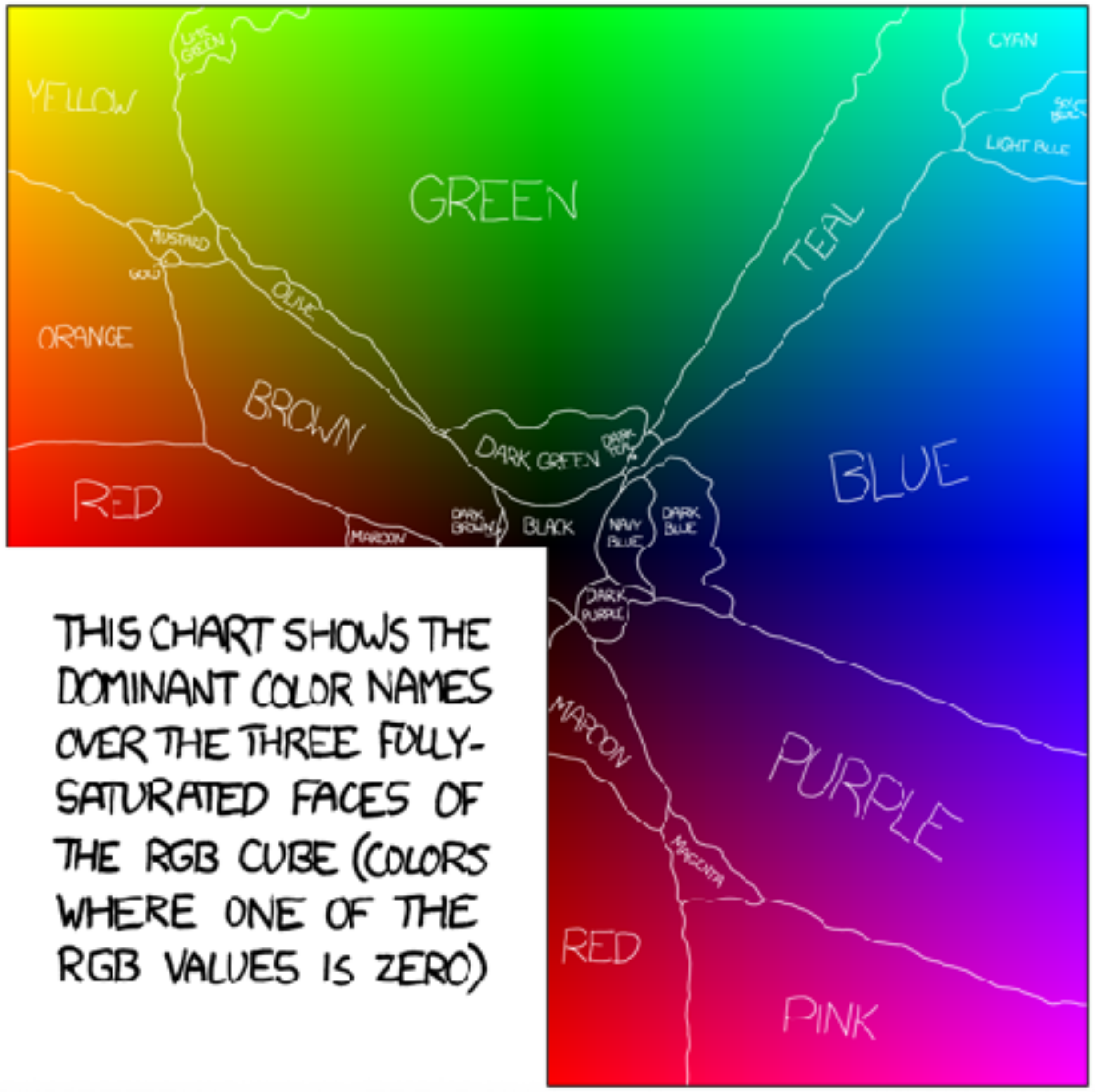


"Who in the rainbow can draw the line where the violet tint ends and the orange tint begins? Distinctly we see the difference of the colors, but where exactly does the one first blendingly enter into the other? So with sanity and insanity."
(H. Melville)

@bgoncalves

Color Perception



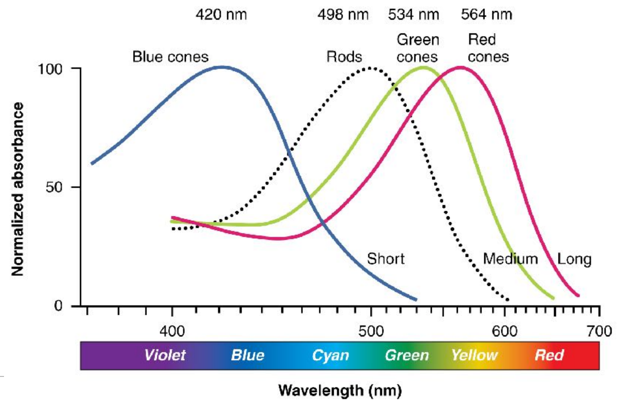


THIS CHART SHOWS THE DOMINANT COLOR NAMES OVER THE THREE FULLY-SATURATED FACES OF THE RGB CUBE (COLORS WHERE ONE OF THE RGB VALUES IS ZERO)

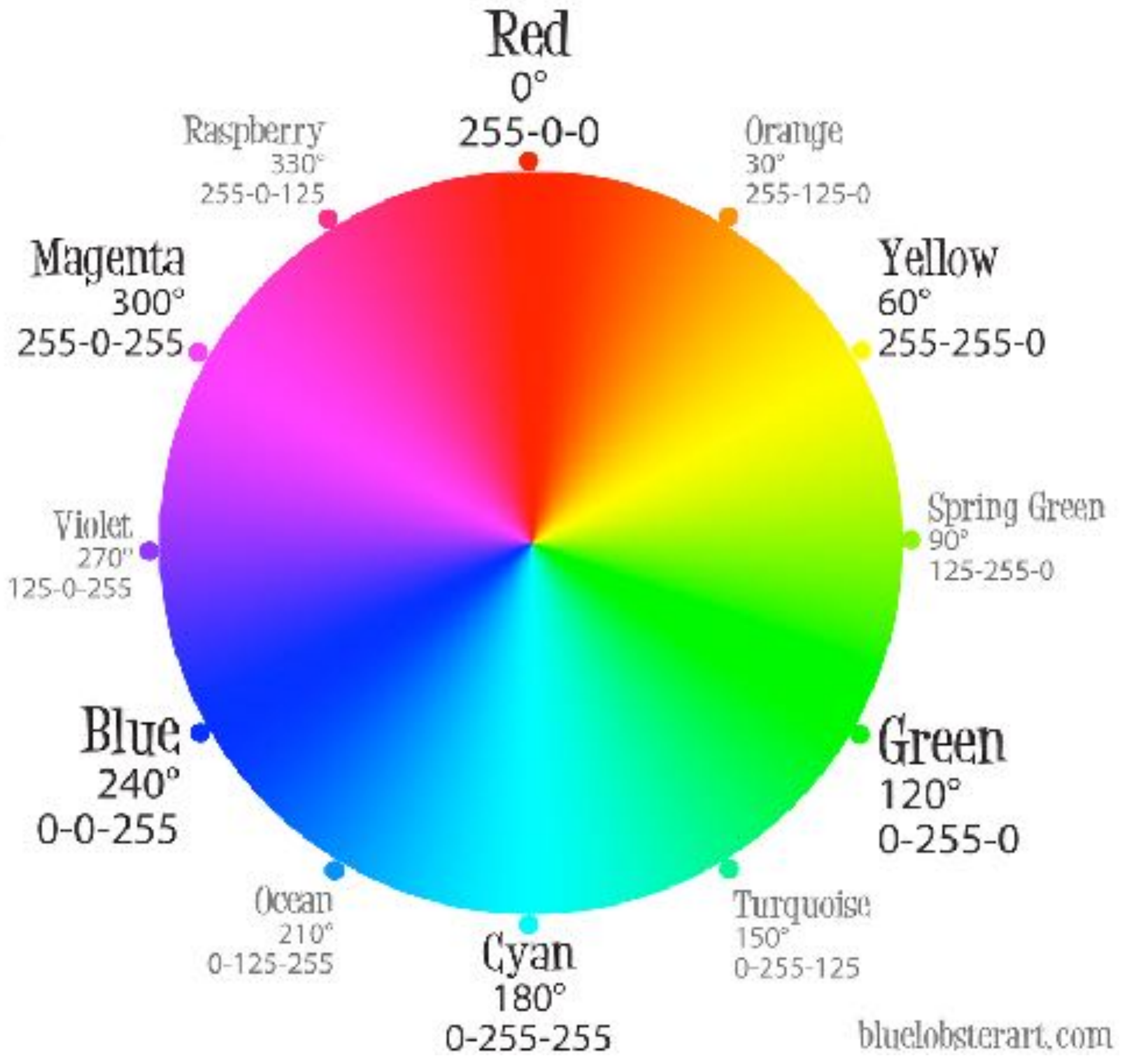
"Who in the rainbow can draw the line where the violet tint ends and the orange tint begins? Distinctly we see the difference of the colors, but where exactly does the one first blendingly enter into the other? So with sanity and insanity."
(H. Melville)

@bgoncalves

Color Wheel

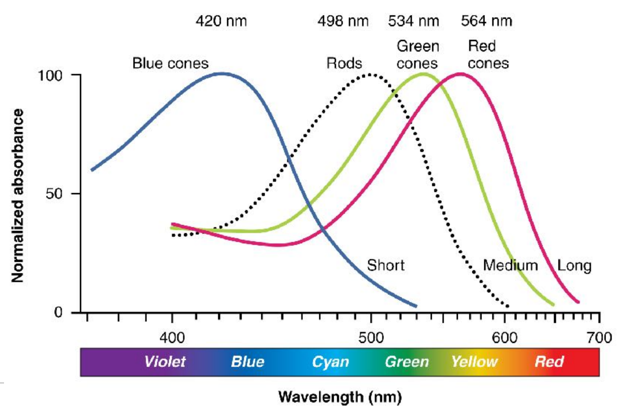


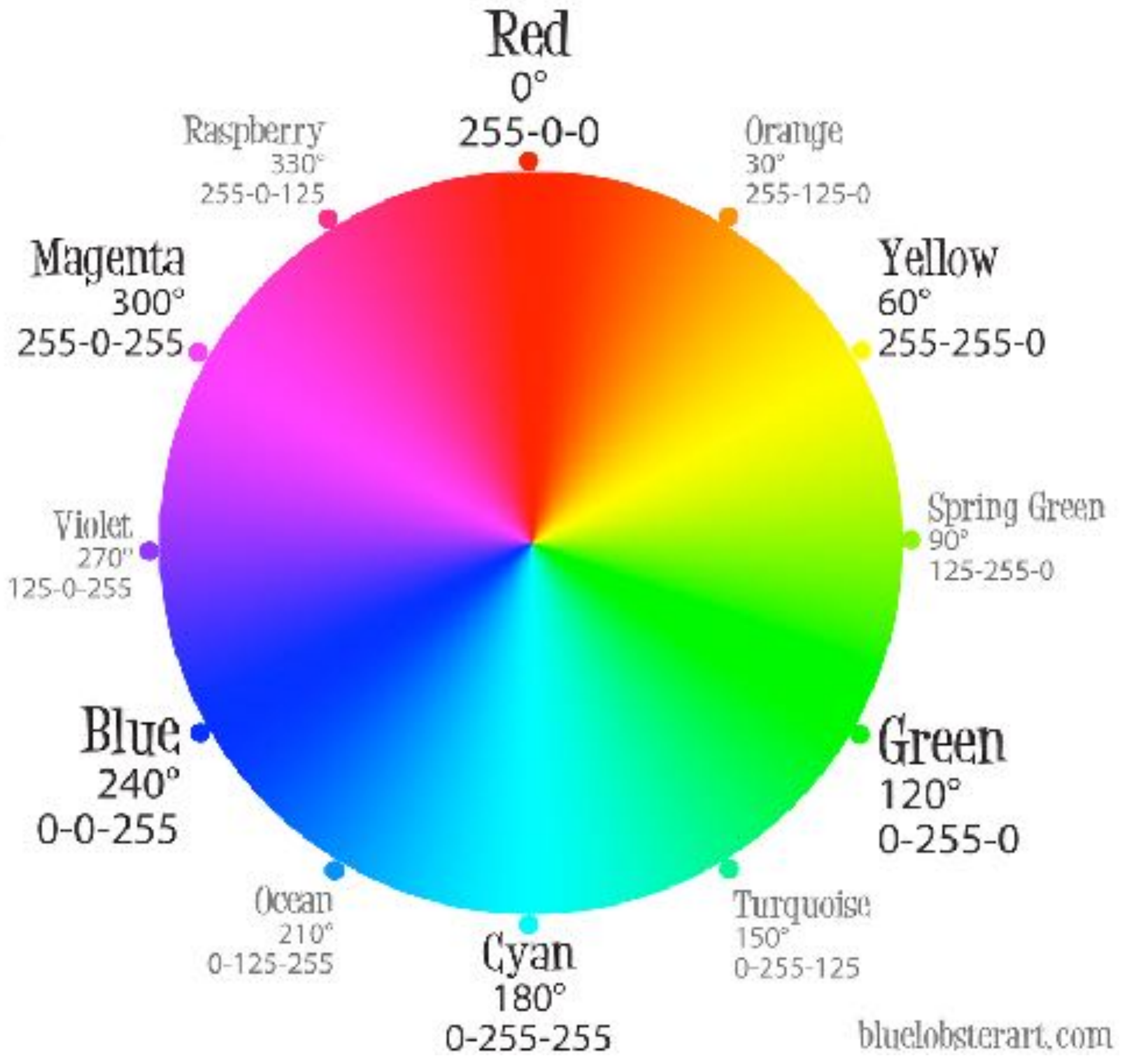
420 nm
498 nm
534 nm
564 nm
Blue cones
Rods
Green cones
Red cones
Normalized absorbance
100
50
0
Short
Medium
Long
400
500
600
700
Violet
Blue
Cyan
Green
Yellow
Red
Wavelength (nm)

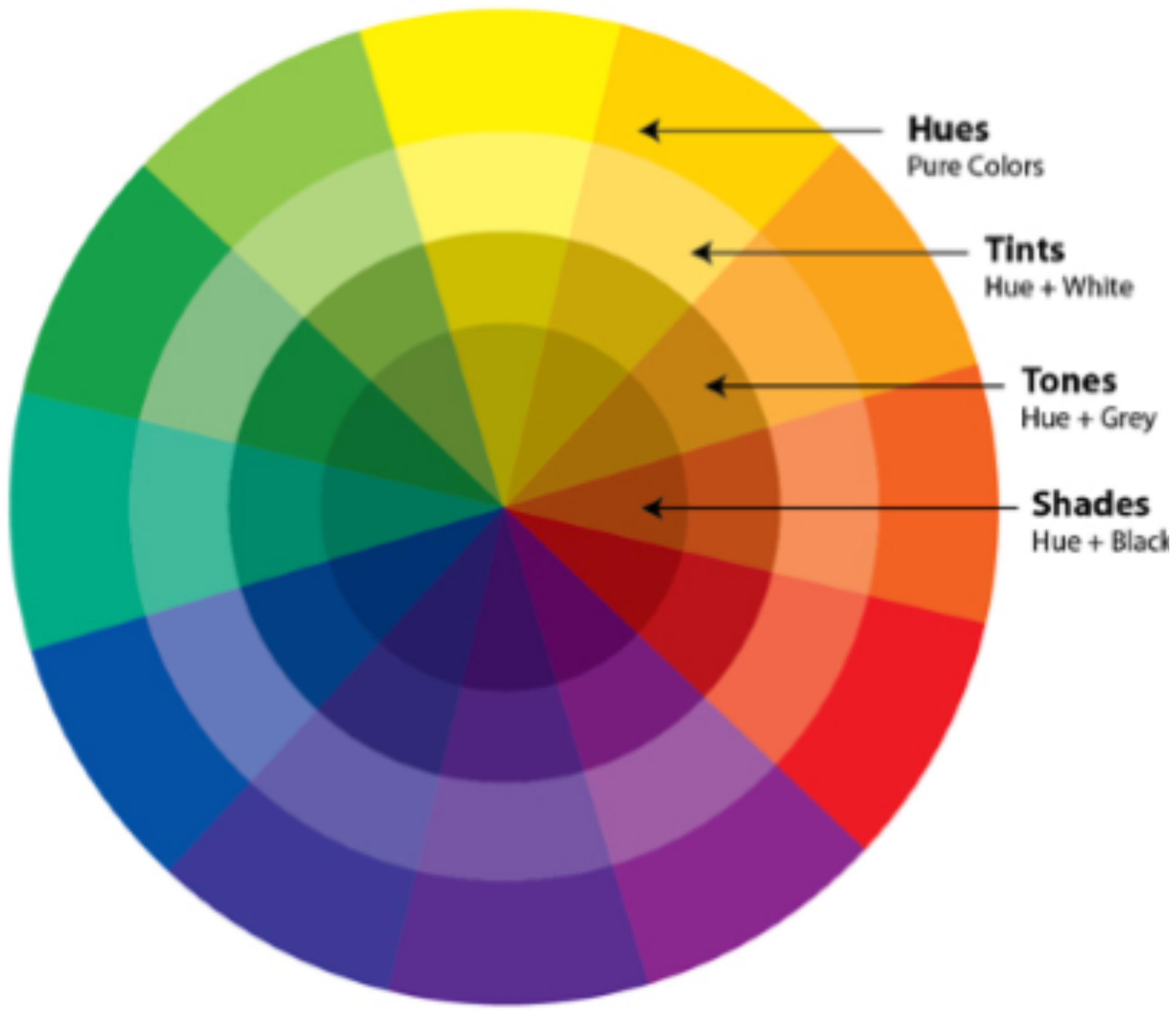


Red
0°
255-0-0
Orange
30°
255-125-0
Yellow
60°
255-255-0
Spring Green
90°
125-255-0
Green
120°
0-255-0
Turquoise
150°
0-255-125
Cyan
180°
0-255-255
Ocean
210°
0-125-255
Blue
240°
0-0-255
Violet
270°
125-0-255
Magenta
300°
255-0-255
Raspberry
330°
255-0-125
bluelobsterart.com

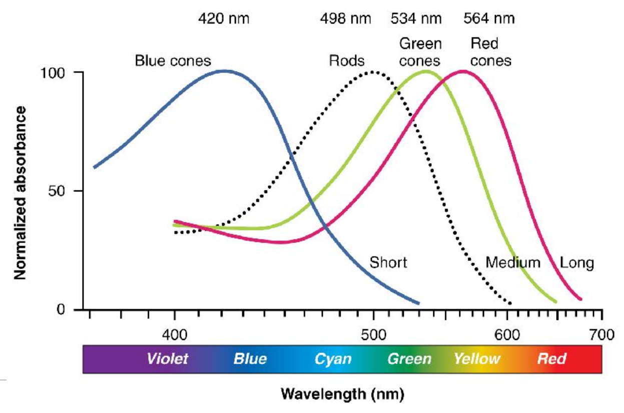
Color Wheel


420 nm
498 nm
534 nm
564 nm
Blue cones
Rods
Green cones
Red cones
Normalized absorbance
100
50
0
Short
Medium
Long
400
500
600
700
Violet
Blue
Cyan
Green
Yellow
Red
Wavelength (nm)

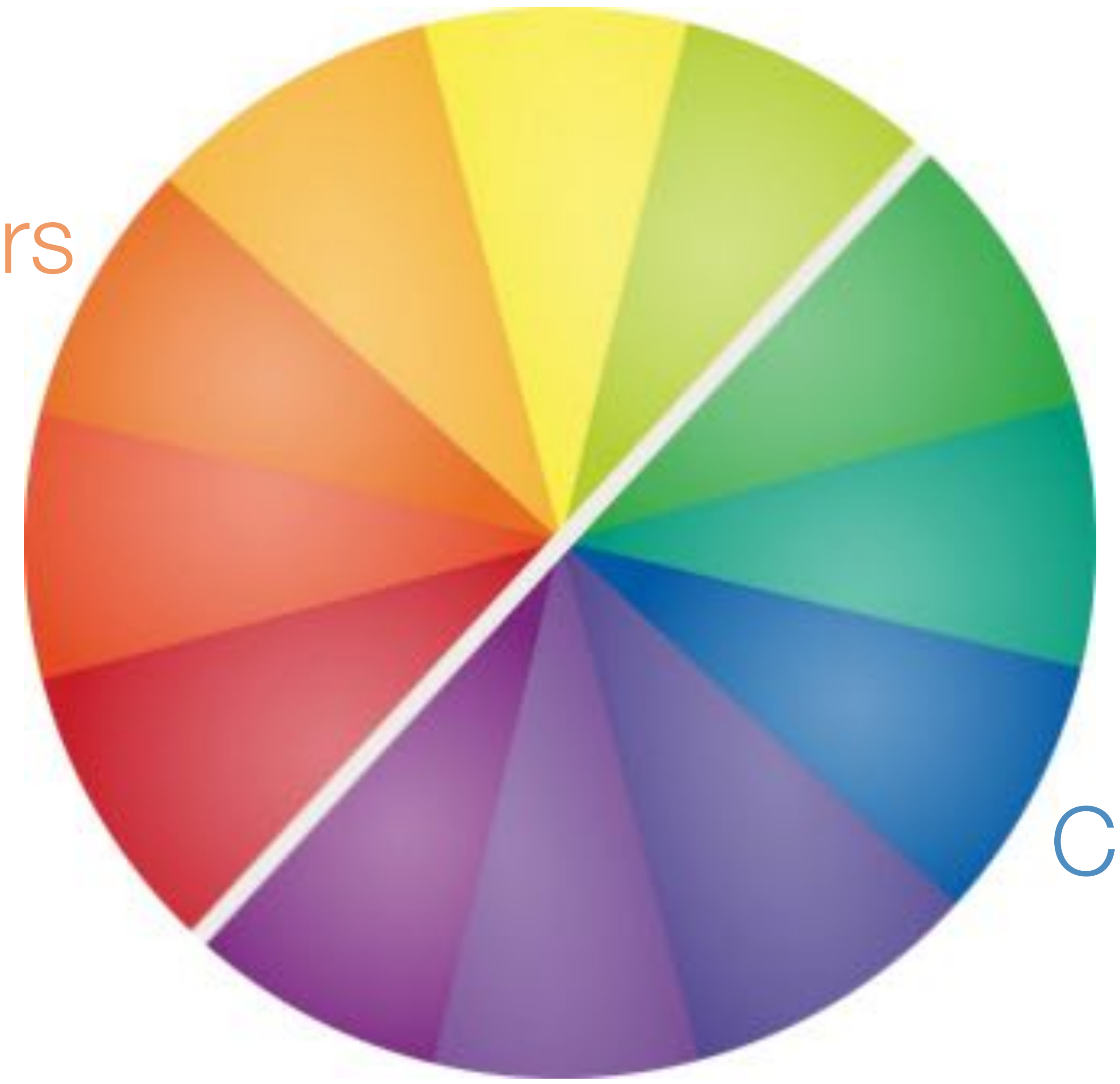

Red
0°
255-0-0
Raspberry
330°
255-0-125
Orange
30°
255-125-0
Magenta
300°
255-0-255
Yellow
60°
255-255-0
Violet
270°
125-0-255
Spring Green
90°
125-255-0
Blue
240°
0-0-255
Green
120°
0-255-0
Ocean
210°
0-125-255
Turquoise
150°
0-255-125
Cyan
180°
0-255-255
bluelobsterart.com


Hues
Pure Colors
Tints
Hue + White
Tones
Hue + Grey
Shades
Hue + Black

Color Schemes

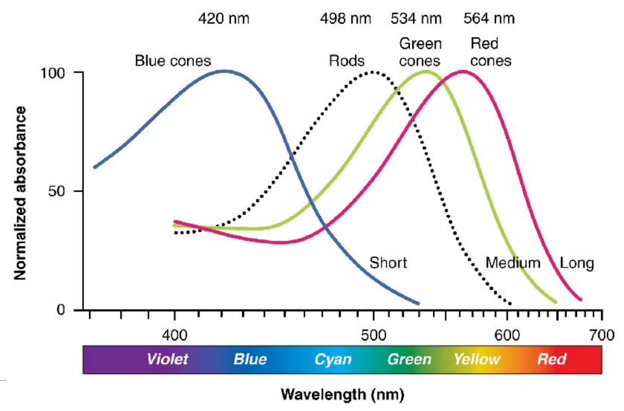


Warm Colors

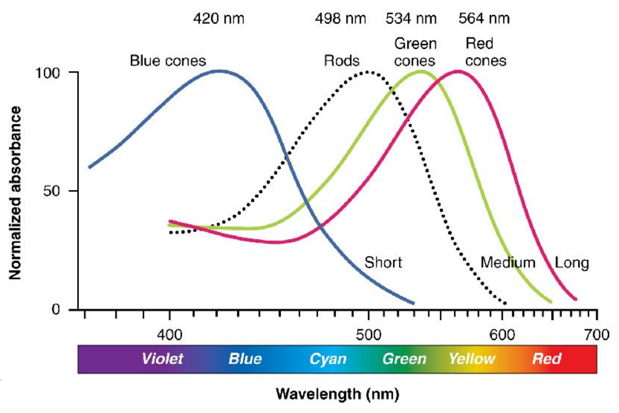


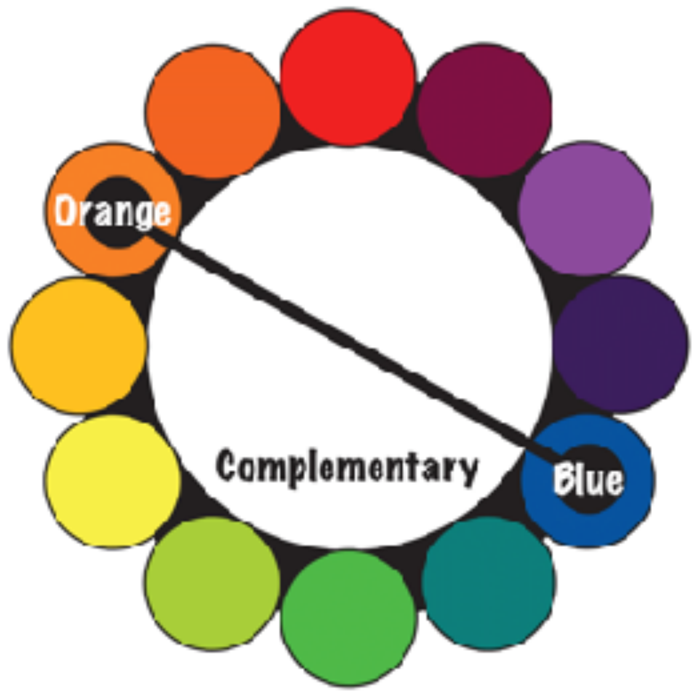
Cold Colors

Color Schemes



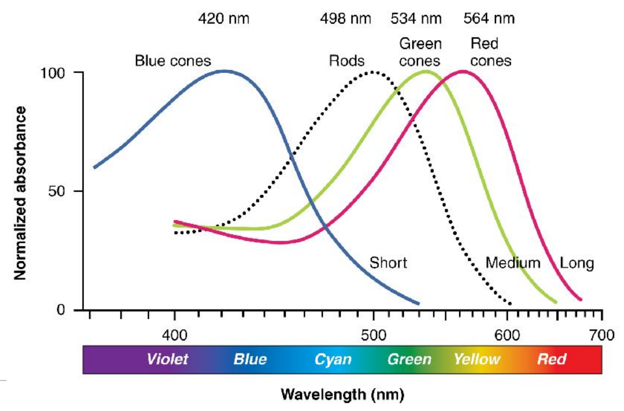
Color Schemes



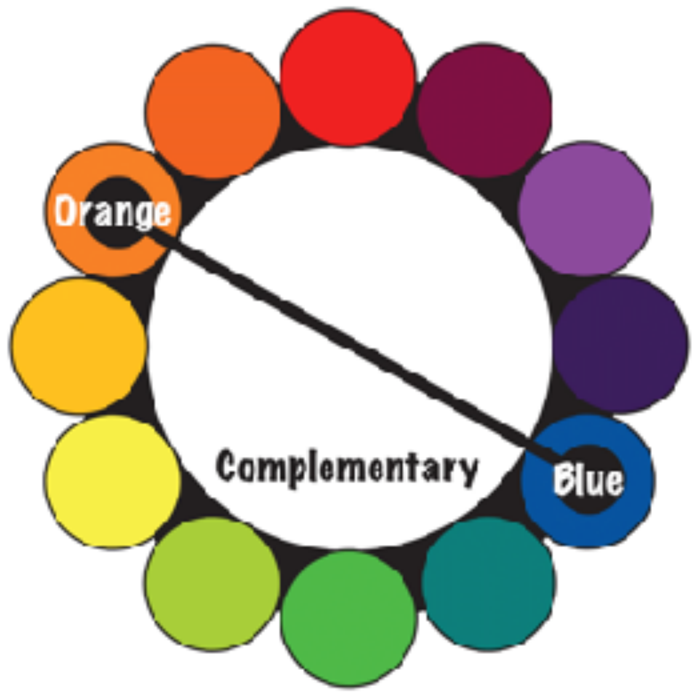


Complementary color scheme
Colors that are opposite each other on the color wheel are considered to be complementary colors

(example: Orange and Blue).

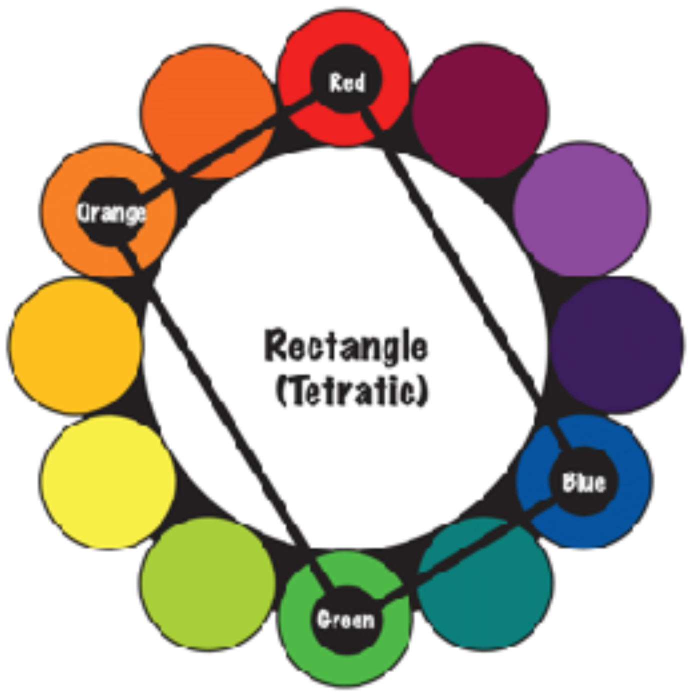


Color Schemes



Complementary color scheme
Colors that are opposite each other on the color wheel are considered to be complementary colors

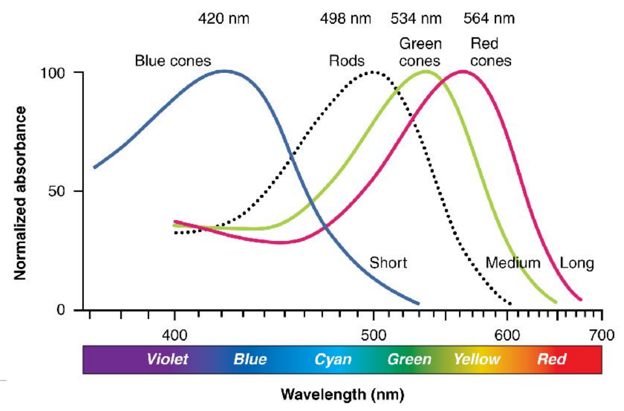
(example: Orange and Blue).

