Data Visualization 2: Encoding data in graphs

Stat 133 with Gaston Sanchez

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

Gallery of Charts

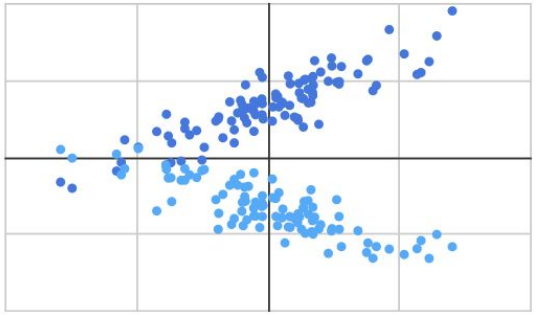
(off-the-self examples)

Examples of charts in Google Sheets

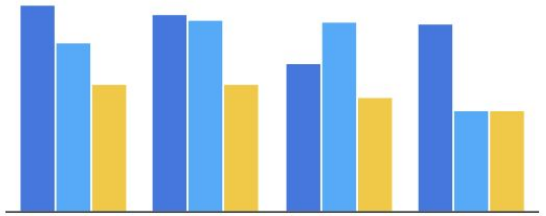
Sample options (from Google Charts)

Geo Chart

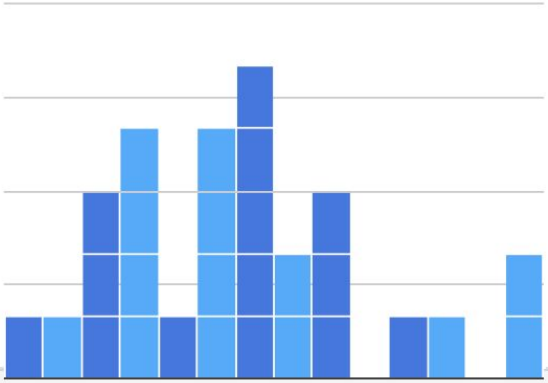
Scatter Chart



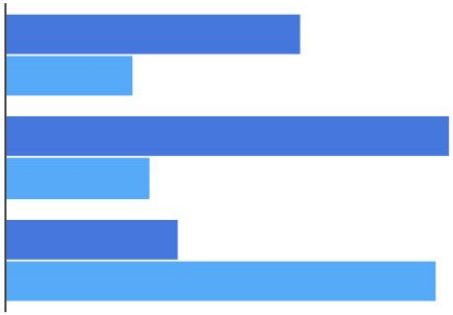
Column Chart



