CMPT 733 – Big Data Programming II

Visualization Principles for Data Science

Instructor Steven Bergner

Course website https://coursys.sfu.ca/2024sp-cmpt-733-g1/pages/

Source: Ch. 6.4 - 6.6 of "Principles of Data Science" by Lau, Gonzales, Nolan

Sources

Books

- Tamara Munzner ”Visualization Analysis and Design”, 2014
- Lau, Gonzalez, Nolan “Principles and Techniques of Data Science”

Slides

- Torsten Möller’s Visualization course, Spring 2018
- UC Berkley Data 100 (Lau, Nolan, Dudoit, Perez)

Defining Visualization (Vis)

Computer-based visualization systems provide visual representations of datasets designed to help people carry out tasks more effectively.

["Visualization Analysis and Design" by T. Munzner, 2014]

Why have a human in the loop?

- Not needed when automatic solution is trusted
- Good for ill-specified anlaysis problems
  - Common setting: "What questions can we ask?"

Why have a human in the loop?

Computer-based visualization systems provide visual representations of datasets designed to help people carry out tasks more effectively.

Munzner, T. (2014)

Long-term use

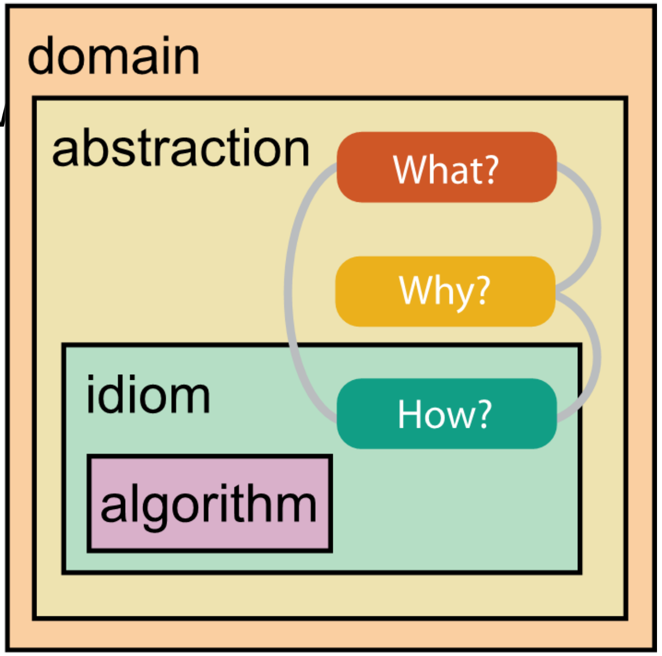
- Exploratory analysis of scientific data
- Presentation of known results

Short-term use

- For **developers** of automatic solutions:
  - Understand requirements for model development
  - Refine/debug and determine parameters
- For **end users** of automatic solutions: verify, build trust

Analysis framework: four levels

- ***Domain*** situation: Who are the target users?
- ***Abstraction***: Translate from specifics of domain to vocabulary of vis
- **What** is shown? *Data abstraction*
  - Don't just draw what you're given: transform to new
- **Why** is the user looking at it? *Task abstraction*
- **How** is it shown? ***Idiom (Vis technique)***
  - Visual encoding idiom: How to draw
  - Interaction idiom: How to manipulate
- ***Algorithm***: efficient computation



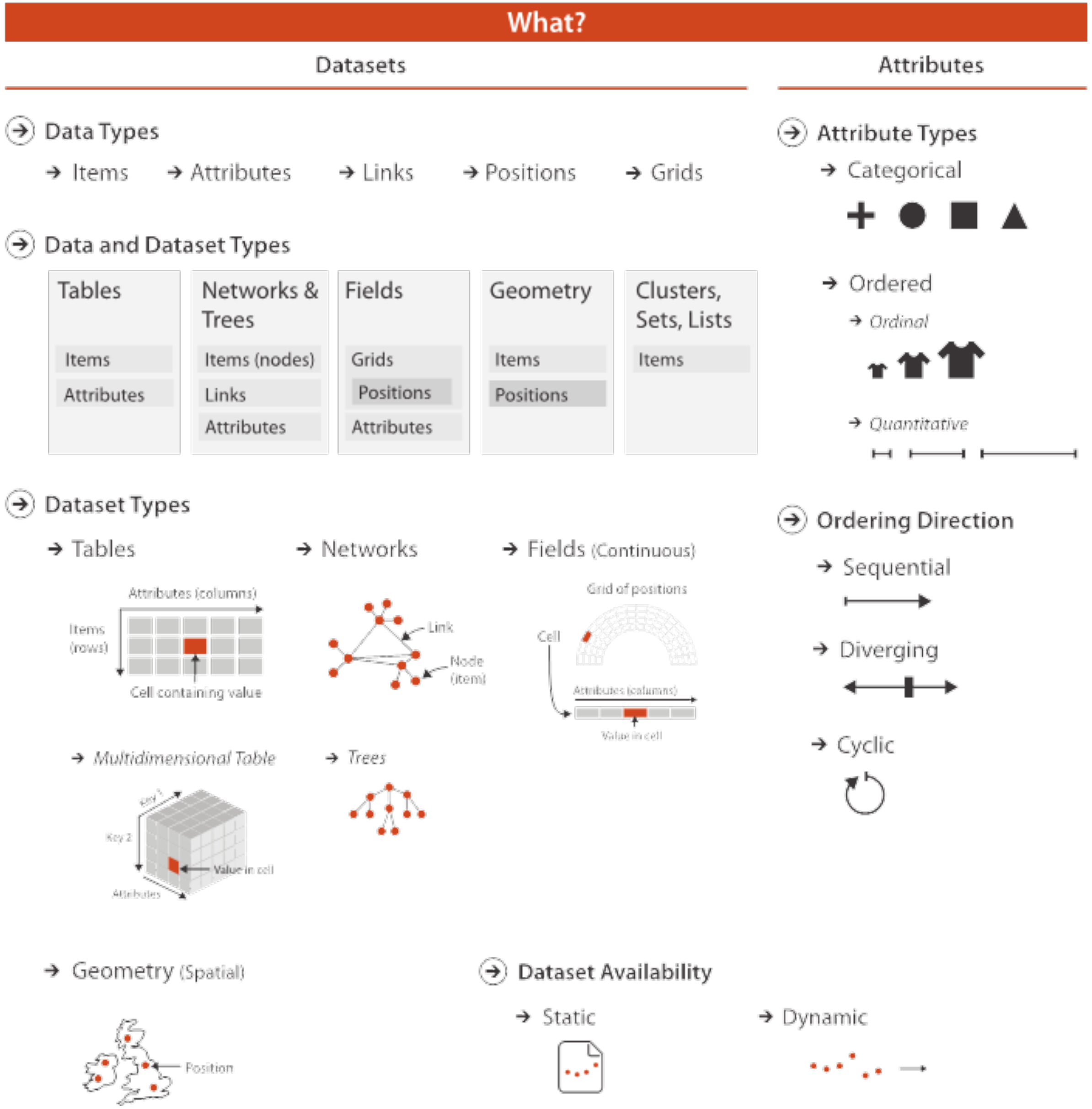
[A Nested Model of Visualization Design and Validation. Munzner. IEEE TVCG 15(6):921-928, 2009 (Proc. InfoVis 2009).]

Resource limitations

- **Computational** limits
  - Processing time and system memory
- **Human** limits
  - Human attention and memory
  - Understanding abstractions
- **Display** limits
  - Pixels are precious
  - Information density tradeoff: Info encoding vs unused whitespace

Understand
Data, Task, and Encoding

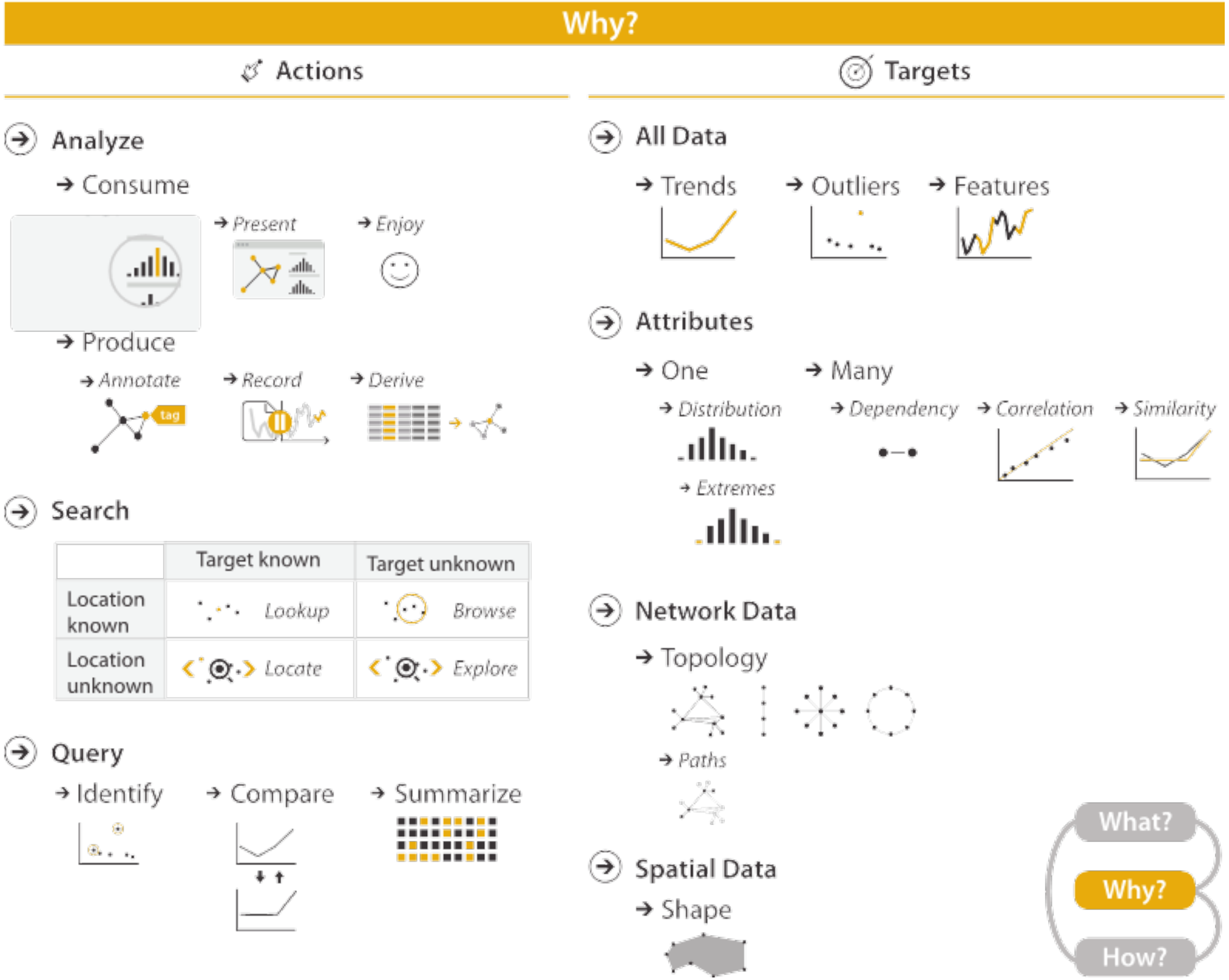
Data Types



[T. Munzner, 2014]

- Items and attributes as rows and columns of tables
- Position and time are special attributes
- Spatial data on grids makes computation easier

Tasks

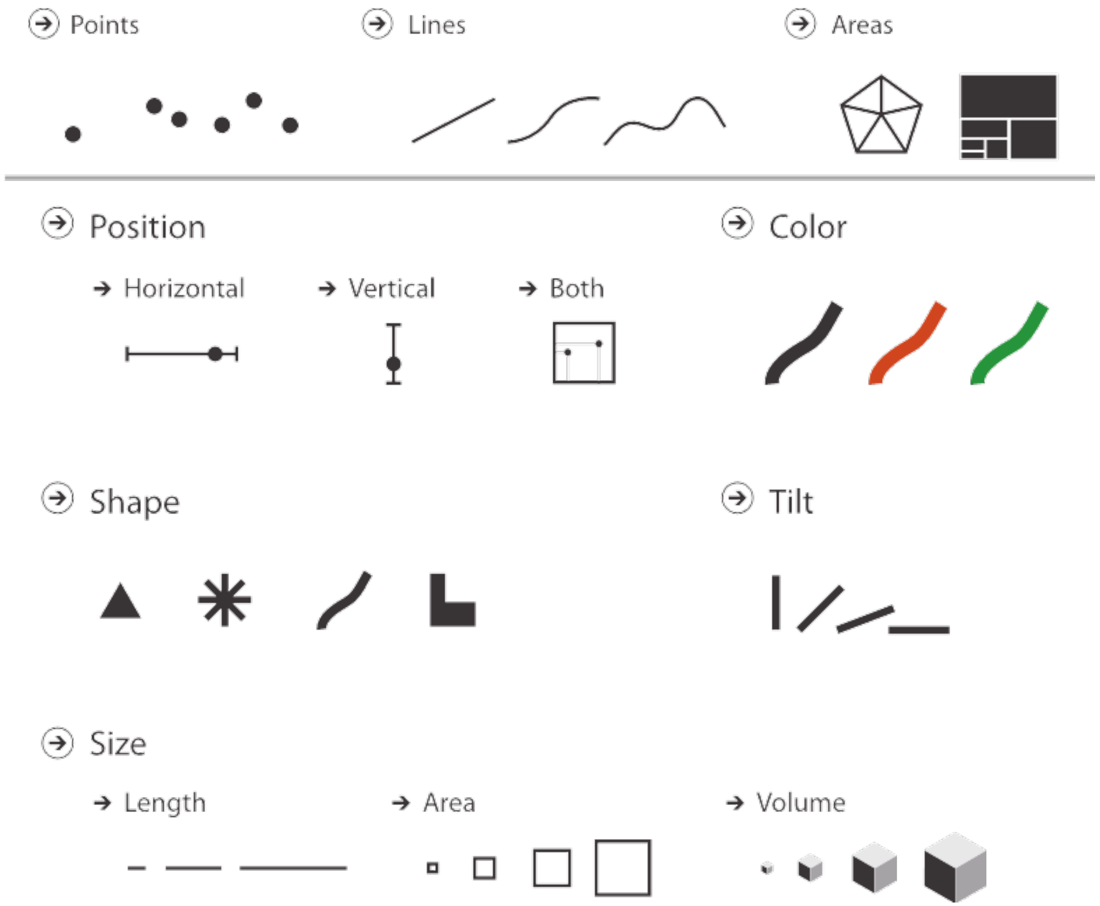


- Actions
  - Analyze
  - Search
  - Query
- Targets
  - Item & Attributes
  - Topology & Shape

[T. Munzner, 2014]

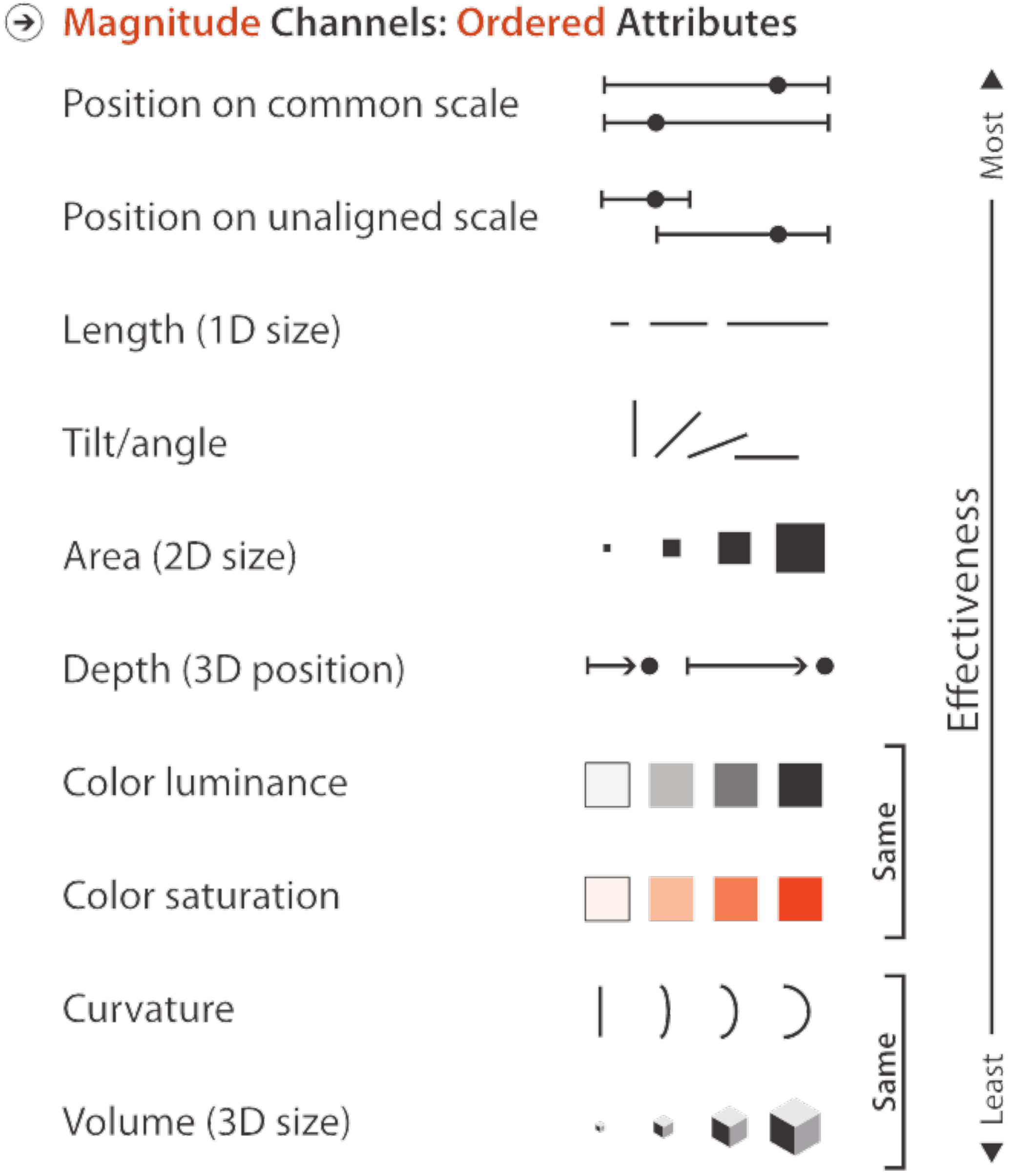
Visual Encoding – How?

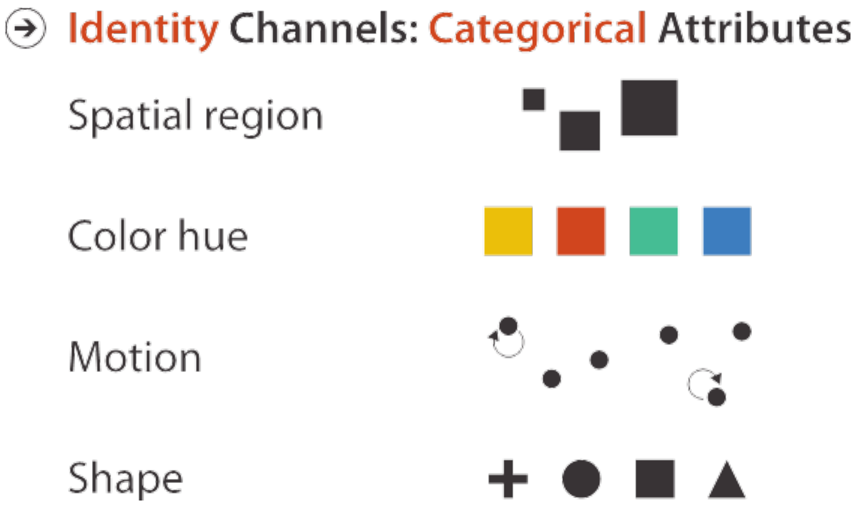
- Marks
  - Geometric primitives
- Channels
  - Appearance of marks
  - Redundant coding of data with multiple channels is possible



[T. Munzner, 2014]

Design Principles
for Task Effective Visualization





Expressiveness principle

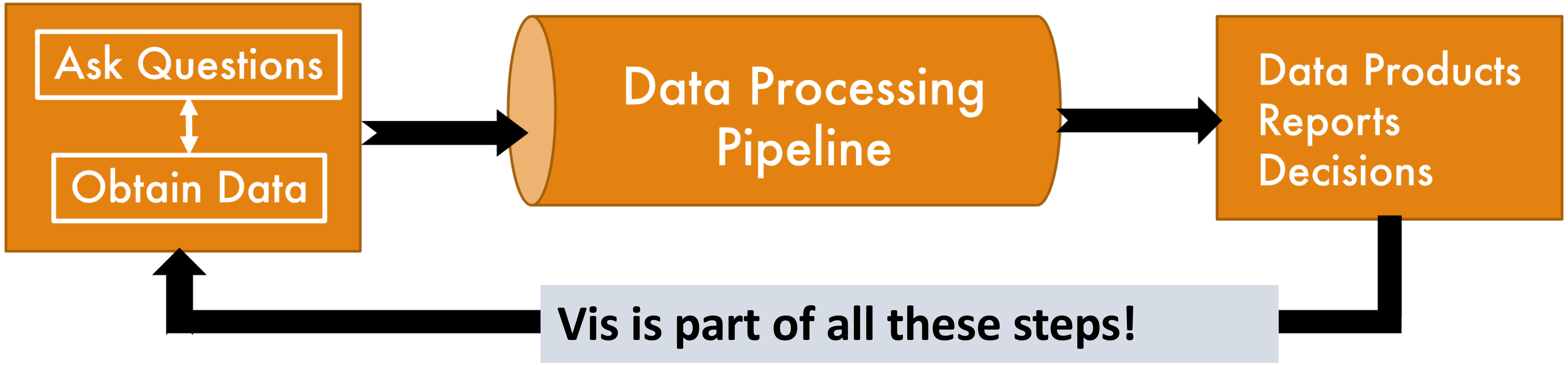
- **Match channel characteristics and data type**

Effectiveness principle

- **Encode important attributes with higher ranked channels**

[T. Munzner, 2014]

Recap: Data Science Lifecycle



Related Processes

Big Data Journey

- Business transformations as a company becomes more data-centric

Data Visualization *Process*

- Acquire, Parse, Filter, Mine, Represent, Refine, Interact [**Ben Fry '07**, Visualizing Data]

Data Visualization *Pipeline*

- Analyse (Wrangling), Filter, Map to visual properties, Render geometry

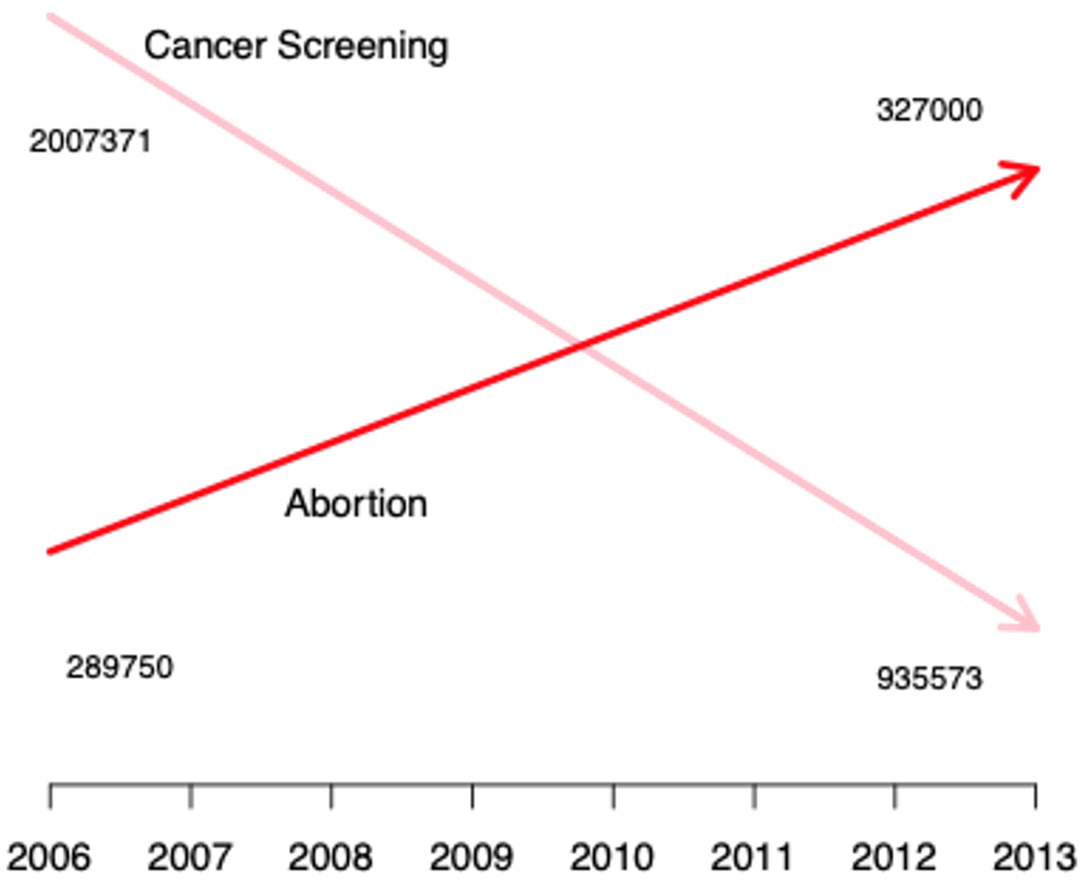
Six Principles Today

1. Scale
2. Conditioning
3. Perception
4. Transformations
5. Context
6. Smoothing

Explored via three case studies.

Case 1: Planned Parenthood 2015 Hearing

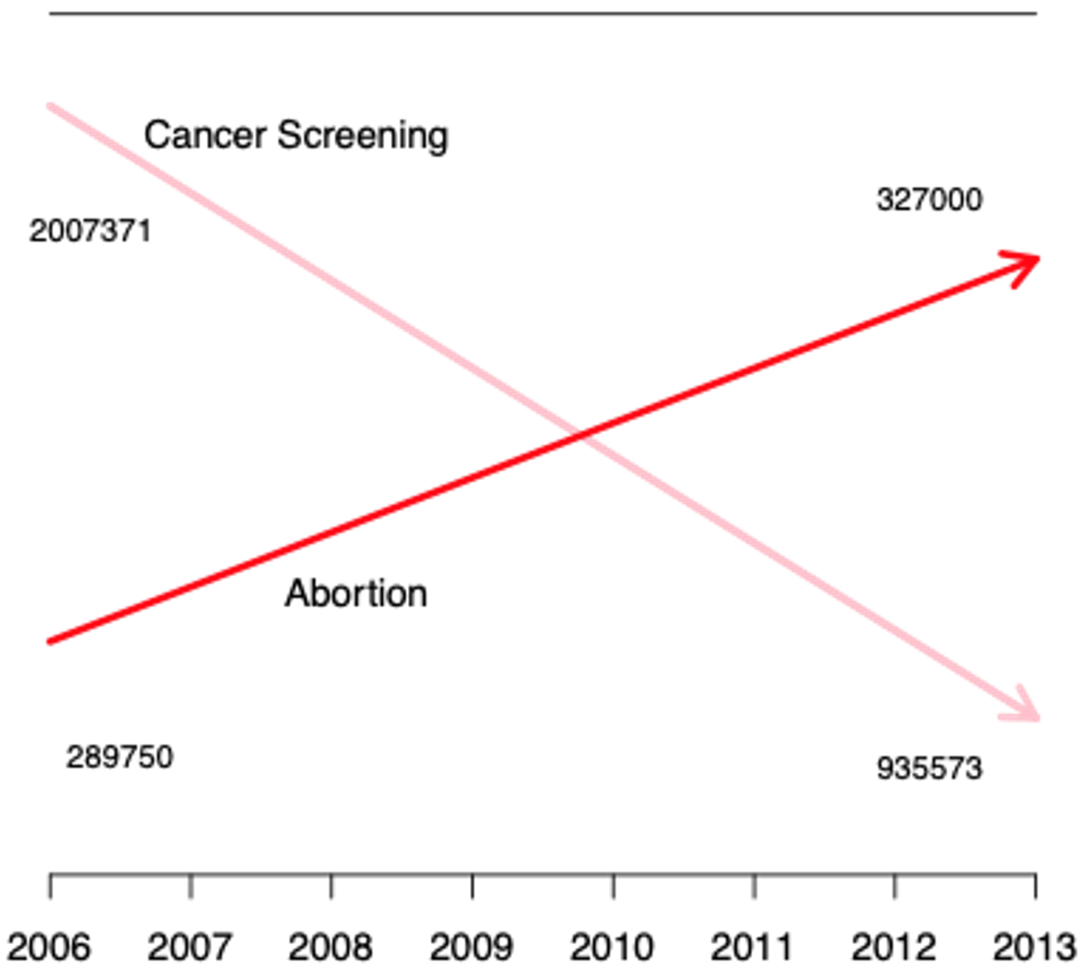
- Investigation of federal funding of Planned Parenthood in light of fetal tissue controversy
- Congressman Chaffetz (R-UT) showed plot which originally appeared in a report by Americans United for Life (http://www.aul.org/)



Full Report available at https://oversight.house.gov/interactivepage/plannedparenthood/ .

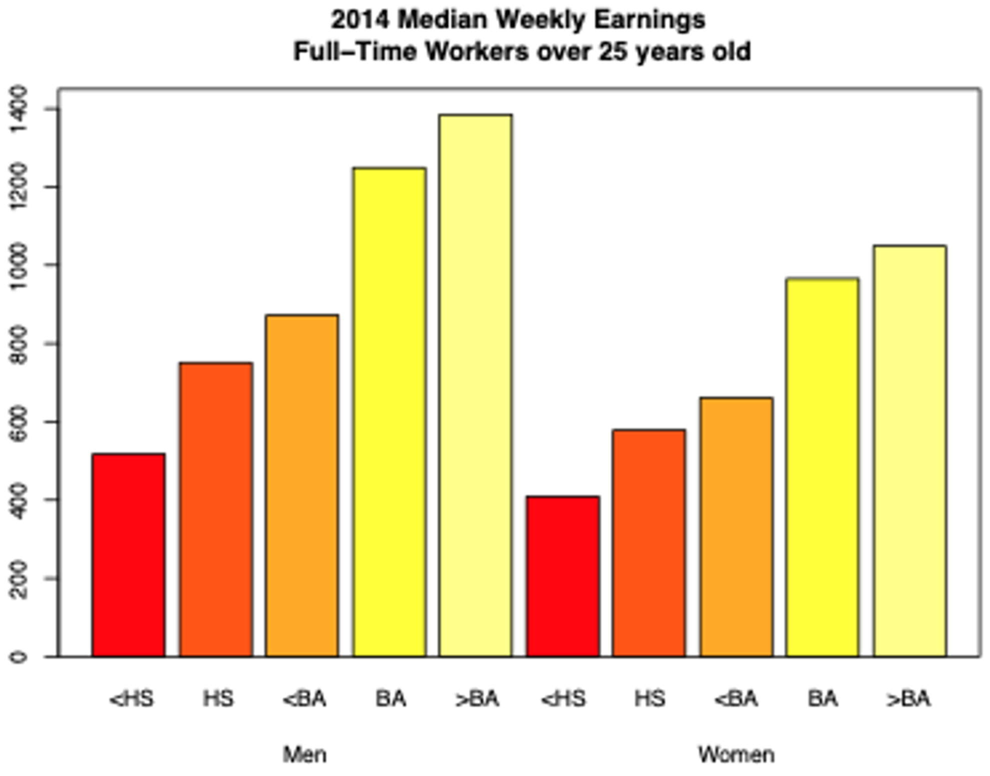
Case 1: Planned Parenthood 2015 Hearing

- Procedures: cancer screenings and abortions
- How many data points are plotted?
- What is suspicious?
- What message is this plot trying to convey?



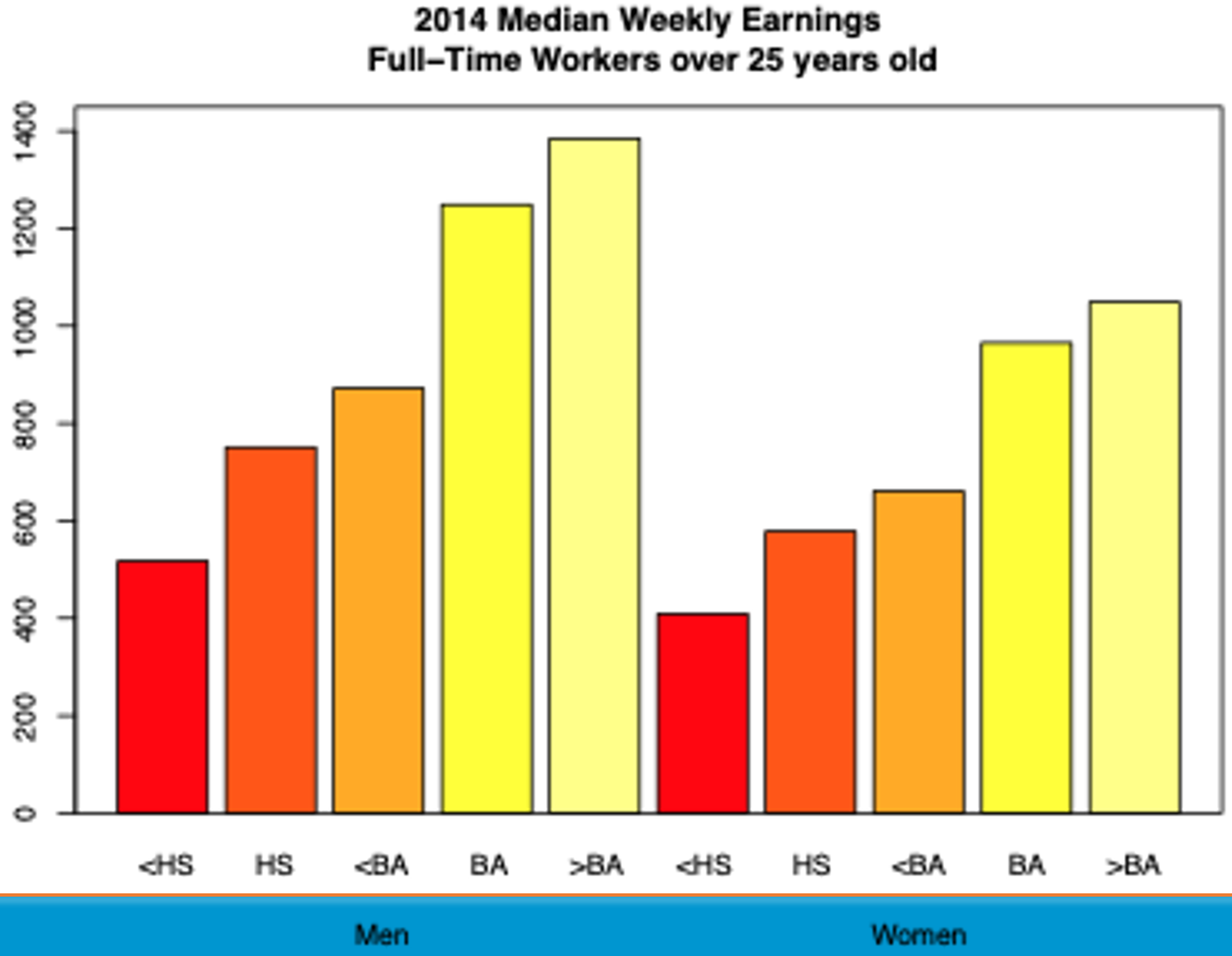
Case 2: Median Weekly Earnings

- Bureau of Labor Statistics surveys economics of labor
- www.bls.gov - Web interface to a report generating app
- Plot of median weekly earnings for males and females by education level



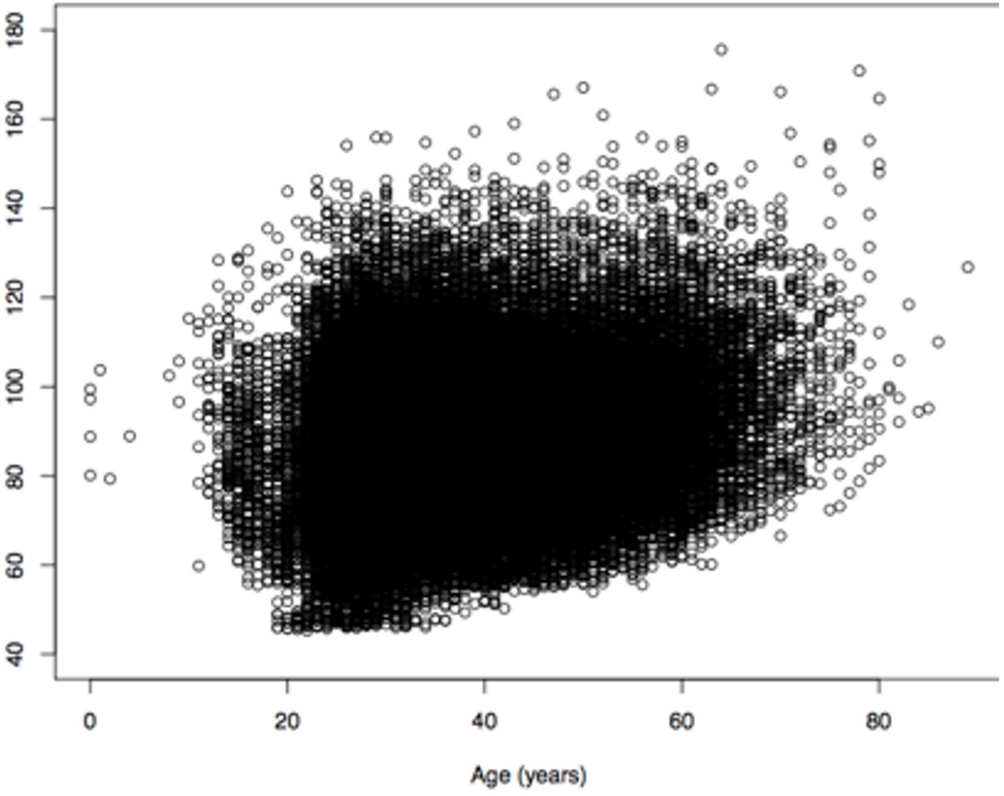
Case 2: Median Weekly Earnings

- What comparisons are easily made with this plot?
- What comparisons are most interesting and important?



Case 3: Cherry Blossom Runners

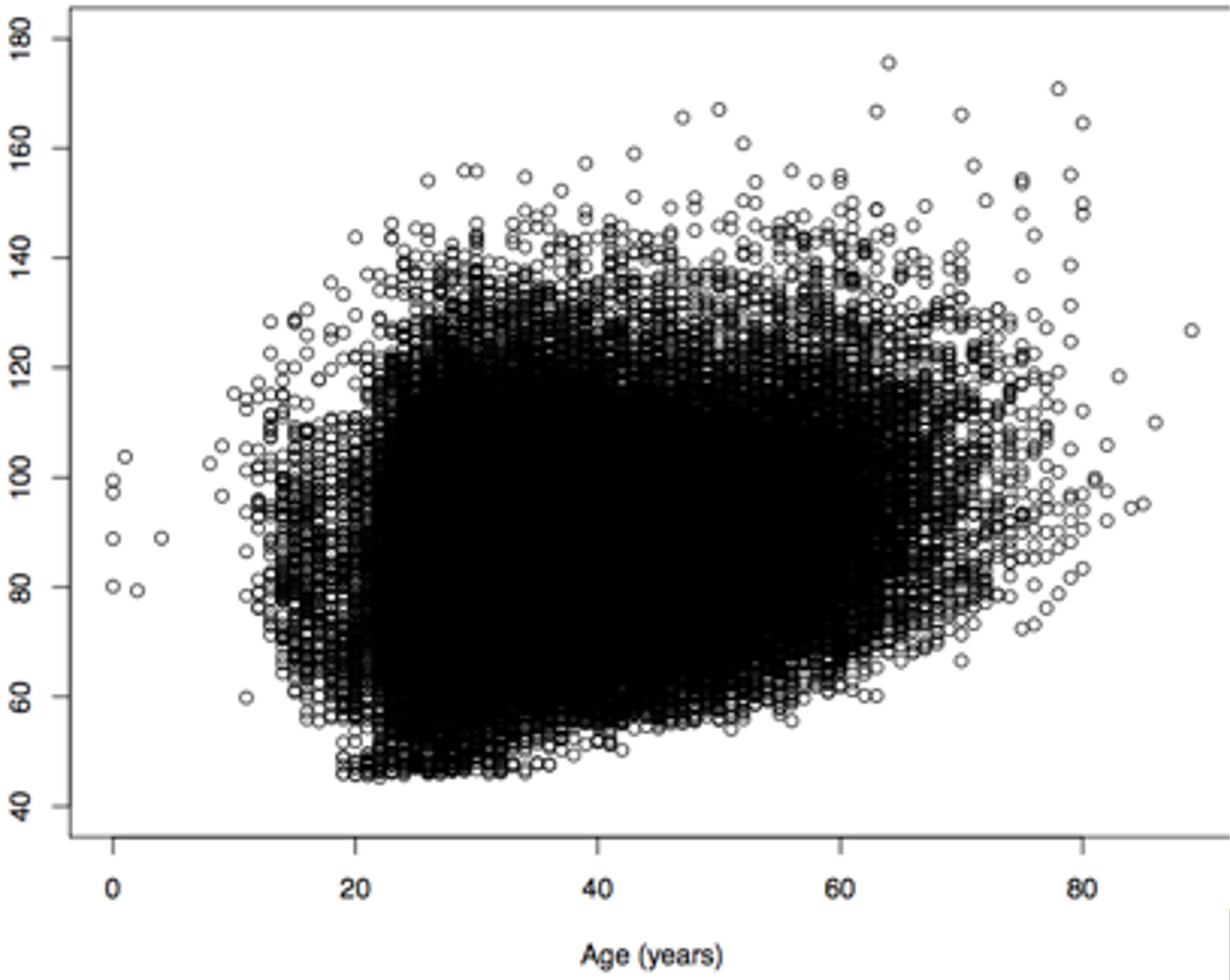
- 10 mi run in DC every April
- Results available from 1999-2019
- In 2019 over 17,000 runners
- Scatter plot of run time (min) against age (yrs)



http://www.cherryblossom.org/

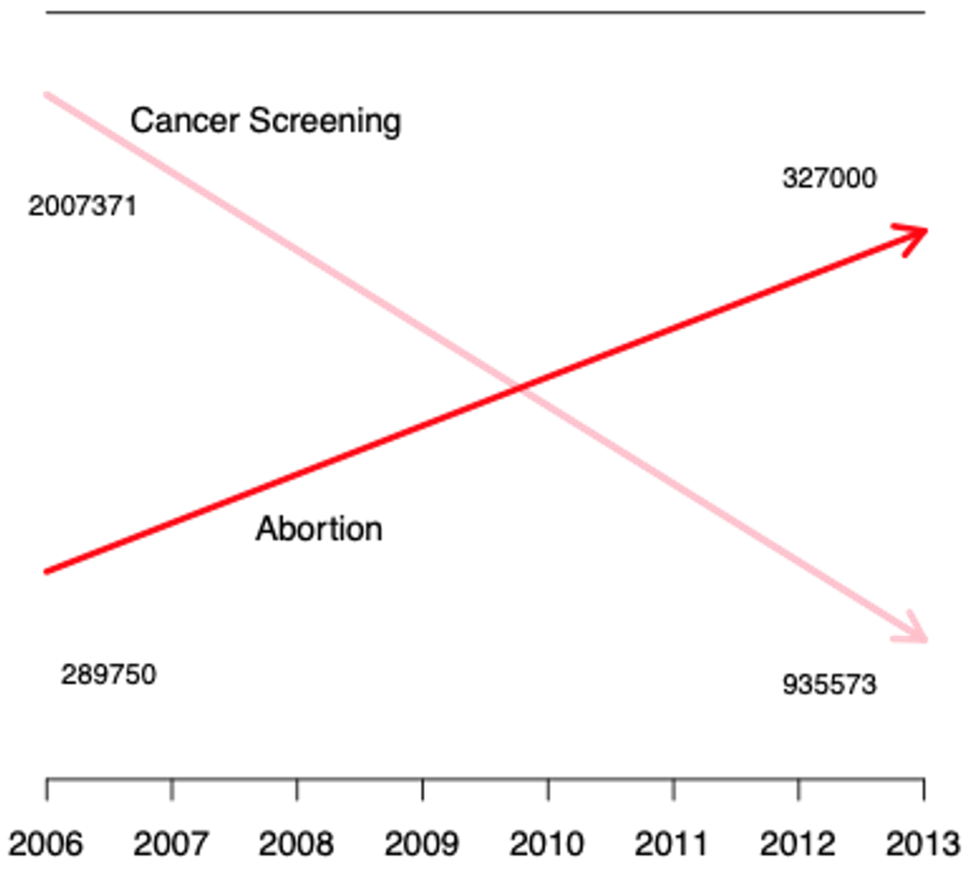
Case 3: Cherry Blossom Runners

- 70,000+ points in the plot!
- What's the relationship between run time and age?

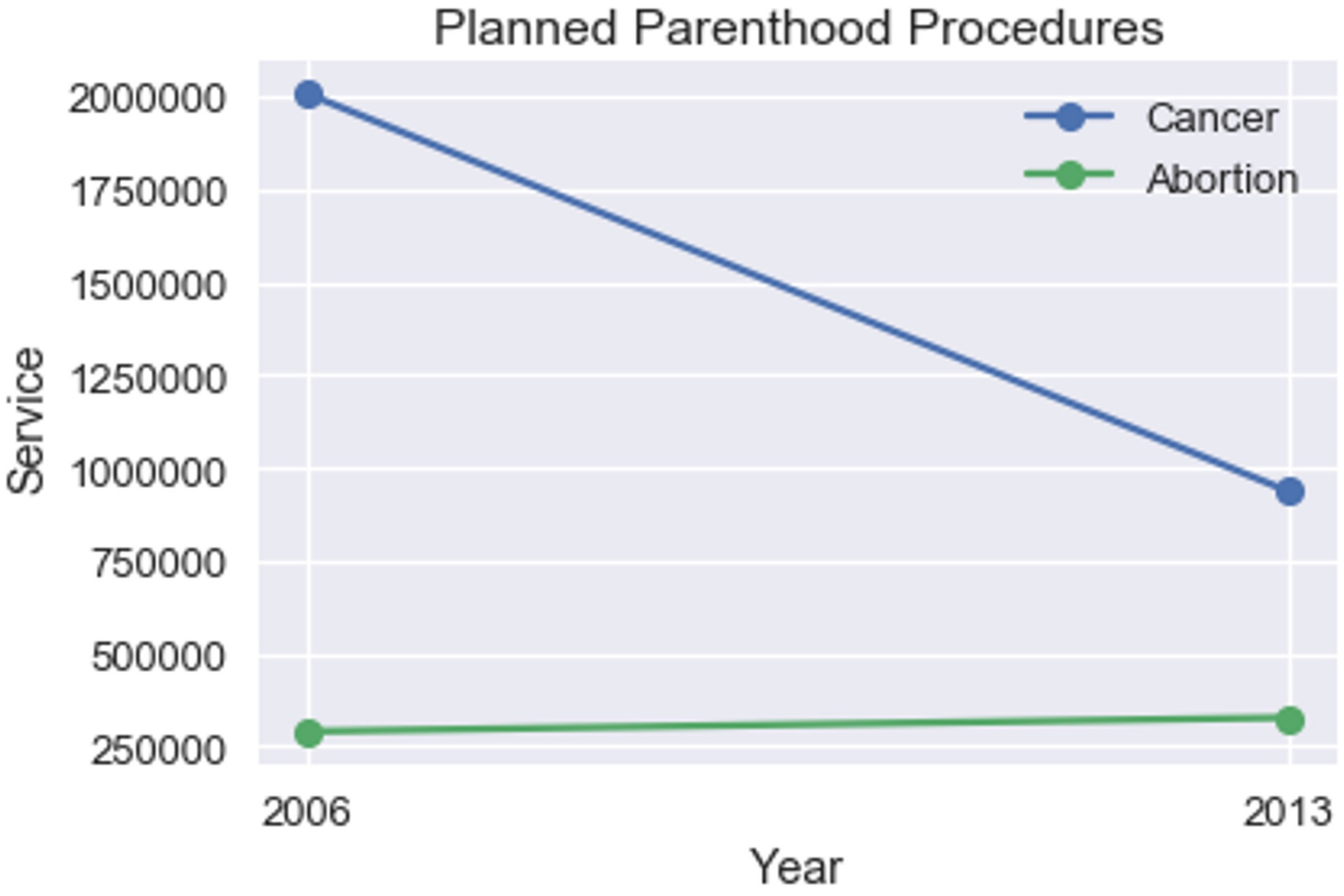


Principles of Scale

Scale



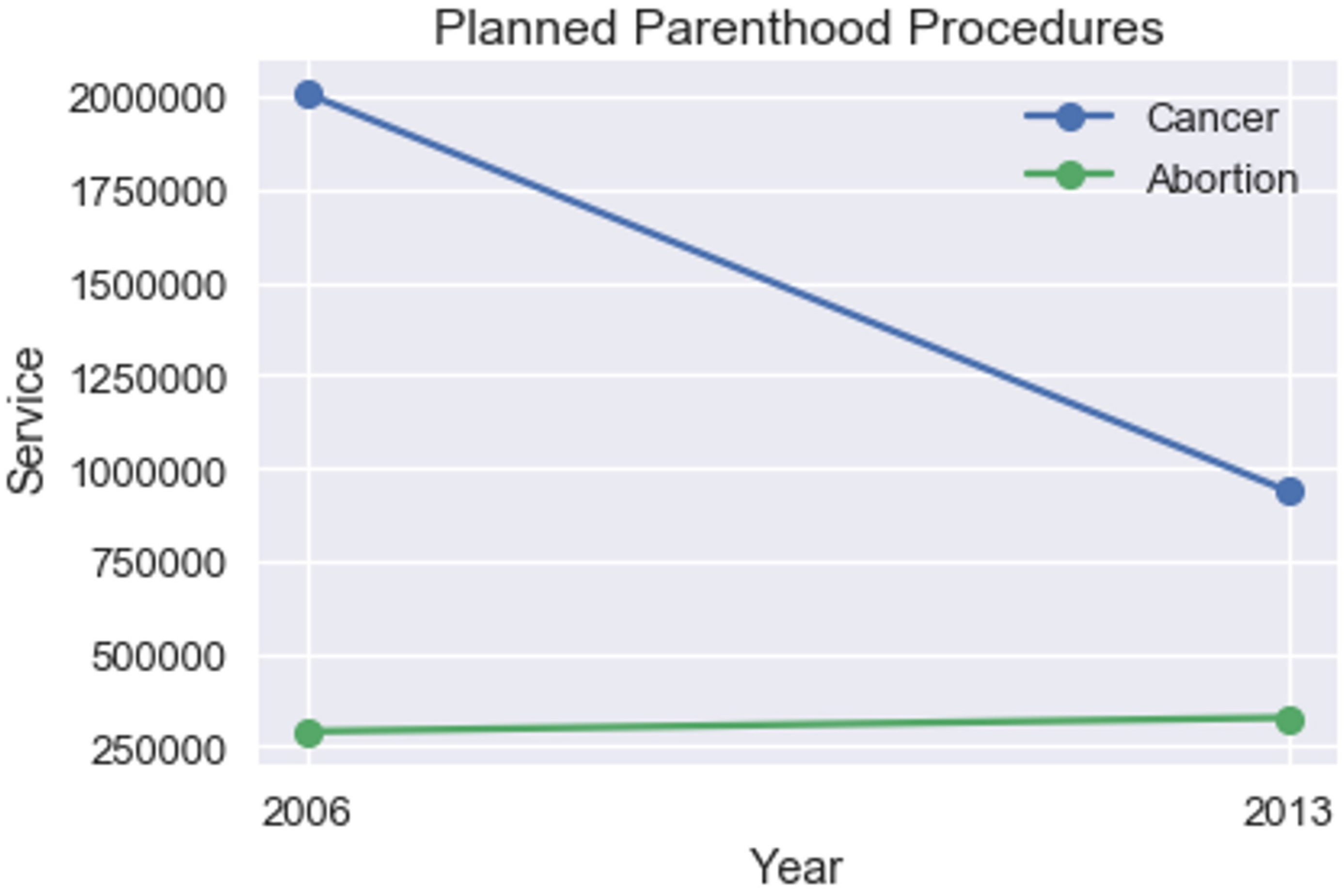
Cancer Screening
2007371
327000
Abortion
289750
935573
2006
2007
2008
2009
2010
2011
2012
2013



Planned Parenthood Procedures
2000000
1750000
1500000
1250000
1000000
750000
500000
250000
Service
Cancer
Abortion
2006
2013
Year

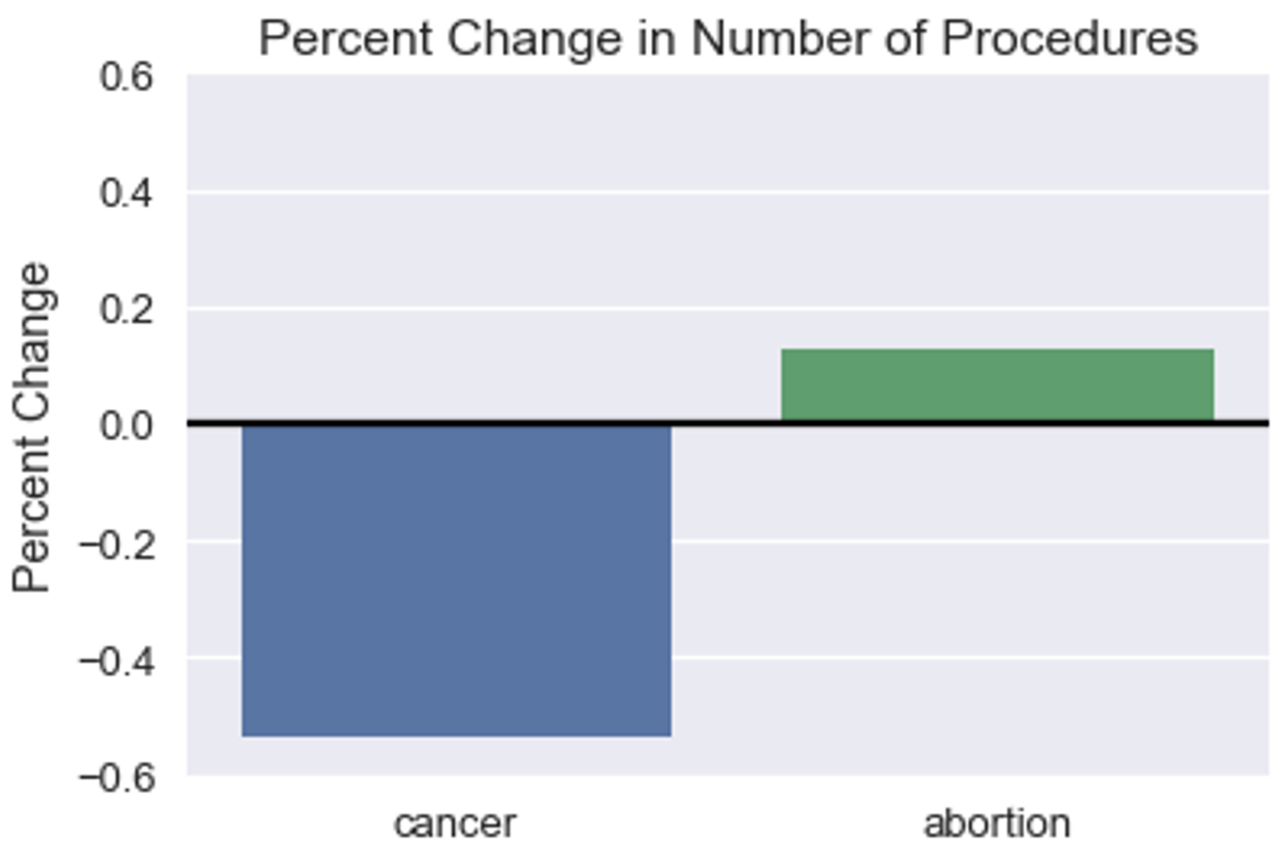
Keep consistent axis scales

- Don’t change scale mid-axis
- Don’t use two different scales for same axis
- How does this plot change perception of information?