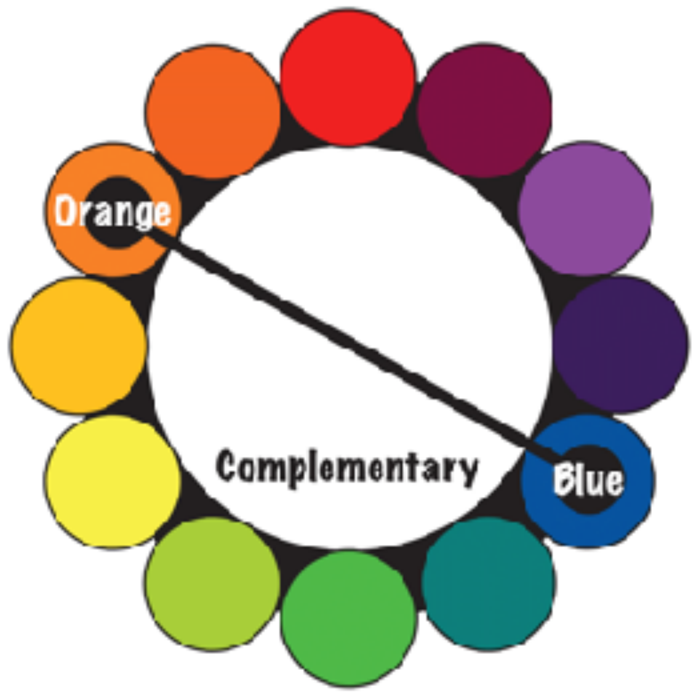


Rectangle (tetradic) color scheme
The rectangle or tetradic color scheme uses four colors arranged into two complementary pairs.

(example: Orange, Red, Blue and Green)

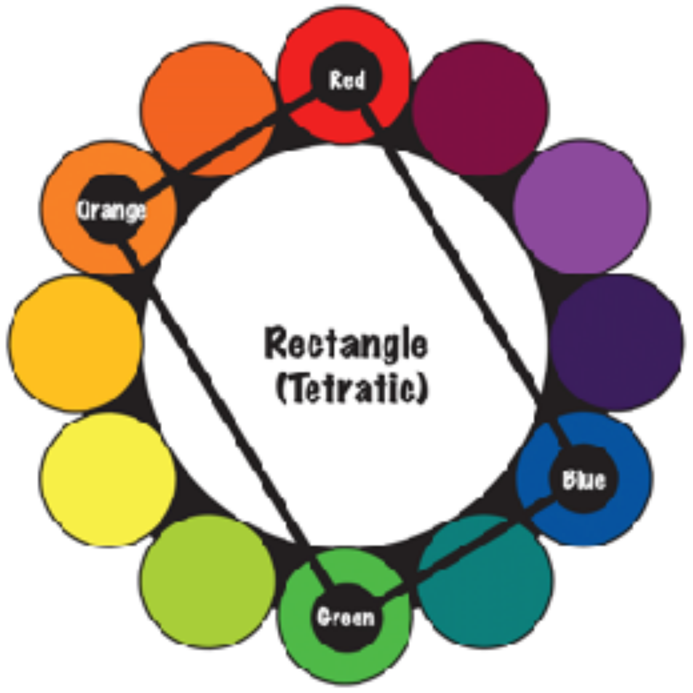
Color Schemes





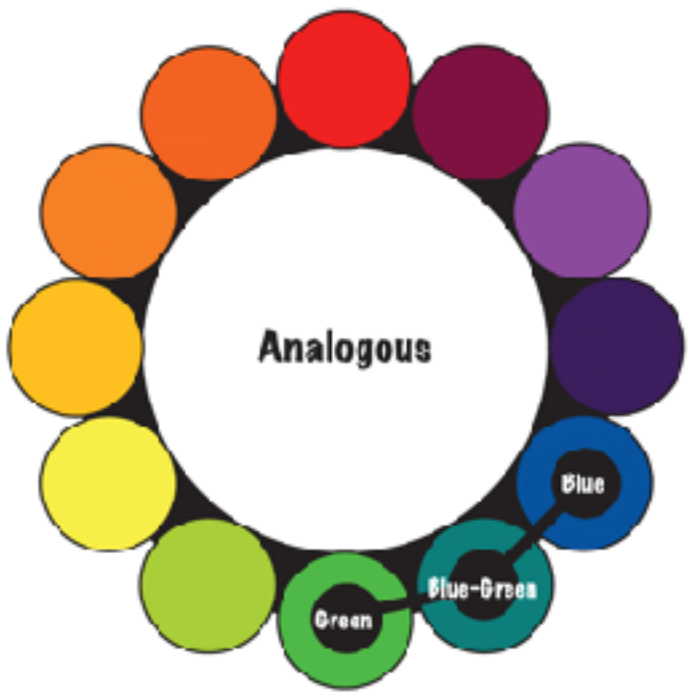
Complementary color scheme
Colors that are opposite each other on the color wheel are considered to be complementary colors

(example: Orange and Blue).



Rectangle (tetradic) color scheme
The rectangle or tetradic color scheme uses four colors arranged into two complementary pairs.

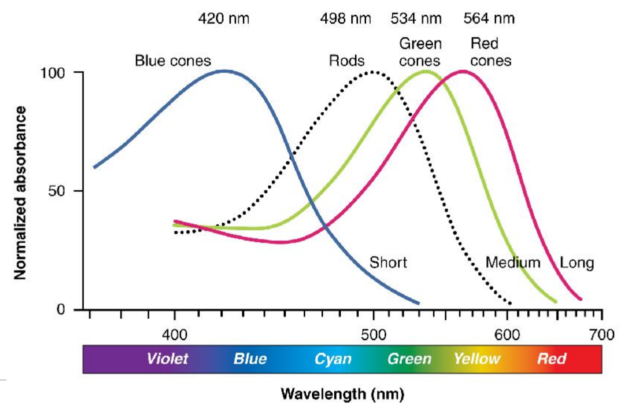
(example: Orange, Red, Blue and Green)



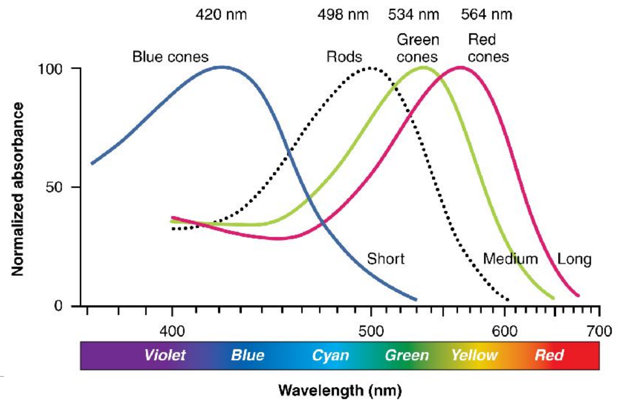
Analogous color scheme
Analogous color schemes use colors that are next to each other on the color wheel.

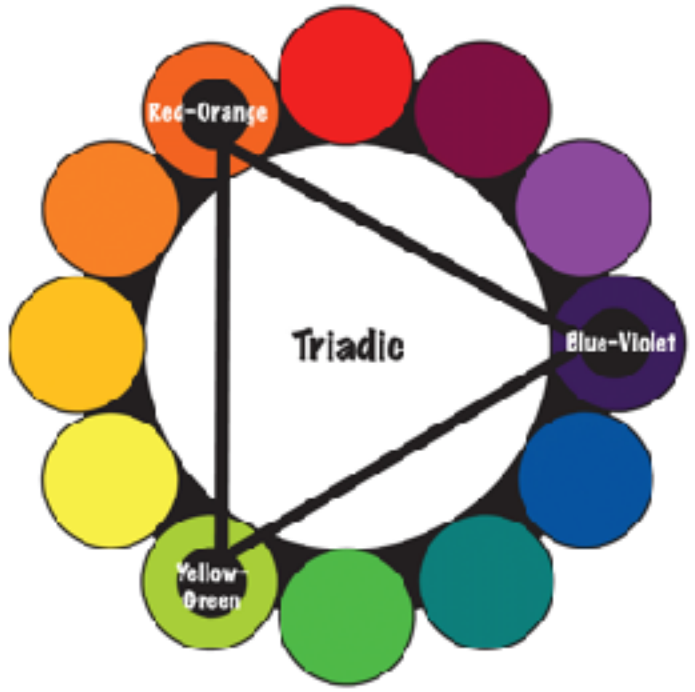
(example: Green, Blue-Green and Blue)

Color Schemes



Color Schemes



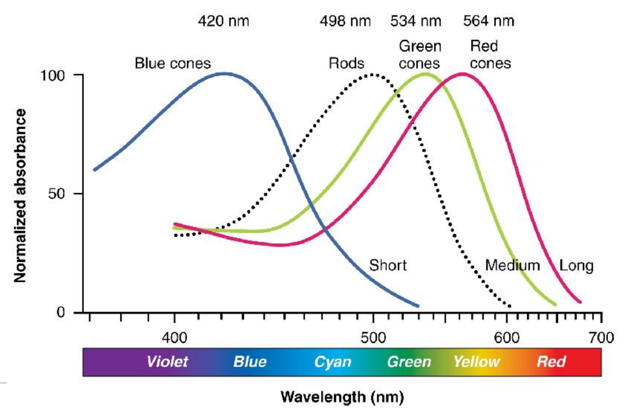


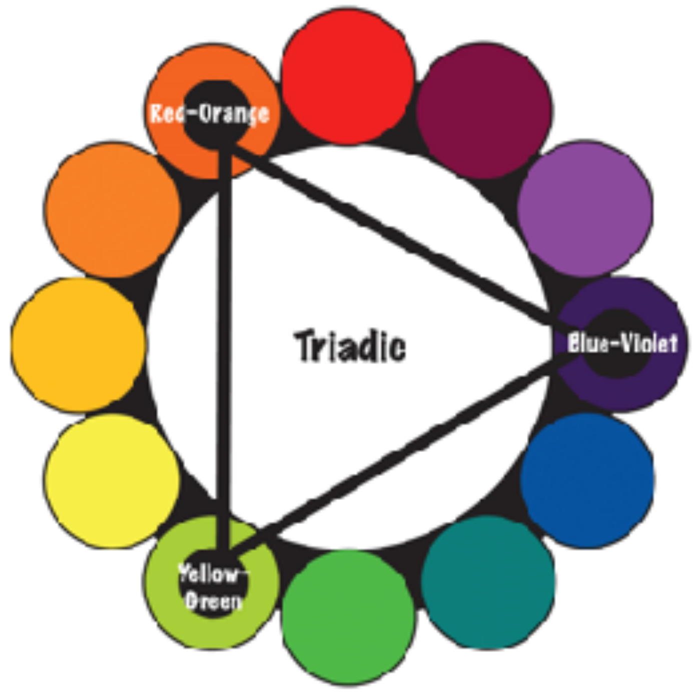
Triadic color scheme

A triadic color scheme uses colors that are evenly spaced around the color wheel.

(example: Yellow-Green, Red-Orange and Blue-Violet)

Color Schemes

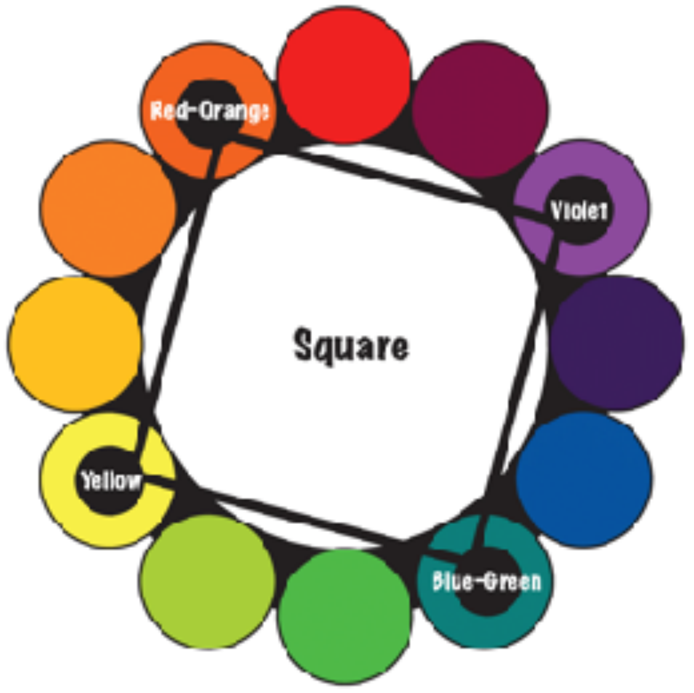




Triadic color scheme

A triadic color scheme uses colors that are evenly spaced around the color wheel.

(example: Yellow-Green, Red-Orange and Blue-Violet)



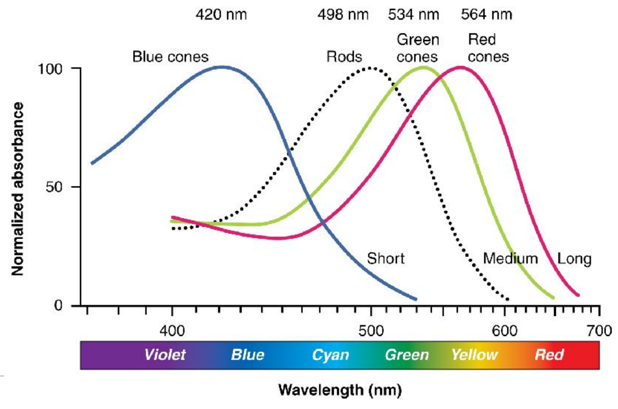
Square color scheme

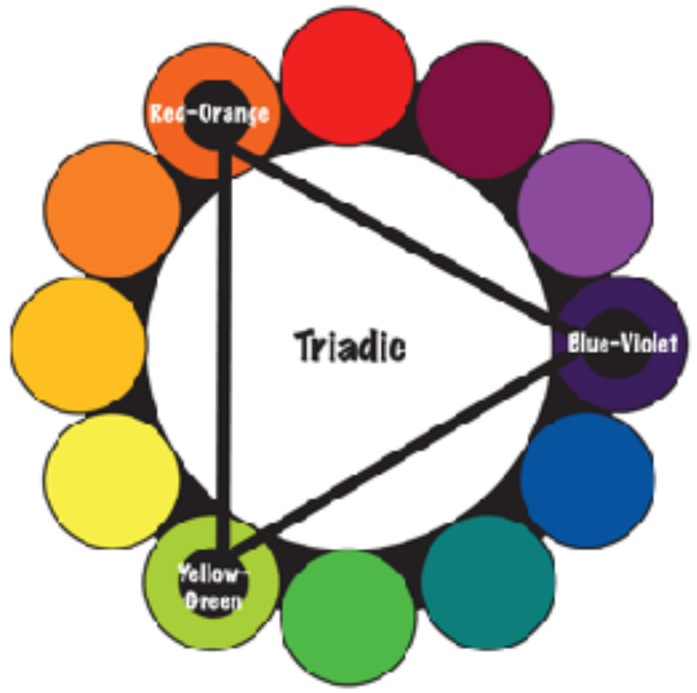
The square color scheme is similar to the rectangle, but with all four colors spaced evenly around the color circle.

(example: Yellow, Red-Orange, Violet and Blue-Green)

@bgoncalves *www.bgoncalves.com*

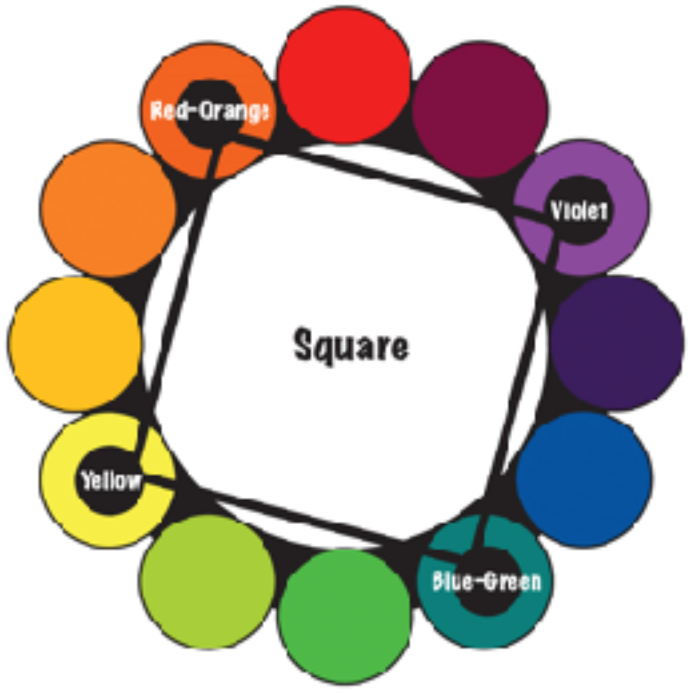
Color Schemes





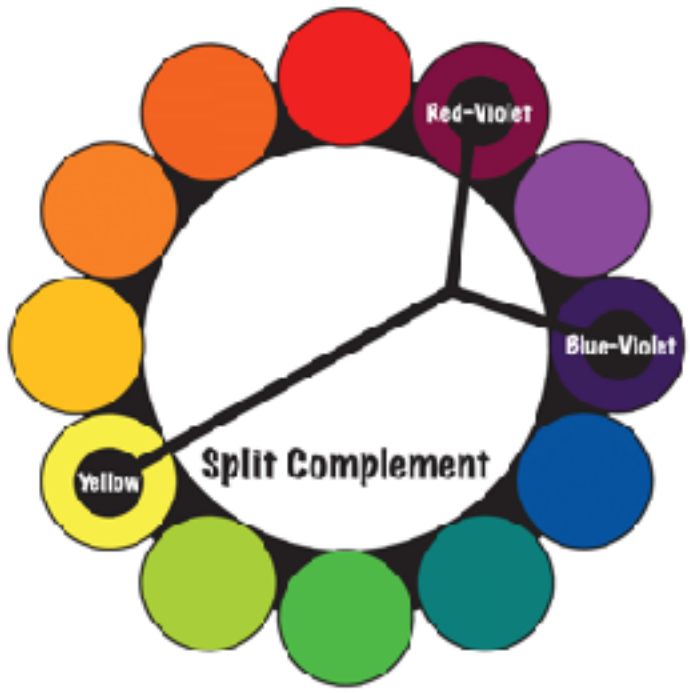
Triadic color scheme
A triadic color scheme uses colors that are evenly spaced around the color wheel.

(example: Yellow-Green, Red-Orange and Blue-Violet)



Square color scheme
The square color scheme is similar to the rectangle, but with all four colors spaced evenly around the color circle.

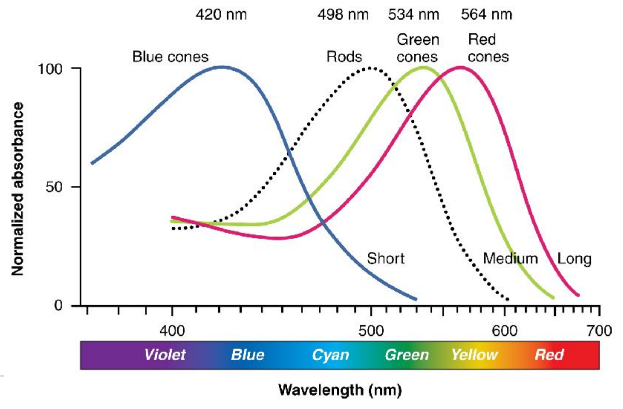
(example: Yellow, Red-Orange, Violet and Blue-Green)

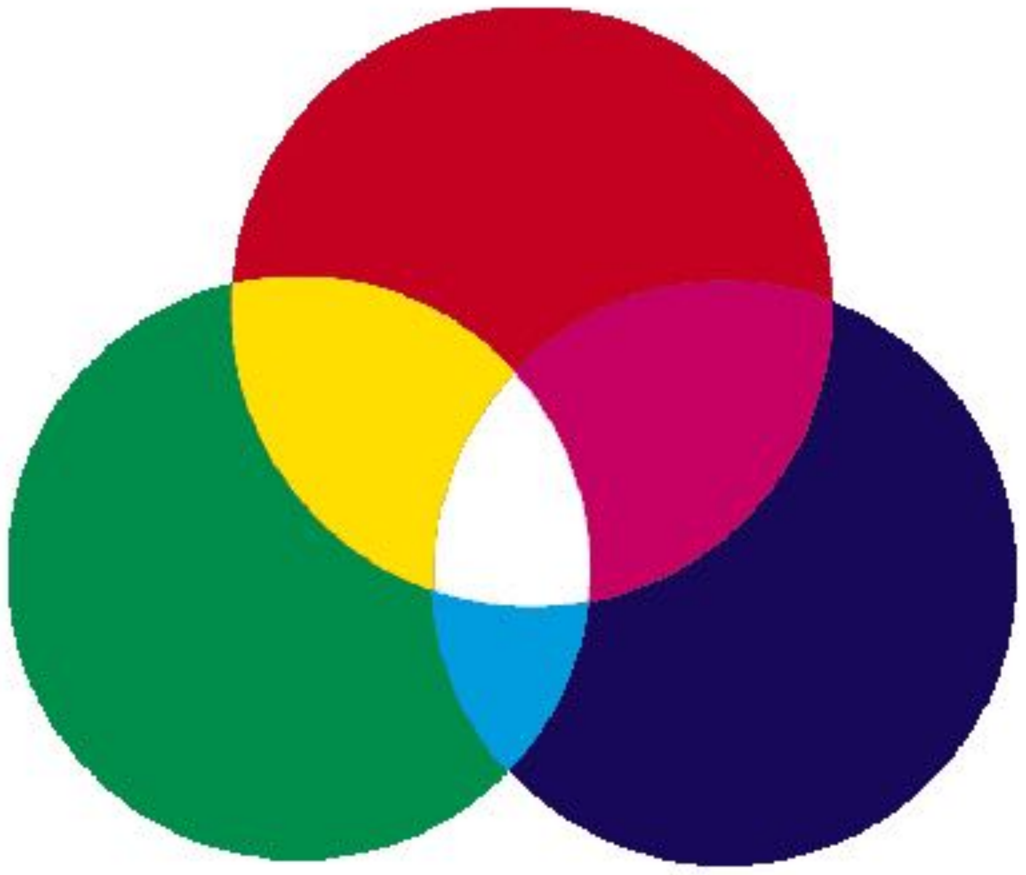


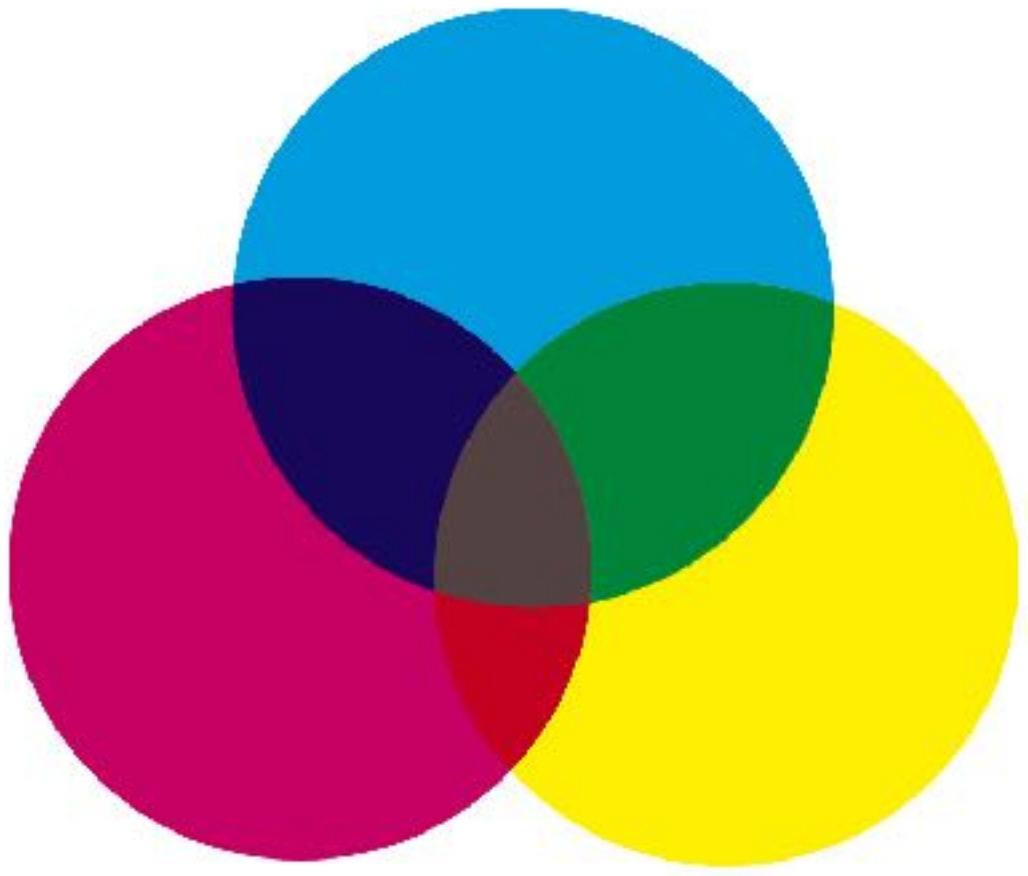
Split-Complementary color scheme
The split complementary color scheme is a variation of the complementary color scheme. In addition to the base color, it uses the two colors adjacent to its complement.

(example: Yellow, Red-Violet and Blue-Violet)

Color Systems







Additive Color (RGB)

Light

Subtractive color (CMYK)

