Visualization Basics

Intro to Data Visualization

Gaston Sanchez

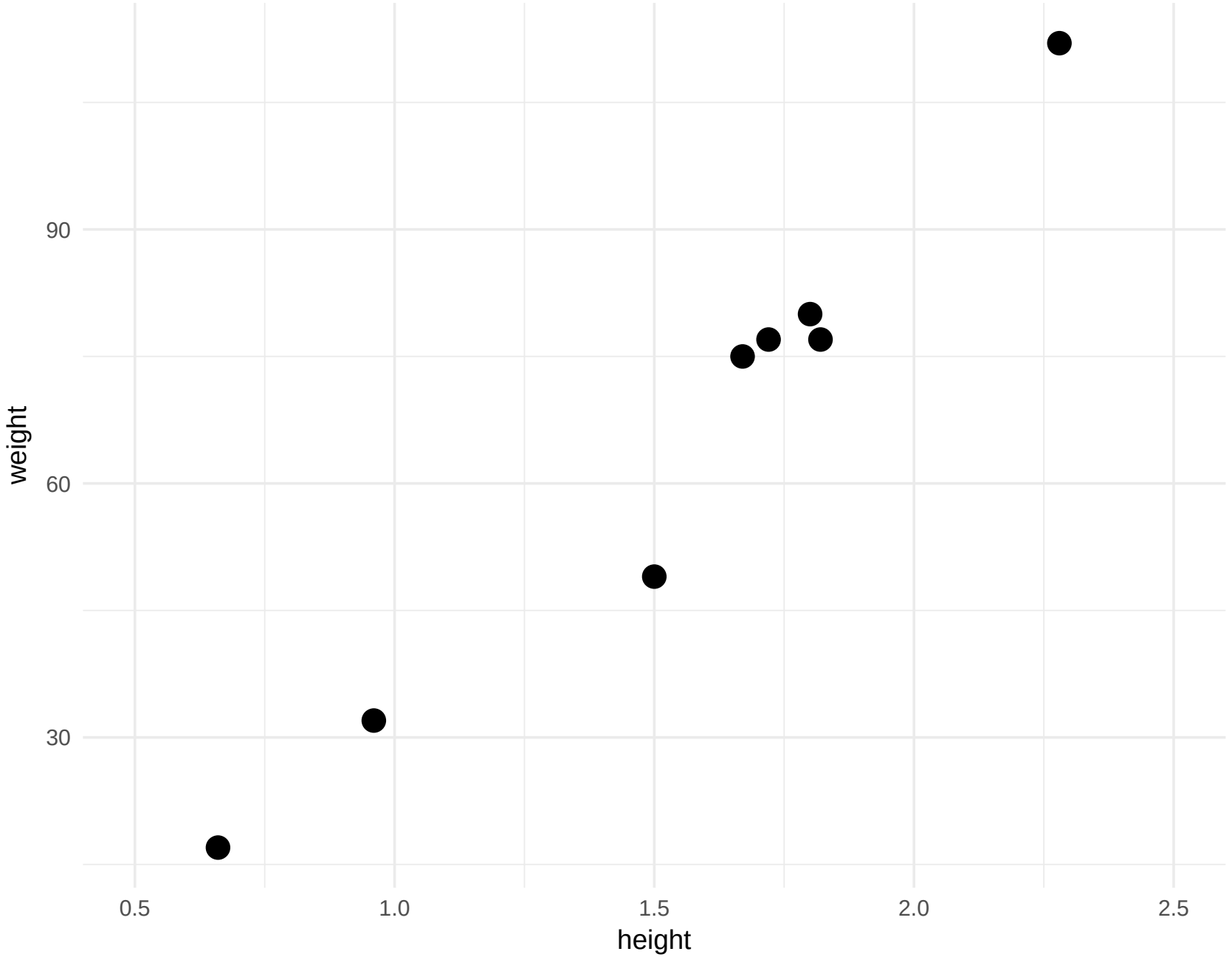
Vision

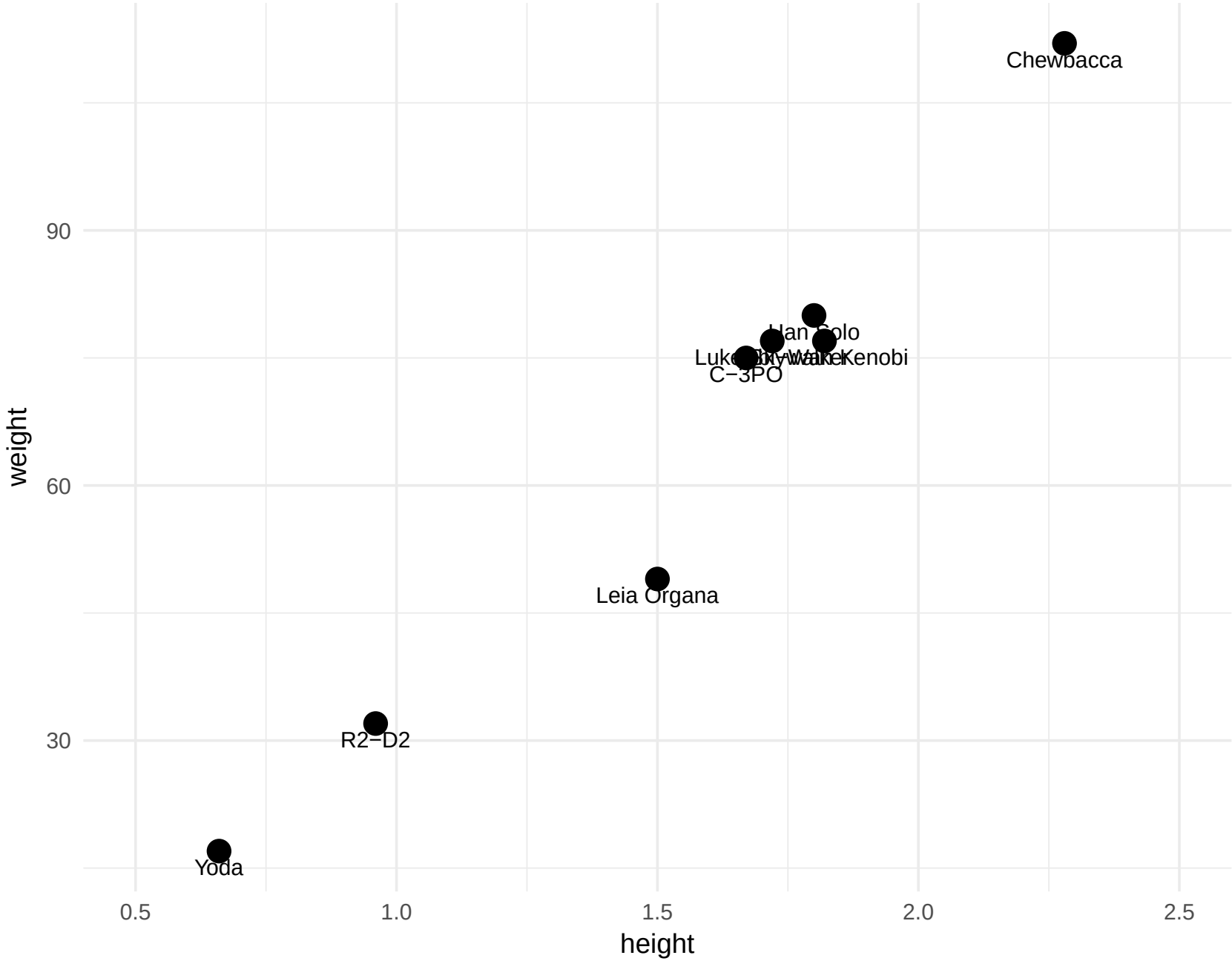
Data Visualization?