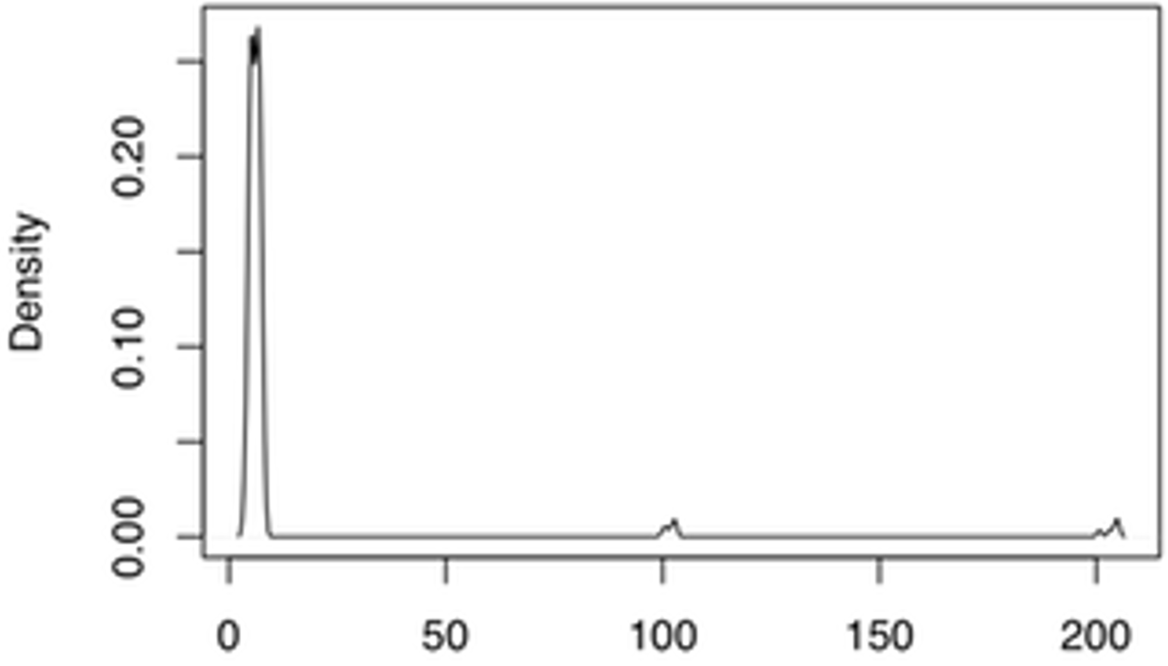
Consider Scale of Data

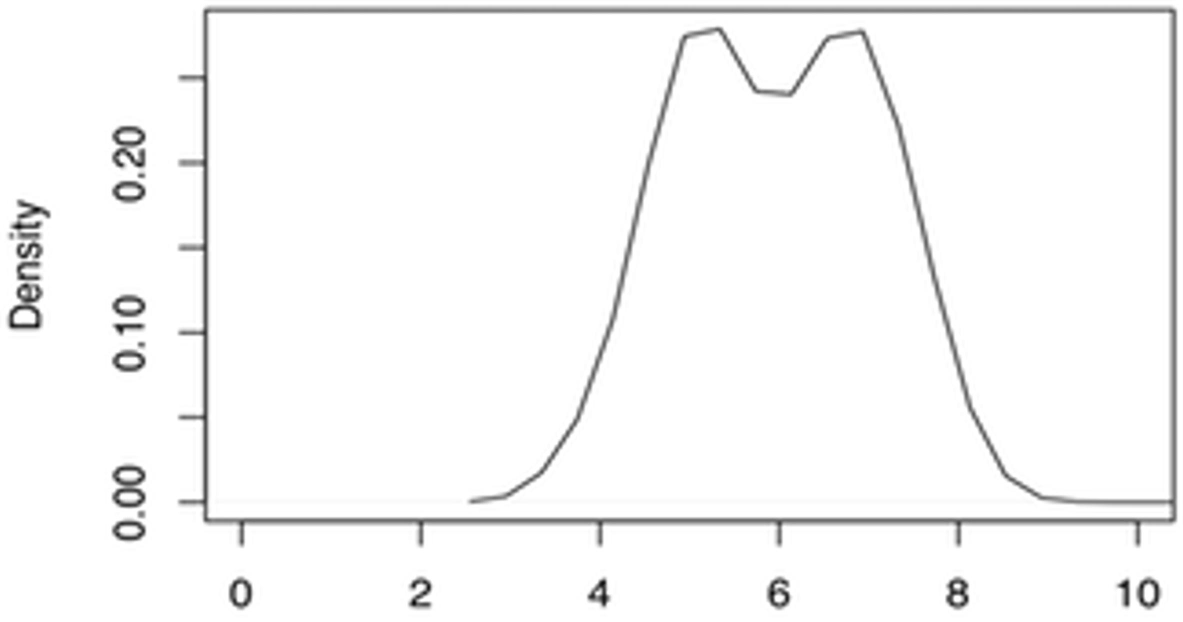
- Scales of cancer screenings vs. abortions quite different
- Can plot percent change instead of raw counts



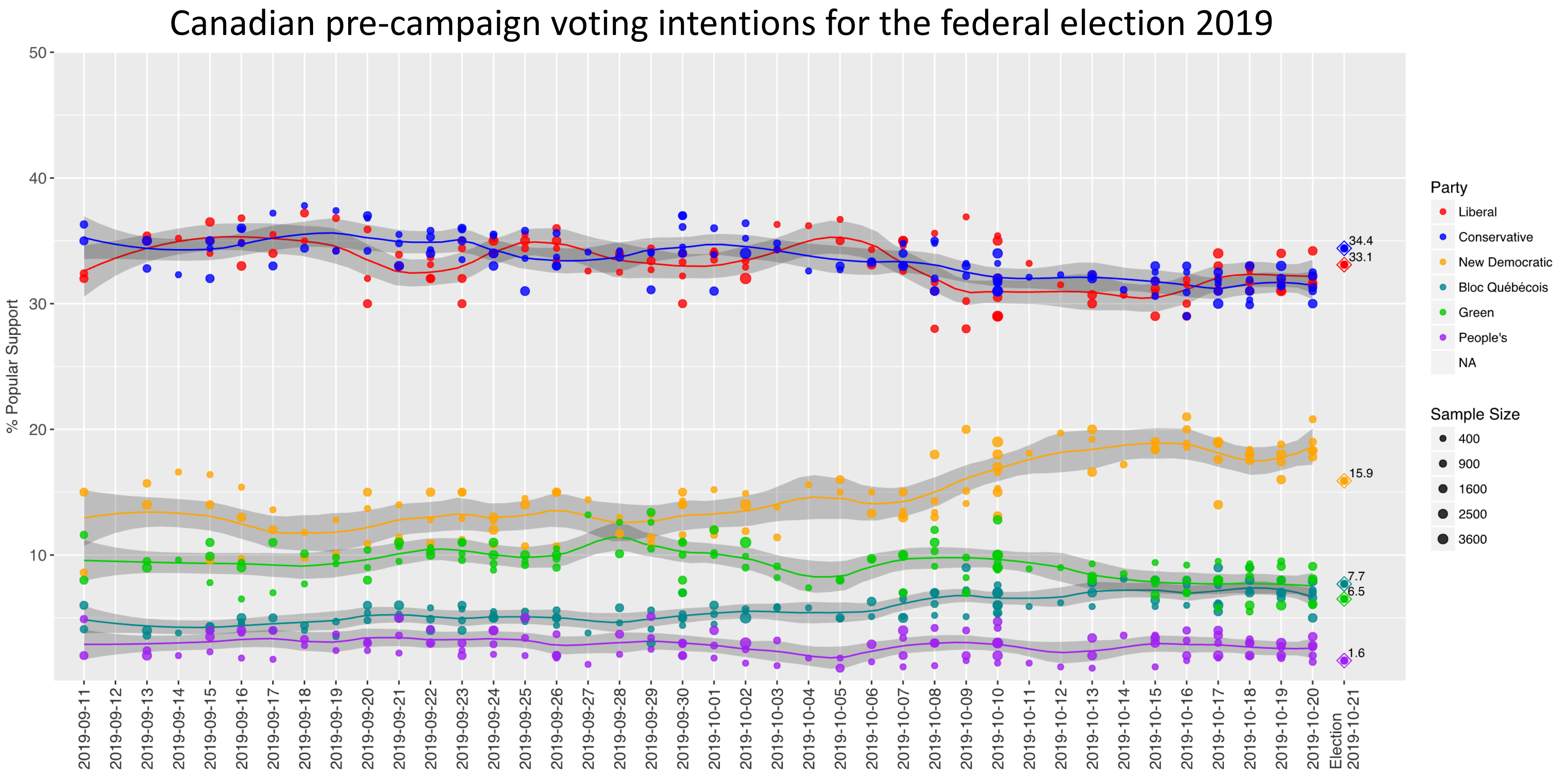
Reveal the Data

- Choose axis limits to fill plot
- If necessary, zoom into region with most of data
  - Can make separate plots for different regions





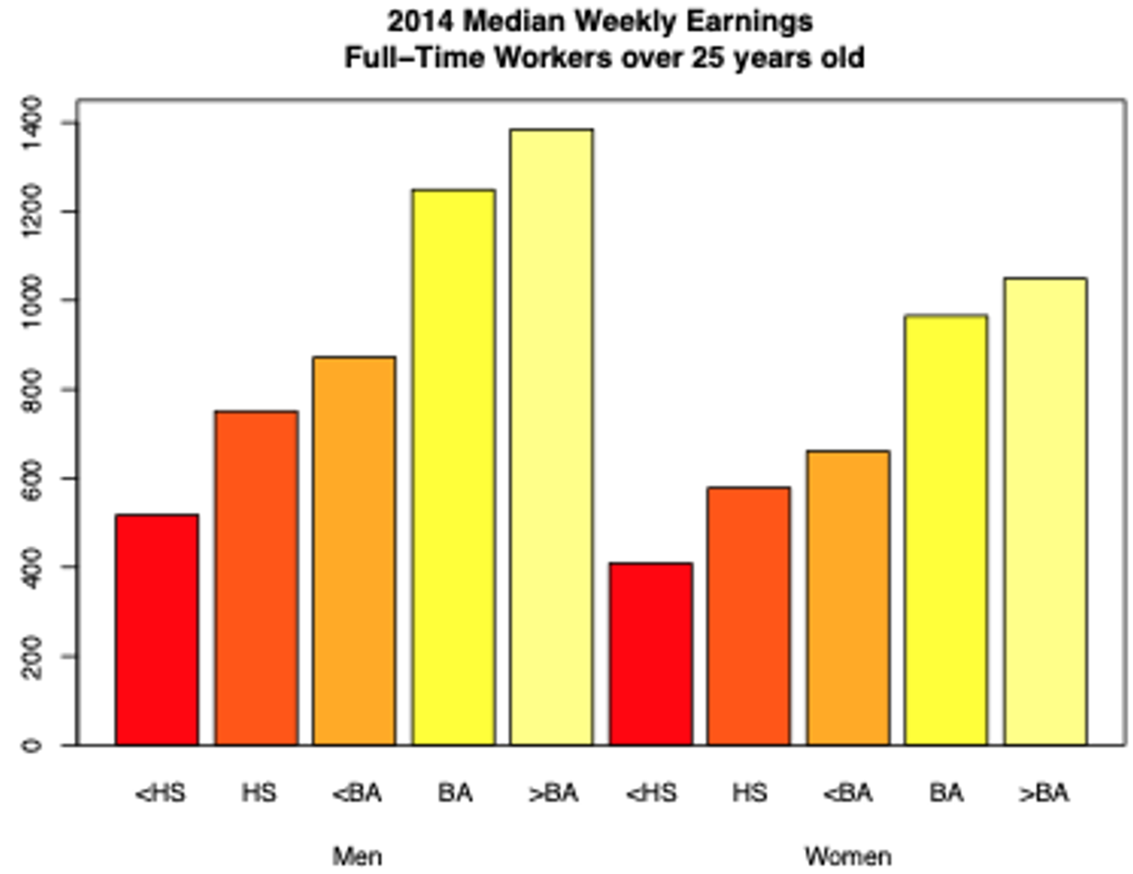
Time Scale

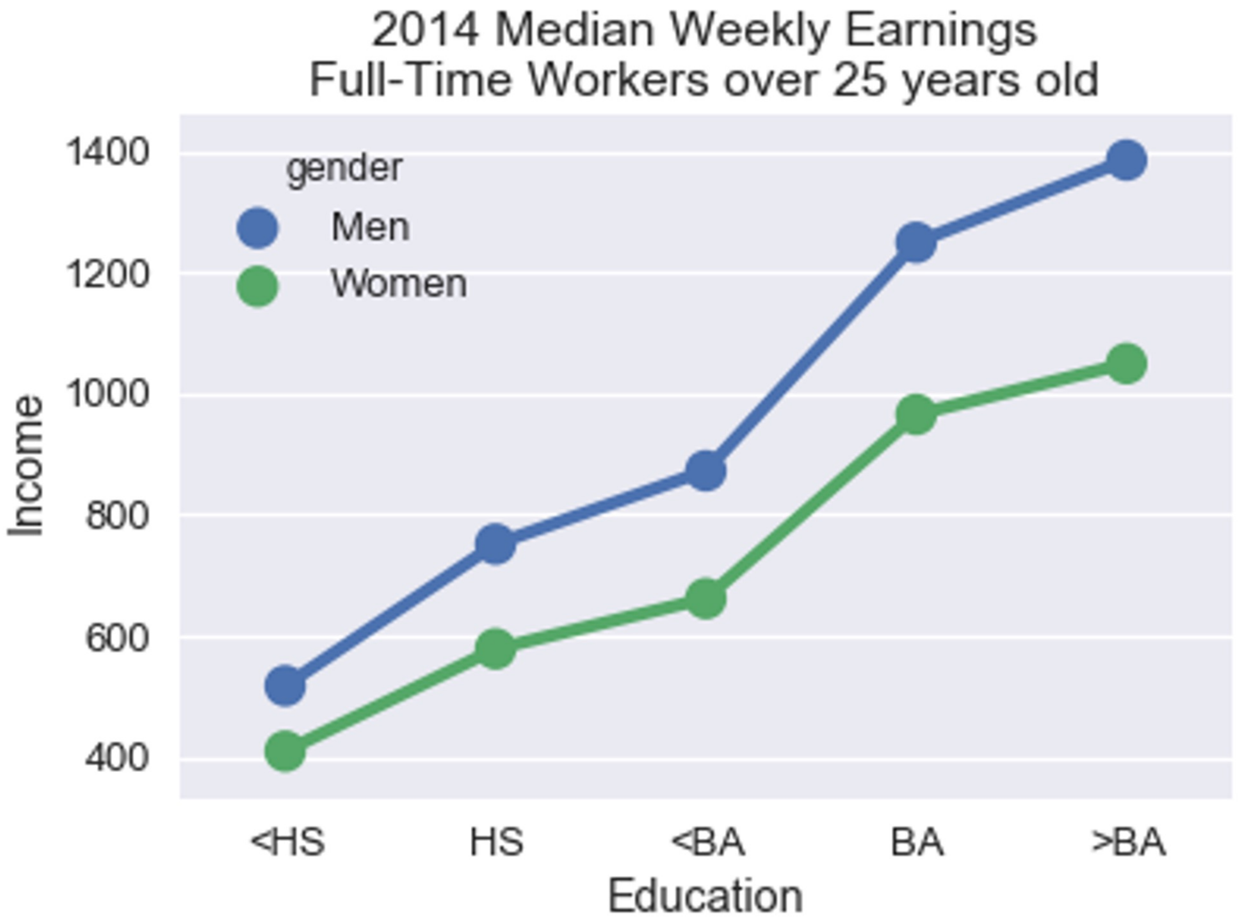


Code available at https://github.com/tylerecouture/wikiplot/blob/master/canadian-federal-polls-pre43rd.R

Principles of Conditioning

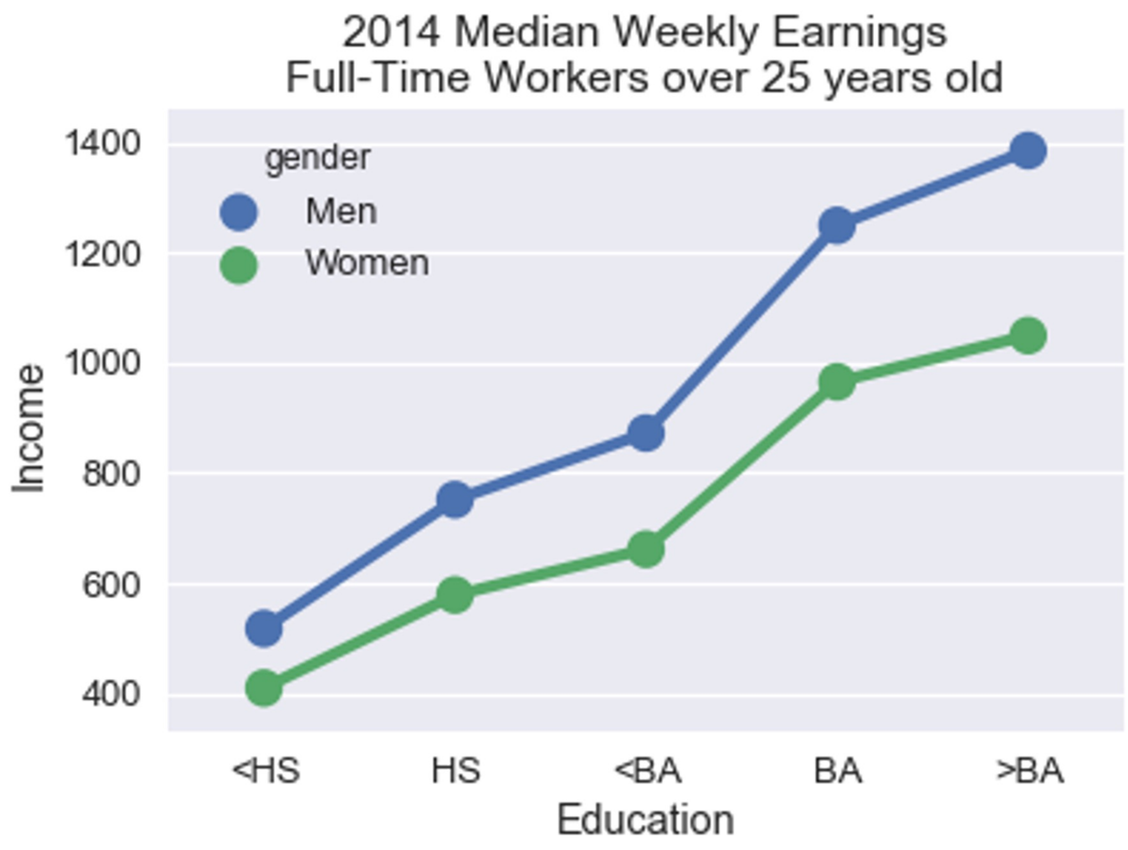
Conditioning





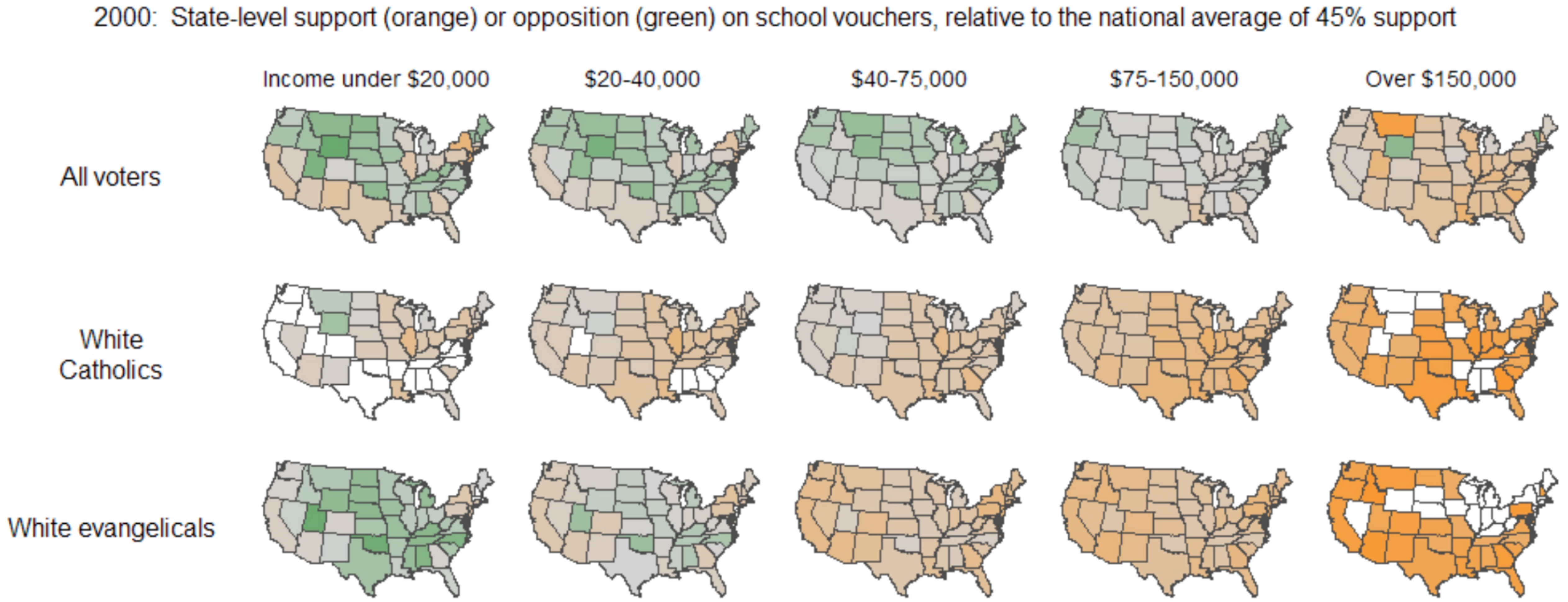
Use Conditioning To Aid Comparison

- Conditioning on male/female aligns points on x-axis
  - What does it reveal?
  - Why is this interesting?



Use Small Multiples To Aid Comparison

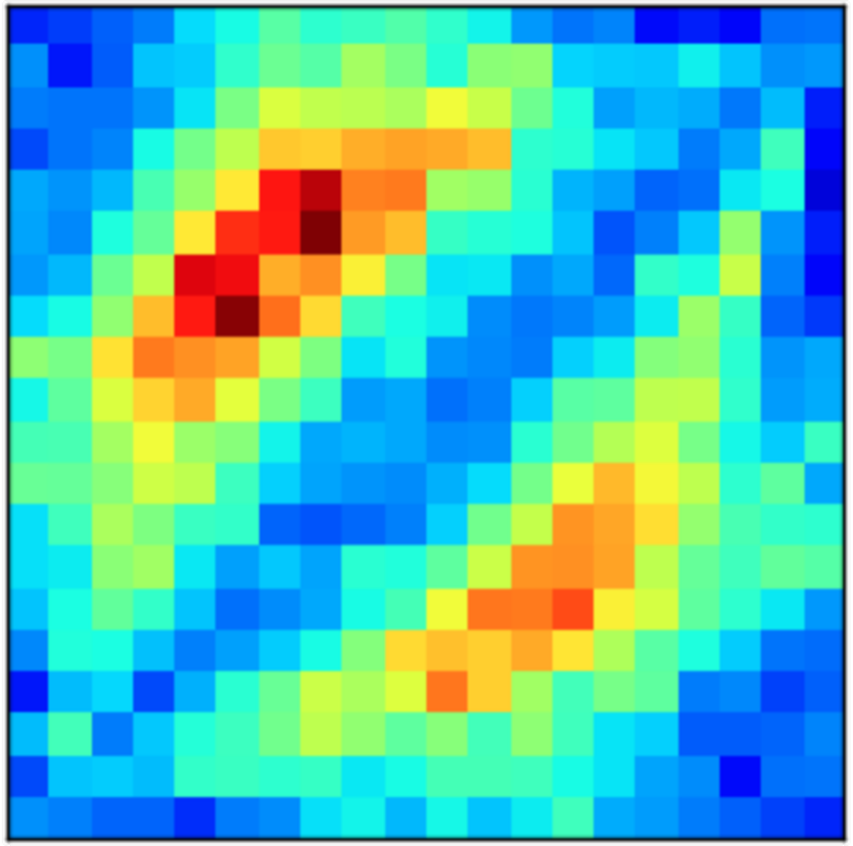
- Faceted plots that share scales are easy to compare
  - https://statmodeling.stat.columbia.edu/2009/07/15/hard_sell_for_b/



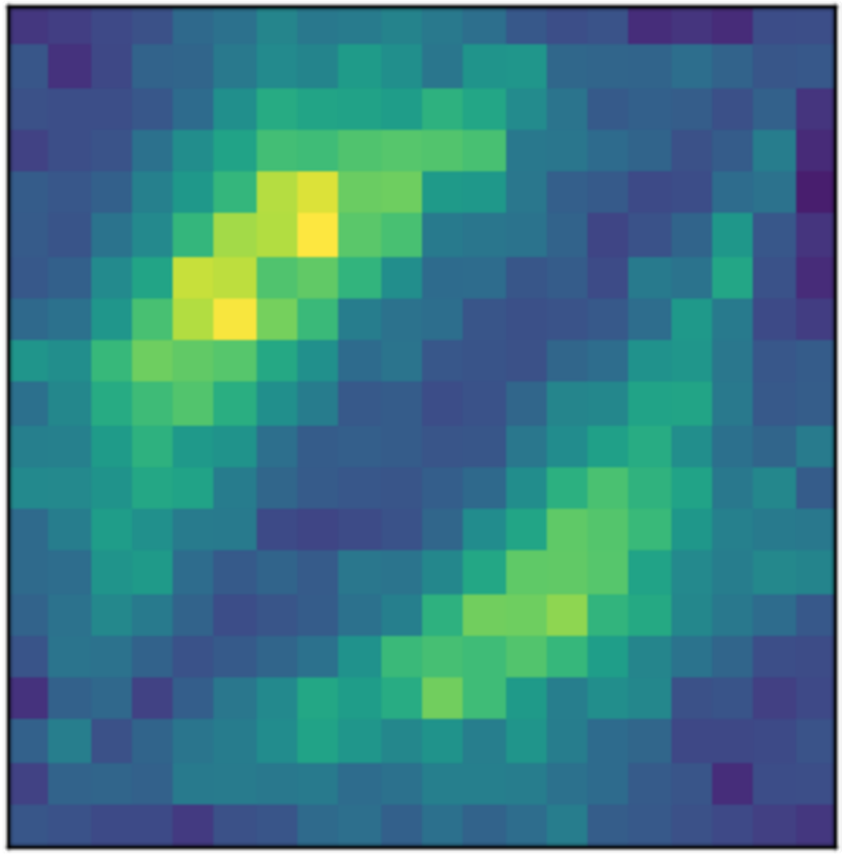
Principles of Perception

Color Choices Matter!

Jet Colormap



Viridis Colormap

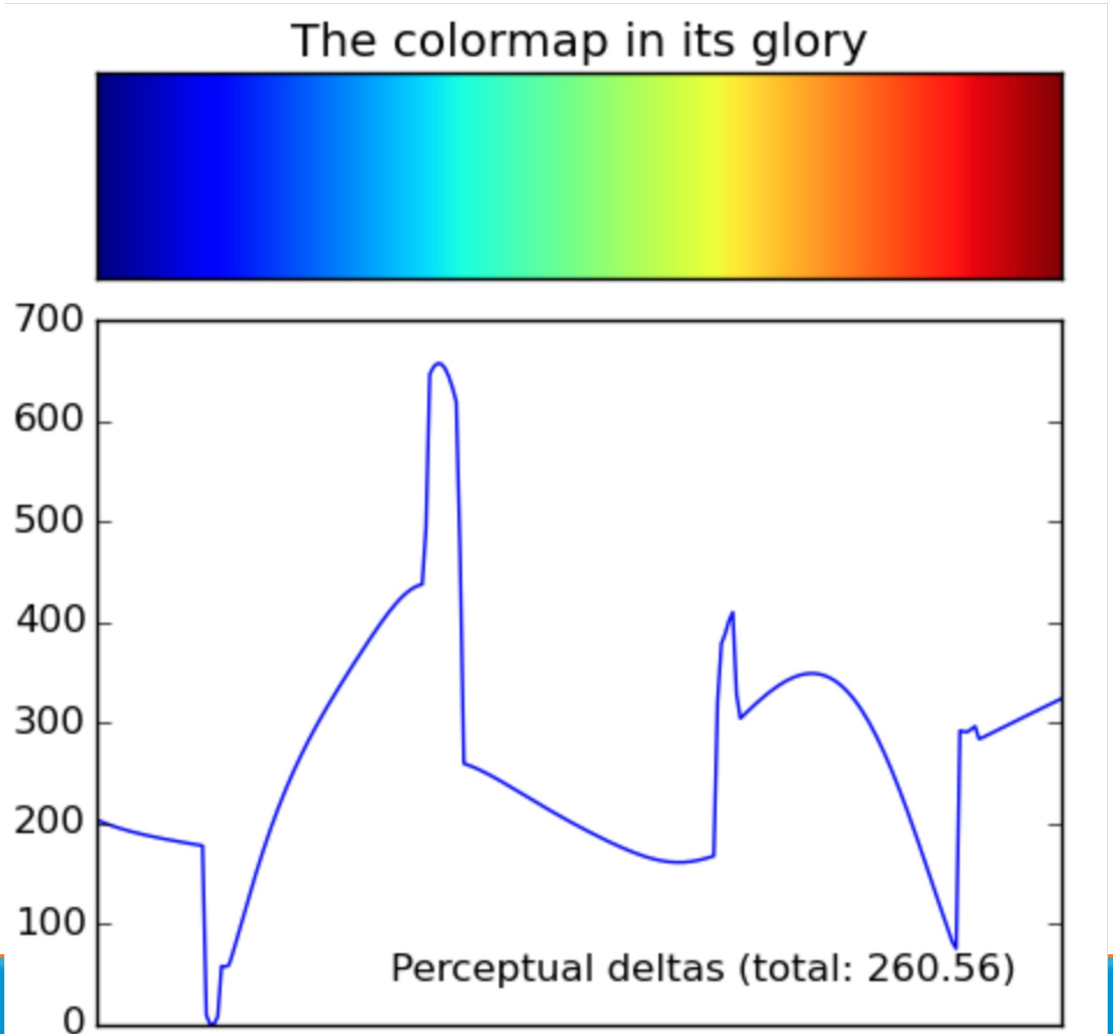


Use a Perceptually Uniform Color Map

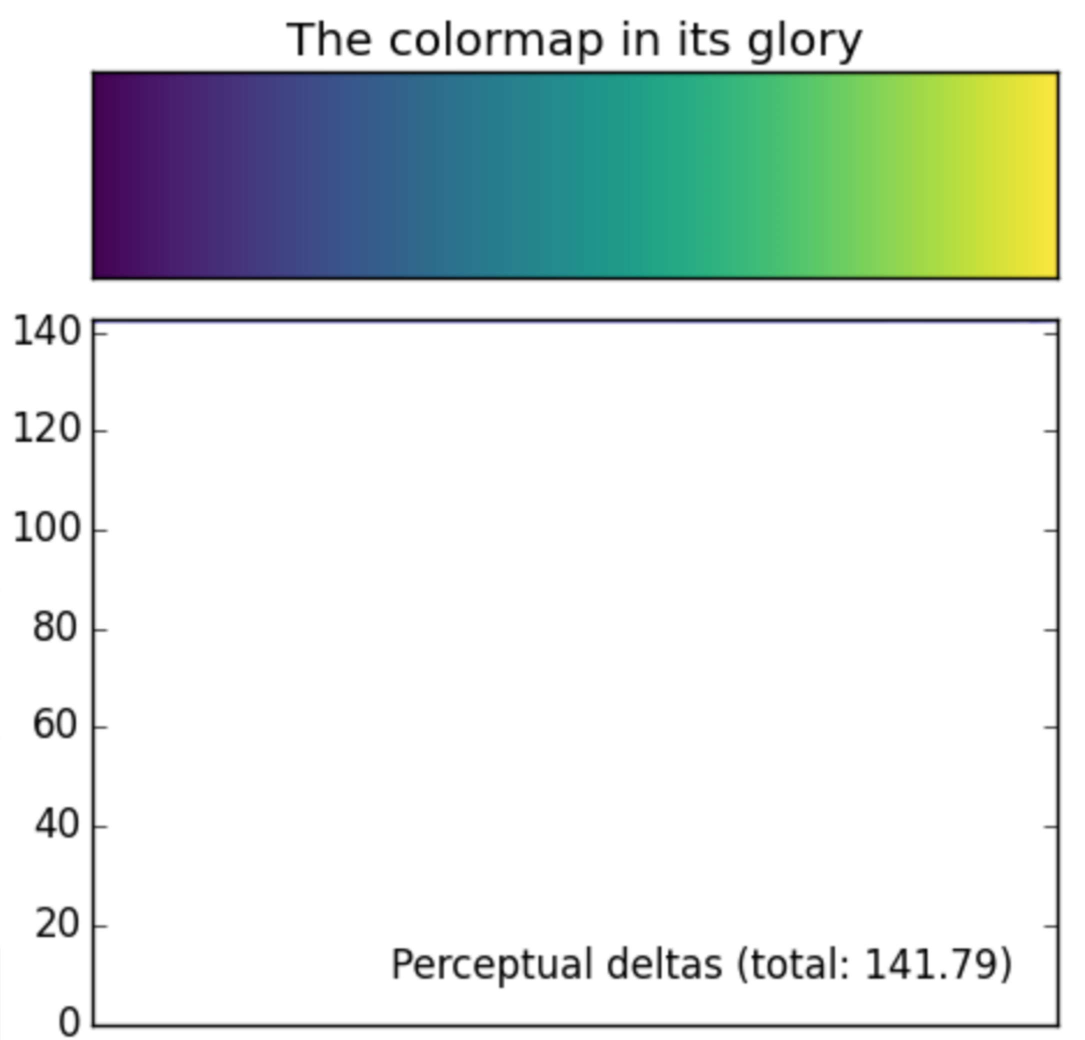
- Perceptually uniform:
  - Changing data from 0.1 to 0.2 appears similar to change from 0.8 to 0.9.
  - Measure by running experiments on people!
- Jet, the old matplotlib default, was far from uniform!
- Now fixed in MPL: https://bids.github.io/colormap/ (Eric Firing et al.)
- Also, avoid red + green since many people are colorblind

Use a Perceptually Uniform Color Map

Jet Colormap

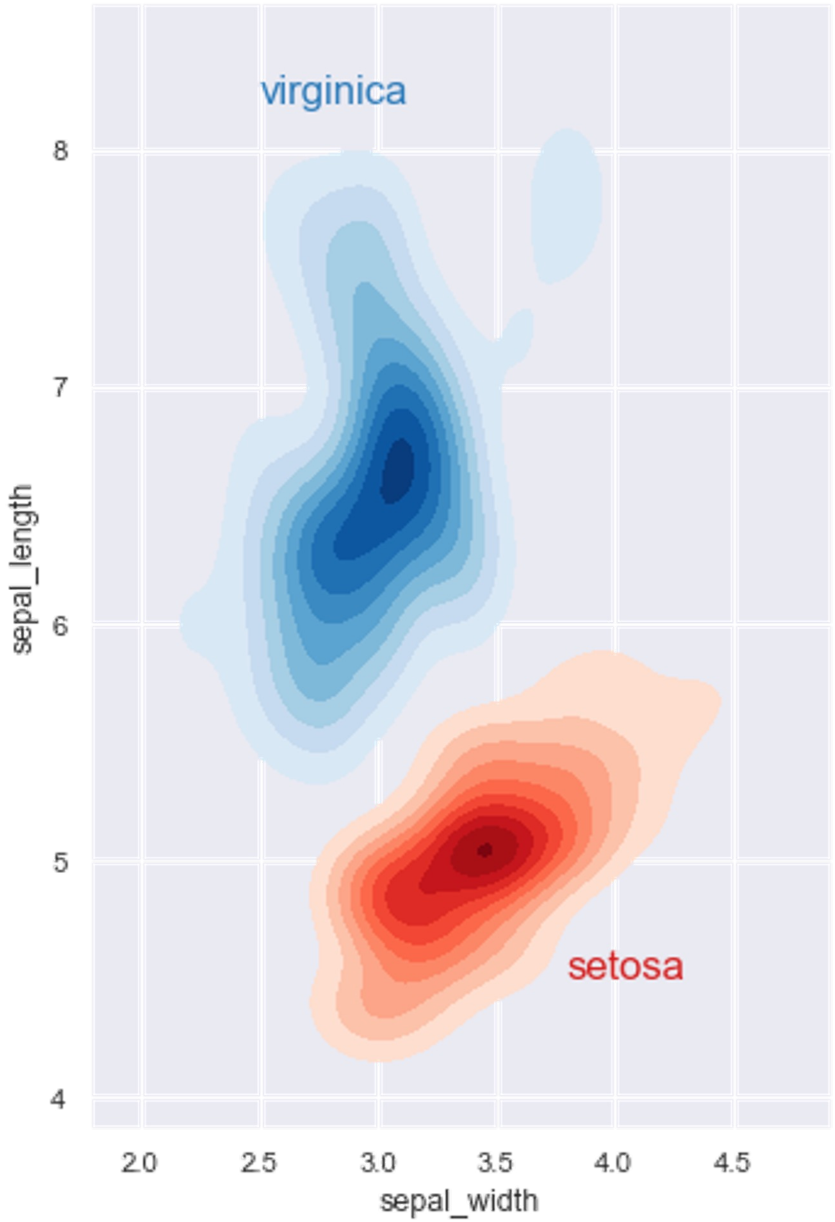


Viridis Colormap



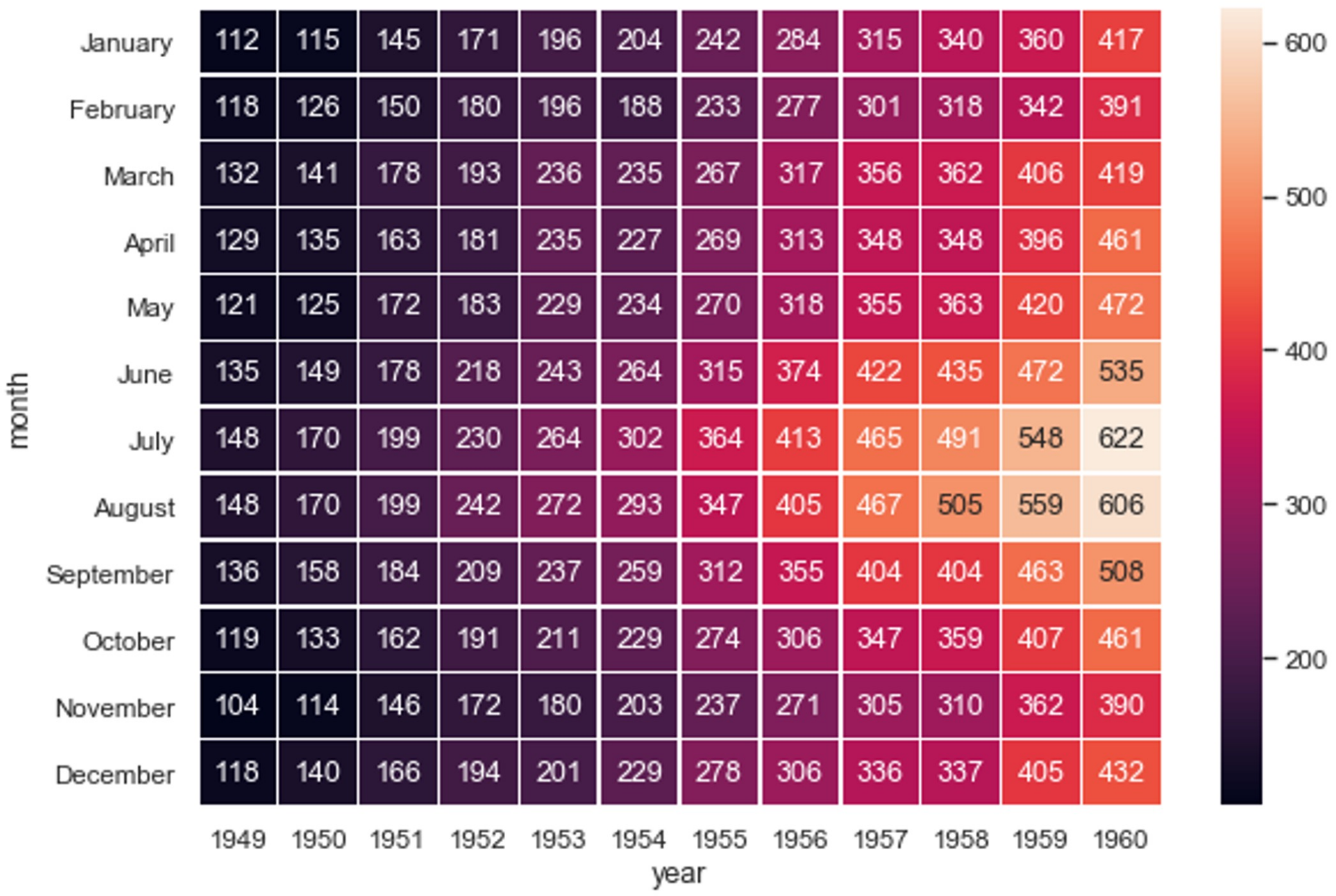
Use Color to Highlight Data Type

- Qualitative: Choose a qualitative scheme that makes it easy to distinguish between categories
- Quantitative: Choose a color scheme that implies magnitude.
- Plot on right has both!



Use Color to Highlight Data Type

- Does the data progress from low to high?
- Use a sequential scheme where light colors are for more extreme values



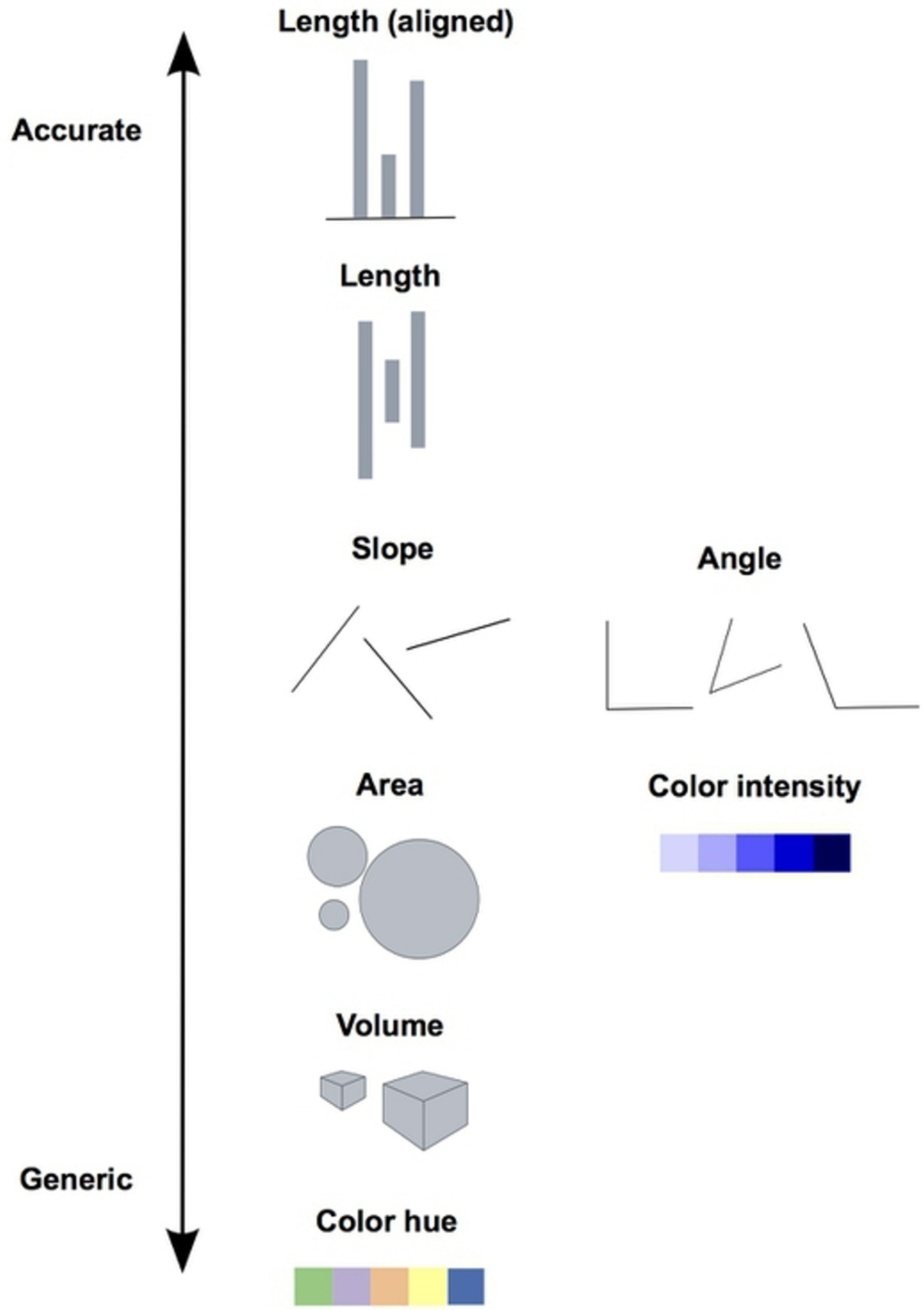
Use Color to Highlight Data Type

- Do both low and high value deserve equal emphasis? Use a diverging scheme where light colors represent middle values

```
sns.palplot(sns.color_palette("RdBu_r", 7))
```

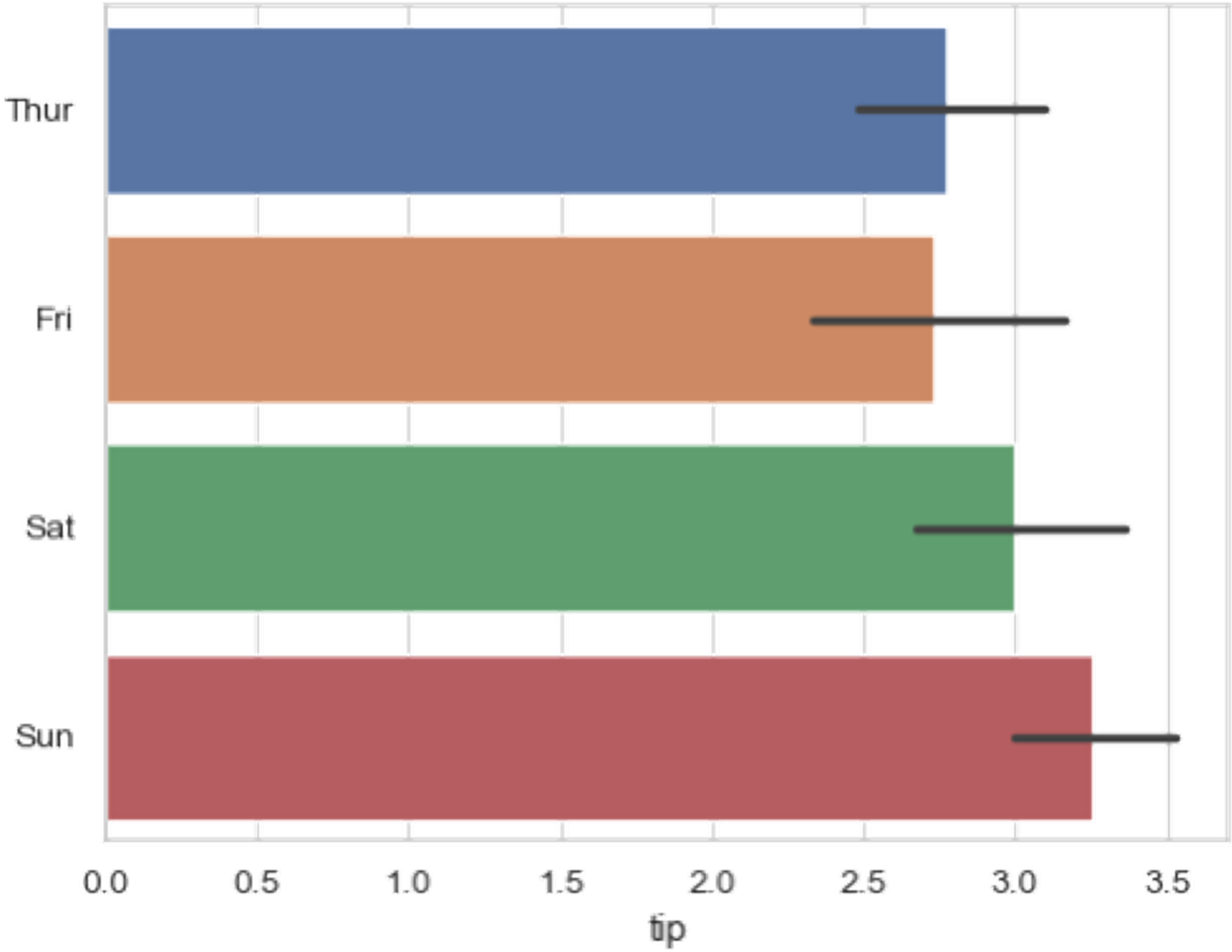
Not All Marks Are Good!

- Accuracy of judgements depend on the type of mark
- Aligned lengths most accurate
- Color least accurate



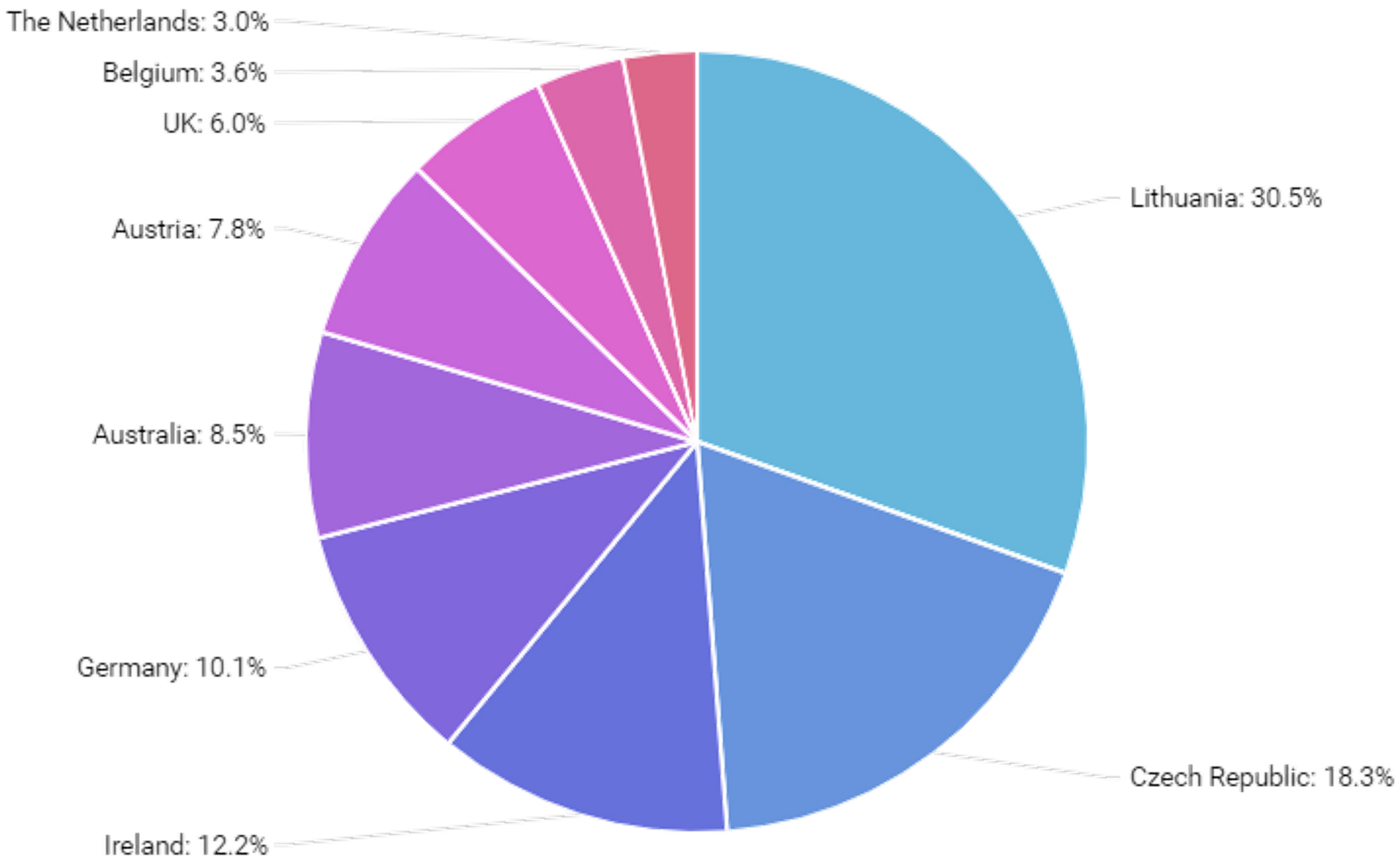
Lengths are Easy to Understand

- People can easily distinguish two different lengths
- E.g. Heights of bars in bar chart

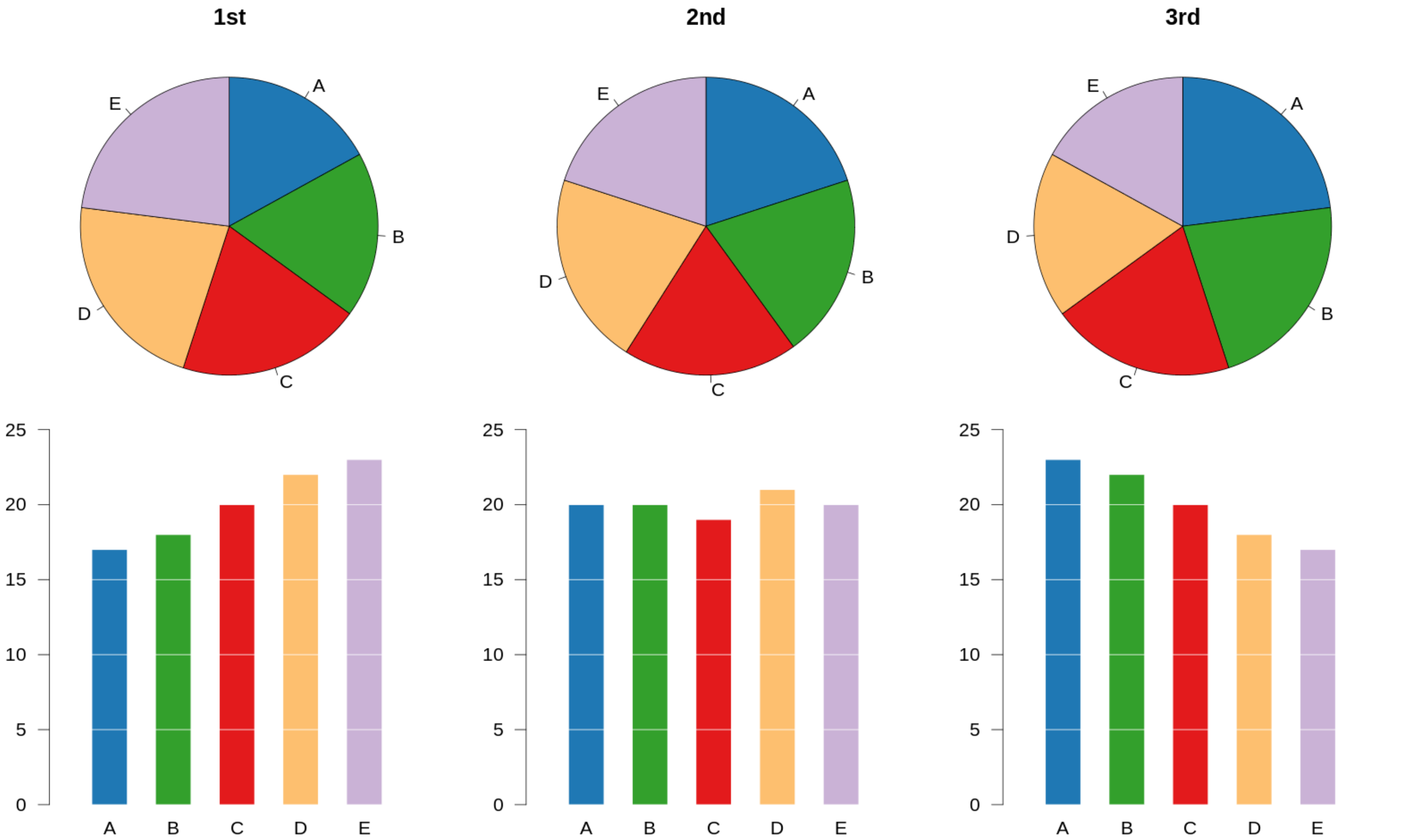


Angles are Hard to Understand

- Avoid pie charts!
- Angle judgements are inaccurate
- In general, underestimate size of larger angle

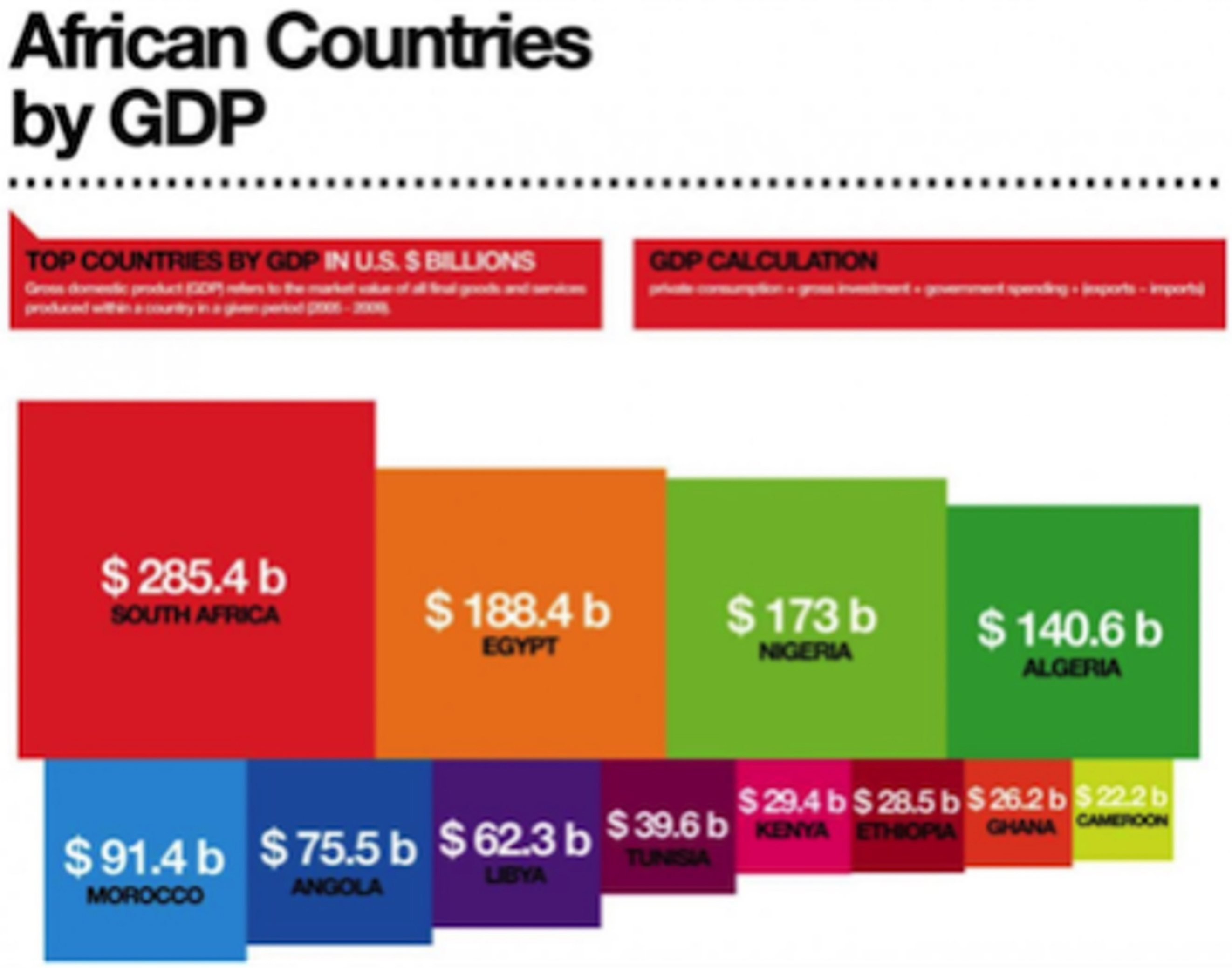


Bar vs Pie Chart



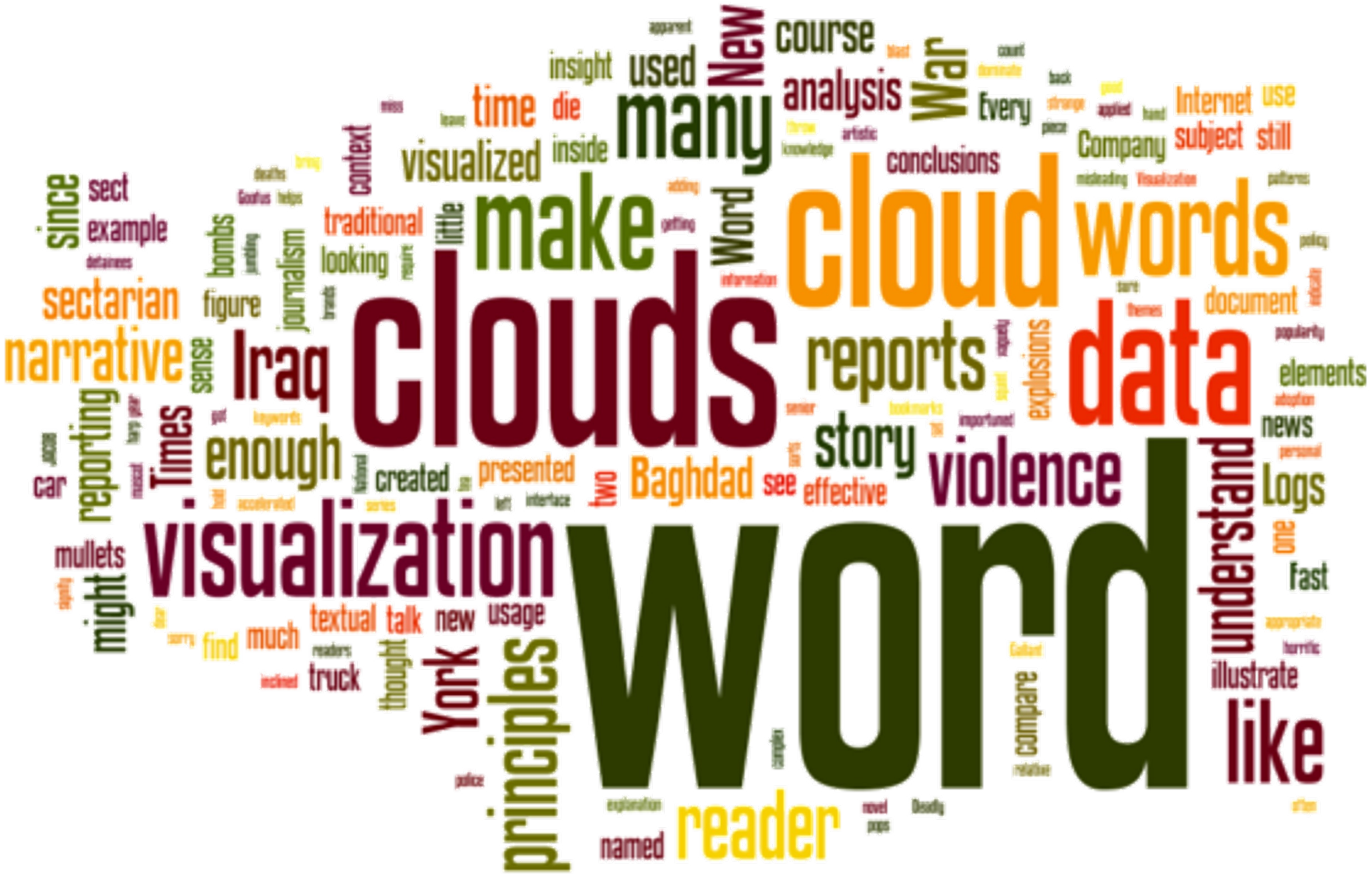
Areas are Hard to Understand

- Avoid area charts!
- Area judgements are inaccurate
- In general, underestimate size of larger area