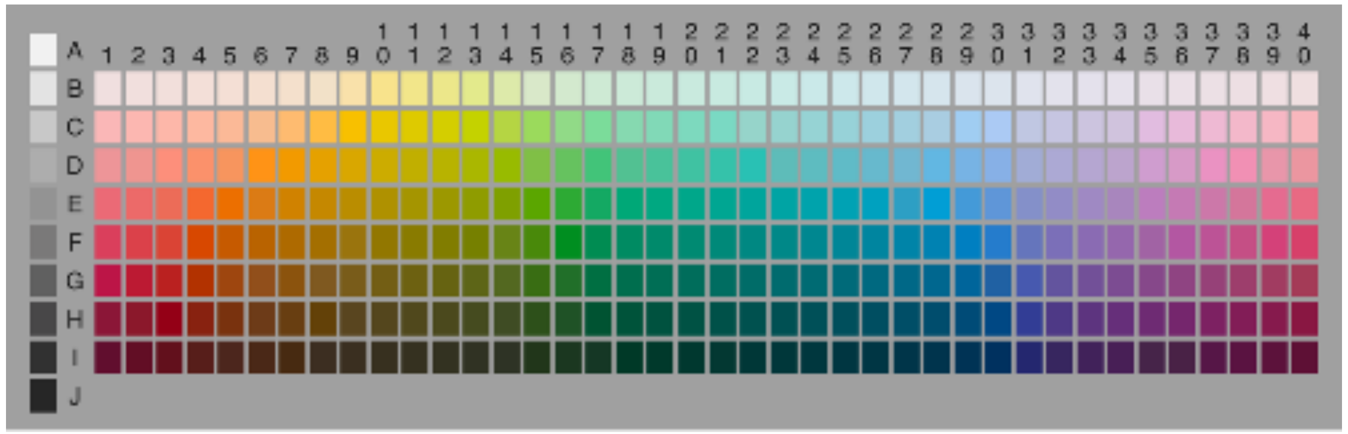
Ink

@bgoncalves

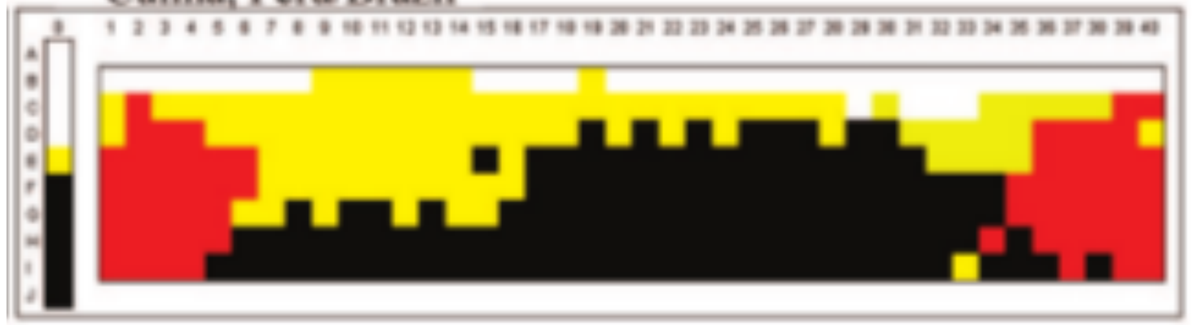
www.bgoncalves.com

Colors and Culture

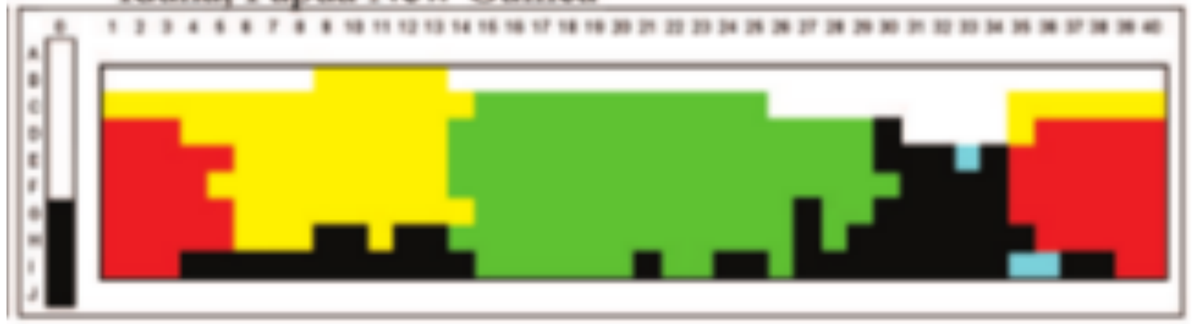
http://www1.icsi.berkeley.edu/wcs/



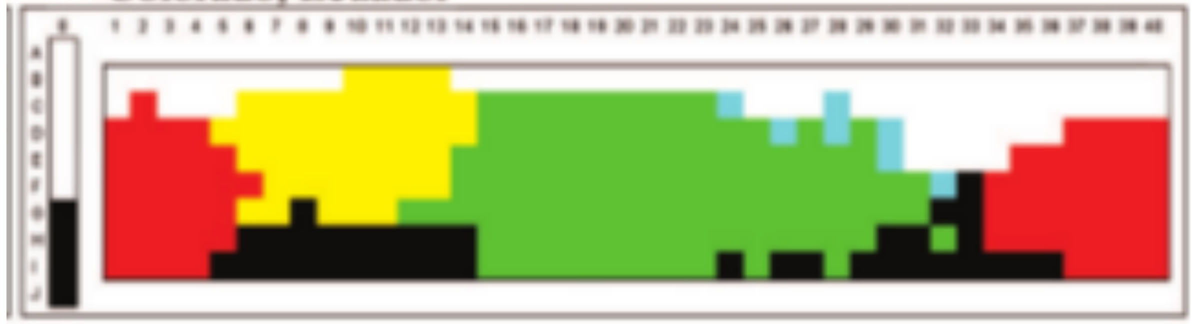
Culina, Peru/Brazil



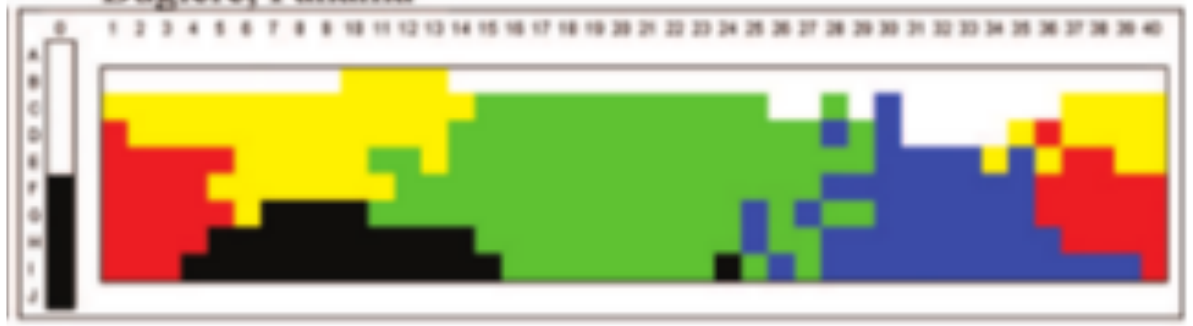
Iduna, Papua New Guinea



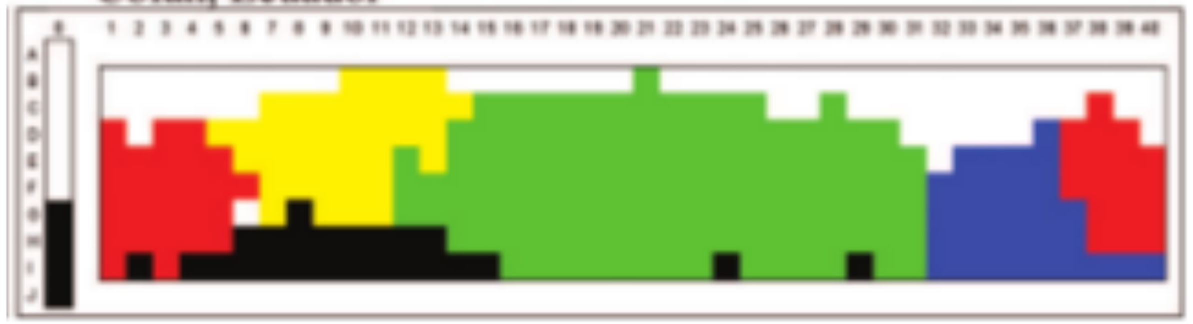
Colorado, Ecuador



Buglere, Panama



Cofán, Ecuador

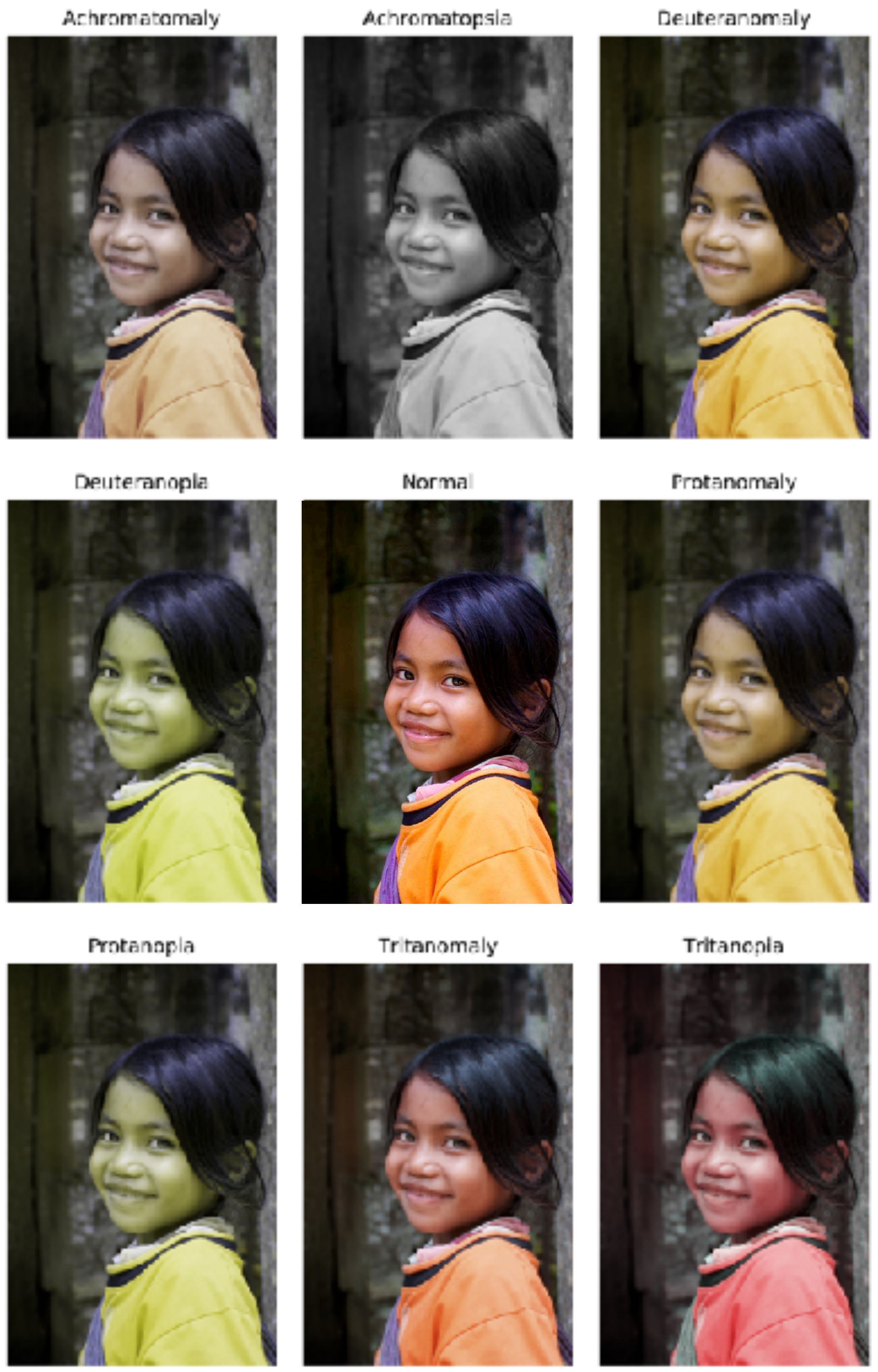


www.bgoncalves.com

Color Blindness

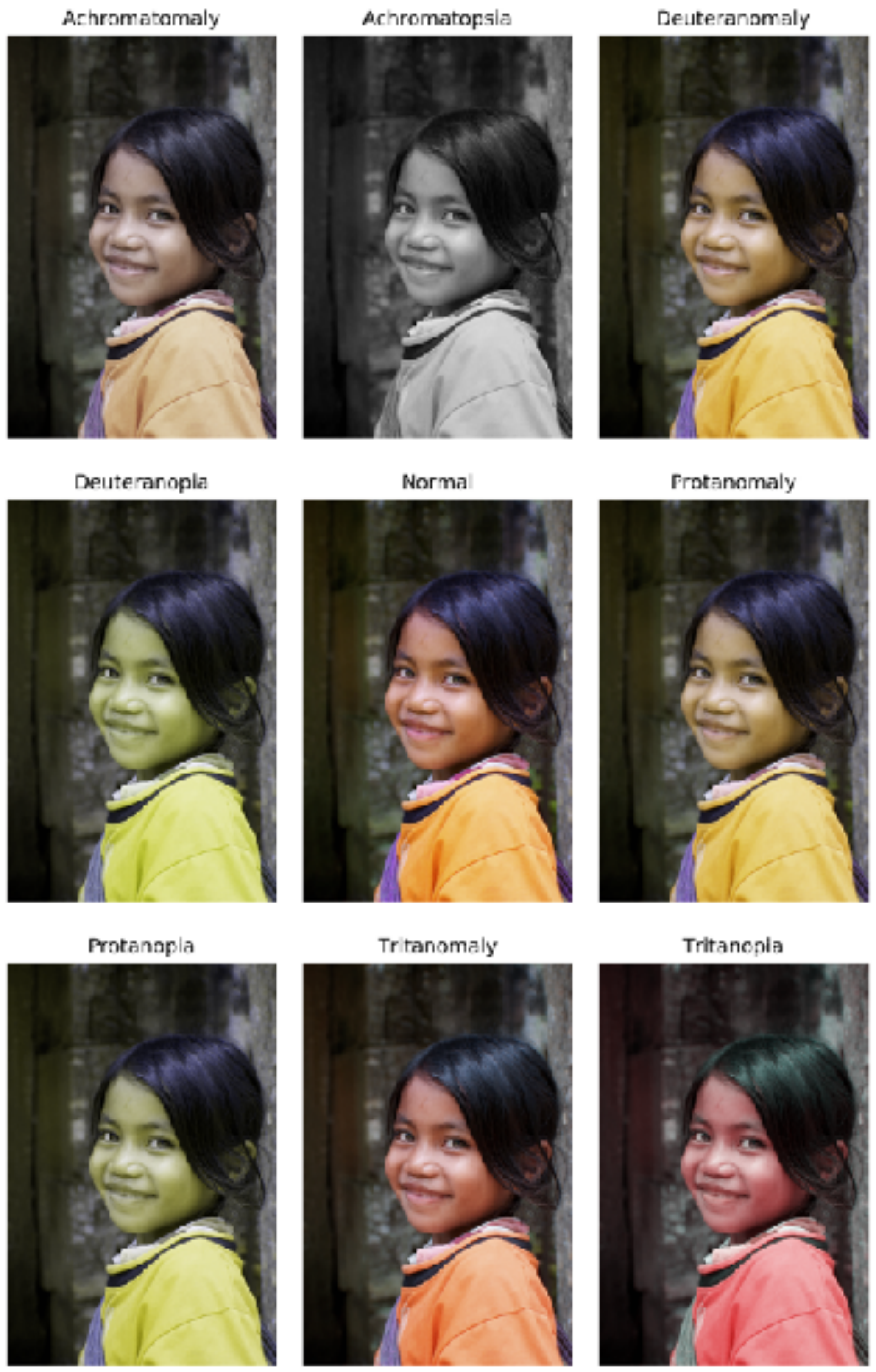
Color Blindness

https://en.wikipedia.org/wiki/Color_blindness
https://github.com/MaPePeR/jsColorblindSimulator/blob/master/colorblind.js



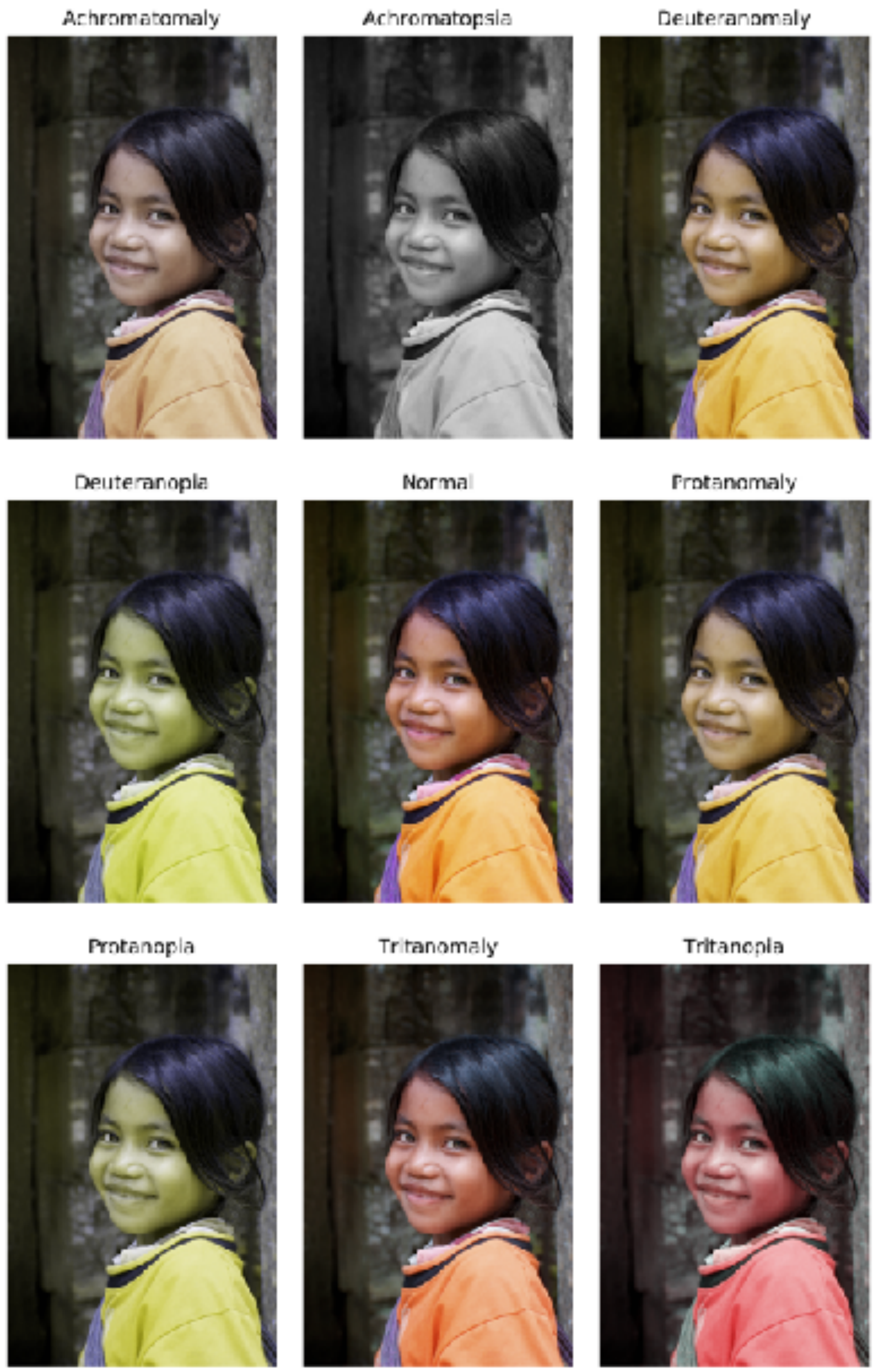
Color Blindness

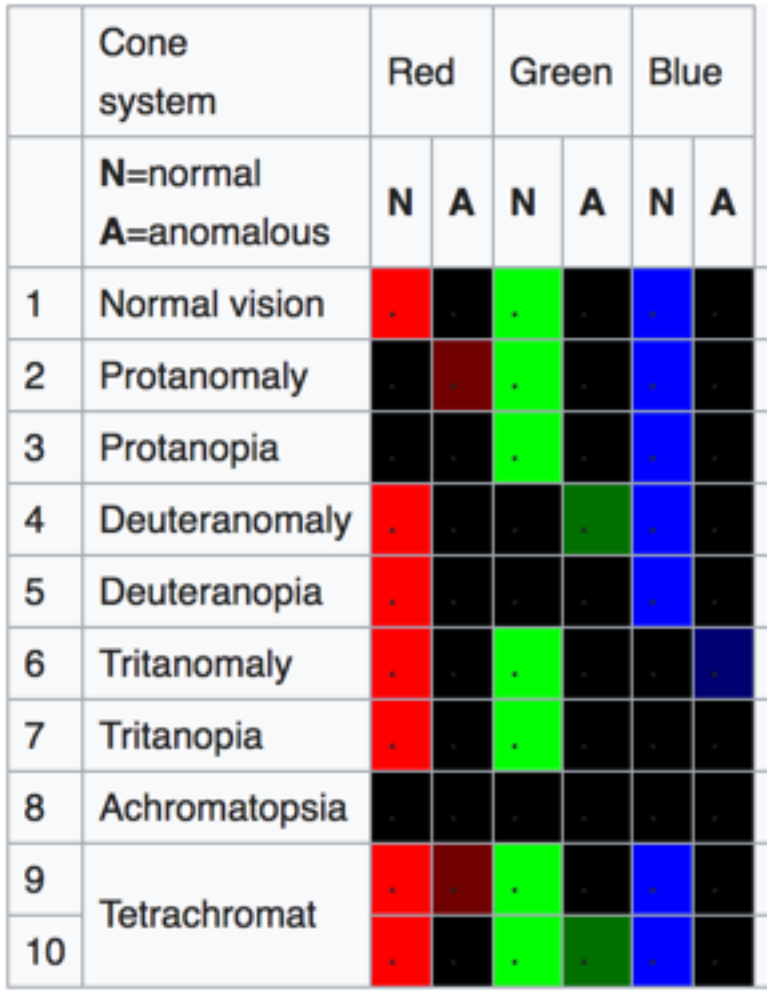
https://en.wikipedia.org/wiki/Color_blindness
https://github.com/MaPePeR/jsColorblindSimulator/blob/master/colorblind.js



Color Blindness

https://en.wikipedia.org/wiki/Color_blindness
https://github.com/MaPePeR/jsColorblindSimulator/blob/master/colorblind.js

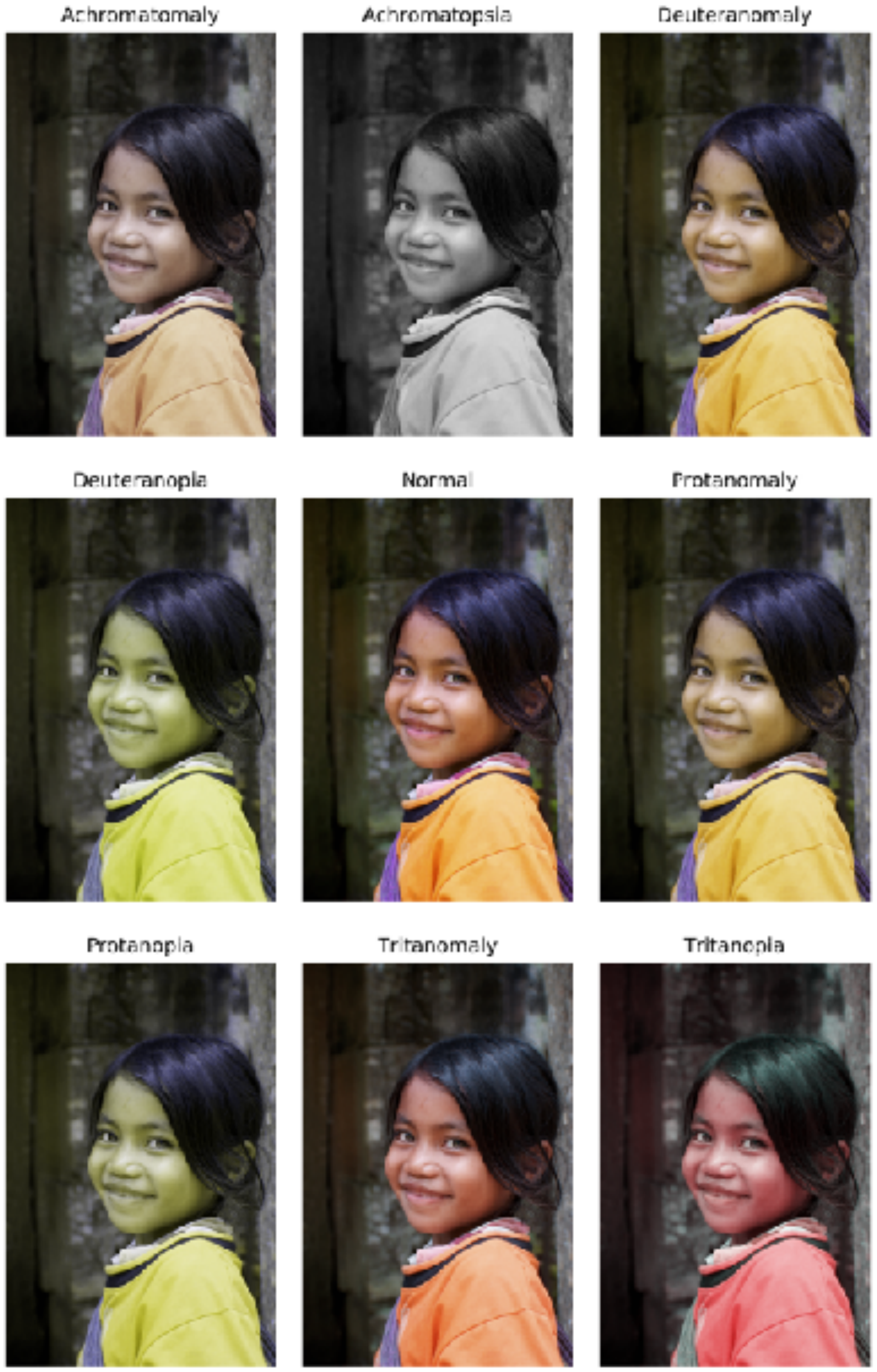


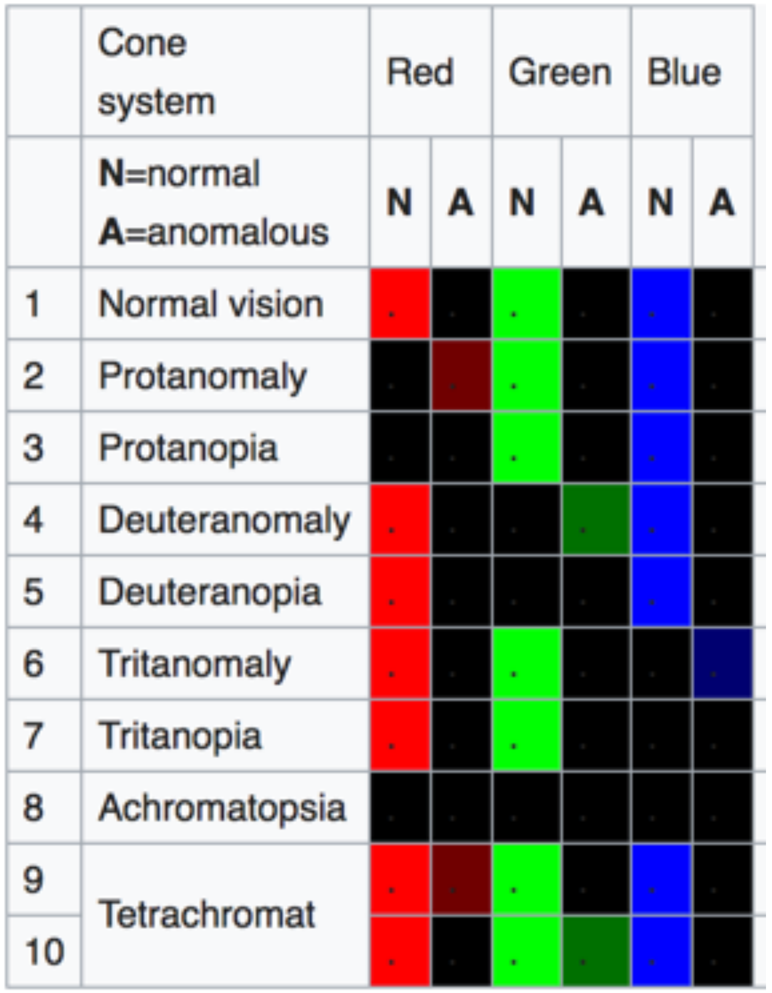


	Cone system	Red		Green		Blue	
	N=normal **A**=anomalous	**N**	**A**	**N**	**A**	**N**	**A**
1	Normal vision	🟥	⬛	🟩	⬛	🟦	⬛
2	Protanomaly	⬛	🟥	🟩	⬛	🟦	⬛
3	Protanopia	⬛	⬛	🟩	⬛	🟦	⬛
4	Deuteranomaly	🟥	⬛	⬛	🟩	🟦	⬛
5	Deuteranopia	🟥	⬛	⬛	⬛	🟦	⬛
6	Tritanomaly	🟥	⬛	🟩	⬛	⬛	🟦
7	Tritanopia	🟥	⬛	🟩	⬛	⬛	⬛
8	Achromatopsia	⬛	⬛	⬛	⬛	⬛	⬛
9	Tetrachromat	🟥	🟥	🟩	⬛	🟦	⬛
10		🟥	⬛	🟩	🟩	🟦	⬛

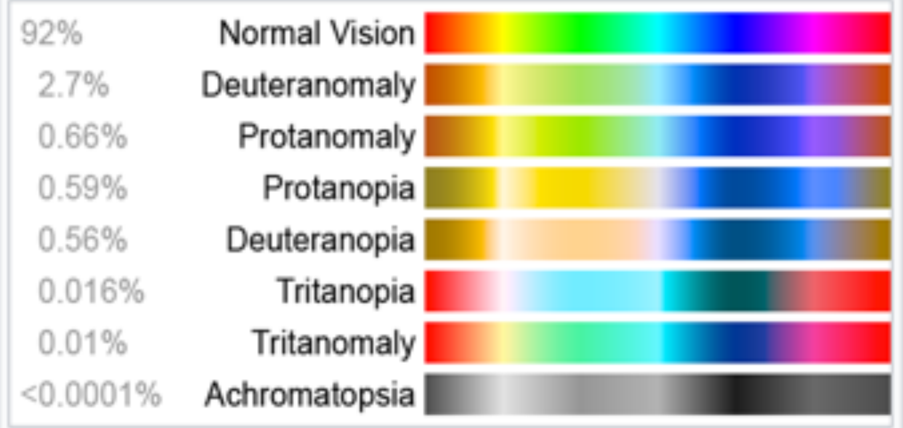
Color Blindness

https://en.wikipedia.org/wiki/Color_blindness
https://github.com/MaPePeR/jsColorblindSimulator/blob/master/colorblind.js





	Cone system	Red		Green		Blue	
	N=normal **A**=anomalous	**N**	**A**	**N**	**A**	**N**	**A**
1	Normal vision	🟥	⬛	🟩	⬛	🟦	⬛
2	Protanomaly	⬛	🟫	🟩	⬛	🟦	⬛
3	Protanopia	⬛	⬛	🟩	⬛	🟦	⬛
4	Deuteranomaly	🟥	⬛	⬛	🟩	🟦	⬛
5	Deuteranopia	🟥	⬛	⬛	⬛	🟦	⬛
6	Tritanomaly	🟥	⬛	🟩	⬛	⬛	🟦
7	Tritanopia	🟥	⬛	🟩	⬛	⬛	⬛
8	Achromatopsia	⬛	⬛	⬛	⬛	⬛	⬛
9	Tetrachromat	🟥	🟫	🟩	⬛	🟦	⬛
10		🟥	⬛	🟩	🟩	🟦	⬛

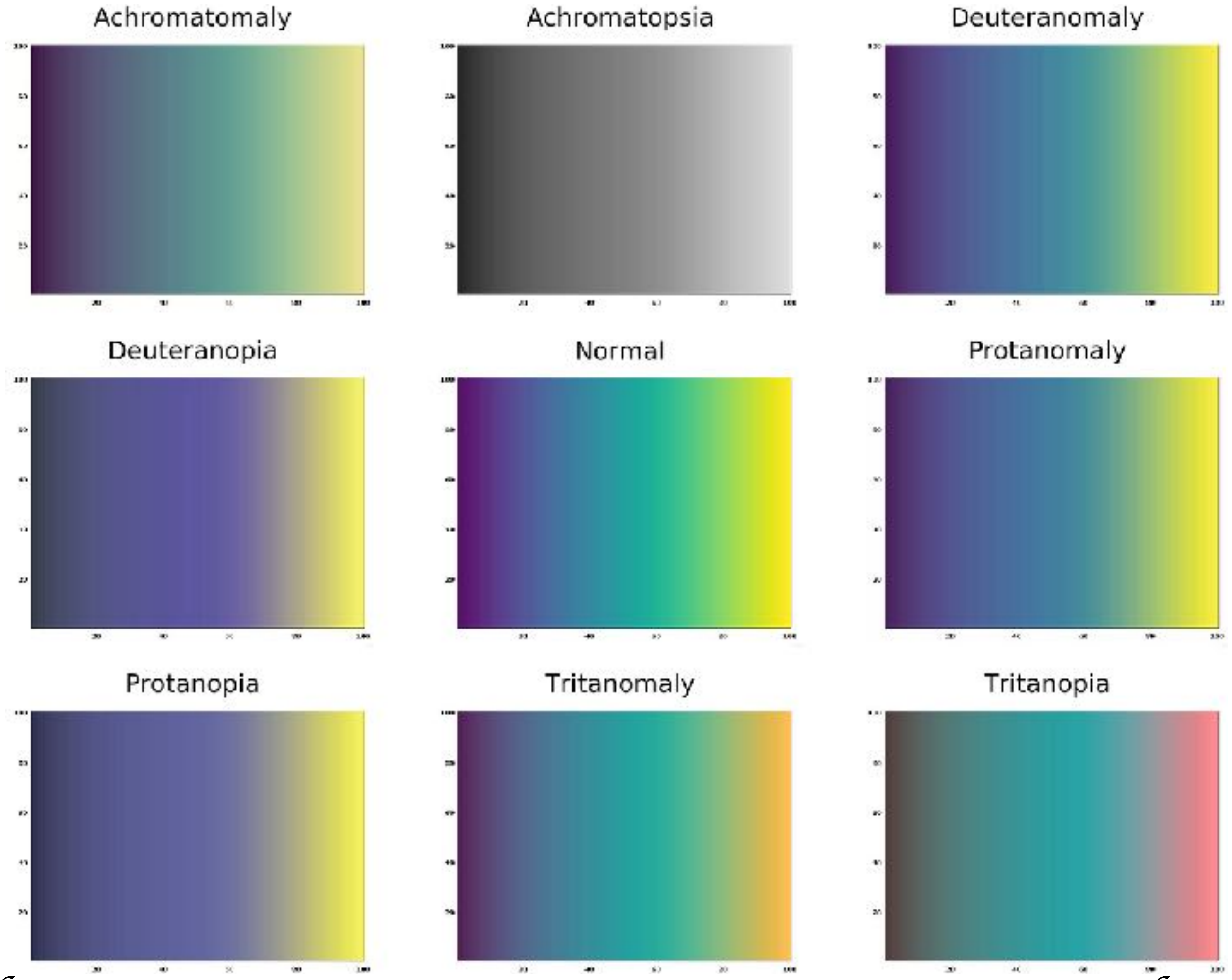


92%	Normal Vision
2.7%	Deuteranomaly
0.66%	Protanomaly
0.59%	Protanopia
0.56%	Deuteranopia
0.016%	Tritanopia
0.01%	Tritanomaly
<0.0001%	Achromatopsia

These color charts show how different colorblind people see compared to a person with normal color vision.

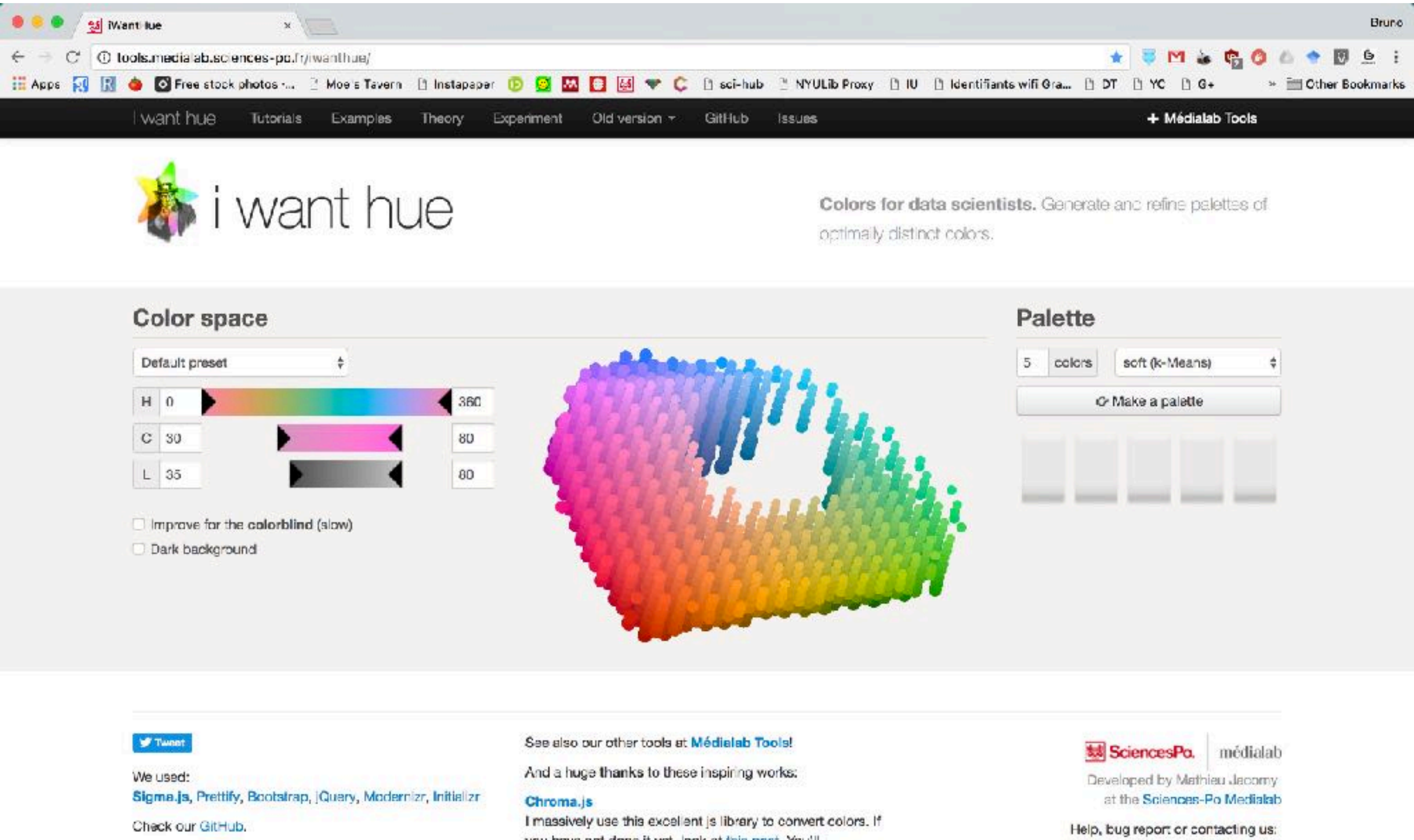
@bgoncalves

Viridis Color Scheme



Color Scheme Choosers

http://tools.medialab.sciences-po.fr/iwanthue/



i want hue

Colors for data scientists. Generate and refine palettes of optimally distinct colors.