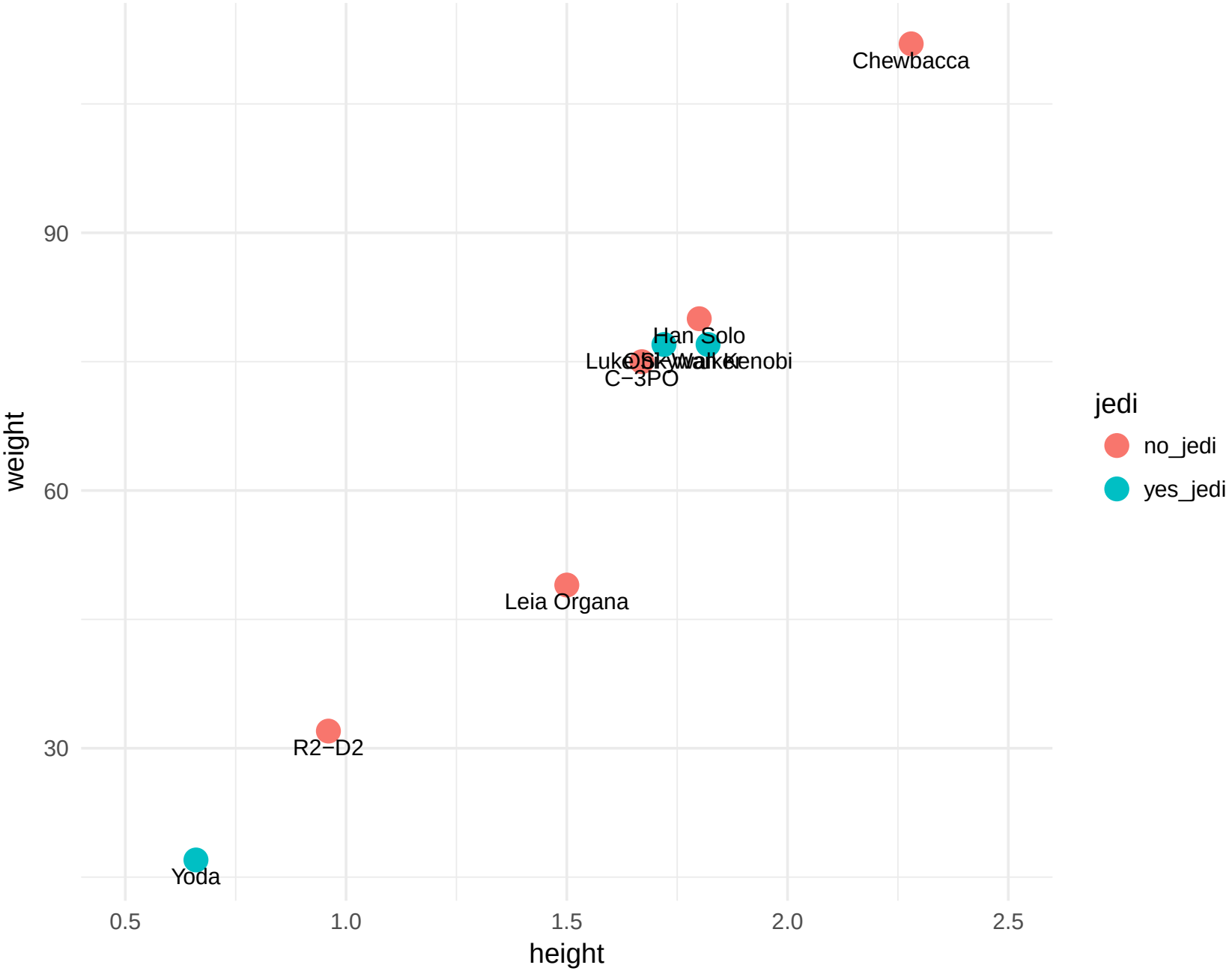
Data visualization is simply **mapping data** to **geometric objects** and their **visual attributes**.

Star Wars data set

	name	gender	height	weight	jedi	species	weapon
1	Luke Skywalker	male	1.72	77	yes_jedi	human	lightsaber
2	Leia Organa	female	1.50	49	no_jedi	human	blaster
3	Obi-Wan Kenobi	male	1.82	77	yes_jedi	human	lightsaber
4	Han Solo	male	1.80	80	no_jedi	human	blaster
5	R2-D2	male	0.96	32	no_jedi	droid	unarmed
6	C-3PO	male	1.67	75	no_jedi	droid	unarmed
7	Yoda	male	0.66	17	yes_jedi	yoda	lightsaber
8	Chewbacca	male	2.28	112	no_jedi	wookiee	bowcaster

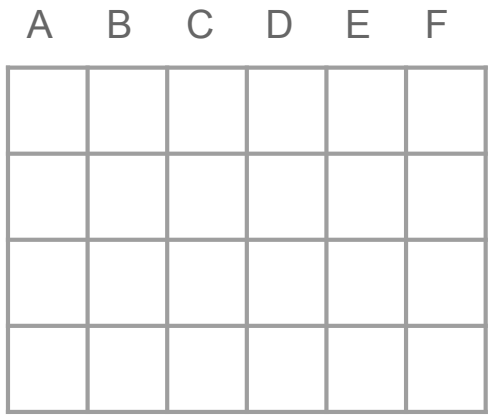

weight
90
60
30
0.5
1.0
1.5
2.0
2.5
height


Chewbacca
Han Solo
Luke Skywalker
Obi-Wan Kenobi
C−3PO
Leia Organa
R2−D2
Yoda
90
60
30
weight
0.5
1.0
1.5
2.0
2.5
height

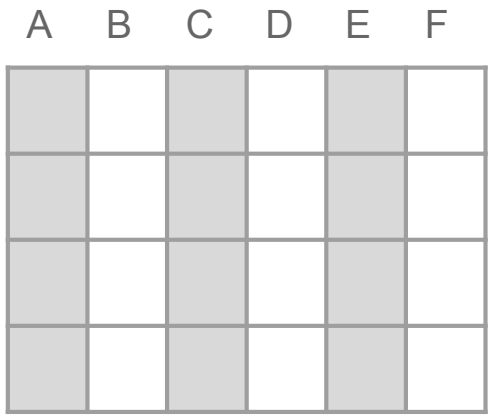

Chewbacca
Han Solo
Luke Skywalker
Obi-Wan Kenobi
C-3PO
Leia Organa
R2-D2
Yoda
weight
90
60
30
0.5
1.0
1.5
2.0
2.5
height
jedi
no_jedi
yes_jedi

How does it (conceptually) work?