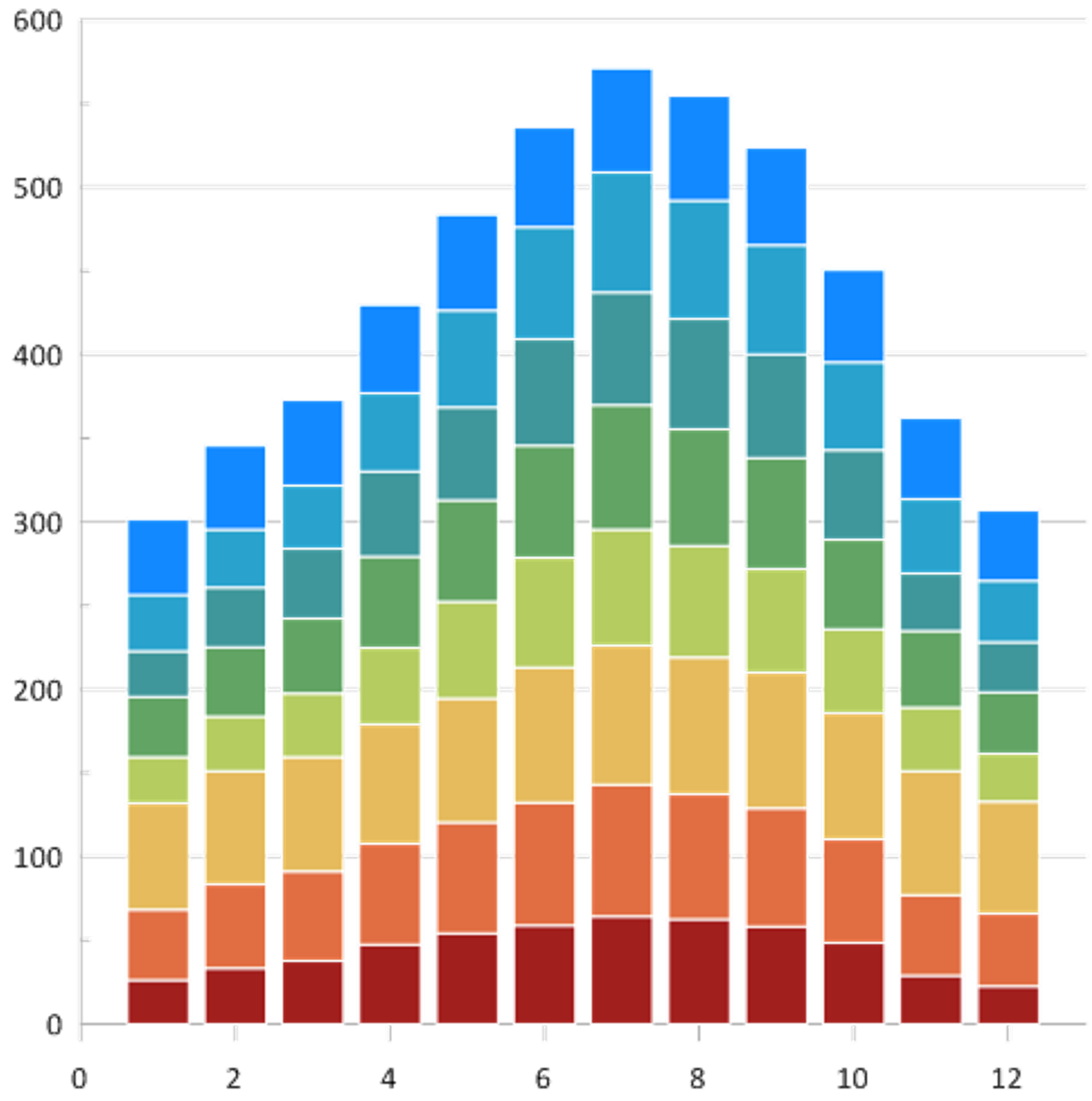
Areas are Hard to Understand

- Avoid word clouds!
- Hard to tell the "area" taken up by a word



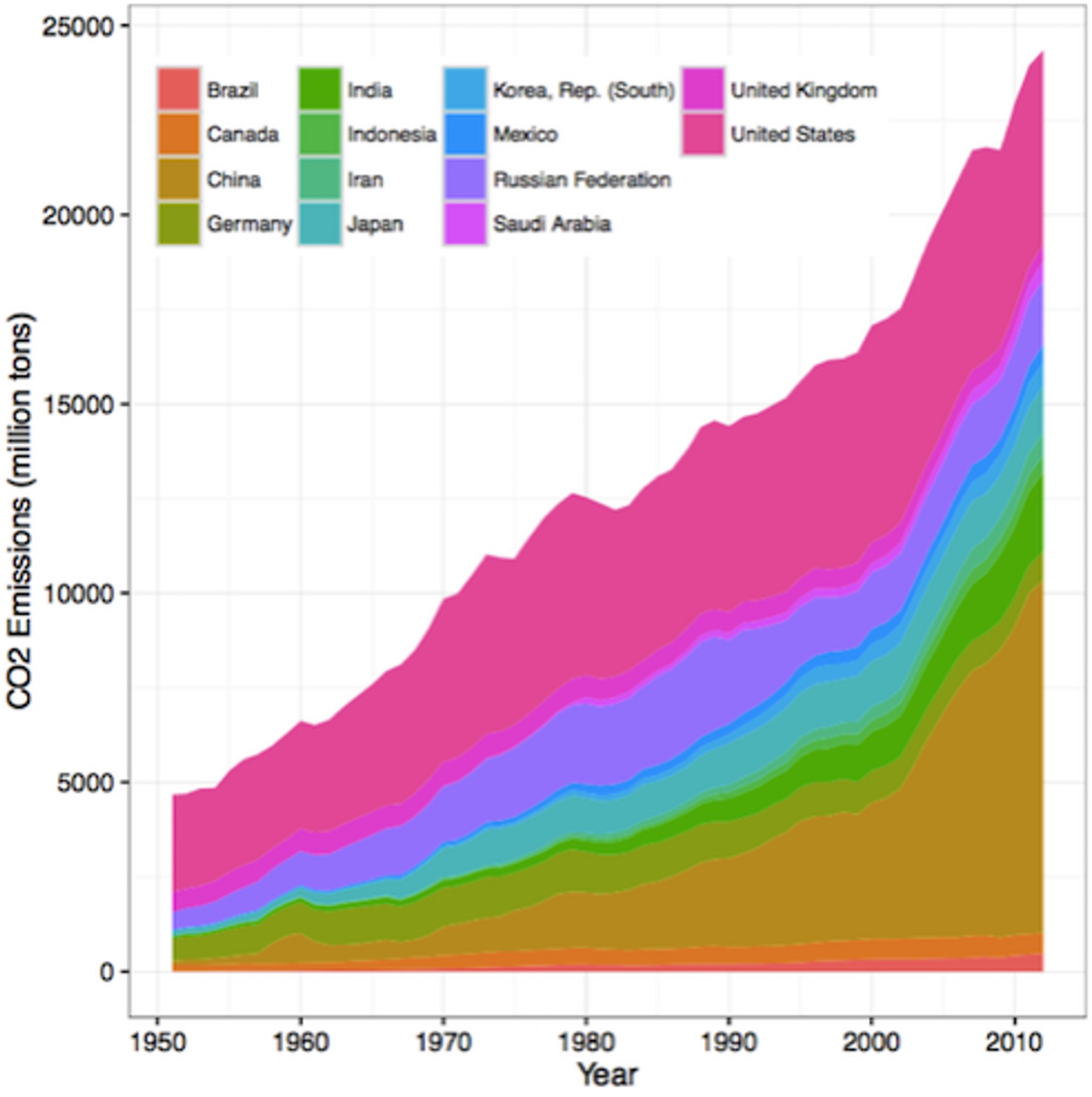
Avoid Jiggling Baseline

- Stacked bar charts / histograms hard to read because baseline moves
- Notice that top bars are all about the same height



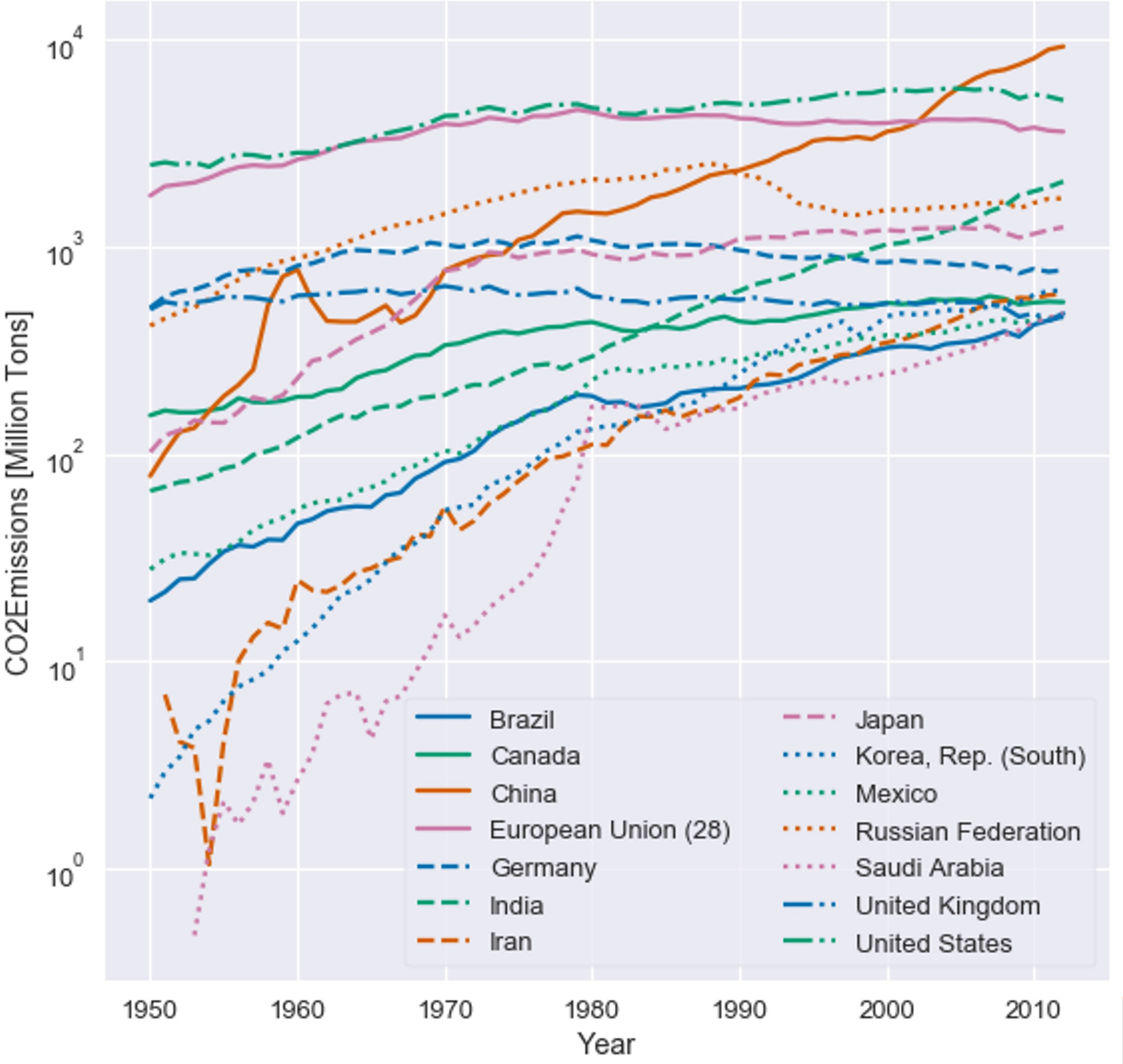
Avoid Jiggling Baseline

- Stacked area charts hard to read because baseline moves



Avoid Jiggling Baseline

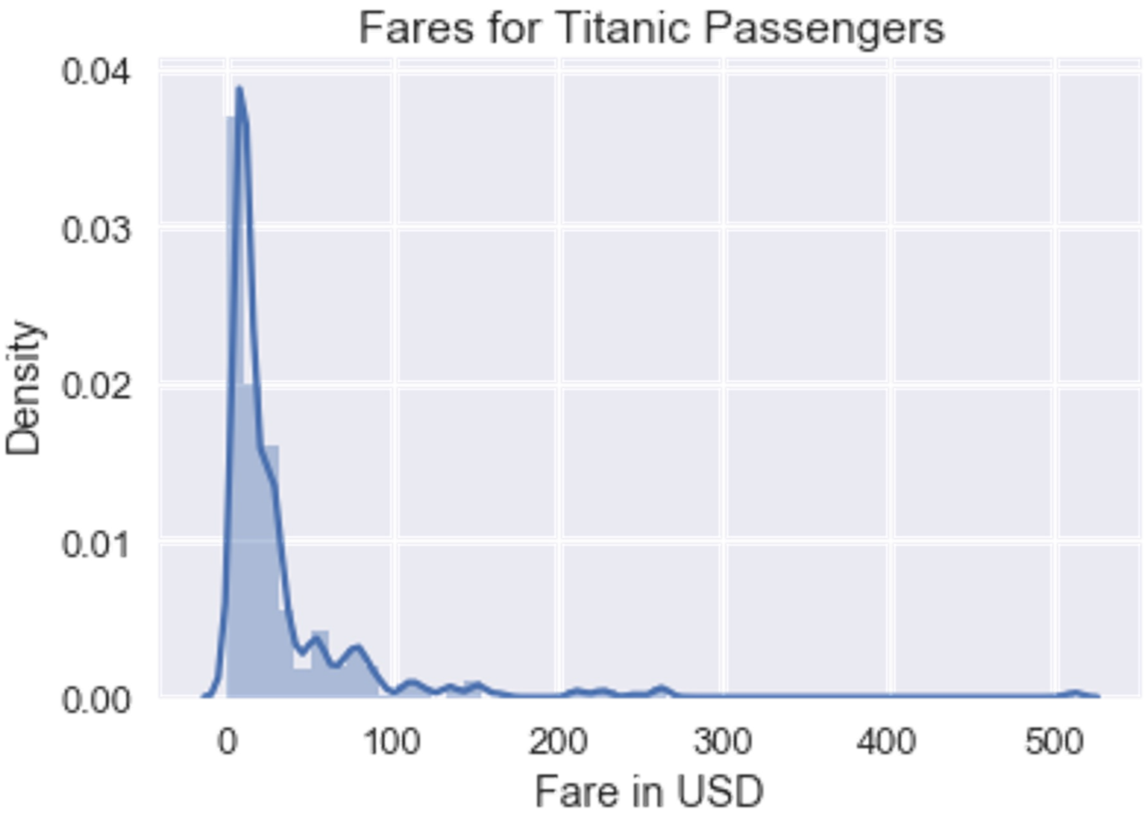
- Instead, plot lines themselves

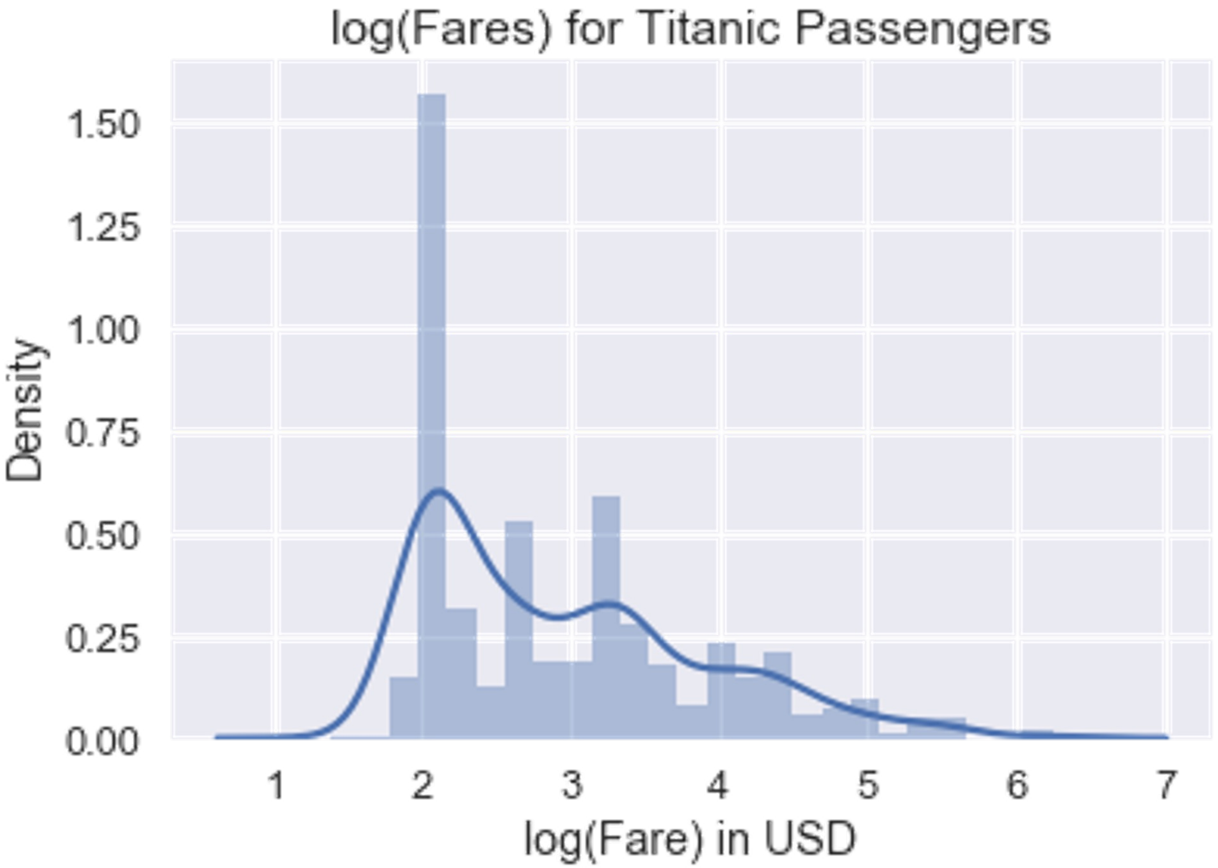


Principles of Transformation

Transforming Data Can Reveal Patterns

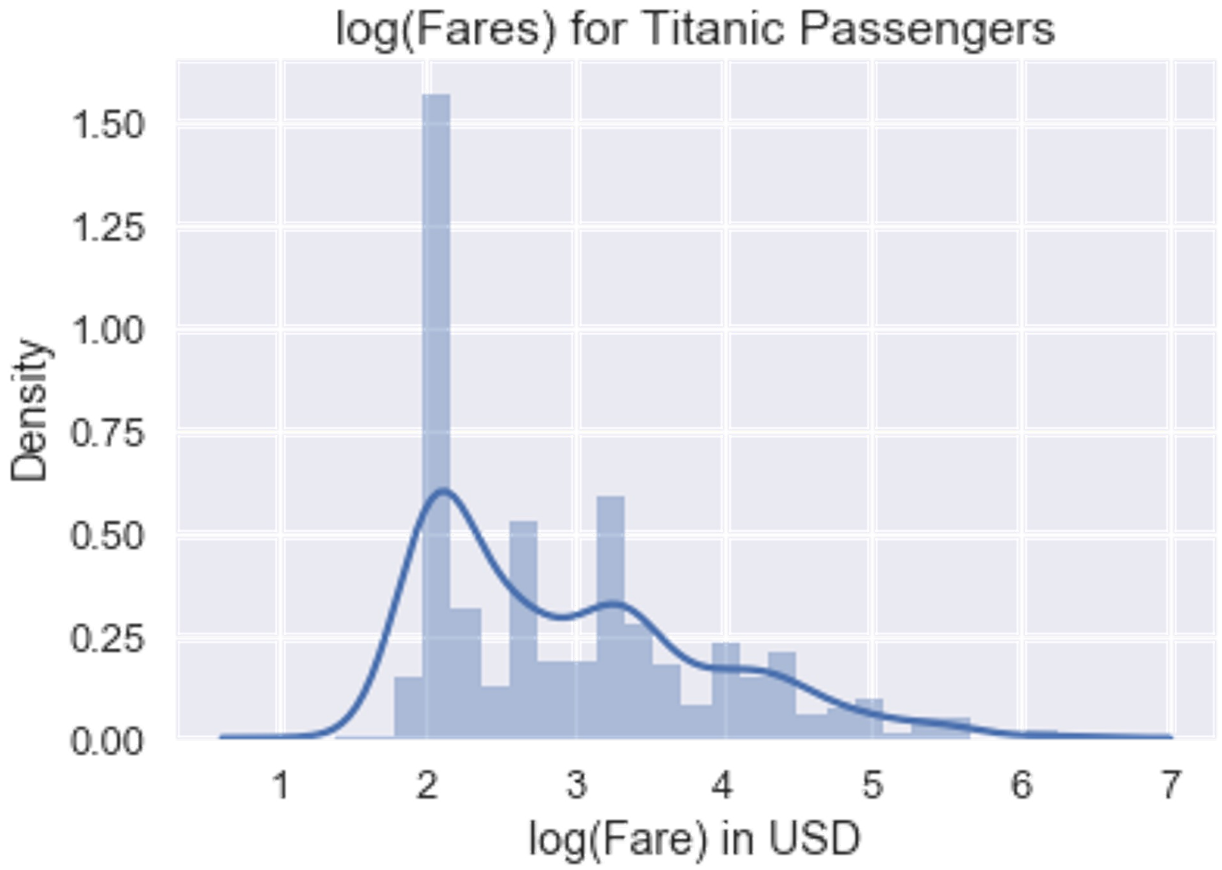
- When data are heavy tailed, useful to take the log and replot





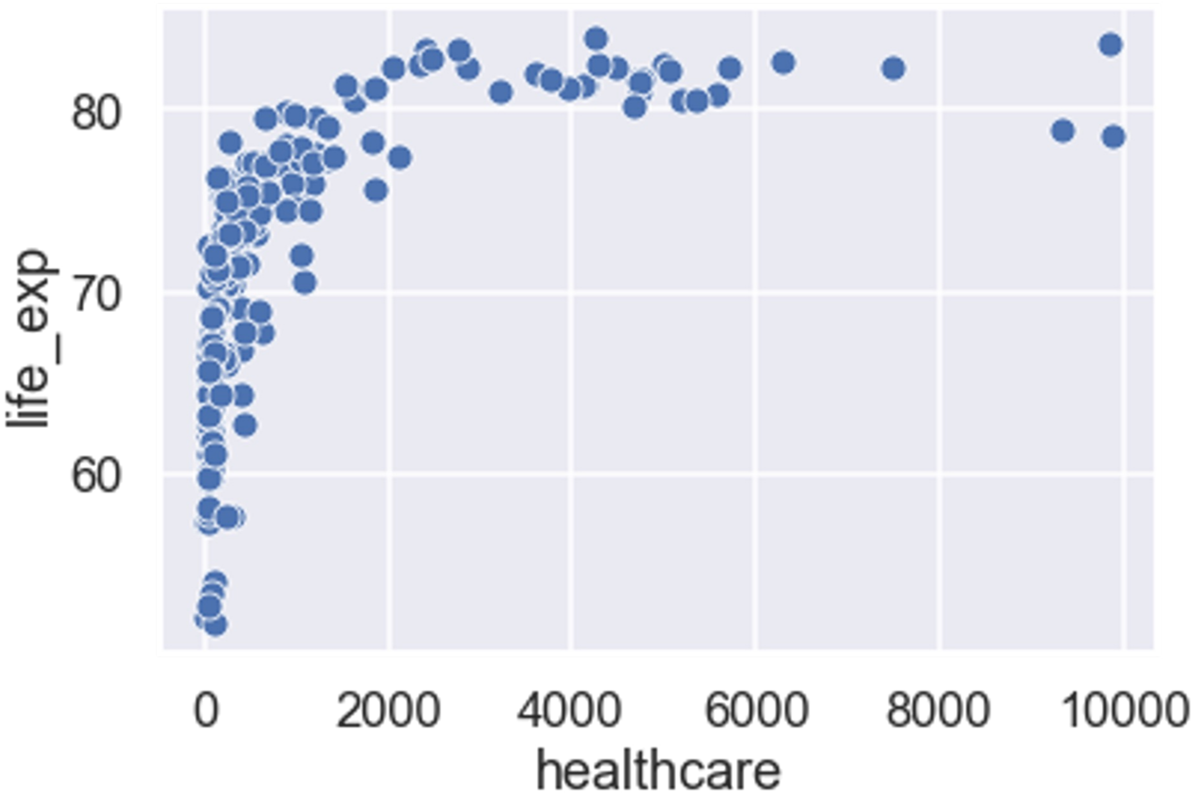
Transforming Data Can Reveal Patterns

- Shows a mode when log(fare) = 2 and a smaller mode at 3.4.
- What do these correspond to in actual dollars?
- exp(2) = $7.4
- exp(3.4) = $30



Transforming Data Can Reveal Patterns

- Log of nonlinear data can reveal pattern in scatter plot!

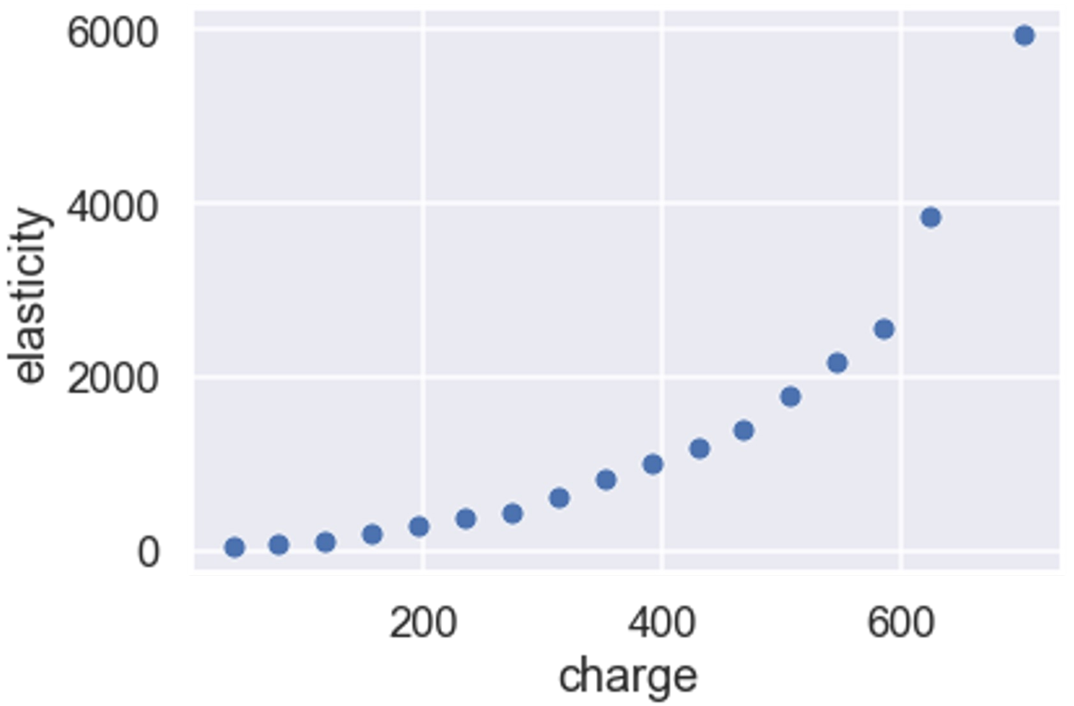


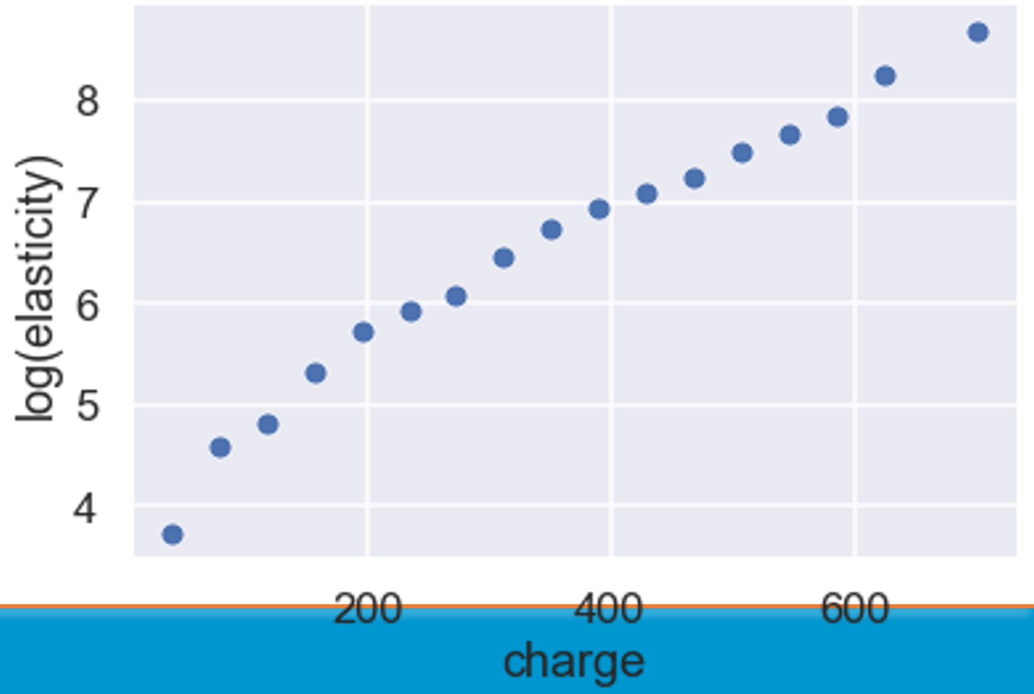
Log of y-values

- Fit line to log of y-values:

$$\log y = ax + b$$
$$y = e^{ax+b}$$
$$y = e^{ax}e^{b}$$
$$y = Ce^{ax}$$

- Linear relationship after log of y-values implies exponential model for original plot





Log of both x and y-values

- Fit line to log of x and y-values:

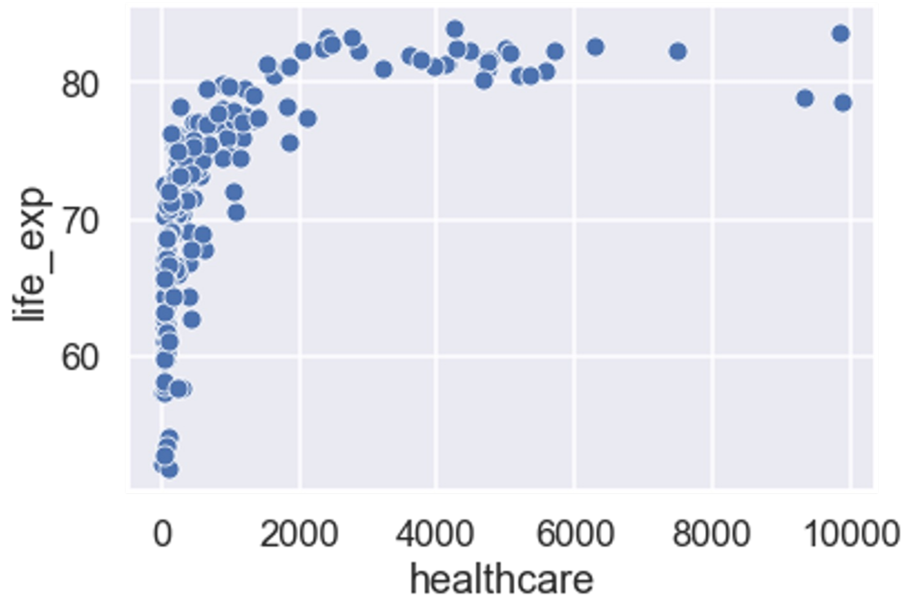
$$\log y = a \cdot \log x + b$$

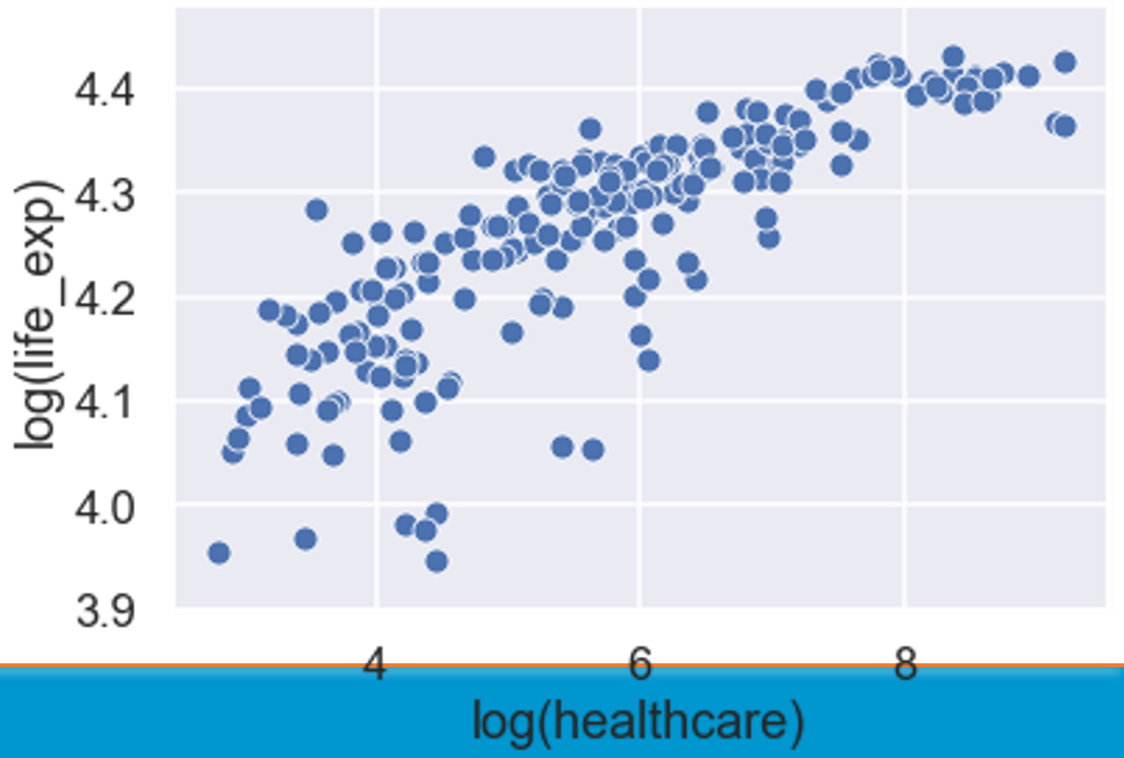
$$y = e^{a \cdot \log x + b}$$

$$y = Ce^{a \cdot \log x}$$

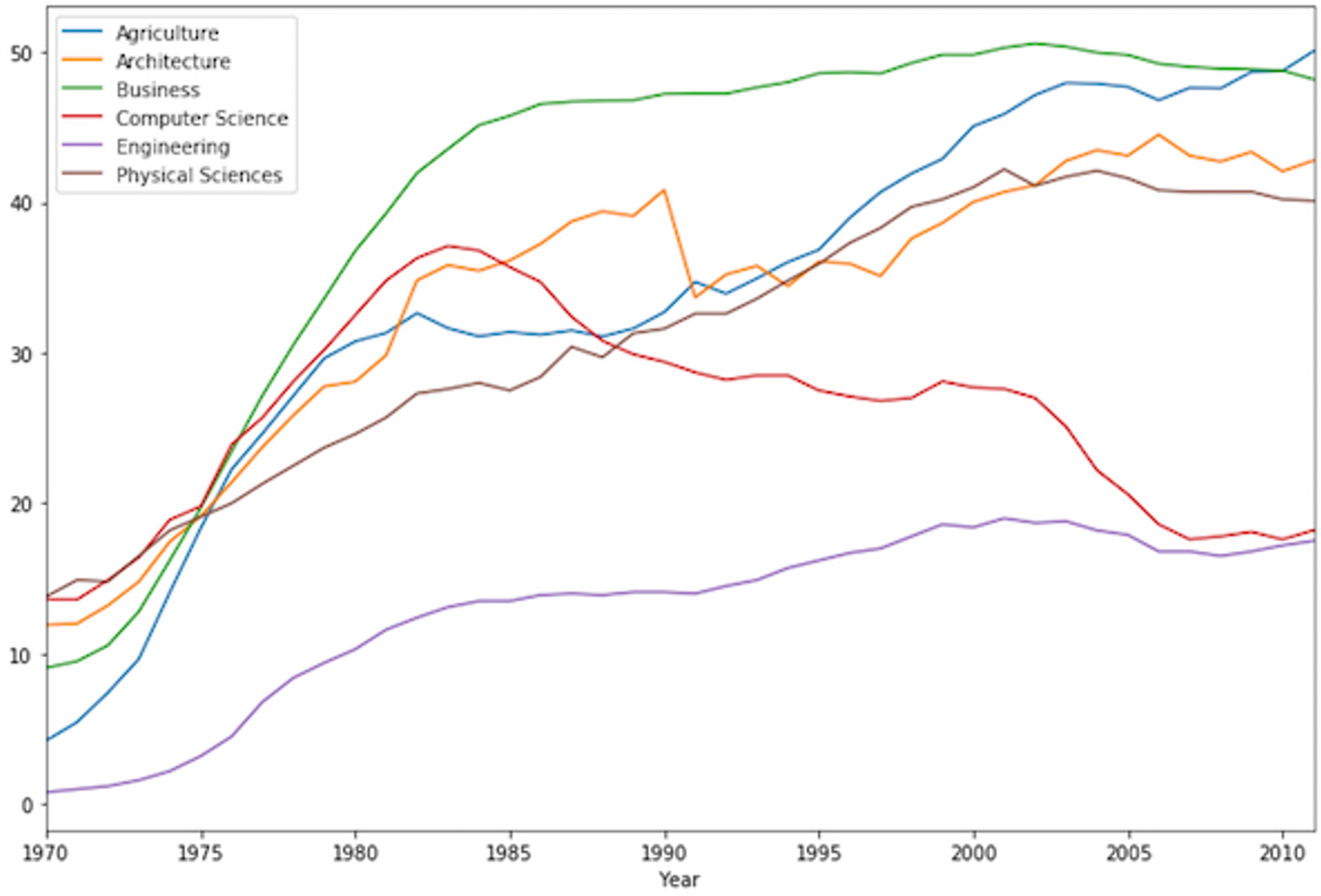
$$y = Cx^a$$

- Linear relationship after log of x and y-values implies polynomial model for original plot

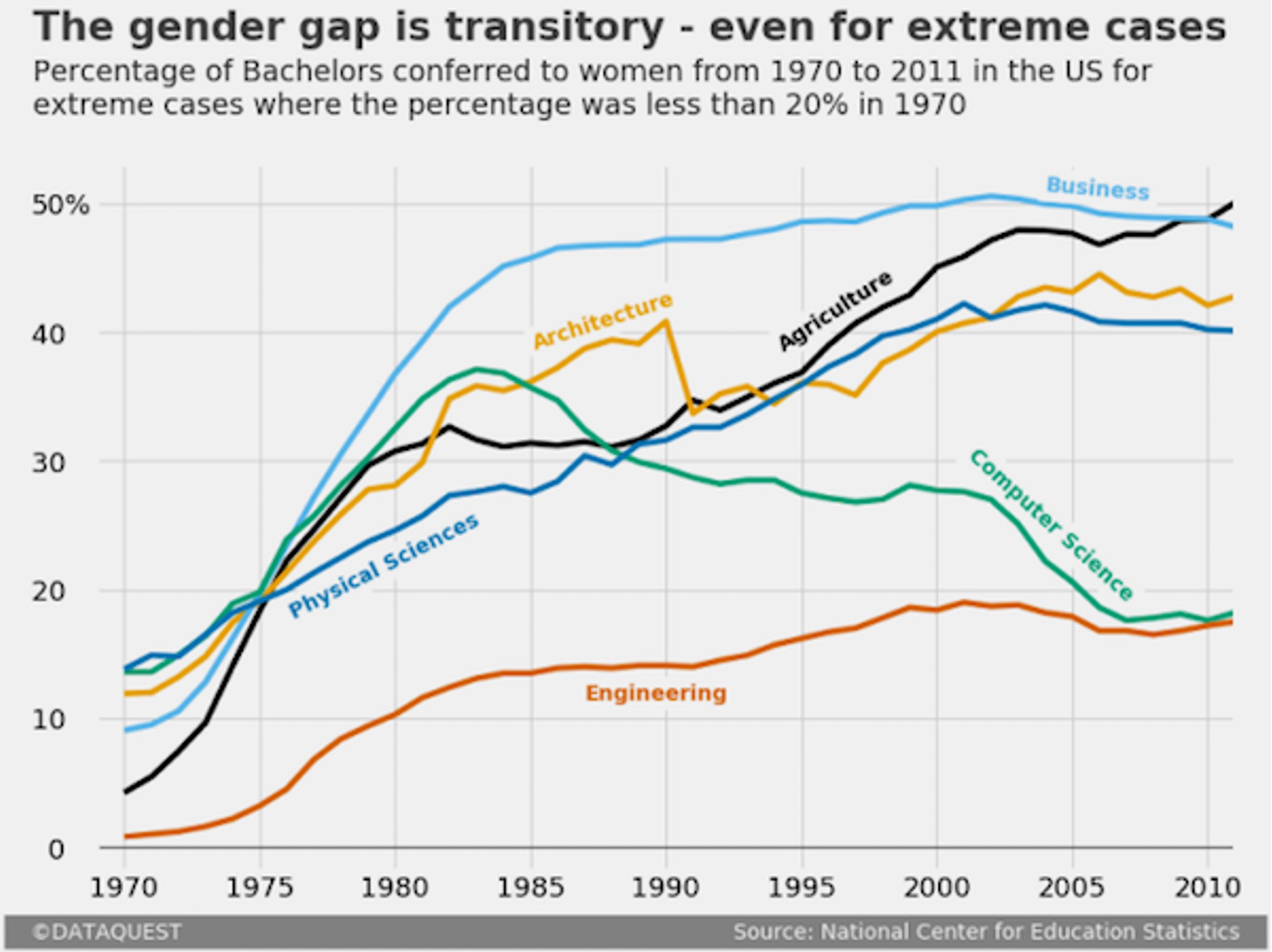




Principles of Context



Agriculture
Architecture
Business
Computer Science
Engineering
Physical Sciences
0
10
20
30
40
50
1970
1975
1980
1985
1990
1995
2000
2005
2010
Year


The gender gap is transitory - even for extreme cases
Percentage of Bachelors conferred to women from 1970 to 2011 in the US for extreme cases where the percentage was less than 20% in 1970
50%
40
30
20
10
0
1970
1975
1980
1985
1990
1995
2000
2005
2010
Business
Agriculture
Architecture
Physical Sciences
Computer Science
Engineering
©DATAQUEST
Source: National Center for Education Statistics

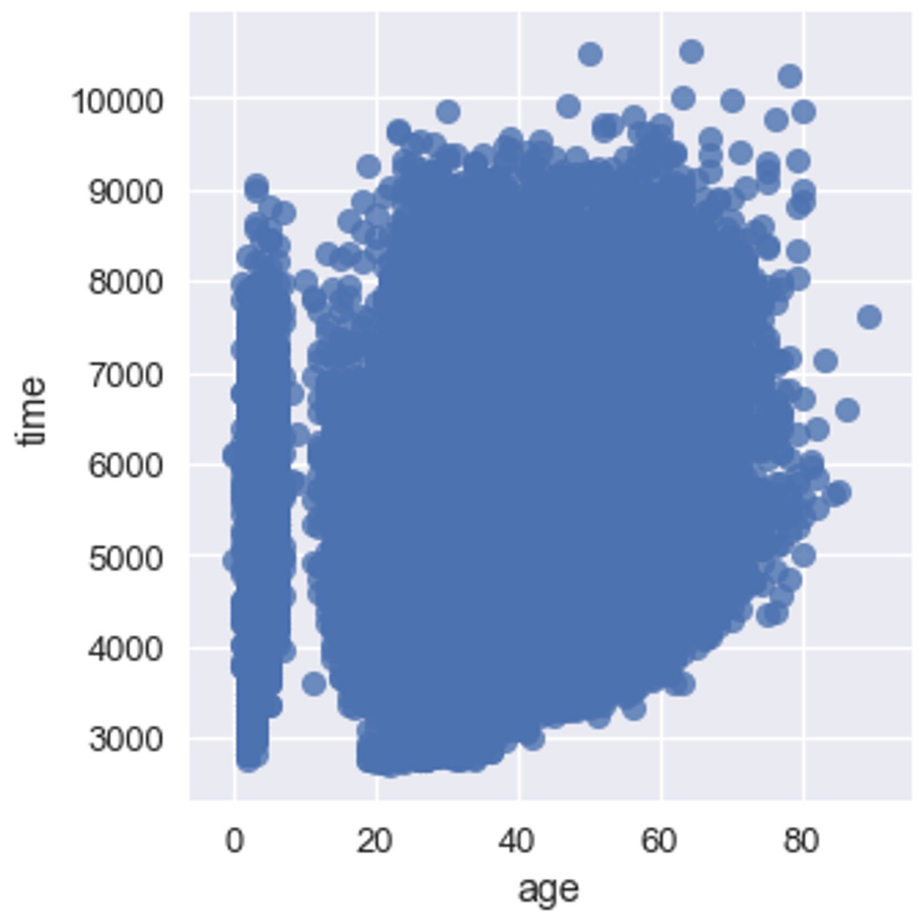
Add Context Directly to Plot

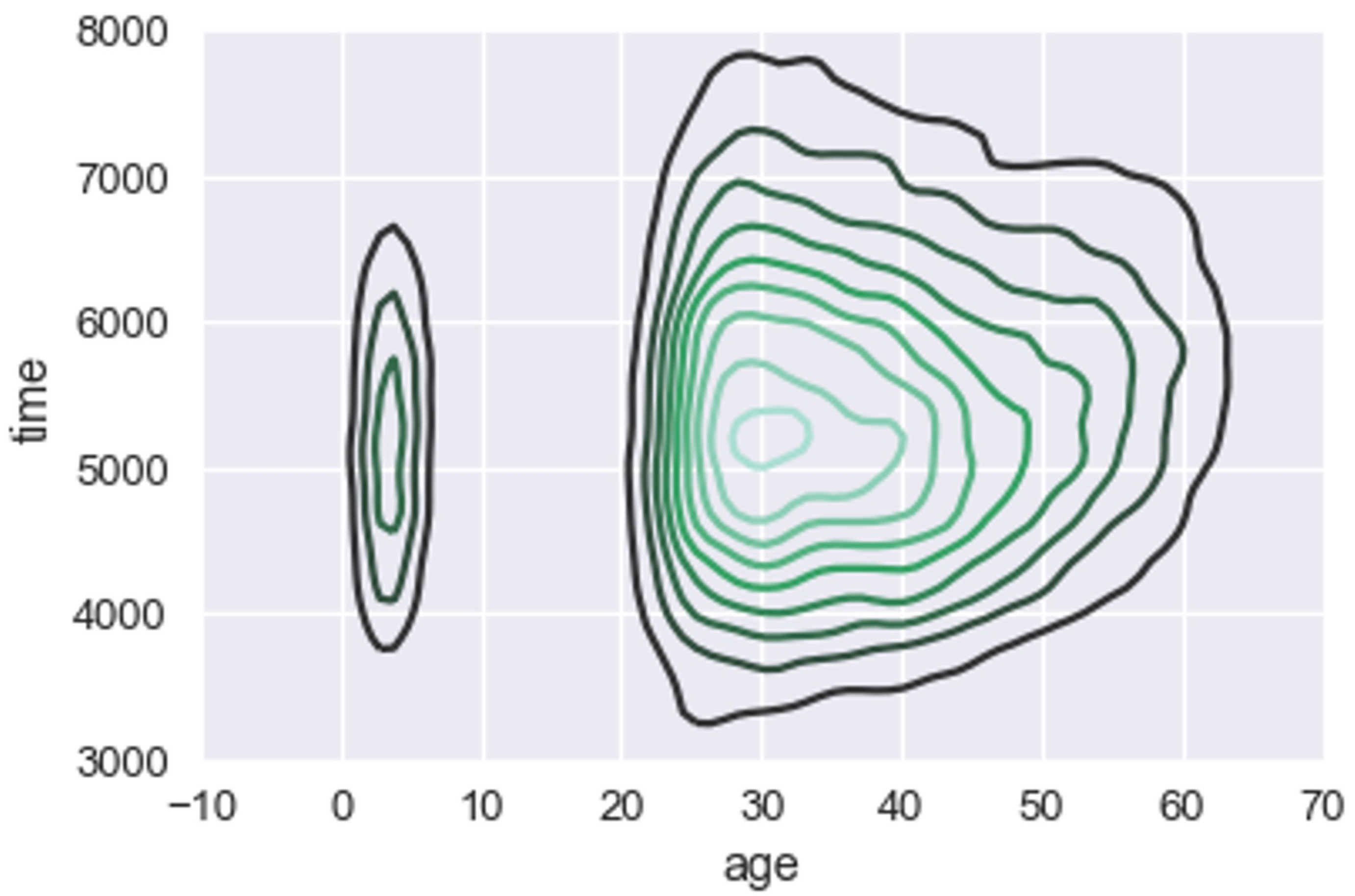
A publication-ready plot needs:

- Informative title (takeaway, not description)
  - “Older passengers spend more on plane tickets” instead of “Scatter plot of price vs. age”.
- Axis labels
- Reference lines and markers for important values
- Labels for unusual points
- Captions that describe data

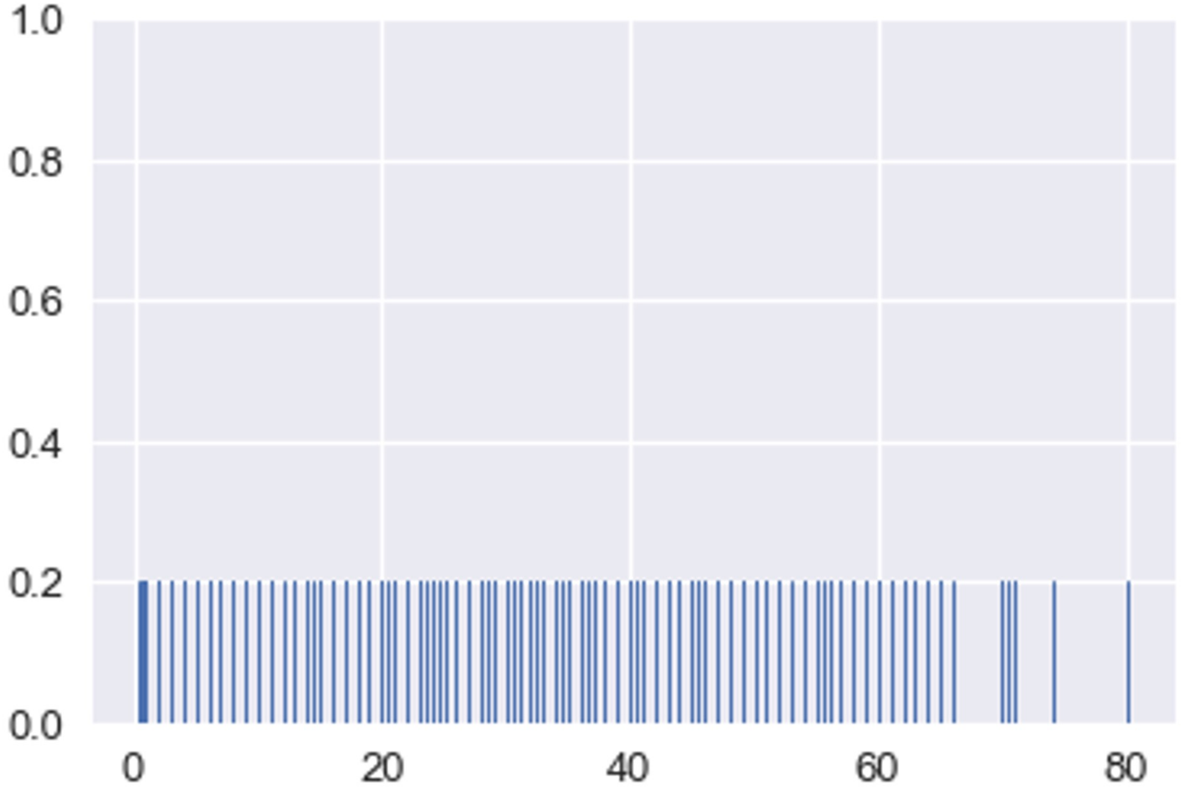
Principles of Smoothing

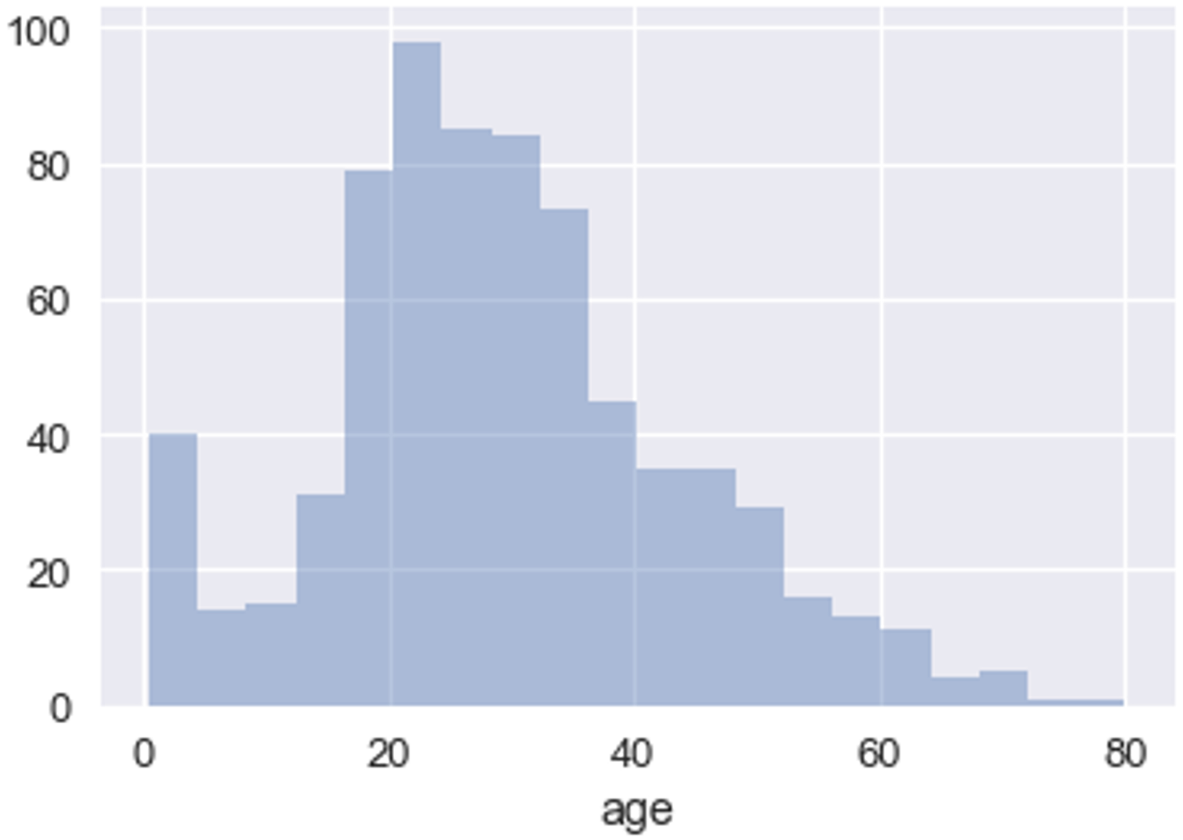
Apply Smoothing for Large Datasets



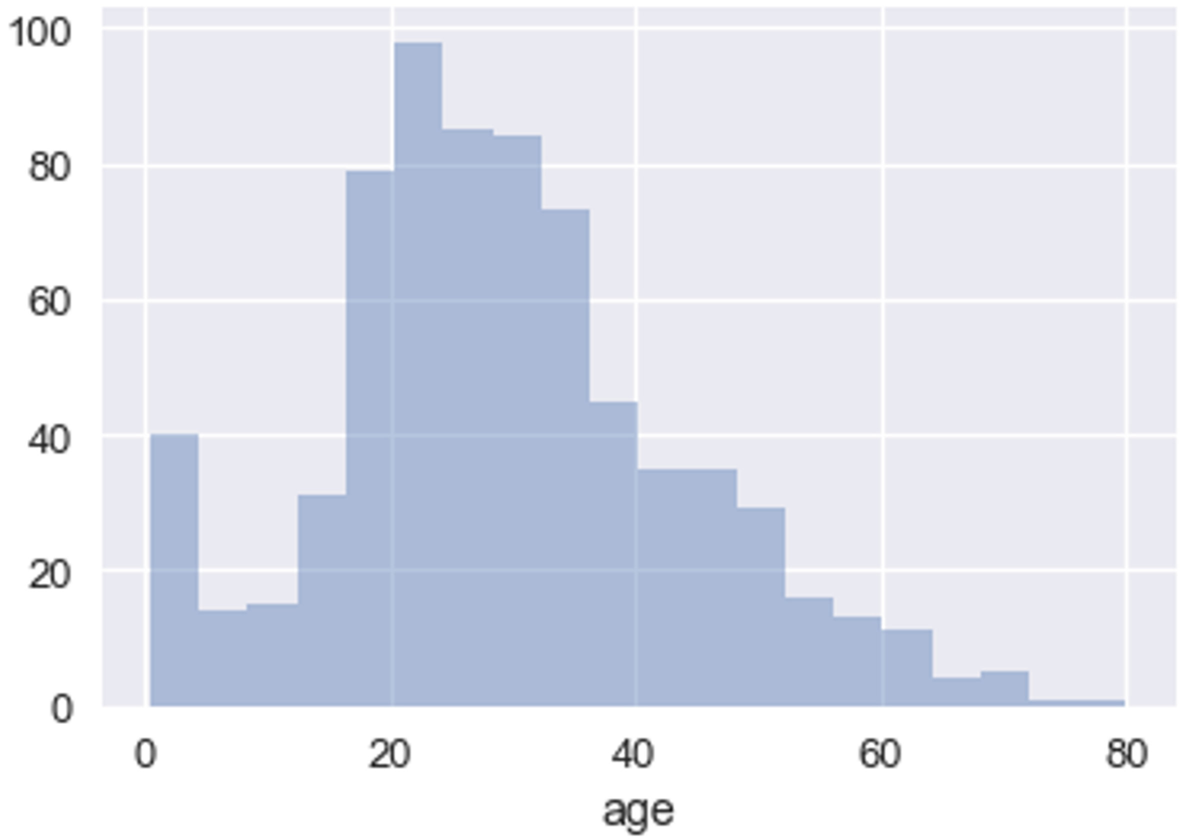


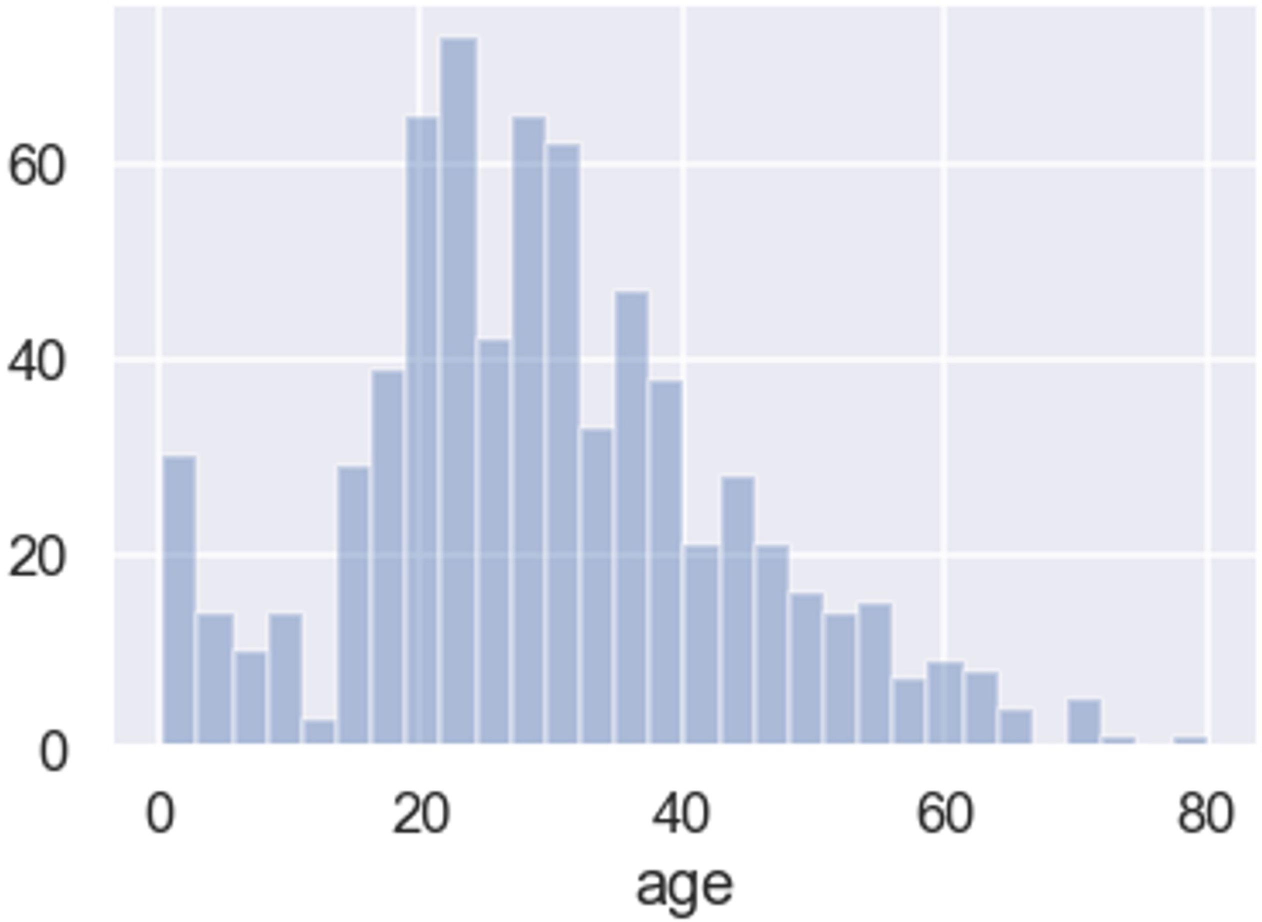
A Histogram is a Smoothed Rug Plot





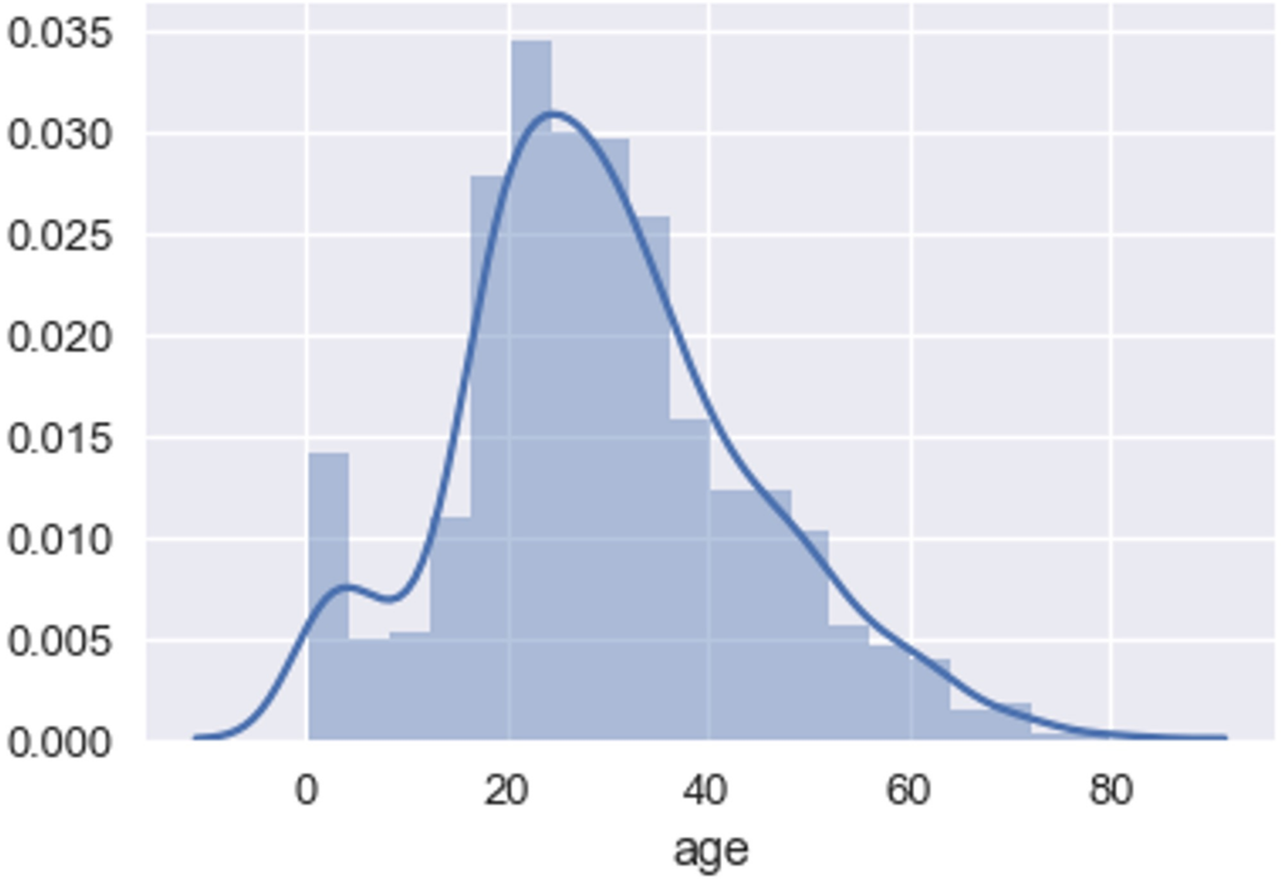
Smoothing Needs Tuning





Kernel Density Estimation (KDE)