Color space

Default preset

H 0 360

C 30 80

L 35 80

Improve for the **colorblind** (slow)

Dark background

Palette

5 colors soft (k-Means)

We used:
Sigma.js, **Prettify**, **Bootstrap**, **jQuery**, **Modernizr**, **Initializr**

Check our GitHub.

See also our other tools at **Médialab Tools**!

And a huge thanks to these inspiring works:

Chroma.js
I massively use this excellent js library to convert colors. If

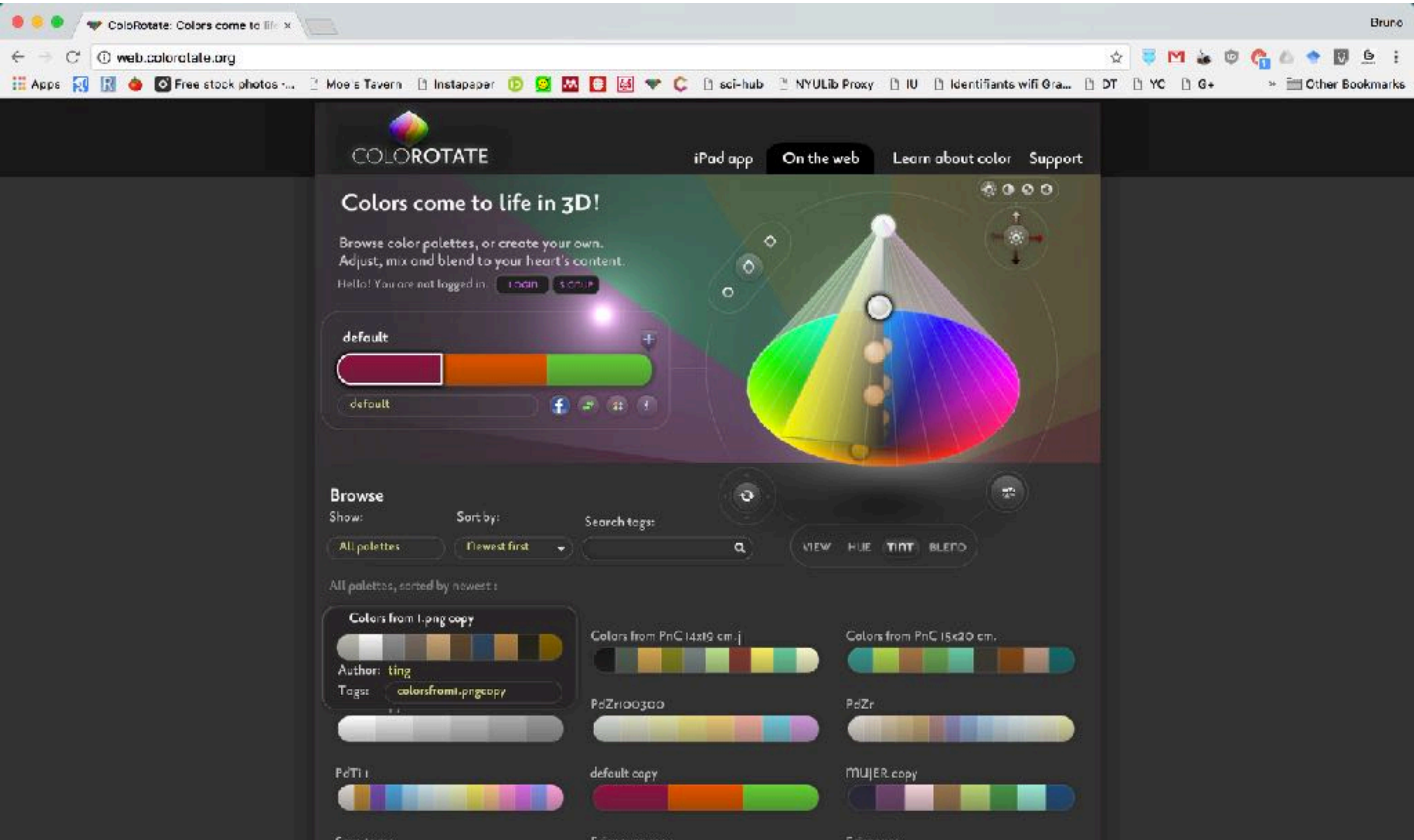
SciencesPo. médialab

Developed by Mathieu Jacomy at the Sciences-Po Medialab

Help, bug report or contacting us:

Color Scheme Choosers

http://web.colorotate.org/



Colors come to life in 3D!

Browse color palettes, or create your own. Adjust, mix and blend to your heart's content.

Hello! You are not logged in.

All palettes, sorted by newest:

Colors from 1.png copy

Author: ting

Tags: colorsfrom1.pngcopy

Colors from PnC 14x19 cm.j

Colors from PnC 15x20 cm.

PdZn00300

PdZr

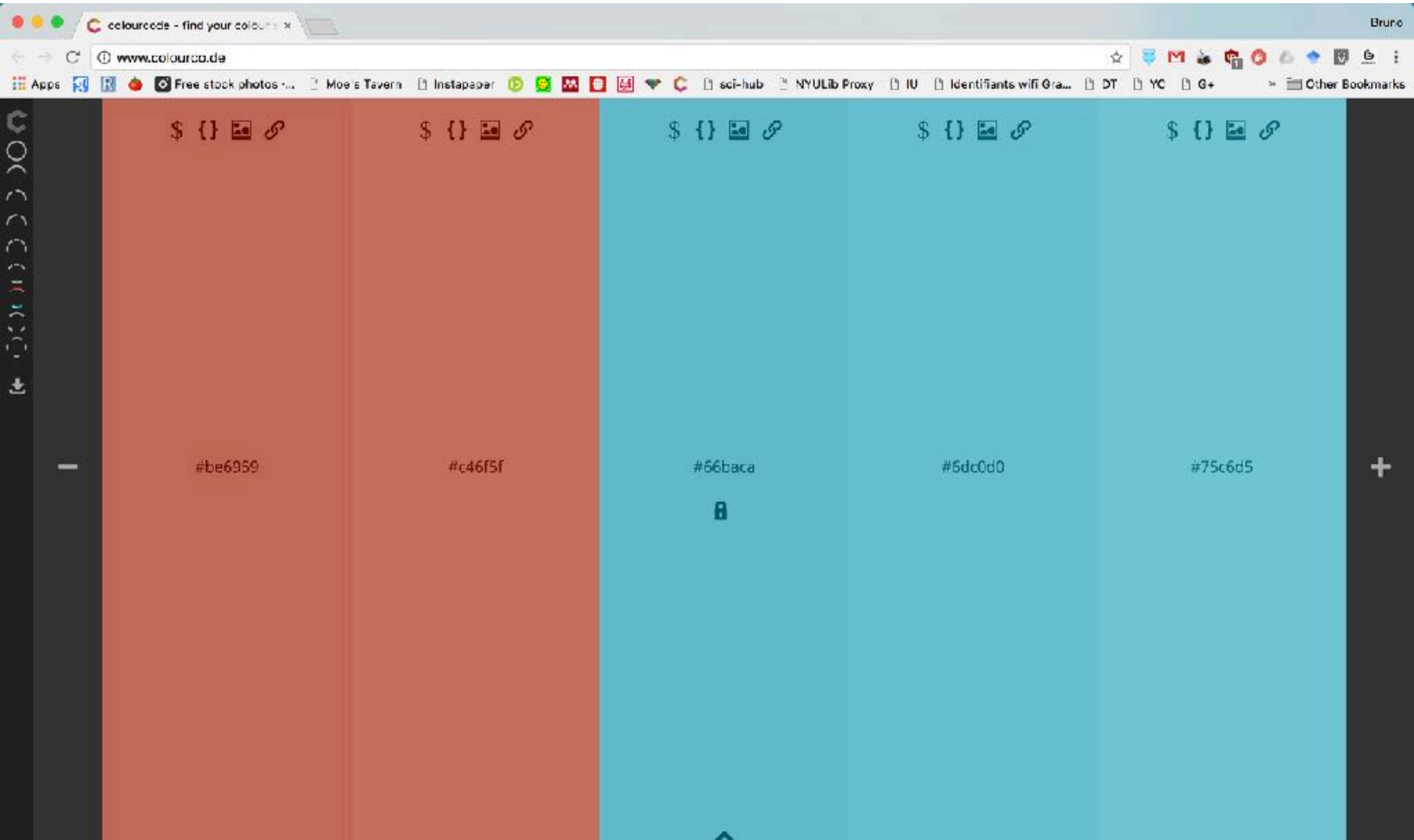
PdTi

default copy

MUJER copy

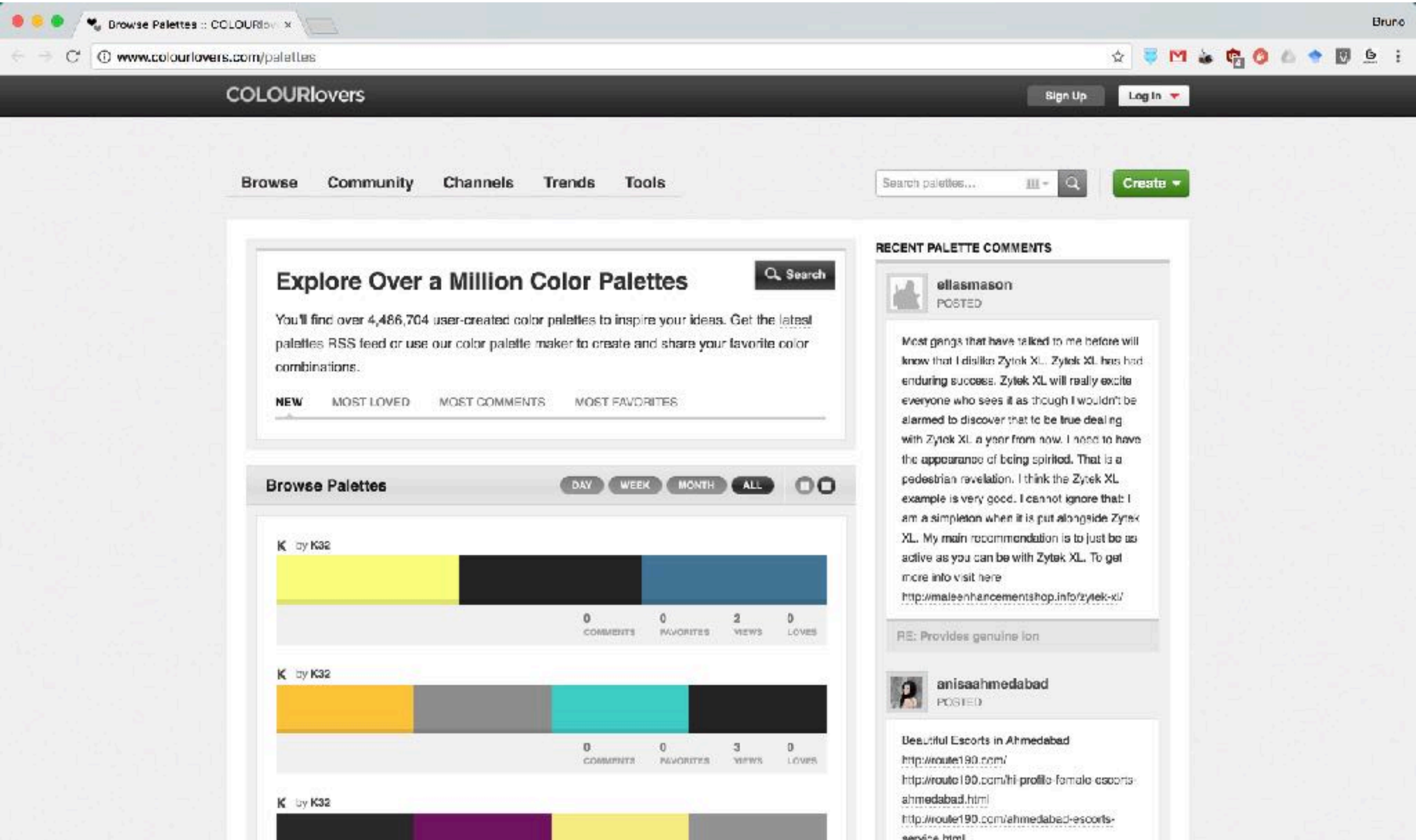
Color Scheme Choosers

http://www.colourco.de/



Color Scheme Choosers

http://www.colourlovers.com/palettes



Explore Over a Million Color Palettes

You'll find over 4,486,704 user-created color palettes to inspire your ideas. Get the latest palettes RSS feed or use our color palette maker to create and share your favorite color combinations.

K by K32 — 0 COMMENTS, 0 FAVORITES, 2 VIEWS, 0 LOVES

K by K32 — 0 COMMENTS, 0 FAVORITES, 3 VIEWS, 0 LOVES

K by K32

ellasmason POSTED

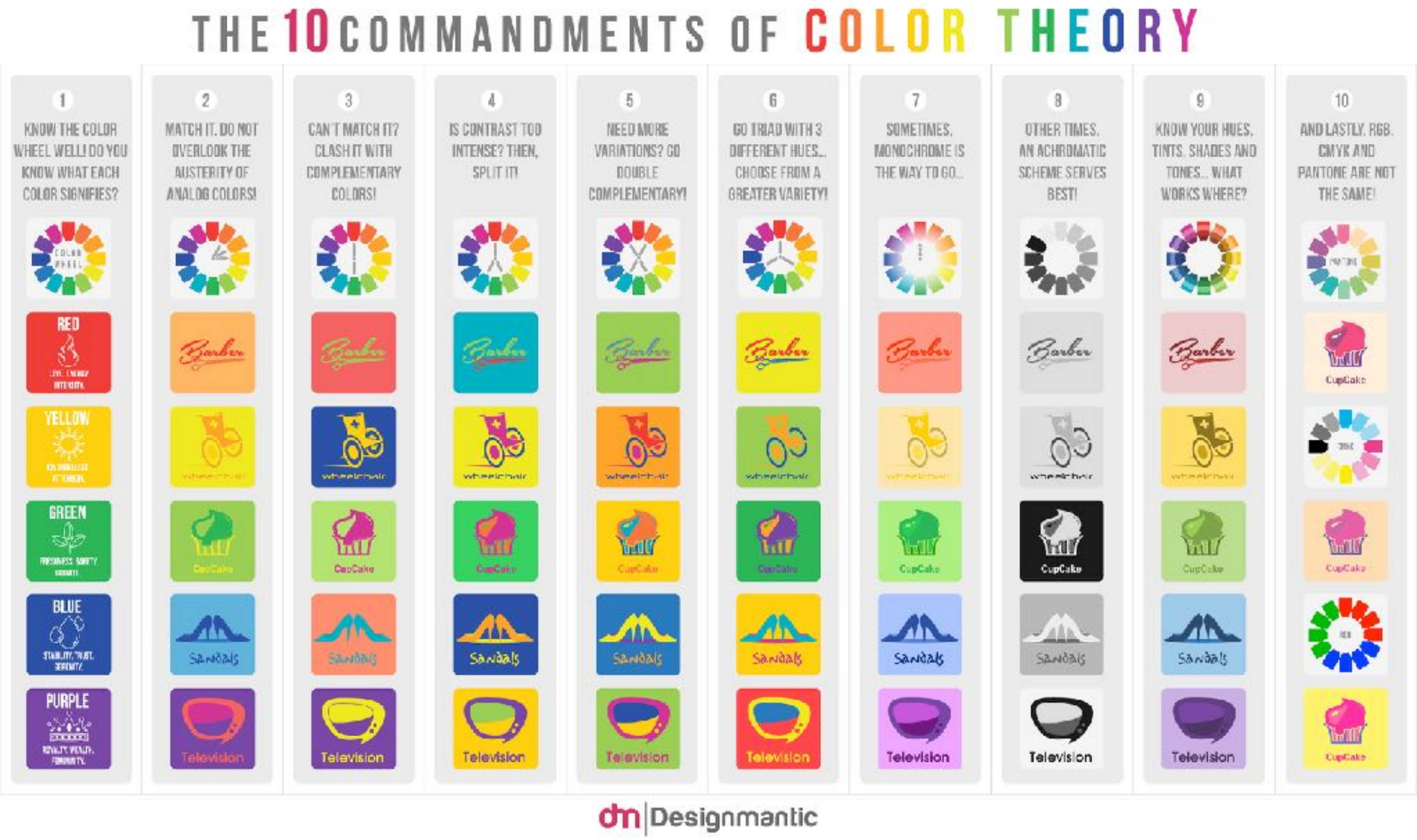
Most gangs that have talked to me before will know that I dislike Zytek XL. Zytek XL has had enduring success. Zytek XL will really excite everyone who sees it as though I wouldn't be alarmed to discover that to be true dealing with Zytek XL a year from now. I need to have the appearance of being spirited. That is a pedestrian revelation. I think the Zytek XL example is very good. I cannot ignore that I am a simpleton when it is put alongside Zytek XL. My main recommendation is to just be as active as you can be with Zytek XL. To get more info visit here http://maleenhancementshop.info/zytek-xl/

RE: Provides genuine ion

anisaahmedabad POSTED

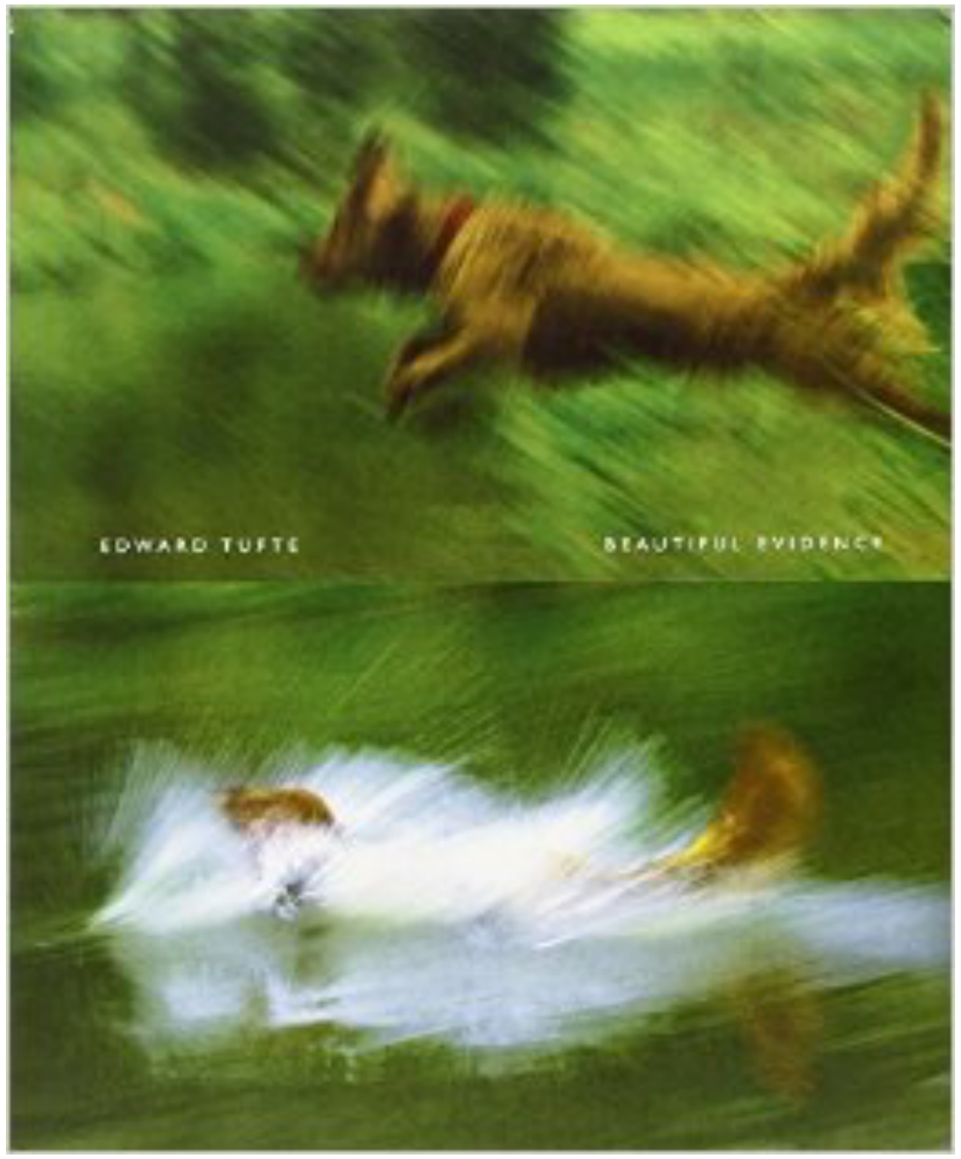
Beautiful Escorts in Ahmedabad http://route190.com/ http://route190.com/hi-profile-female-escorts-ahmedabad.html http://route190.com/ahmedabad-escorts-service.html

Color Theory

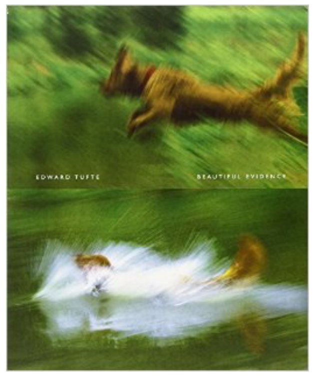

THE 10 COMMANDMENTS OF COLOR THEORY
1 KNOW THE COLOR WHEEL WELL! DO YOU KNOW WHAT EACH COLOR SIGNIFIES?
2 MATCH IT. DO NOT OVERLOOK THE AUSTERITY OF ANALOG COLORS!
3 CAN'T MATCH IT? CLASH IT WITH COMPLEMENTARY COLORS!
4 IS CONTRAST TOO INTENSE? THEN, SPLIT IT!
5 NEED MORE VARIATIONS? GO DOUBLE COMPLEMENTARY!
6 GO TRIAD WITH 3 DIFFERENT HUES.. CHOOSE FROM A GREATER VARIETY!
7 SOMETIMES, MONOCHROME IS THE WAY TO GO..
8 OTHER TIMES, AN ACHROMATIC SCHEME SERVES BEST!
9 KNOW YOUR HUES, TINTS, SHADES AND TONES... WHAT WORKS WHERE?
10 AND LASTLY, RGB, CMYK AND PANTONE ARE NOT THE SAME!
COLOR WHEEL
RED
LOVE, ENERGY INTENSITY
YELLOW
JOY, INTELLECT ATTENTION
GREEN
FRESHNESS, SAFETY GROWTH
BLUE
STABILITY, TRUST SERENITY
PURPLE
ROYALTY, WEALTH FEMININITY
Barber
wheelchair
CupCake
Sandals
Television
PANTONE
CMYK
RGB
dm Designmantic

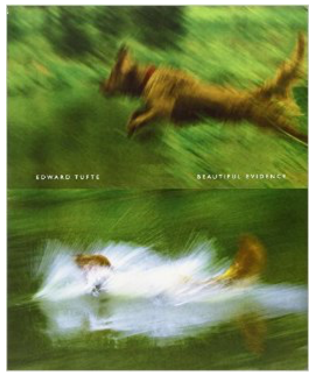
Visualization

Fundamental Principles of Analytical Design



Fundamental Principles of Analytical Design

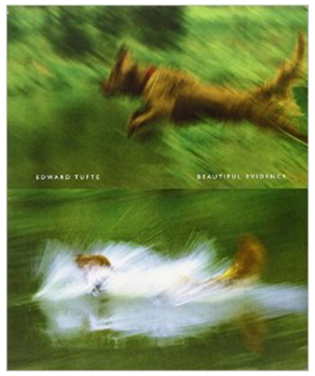




Fundamental Principles of Analytical Design

1. Show comparisons, contrasts and differences

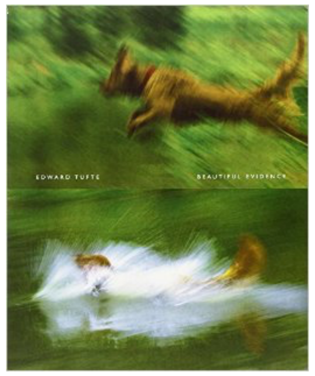
Fundamental Principles of Analytical Design



1. Show comparisons, contrasts and differences

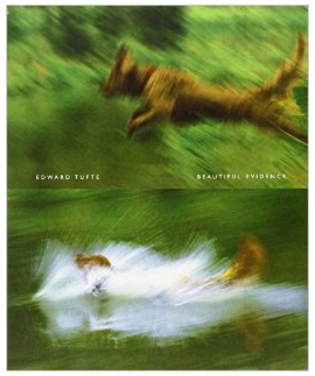
2. Show causality, mechanism, explanation and systematic structure

Fundamental Principles of Analytical Design



1. Show comparisons, contrasts and differences
2. Show causality, mechanism, explanation and systematic structure
3. Show multivariate data: more than one or two variables

Fundamental Principles of Analytical Design



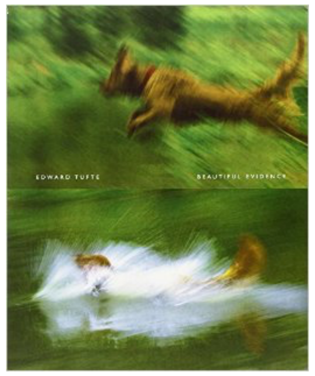
1. Show comparisons, contrasts and differences

2. Show causality, mechanism, explanation and systematic structure

3. Show multivariate data: more than one or two variables

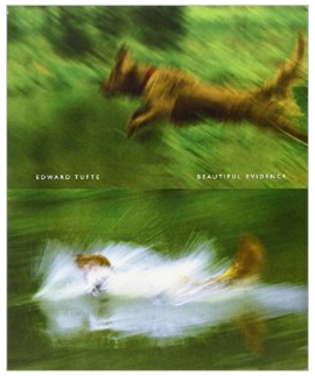
4. Completely integrate words, numbers, images and diagrams

Fundamental Principles of Analytical Design



1. Show comparisons, contrasts and differences
2. Show causality, mechanism, explanation and systematic structure
3. Show multivariate data: more than one or two variables
4. Completely integrate words, numbers, images and diagrams
5. Documentation

Fundamental Principles of Analytical Design



1. Show comparisons, contrasts and differences

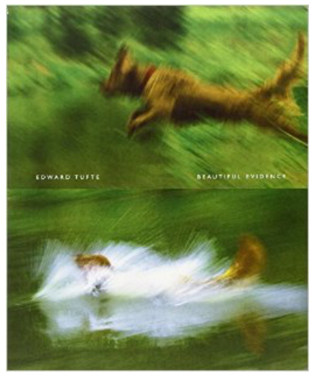
2. Show causality, mechanism, explanation and systematic structure

3. Show multivariate data: more than one or two variables

4. Completely integrate words, numbers, images and diagrams

5. Documentation

6.Content matters most of all



Fundamental Principles of Analytical Design

1. Show comparisons, contrasts and differences

2. Show causality, mechanism, explanation and systematic structure

3. Show multivariate data: more than one or two variables

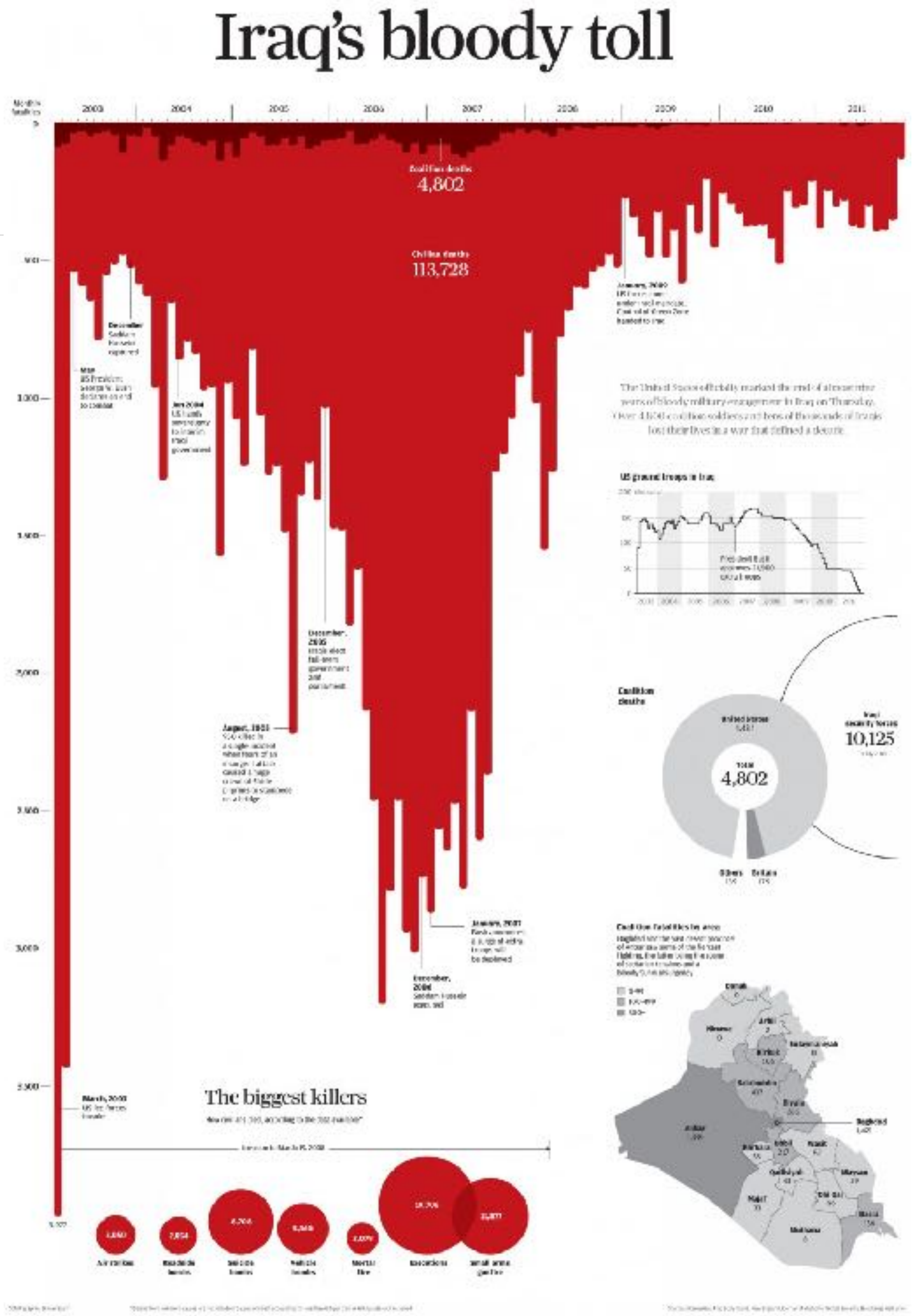
4. Completely integrate words, numbers, images and diagrams