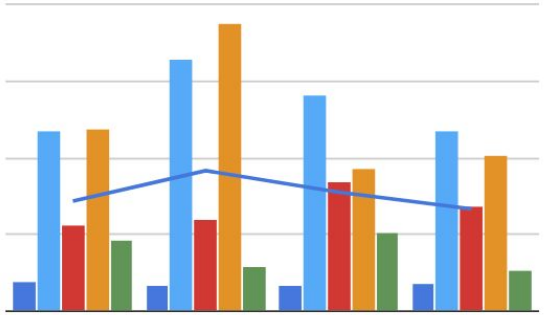
Histogram



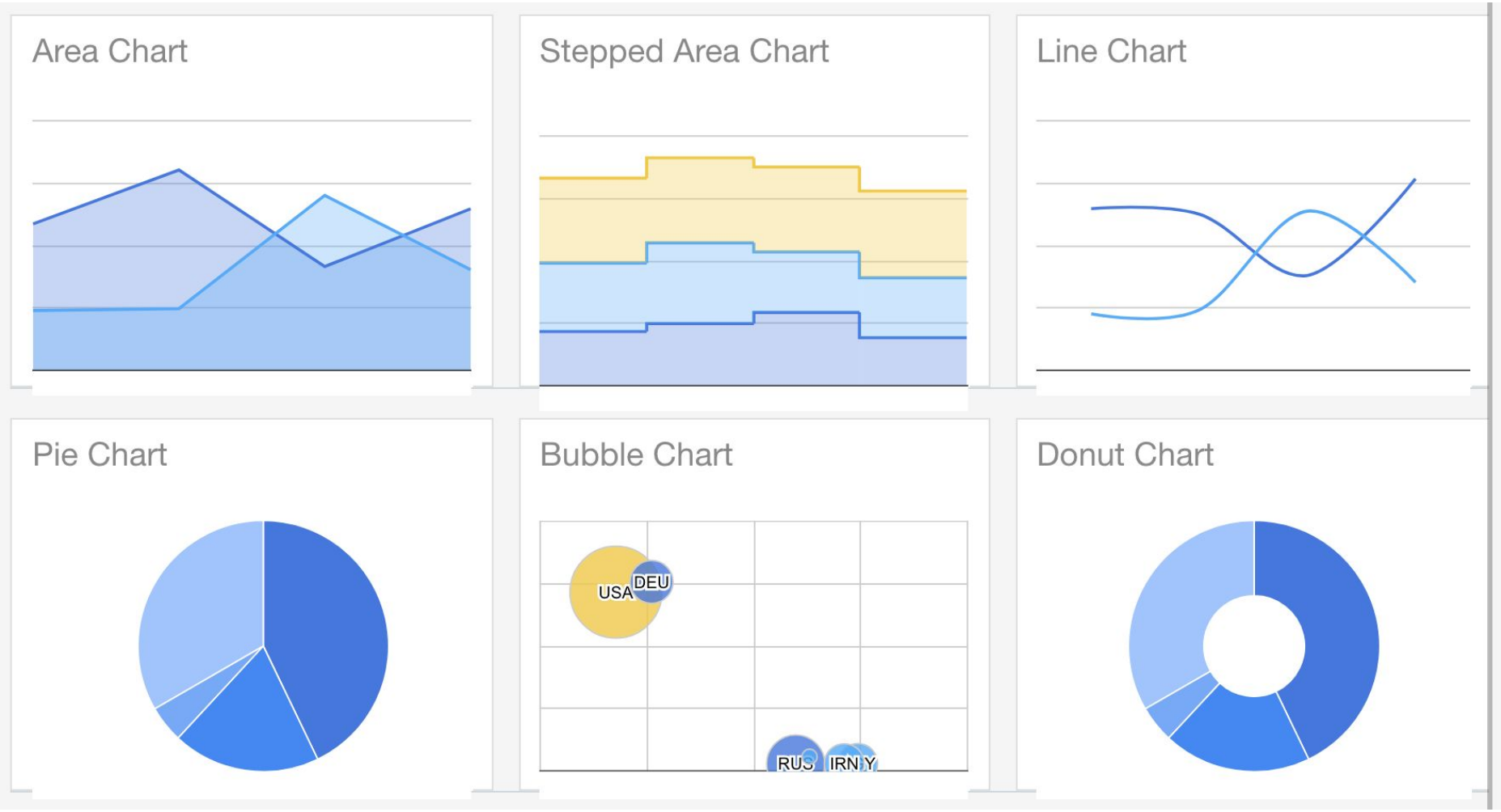
Bar Chart



Combo Chart

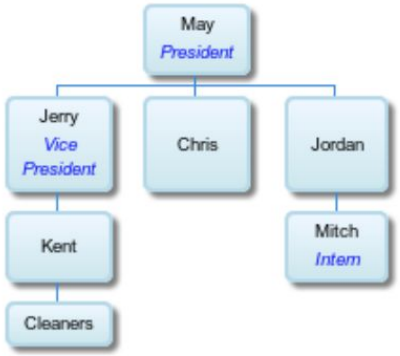


Sample options (from Google Charts)

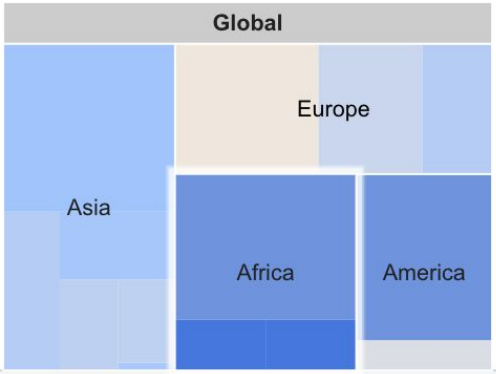


Sample options (from Google Charts)

Org Chart



Treemap

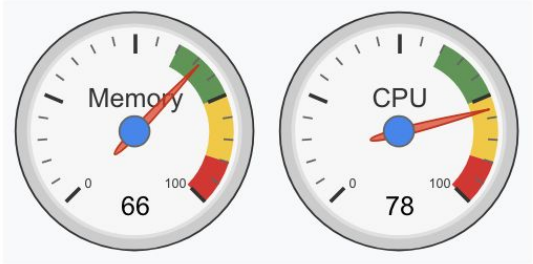


Table

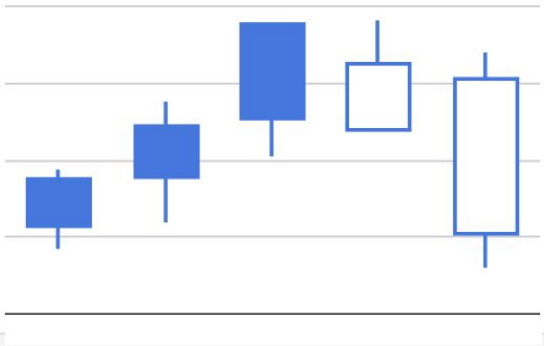
	Name	Salary	Full Time
1	Marie	$24,700	✓
2	Albert	$25,200	✗
3	Enrico	$25,700	✓
4	Lise	$26,600	✓