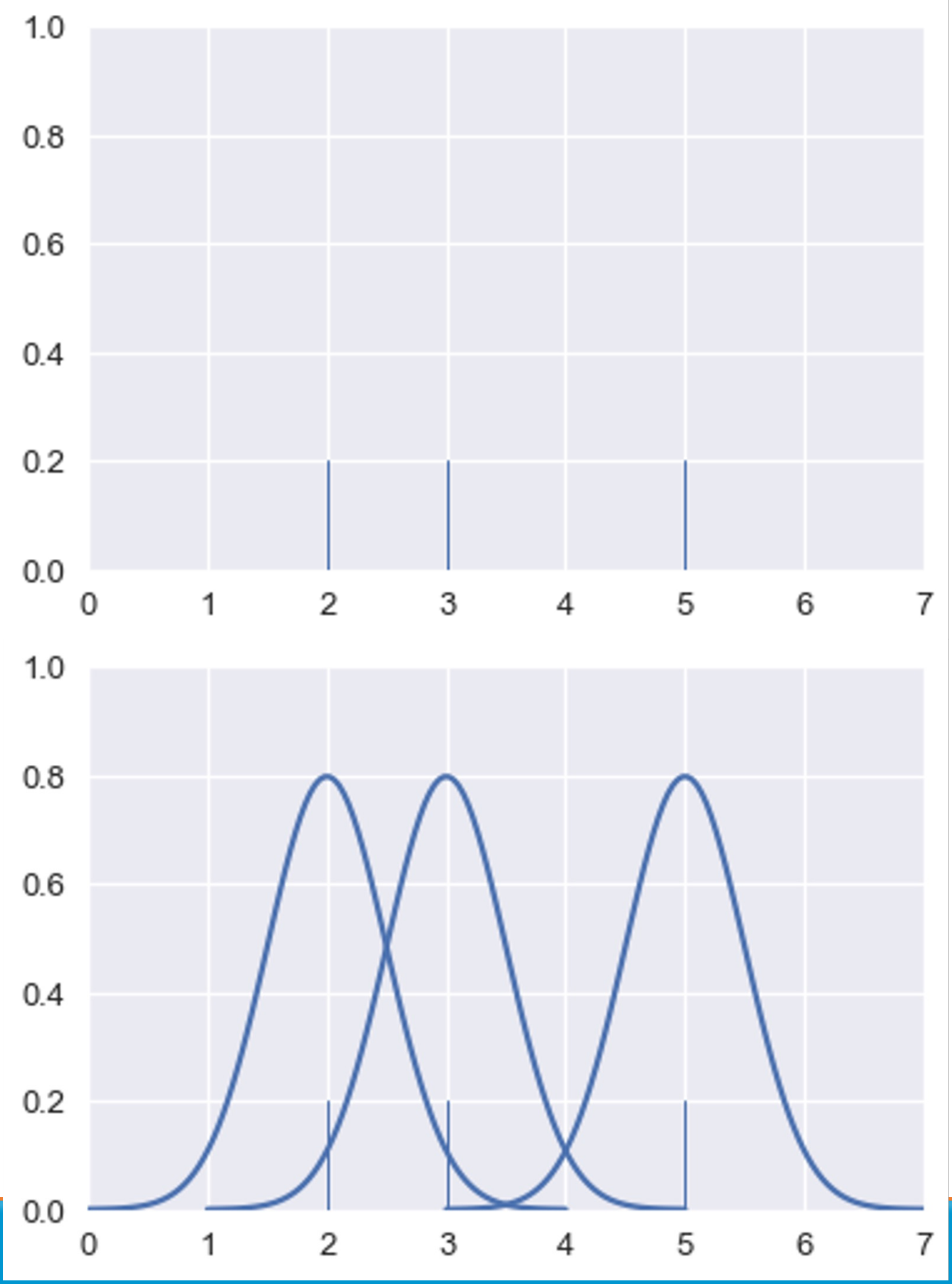
- Sophisticated smoothing technique
- Used to estimate a probability density function from a set of data



Kernel Density Estimation

Intuition:

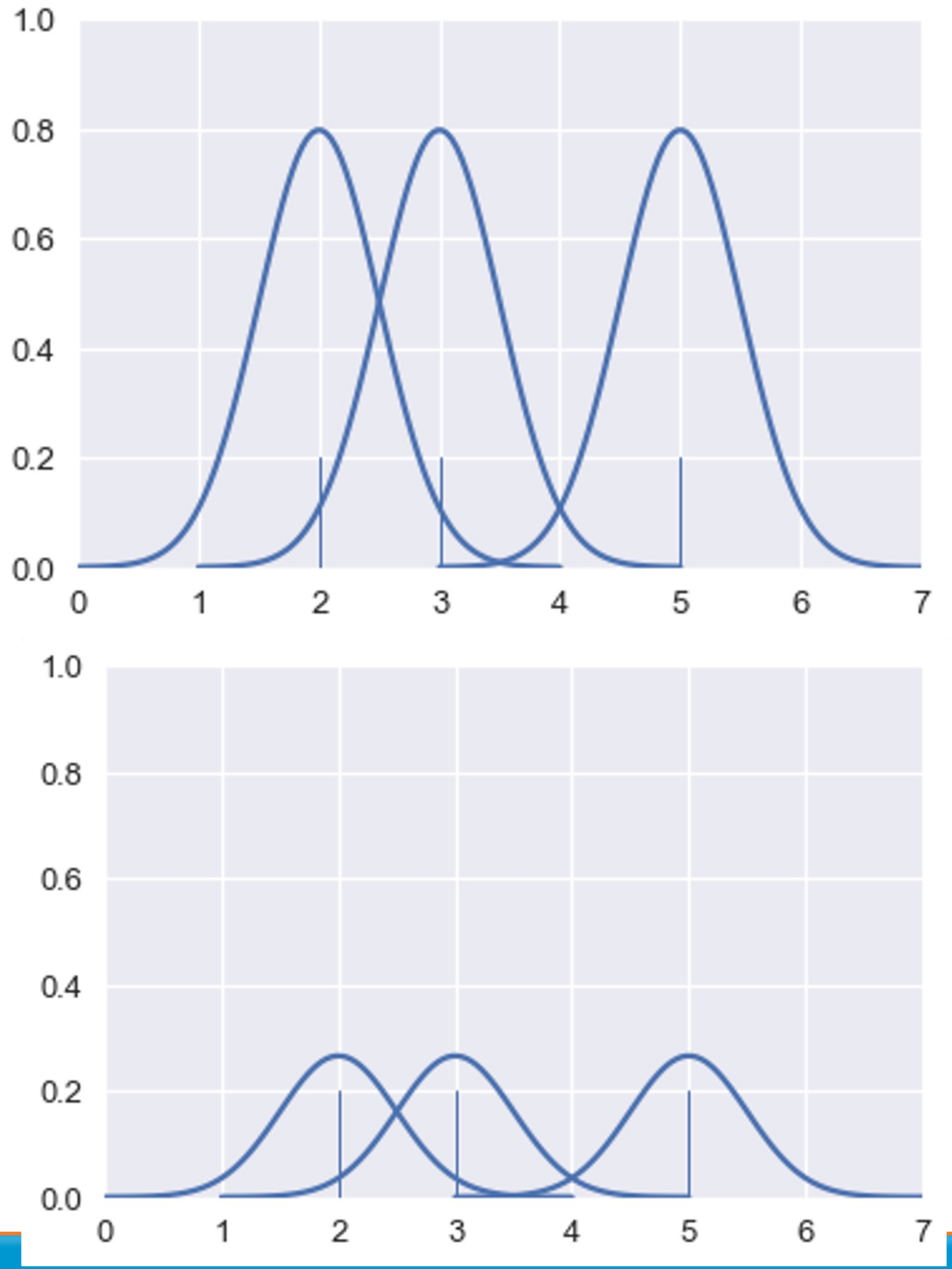
1. Place a “kernel” at each data point



Kernel Density Estimation

Intuition:

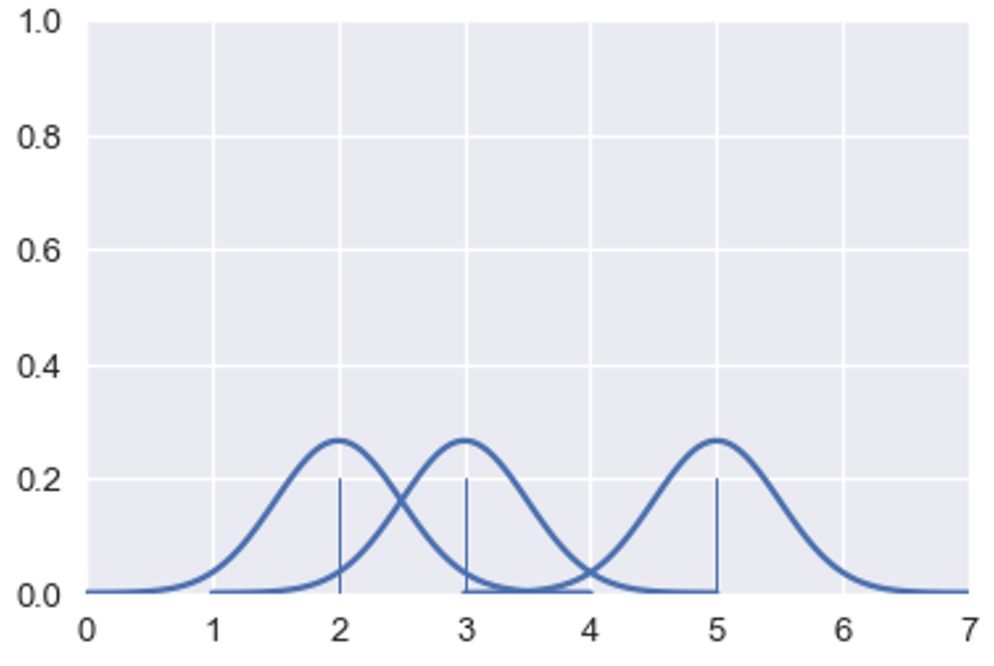
1. Place a "kernel" at each data point
2. Normalize kernels so that total area = 1

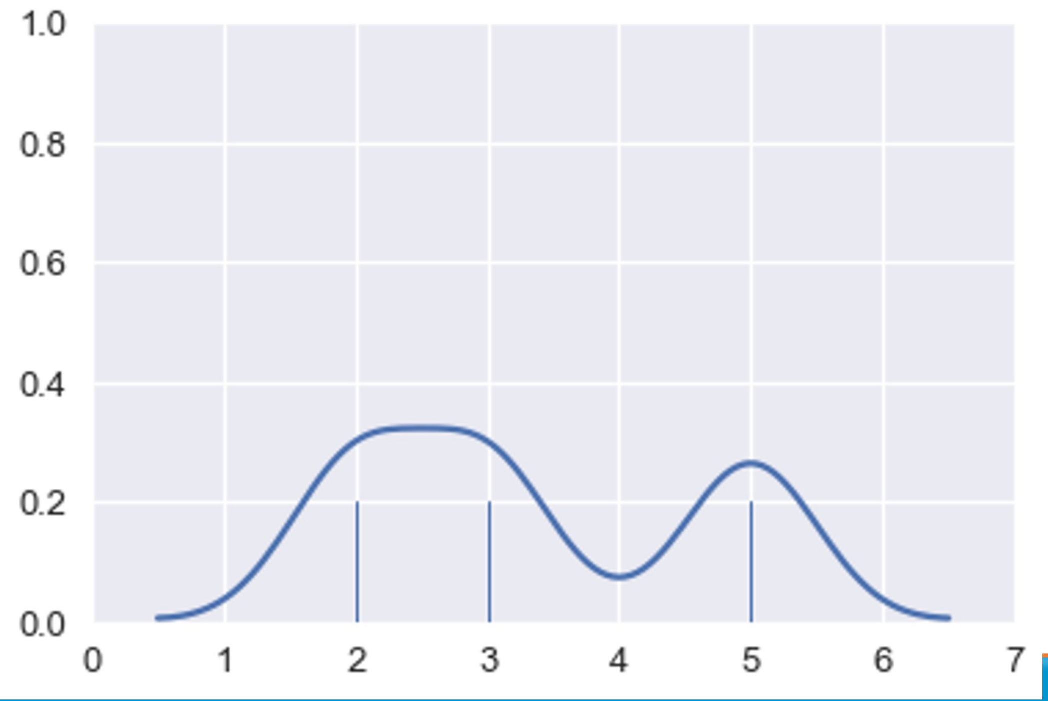


Kernel Density Estimation

Intuition:

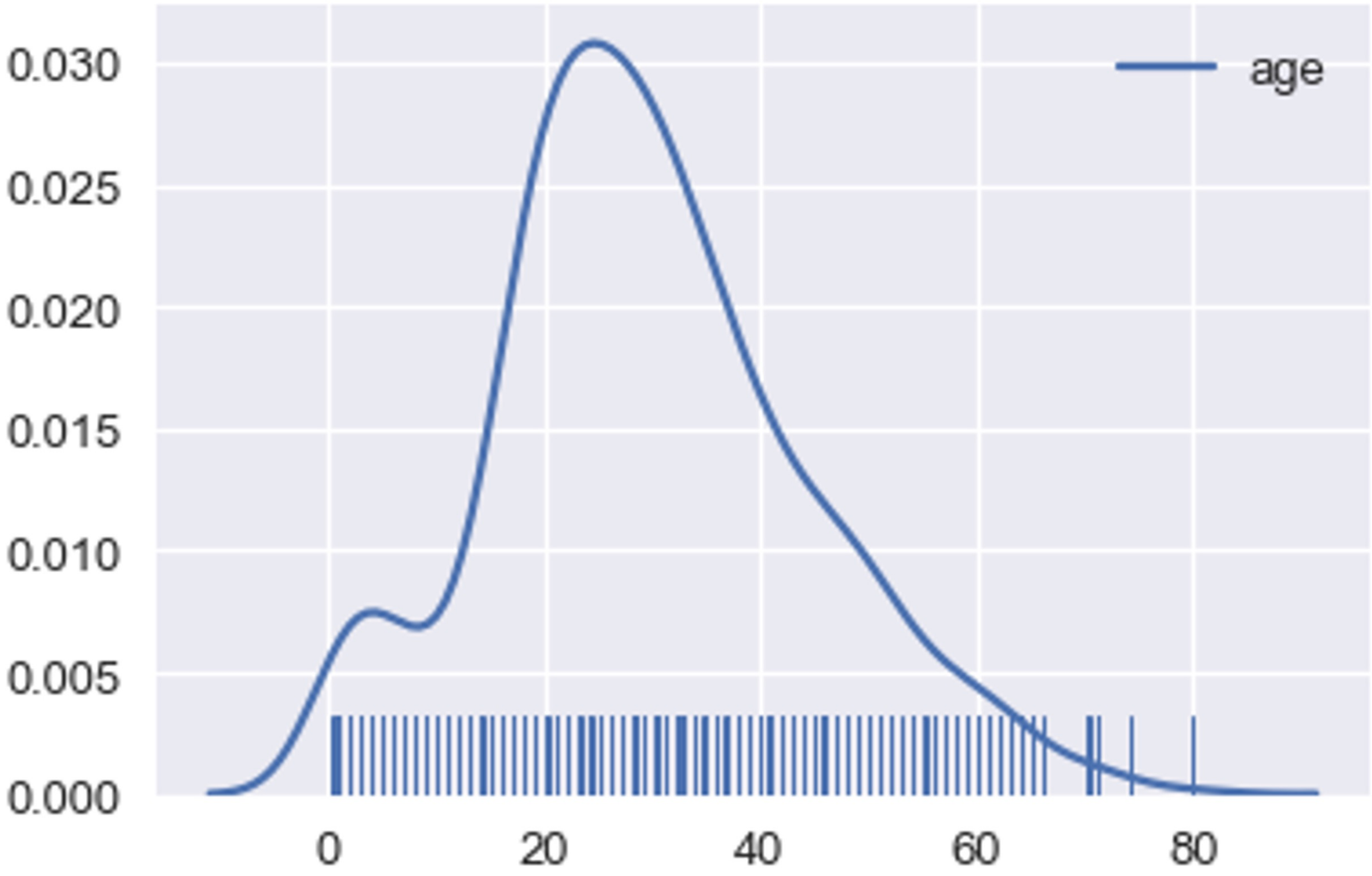
1. Place a “kernel” at each data point
2. Normalize kernels so that total area = 1
3. Sum all kernels together





Kernel Density Estimation

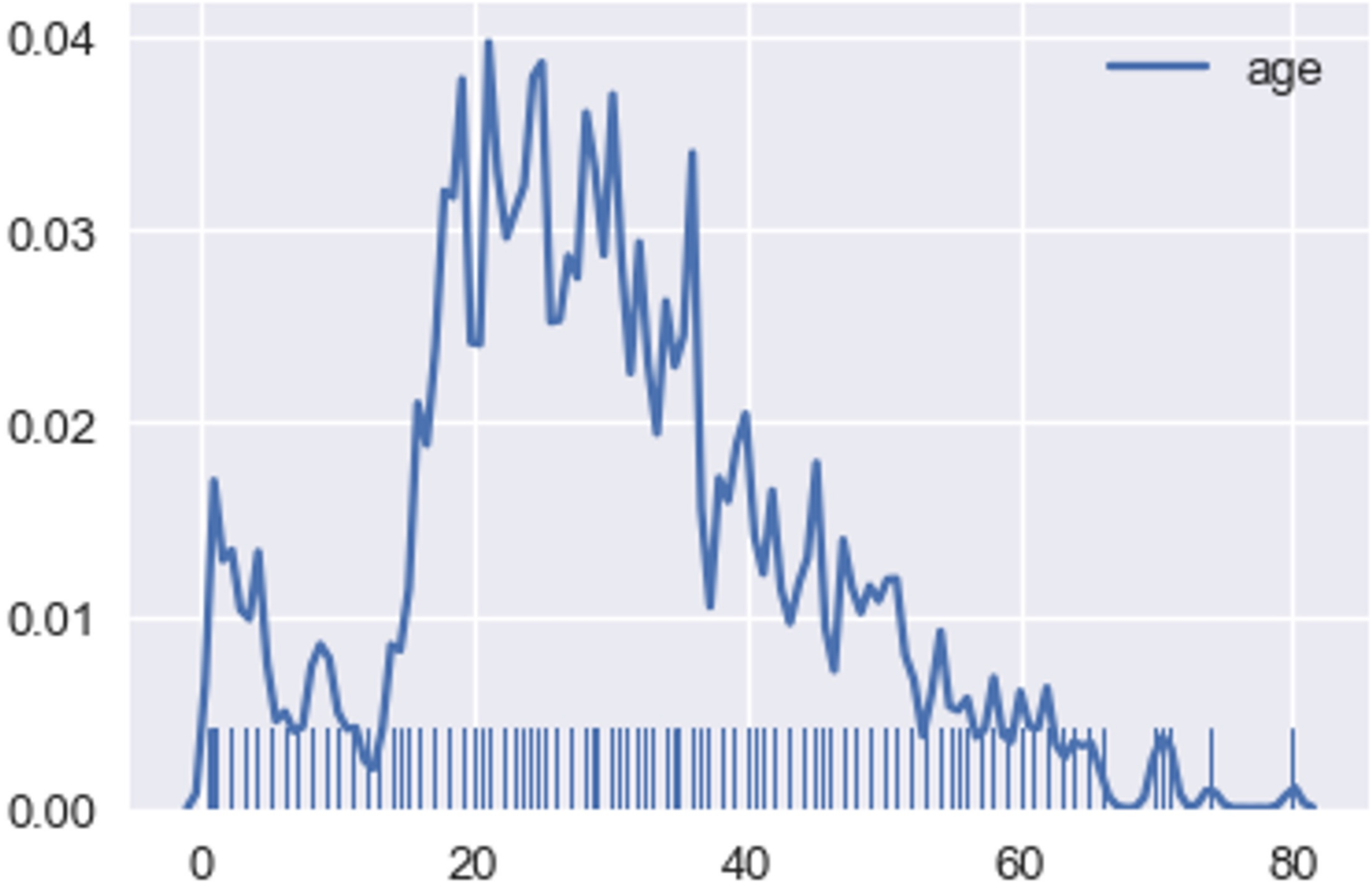
Gaussian kernel most common (default for seaborn).



Kernel Density Estimation

Changing width of each kernel = changing bandwidth

Narrow bandwidth is analogous to narrow bins for histogram

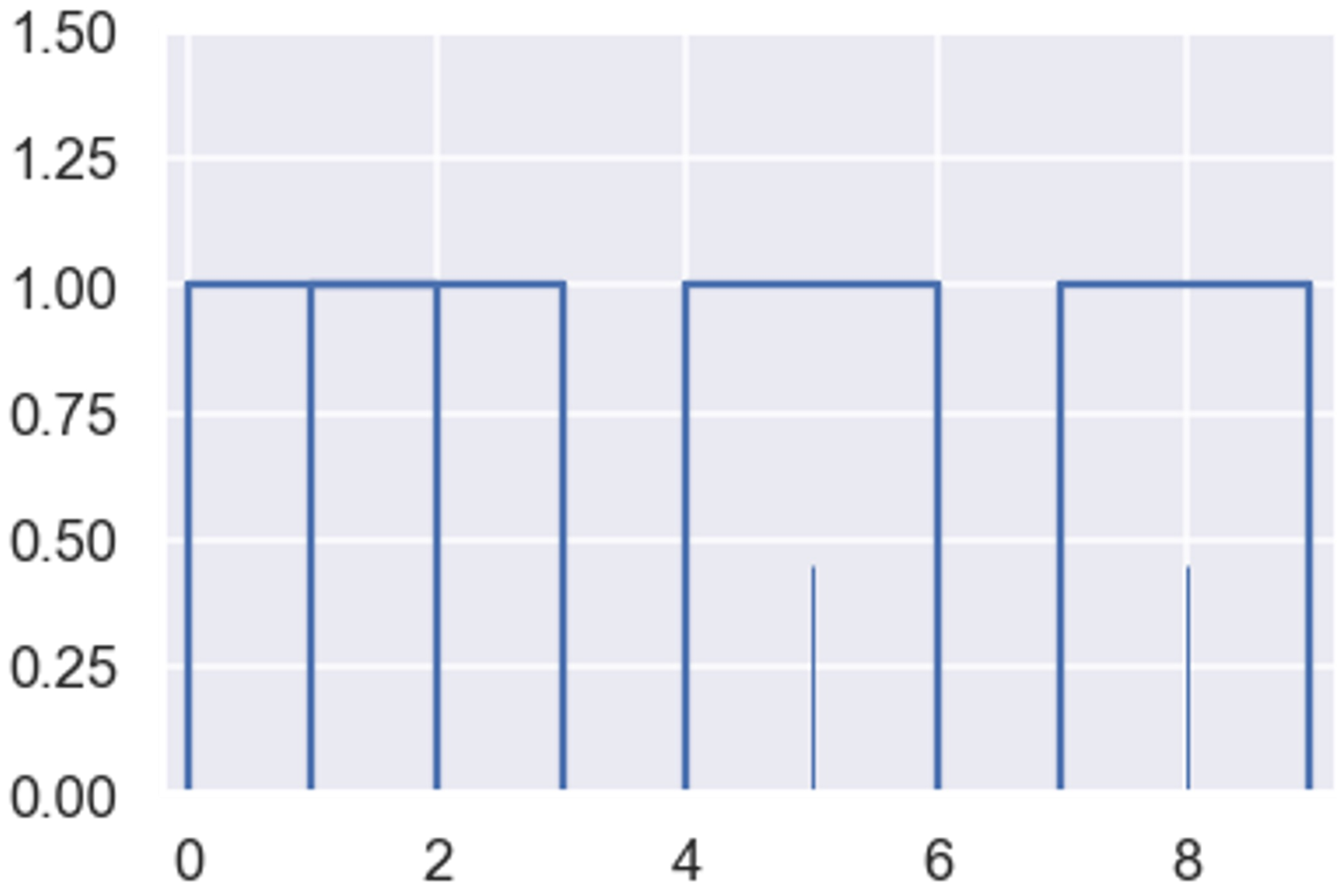


KDE Example – Uniform Kernel

Uniform kernel with bandwidth of 2.

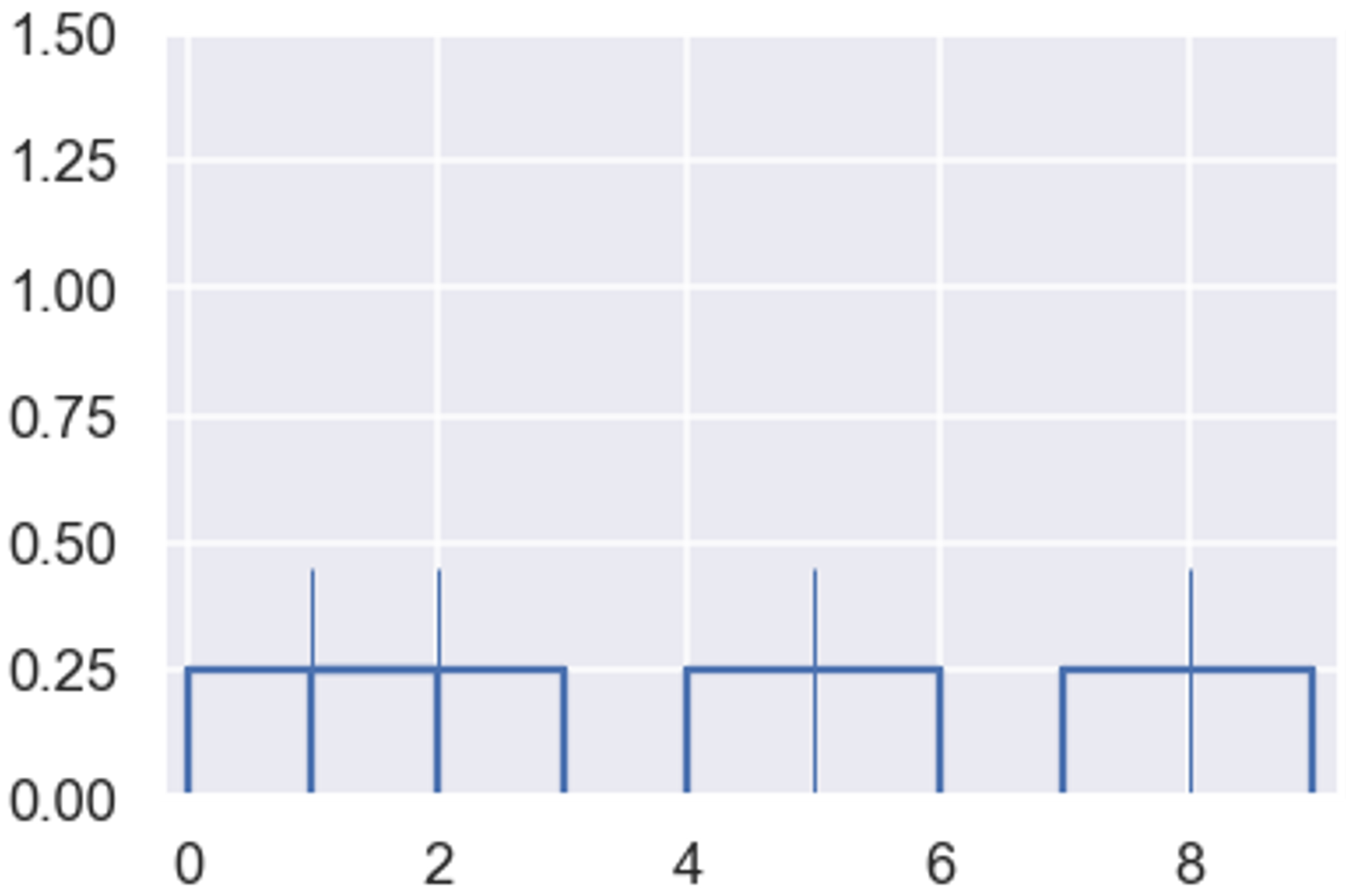
Data points at:

Kernel at each x: $x = [1, 2, 5, 8]$



KDE Example – Uniform Kernel

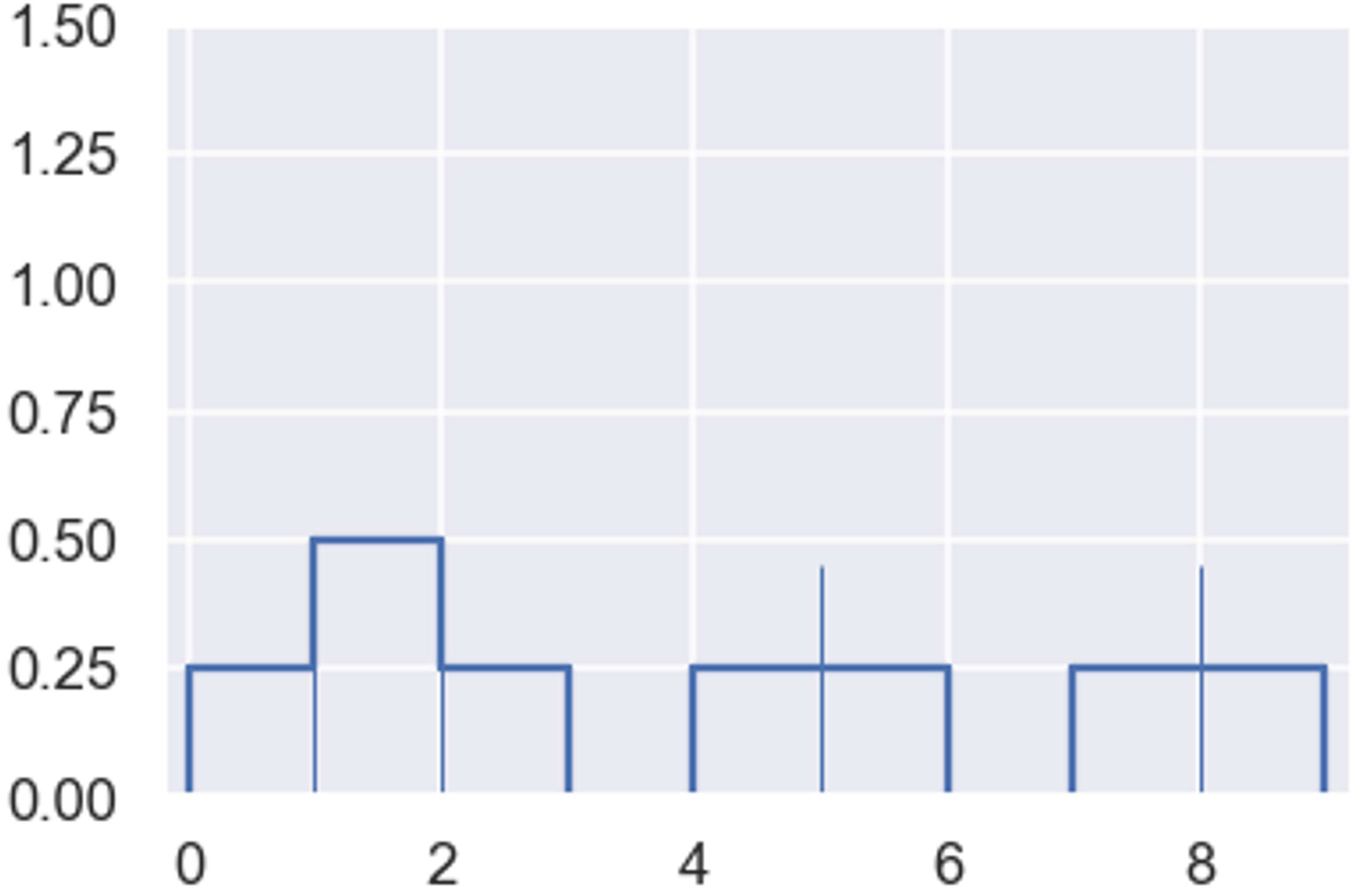
Scale each kernel by 1/4 since there are four points:



KDE Example – Uniform Kernel

Add kernels together:

Height at 1.5? 0.5



Summary

- When choosing a visualization, consider the principles of Scale, Conditioning, Perception, Transformation, Context, and Smoothing!
- In general: show the data!
  - Maximize data-ink ratio: cut out everything that isn’t data-related

VIS EXAMPLES

FOR

MULTIVARIATE NETWORKS

CMPT 733

INSTRUCTOR: **STEVEN BERGNER**

SURVEY PAPER AND TUTORIAL SLIDES BY:
CAROLINA NOBRE, MARC STREIT, ALEXANDER LEX

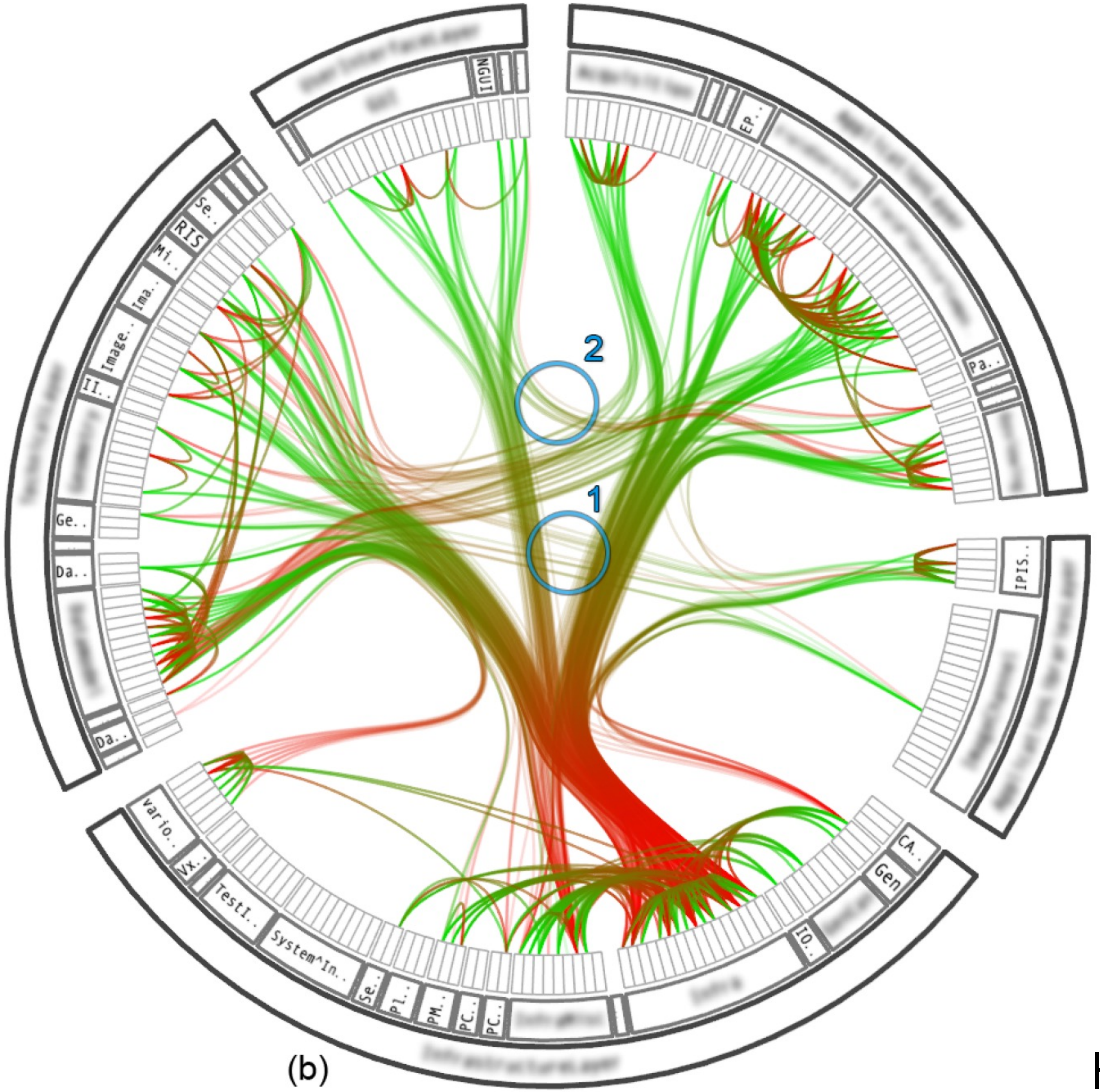
Photo by Rob Curran


Name: Samuel
Age: 41
Job: Nurse
Name: Julia
Age: 34
Job: Vet
Name: Ellen
Age: 31
Job: Actress
Name: Gordon
Age: 54
Job: Chef
Name: Roger
Age: 51
Job: Doctor
Name: Camille
Age: 42
Job: Teacher
Photo by Rob Curran

A MULTIVARIATE NETWORK IS
NETWORK TOPOLOGY +
NODE AND EDGE ATTRIBUTES


LinkedIn
LiVEJOURNAL
foursquare
vimeo
YouTube
Insta
facebook
Blogger
my
twitter
flickr
Behance
instagram
tumblr.
Like

Glyoxylate cycle
S–CoA
LASE
H$_2$O
CoA-SH
OXALATE DECARBOXYLASE
CO$_2$
OXALYL-CoA DECARBOXYLASE
OXALATE-CoA-TRANS
FERREDOXIN (red)
FERREDOXIN (ox.)
2-OXO-GLUTARATE SYNTHASE
Deficient in hyperoxaluria I
PYRUVATE
ThPP
H–C–OH
CH$_2$
CH$_2$
COOH
ALANINE-GLYOXYLATE TRANSAMINASE
OXALYL-CoA
OXALATE
O=C–S–CoA
O=C–OH
CoA-SH
PyrP
L-ALANINE
TOSE,
EROL
NADPH +H$^+$
NADP$^+$
CoA-SH
GLYOXYLATE DEHYDROGENASE (ACYLATING)
FMNH$_2$ · E
(S)-2-HYDROXY ACID DEHYDROGENASE
O$_2$
FMN · E
H$_2$O$_2$
HOOC–C–OH
NAD$^+$
NADH + H$^+$
GLYCOLATE
GLYOXYLATE REDUCTASE
GLYOXYLATE
NH$_3$
H$_2$O$_2$
D-AMINO ACID OXIDASE
O$_2$
H$_2$O
FAD
ThPP
Mg^{++}
2-OXO-GLUTARATE
H$_2$C–OH
COOH
COOH
(S)-2-HYDROXY-ACID OXIDASE
GLYCINE TRANSAMINASE
PyrP
LUORO-
ITRATE
FMN · E
FMNH$_2$ · E
L-GLUT-
DEHY
NADH +



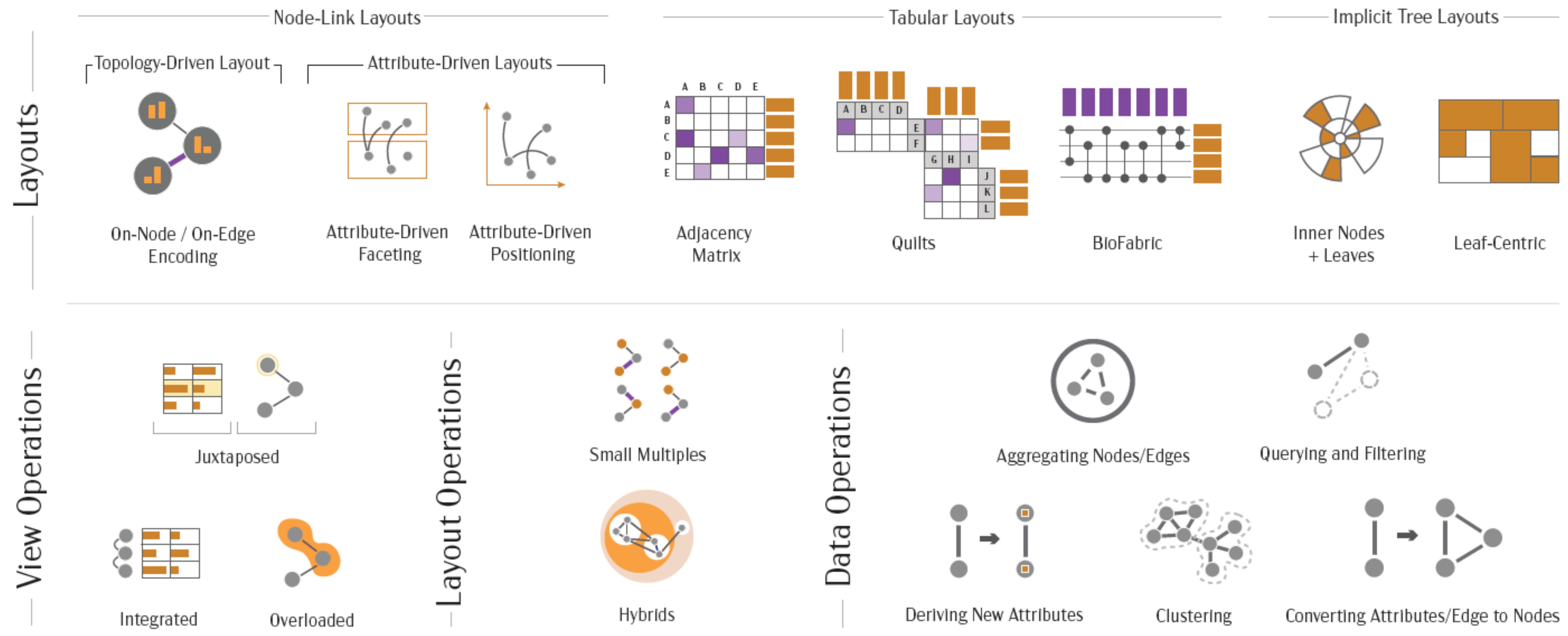
(b)

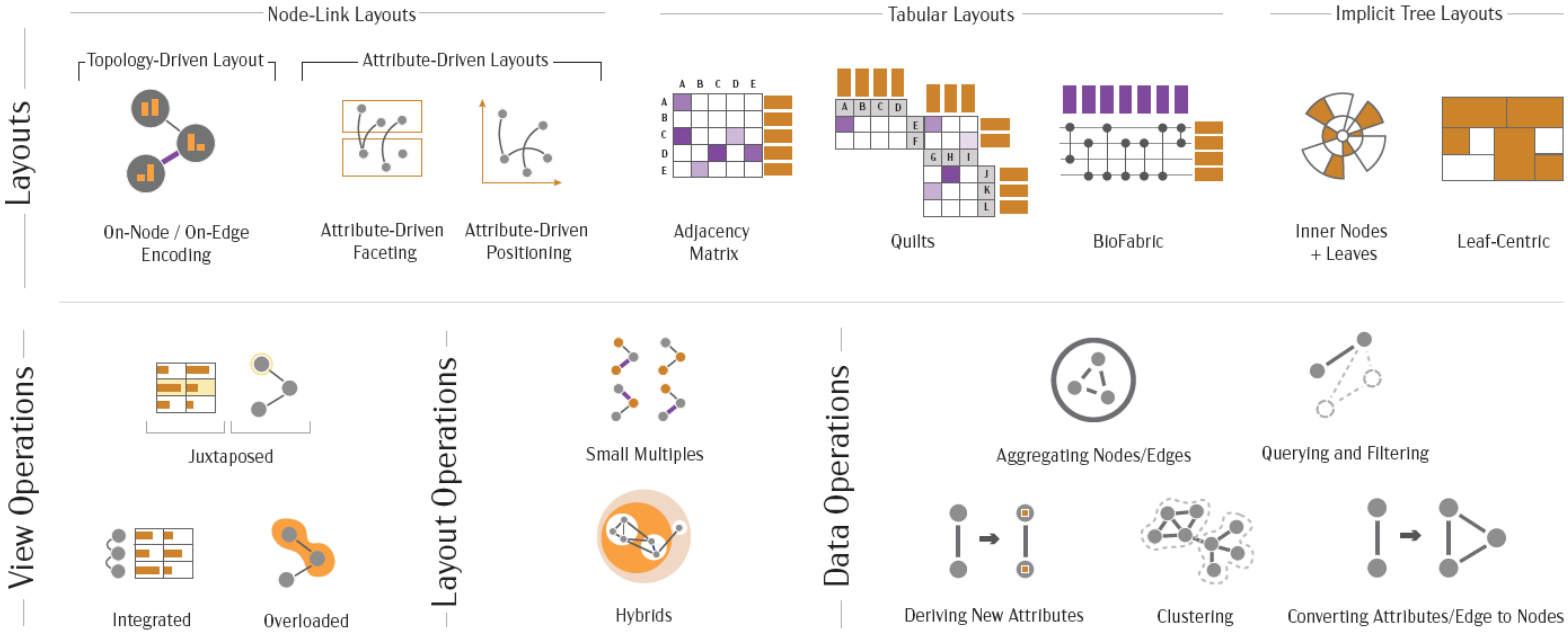
Holten and Wijk, 2009

The State of the Art in Visualizing Multivariate Networks

C. Nobre[1], M. Meyer[1], M. Streit[2], and A. Lex[1]

[1]University of Utah, Utah, USA
[2]Johannes Kepler University Linz, Austria

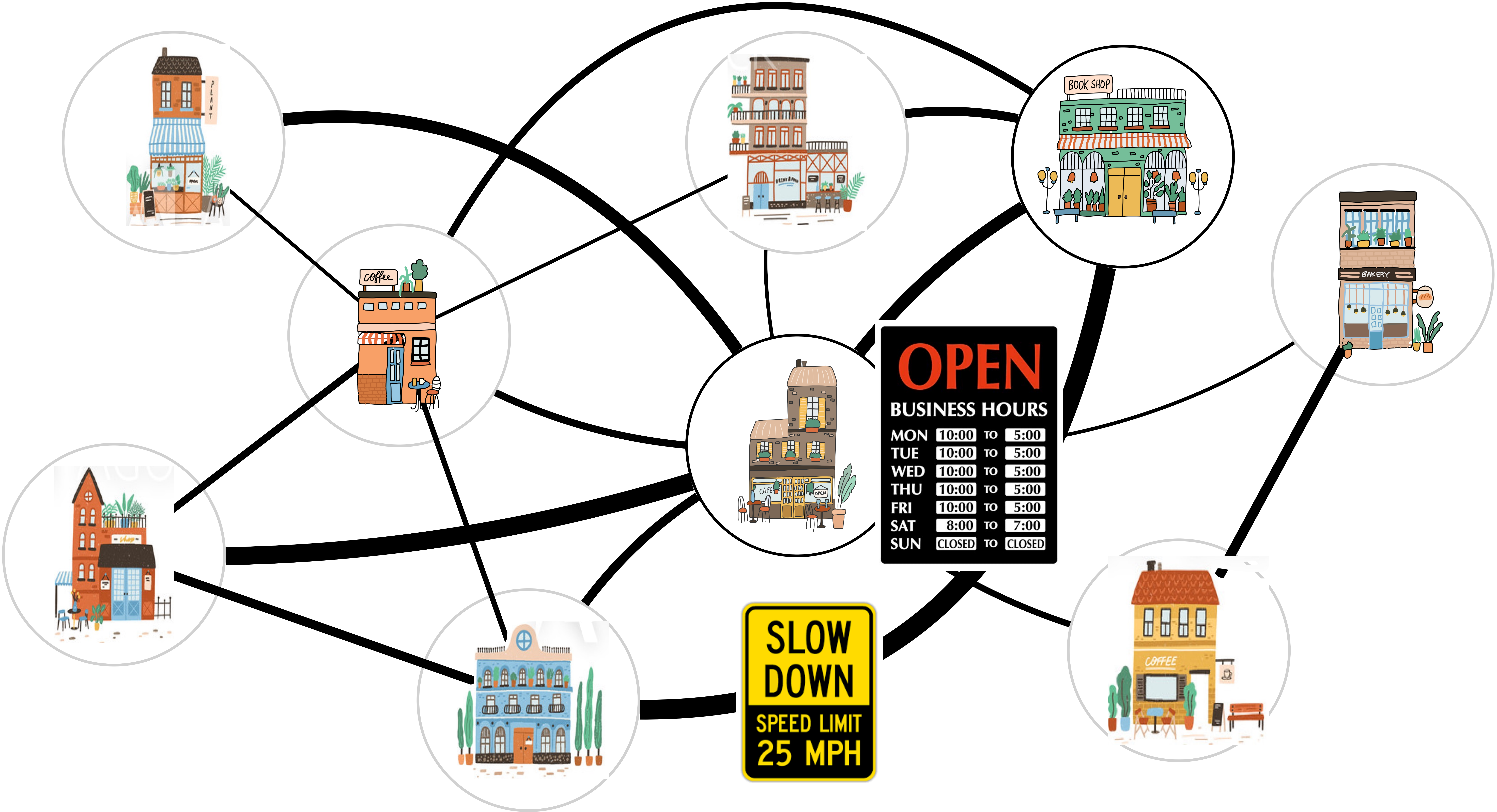



Layouts
Node-Link Layouts
Topology-Driven Layout
Attribute-Driven Layouts
On-Node / On-Edge Encoding
Attribute-Driven Faceting
Attribute-Driven Positioning
Tabular Layouts
Adjacency Matrix
Quilts
BioFabric
Implicit Tree Layouts
Inner Nodes + Leaves
Leaf-Centric
View Operations
Juxtaposed
Integrated
Overloaded
Layout Operations
Small Multiples
Hybrids
Data Operations
Aggregating Nodes/Edges
Querying and Filtering
Deriving New Attributes
Clustering
Converting Attributes/Edge to Nodes

MVNV Tasks

How is an MVN task different than a regular graph task?

MVN Tasks rely on both the topology of the network and the attributes of the nodes and edges



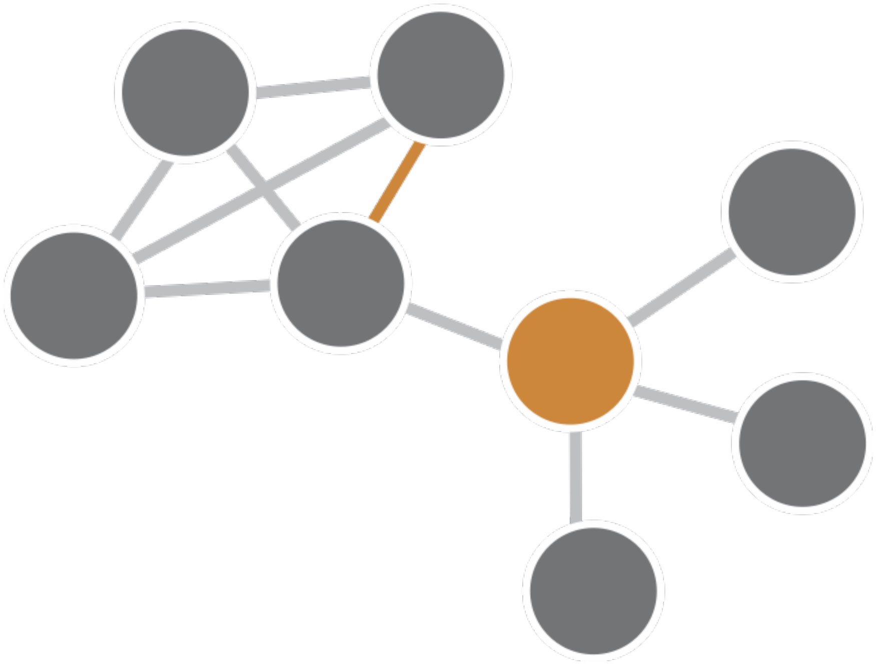
What is an efficient way I can complete all my errands?

- What is the fastest route to get all my errands done?