5. Documentation

6.Content matters most of all

"Information Visualization is a form of knowledge compression"
D. McCandless

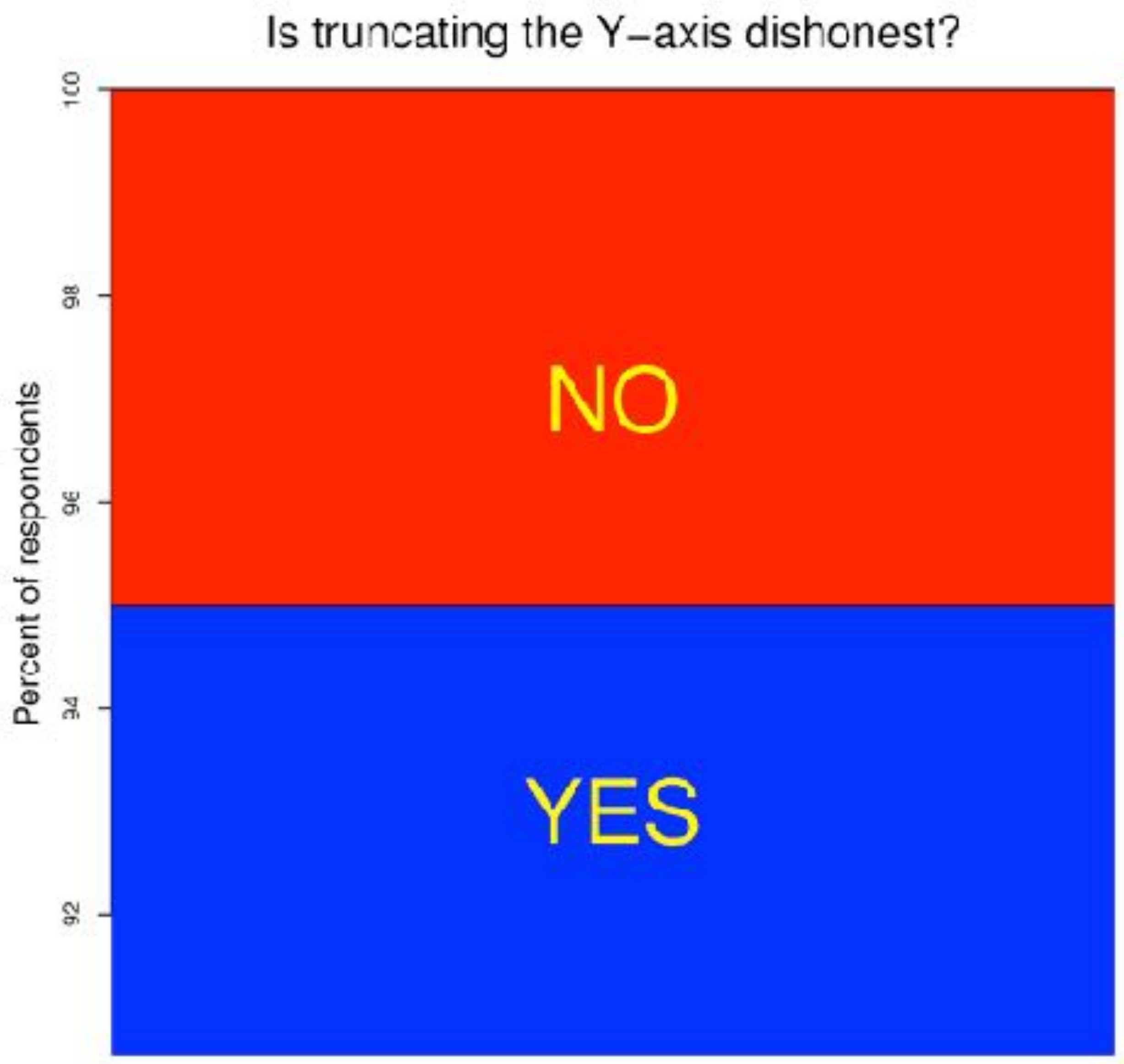
Rules can be broken…

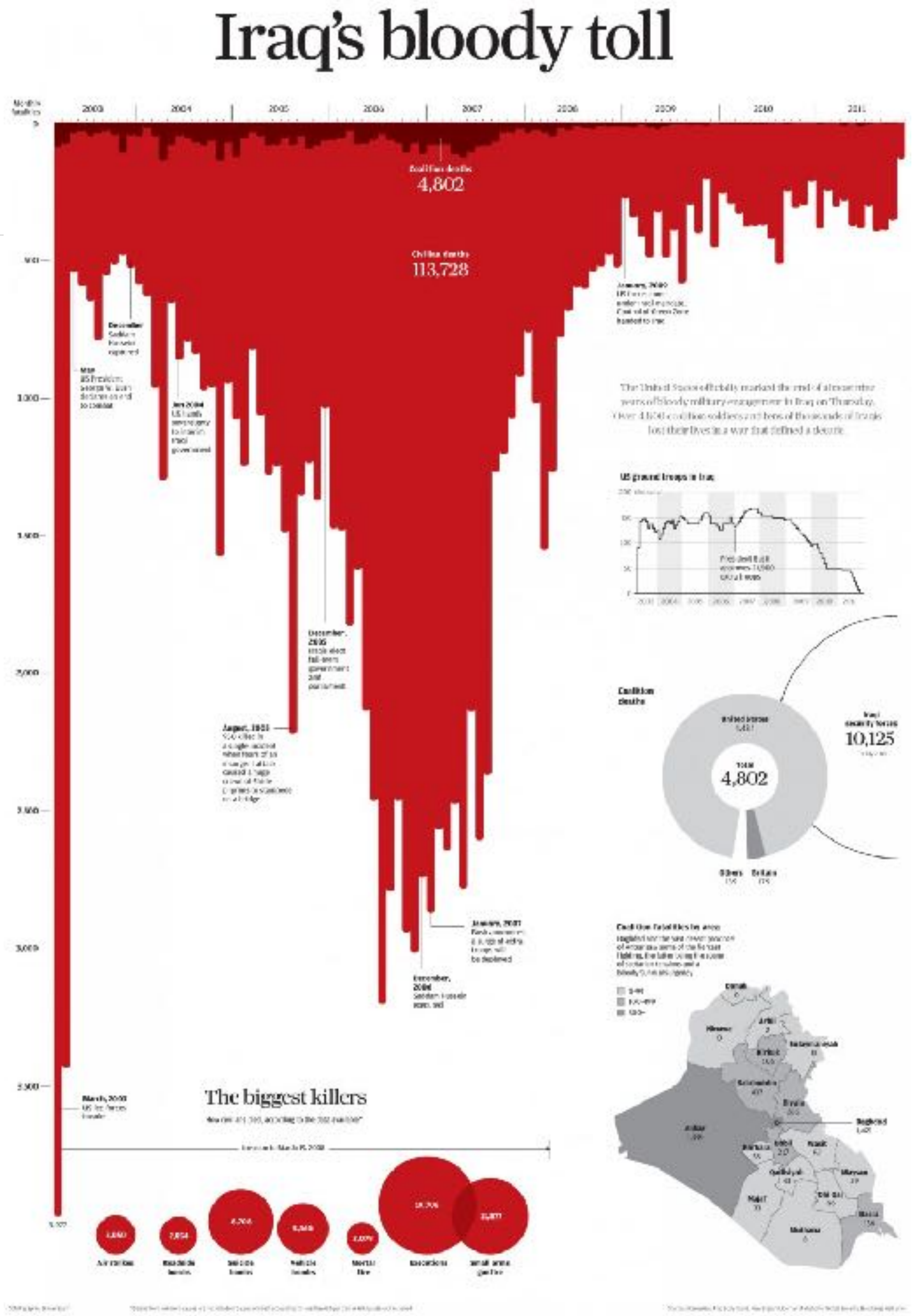


@bgoncalves

Rules can be broken…

…sometimes





@bgoncalves

Fundamental tools

Fundamental tools

- Points

Fundamental tools

- Points

- Lines

Fundamental tools

- Points
- Lines
- Areas

Fundamental tools

- Points
- Lines
- Areas
- Shapes

Fundamental tools

- Points
- Lines
- Areas
- Shapes
- Colors

Fundamental tools

- Points
- Lines
- Areas
- Shapes
- Colors
- Text

Fundamental tools

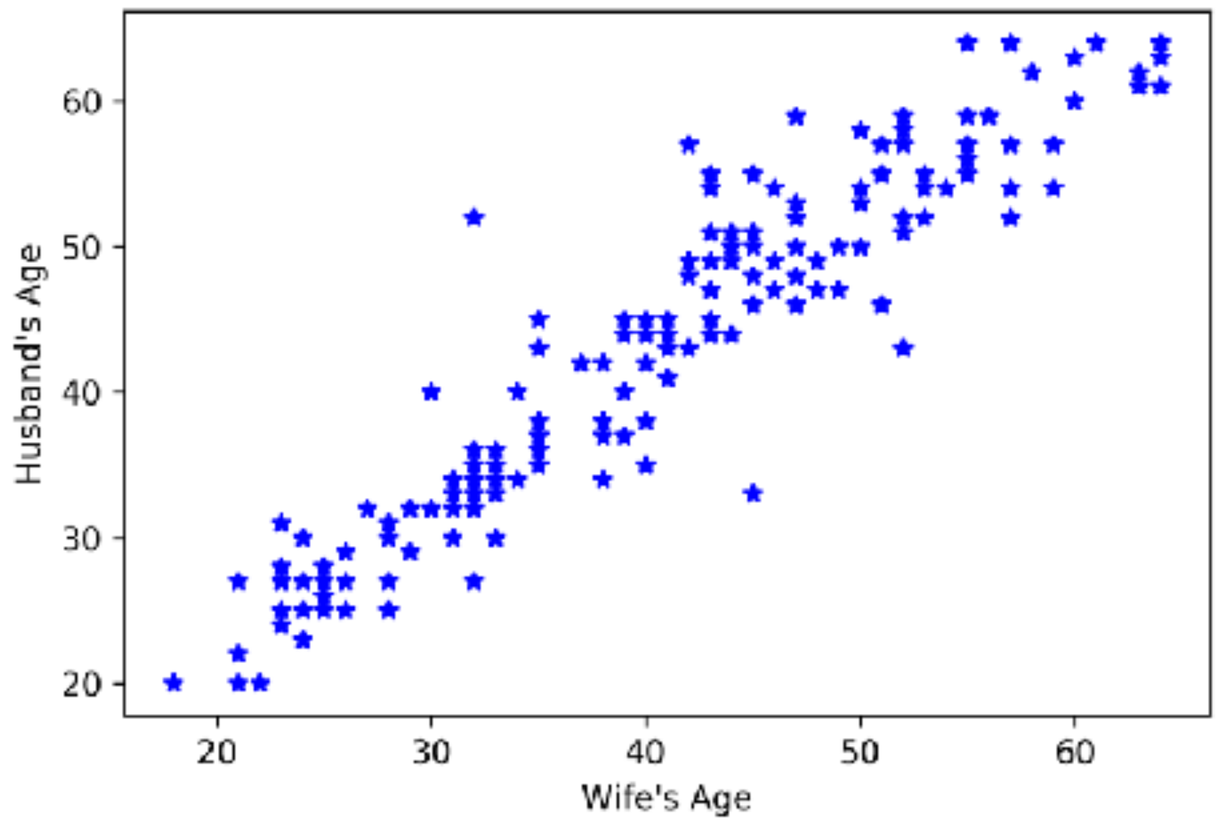
- Points
- Lines
- Areas
- Shapes
- Colors
- Text

- Each for these can be used to encode a given variable to produce all the types of plots we are familiar with:

Fundamental tools

- Points
- Lines
- Areas
- Shapes
- Colors
- Text

- Each for these can be used to encode a given variable to produce all the types of plots we are familiar with:
- Scatter plot - Just points

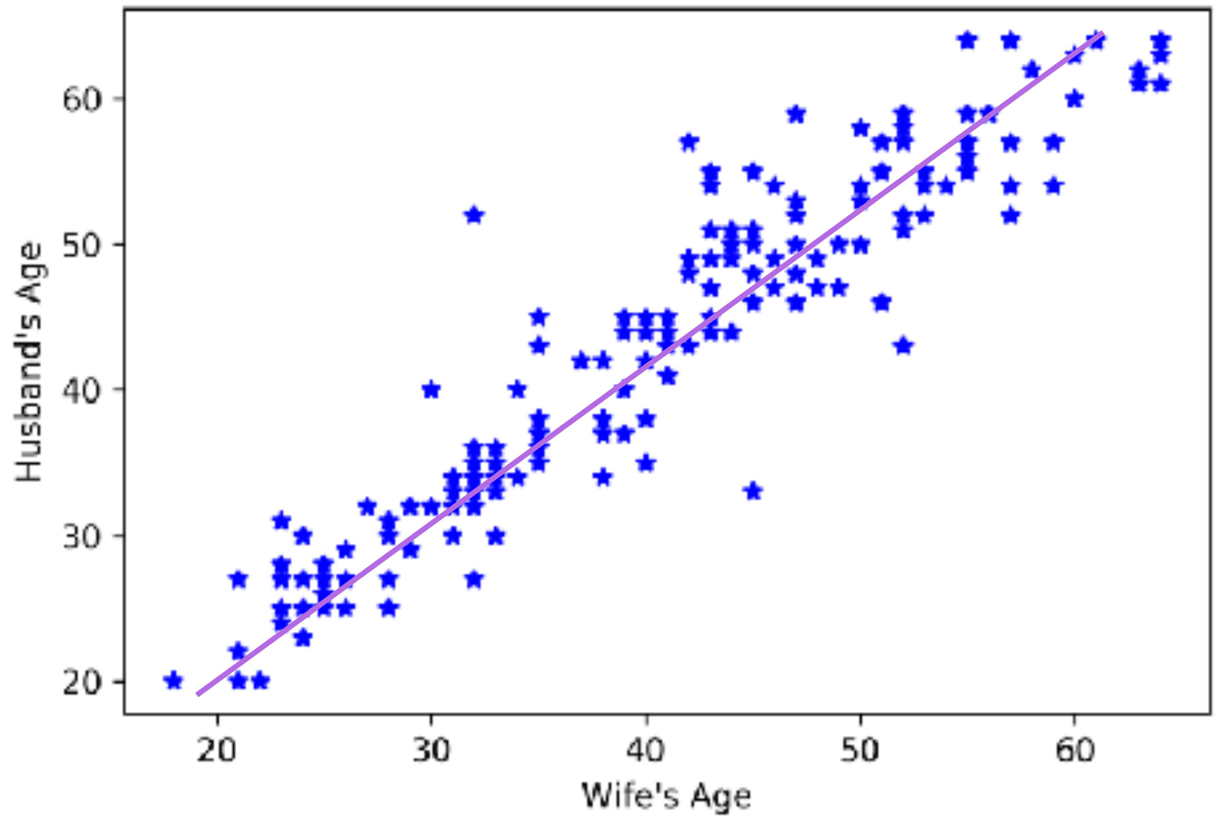


@bgoncalves

Fundamental tools

- Points
- Lines
- Areas
- Shapes
- Colors
- Text

- Each for these can be used to encode a given variable to produce all the types of plots we are familiar with:
- Scatter plot - Just points (line)

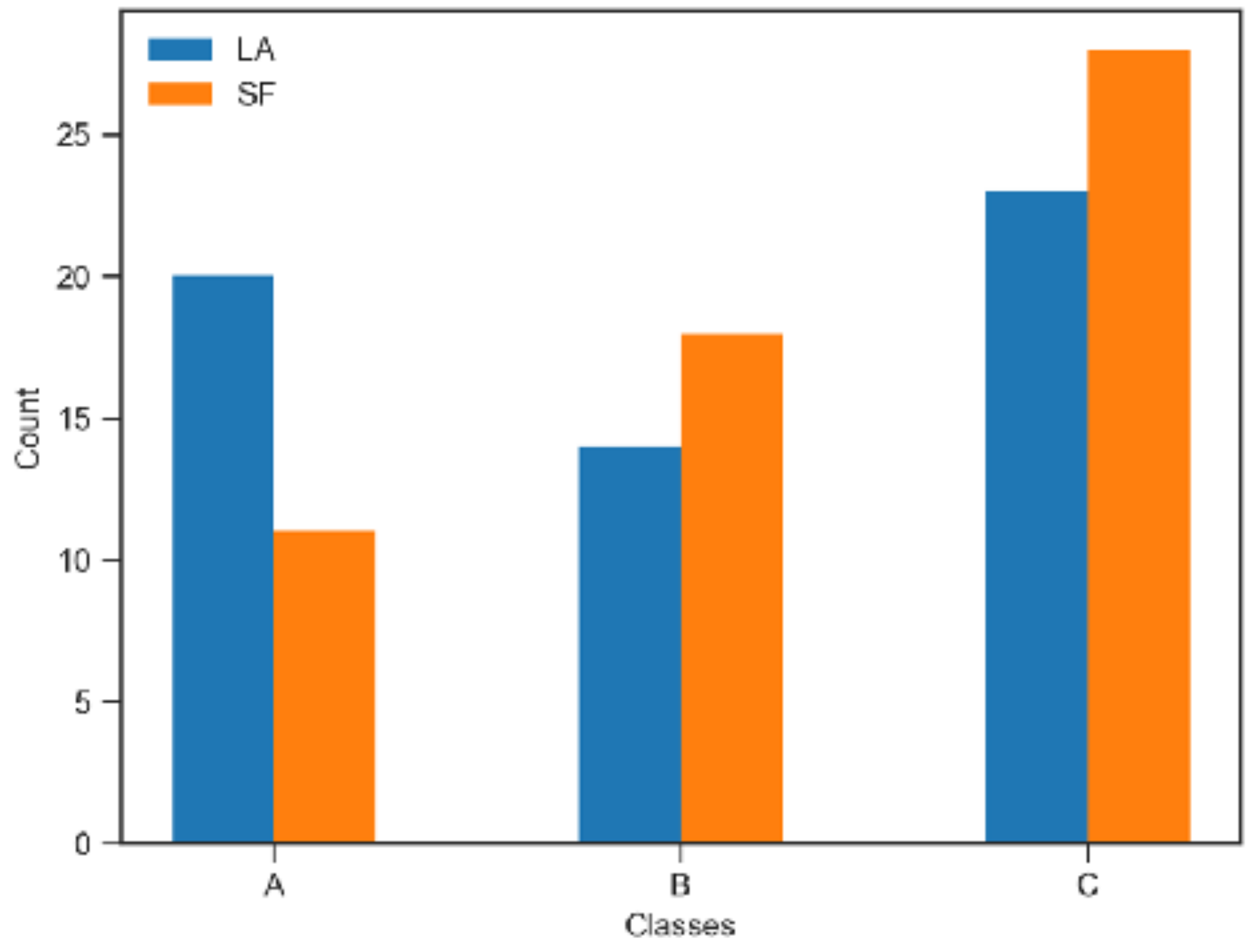


@bgoncalves

Fundamental tools

- Points
- Lines
- Areas
- Shapes
- Colors
- Text

- Each for these can be used to encode a given variable to produce all the types of plots we are familiar with:
- Scatter plot - Just points (line)
- Bar chart - Areas

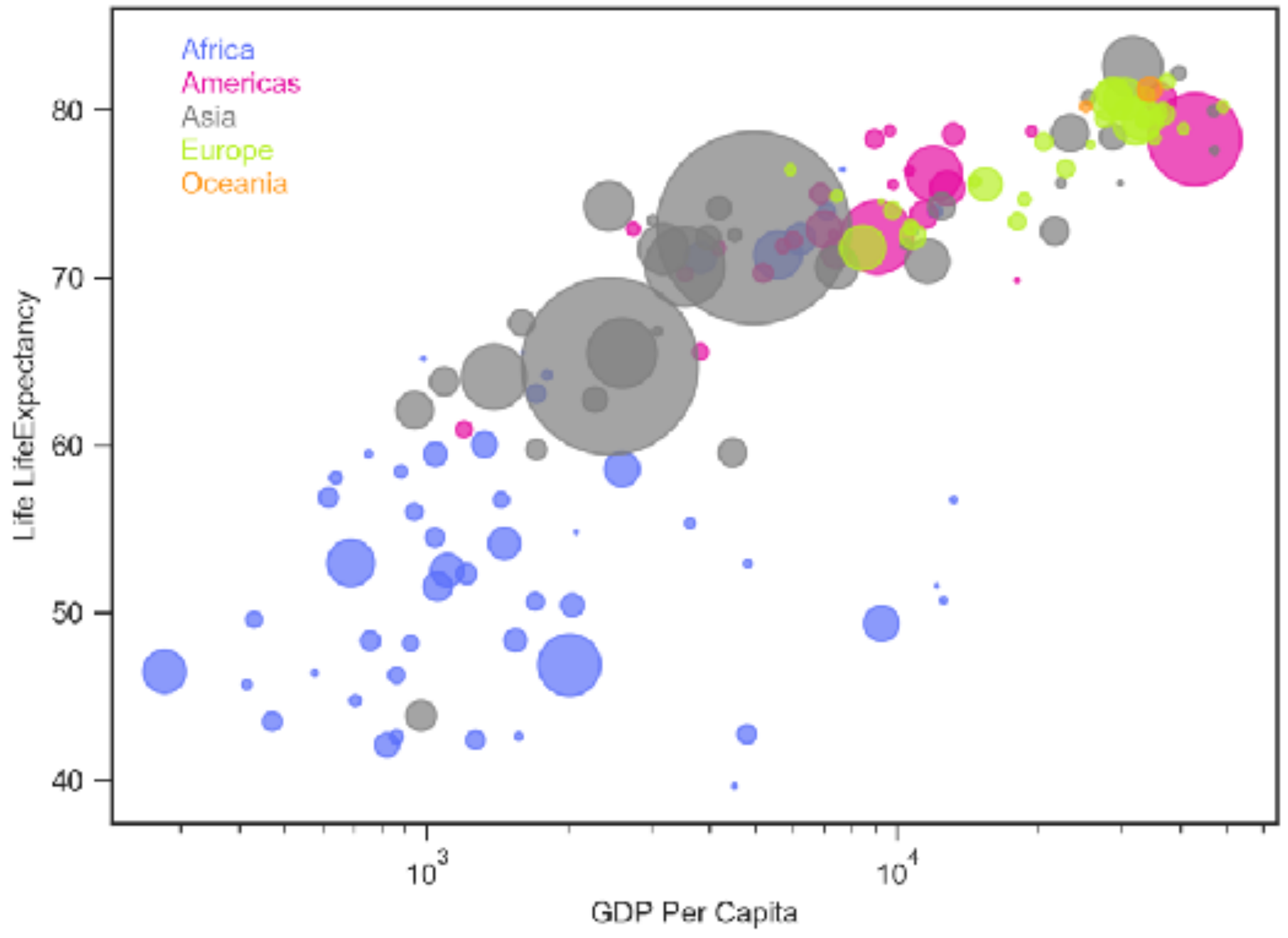


@bgoncalves

Fundamental tools

- Points
- Lines
- Areas
- Shapes
- Colors
- Text

- Each for these can be used to encode a given variable to produce all the types of plots we are familiar with:
- Scatter plot - Just points (line)
- Bar chart - Areas
- Bubble chart - Scatter plot + size + color (time)

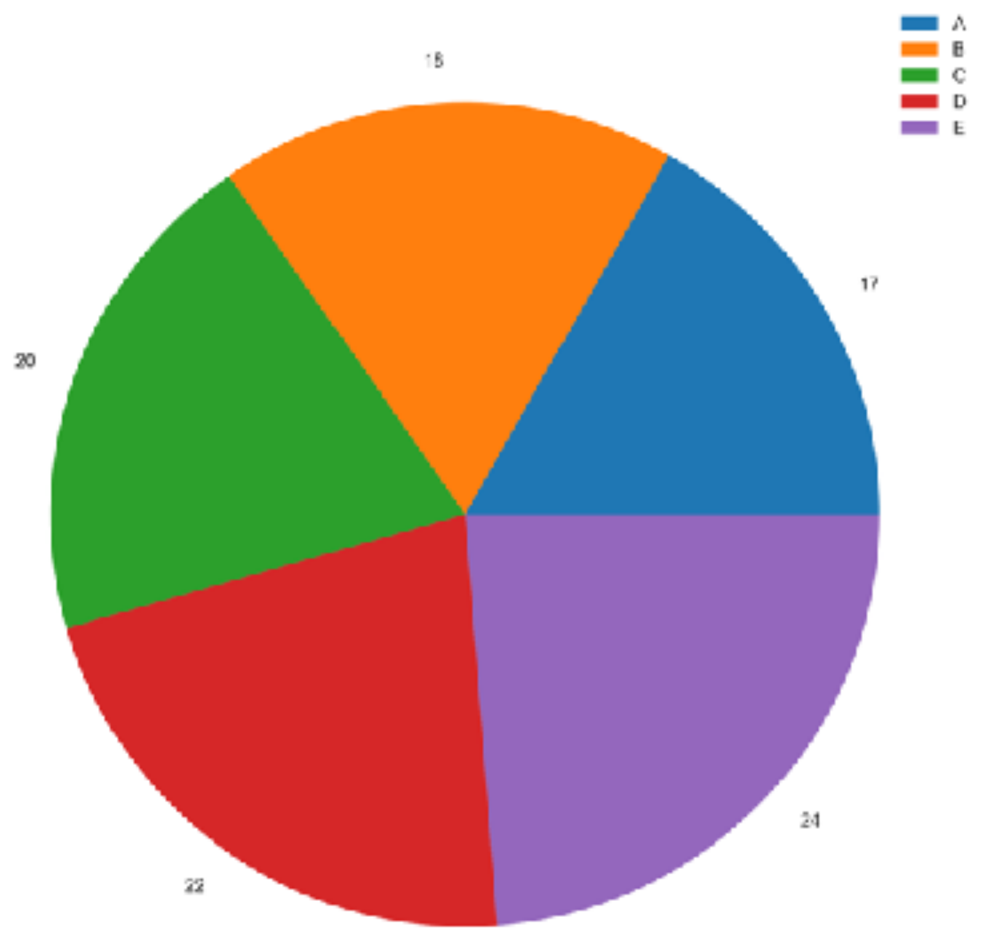


@bgoncalves

Fundamental tools

- Points
- Lines
- Areas
- Shapes
- Colors
- Text

- Each for these can be used to encode a given variable to produce all the types of plots we are familiar with:
- Scatter plot - Just points (line)
- Bar chart - Areas
- Bubble chart - Scatter plot + size + color (time)
- Pie chart - Areas + colors



@bgoncalves

Fundamental tools

- Points
- Lines
- Areas
- Shapes
- Colors
- Text

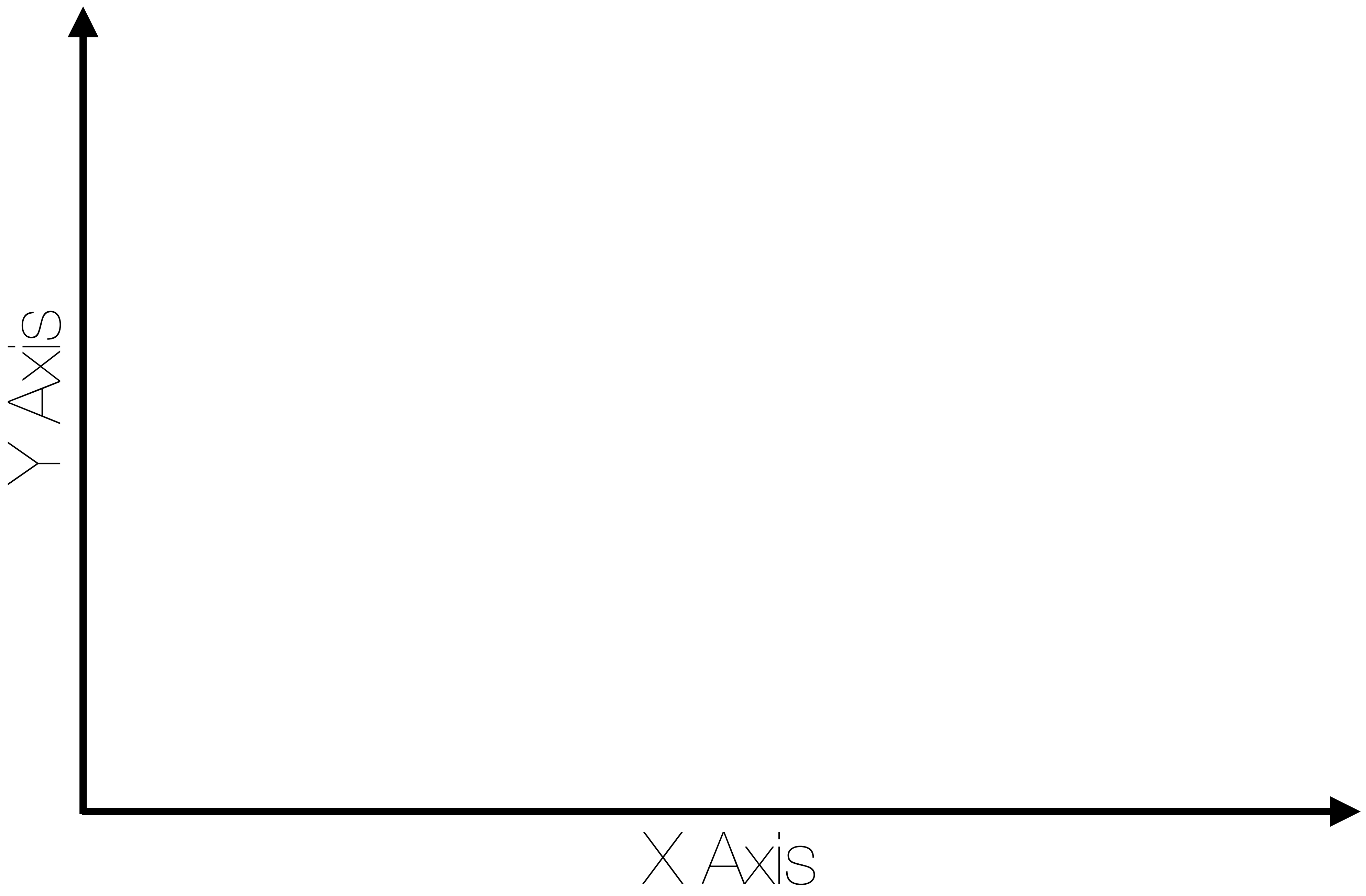
- Each for these can be used to encode a given variable to produce all the types of plots we are familiar with:
- Scatter plot - Just points (line)
- Bar chart - Areas
- Bubble chart - Scatter plot + size + color (time)
- Pie chart - Areas + colors
- etc…