Timeline

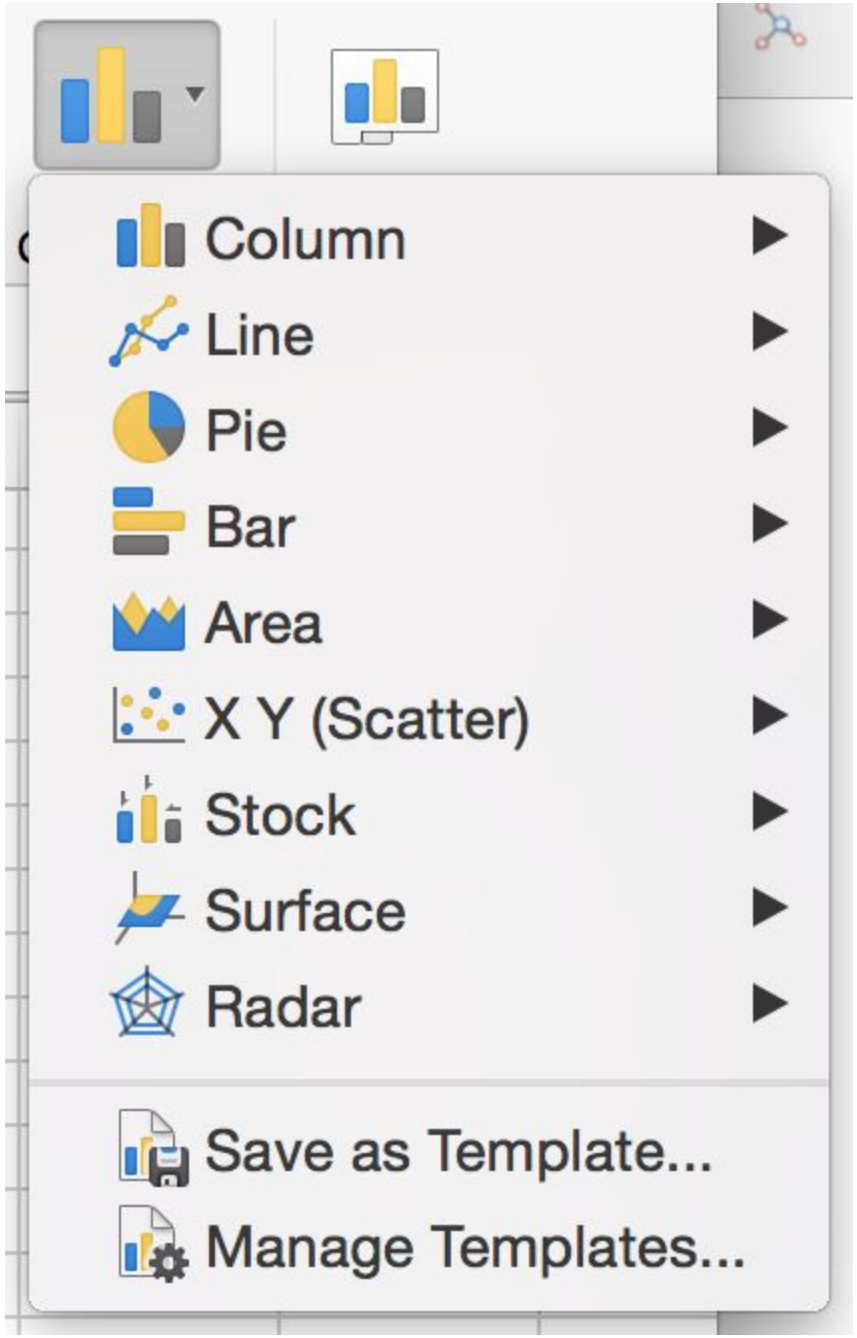
Gauge



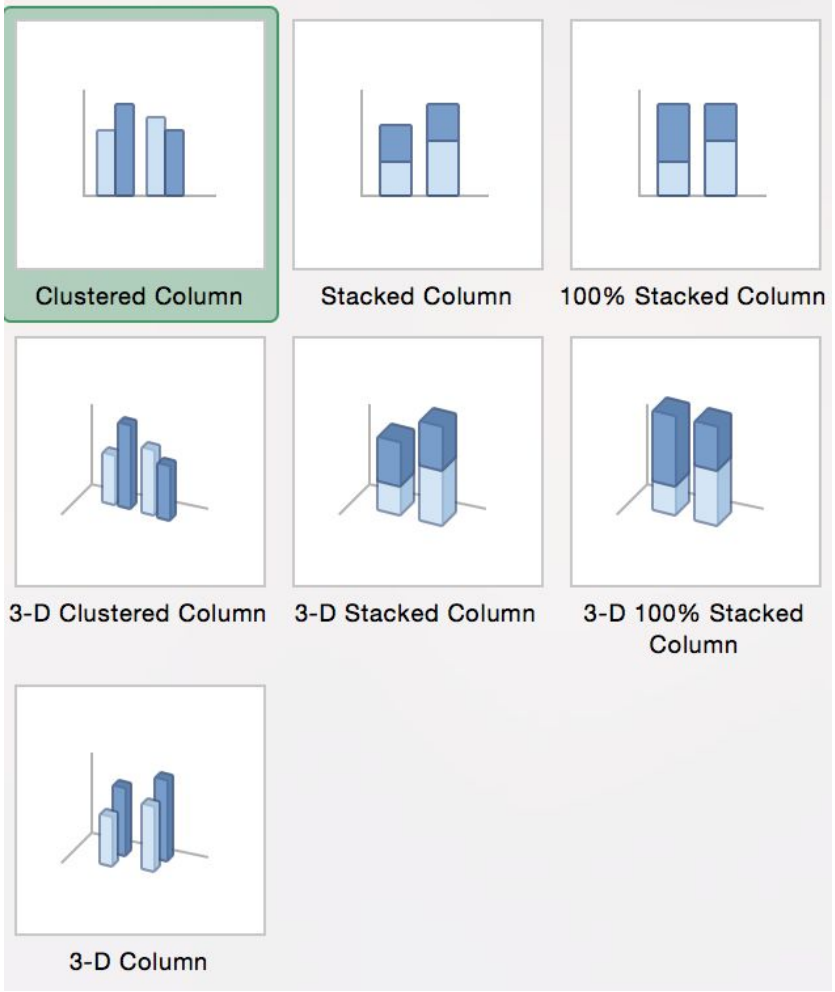
Candlestick Chart



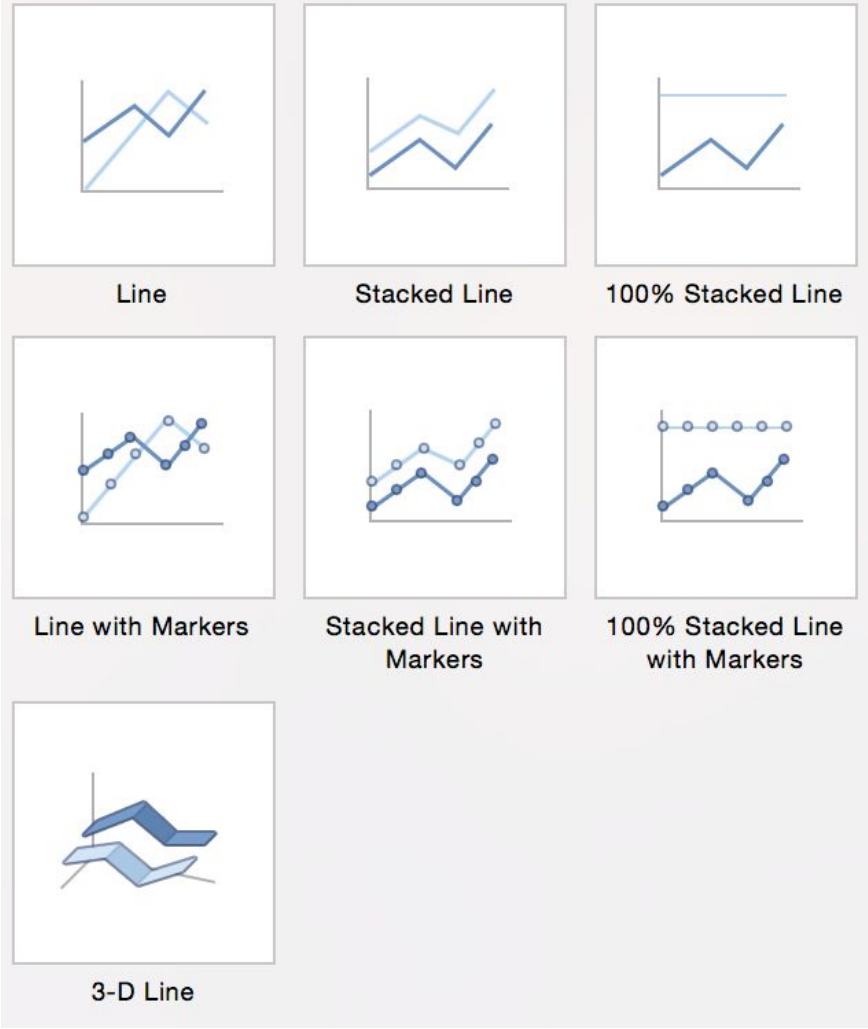
Examples of Charts in Excel


Column
Line
Pie
Bar
Area
X Y (Scatter)
Stock
Surface
Radar