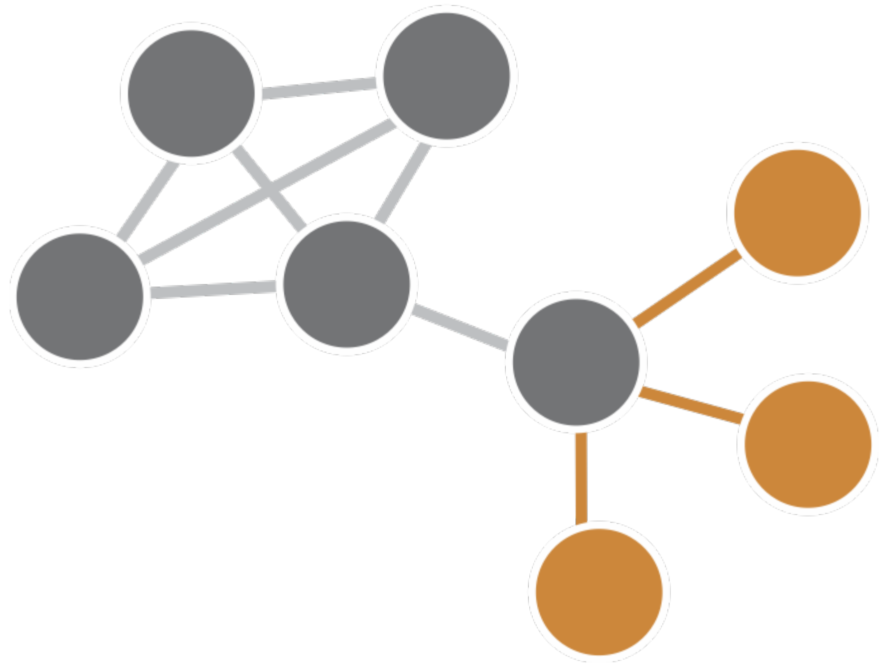
Tasks that rely on the **topology** of the network and the **attributes** of the nodes and edges

MVNV tasks are applied to topological structures

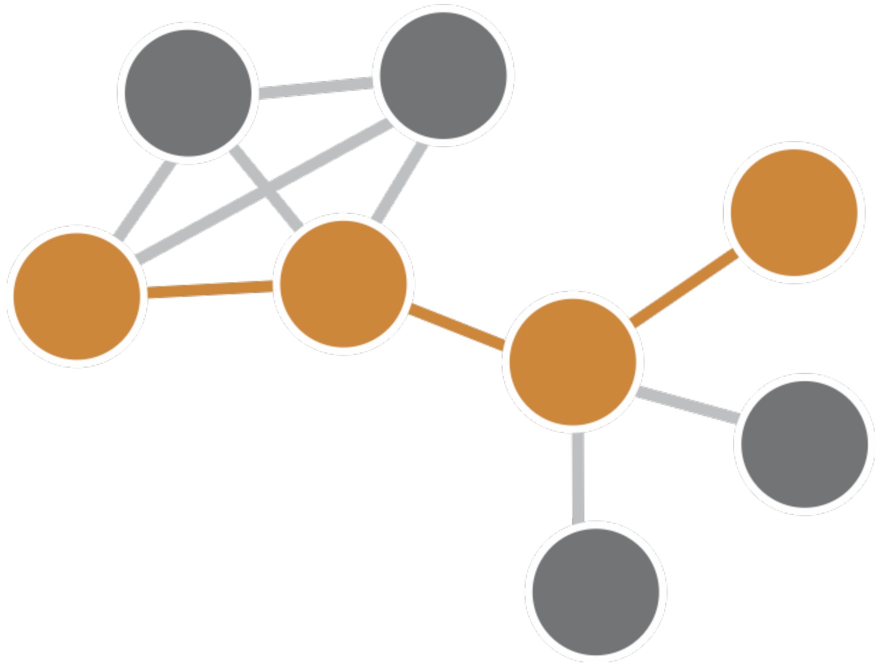
Single Node/Edge



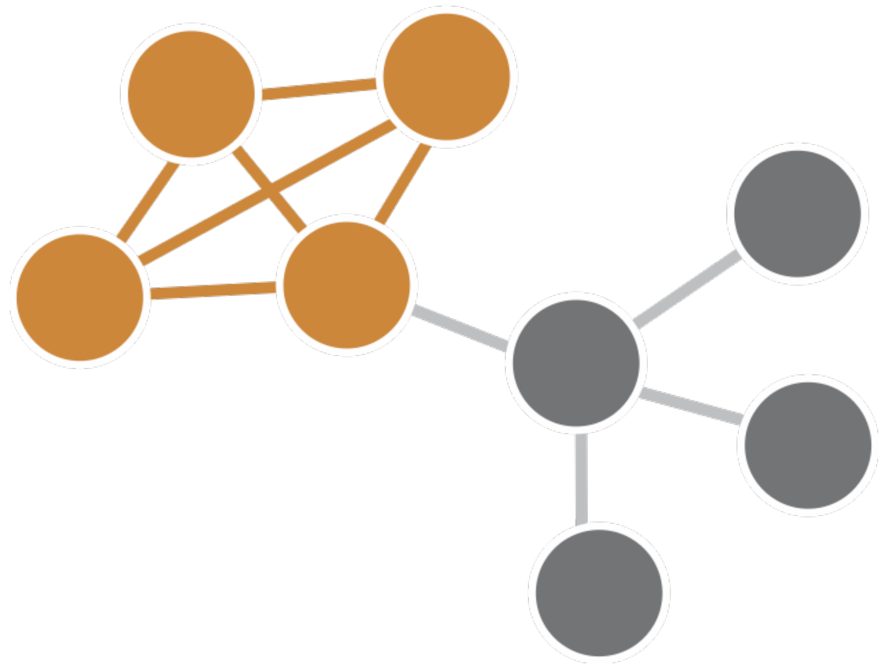
Node Neighbors



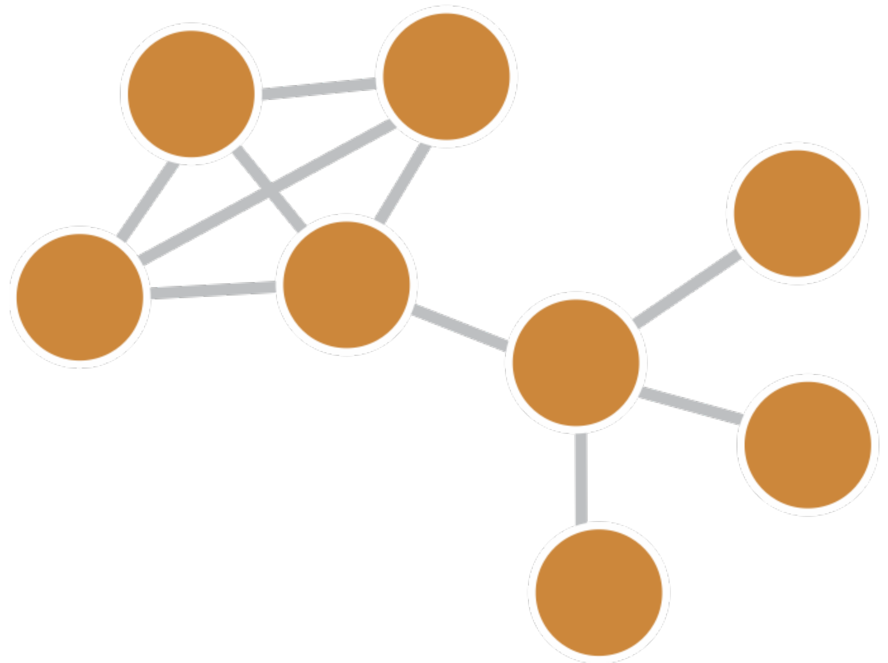
Path



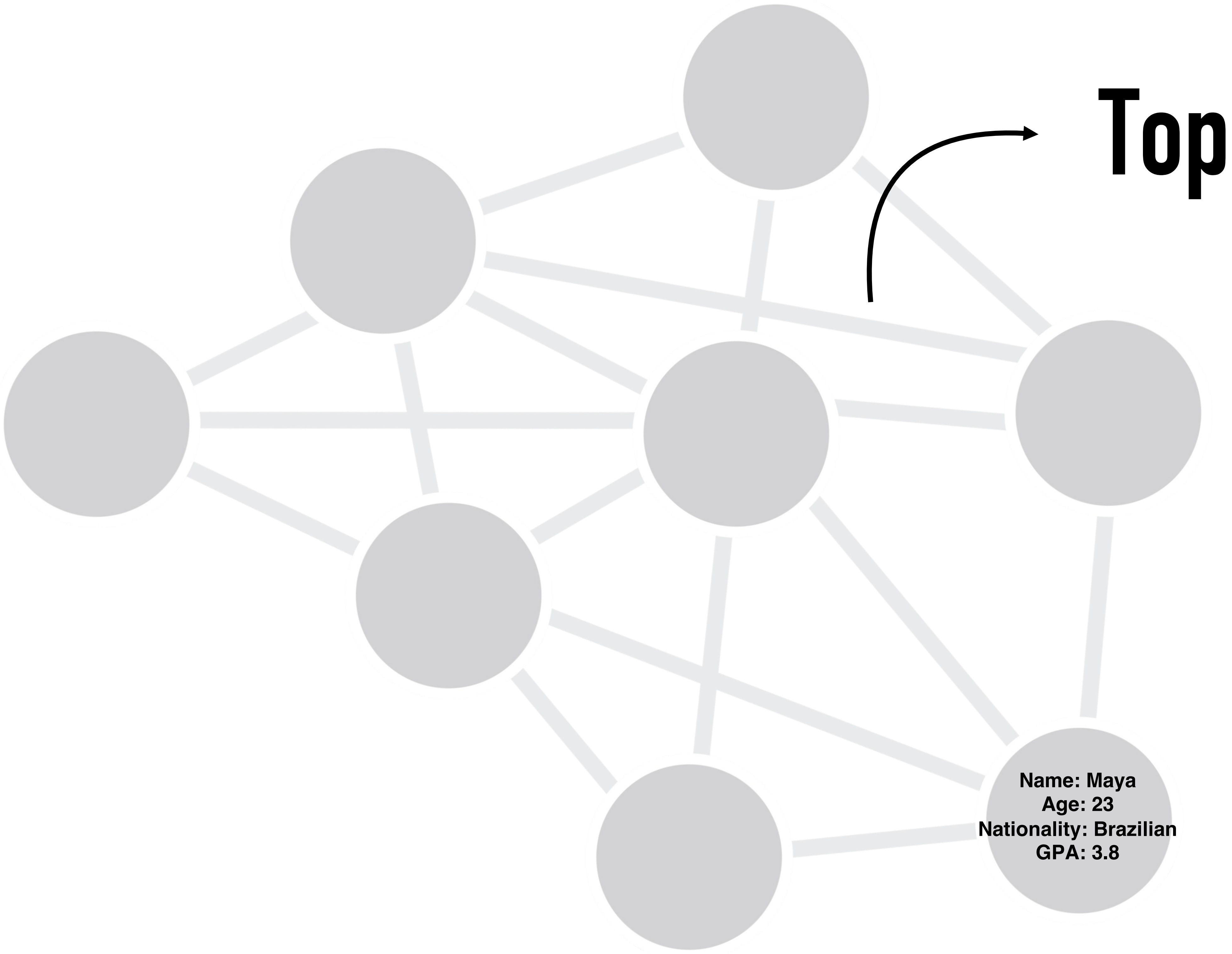
Cluster



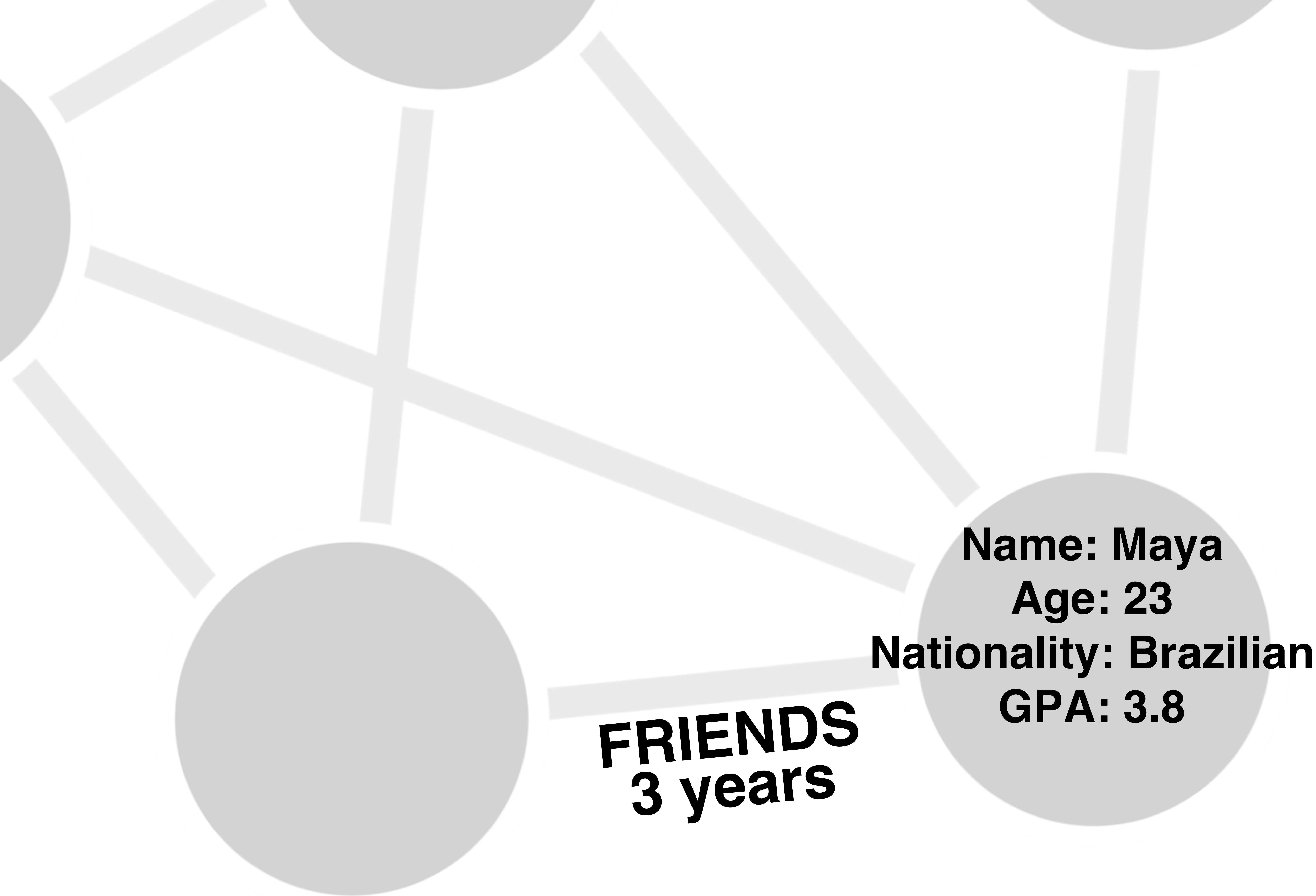
Network/Subnetwork

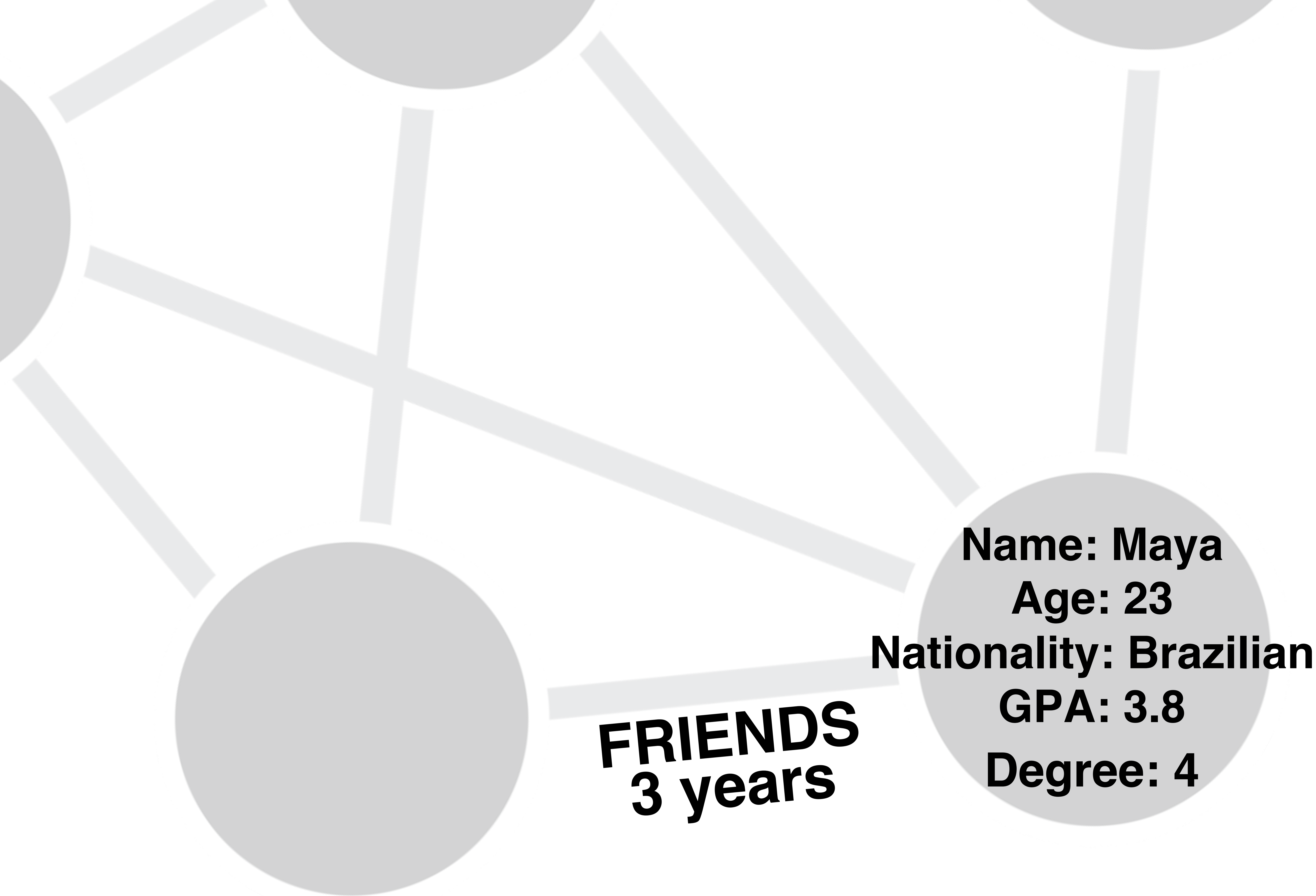


Network and Attribute Characteristics

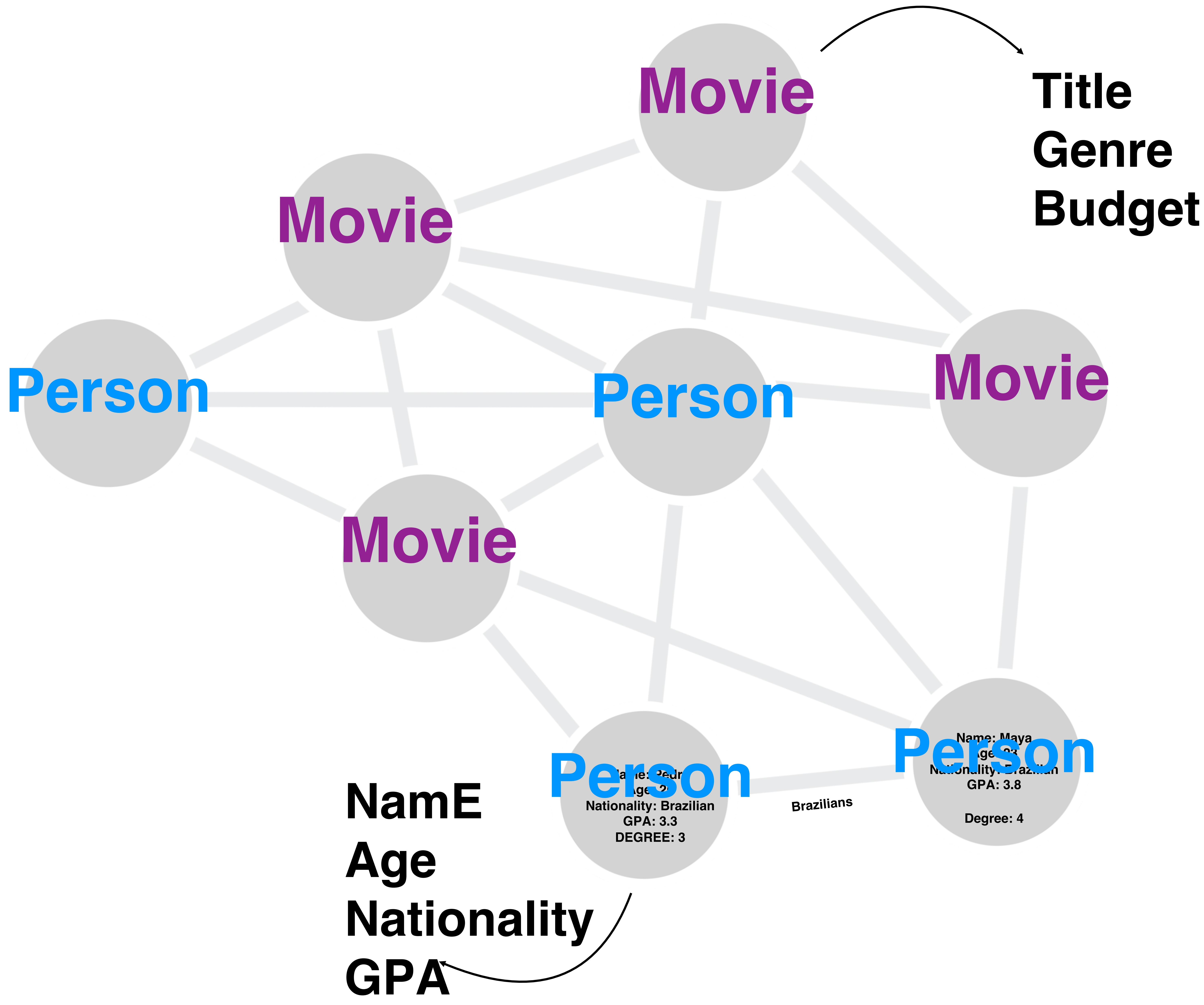


Topology

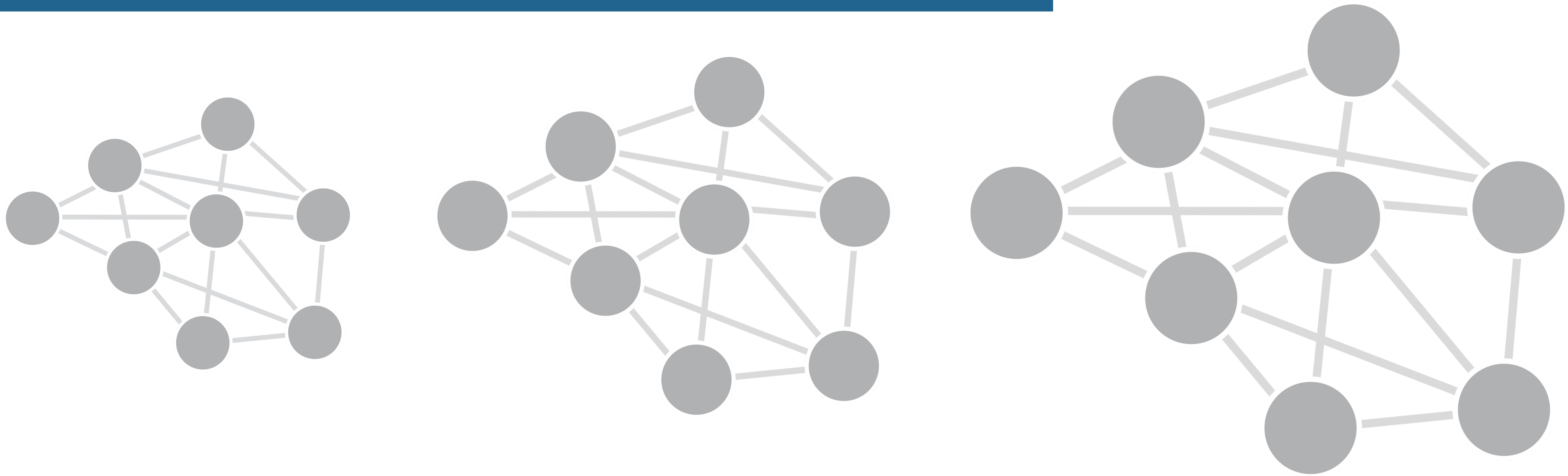

Name: Maya
Age: 23
Nationality: Brazilian
GPA: 3.8
FRIENDS
3 years


Name: Maya
Age: 23
Nationality: Brazilian
GPA: 3.8
Degree: 4
FRIENDS
3 years


Person
Name: Pedro
Age: 25
Nationality: Brazilian
GPA: 3.3
DEGREE: 3
Brazilians
Person
Name: Maya
Age: 23
Nationality: Brazilian
GPA: 3.8
Degree: 4
ity


Movie
Title
Genre
Budget
Movie
Person
Person
Movie
Movie
Person
Nationality: Brazilian
GPA: 3.3
DEGREE: 3
Person
Name: Maya
GPA: 3.8
Degree: 4
Brazilians
NamE
Age
Nationality
GPA

Network Size

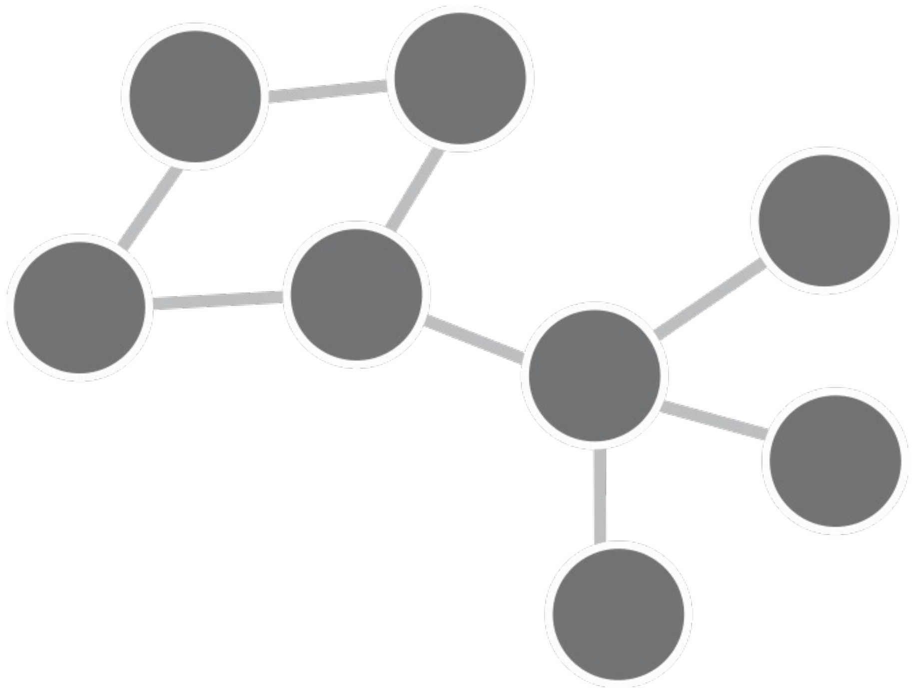


Small
<100

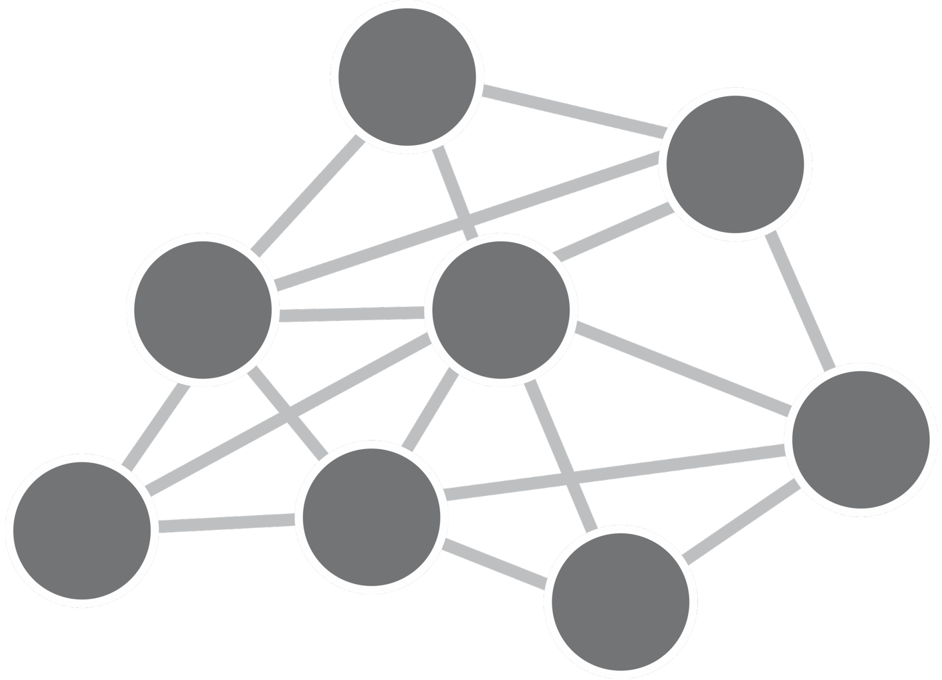
Medium
100–1000

Large
>1000

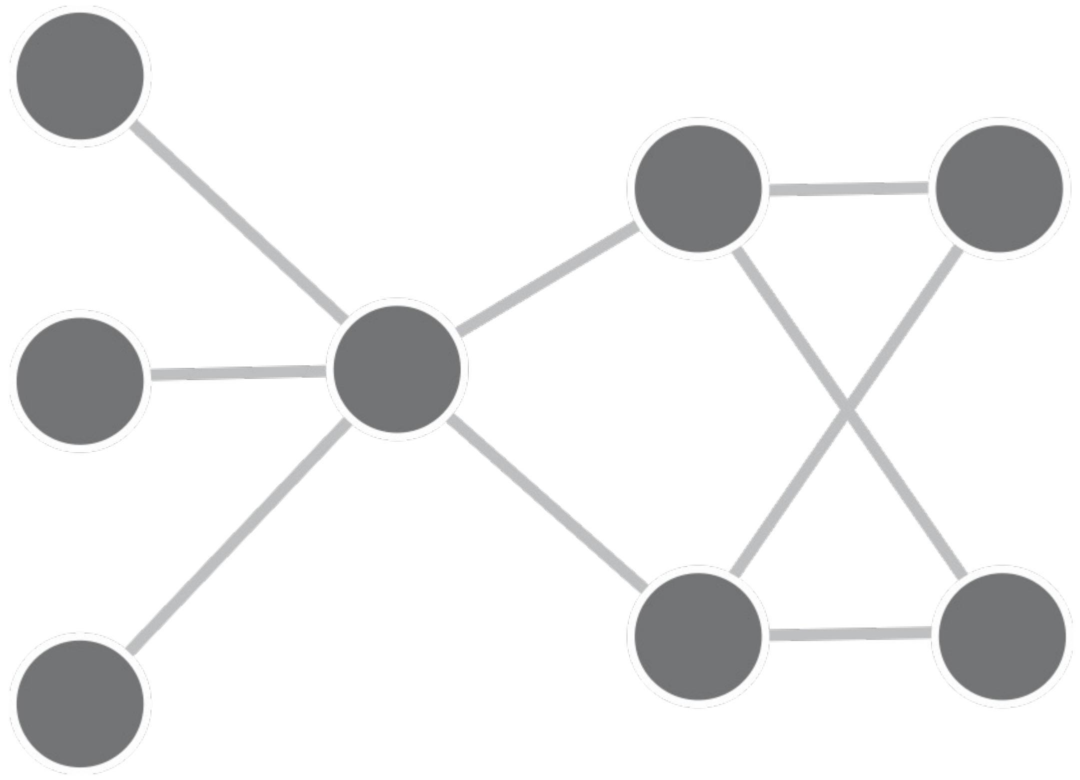
Network Types



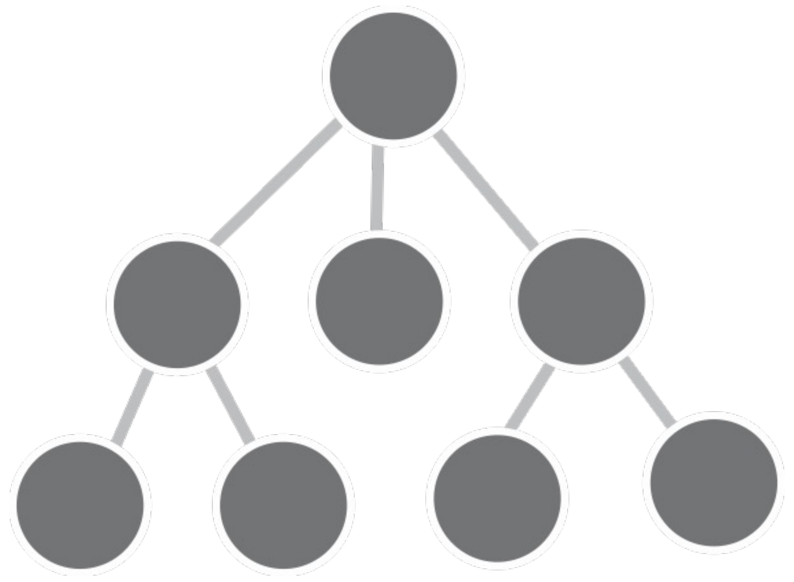
Sparse



Dense

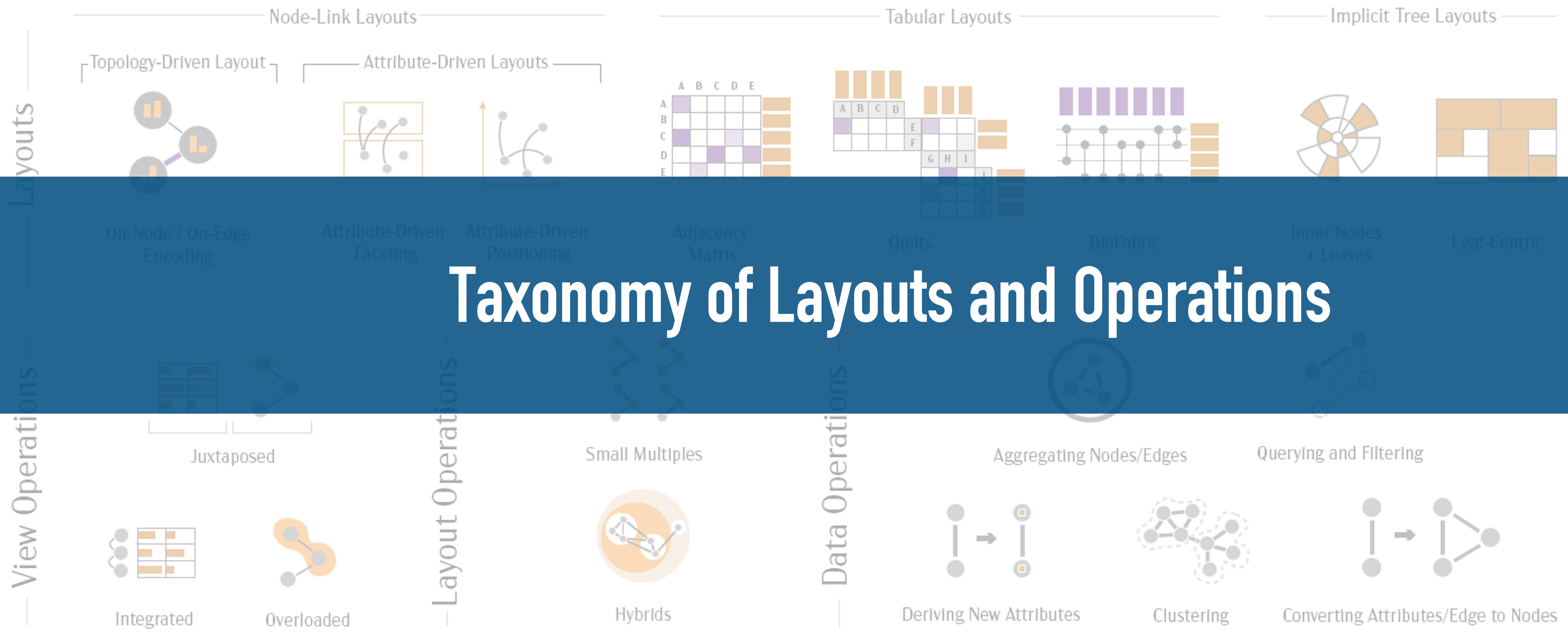


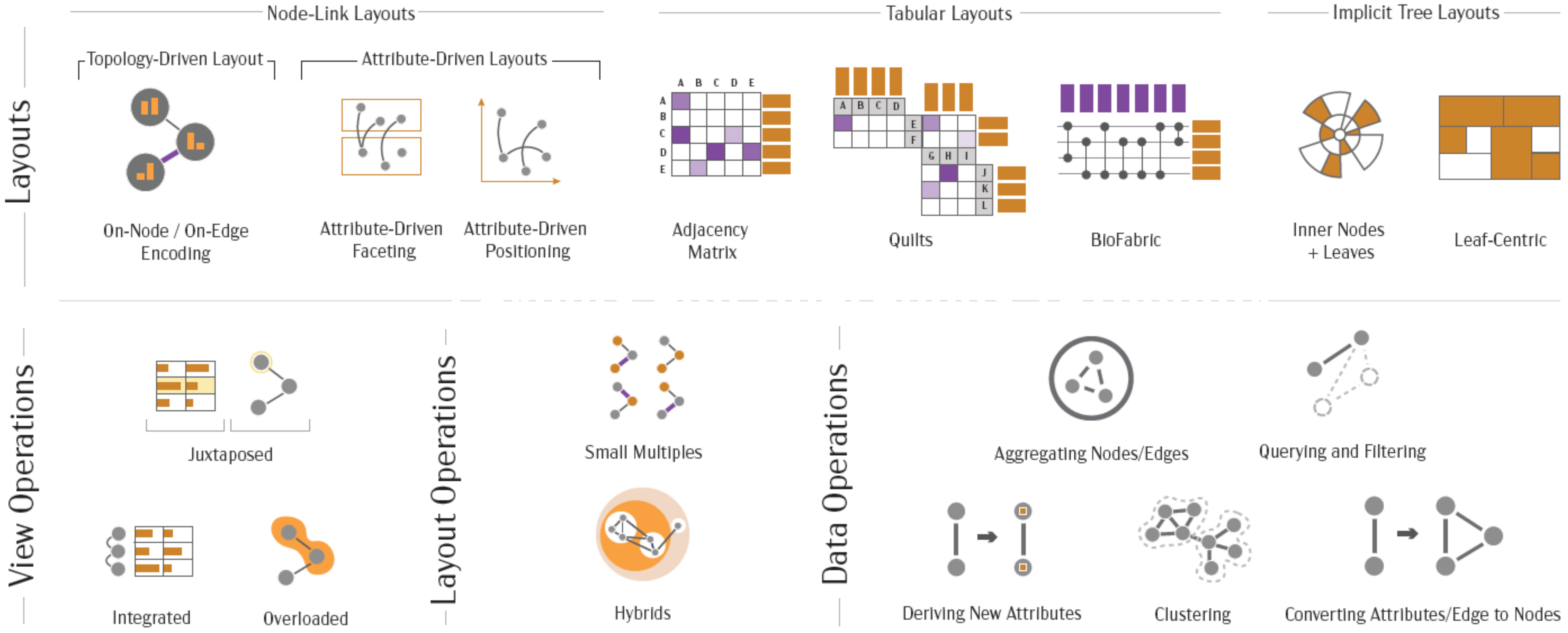
Layered

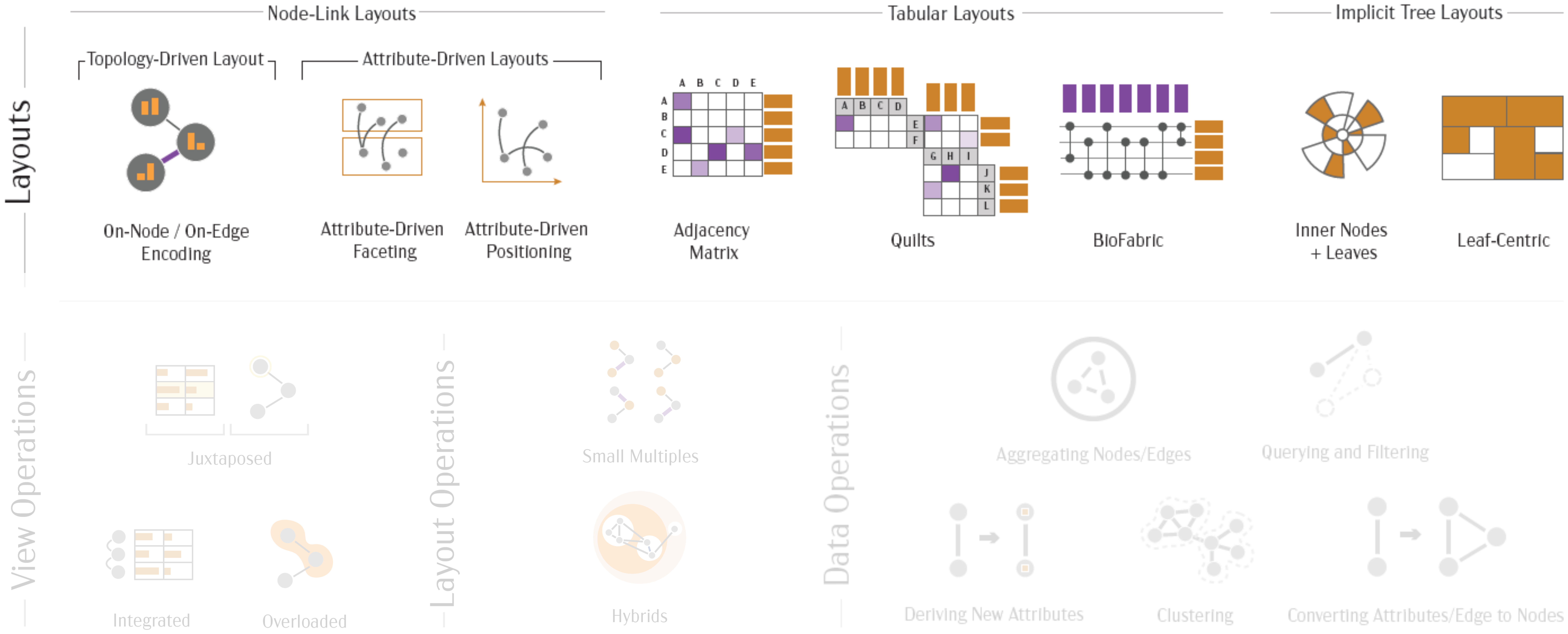


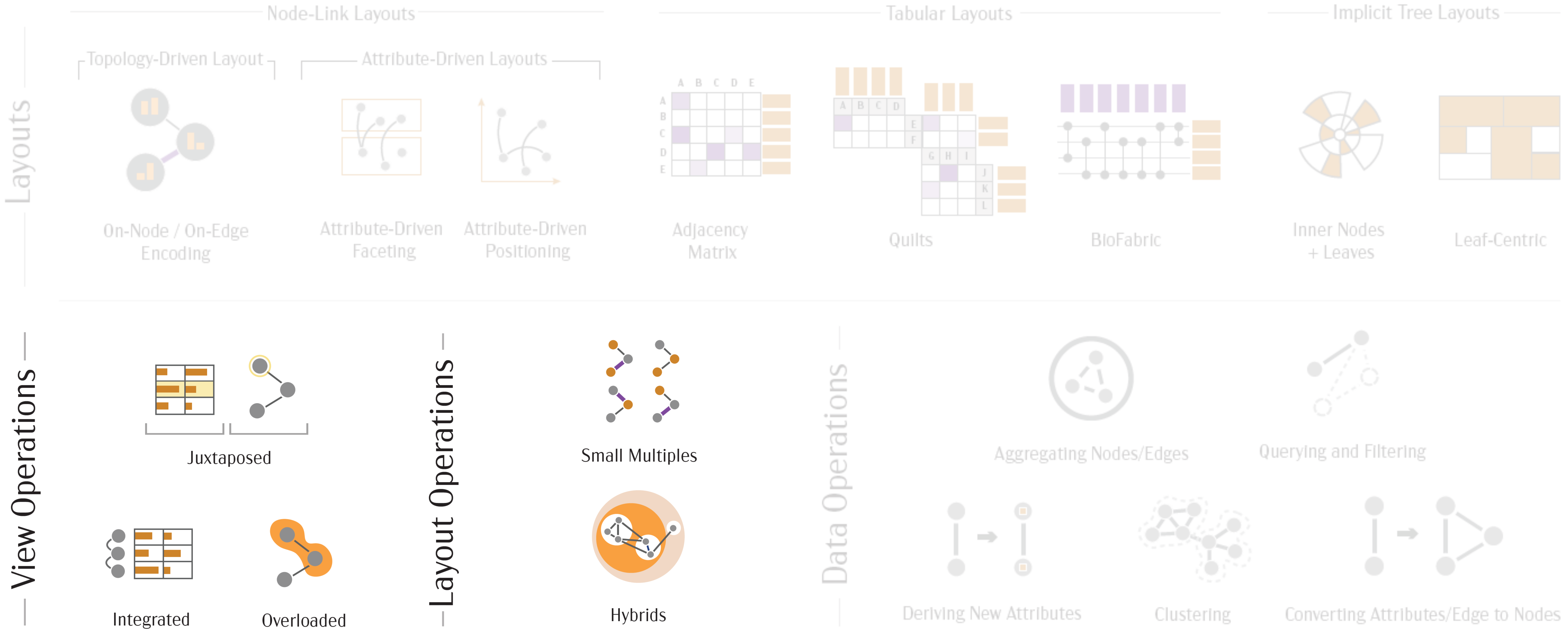
Trees

Taxonomy of Layouts and Operations

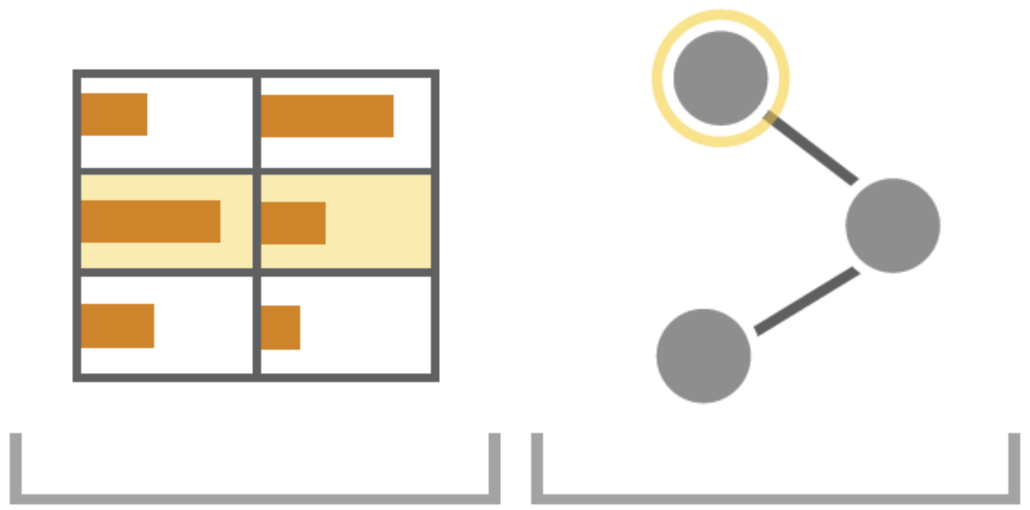



Layouts
Node-Link Layouts
Topology-Driven Layout
Attribute-Driven Layouts
On-Node / On-Edge Encoding
Attribute-Driven Faceting
Attribute-Driven Positioning
Tabular Layouts
Adjacency Matrix
Quilts
BioFabric
Implicit Tree Layouts
Inner Nodes + Leaves
Leaf-Centric
View Operations
Juxtaposed
Integrated
Overloaded
Layout Operations
Small Multiples
Hybrids
Data Operations
Aggregating Nodes/Edges
Querying and Filtering
Deriving New Attributes
Clustering
Converting Attributes/Edge to Nodes

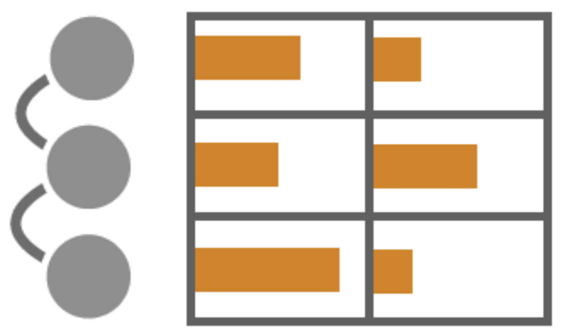

Layouts
Node-Link Layouts
Topology-Driven Layout
Attribute-Driven Layouts
On-Node / On-Edge Encoding
Attribute-Driven Faceting
Attribute-Driven Positioning
Tabular Layouts
Adjacency Matrix
Quilts
BioFabric
Implicit Tree Layouts
Inner Nodes + Leaves
Leaf-Centric
View Operations
Juxtaposed
Integrated
Overloaded
Layout Operations
Small Multiples
Hybrids
Data Operations
Aggregating Nodes/Edges
Querying and Filtering
Deriving New Attributes
Clustering
Converting Attributes/Edge to Nodes


Layouts
Node-Link Layouts
Topology-Driven Layout
Attribute-Driven Layouts
On-Node / On-Edge Encoding
Attribute-Driven Faceting
Attribute-Driven Positioning
Tabular Layouts
Adjacency Matrix
Quilts
BioFabric
Implicit Tree Layouts
Inner Nodes + Leaves
Leaf-Centric
View Operations
Juxtaposed
Integrated
Overloaded
Layout Operations
Small Multiples
Hybrids
Data Operations
Aggregating Nodes/Edges
Querying and Filtering
Deriving New Attributes
Clustering
Converting Attributes/Edge to Nodes

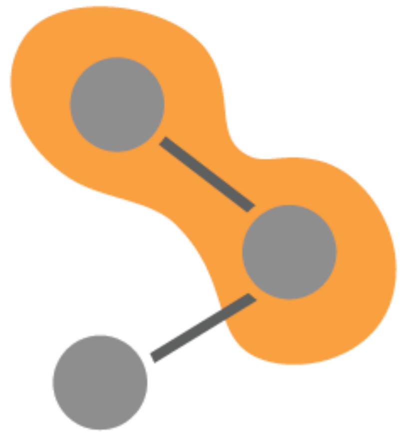
View Operations



Juxtaposed

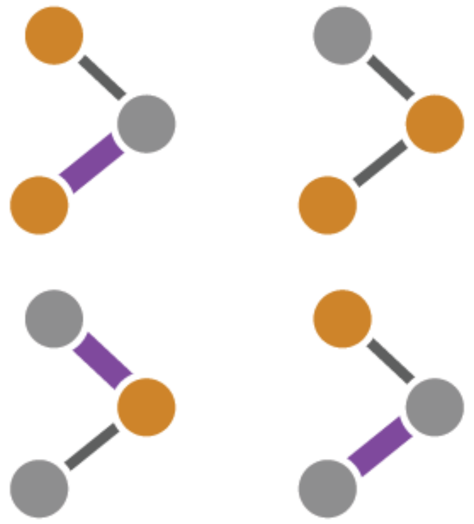


Integrated



Overloaded

Layout Operations

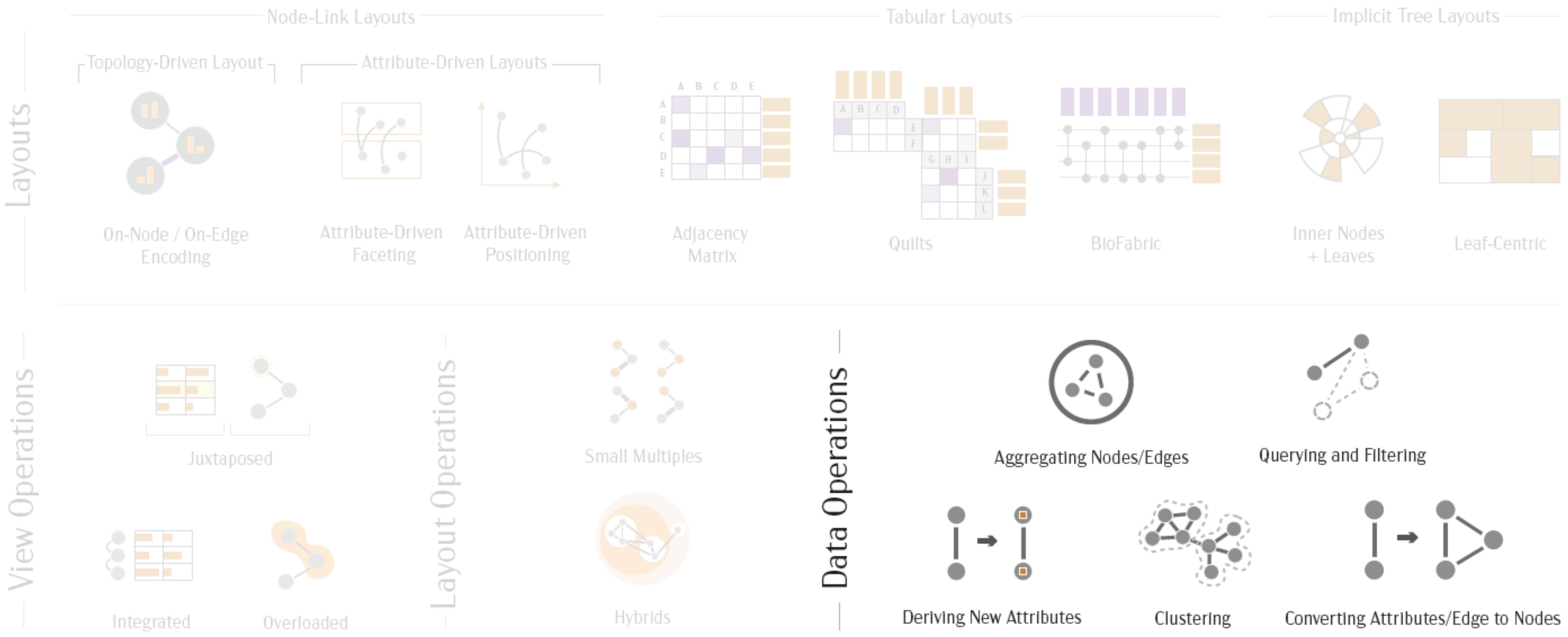


Small Multiples

Hybrids

Separate views for Topology and Attributes

Multiple layouts for Topology or Attributes


Layouts
Node-Link Layouts
Topology-Driven Layout
Attribute-Driven Layouts
On-Node / On-Edge Encoding
Attribute-Driven Faceting
Attribute-Driven Positioning
Tabular Layouts
Adjacency Matrix
Quilts
BioFabric
Implicit Tree Layouts
Inner Nodes + Leaves
Leaf-Centric
View Operations
Juxtaposed
Integrated
Overloaded
Layout Operations
Small Multiples
Hybrids
Data Operations
Aggregating Nodes/Edges
Querying and Filtering
Deriving New Attributes
Clustering
Converting Attributes/Edge to Nodes

		Size			Type				Node Attributes				Edge Attributes				Topolog. Structures				
		Small <100 nodes)	Medium (<1,000)	Large (>1,000 nodes)	Complex (sparse)	Complex (dense)	Layered/K-Partite	Trees	Few (<5)	Several (≥5)	Homog. (1 type)	Hetero. (>1 type)	Few (<3)	Several (≥3)	Homog. (1 type)	Hetero. (>1 type)	Single node/edge	Neighbors	Paths	Clusters	Entire/sub network
Node-Link Layouts	On-node/edge encoding	3	2	1	3	1	3	3	2	1	3	2	2	1	3	1	3	3	2	2	2
	Attr.-driven faceting	3	1	1	3	1	3	1	3	1	3	3	2	1	2	1	3	2	1	1	1
	Attr.-driven positioning	3	1	1	3	1	1	1	3	1	3	1	2	1	2	1	3	2	1	1	2
Tabular Layouts	Adjacency matrix	3	1	1	2	3	2	1	2	3	3	2	3	2	3	2	3	3	1	3	2
	Quilts	3	1	1	3	1	3	3	3	3	3	3	3	3	3	2	3	3	2	2	2
	BioFabric	3	1	1	3	3	2	1	3	3	3	3	3	3	3	3	3	1	1	1	2
Implicit	Inner nodes & leaves	3	2	1	0	0	0	3	3	1	3	1	0	0	0	0	3	3	3	0	3
	Leaves	3	2	2	0	0	0	3	3	1	3	1	0	0	0	0	3	2	1	0	3
View Operations	Juxtaposed	3	2	1	3	1	3	3	3	3	3	3	3	3	3	3	2	1	1	2	2
	Integrated	3	2	1	3	1	3	3	3	3	3	3	2	2	3	3	3	3	3	1	2
	Overloaded	3	2	1	3	1	3	3	3	1	3	1	1	1	1	1	3	3	2	3	2

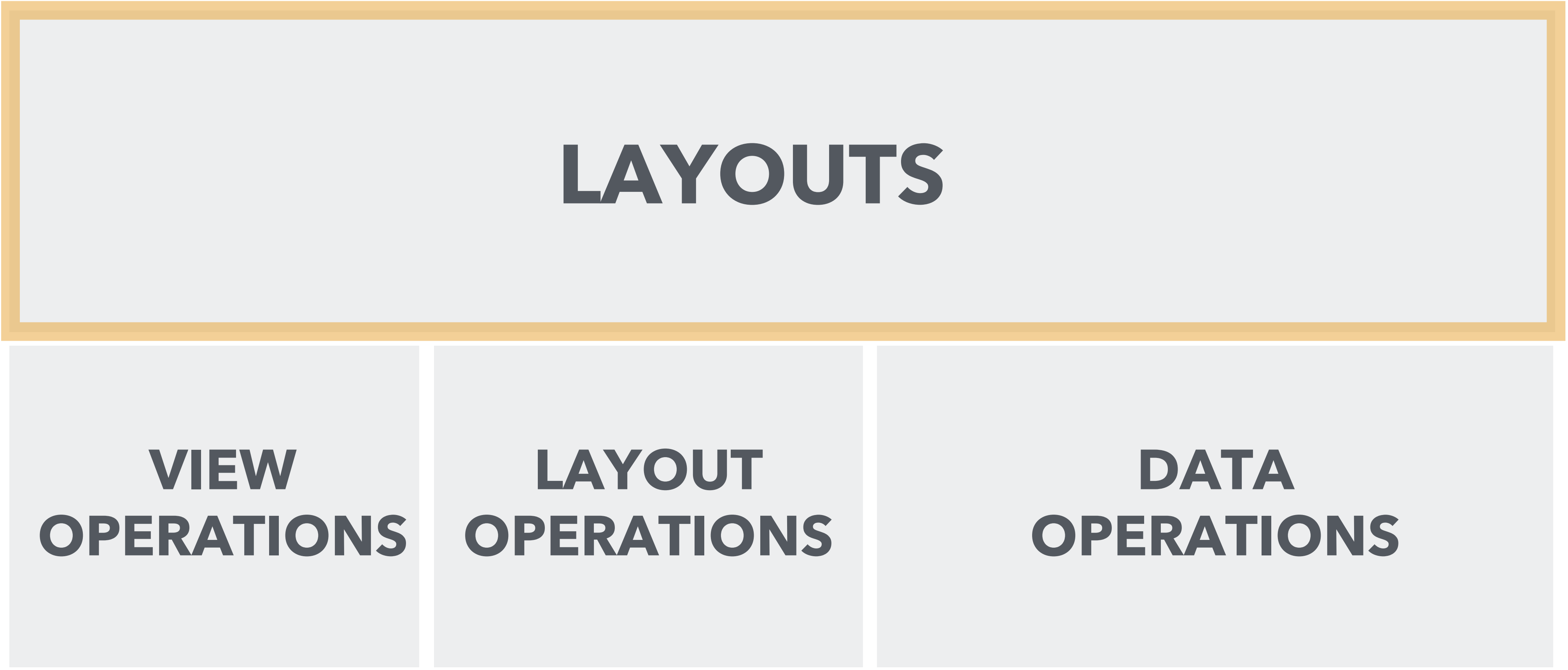
0 Does *not* support

1 Supports poorly

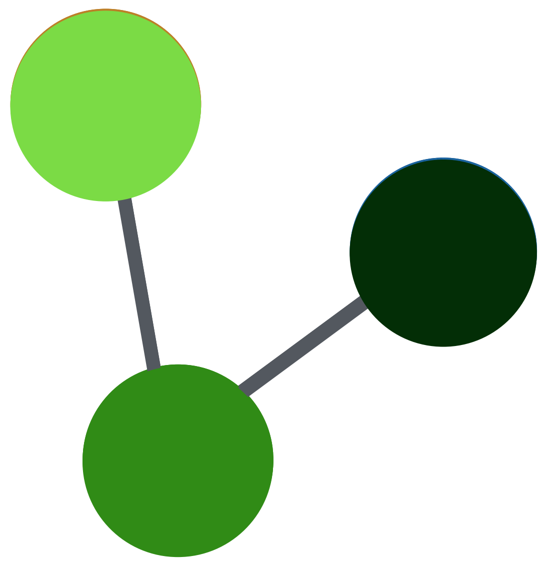
2 Supports

3 Optimized for


LAYOUTS
Layouts
Node-Link Layouts
Topology-Driven Layout
Attribute-Driven Layouts
On-Node / On-Edge Encoding
Attribute-Driven Faceting
Attribute-Driven Positioning
Tabular Layouts
Adjacency Matrix
Quilts
BioFabric
Implicit Tree Layouts
Inner Nodes + Leaves
Leaf-Centric
VIEW OPERATIONS
View Operations
Integrated
Overloaded
LAYOUT OPERATIONS
Layout Operations
Hybrids
DATA OPERATIONS
Data Operations
Deriving New Attributes
Clustering
Converting Attributes/Edge to Nodes