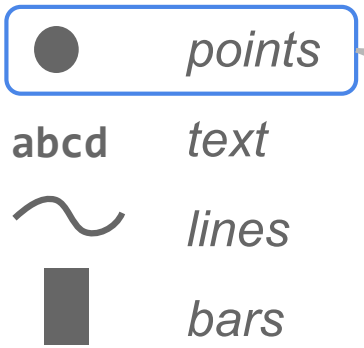
1 Dataset



2 Which variables



3 Which Geometric objects



4 Which visual attributes

- position (coordinates)
- color
- size
- shape

Building a Scatterplot

- **Dataset**: starwars
- **Variables**: height, weight, jedi
- **Geometric objects**: points
- **Visual attributes**:
  - X-axis: height, Y-axis: weight
  - Shape: dots
  - Color: based on jedi categories

Mapping Data

data values

height	weight	jedi
1.72	77	yes_jedi
1.50	49	no_jedi
1.82	77	yes_jedi
1.80	80	no_jedi
0.96	32	no_jedi
1.67	75	no_jedi
0.66	17	yes_jedi
2.28	112	no_jedi

These values are meaningful to us, but not to the computer

mapping

visual attributes

x	y	color
x_1	y_1	#F8766D
x_2	y_2	#00BFC4
x_3	y_3	#F8766D
x_4	y_4	#00BFC4
x_5	y_5	#00BFC4
x_6	y_6	#00BFC4
x_7	y_7	#F8766D
x_8	y_8	#00BFC4

They need to be converted from data units to physical units that the computer can display

Supporting Elements

- Axis labels
- Legends (positions, labels, symbols)
- Choice of colors for points
- Background color (i.e. gray)
- Grid lines (major and minor)
- Axis tick marks

In Summary

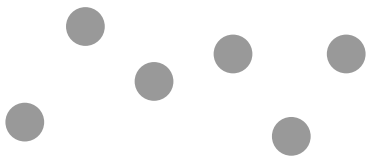
- Graphs consist of several components
- Some components represent quantitative values (e.g. lines, bars, etc.)
- Some represent categorical values (e.g. color, shape, orientation)
- Some play a supporting role (e.g. grid lines, legends, scales on axes)

Geometric Objects and their Visual Attributes

Mapping Fundamentals

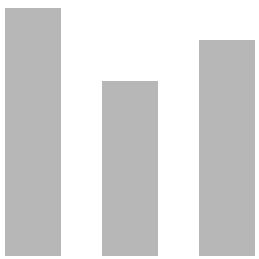
Geometric Objects (primitives)

Points

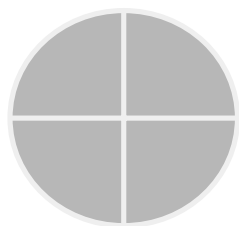


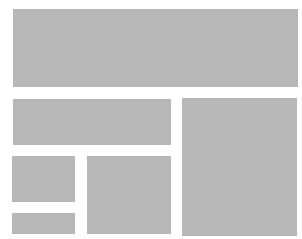
Lines

Bars



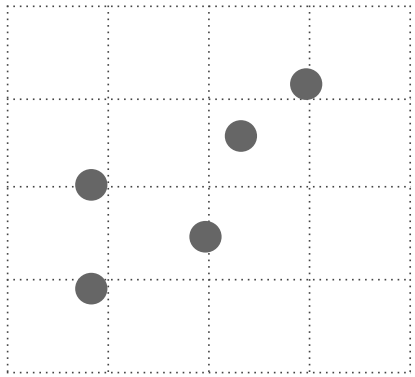
2D Areas / Polygons



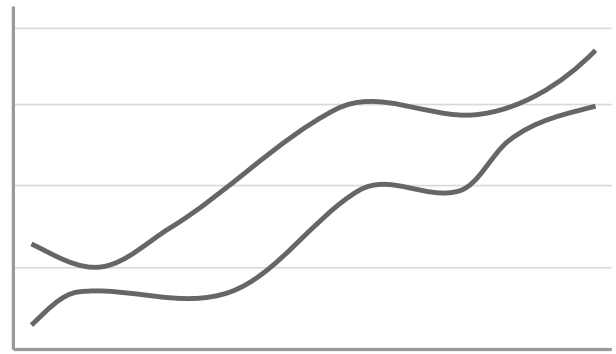


Example of Graphs with Geometric Objects

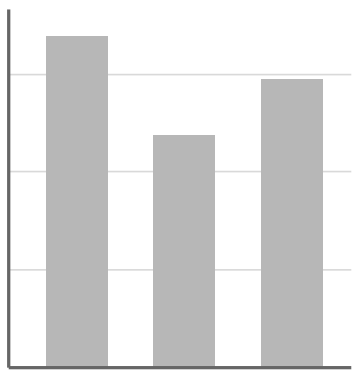
Points: e.g. scatterplot



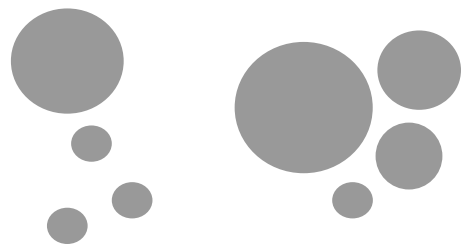
Lines: e.g. timeline



Bars: e.g. bar chart



2D-areas / Polygons: e.g. densities



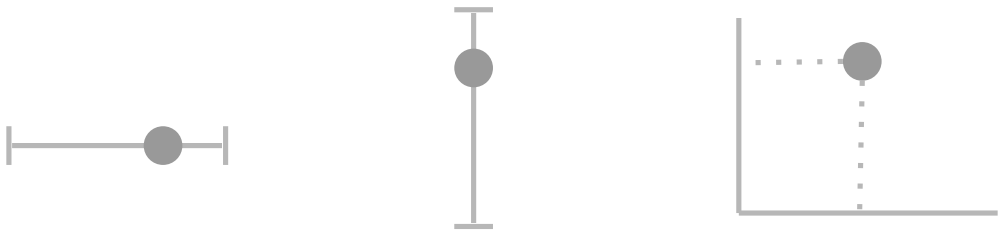
Geometric objects

Graphical objects (typically) used to encode quantitative values

- Points
- Lines
- Bars
- 2D areas and polygons

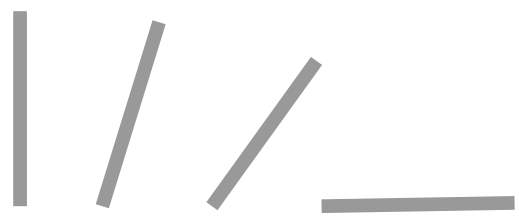
Visual Attributes

Position



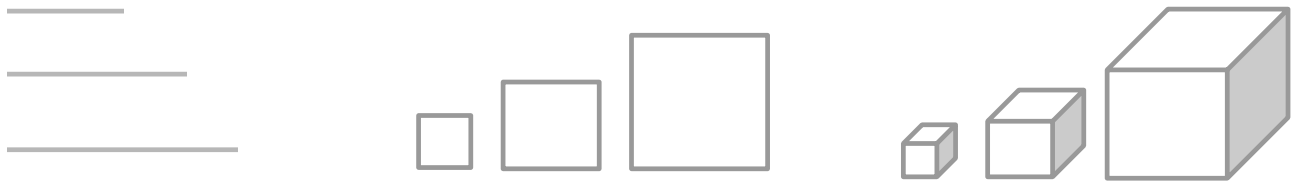
Shape

Orientation (tilt)