Save as Template...
Manage Templates...

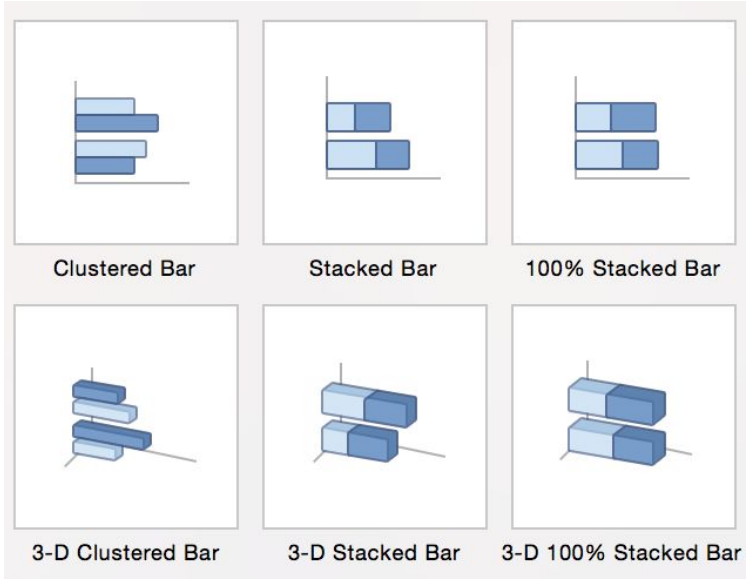
Column

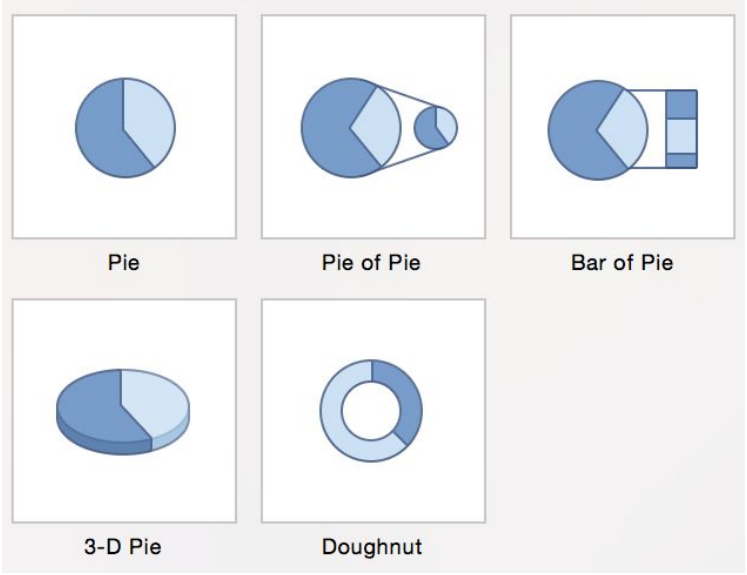

Clustered Column
Stacked Column
100% Stacked Column
3-D Clustered Column
3-D Stacked Column
3-D 100% Stacked Column
3-D Column