Node-Link Diagram with on-node encoding



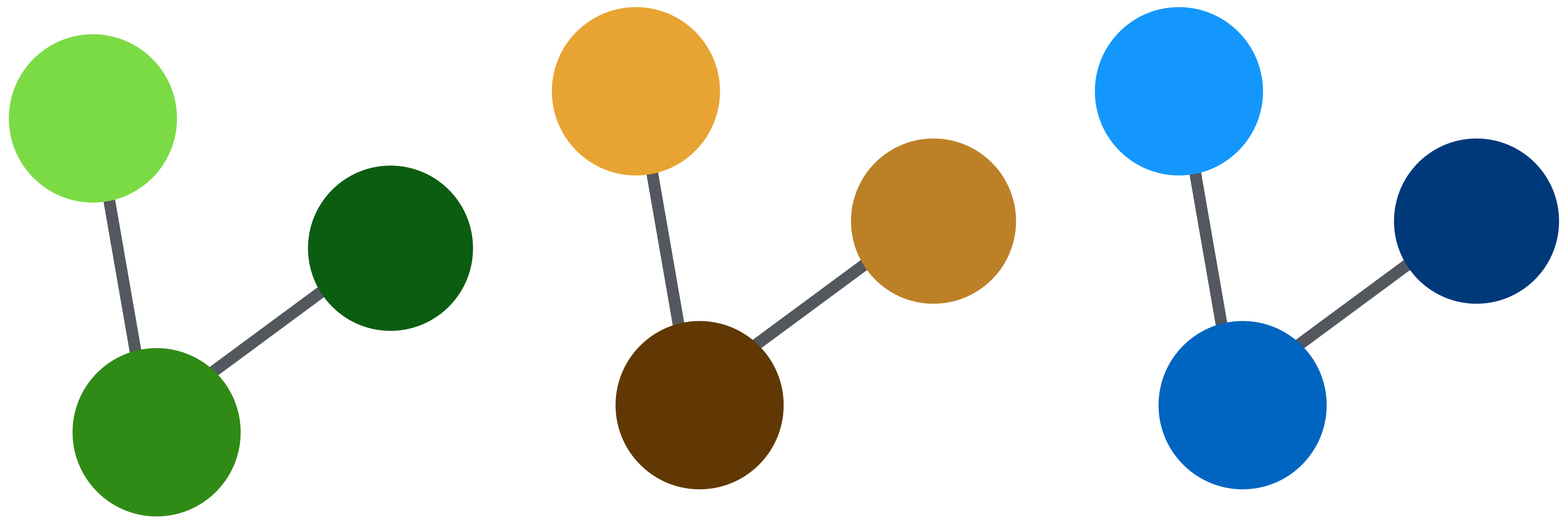
LAYOUTS

VIEW OPERATIONS

LAYOUT OPERATIONS

DATA OPERATIONS

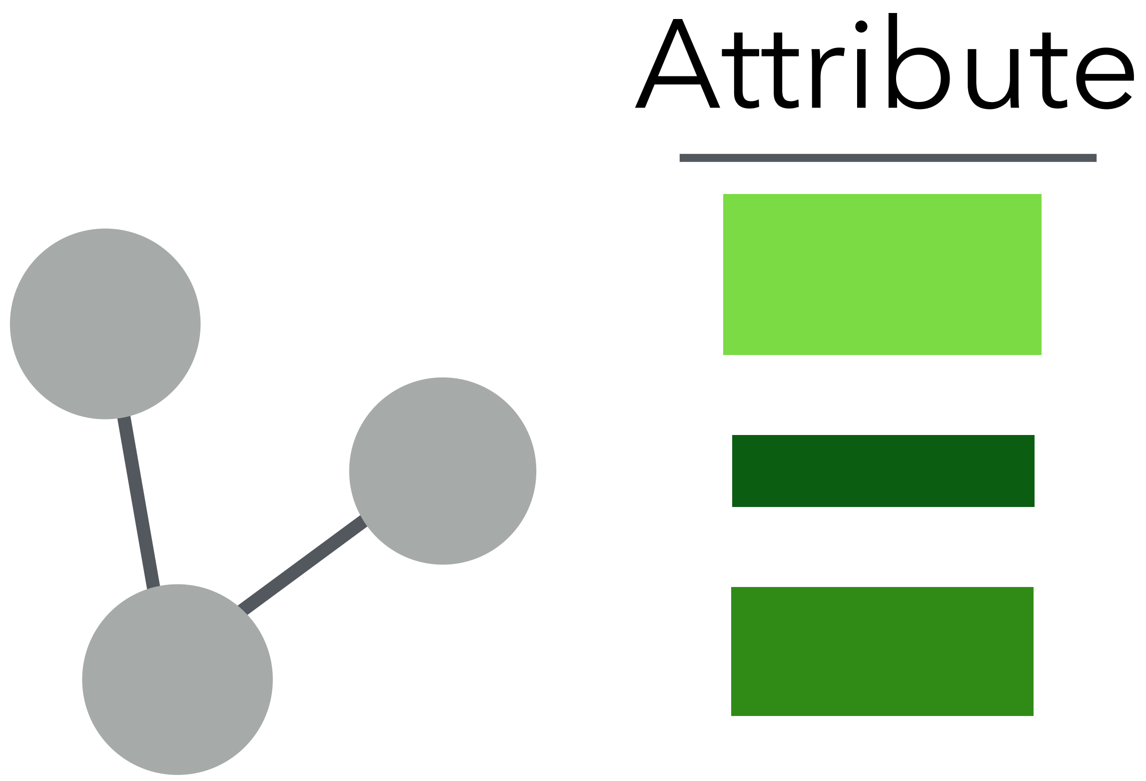
Small Multiples

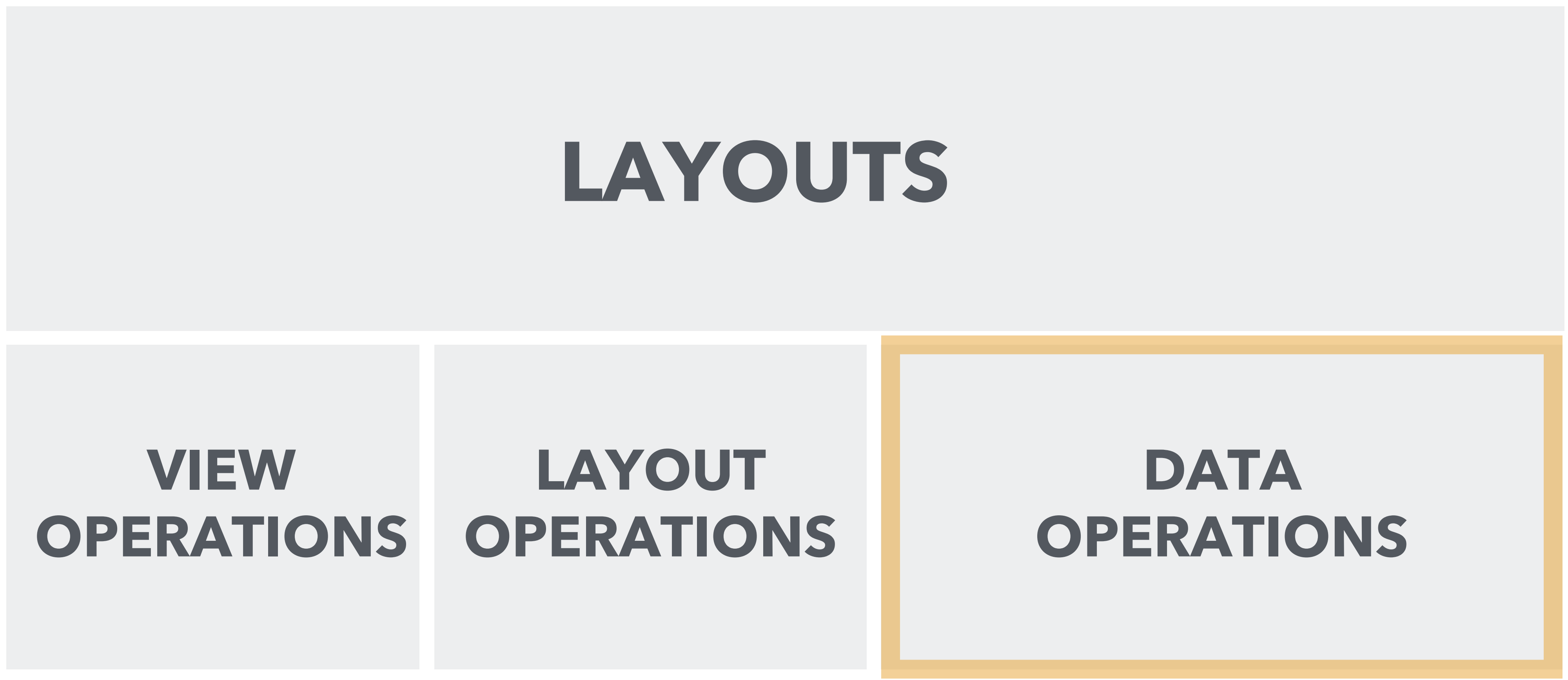


LAYOUTS

VIEW OPERATIONS | LAYOUT OPERATIONS | DATA OPERATIONS

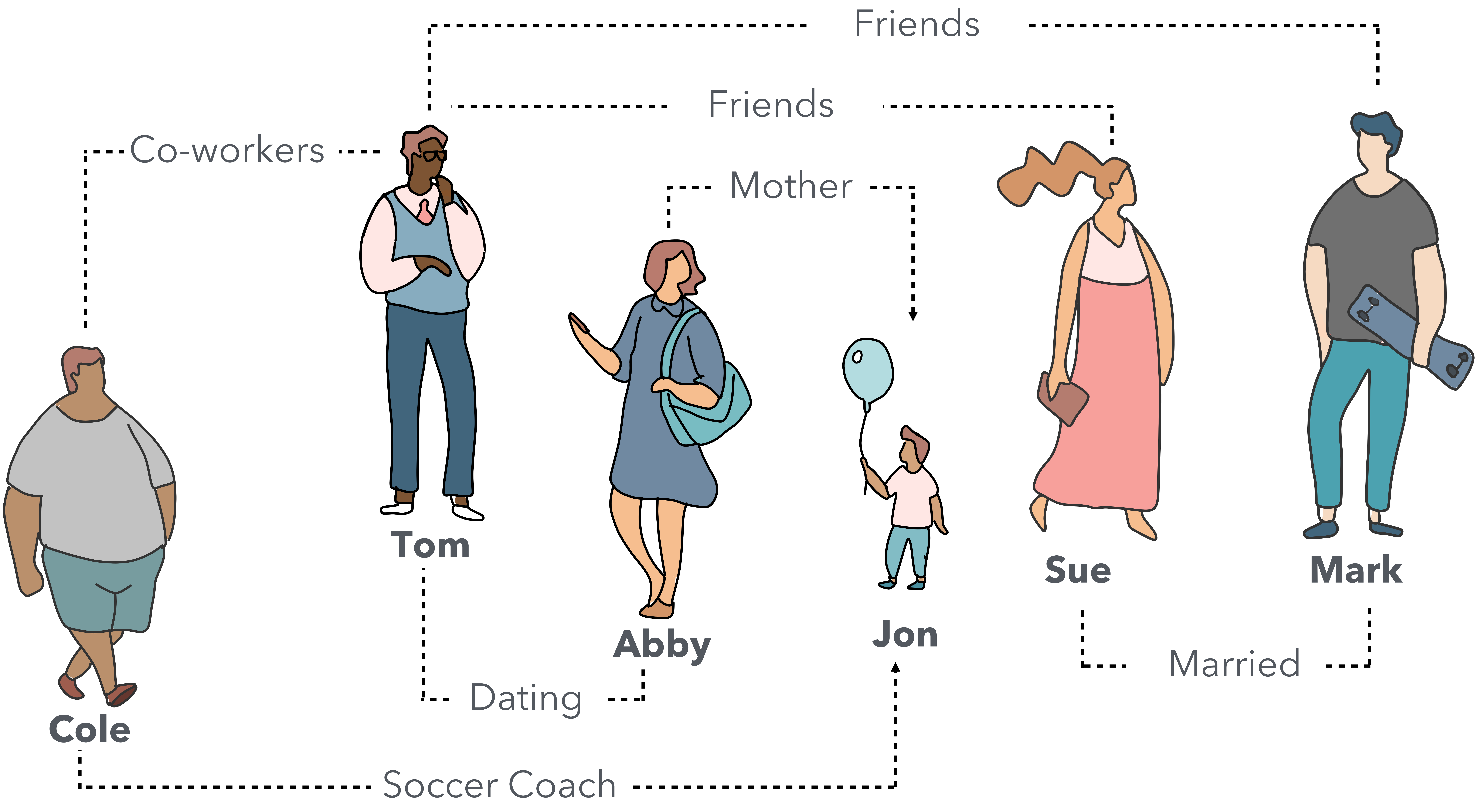
Juxtaposed Views

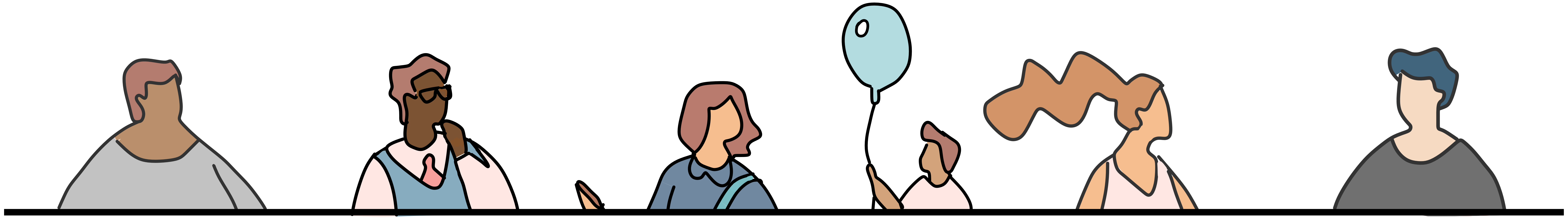



LAYOUTS
VIEW OPERATIONS
LAYOUT OPERATIONS
DATA OPERATIONS

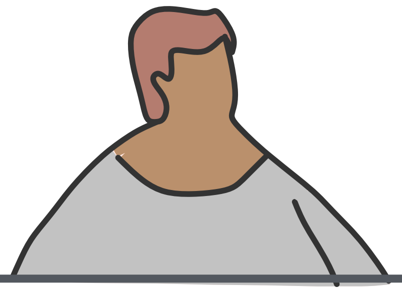
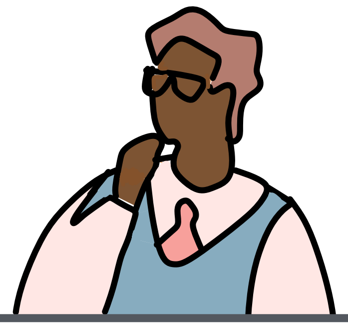
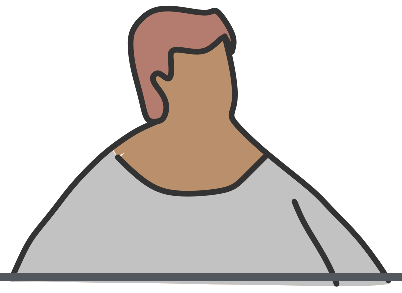
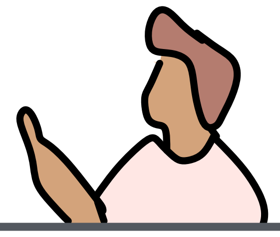
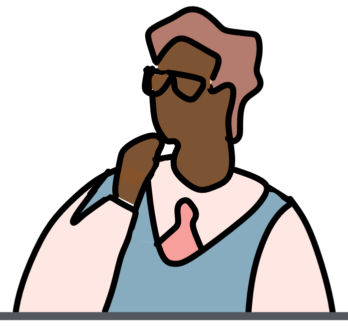
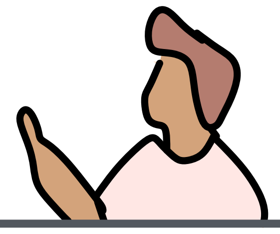
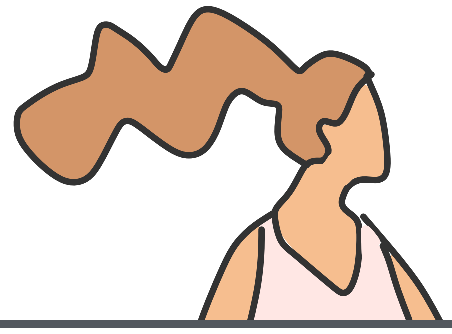
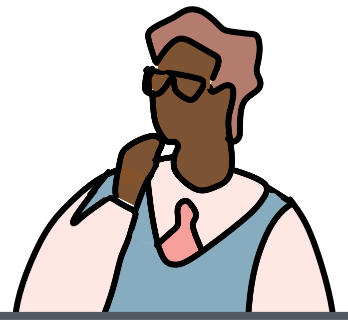
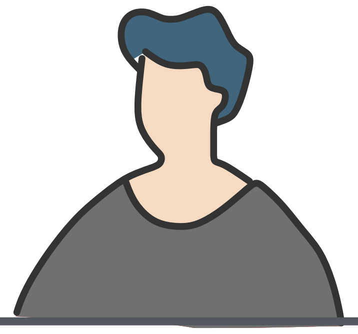
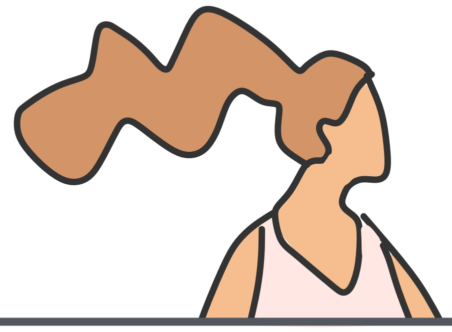
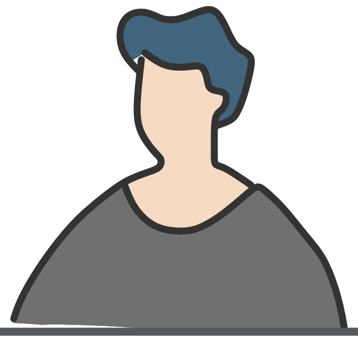
Filter Data


Attribute

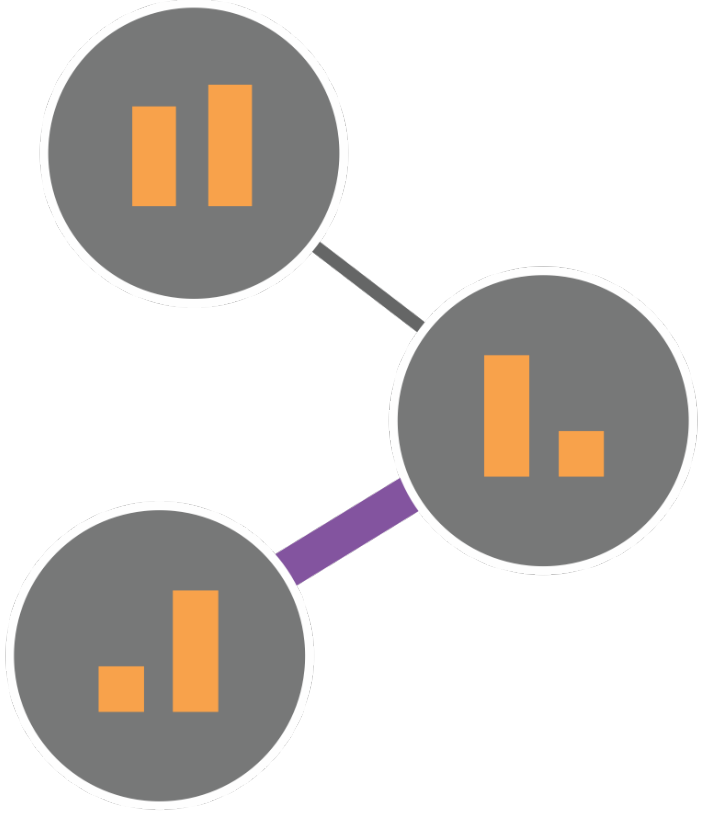

Friends
Friends
Co-workers
Mother
Tom
Abby
Jon
Sue
Mark
Cole
Dating
Married
Soccer Coach

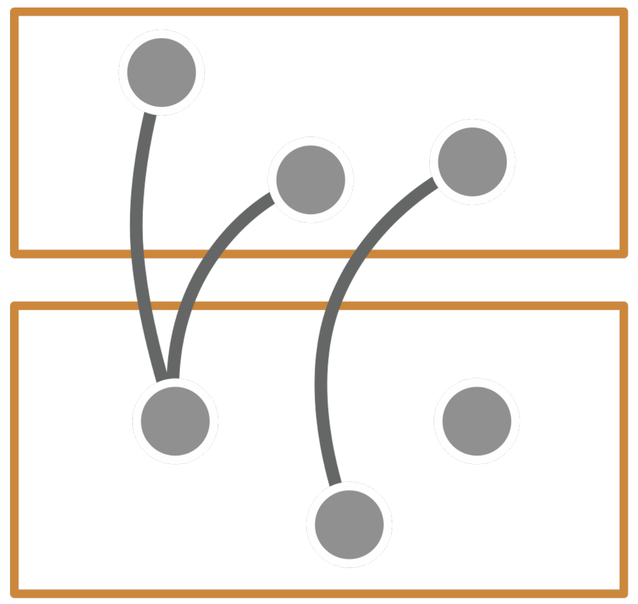


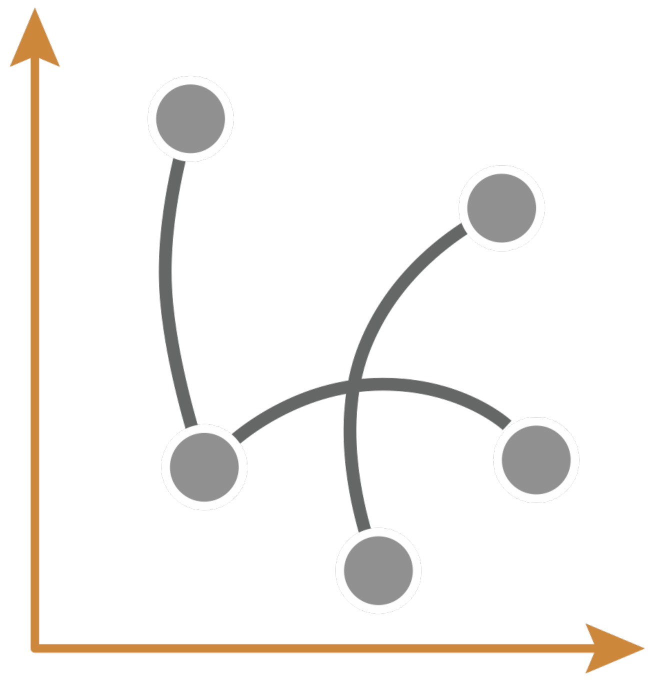
Name	Cole	Tom	Abby	Jon	Sue	Mark
Beverage	Port	Beer	Port	Coke	Coke	Beer
Day 1	1	0	4	3	3	5
Day 2	0	2	5	3	5	5
Day 3	4	1	2	2	4	3

Source	Target	Type	Duration
		Co-workers	3 years
		Soccer Coach	2 years
		Dating	1 year
		Mother / Son	7 years
		Friends	12 years
		Friends	3 years
		Married	6 years

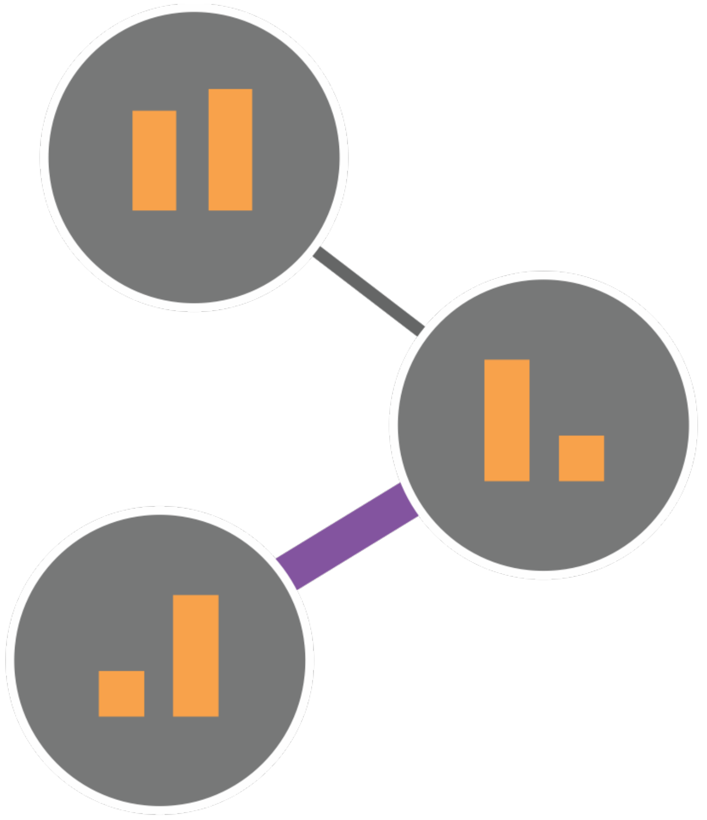
Node-Link Layouts



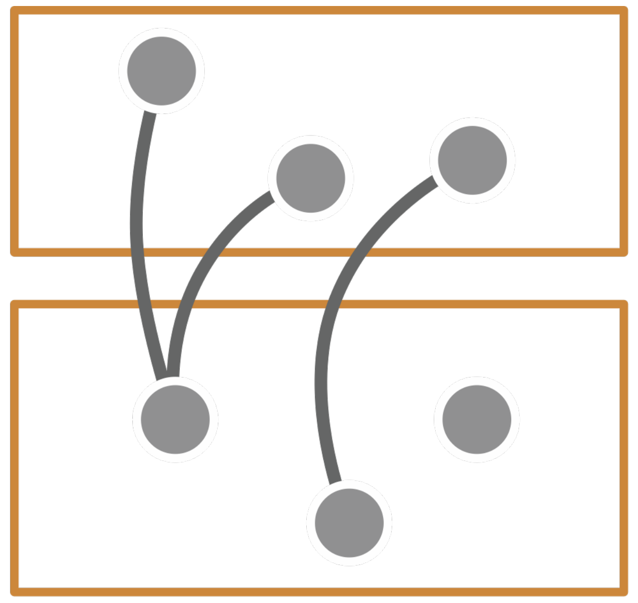


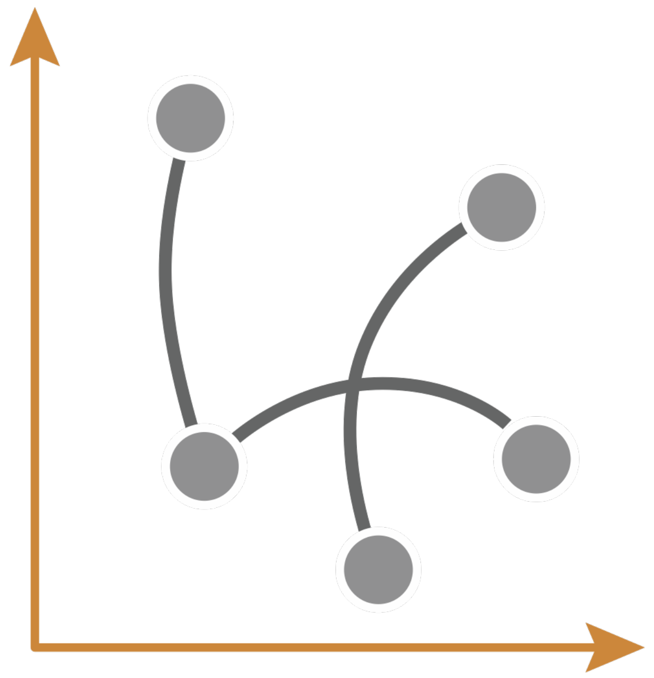


Topology Driven Layout

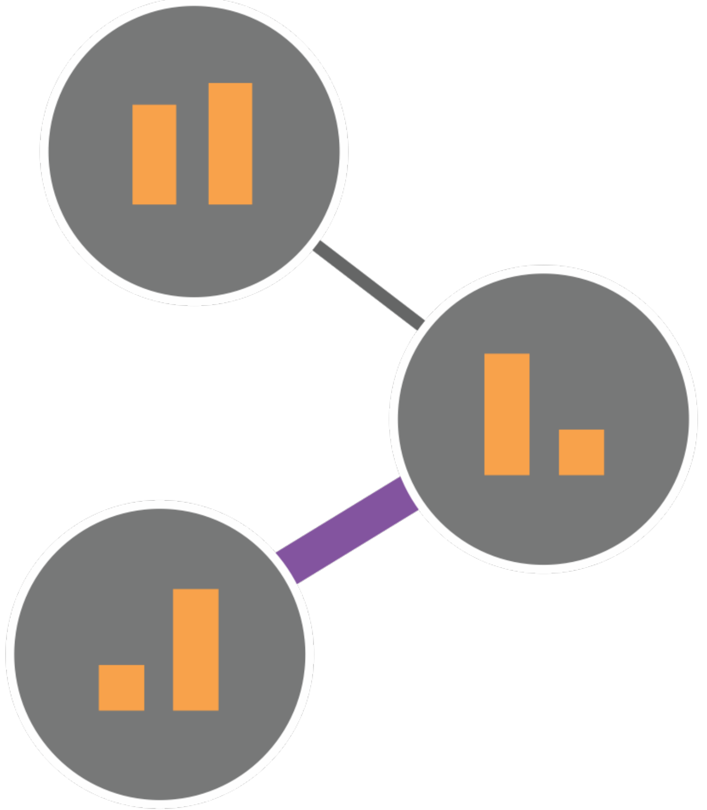


Attribute Driven Layouts



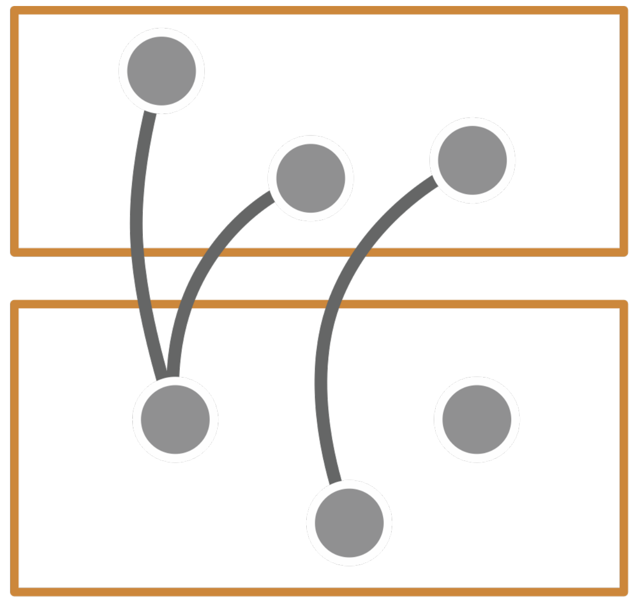


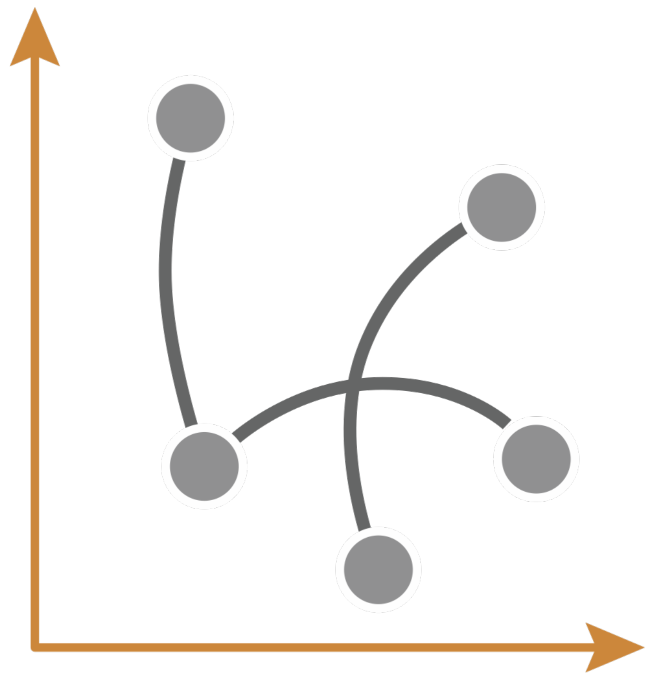
Topology Driven Layout



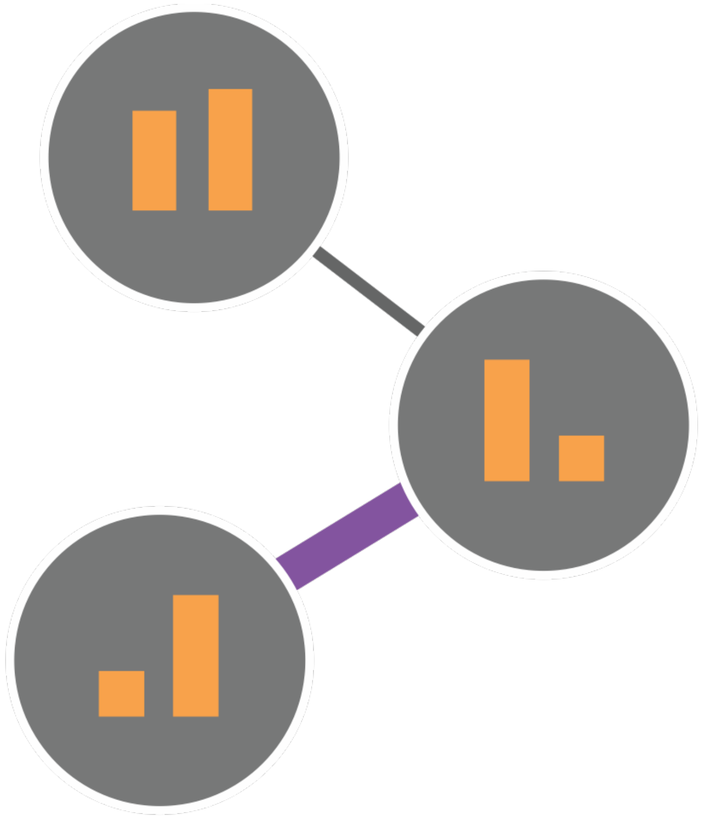
On-Node / On-Edge
Encoding

Attribute Driven Layouts



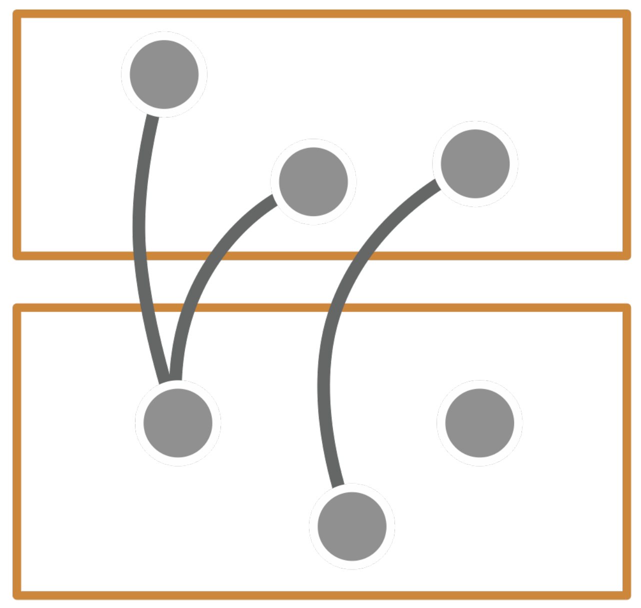


Topology Driven Layout

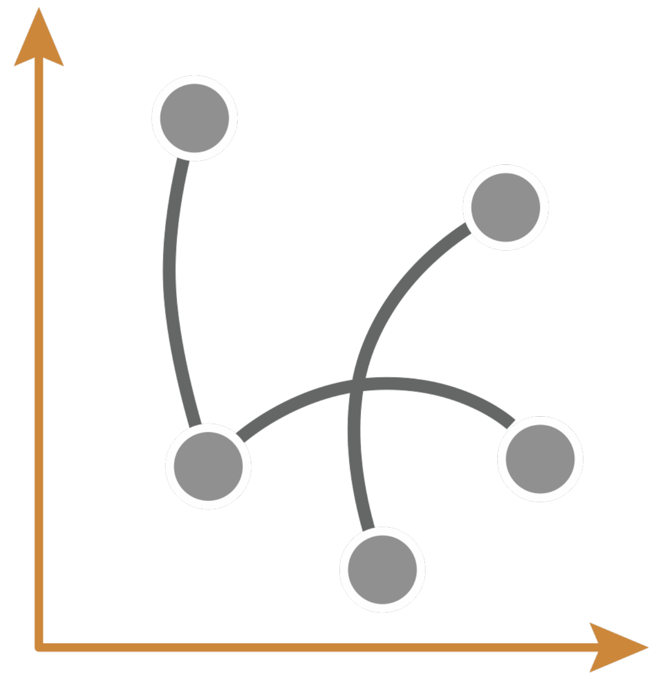


On-Node / On-Edge Encoding

Attribute Driven Layouts

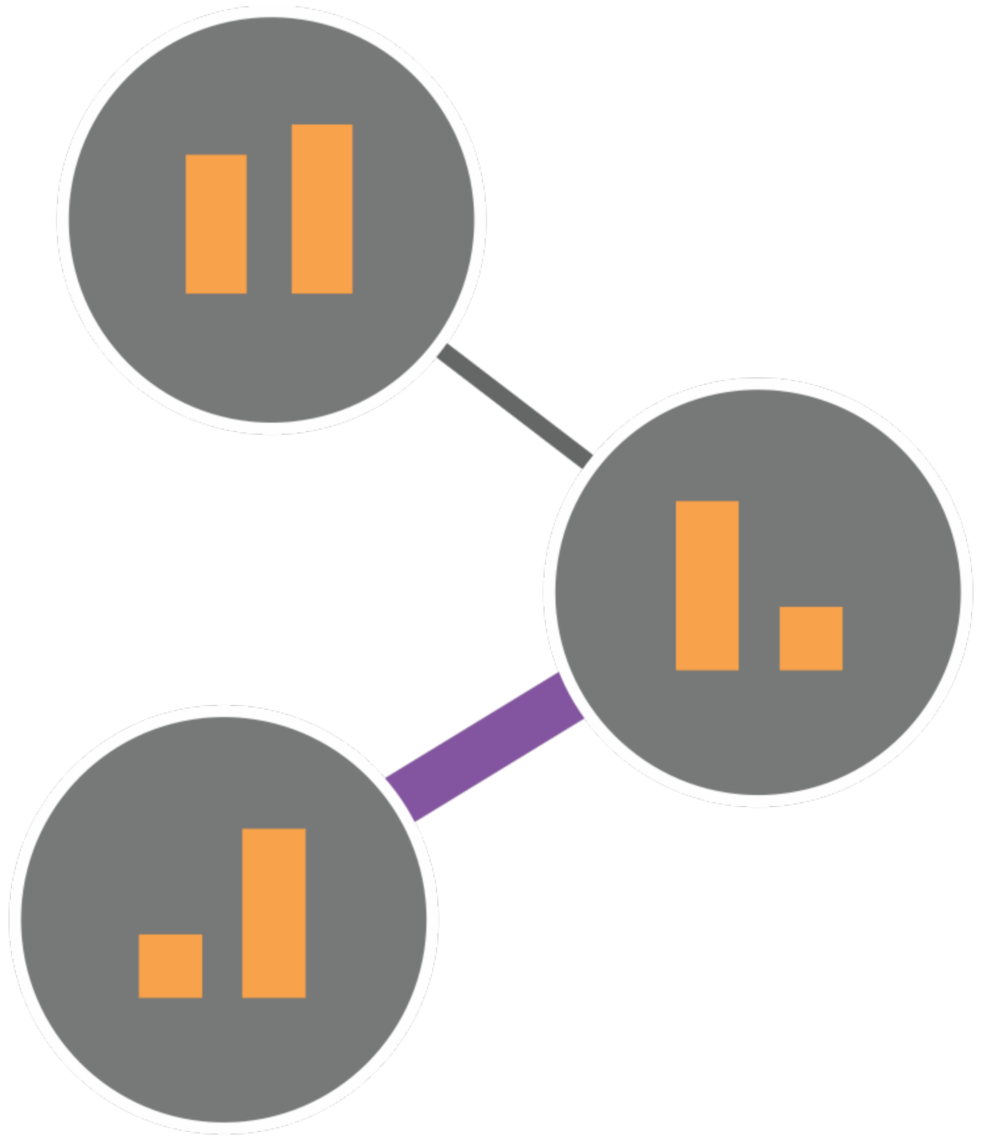


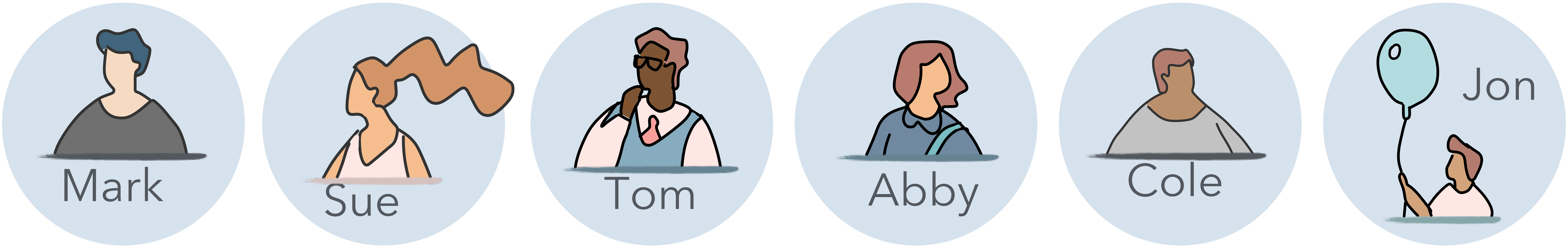
Attribute-Driven Faceting

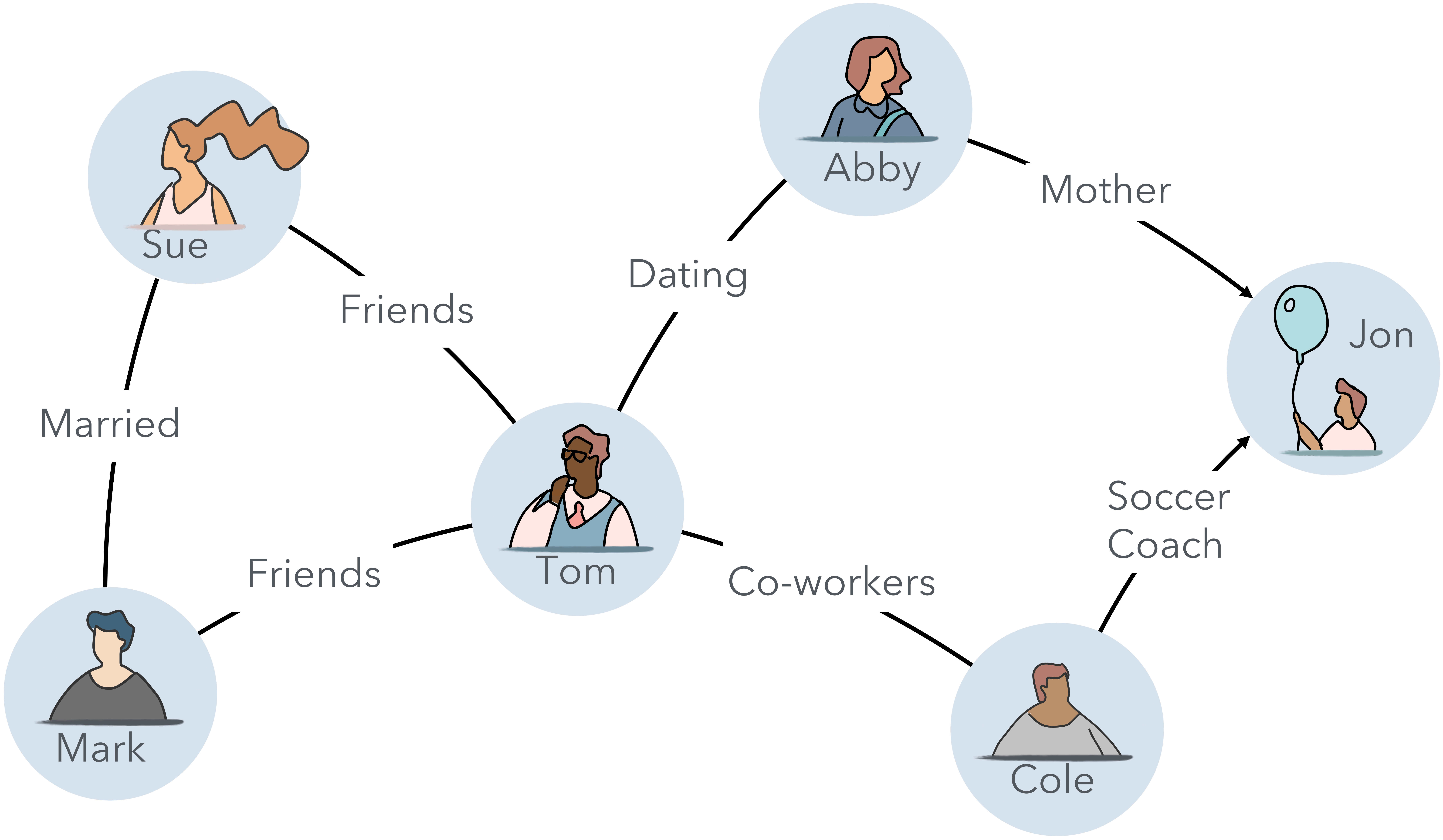


Attribute-Driven Positioning

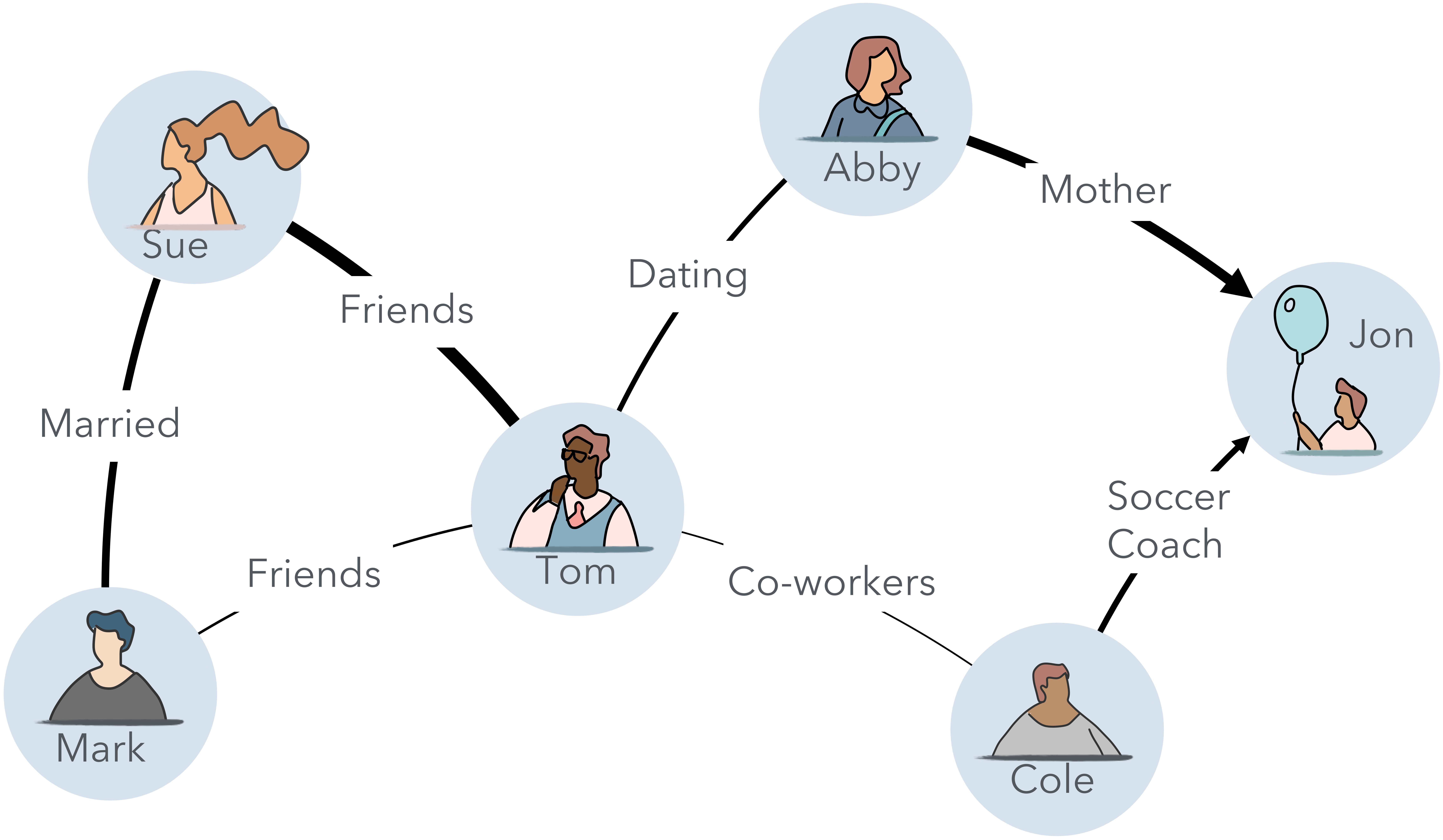
On-Node / On-Edge Encoding

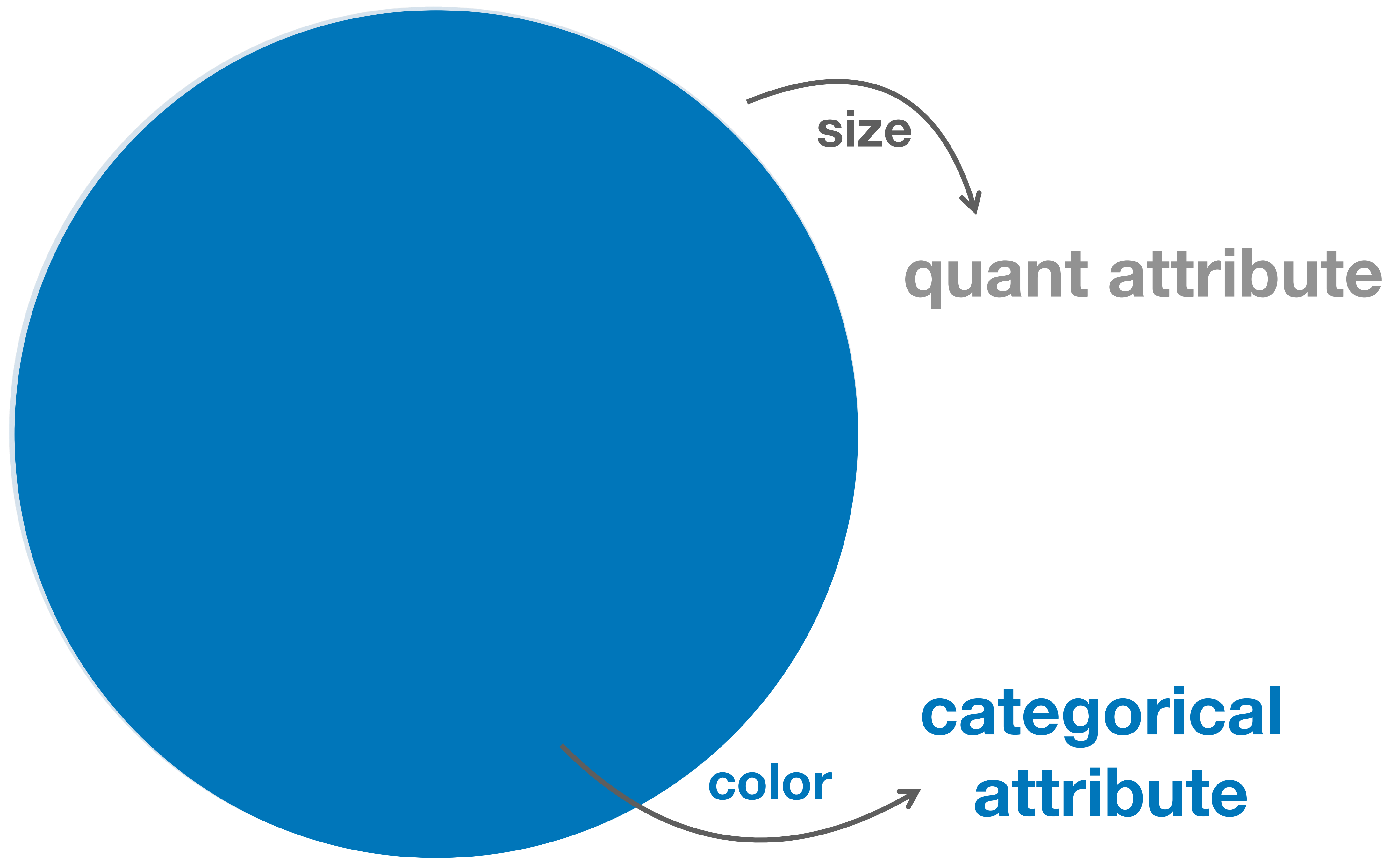


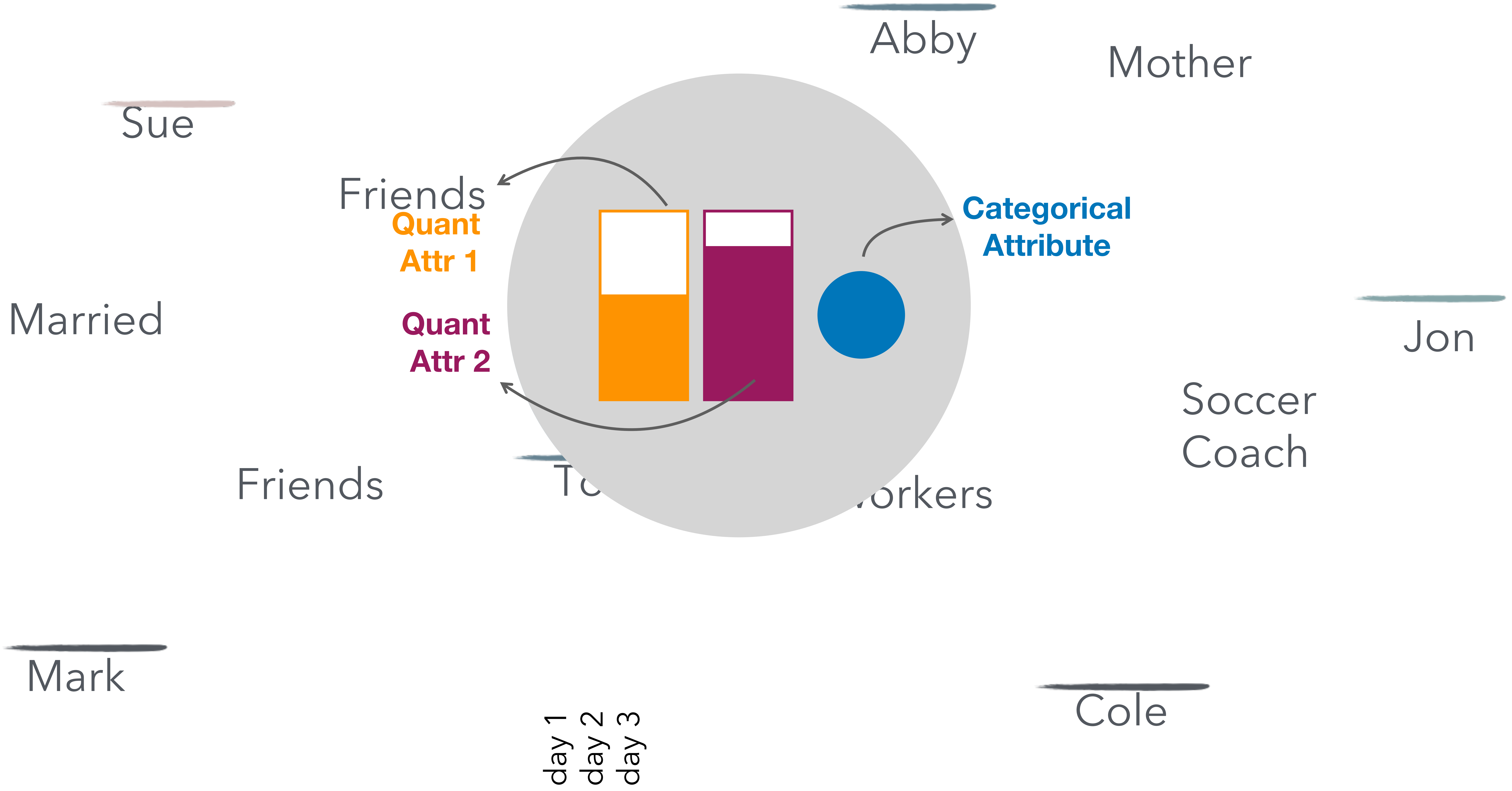

Mark
Sue
Tom
Abby
Cole
Jon

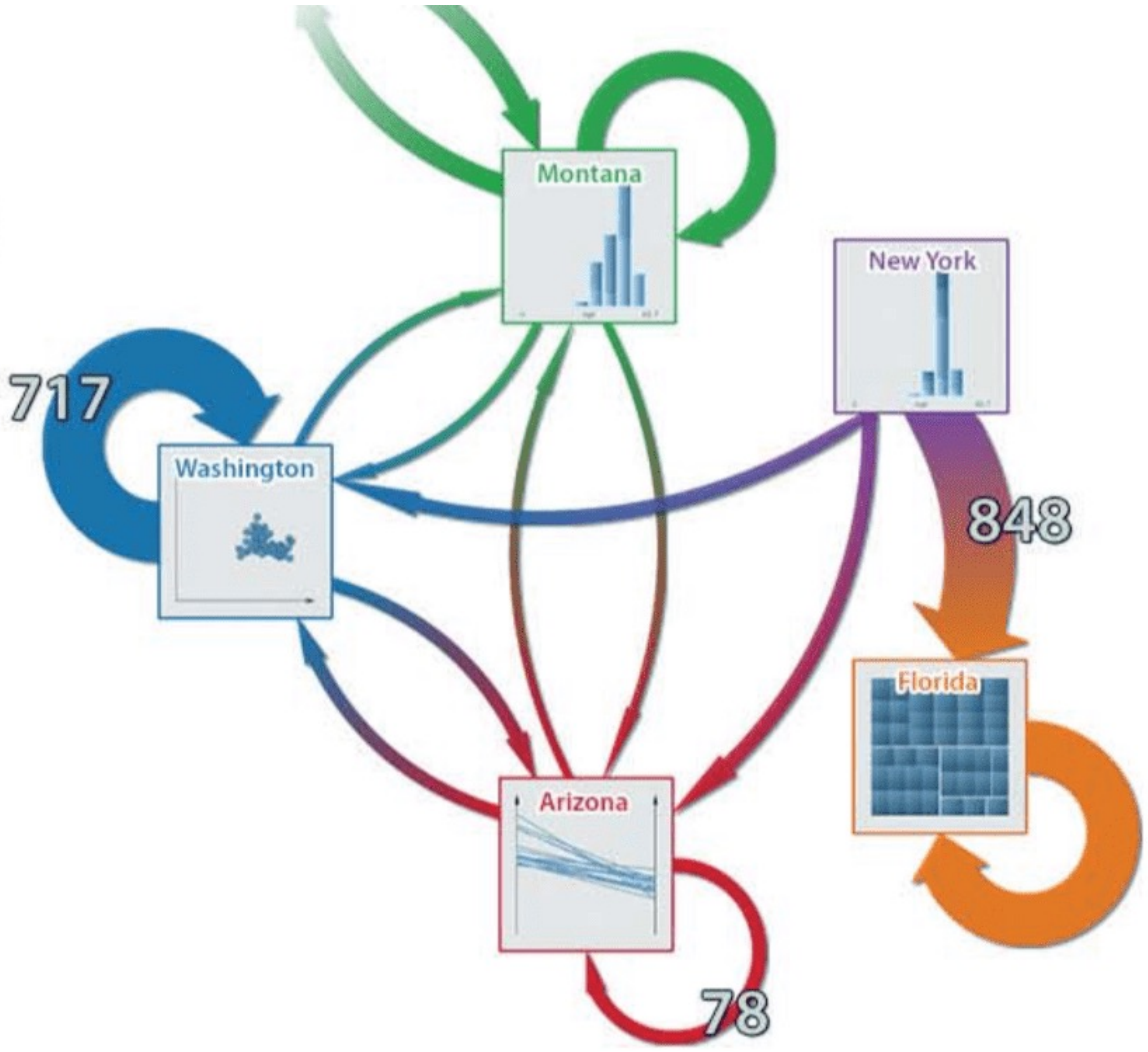


Sue
Abby
Mother
Friends
Dating
Jon
Married
Tom
Soccer
Coach
Friends
Co-workers
Mark
Cole

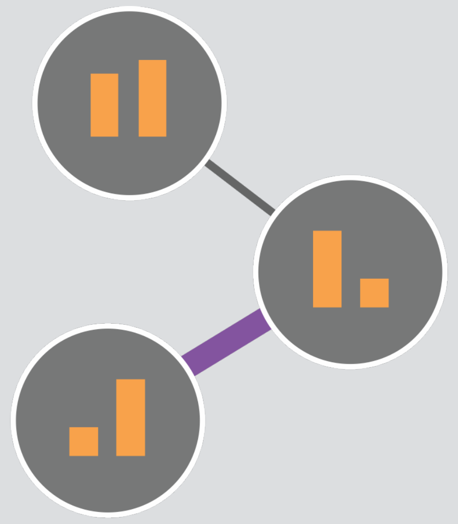

Sue
Abby
Mother
Jon
Friends
Dating
Married
Tom
Friends
Co-workers
Soccer
Coach
Mark
Cole


size
quant attribute
color
categorical attribute

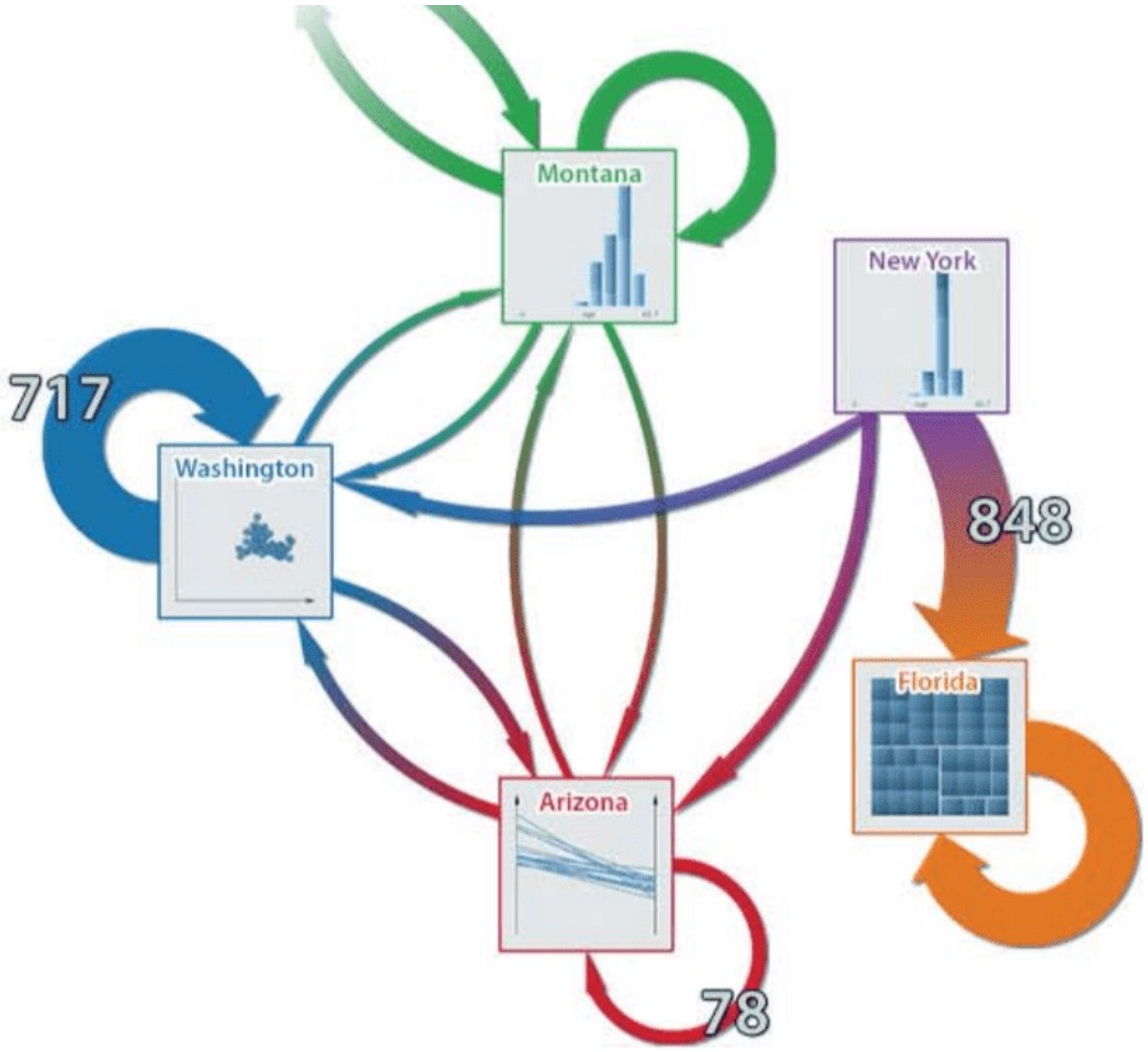

Abby
Mother
Sue
Friends
Quant
Attr 1
Categorical
Attribute
Married
Quant
Attr 2
Jon
Soccer
Coach
Friends
orkers
Mark
Cole
day 1
day 2
day 3



Elzen and Wijk, 2014

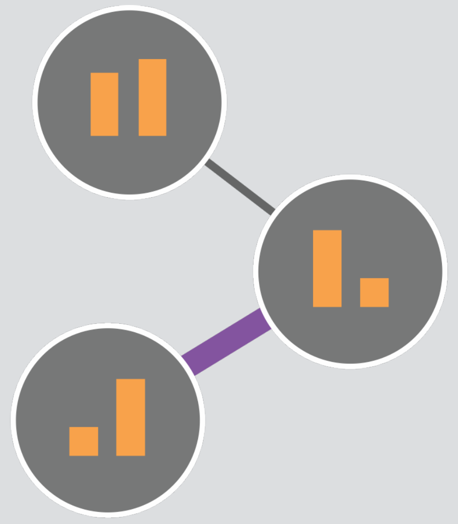


On-Node / On-Edge Encoding

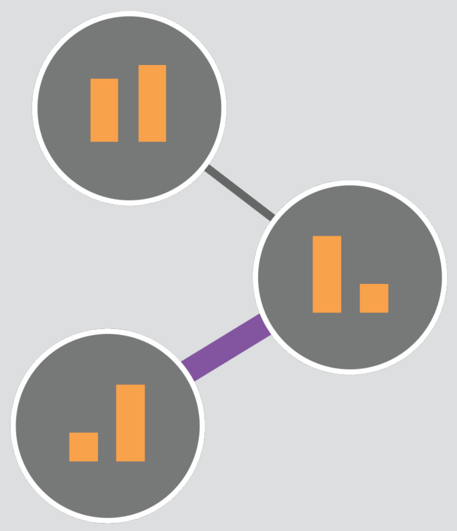


Elzen and Wijk, 2014

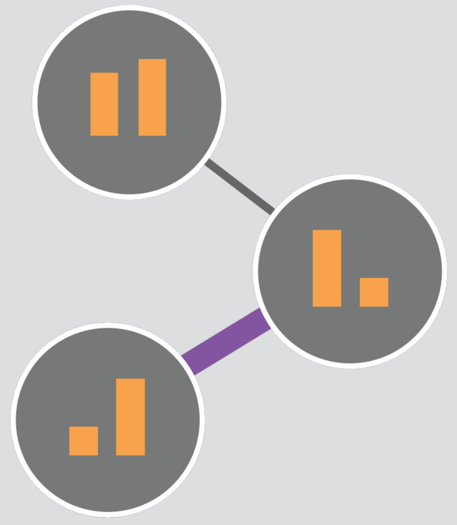
Aggregating Nodes/Edges



On-Node / On-Edge
Encoding



On-Node / On-Edge Encoding



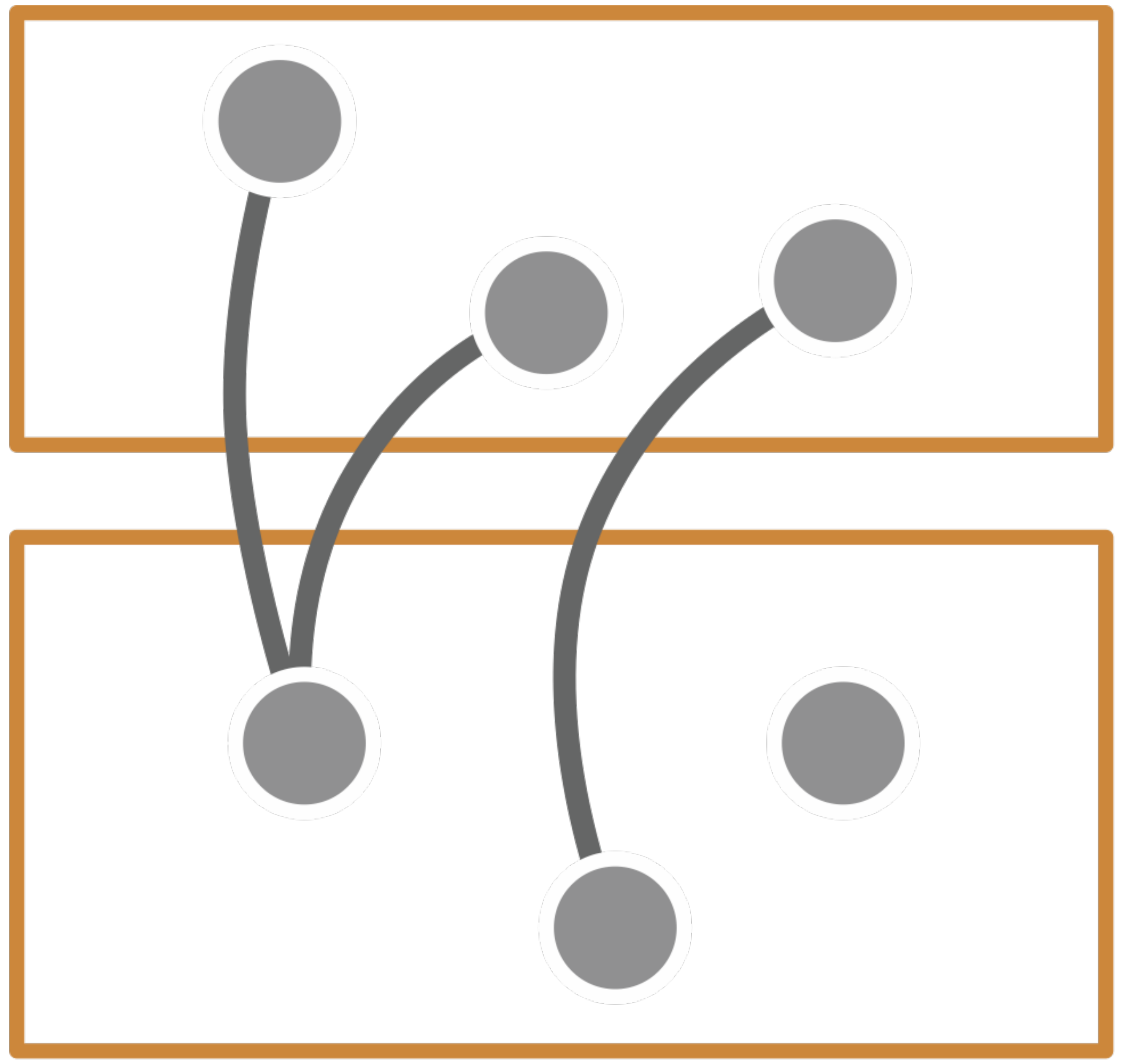
On-Node / On-Edge Encoding

Is easily understood by most users
Works well for all types of networks

Scalability.
Node size leaves little space to encode attributes.