Visual Attributes

Size



length area volumne

Color Luminance

Color Hue

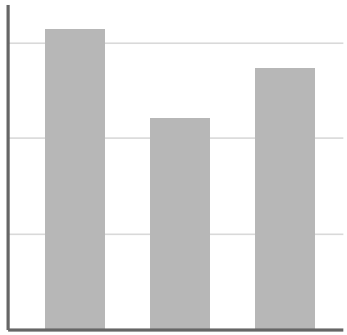
Color Saturation

Visual Attributes of Geometric objects

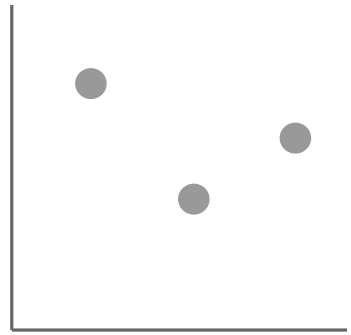
Used to encode both quantitative and categorical

- Position
- Color
- Size
- Shape
- Fill pattern
- Border
- Line style

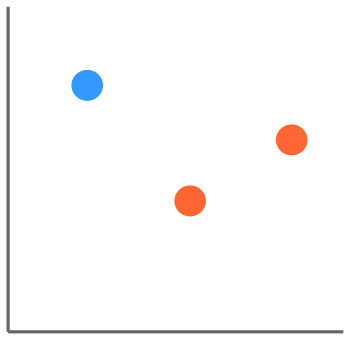
Examples of Visual Attributes



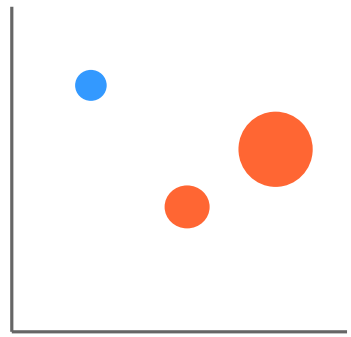
Vertical position



Vertical position
Horizontal position



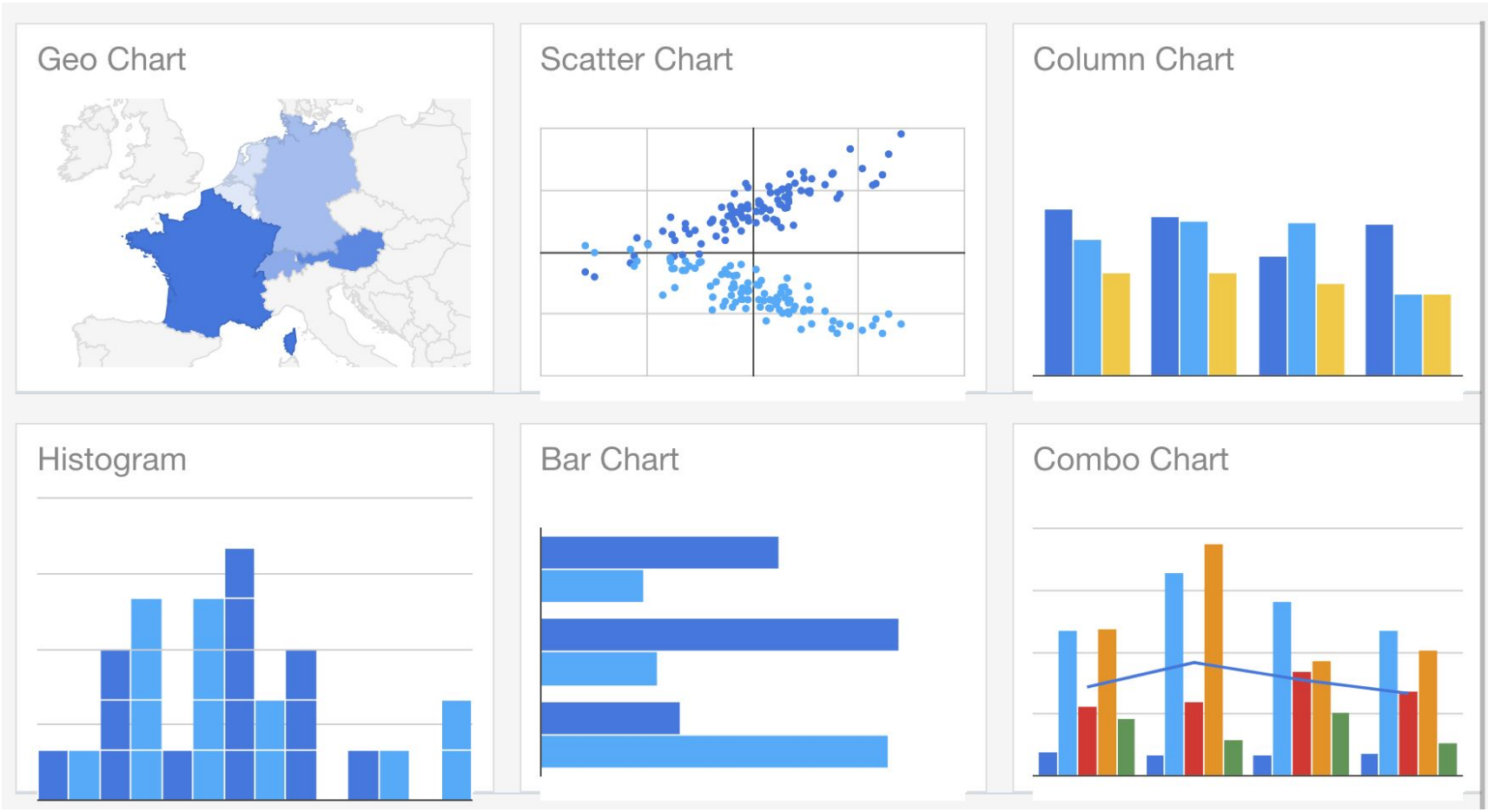
Vertical position
Horizontal position
Color hue