Line

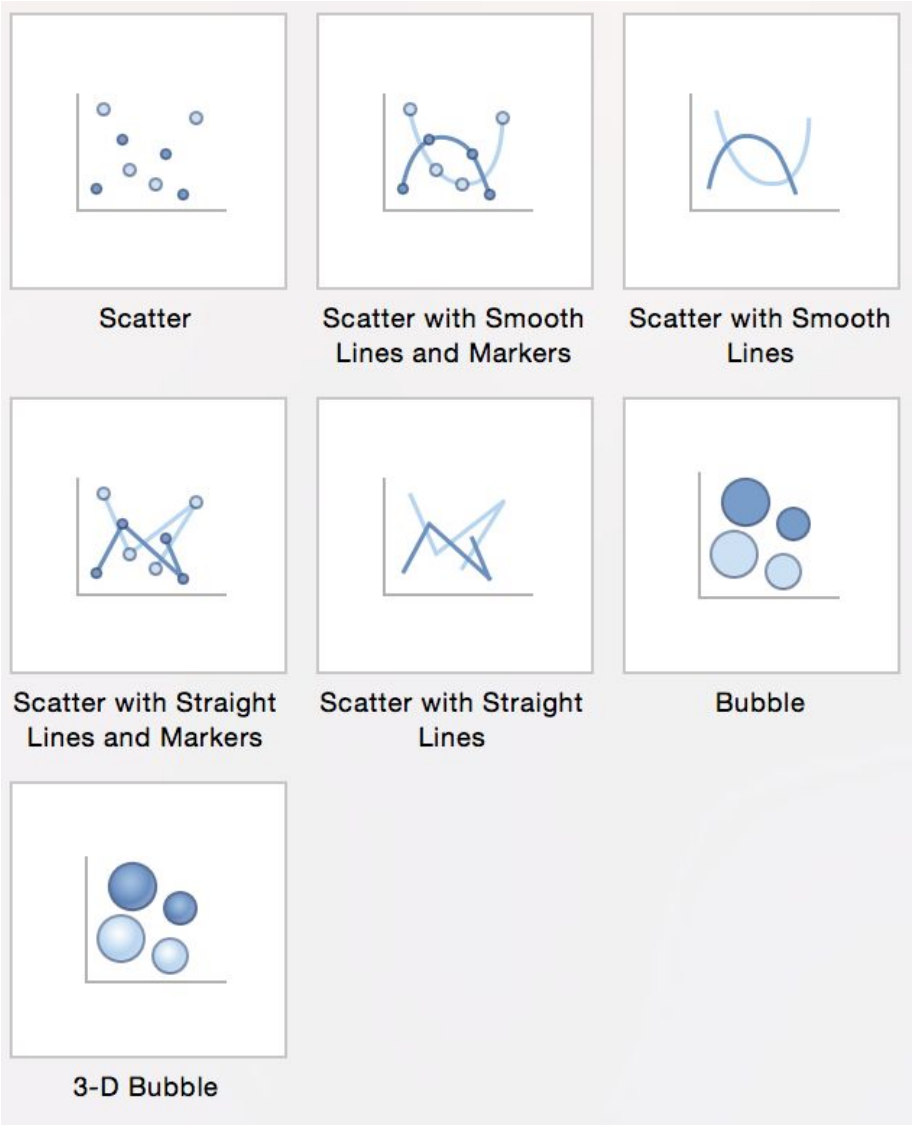

Line
Stacked Line
100% Stacked Line
Line with Markers
Stacked Line with Markers
100% Stacked Line with Markers
3-D Line

Bar and Pie



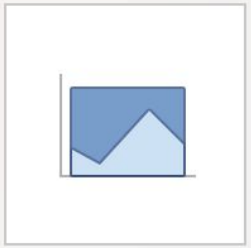


X Y scatter



Area

Stacked Area

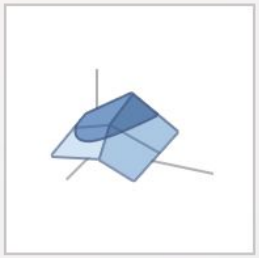


100% Stacked Area

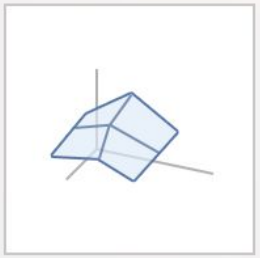
3-D Area

3-D Stacked Area

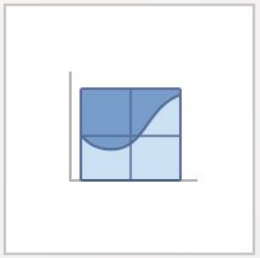
3-D 100% Stacked Area



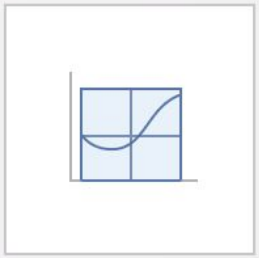
3-D Surface



Wireframe 3-D Surface



Contour

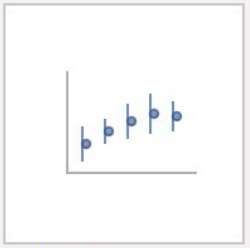


Wireframe Contour

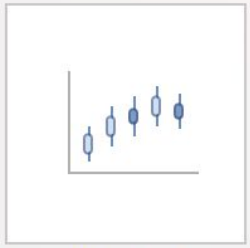
Radar

Radar with Markers

Filled Radar



High-Low-Close



Open-High-Low-Close

Volume-High-Low-Close

Volume-Open-High-Low-Close

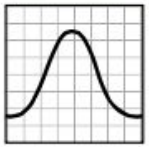
Examples of “ggplot2” charts in R

One Variable

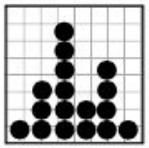
Continuous

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

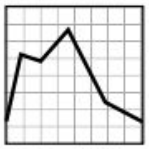
c + geom_area(stat = "bin")
x, y, alpha, color, fill, linetype, size



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

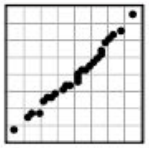


c + geom_dotplot()
x, y, alpha, color, fill



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

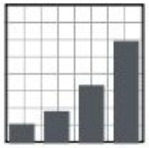
c + geom_histogram(binwidth = 5**)**
x, y, alpha, color, fill, linetype, size, weight



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

Discrete

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

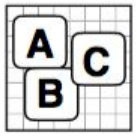


d + geom_bar()
x, alpha, color, fill, linetype, size, weight

Two Variables

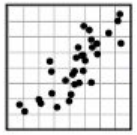
Continuous X, Continuous Y

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



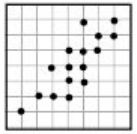
e + geom_label(aes(label = cty), nudge_x = 1, nudge_y = 1, check_overlap = TRUE**)**

x, y, label, alpha, angle, color, family, fontface, hjust, lineheight, size, vjust