Recommended for small networks when only a few (usually under five) attributes on the nodes are shown, or in combination with a zooming/filtering strategy

Attribute-Driven Faceting




Coca-Cola
classic

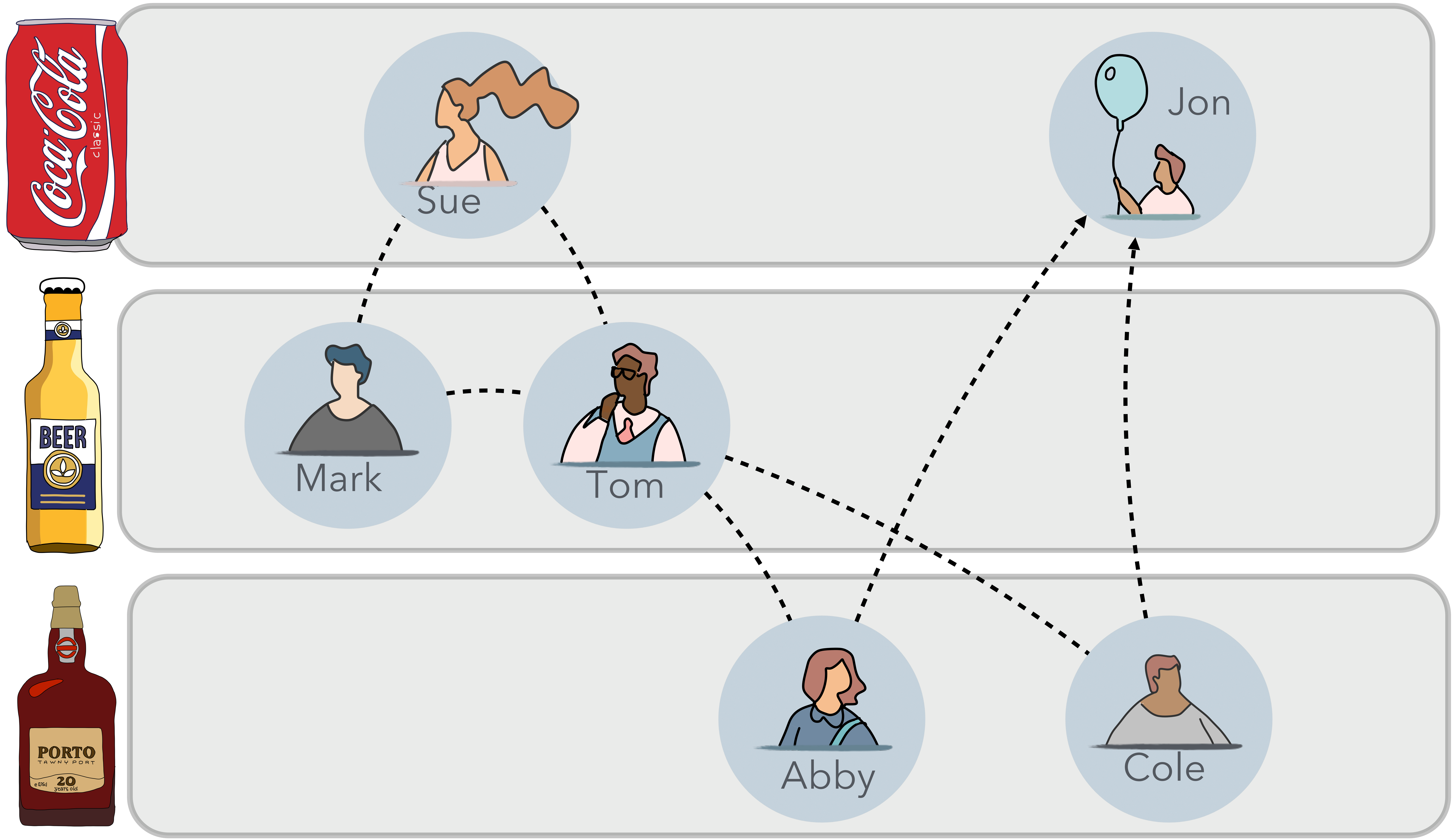

BEER


PORTO
TAWNY PORT
e 0,75cl
20
years old

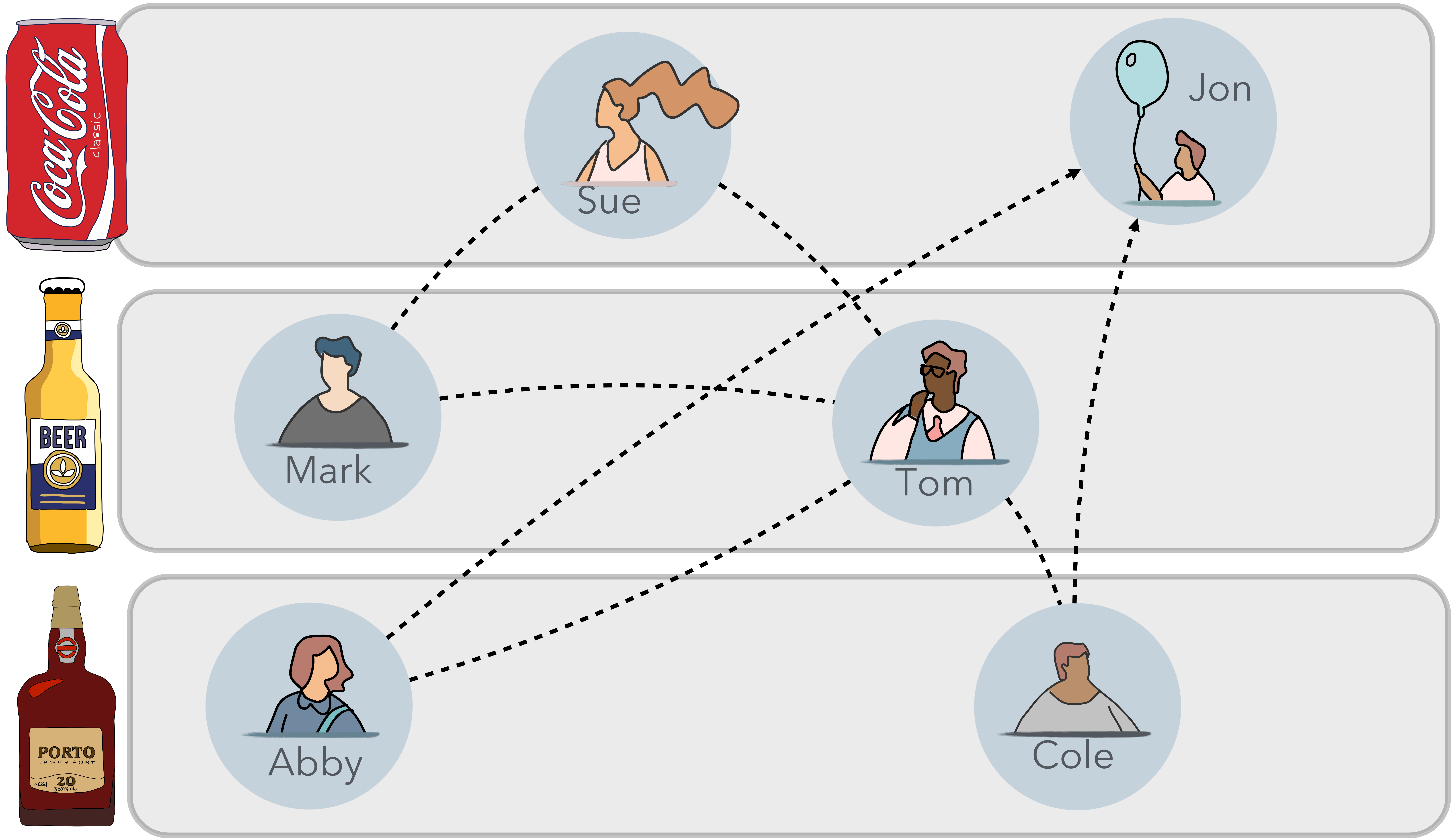

Coca-Cola
classic


BEER


PORTO
TAWNY PORT
20
years old

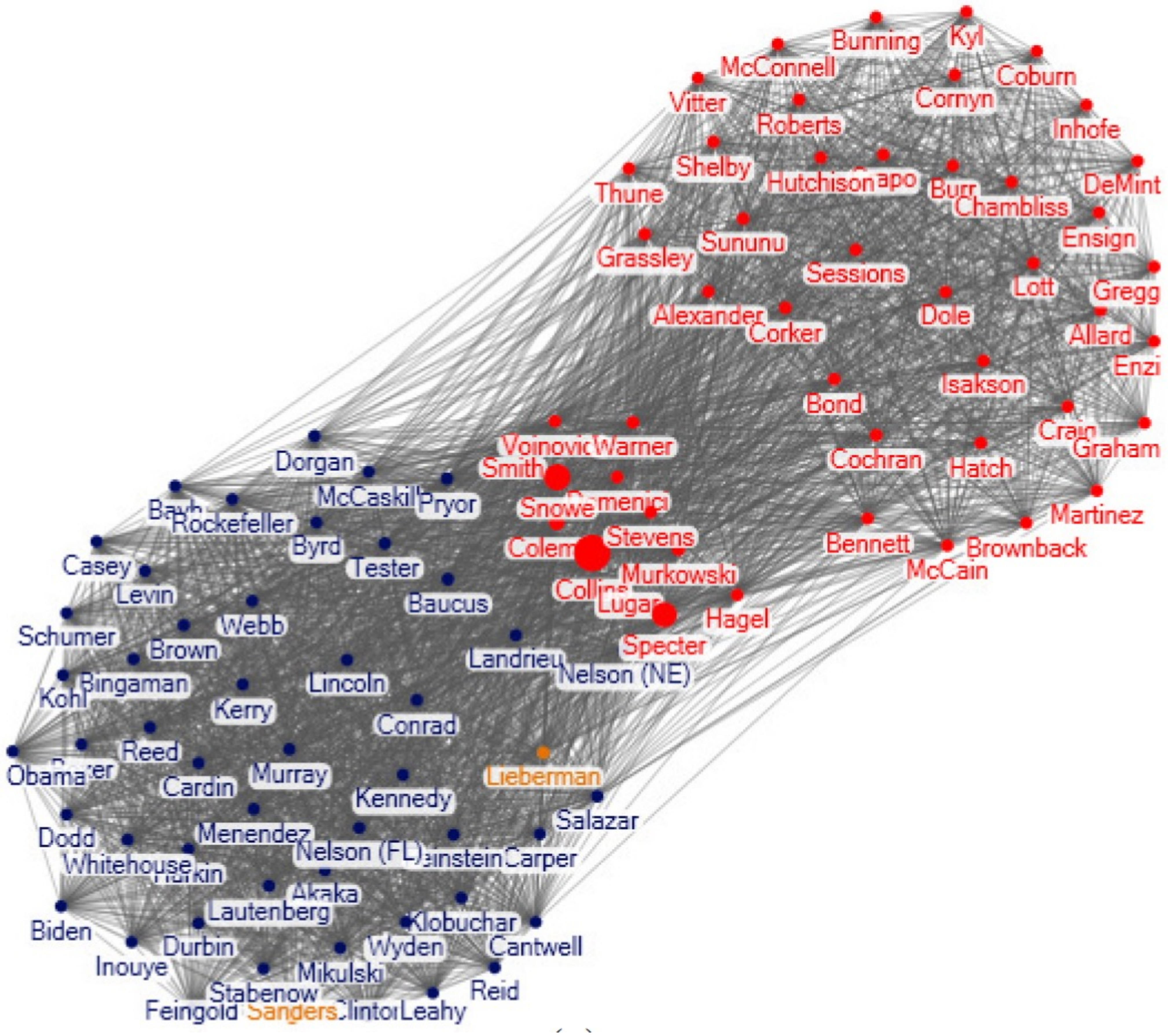


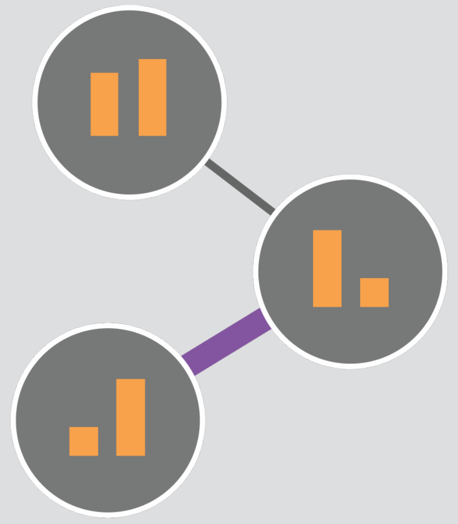
Coca-Cola classic
Sue
Jon
BEER
Mark
Tom
PORTO
TAWNY PORT
20
years old
Abby
Cole



Coca-Cola classic
BEER
PORTO
TAWNY PORT
20 years old
Sue
Jon
Mark
Tom
Abby
Cole

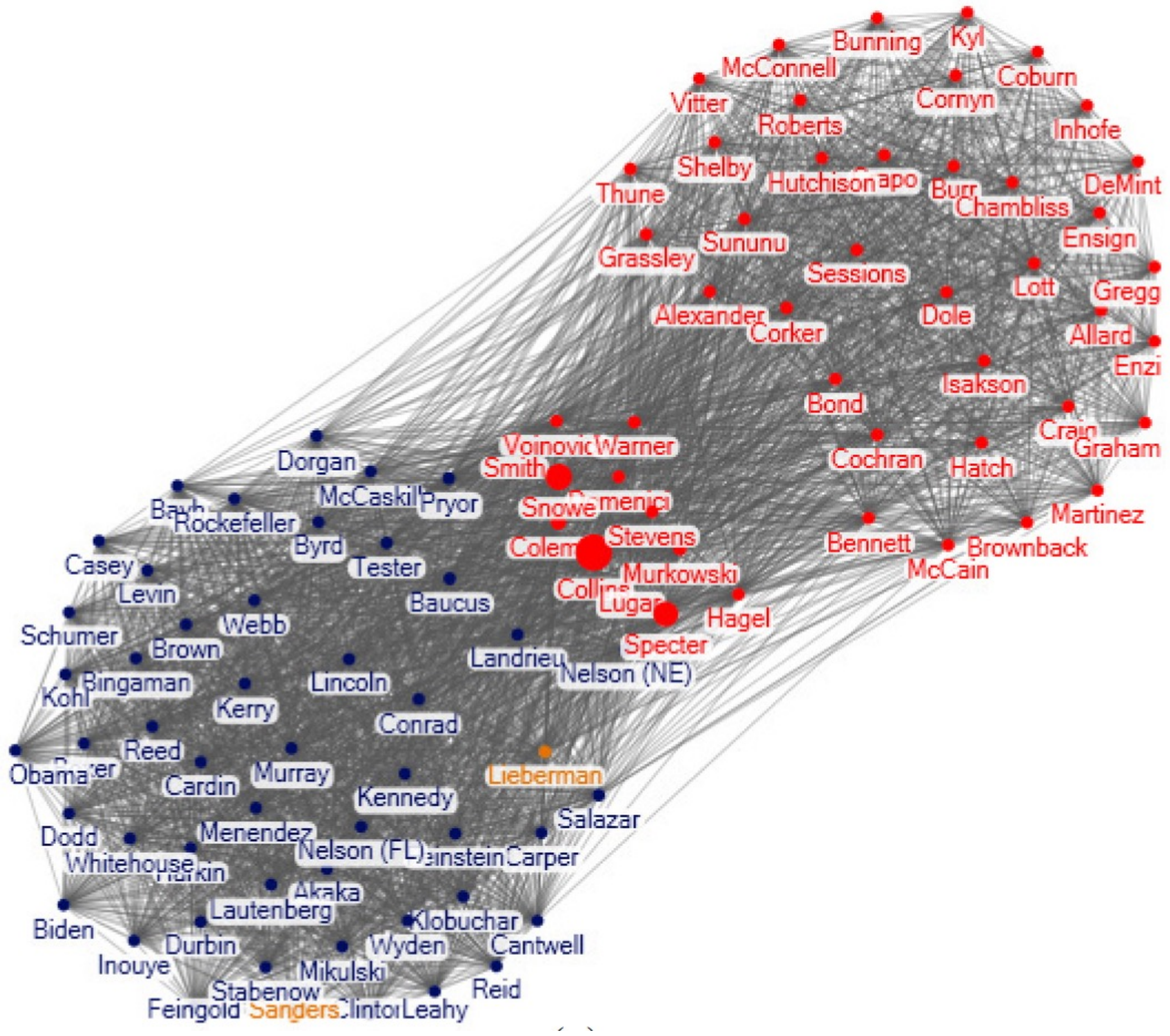
Group-in-a-box *Rodrigues et al. 2011*

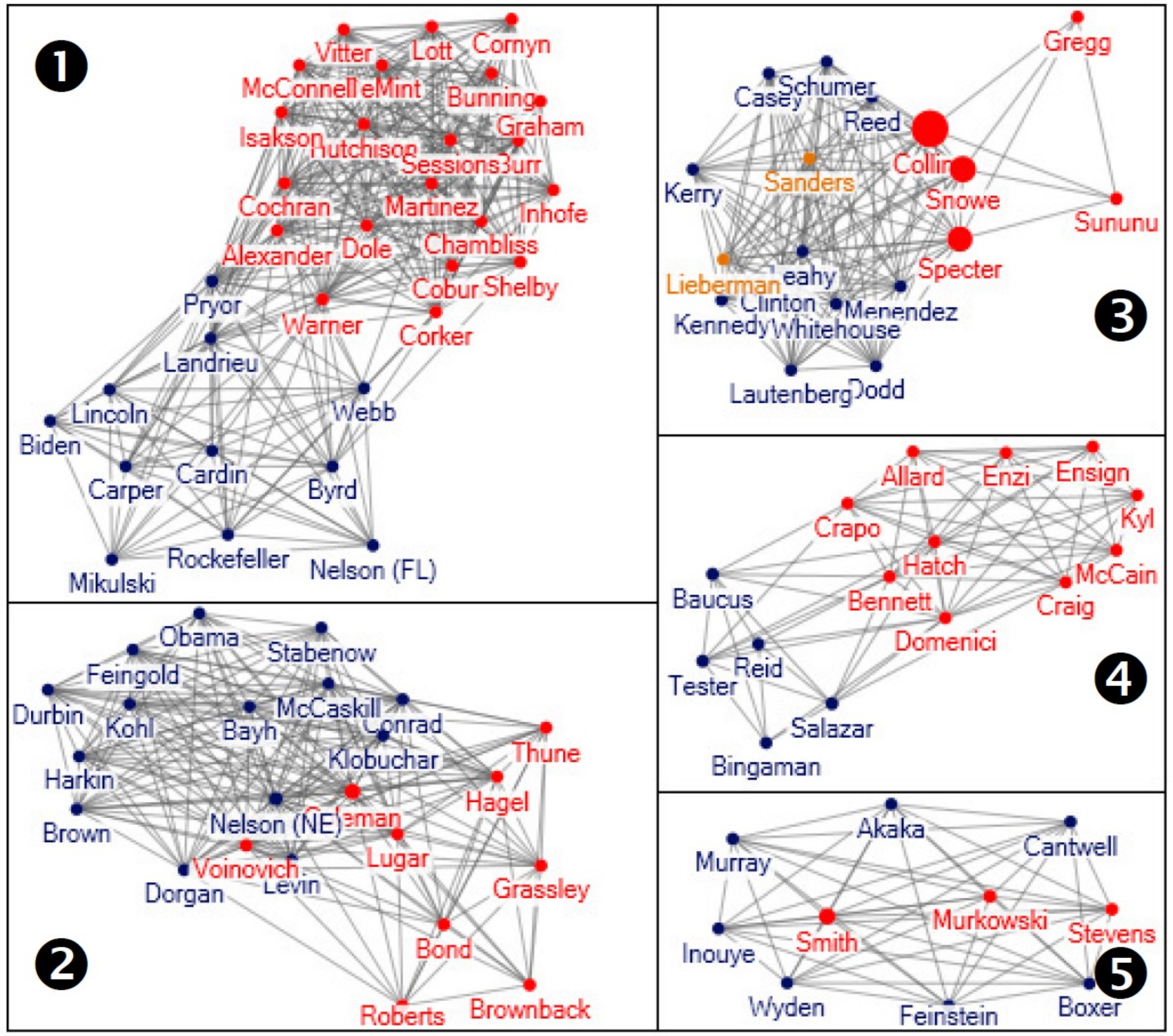


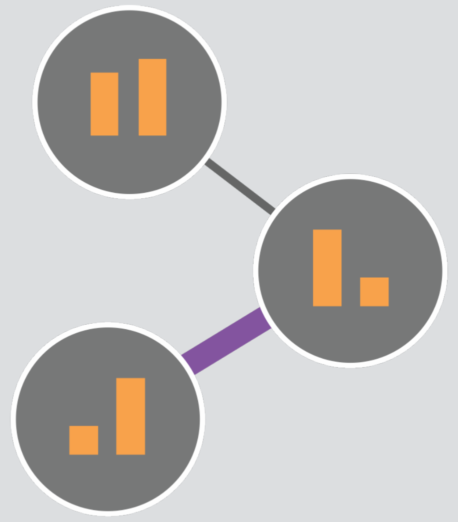


On-Node / On-Edge
Encoding

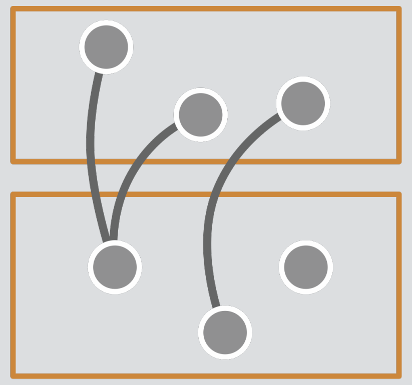
Group-in-a-box *Rodrigues et al. 2011*





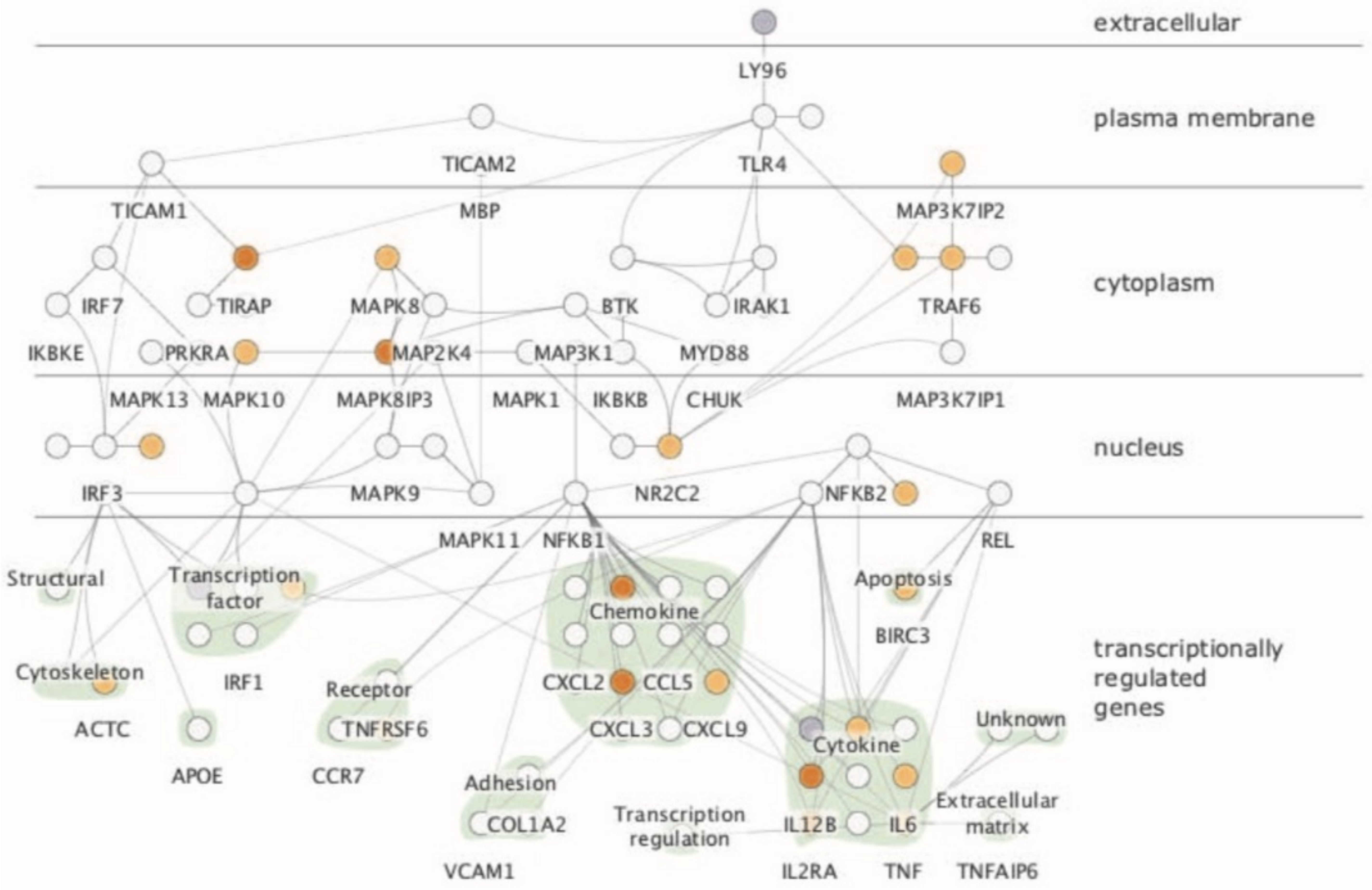


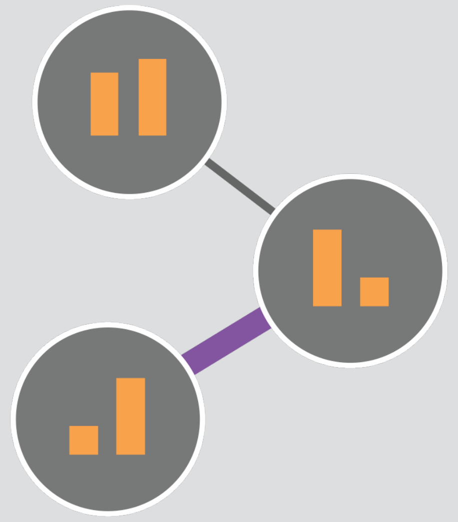
On-Node / On-Edge Encoding



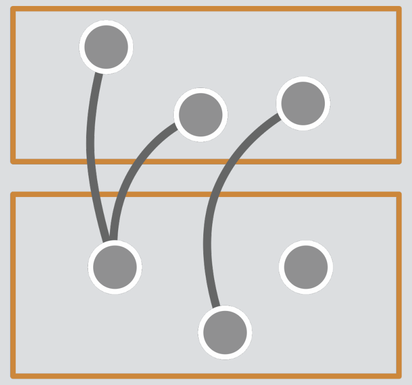
Attribute-Driven Faceting

Cerebral *Barsky et al. 2008*

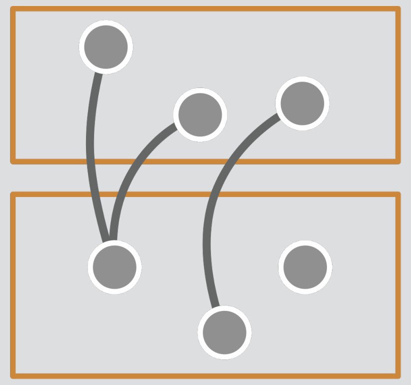




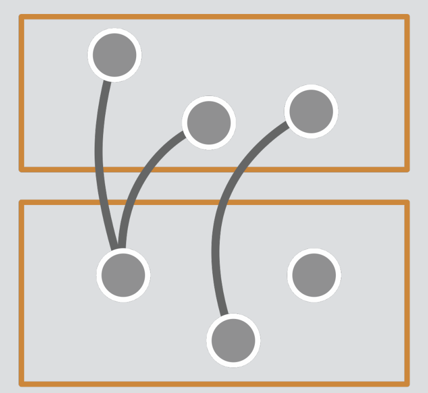
On-Node / On-Edge Encoding



Attribute-Driven Faceting



Attribute-Driven Faceting



Attribute-Driven Faceting

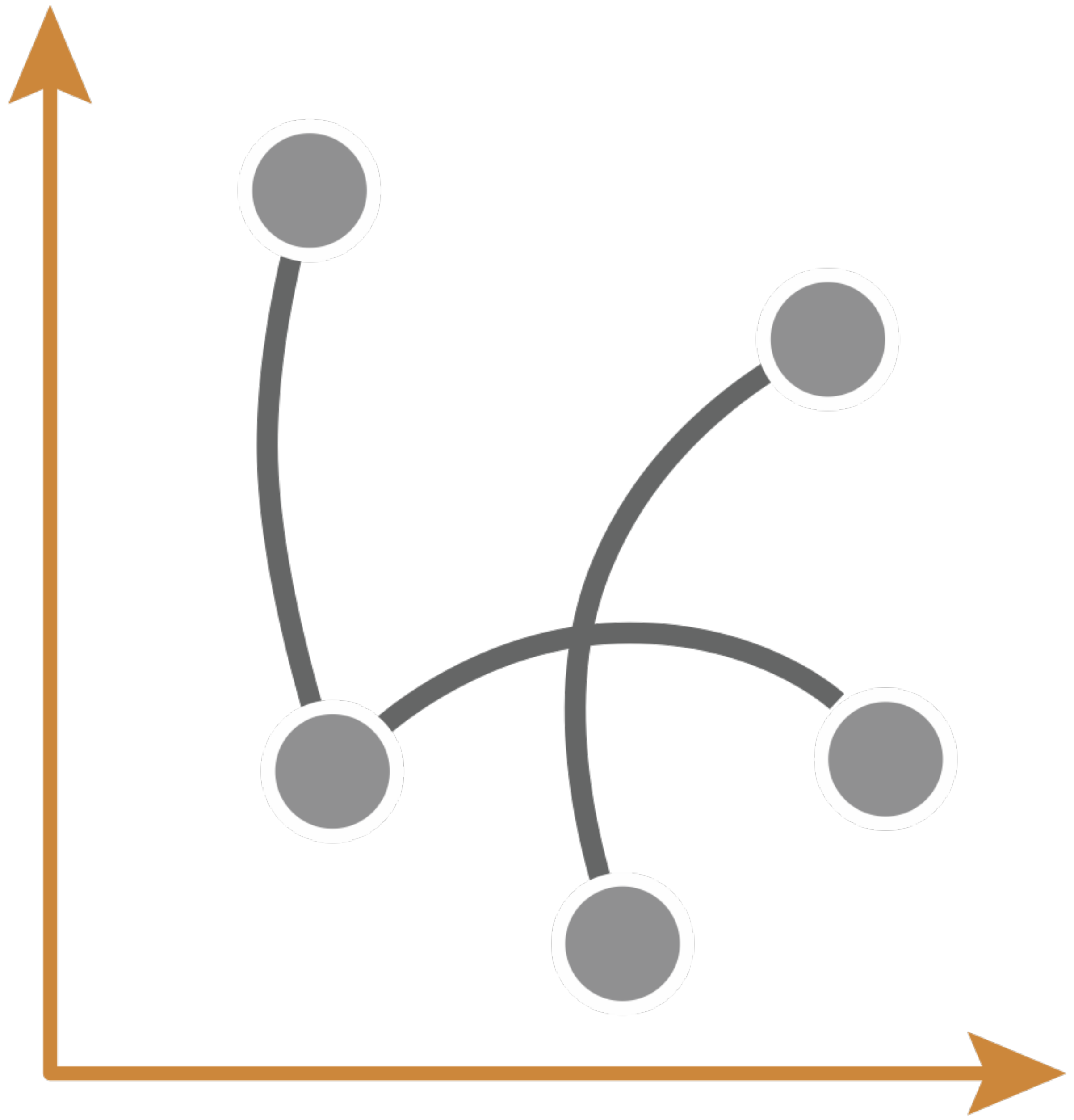
Well suited for networks with different node types or with an important categorical or set-like attribute.

Less scalable with respect to the number of nodes and network density than node-link layouts.

Neighborhoods, paths, and clusters are not easily visible if they span different facets.

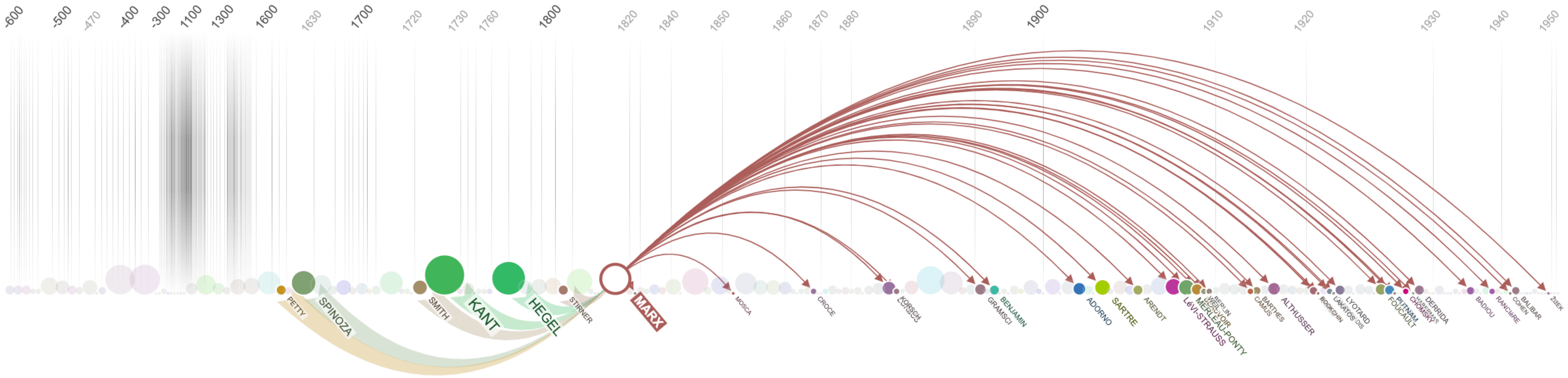
Recommended for networks where nodes can be separated into groups easily and where these groups are central to the analysis

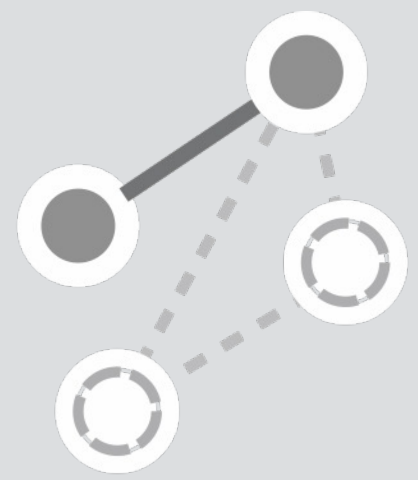
Attribute-Driven Positioning



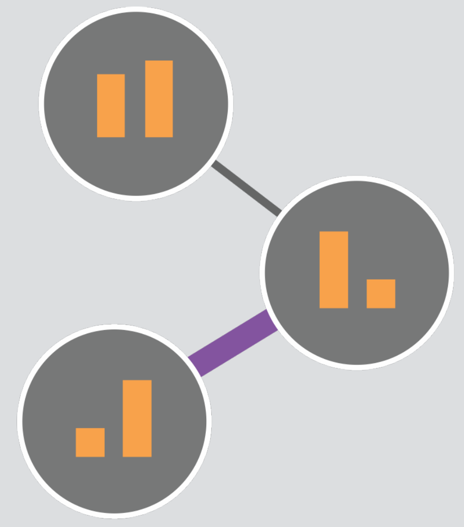

ANCHORAGE
VANCOUVER
EDMONTON
SEATTLE
PORTLAND
SAN FRANCISCO
DENVER
MINNEAPOLIS / ST. PAUL
CHICAGO
KANSAS CITY
DALLAS
TORONTO
CLEVELAND
MONTREAL
BALTIMORE
WASHINGTON D.C.
PHILADELPHIA
NEW YORK
JFK & NEWARK
BOSTON
HALIFAX
TAMPA BAY
ORLANDO
ICELAND
HELSINKI
STOCKHOLM
OSLO
GOTHENBURG
BERGEN
COPENHAGEN
BILLUND
BERLIN
HAMBURG
FRANKFURT
MUNICH
AMSTERDAM
GLASGOW
BRUSSELS
ZURICH
MANCHESTER
MILAN
DUBLIN
LONDON
HEATHROW
& GATWICK
PARIS
ORLY & CDG
GENEVA
MADRID

Edge Map *Dork et al. 2011*

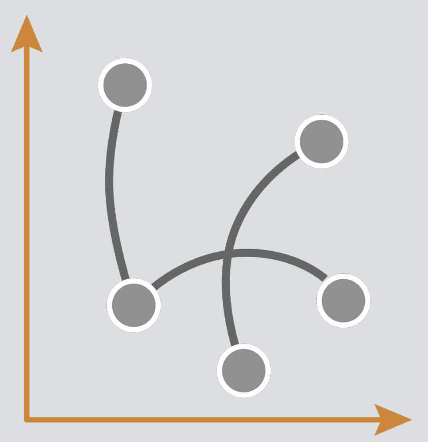




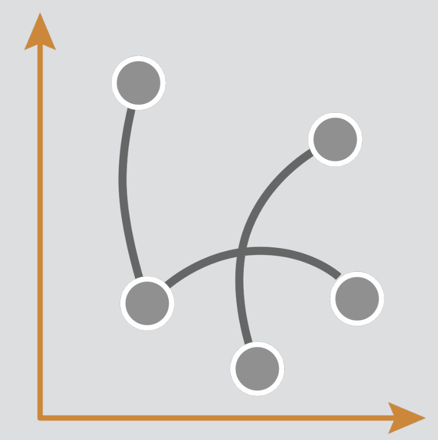
Querying and Filtering



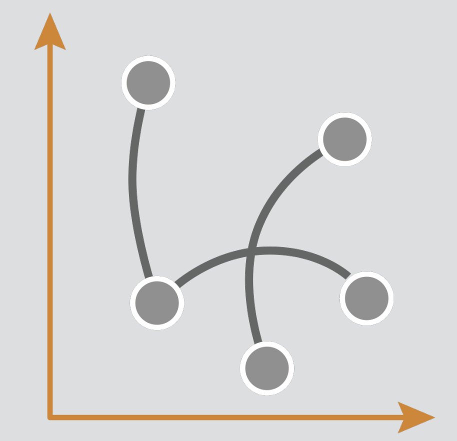
On-Node / On-Edge Encoding



Attribute-Driven Positioning



Attribute-Driven Positioning



Attribute-Driven Positioning

Well suited for quantitative attributes

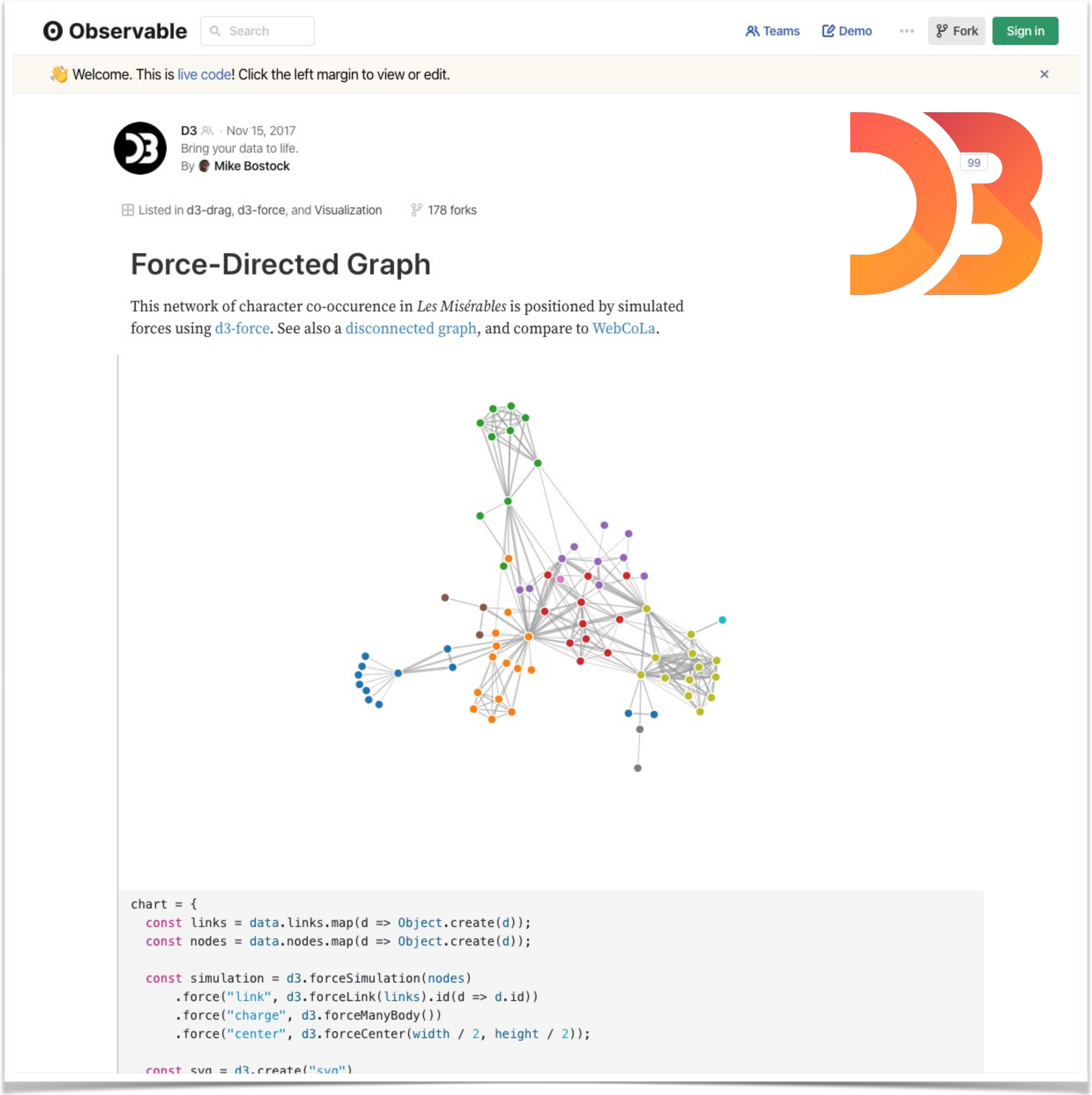
Does not lend itself well to visualizing the topology of the network.

Recommended for smaller, sparse networks where relationships between node attributes are paramount to the analysis task, and topological features only provide context

Tools and Applications

For graphic designer and developers

developer




```
chart = {
  const links = data.links.map(d => Object.create(d));
  const nodes = data.nodes.map(d => Object.create(d));

  const simulation = d3.forceSimulation(nodes)
      .force("link", d3.forceLink(links).id(d => d.id))
      .force("charge", d3.forceManyBody())
      .force("center", d3.forceCenter(width / 2, height / 2));

  const svg = d3.create("svg")
```

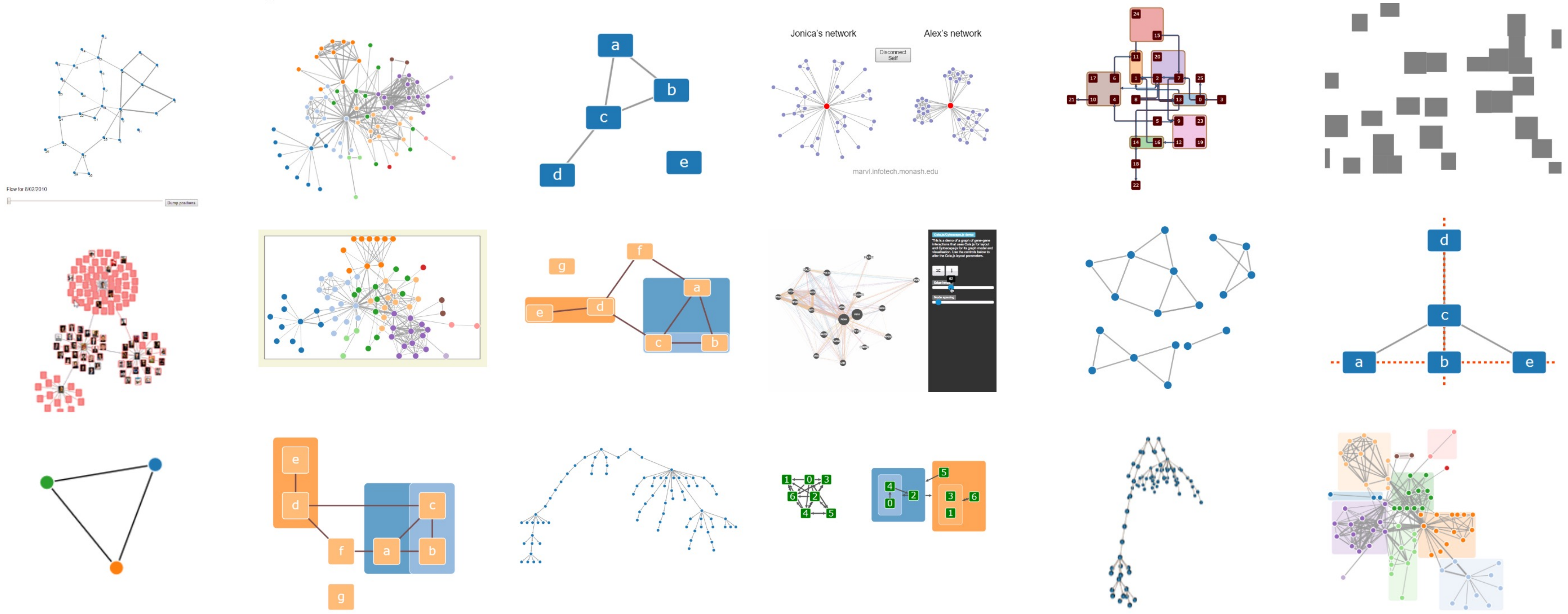
Cola.js (A.K.A. "WebCoLa") is an open-source JavaScript library for arranging your HTML5 documents and diagrams using constraint-based optimization techniques.

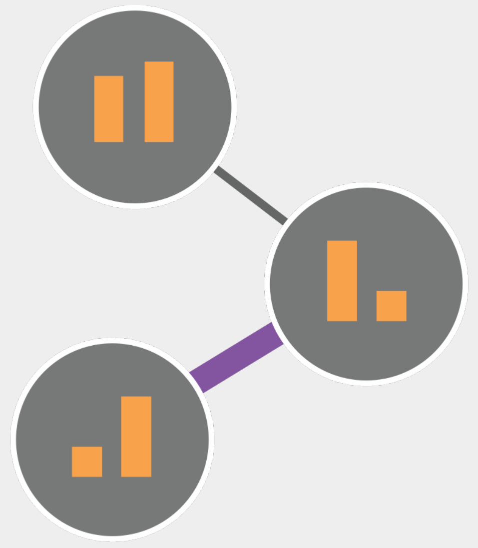
developer

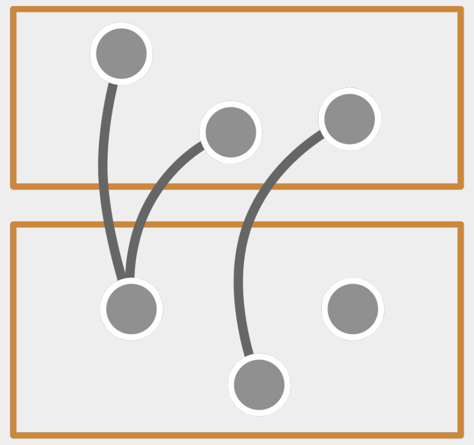
Overview Wiki API Source

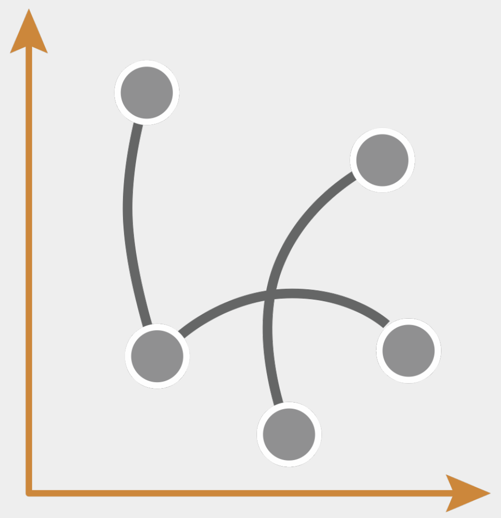
cola.js

Constraint-Based Layout in the Browser









developer

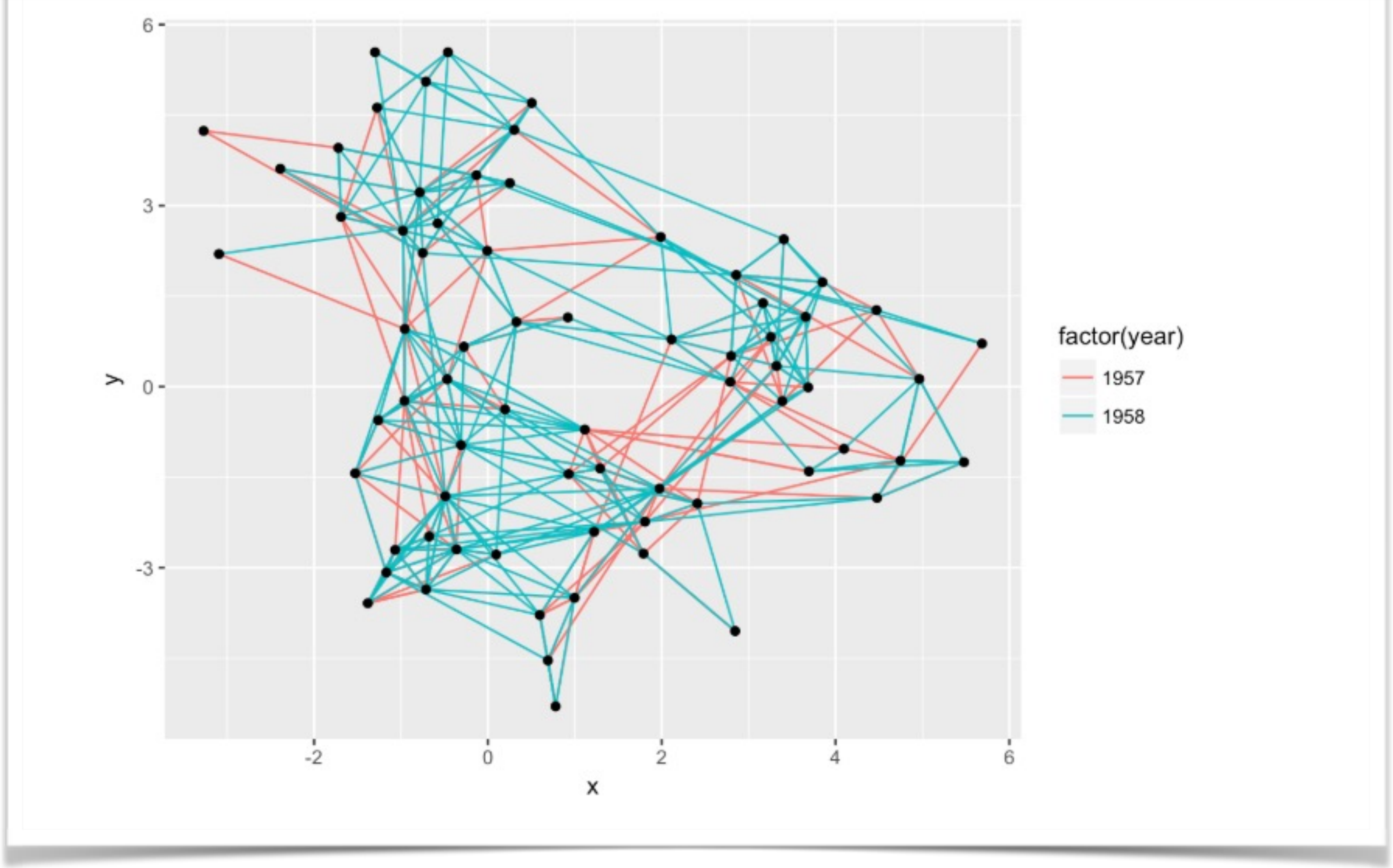
GGRAPH 1.0.2.9999 | Reference | Getting Started | Articles | News

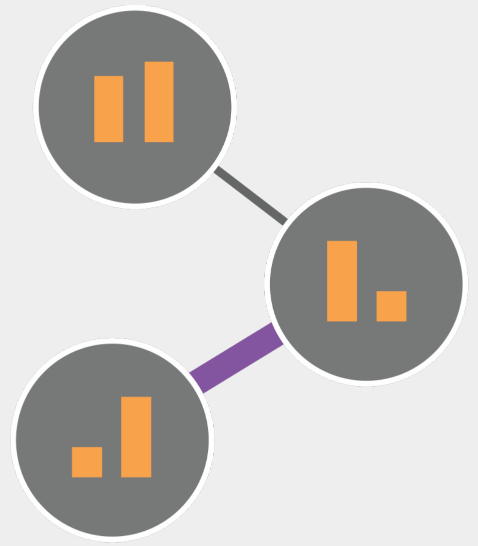
ggraph

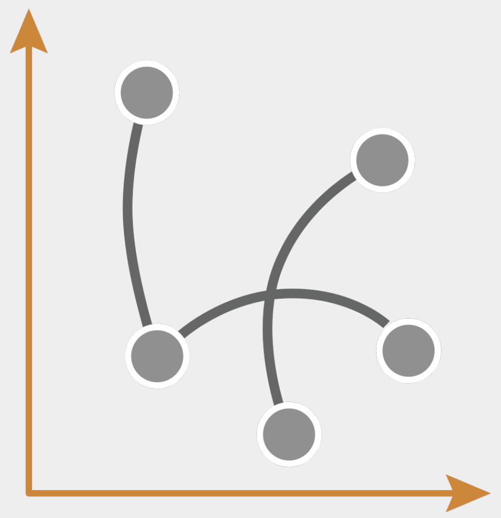
/dʒiː.dʒɪˈrɑːf/ (or g-giraffe)

A grammar of graphics for relational data

ggraph is an extension of `ggplot2` aimed at supporting relational data structures such as networks, graphs, and trees. While it builds upon the foundation of `ggplot2` and its API it comes with its own self-contained set of geoms, facets, etc., as well as adding the concept of *layouts* to the grammar.