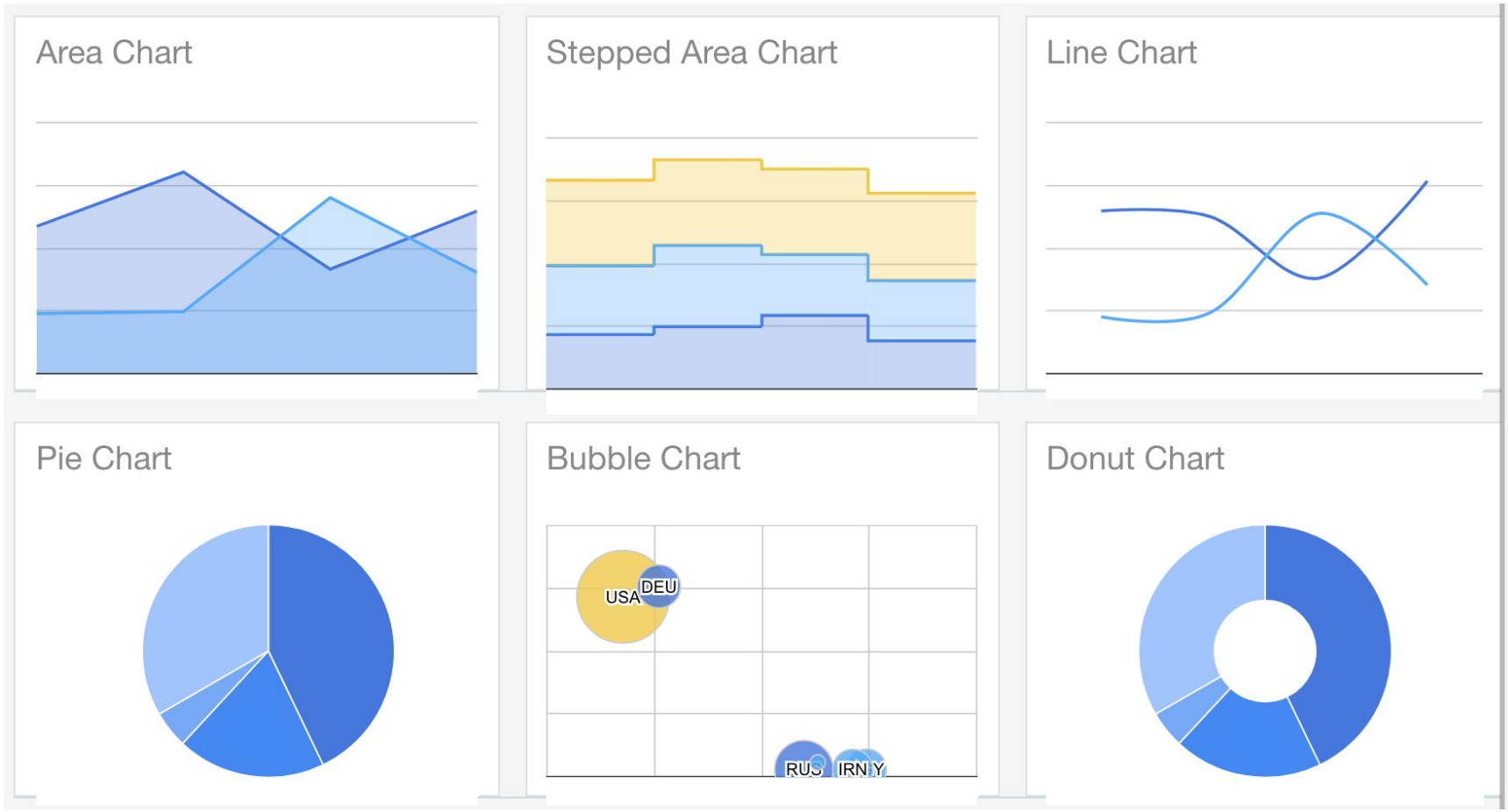
Vertical position
Horizontal position
Color hue
Size (area)

Gallery of Charts
(off-the-self examples)