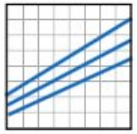
e + geom_jitter(height = 2, width = 2**)**

x, y, alpha, color, fill, shape, size



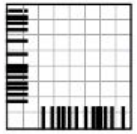
e + geom_point()

x, y, alpha, color, fill, shape, size, stroke



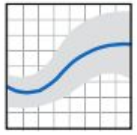
e + geom_quantile()

x, y, alpha, color, group, linetype, size, weight



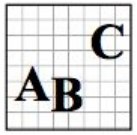
e + geom_rug(sides = "bl"**)**

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()

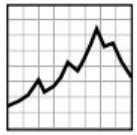
x, y, alpha, colour, fill, size

Continuous Function

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

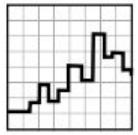
i + geom_area()

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?

> “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

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

Geometric Objects and their Visual Attributes

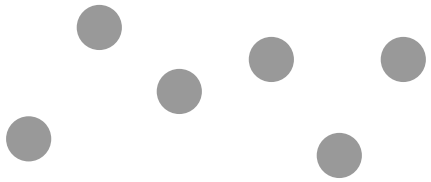
Datavis core idea

Simply put, data visualization is nothing else than mapping/encoding data (e.g. quantitative & categorical values) into geometric objects and their visual attributes.

Fundamentals

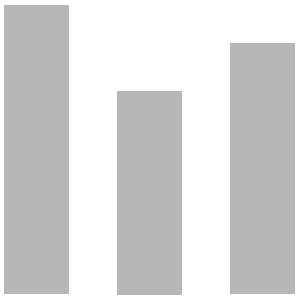
Geometric Objects (primitives)