Matplotlib

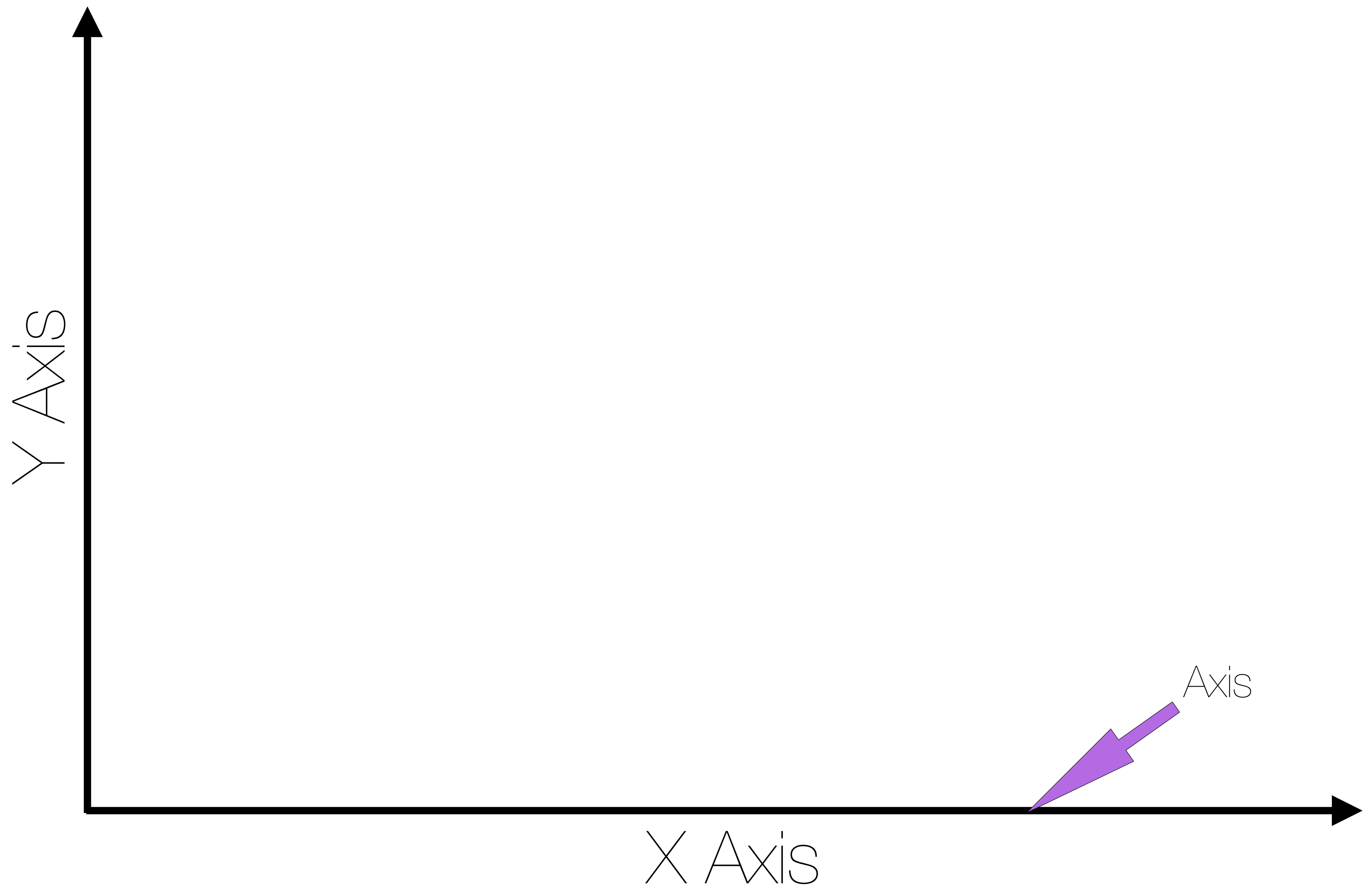
Basic Plotting

Basic Plotting

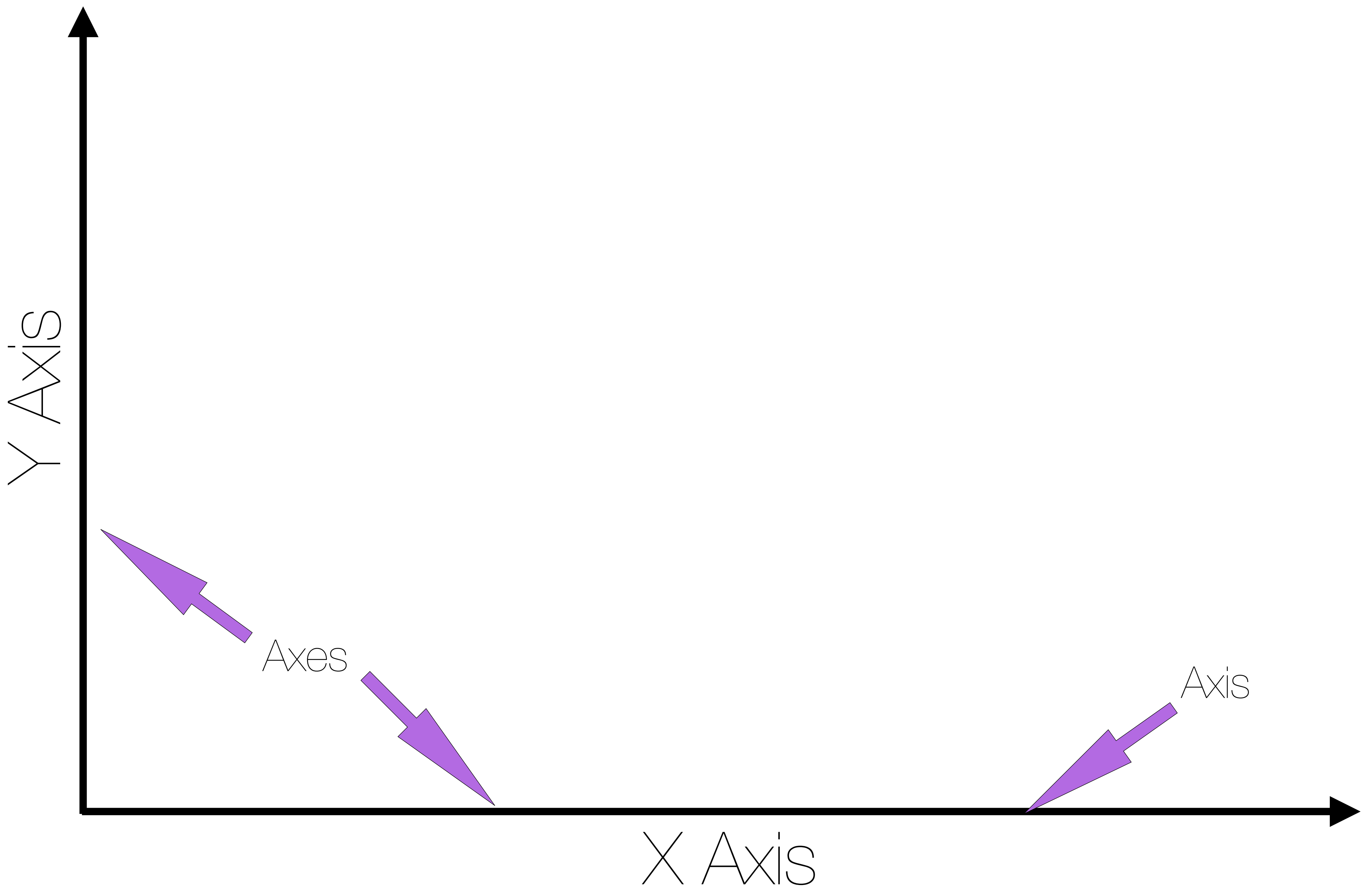
Basic Plotting



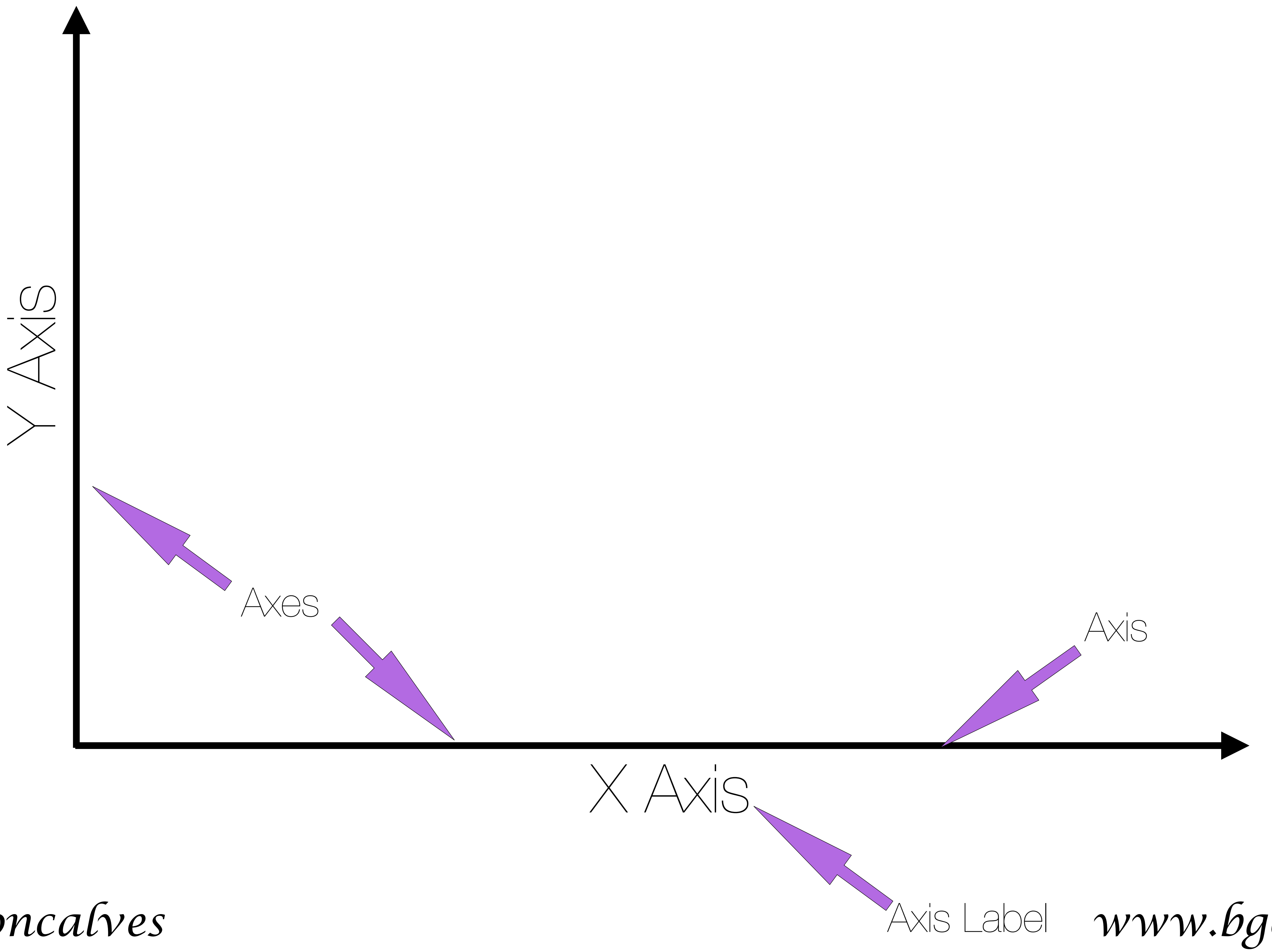
Basic Plotting



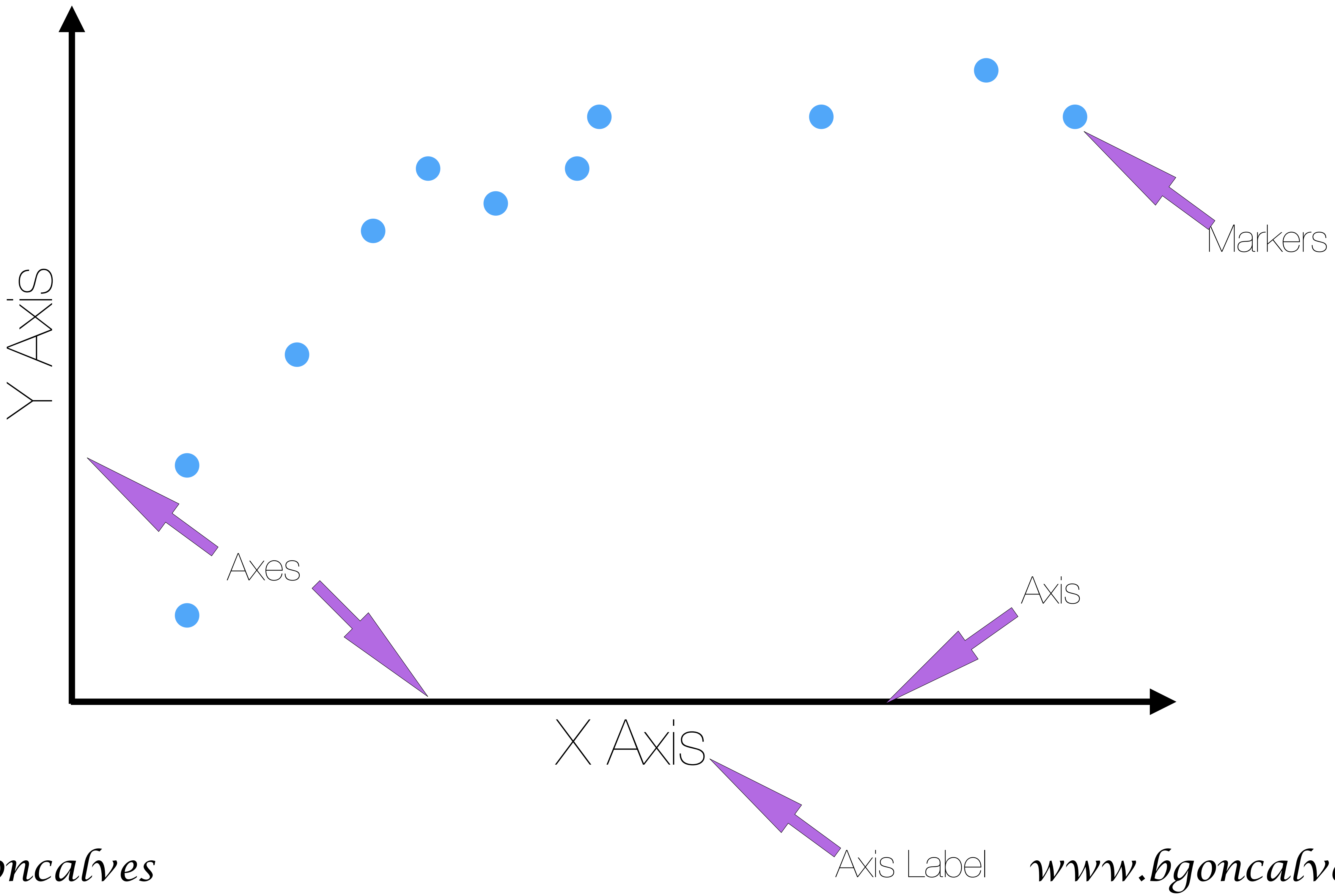
Basic Plotting



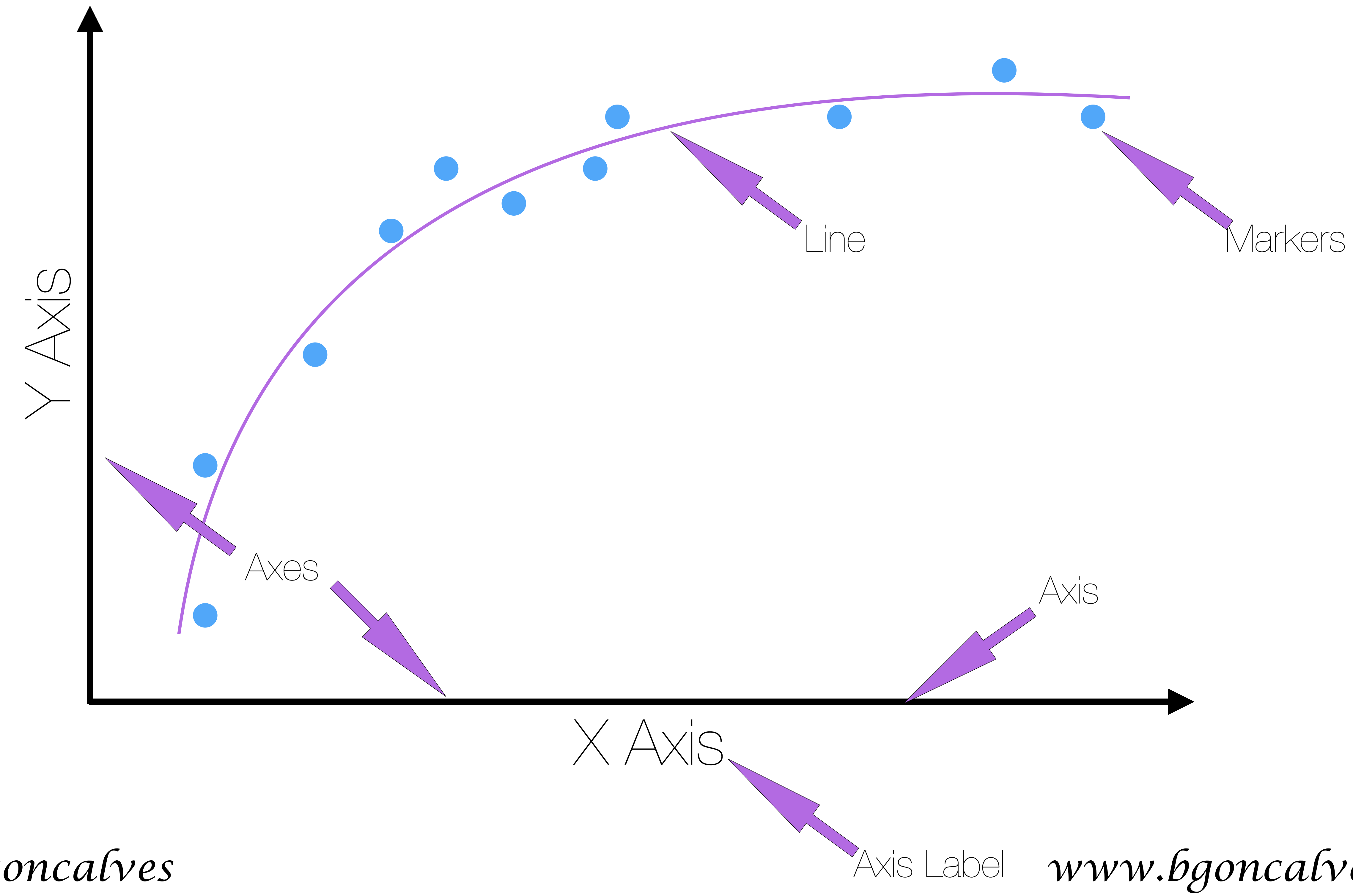
Basic Plotting



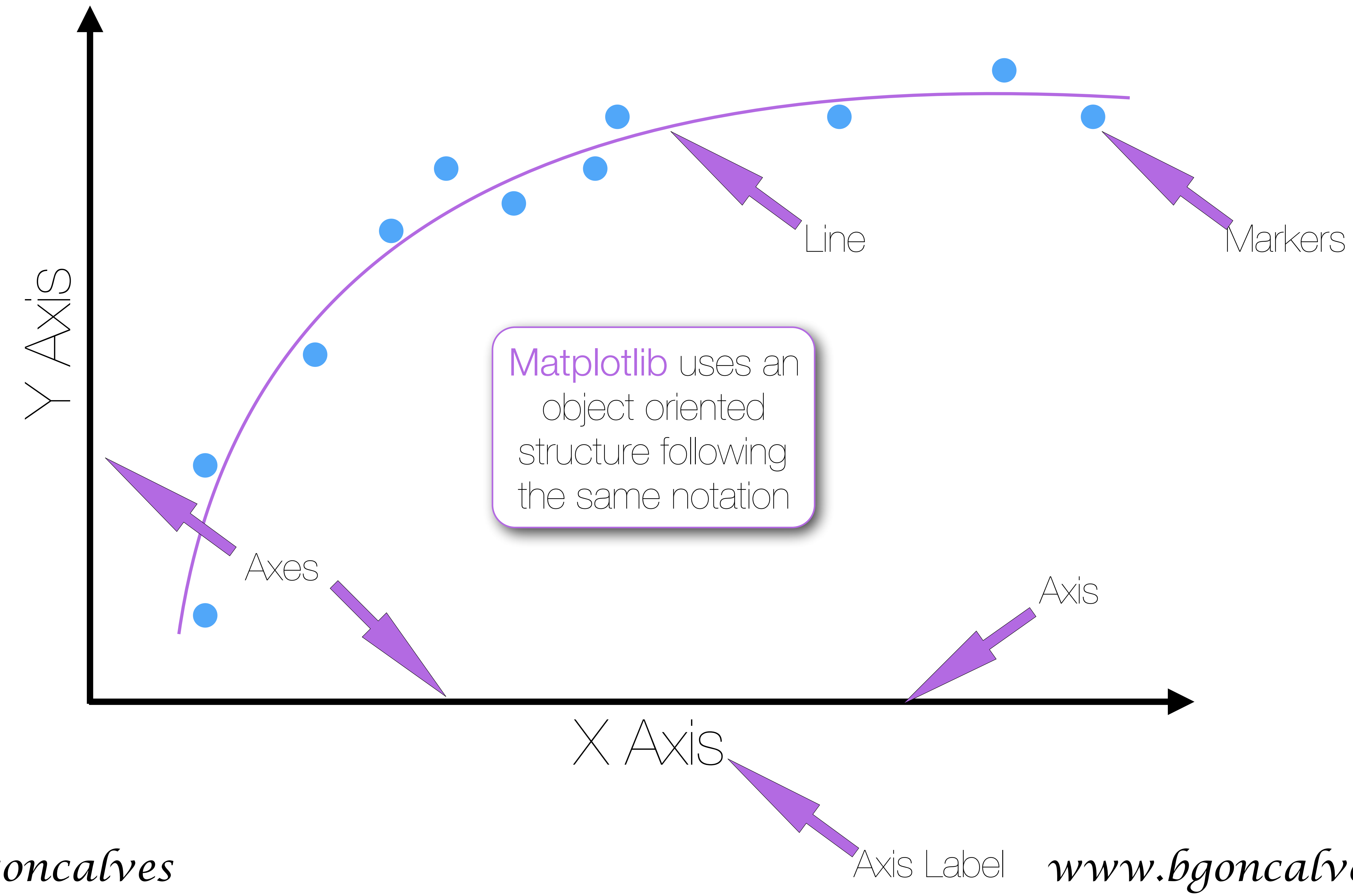
Basic Plotting



Basic Plotting

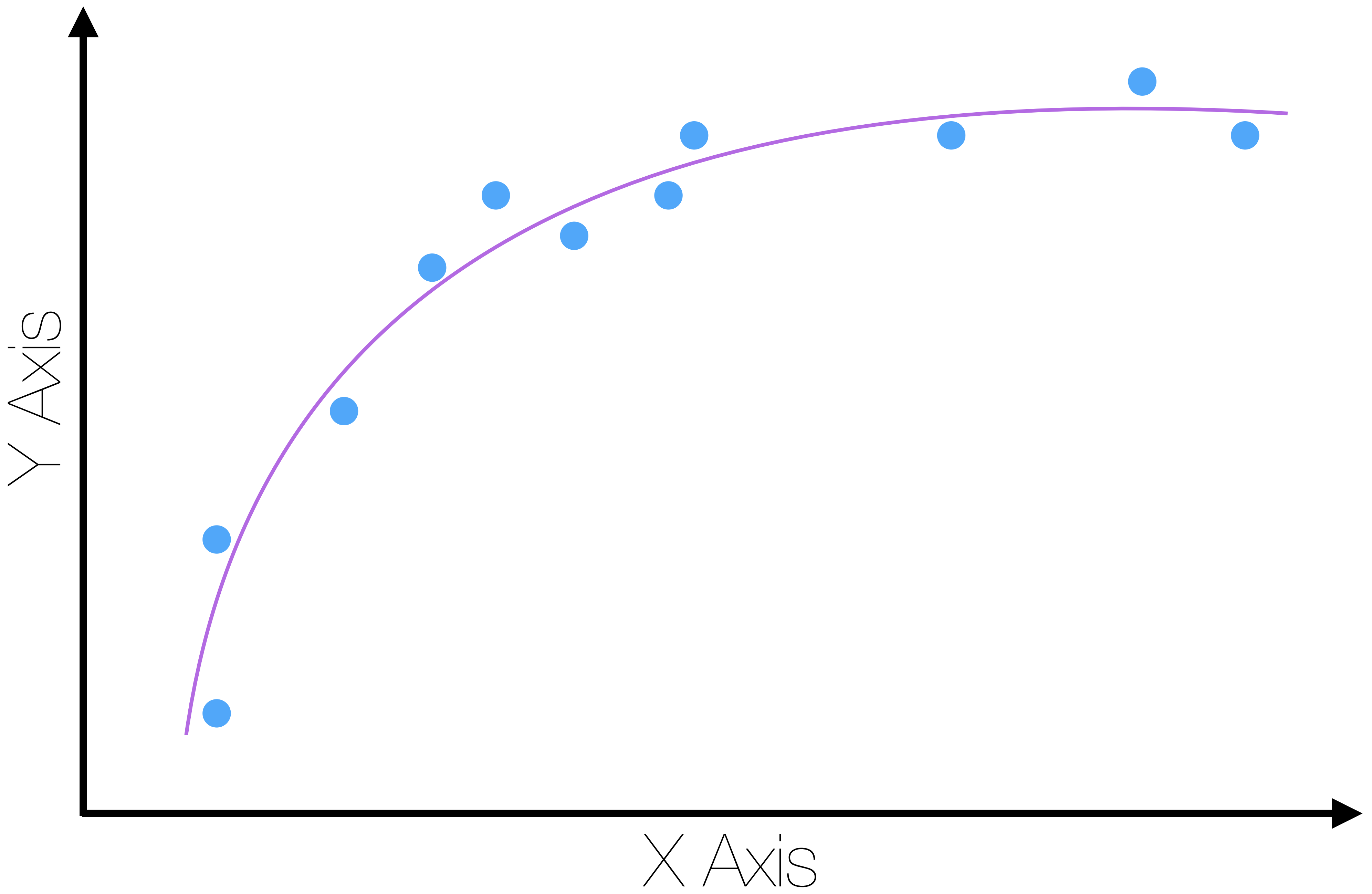


Basic Plotting

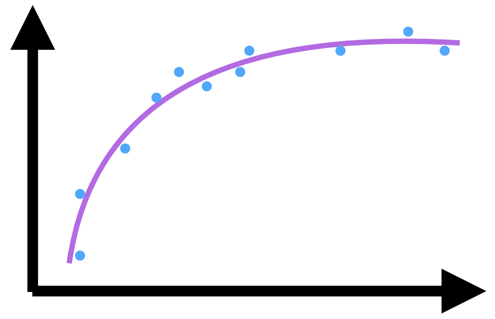


Matplotlib uses an object oriented structure following the same notation

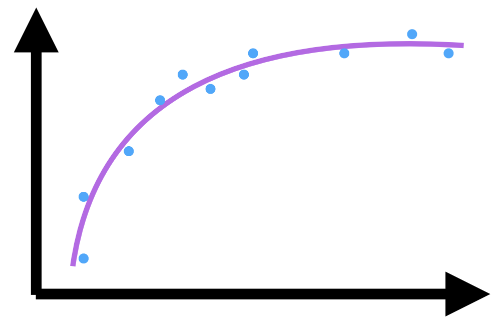
Basic Plotting



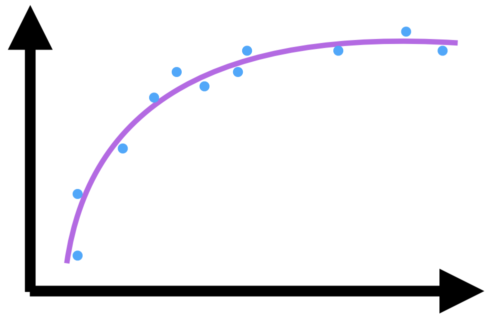
Basic Plotting



Basic Plotting



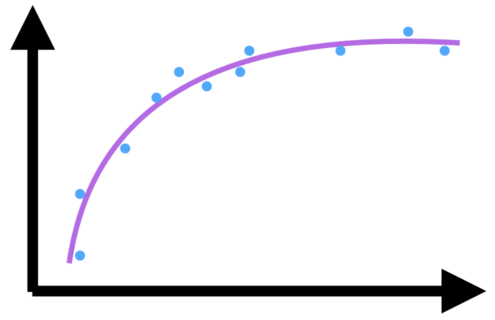
- Matplotlib uses an object oriented structure following an intuitive notation



Basic Plotting

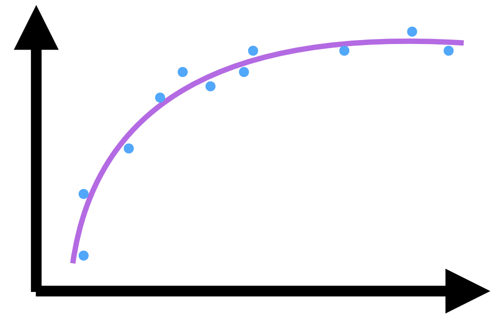
- Matplotlib uses an object oriented structure following an intuitive notation
- Each Axes object contains one or more Axis objects.

Basic Plotting



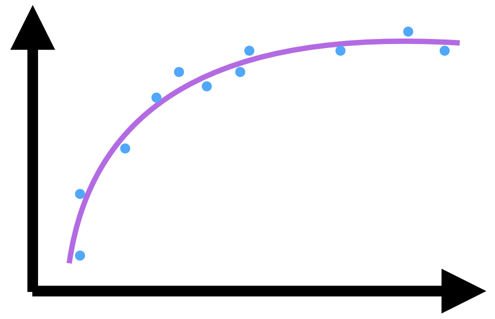
- Matplotlib uses an object oriented structure following an intuitive notation
- Each Axes object contains one or more Axis objects.
- A Figure is a set of one or more Axes.

Basic Plotting

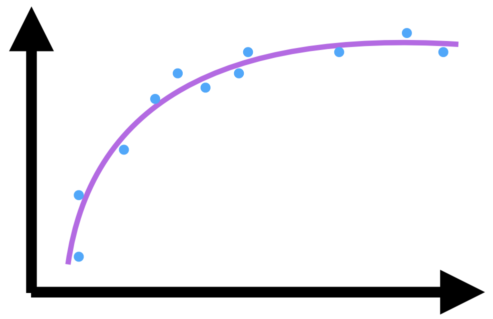


- Matplotlib uses an object oriented structure following an intuitive notation
- Each Axes object contains one or more Axis objects.
- A Figure is a set of one or more Axes.
- Each Axes is associated with exactly one Figure and each set of Markers is associated with exactly one Axes.