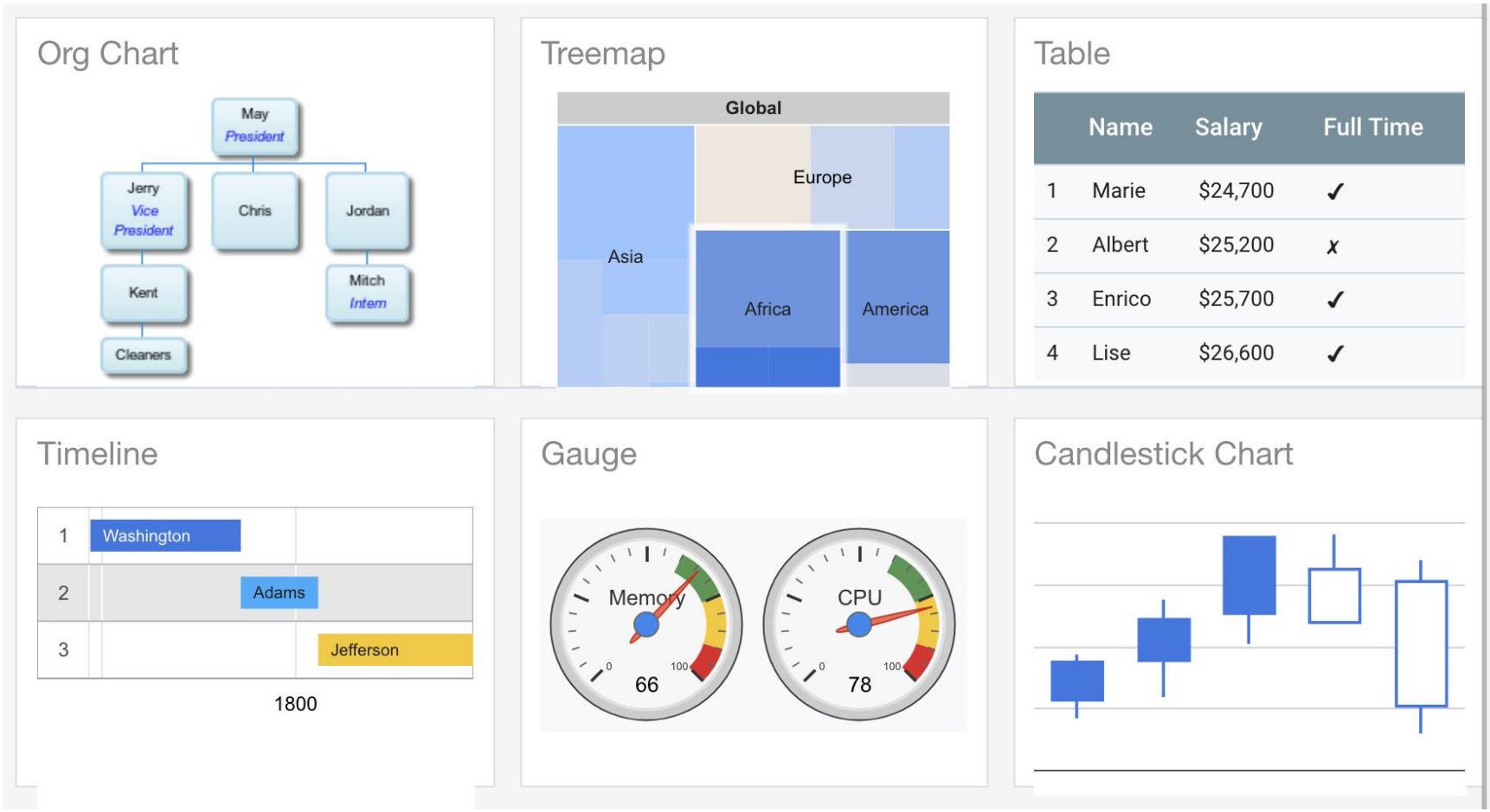
Examples from Google Charts



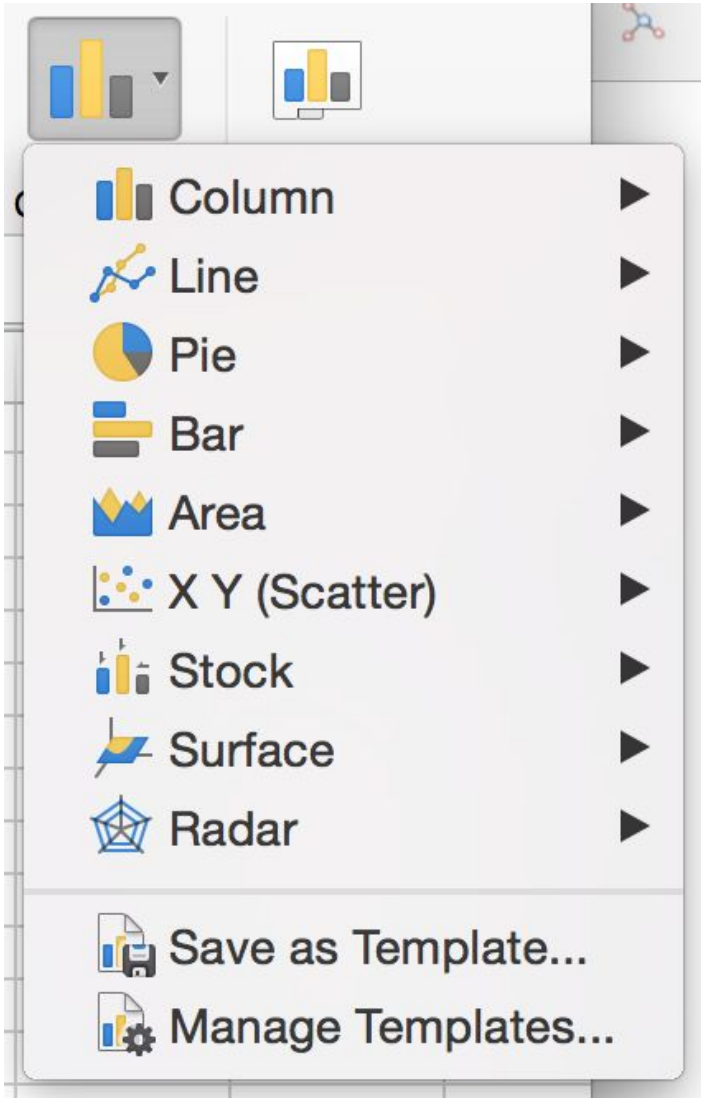
Examples from Google Charts



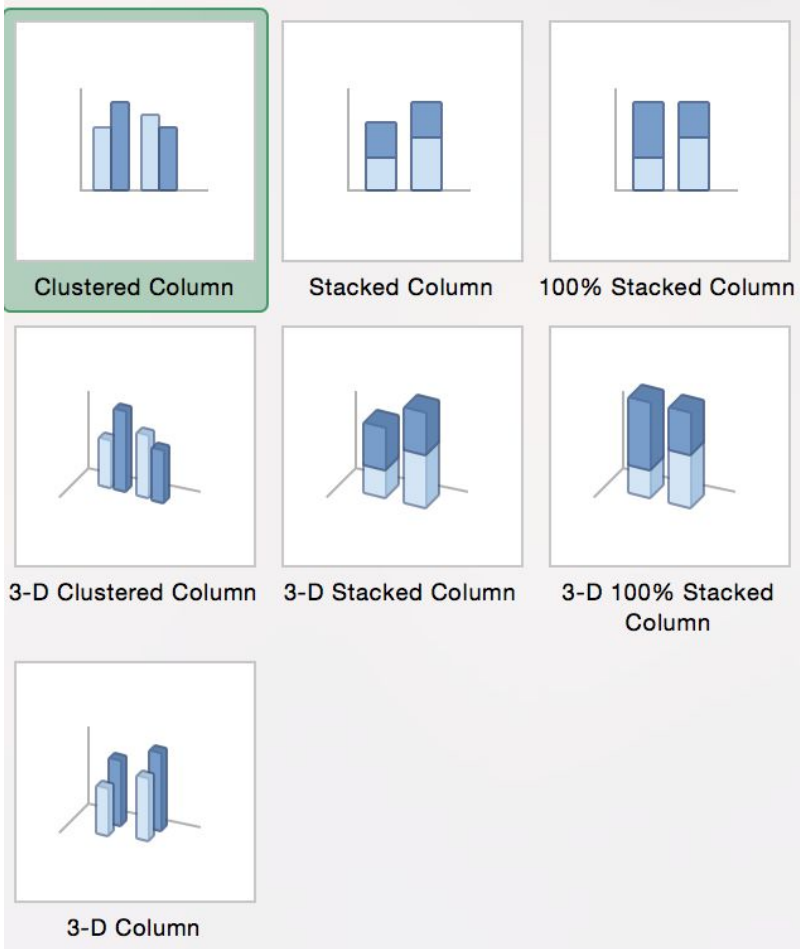
Examples from Google Charts

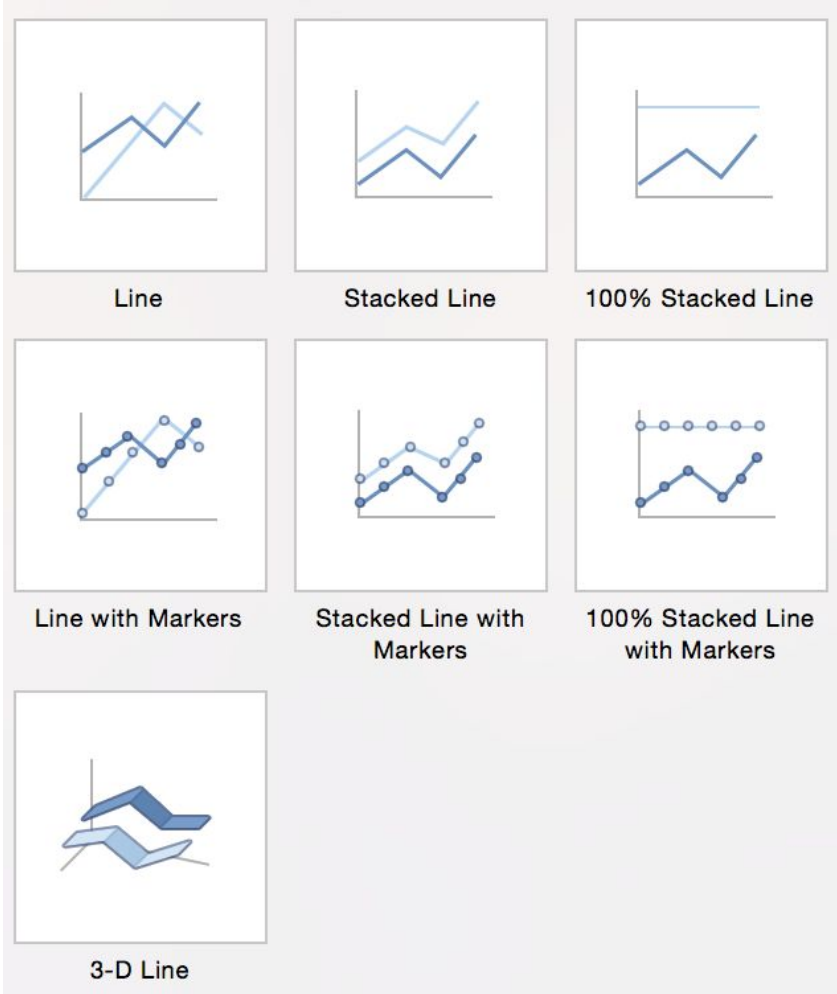


Examples from MS Excel

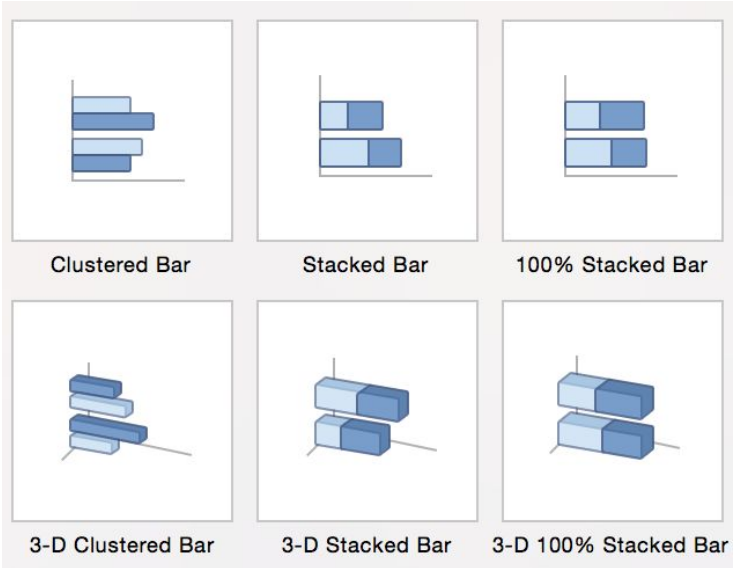


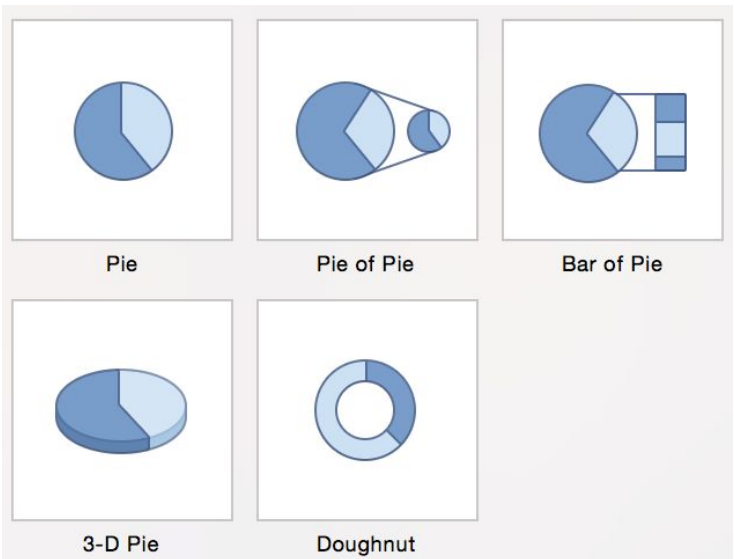
Examples from MS Excel

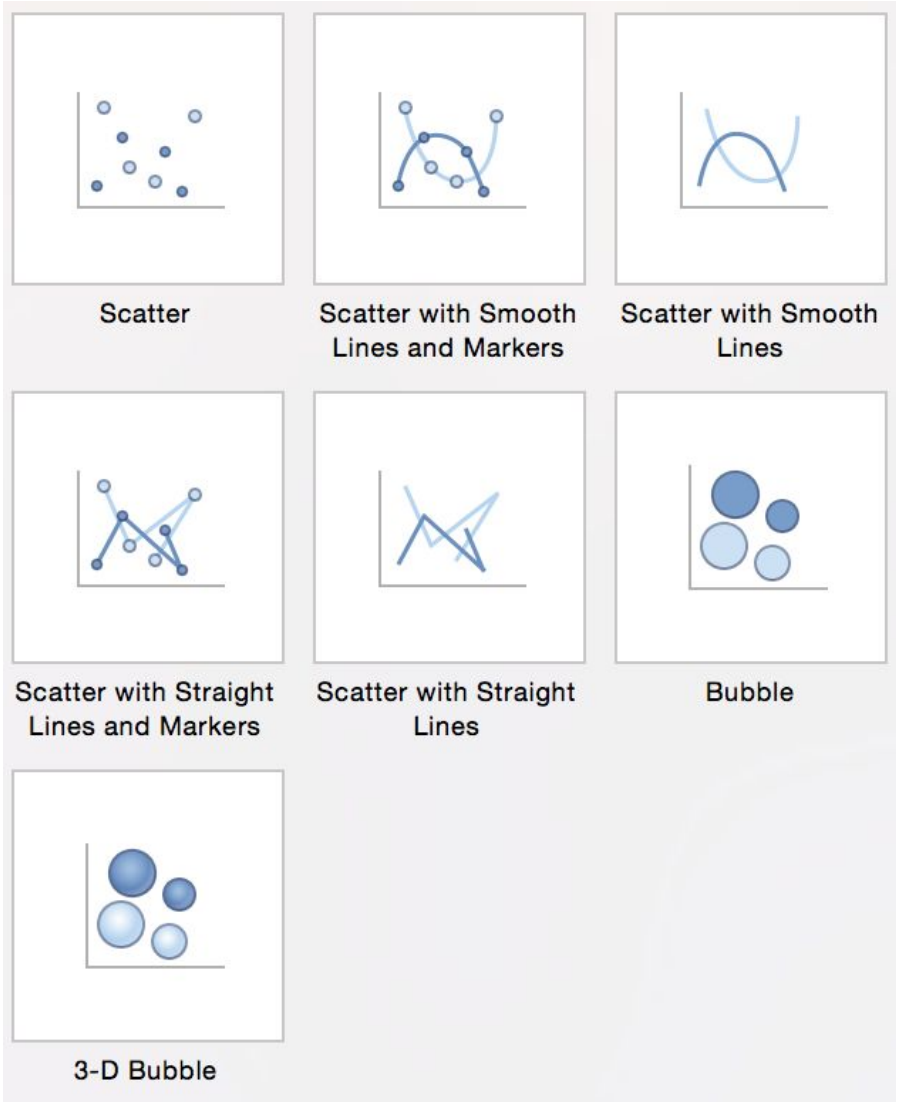




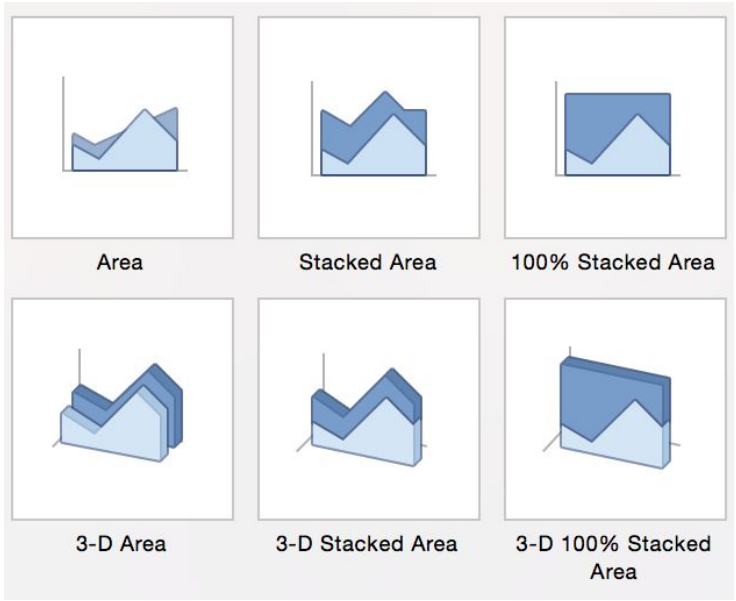
Examples from MS Excel

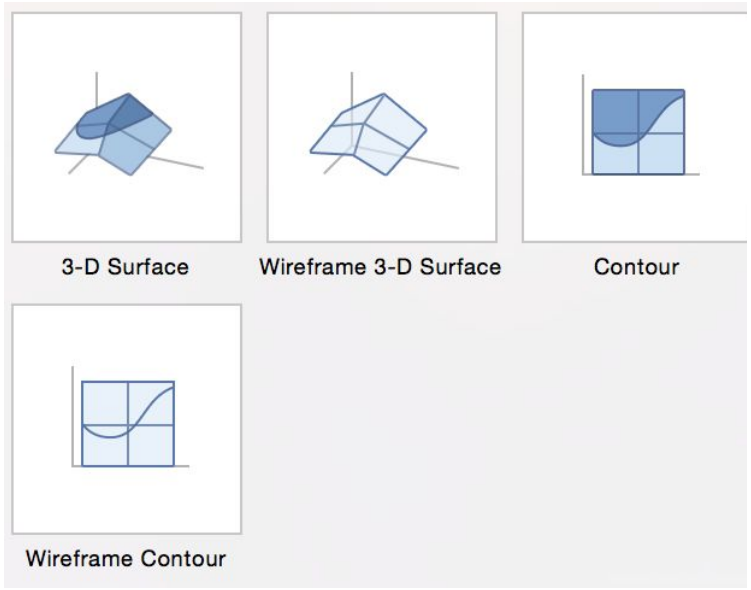


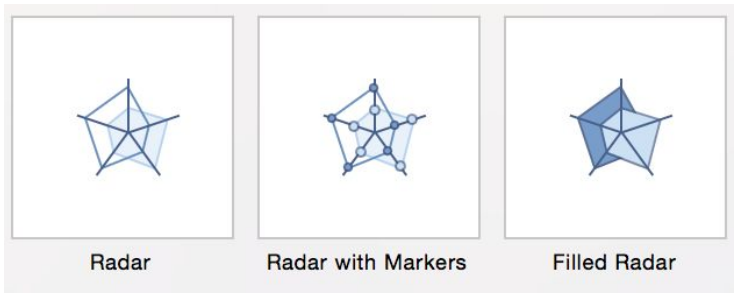


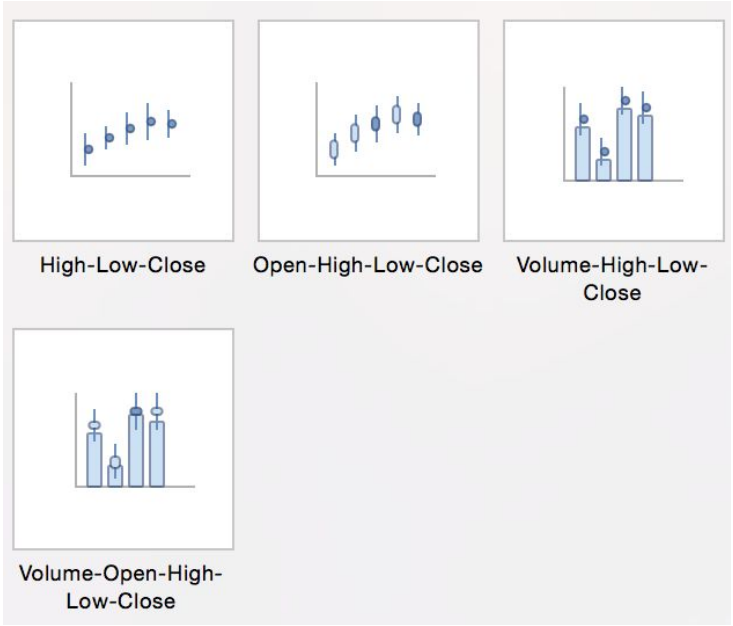


Examples from MS Excel









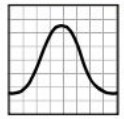
Examples from ggplot2

One Variable

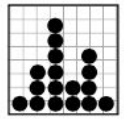
Continuous

c <- ggplot(mpg, aes(hwy)); c2 <- ggplot(mpg)

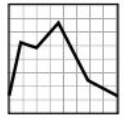
c + geom_area(stat = "bin")
x, y, alpha, color, fill, linetype, size



c + geom_density(kernel = "gaussian"**)**
x, y, alpha, color, fill, group, linetype, size, weight

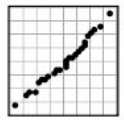


c + geom_dotplot()
x, y, alpha, color, fill



c + geom_freqpoly()
x, y, alpha, color, group, linetype, size

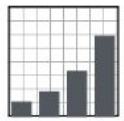