Points

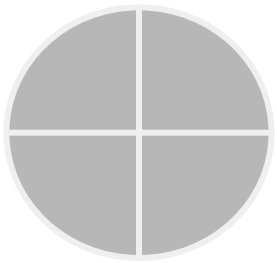


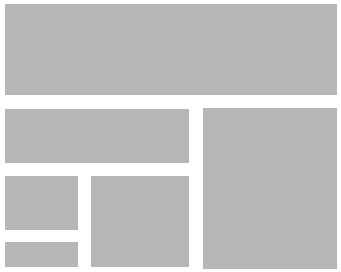
Lines

Bars

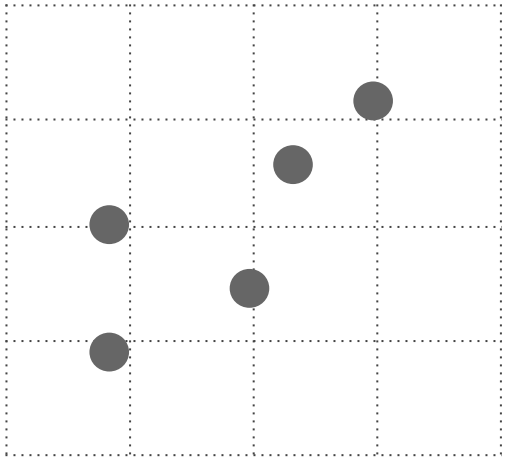


2D Areas / Polygons

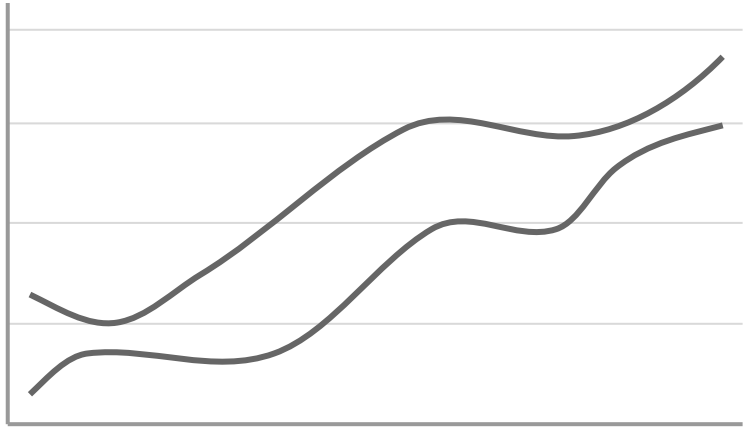




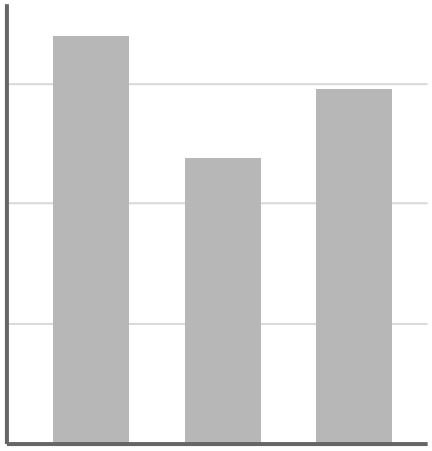
Points: e.g. scatterplot



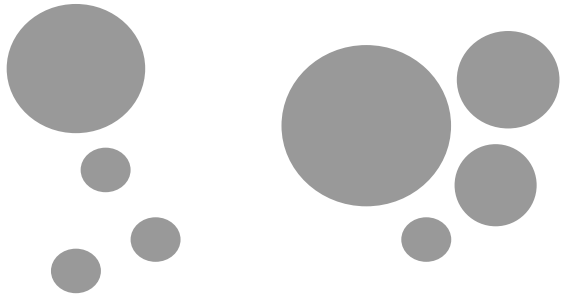
Lines: e.g. timeline



Bars: e.g. bar chart



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



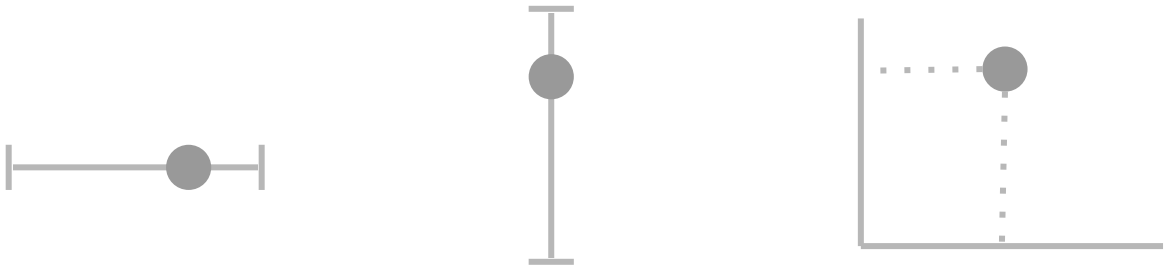
Geometric Objects

Graphical objects (typically) used to encode **quantitative** values

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

Visual Attributes

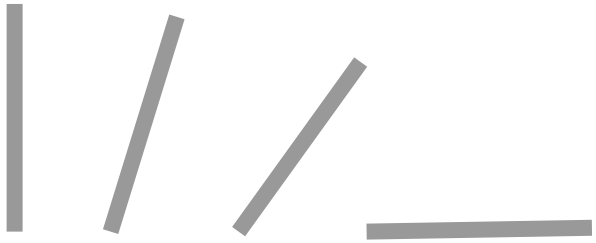
Position



horizontal vertical both

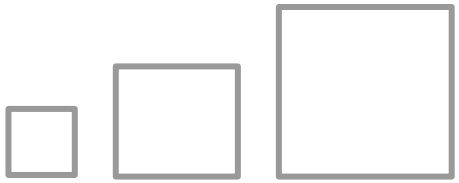
Shape

Orientation (tilt)

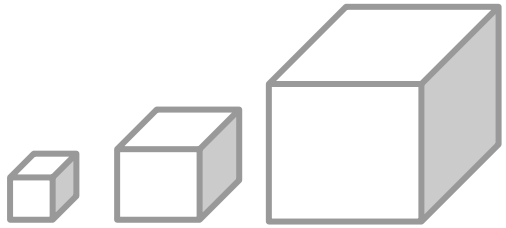


Size

length



area



volumne

Color Hue

Color Luminance

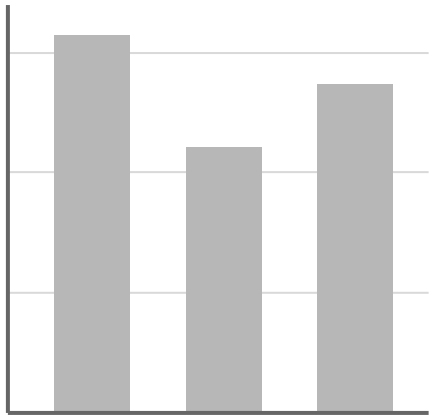
Color Saturation

Visual Attributes of Geometric objects

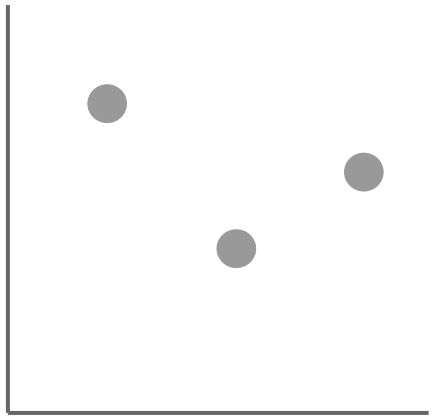
Used to encode both quantitative and categorical