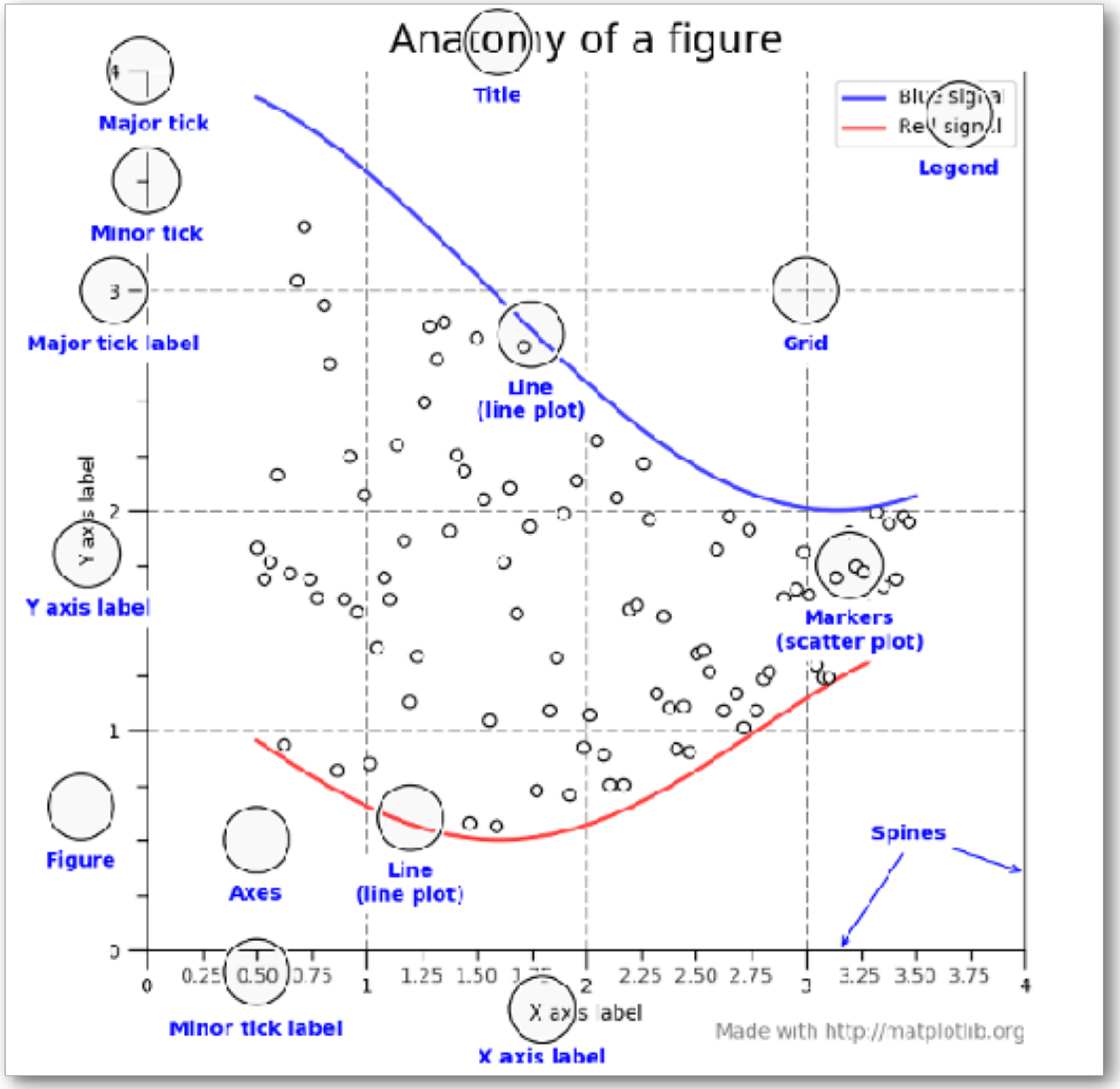
Basic Plotting



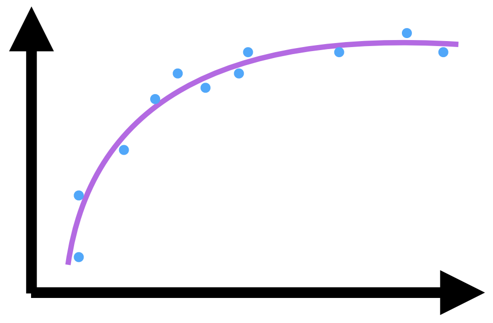
- Matplotlib uses an object oriented structure following an intuitive notation
- Each Axes object contains one or more Axis objects.
- A Figure is a set of one or more Axes.
- Each Axes is associated with exactly one Figure and each set of Markers is associated with exactly one Axes.
- In other words, Markers/Lines represent a dataset that is plotted against one or more Axis. An Axes object is (effectively) a subplot of a Figure.



Basic Plotting - Programmatically!

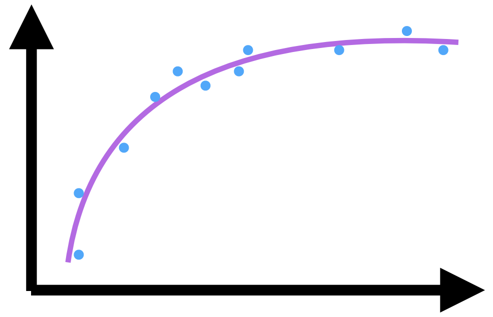


Basic Plotting - Programmatically!



https://matplotlib.org/2.0.0/

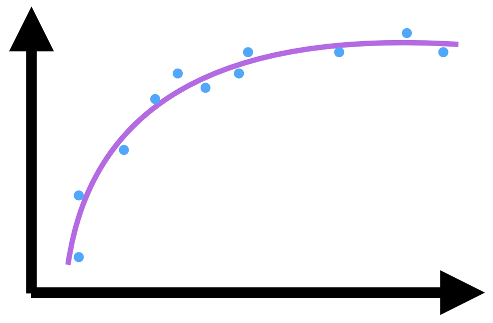
Basic Plotting - Programmatically!



https://matplotlib.org/2.0.0/

- While the Figure object controls the way in which the figure is displayed.

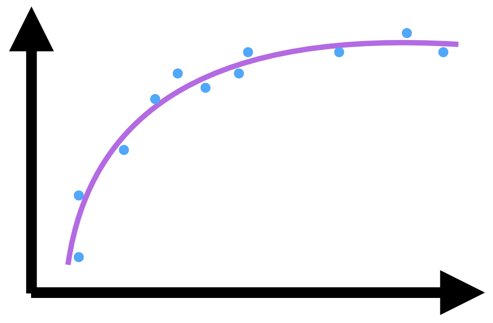
Basic Plotting - Programmatically!



https://matplotlib.org/2.0.0/

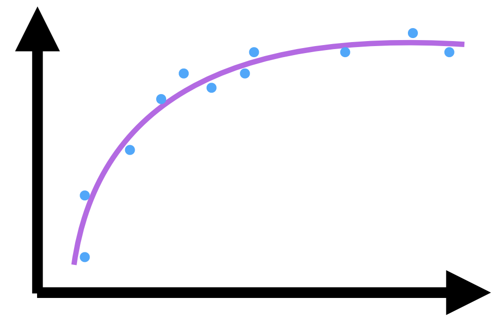
- While the Figure object controls the way in which the figure is displayed.
    - .gca() - Get the current Axes, creating one if necessary

Basic Plotting - Programmatically!



https://matplotlib.org/2.0.0/

- While the Figure object controls the way in which the figure is displayed.
    - .gca() - Get the current Axes, creating one if necessary
    - .show() - Show the final figure



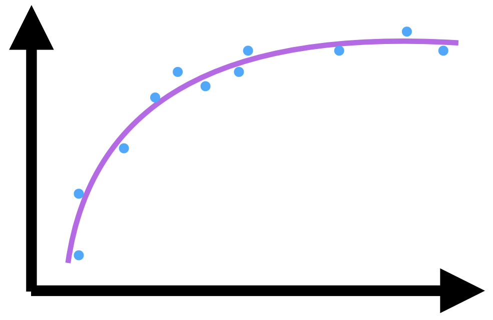
https://matplotlib.org/2.0.0/

Basic Plotting - Programmatically!

- While the Figure object controls the way in which the figure is displayed.
  - .gca() - Get the current Axes, creating one if necessary
  - .show() - Show the final figure
  - .savefig("filename.ext", dpi=300) - Save the figure to "filename.ext" where ".ext" defines the format the saved image ()

```
filetypes = {'ps': 'Postscript', 'eps': 'Encapsulated Postscript', 'pdf': 'Portable Document Format',
'pgf': 'PGF code for LaTeX', 'png': 'Portable Network Graphics', 'raw': 'Raw RGBA bitmap', 'rgba': 'Raw
RGBA bitmap', 'svg': 'Scalable Vector Graphics', 'svgz': 'Scalable Vector Graphics', 'jpg': 'Joint
Photographic Experts Group', 'jpeg': 'Joint Photographic Experts Group', 'tif': 'Tagged Image File
Format', 'tiff': 'Tagged Image File Format'}
```

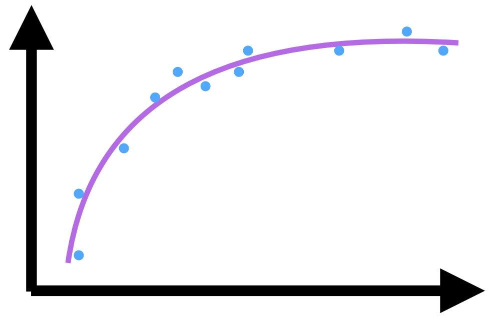
Basic Plotting - Programmatically!



Basic Plotting - Programmatically!

- The first step is to import the pyplot module from matplotlib and instanciating a Figure object:

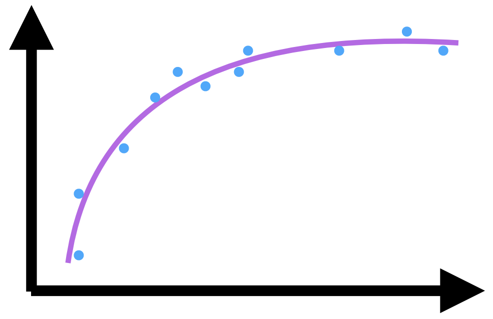
Basic Plotting - Programmatically!



- The first step is to import the pyplot module from matplotlib and instanciating a Figure object:

```
import matplotlib.pyplot as plt
fig = plt.figure()
```

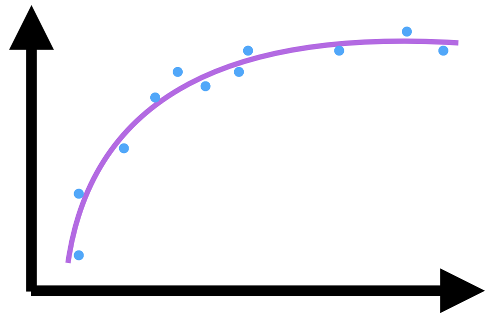
Basic Plotting - Programmatically!



- The first step is to import the pyplot module from matplotlib and instanciating a Figure object:

```
import matplotlib.pyplot as plt
fig = plt.figure()
```

- The convention is to import pyplot as plt



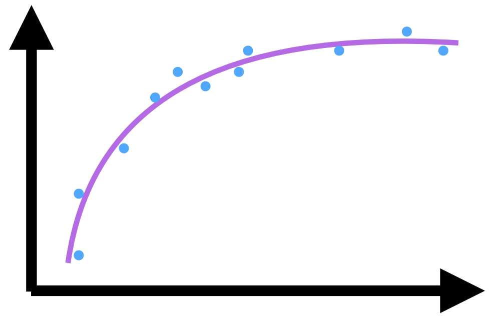
Basic Plotting - Programmatically!

- The first step is to import the pyplot module from matplotlib and instanciating a Figure object:

```
import matplotlib.pyplot as plt
fig = plt.figure()
```

- The convention is to import pyplot as plt
- To create subplots (Axes) you use .subplots(nrows, ncols, sharex=False, sharey=False) instead of .figure(). set sharex and/or sharey to True to keep the same scale in both cases.

Basic Plotting - Programmatically!

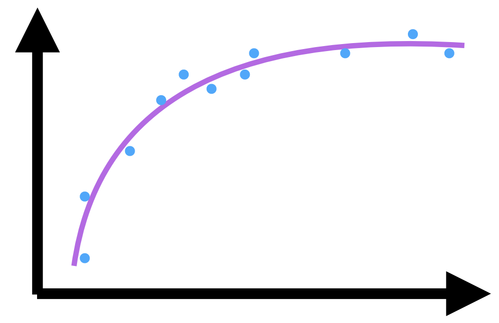


- The first step is to import the pyplot module from matplotlib and instanciating a Figure object:

```
import matplotlib.pyplot as plt
fig = plt.figure()
```

- The convention is to import pyplot as plt
- To create subplots (Axes) you use .subplots(nrows, ncols, sharex=False, sharey=False) instead of .figure(). set sharex and/or sharey to True to keep the same scale in both cases.
- .subplots - returns a (fig, ax_lst) tuple where ax_lst is a list of Axes and fig is the Figure.

Basic Plotting - Programmatically!

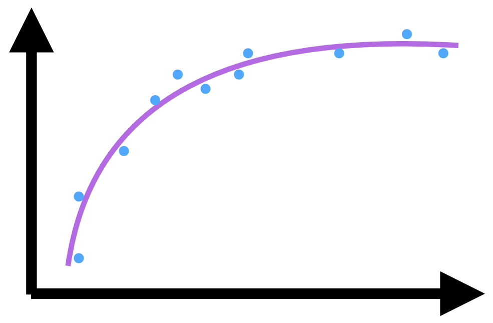


- The first step is to import the pyplot module from matplotlib and instanciating a Figure object:

```
import matplotlib.pyplot as plt
fig = plt.figure()
```

- The convention is to import pyplot as plt
- To create subplots (Axes) you use .subplots(nrows, ncols, sharex=False, sharey=False) instead of .figure(). set sharex and/or sharey to True to keep the same scale in both cases.
- .subplots - returns a (fig, ax_lst) tuple where ax_lst is a list of Axes and fig is the Figure.
- Axes have several methods of interest:

Basic Plotting - Programmatically!

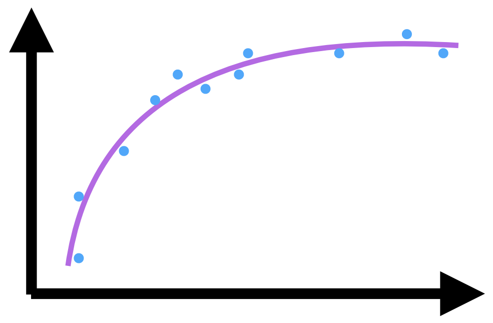


- The first step is to import the pyplot module from matplotlib and instanciating a Figure object:

```
import matplotlib.pyplot as plt
fig = plt.figure()
```

- The convention is to import pyplot as plt
- To create subplots (Axes) you use .subplots(nrows, ncols, sharex=False, sharey=False) instead of .figure(). set sharex and/or sharey to True to keep the same scale in both cases.
- .subplots - returns a (fig, ax_lst) tuple where ax_lst is a list of Axes and fig is the Figure.
- Axes have several methods of interest:
  - .plot(x, y) - Make a scatter or line plot from a list of x, y coordinates.

Basic Plotting - Programmatically!

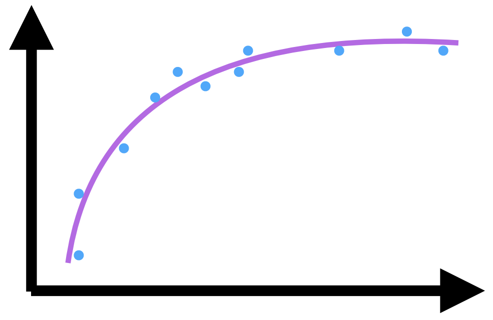


- The first step is to import the pyplot module from matplotlib and instanciating a Figure object:

```
import matplotlib.pyplot as plt
fig = plt.figure()
```

- The convention is to import pyplot as plt
- To create subplots (Axes) you use .subplots(nrows, ncols, sharex=False, sharey=False) instead of .figure(). set sharex and/or sharey to True to keep the same scale in both cases.
- .subplots - returns a (fig, ax_lst) tuple where ax_lst is a list of Axes and fig is the Figure.
- Axes have several methods of interest:
    - .plot(x, y) - Make a scatter or line plot from a list of x, y coordinates.
    - .imshow(mat) - Plot a matrix as if it were an image. Element 0,0 is plotted in the top right corner.

Basic Plotting - Programmatically!

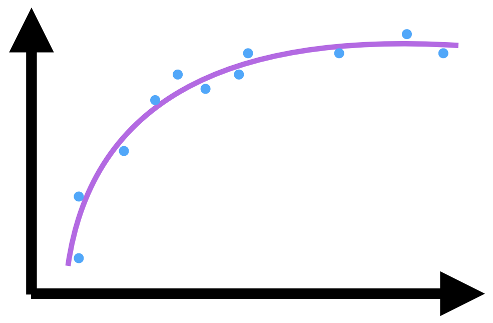


- The first step is to import the pyplot module from matplotlib and instanciating a Figure object:

```
import matplotlib.pyplot as plt
fig = plt.figure()
```

- The convention is to import pyplot as plt
- To create subplots (Axes) you use .subplots(nrows, ncols, sharex=False, sharey=False) instead of .figure(). set sharex and/or sharey to True to keep the same scale in both cases.
- .subplots - returns a (fig, ax_lst) tuple where ax_lst is a list of Axes and fig is the Figure.
- Axes have several methods of interest:
  - .plot(x, y) - Make a scatter or line plot from a list of x, y coordinates.
  - .imshow(mat) - Plot a matrix as if it were an image. Element 0,0 is plotted in the top right corner.
  - .bar(x, y) - Make a bar plot where x is a list of the lower left coordinates of each bar and y is the respective height.

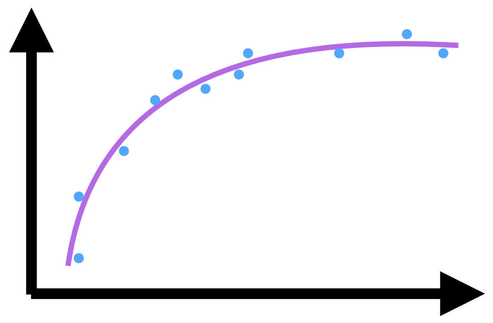
Basic Plotting - Programmatically!



- The first step is to import the pyplot module from matplotlib and instanciating a Figure object:

```
import matplotlib.pyplot as plt
fig = plt.figure()
```

- The convention is to import pyplot as plt
- To create subplots (Axes) you use .subplots(nrows, ncols, sharex=False, sharey=False) instead of .figure(). set sharex and/or sharey to True to keep the same scale in both cases.
- .subplots - returns a (fig, ax_lst) tuple where ax_lst is a list of Axes and fig is the Figure.
- Axes have several methods of interest:
  - .plot(x, y) - Make a scatter or line plot from a list of x, y coordinates.
  - .imshow(mat) - Plot a matrix as if it were an image. Element 0,0 is plotted in the top right corner.
  - .bar(x, y) - Make a bar plot where x is a list of the lower left coordinates of each bar and y is the respective height.
  - .pie(values, labels=labels) - Produce a pie plot out of a list of values list and labeled with labels



Basic Plotting - Programmatically!

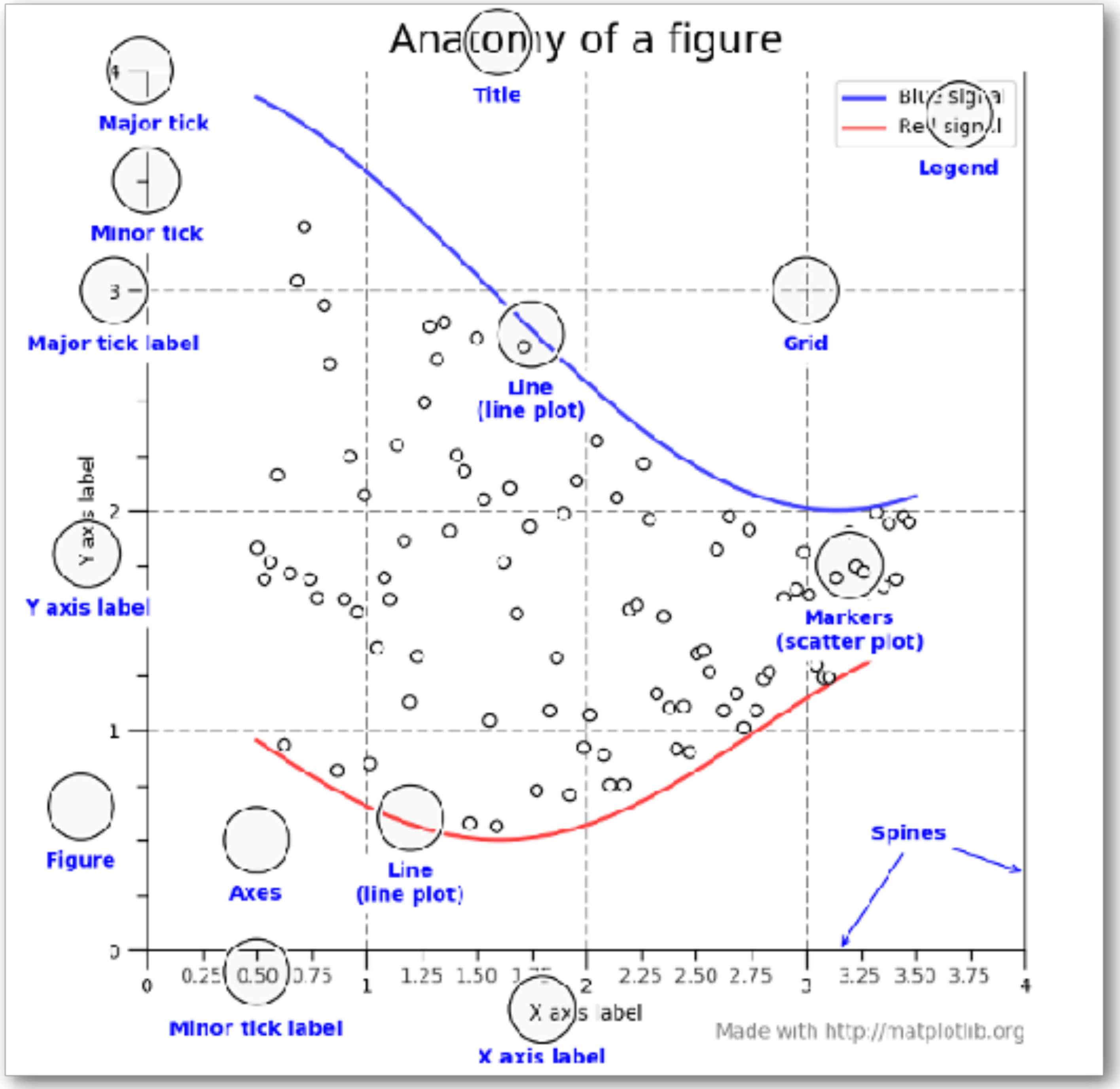
- The first step is to import the pyplot module from matplotlib and instanciating a Figure object:

```
import matplotlib.pyplot as plt
fig = plt.figure()
```

- The convention is to import pyplot as plt
- To create subplots (Axes) you use .subplots(nrows, ncols, sharex=False, sharey=False) instead of .figure(). set sharex and/or sharey to True to keep the same scale in both cases.
- .subplots - returns a (fig, ax_lst) tuple where ax_lst is a list of Axes and fig is the Figure.
- Axes have several methods of interest:
    - .plot(x, y) - Make a scatter or line plot from a list of x, y coordinates.
    - .imshow(mat) - Plot a matrix as if it were an image. Element 0,0 is plotted in the top right corner.
    - .bar(x, y) - Make a bar plot where x is a list of the lower left coordinates of each bar and y is the respective height.
    - .pie(values, labels=labels) - Produce a pie plot out of a list of values list and labeled with labels
    - .savefig(filename) - Write the current figure as an static image

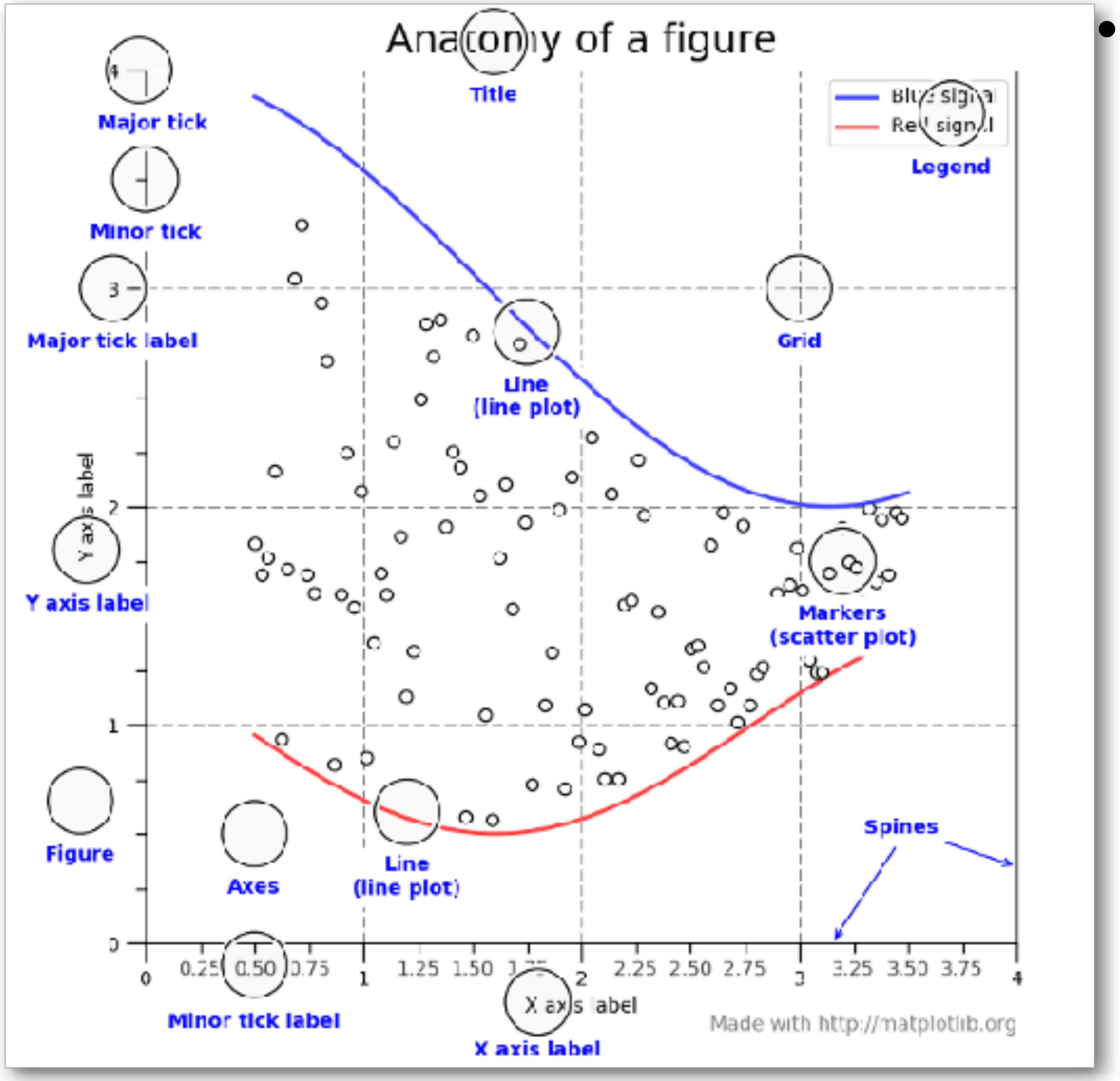
Matplotlib - decorations

https://matplotlib.org/2.0.0/



Matplotlib - decorations

https://matplotlib.org/2.0.0/



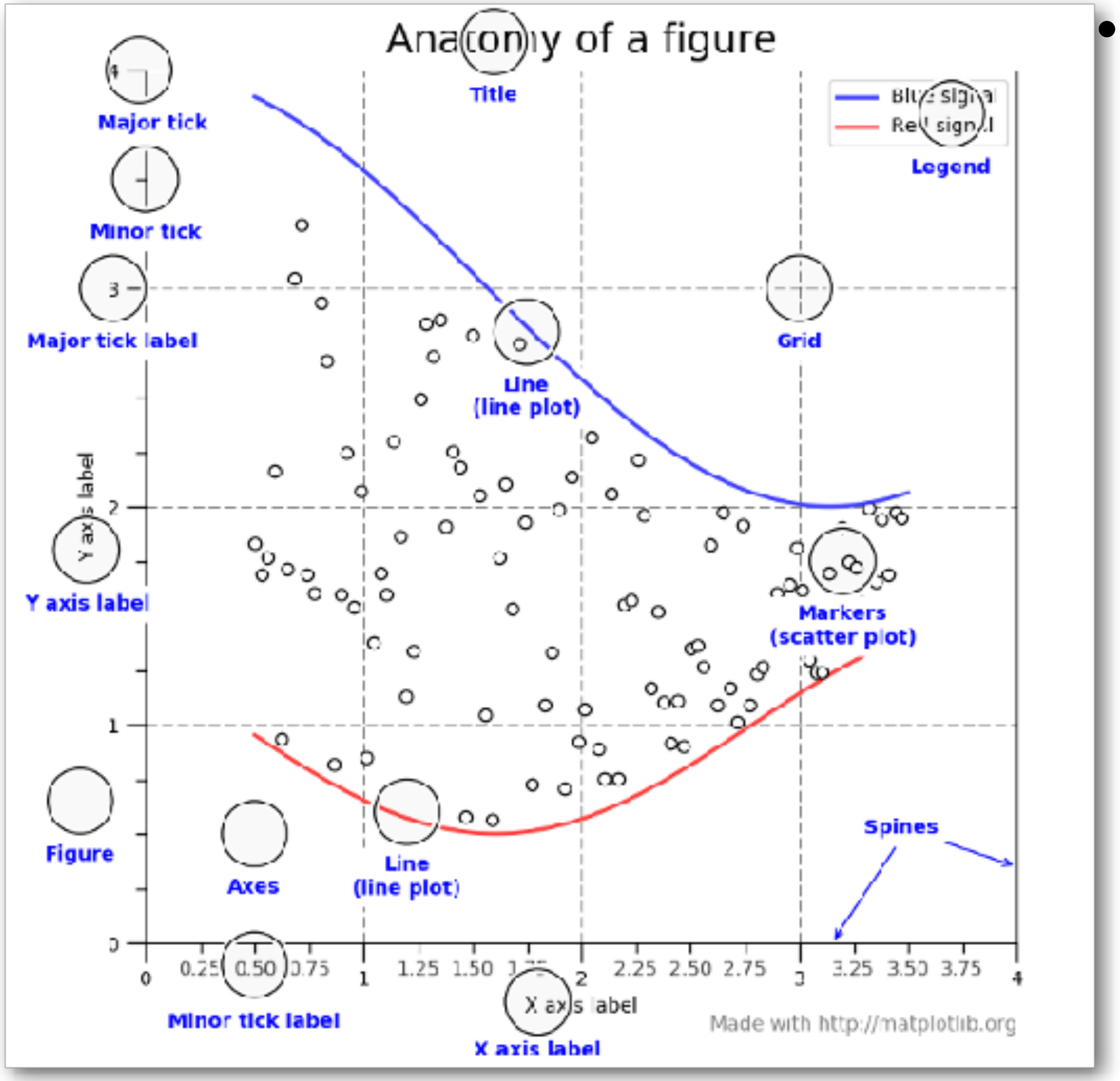
- The respective functions are named in an intuitive wa Every Axes object has as methods:

@bgoncalves

www.bgoncalves.com

Matplotlib - decorations

https://matplotlib.org/2.0.0/

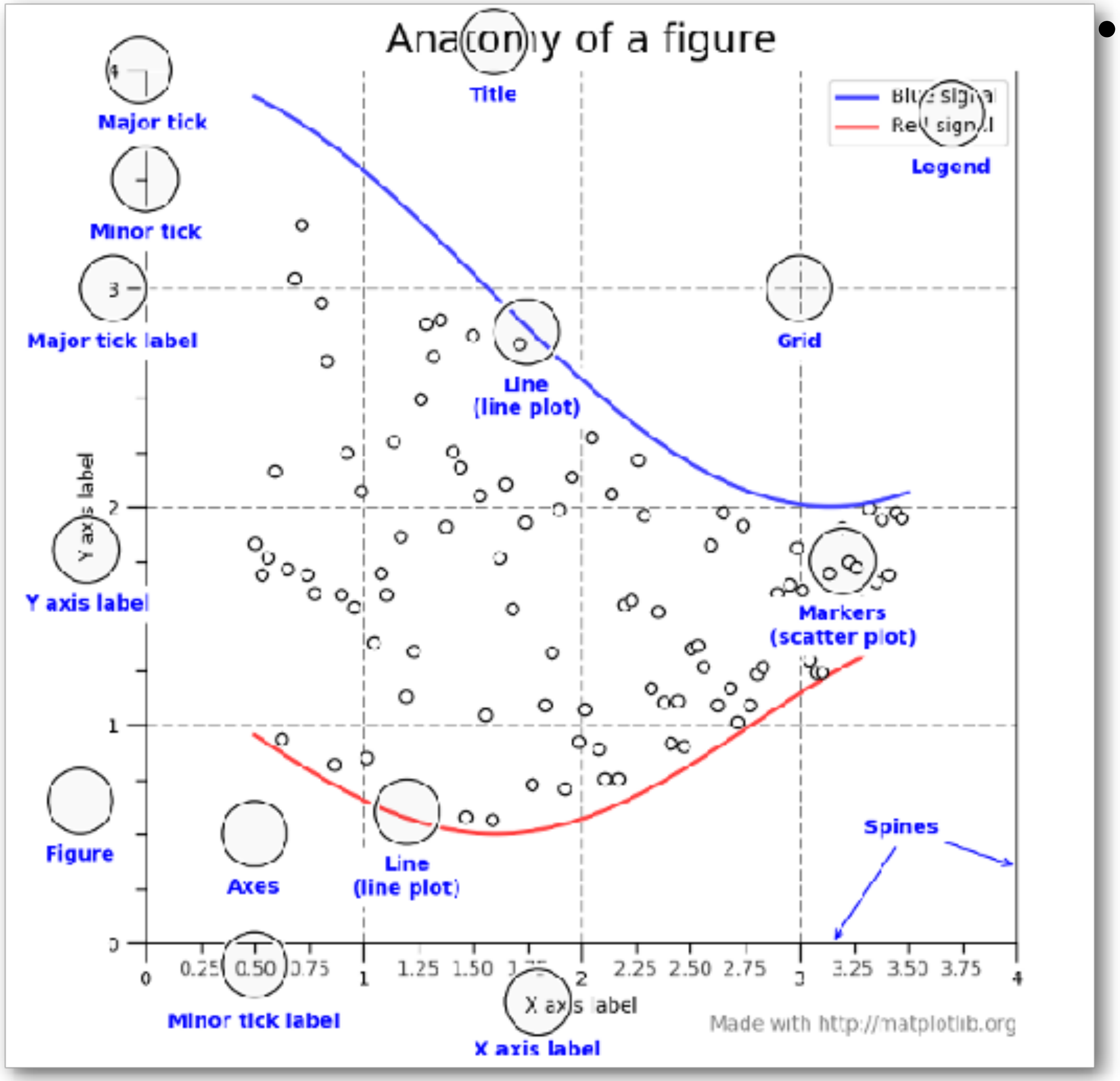


- The respective functions are named in an intuitive wa Every Axes object has as methods:
  - .set_xlabel(label)

@bgoncalves www.bgoncalves.com

Matplotlib - decorations

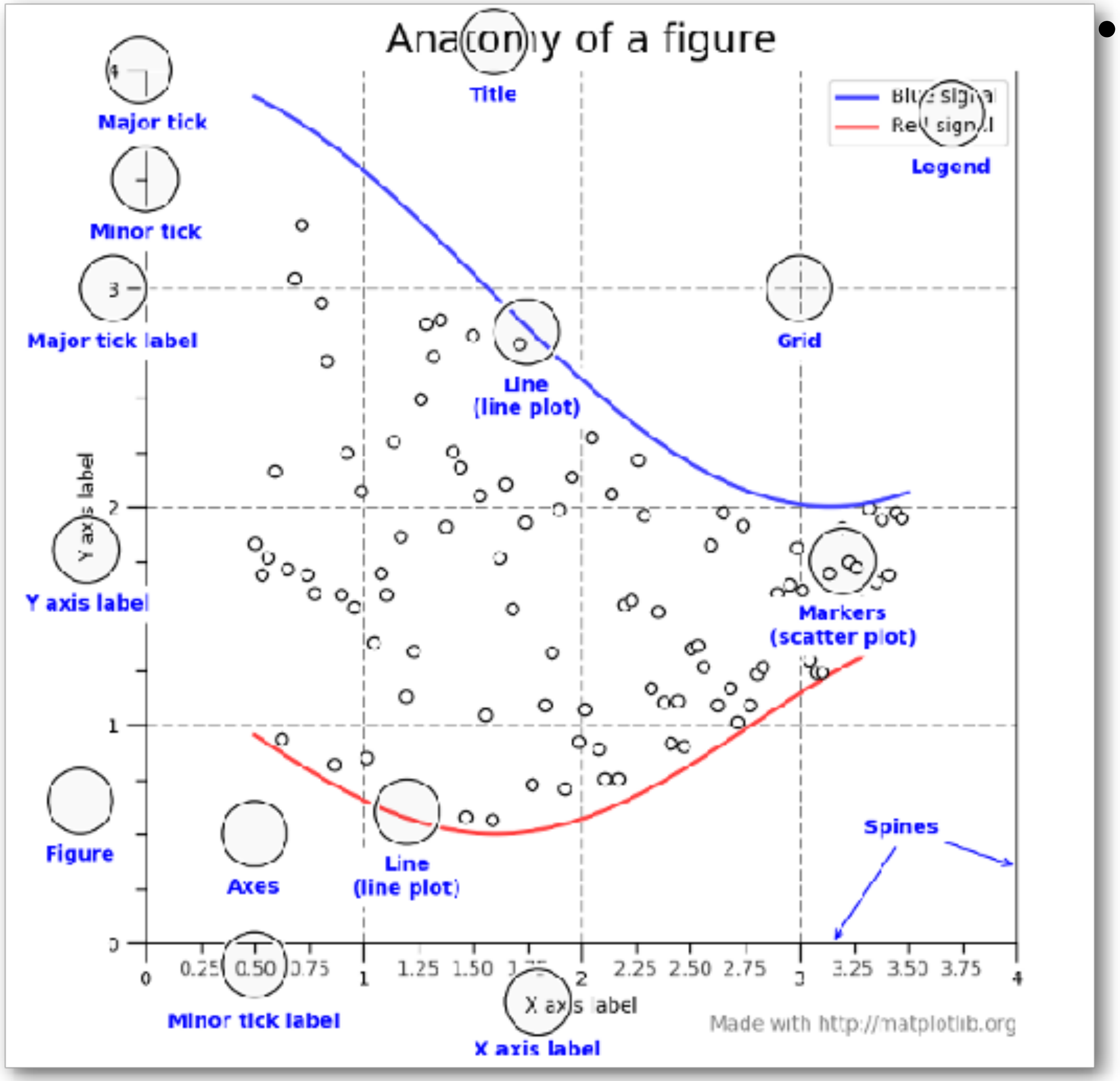
https://matplotlib.org/2.0.0/



- The respective functions are named in an intuitive wa
Every Axes object has as methods:
    - .set_xlabel(label)
    - .set_ylabel(label)

Matplotlib - decorations

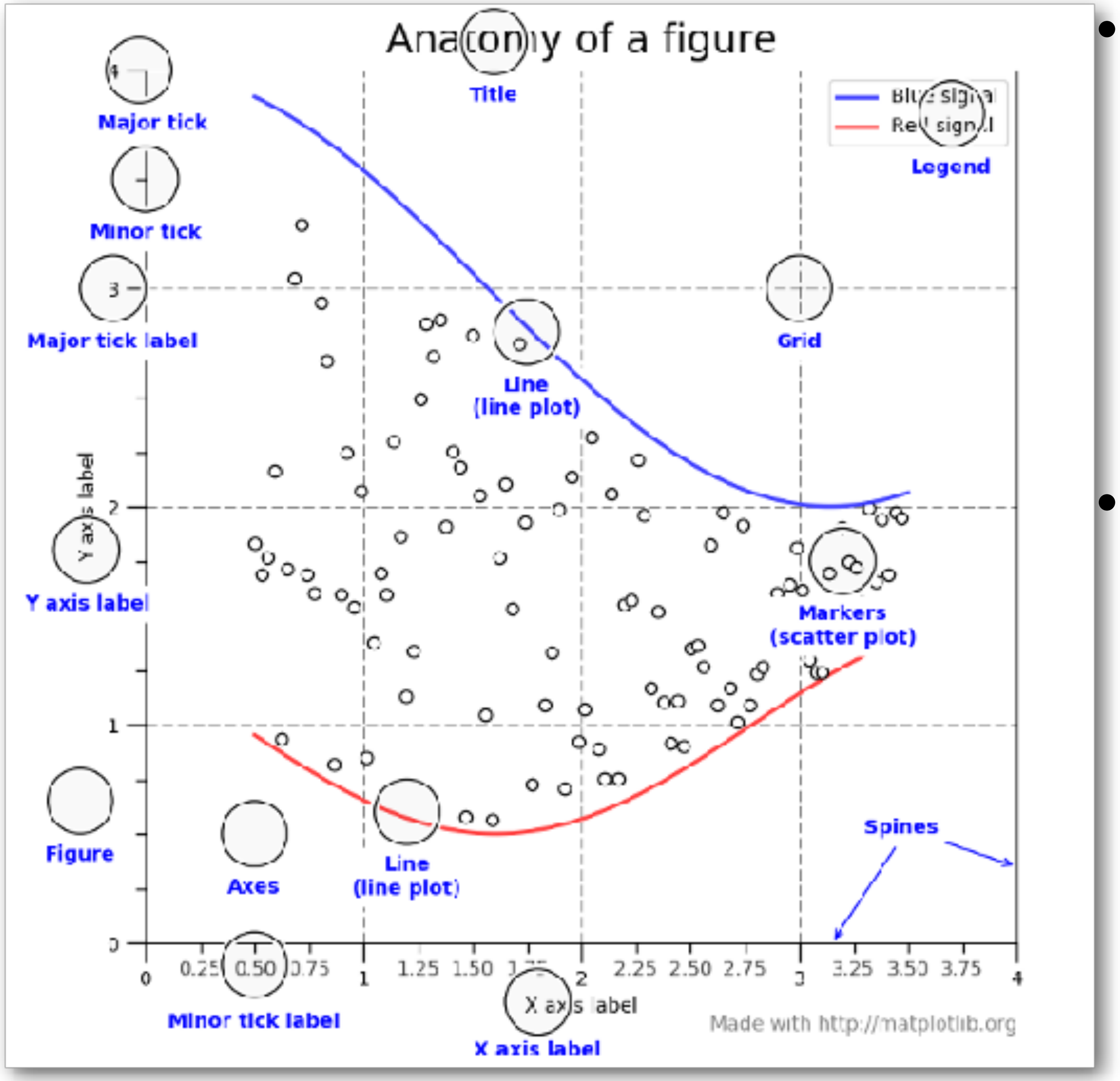
https://matplotlib.org/2.0.0/



- The respective functions are named in an intuitive wa
  Every Axes object has as methods:
    - .set_xlabel(label)
    - .set_ylabel(label)
    - .set_title(title)

Matplotlib - decorations

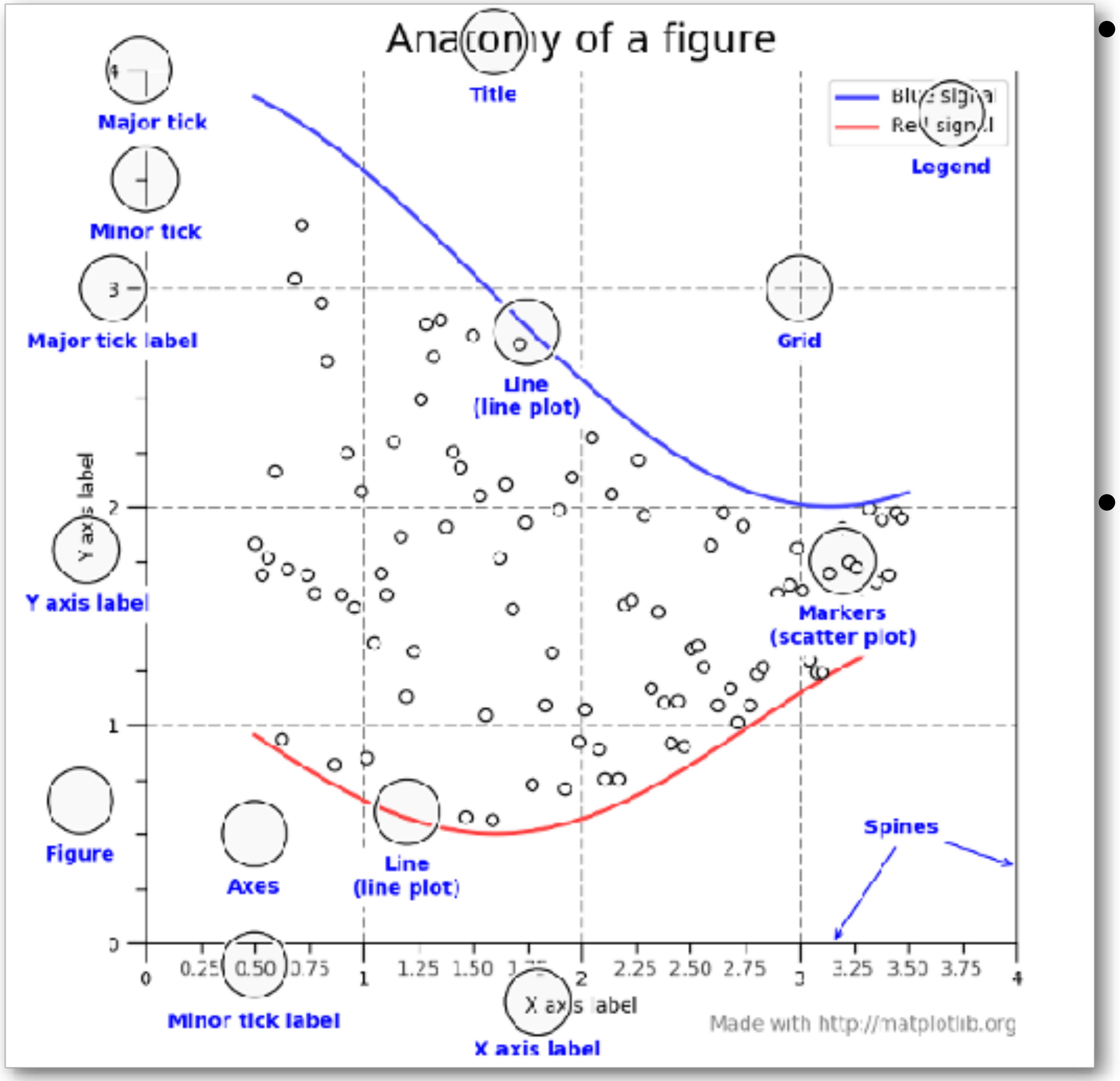
https://matplotlib.org/2.0.0/



- The respective functions are named in an intuitive wa Every Axes object has as methods:
    - .set_xlabel(label)
    - .set_ylabel(label)
    - .set_title(title)
- And axis limits can be set using:

Matplotlib - decorations

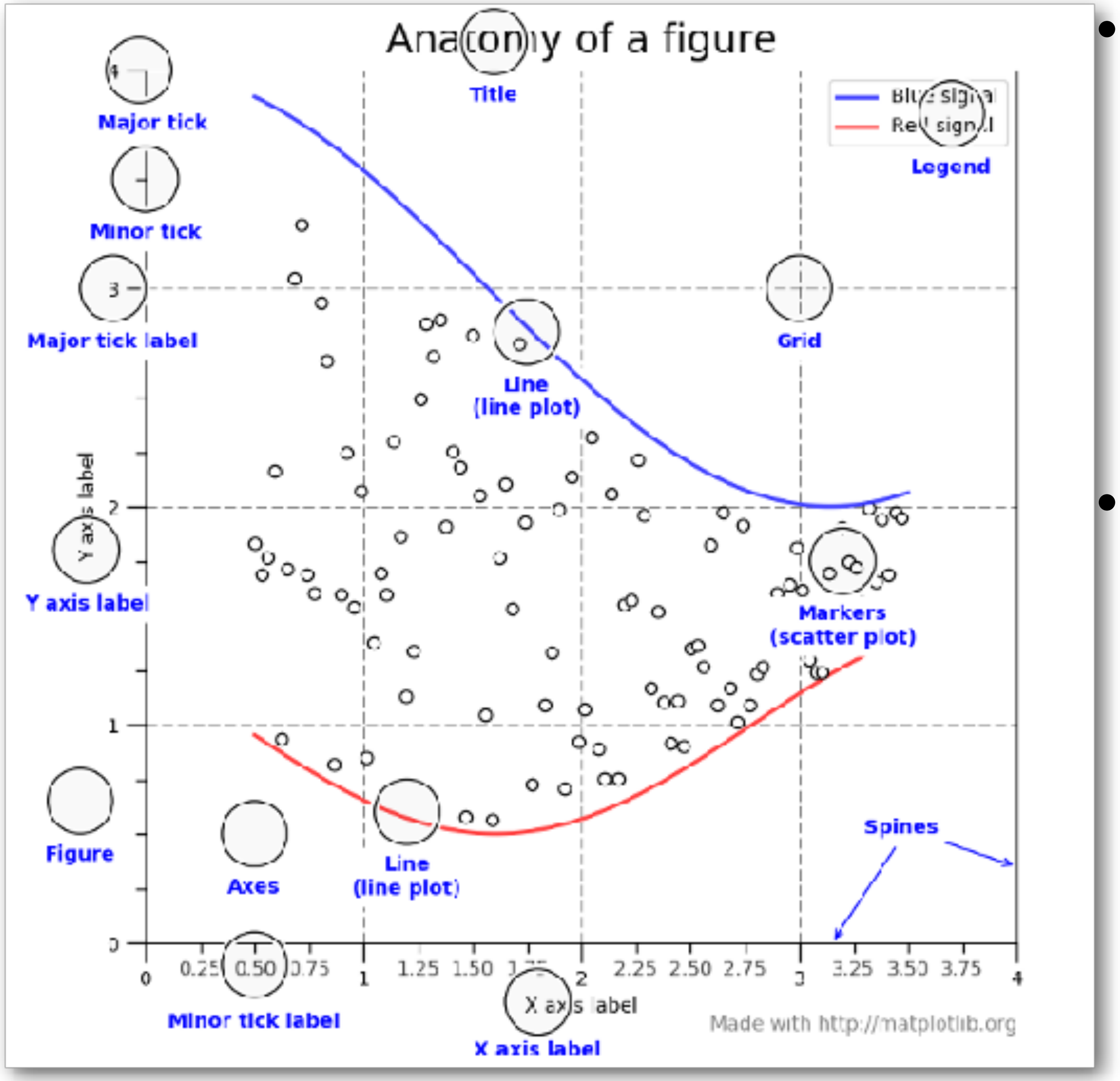
https://matplotlib.org/2.0.0/



- The respective functions are named in an intuitive wa
  Every Axes object has as methods:
  - .set_xlabel(label)
  - .set_ylabel(label)
  - .set_title(title)
- And axis limits can be set using:
  - .set_xlim(xmin, xmax)

Matplotlib - decorations

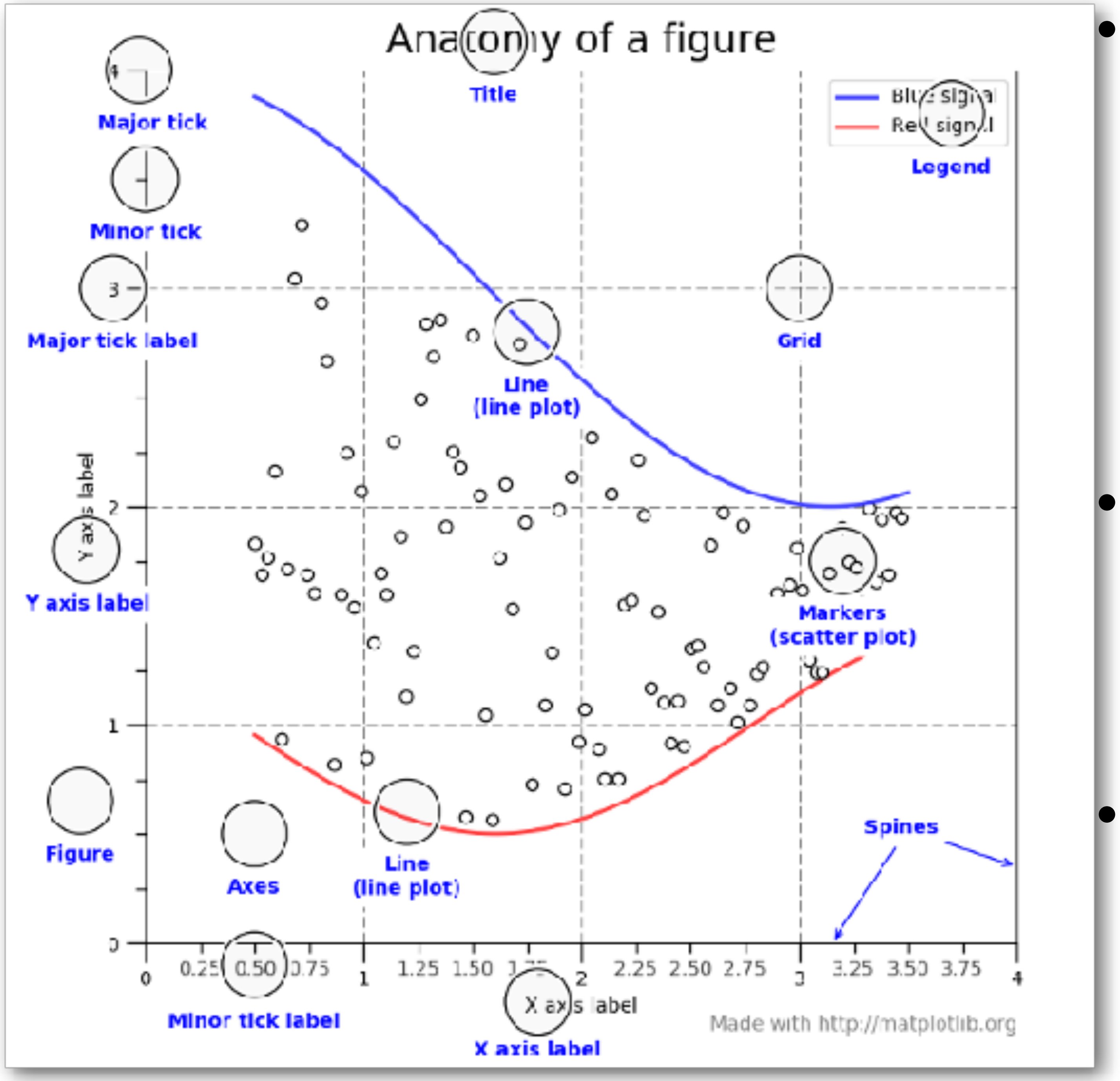
https://matplotlib.org/2.0.0/



- The respective functions are named in an intuitive wa
  Every Axes object has as methods:
    - .set_xlabel(label)
    - .set_ylabel(label)
    - .set_title(title)
- And axis limits can be set using:
    - .set_xlim(xmin, xmax)
    - .set_ylim(ymin, ymax)

Matplotlib - decorations

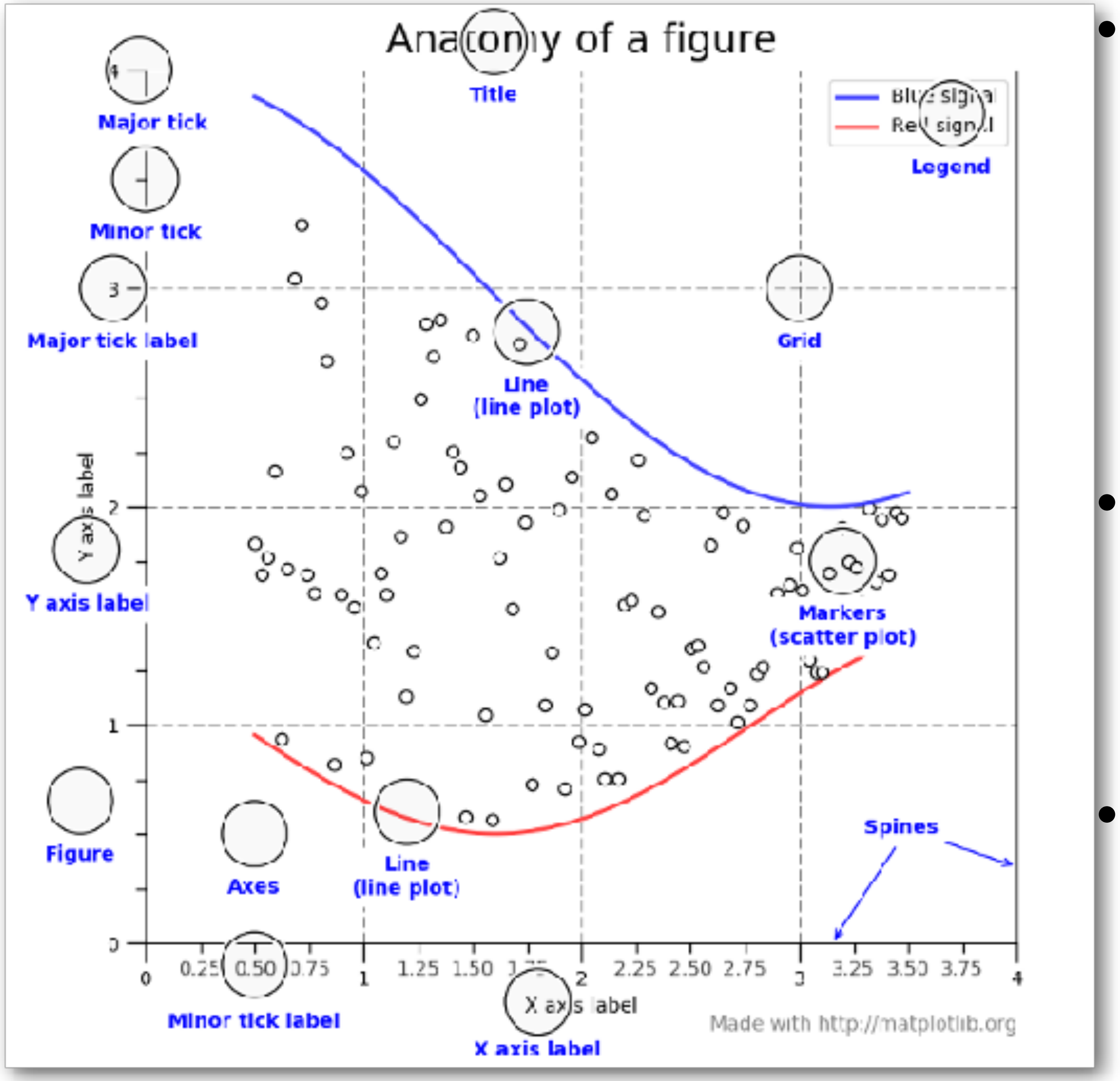
https://matplotlib.org/2.0.0/



- The respective functions are named in an intuitive wa
Every Axes object has as methods:
    - .set_xlabel(label)
    - .set_ylabel(label)
    - .set_title(title)
- And axis limits can be set using:
    - .set_xlim(xmin, xmax)
    - .set_ylim(ymin, ymax)
- Tick marks and labels are set using:

Matplotlib - decorations

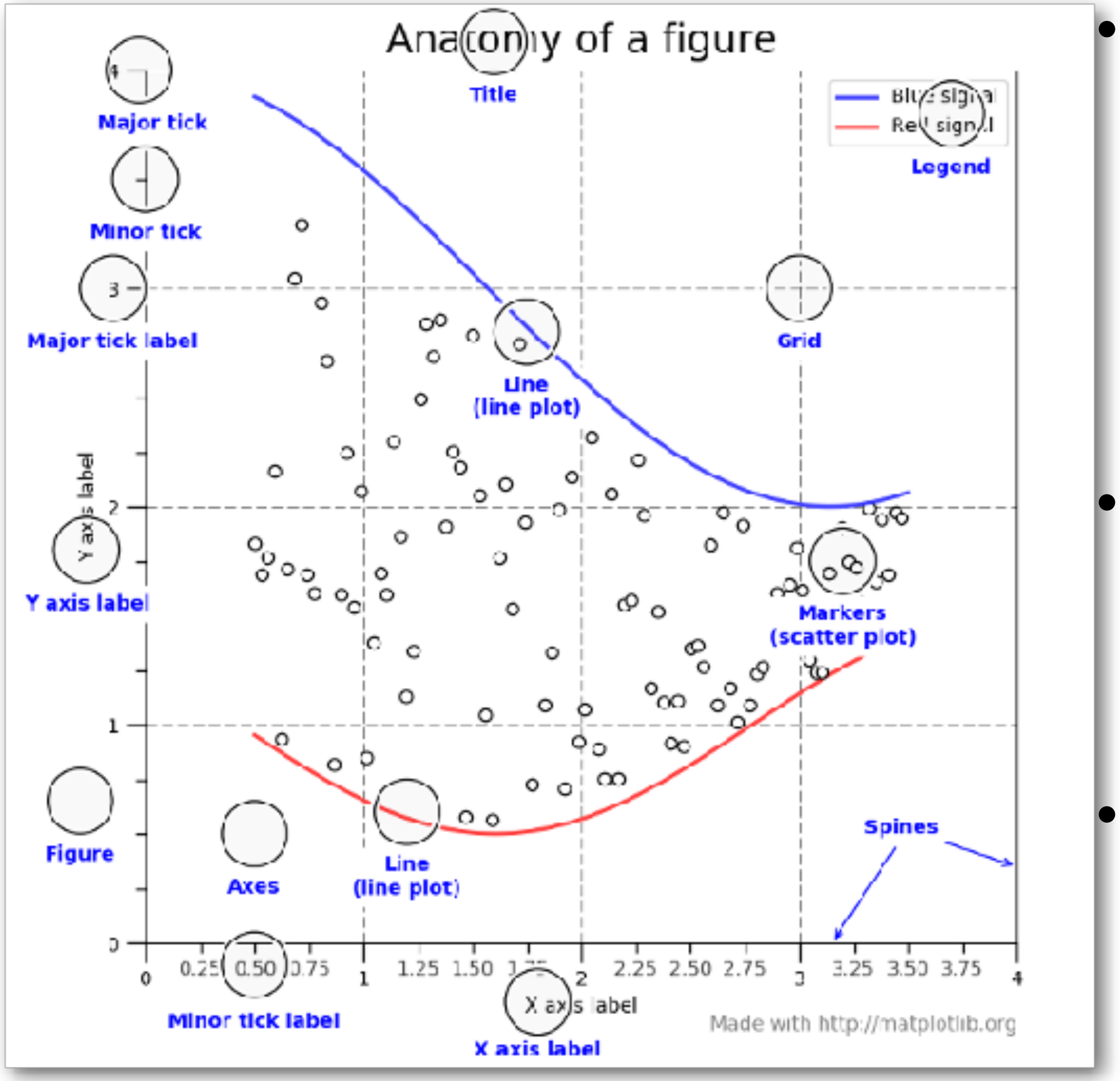
https://matplotlib.org/2.0.0/



- The respective functions are named in an intuitive wa
  Every Axes object has as methods:
  - .set_xlabel(label)
  - .set_ylabel(label)
  - .set_title(title)
- And axis limits can be set using:
  - .set_xlim(xmin, xmax)
  - .set_ylim(ymin, ymax)
- Tick marks and labels are set using:
  - .set_xticks(ticks)/.set_yticks(ticks)

Matplotlib - decorations

https://matplotlib.org/2.0.0/



- The respective functions are named in an intuitive wa Every Axes object has as methods:
  - .set_xlabel(label)
  - .set_ylabel(label)
  - .set_title(title)
- And axis limits can be set using:
  - .set_xlim(xmin, xmax)
  - .set_ylim(ymin, ymax)
- Tick marks and labels are set using:
  - .set_xticks(ticks)/.set_yticks(ticks)
  - .set_xticklabels(labels)/.set_yticklabels(labels)

Matplotlib - Images

https://matplotlib.org/2.0.0/

Matplotlib - Images

https://matplotlib.org/2.0.0/

- .imshow(fig) - Display an image on a set of axes.

Matplotlib - Images

https://matplotlib.org/2.0.0/

- .imshow(fig) - Display an image on a set of axes.

- fig can be any matrix of numbers.

Matplotlib - Images

https://matplotlib.org/2.0.0/

- .imshow(fig) - Display an image on a set of axes.
- fig can be any matrix of numbers.
- Further plotting can occur by simply using the functions described above

https://bmtgoncalves.github.io/DataVisualization/
https://github.com/bmtgoncalves/DataVisualization