c + geom_histogram(binwidth = 5**)**
x, y, alpha, color, fill, linetype, size, weight



c2 + geom_qq(aes(sample = hwy)**)**
x, y, alpha, color, fill, linetype, size, weight

Discrete

d <- ggplot(mpg, aes(fl))



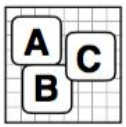
d + geom_bar()
x, alpha, color, fill, linetype, size, weight

Examples from ggplot2

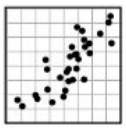
Two Variables

Continuous X, Continuous Y

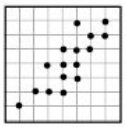
e <- ggplot(mpg, aes(cty, hwy))



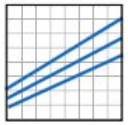
e + **geom_label(**aes(label = cty), nudge_x = 1, nudge_y = 1, check_overlap = TRUE**)**
x, y, label, alpha, angle, color, family, fontface, hjust, lineheight, size, vjust



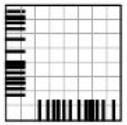
e + **geom_jitter(**height = 2, width = 2**)**
x, y, alpha, color, fill, shape, size



e + **geom_point()**
x, y, alpha, color, fill, shape, size, stroke

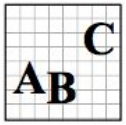


e + **geom_quantile()**
x, y, alpha, color, group, linetype, size, weight



e + **geom_rug(**sides = "bl"**)**
x, y, alpha, color, linetype, size

e + **geom_smooth(**method = lm**)**
x, y, alpha, color, fill, group, linetype, size, weight