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

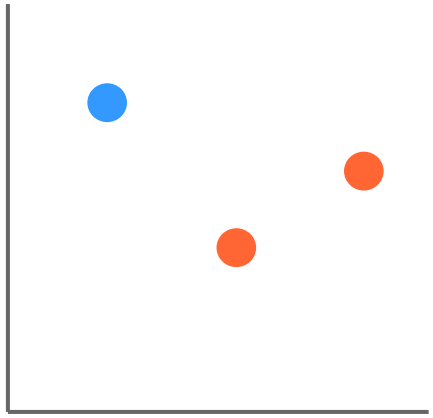
example



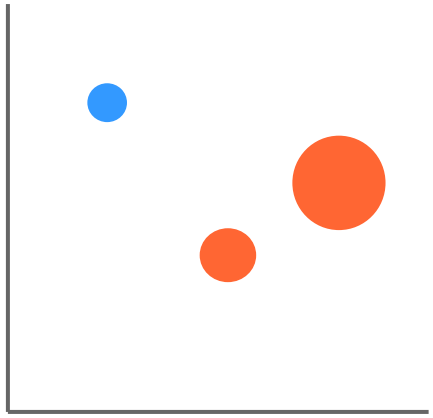
Vertical position



Vertical position
Horizontal position



Vertical position
Horizontal position
Color hue

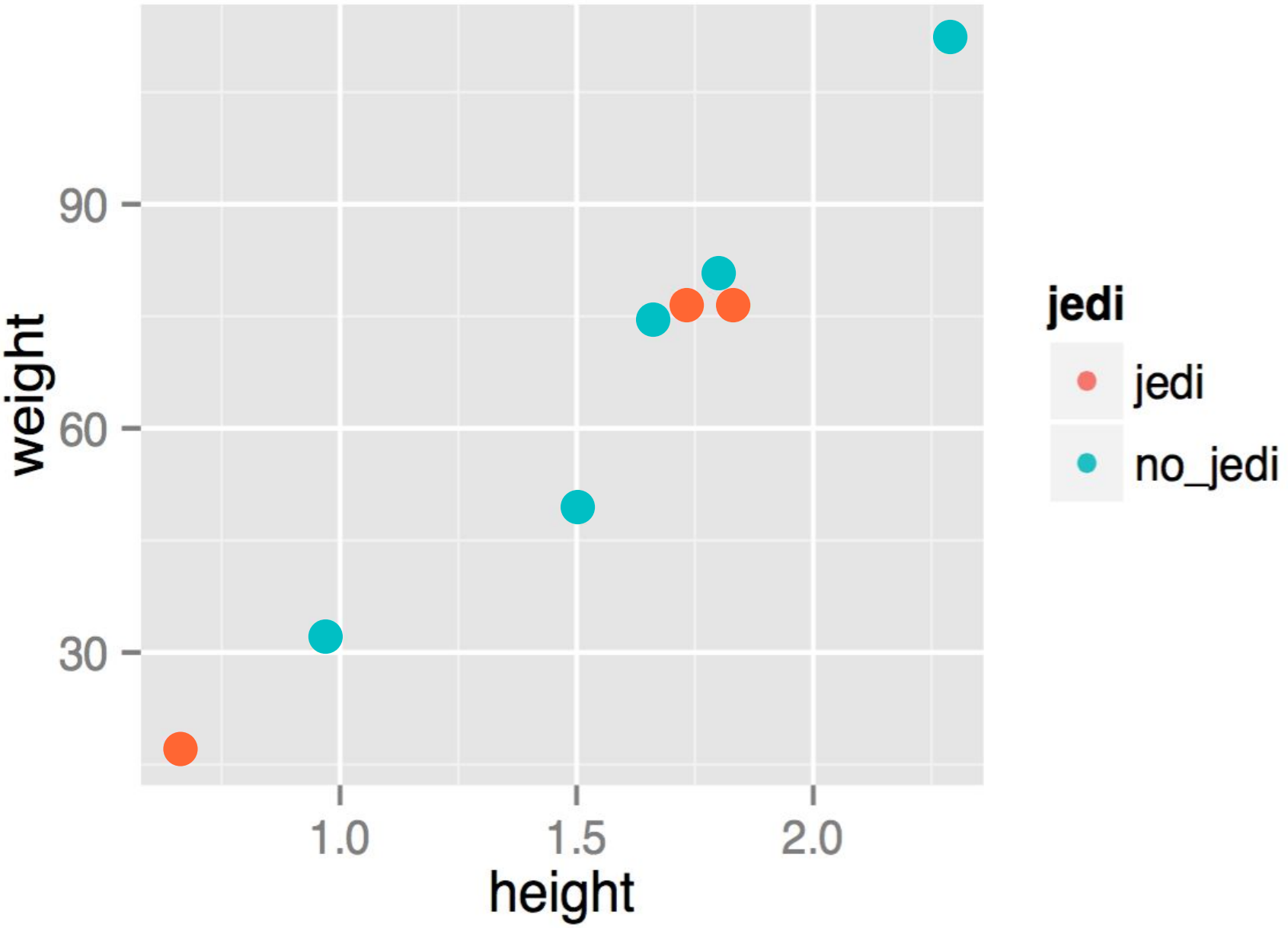


Vertical position
Horizontal position
Color hue
Size (area)

Example

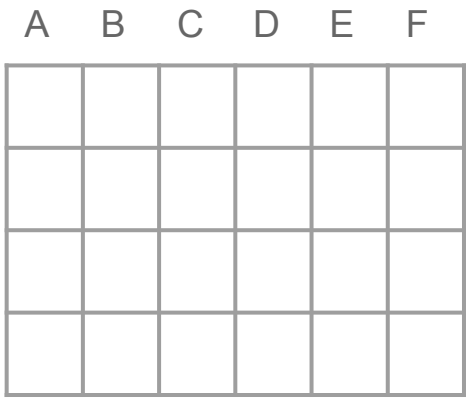
name	gender	height	weight	jedi	species	weapon
Luke Skywalker	male	1.72	77	jedi	human	lightsaber
Leia Skywalker	female	1.5	49	no_jedi	human	blaster
Obi-Wan Kenobi	male	1.82	77	jedi	human	lightsaber
Han Solo	male	1.8	80	no_jedi	human	blaster
R2-D2	male	0.96	32	no_jedi	droid	unarmed
C-3PO	male	1.67	75	no_jedi	droid	unarmed
Yoda	male	0.66	17	jedi	yoda	lightsaber
Chewbacca	male	2.28	112	no_jedi	wookiee	bowcaster

Let's use these variables to make a scatterplot

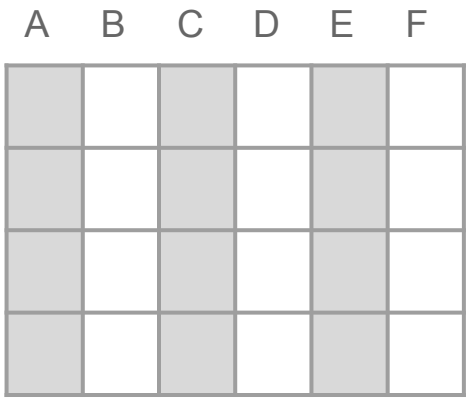

weight
90
60
30
1.0
1.5
2.0
height
jedi
jedi
no_jedi

How does it (conceptually) work?

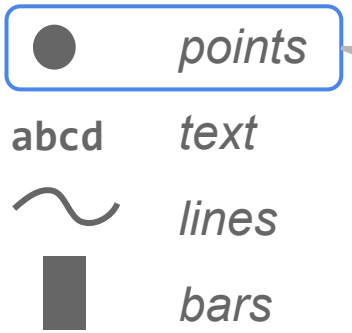
1 Dataset



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 values

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