e + **geom_text(**aes(label = cty), nudge_x = 1, nudge_y = 1, check_overlap = TRUE**)**
x, y, label, alpha, angle, color, family, fontface,

Continuous Bivariate Distribution

h <- ggplot(diamonds, aes(carat, price))

h + **geom_bin2d(**binwidth = c(0.25, 500)**)**
x, y, alpha, color, fill, linetype, size, weight

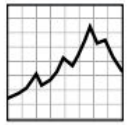
h + **geom_density2d()**
x, y, alpha, colour, group, linetype, size

h + **geom_hex()**
x, y, alpha, colour, fill, size

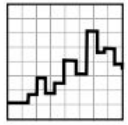
Continuous Function

i <- ggplot(economics, aes(date, unemploy))

i + **geom_area()**
x, y, alpha, color, fill, linetype, size



i + **geom_line()**
x, y, alpha, color, group, linetype, size



i + **geom_step(**direction = "hv"**)**
x, y, alpha, color, group, linetype, size

So how do you approach graphing data?

Creating graphs . . .

With computer technology, anyone can create graphics, but few of us know how to do it well.

Donna Wong

Approaching graphing data

With so many chart options, and various software tools, how can you determine what type of graph should you use?

In my opinion, there are a couple of aspects to always keep in mind:

- Data encoding (core idea)
- Common analytical tasks
- Visual perception basics
- Effective charts suggestions