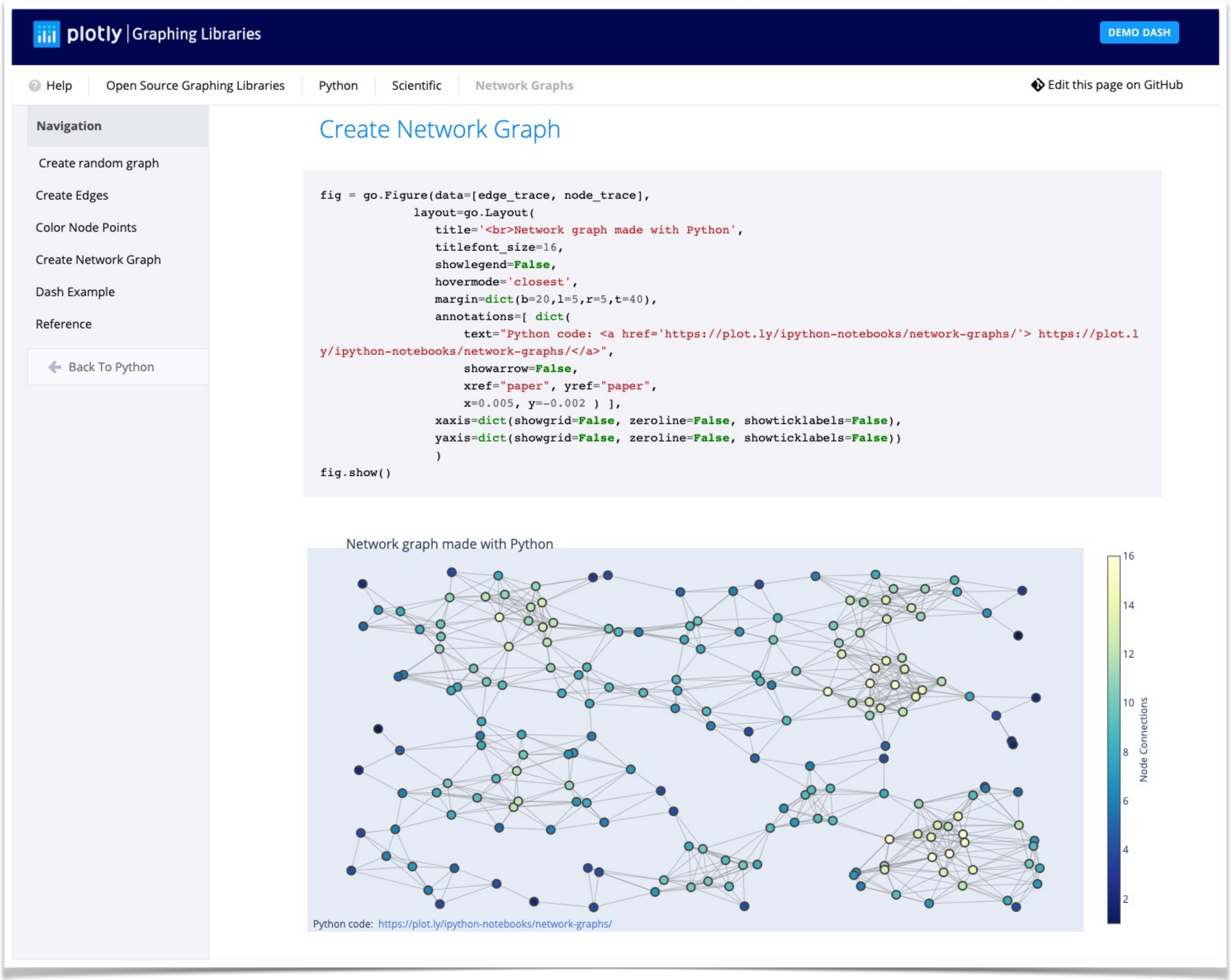


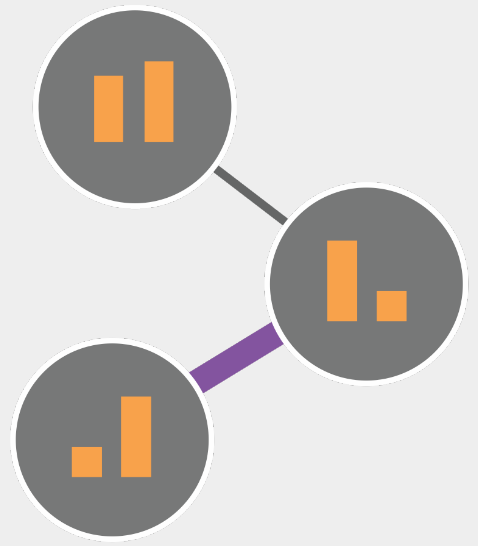


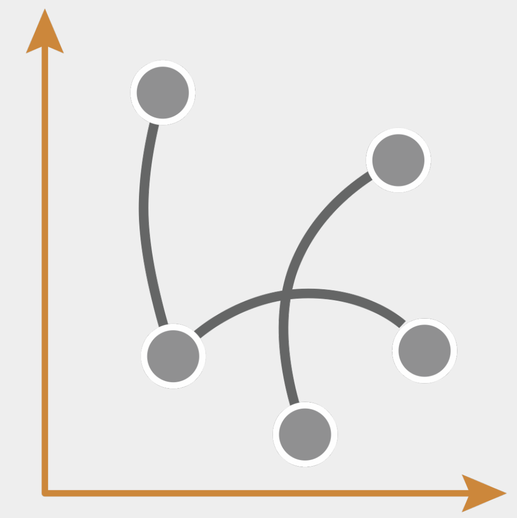


developer

python







developer

python

NetworkX

Stable (notes)

2.3 — April 2019
download | doc | pdf

Latest (notes)

2.4 development
github | doc | pdf

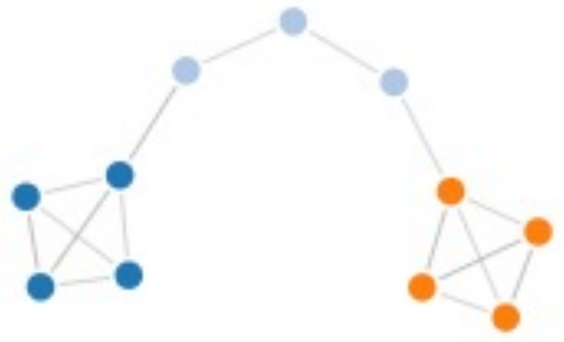
Archive

Contact

Mailing list
Issue tracker

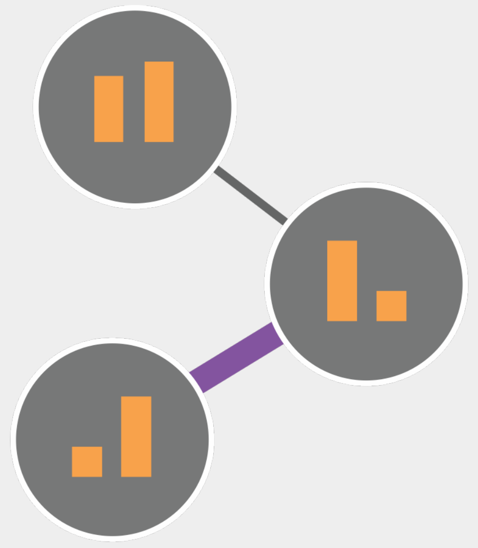
Software for complex networks

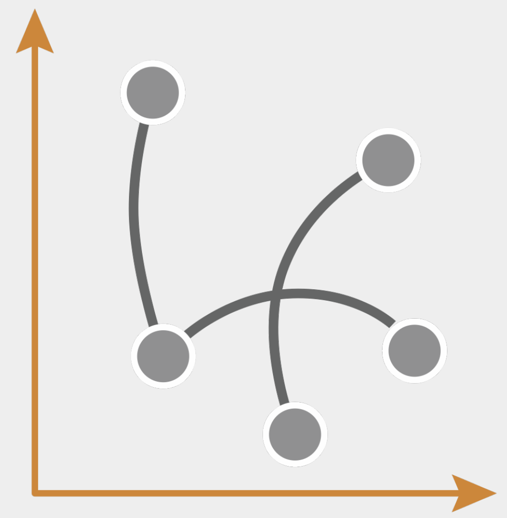
NetworkX is a Python package for the creation, manipulation, and study of the structure, dynamics, and functions of complex networks.



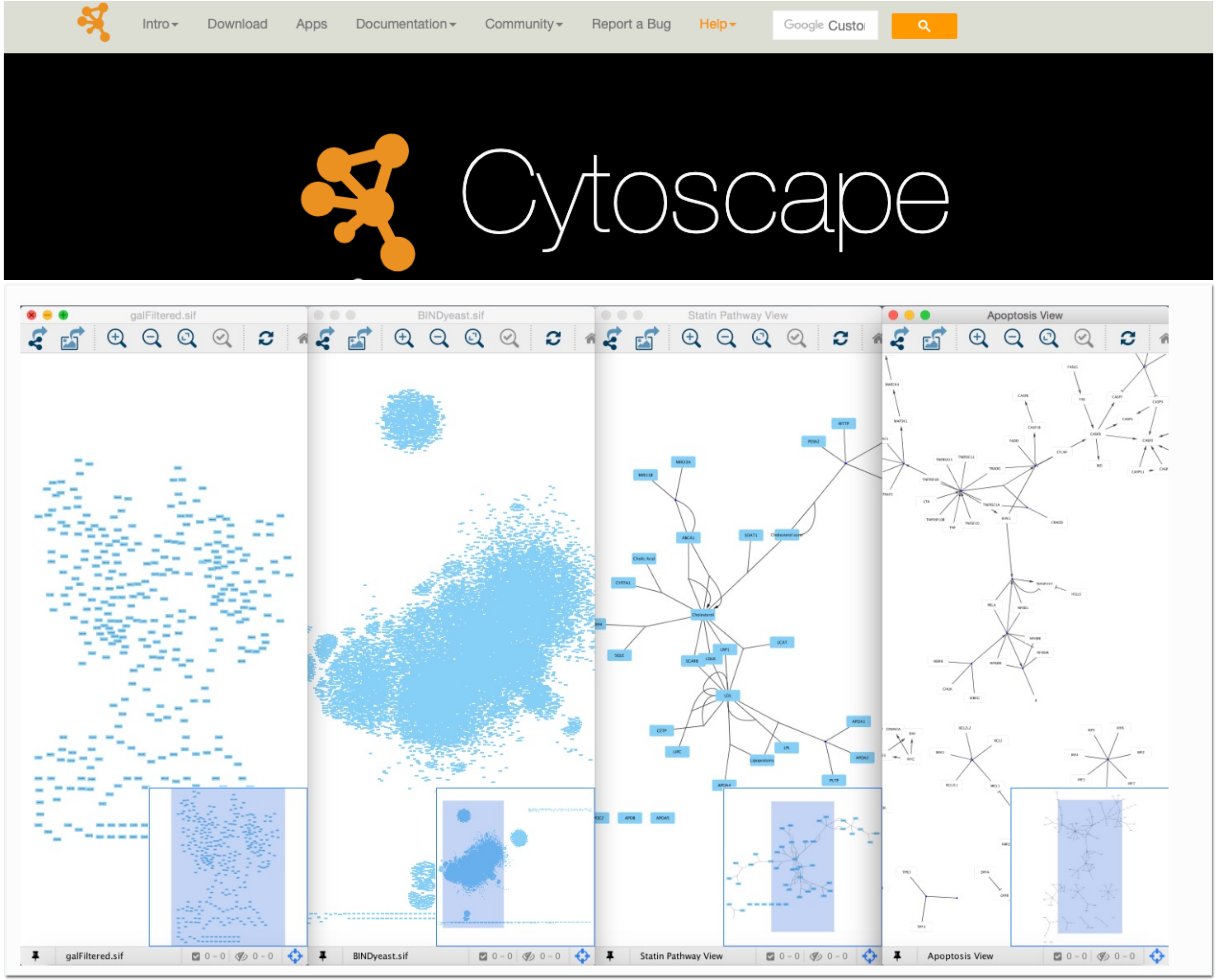
Features

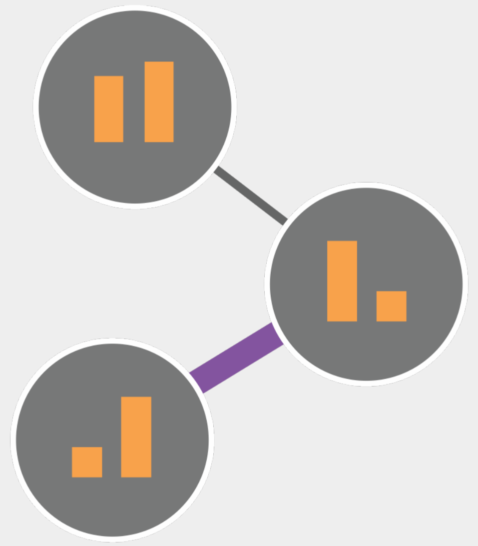
- Data structures for graphs, digraphs, and multigraphs
- Many standard graph algorithms
- Network structure and analysis measures
- Generators for classic graphs, random graphs, and synthetic networks
- Nodes can be "anything" (e.g., text, images, XML records)
- Edges can hold arbitrary data (e.g., weights, time-series)
- Open source 3-clause BSD license
- Well tested with over 90% code coverage
- Additional benefits from Python include fast prototyping, easy to teach, and multi-platform

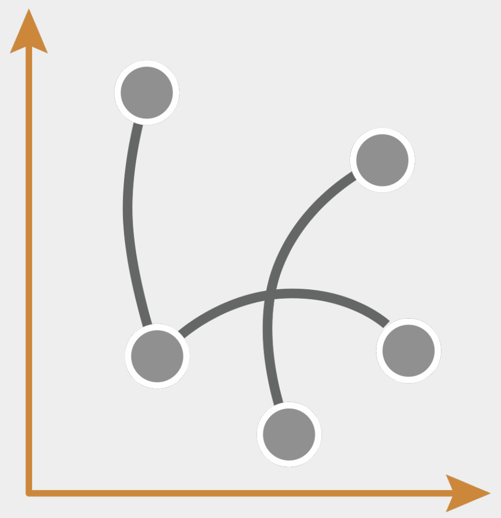




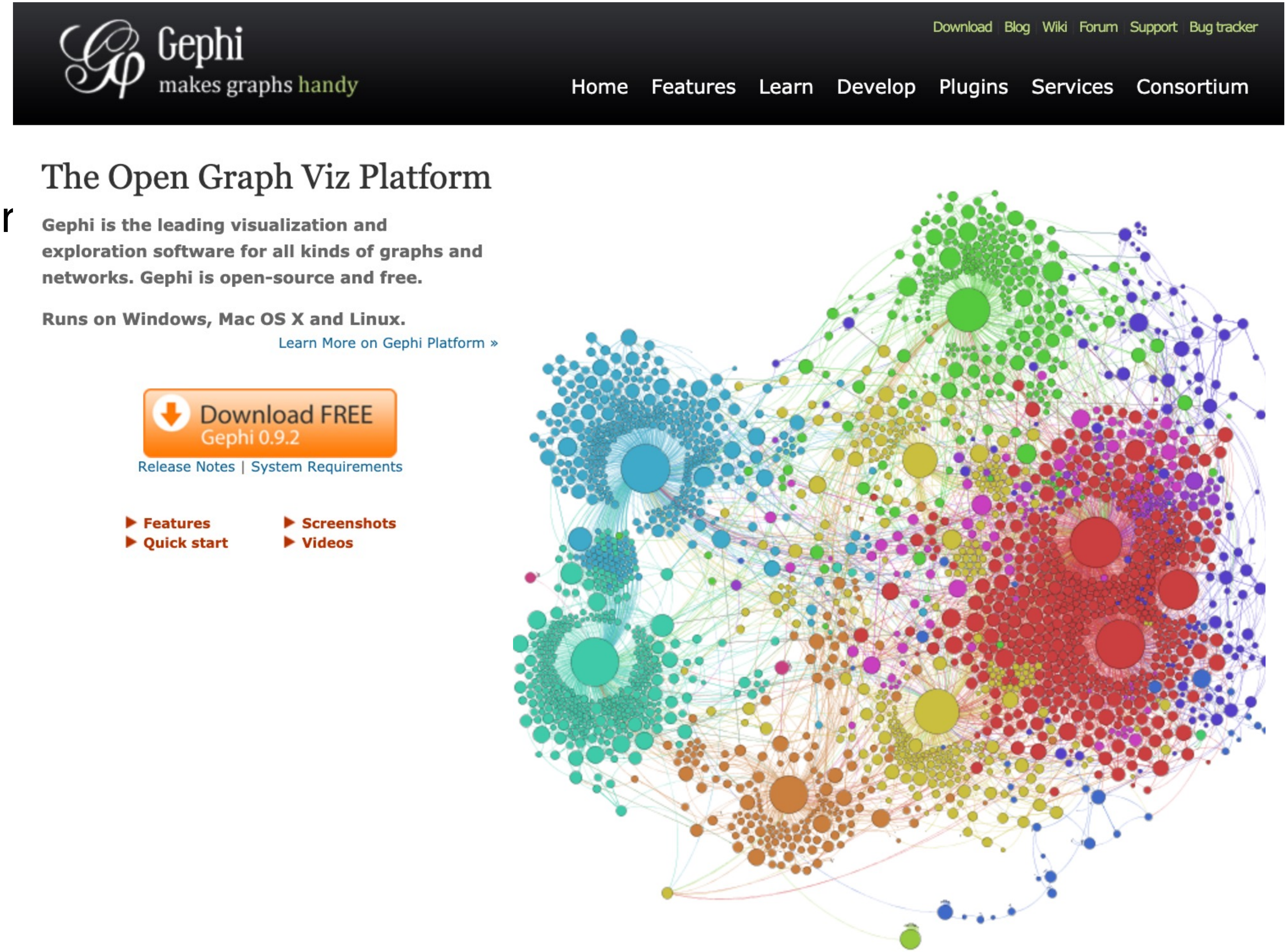
graphic
designer

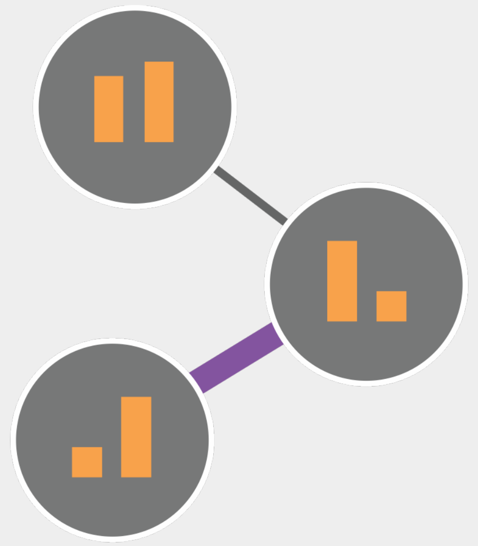


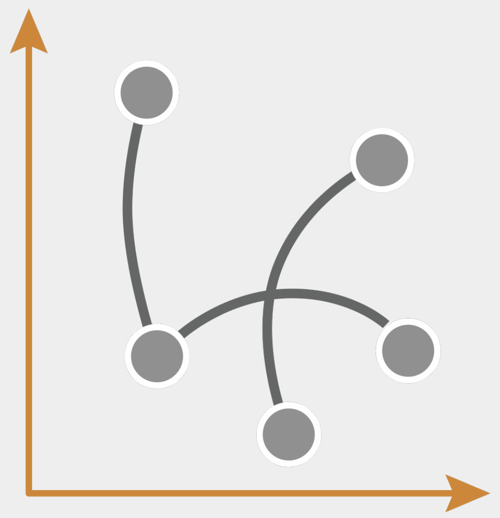




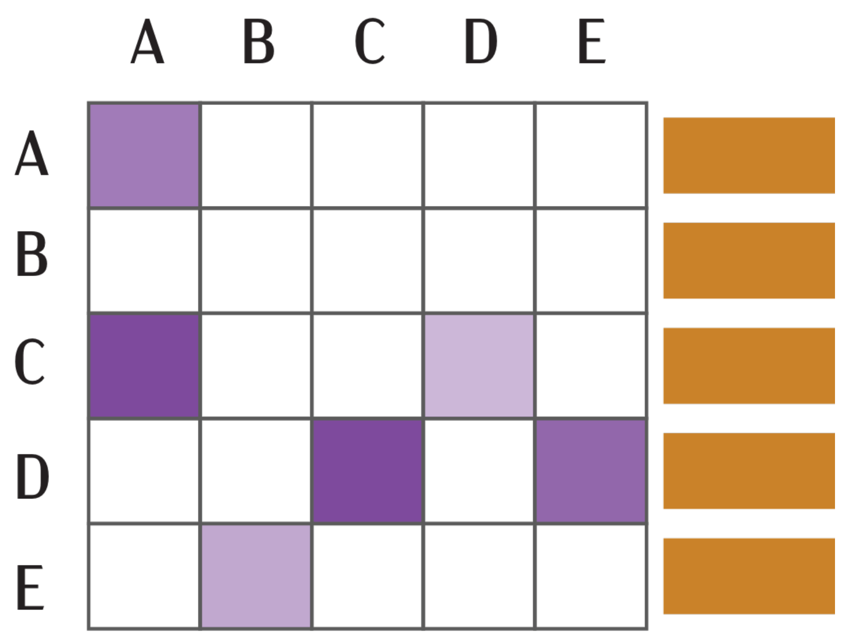
graphic designer



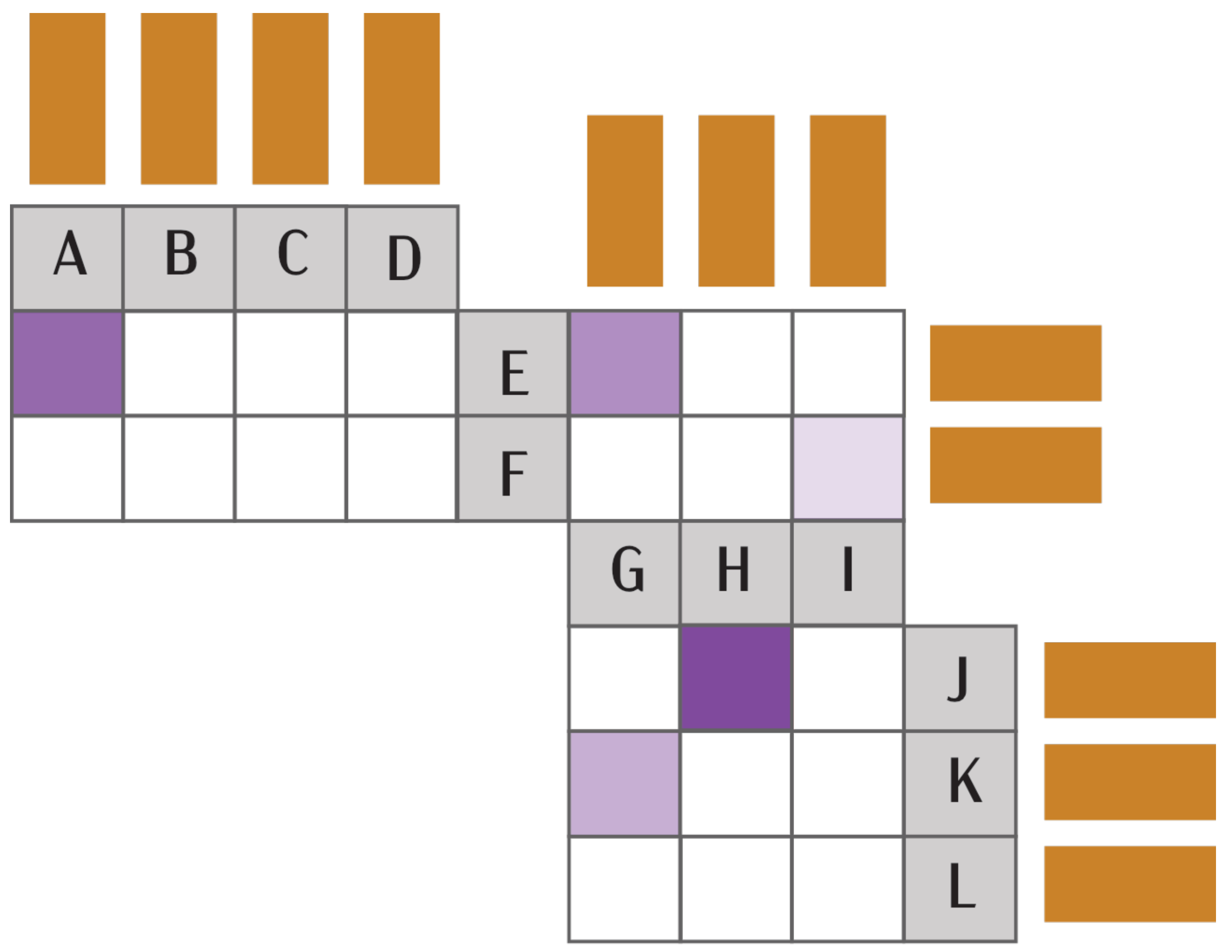




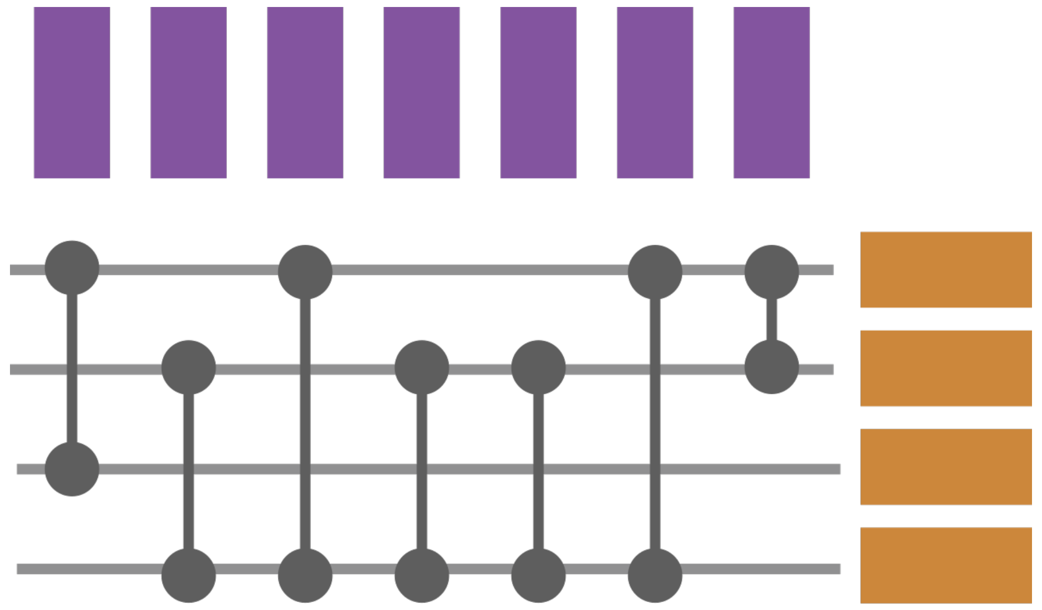
Tabular Layouts



Adjacency Matrix

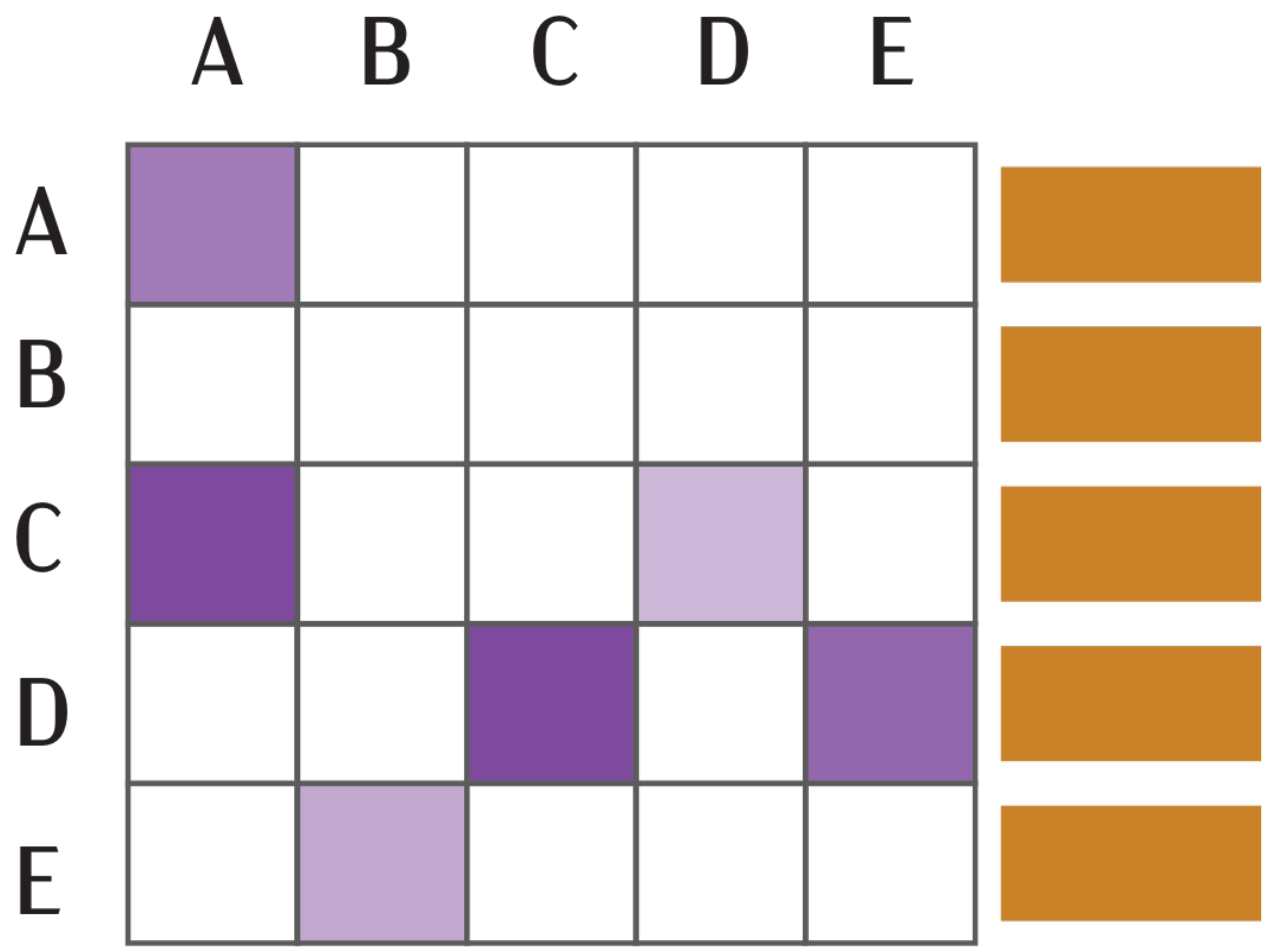


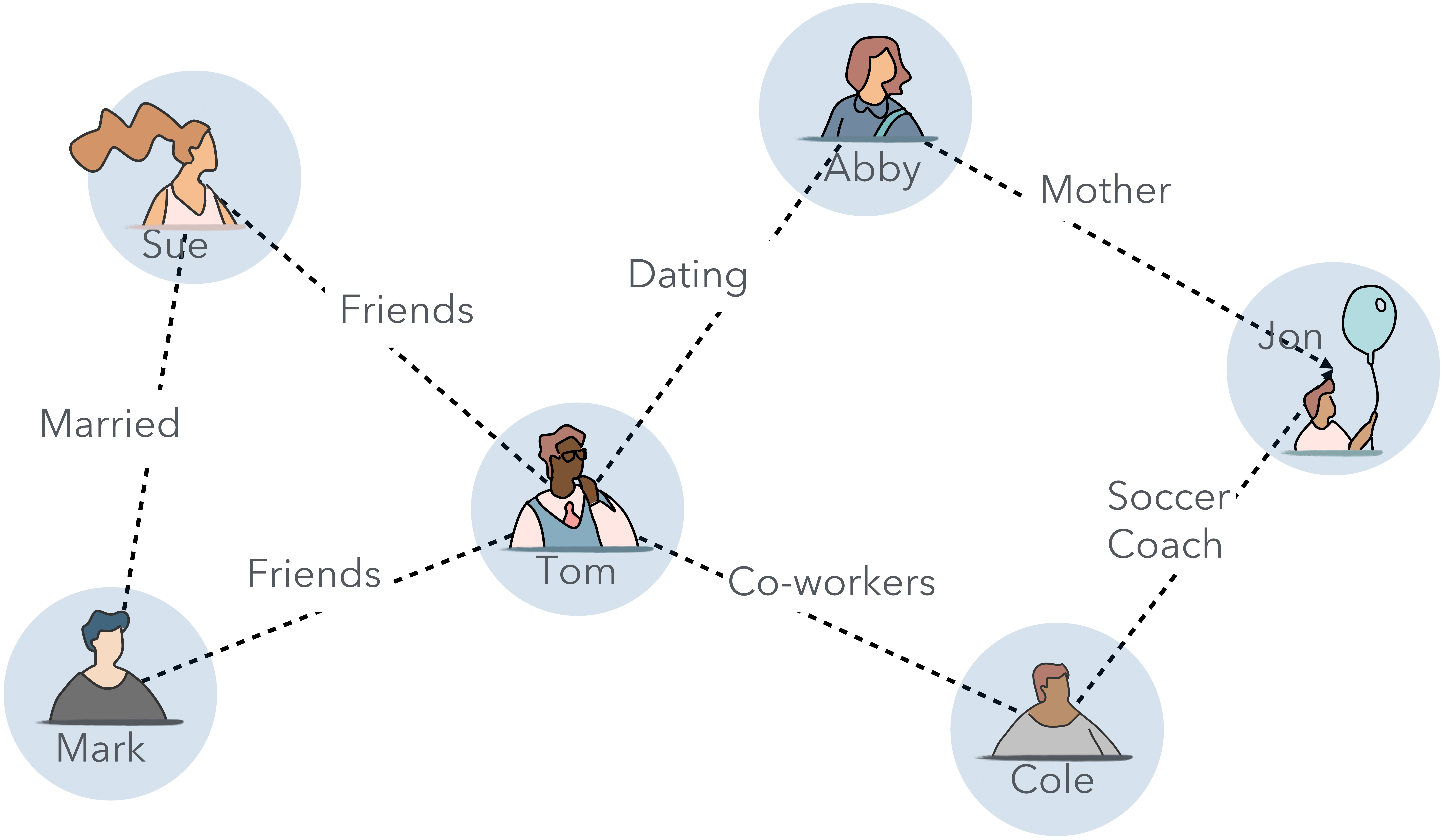
Quilts



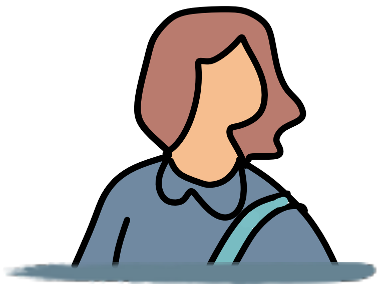
BioFabric

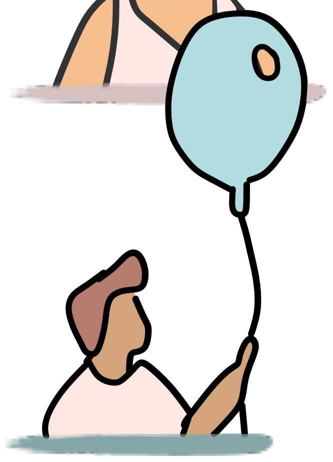
Adjacency Matrix

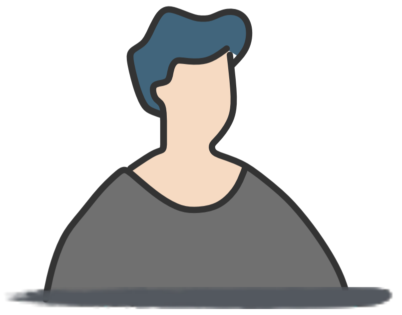
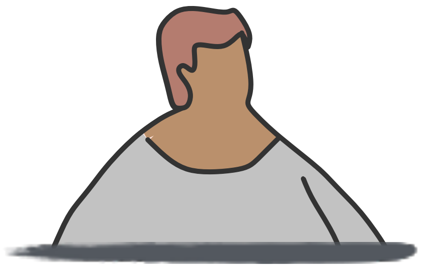


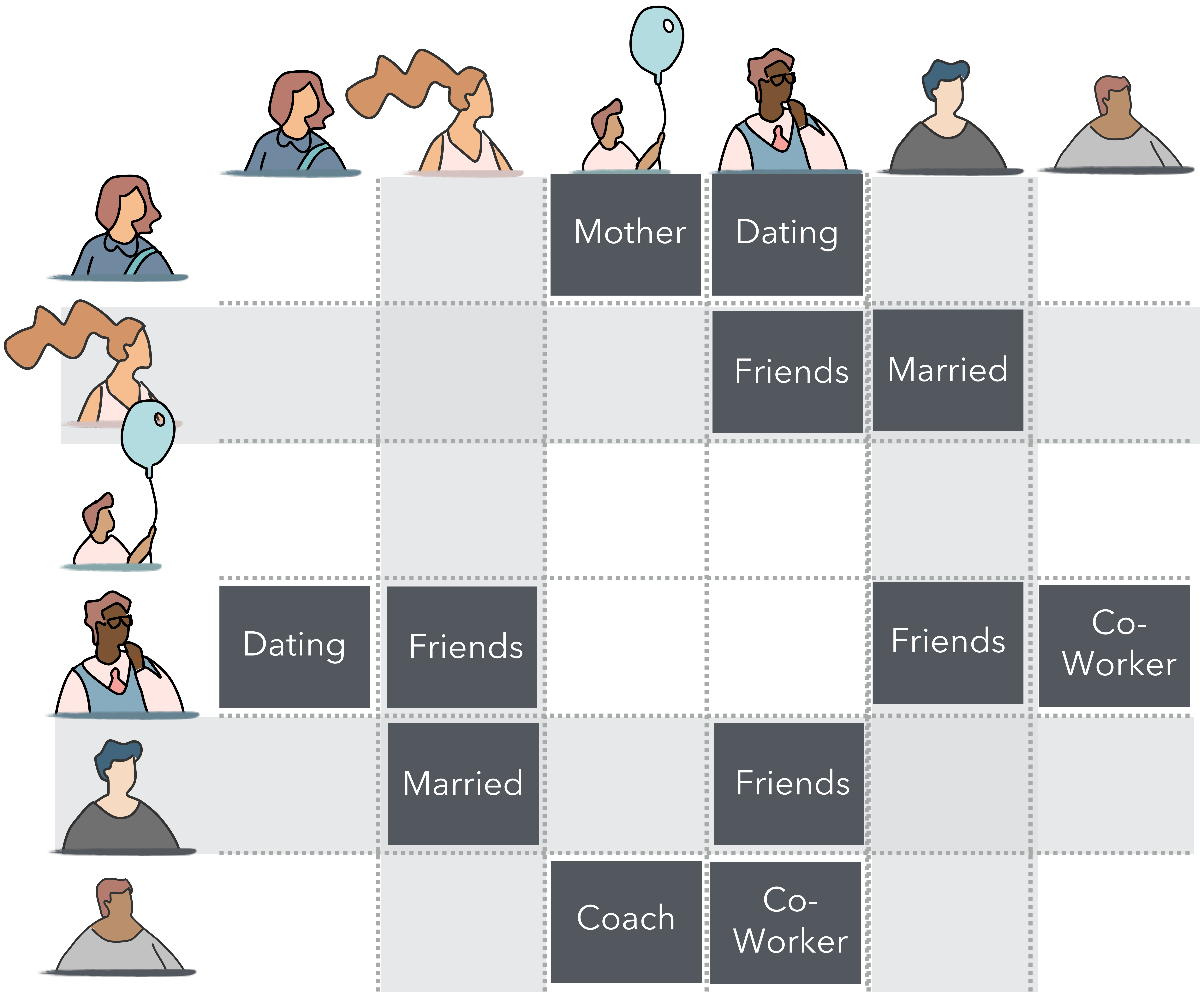


Sue
Abby
Mother
Friends
Dating
Jon
Married
Tom
Soccer Coach
Friends
Co-workers
Mark
Cole

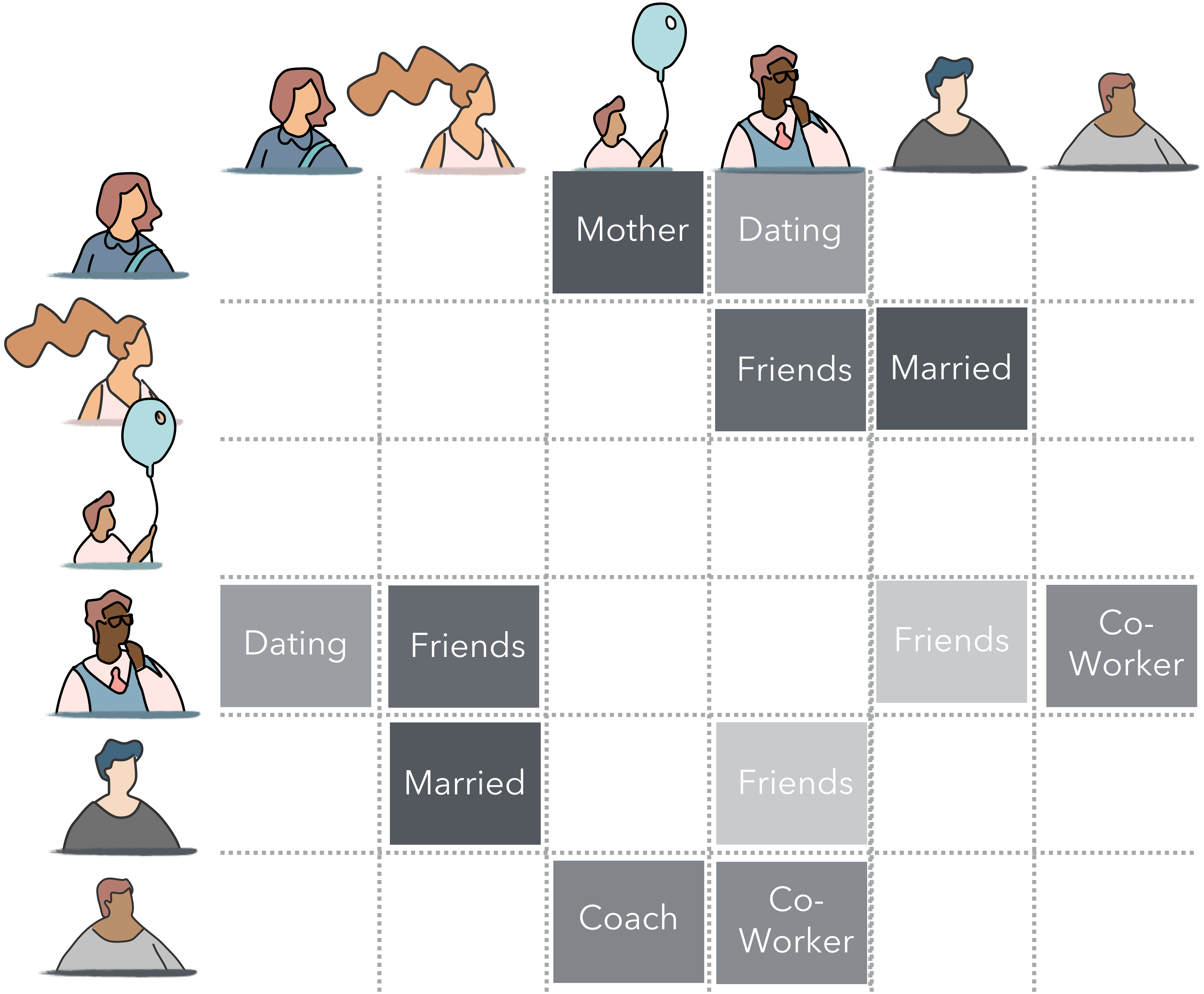


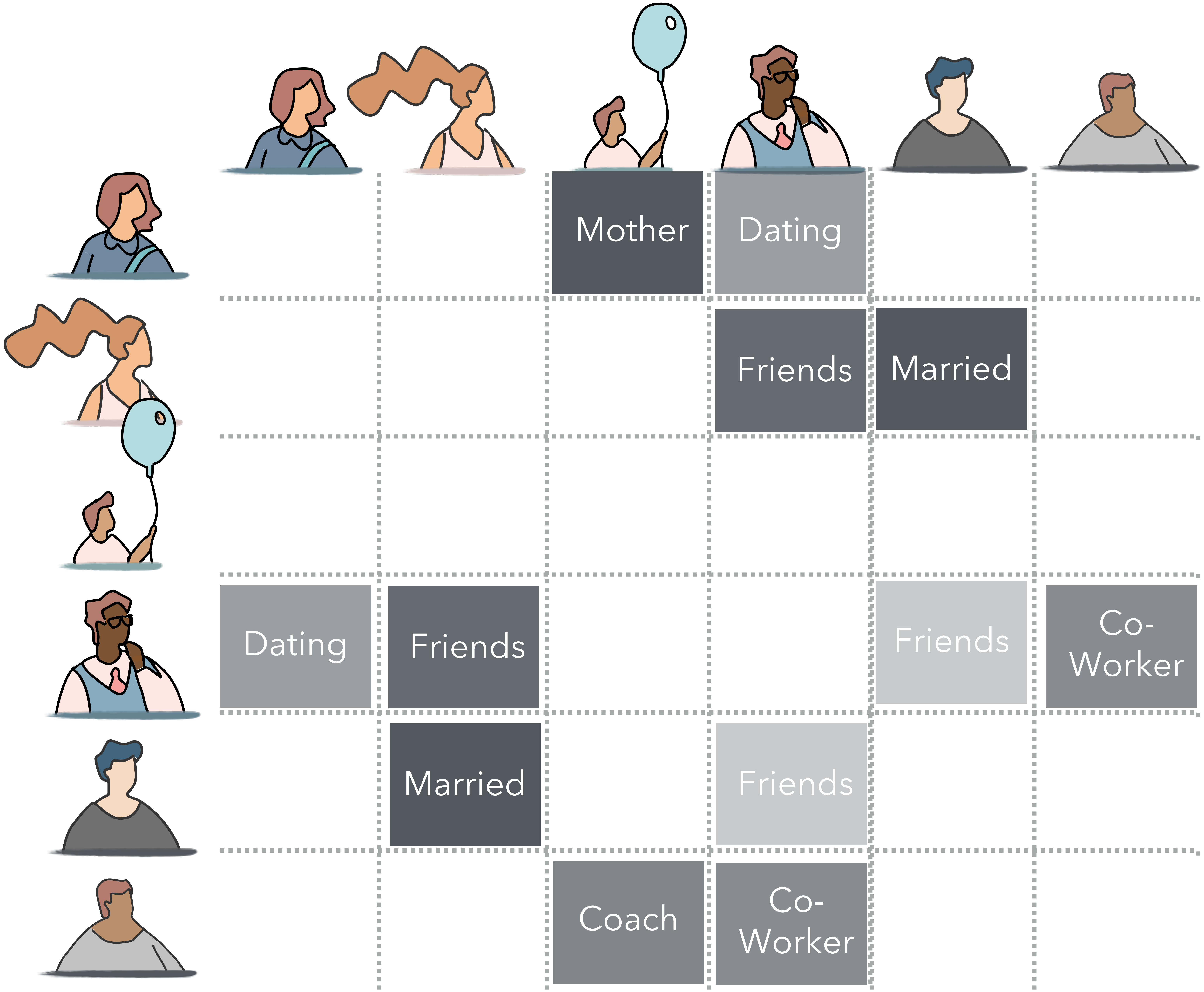


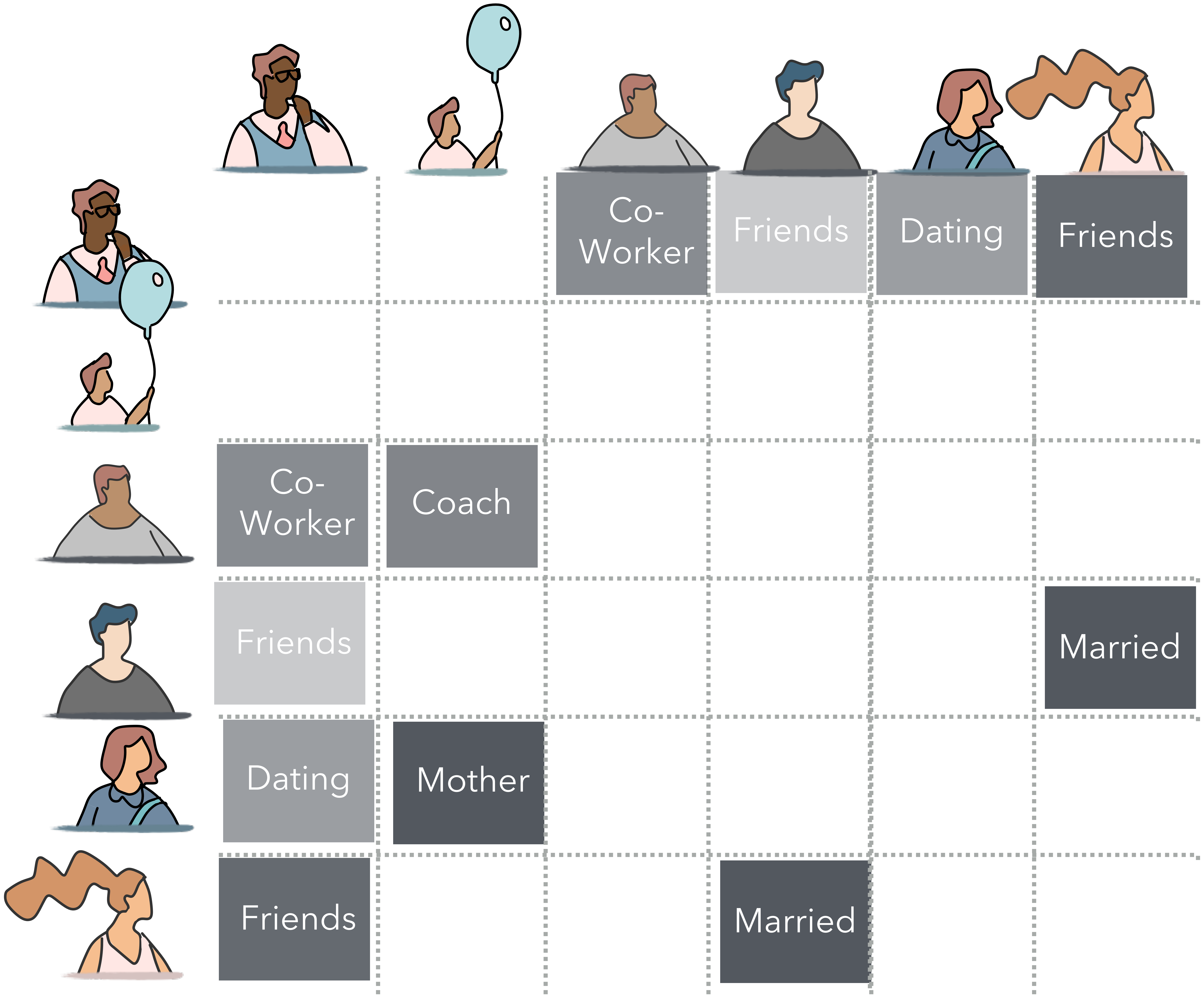


Mother
Dating
Friends
Married
Dating
Friends
Friends
Co-
Worker
Married
Friends
Coach
Co-
Worker

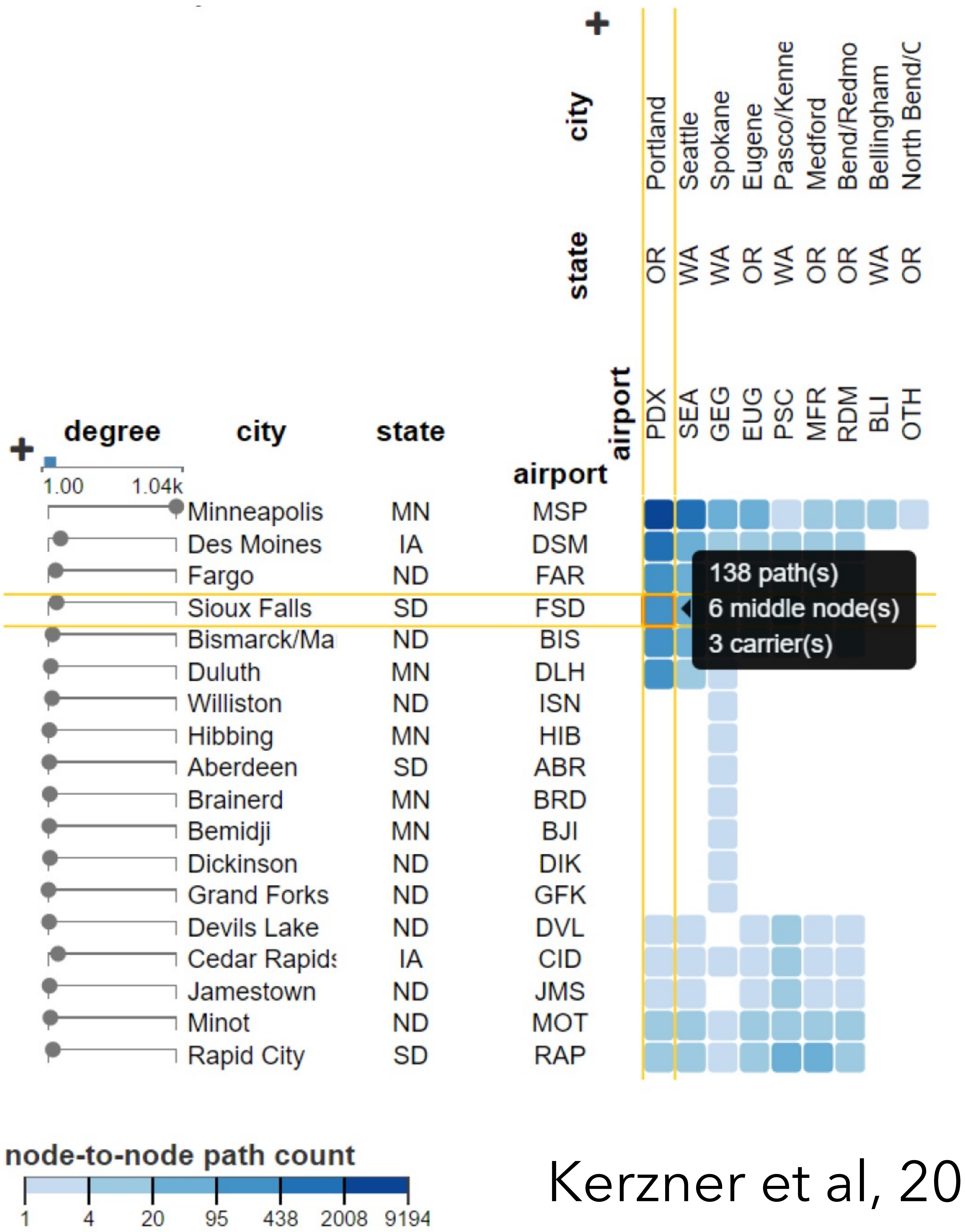

Mother
Dating
Friends
Married
Dating
Friends
Friends
Co-Worker
Married
Friends
Coach
Co-Worker



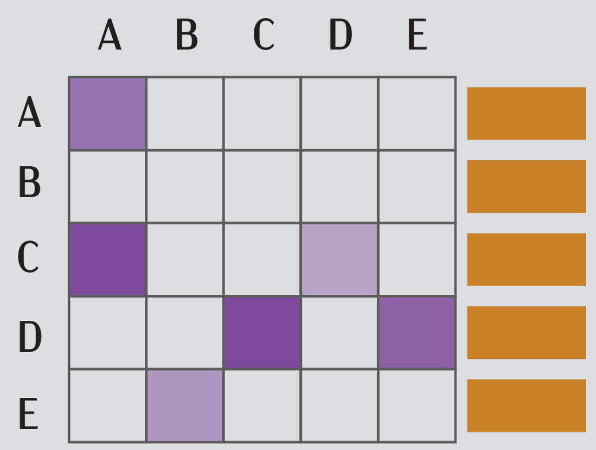
Name	**Beverage**	**Day 1**
Abby	Port	1
Sue	Coke	0
Jon	Coke	4
Tom	Beer	5
Mark	Beer	2
Cole	Port	3



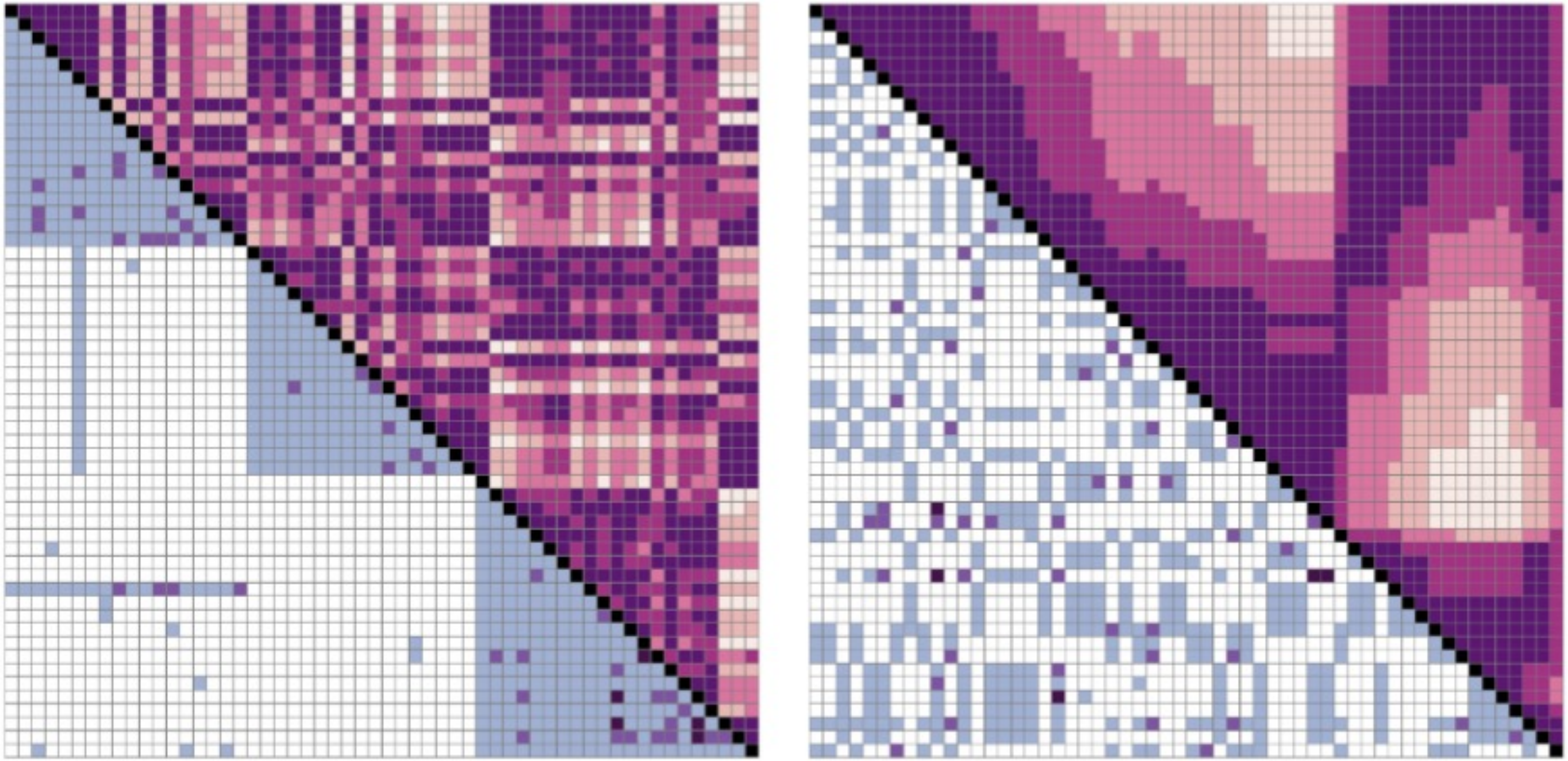
Name	Beverage	Day 1
Tom	Beer	5
Jon	Coke	4
Cole	Port	3
Mark	Beer	2
Abby	Port	1
Sue	Coke	0



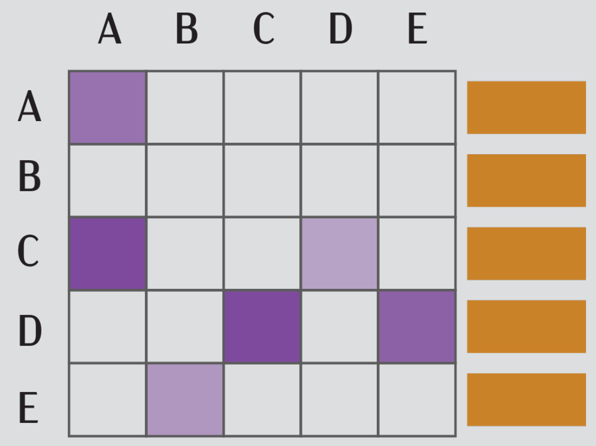
Kerzner et al, 2017



Adjacency Matrix



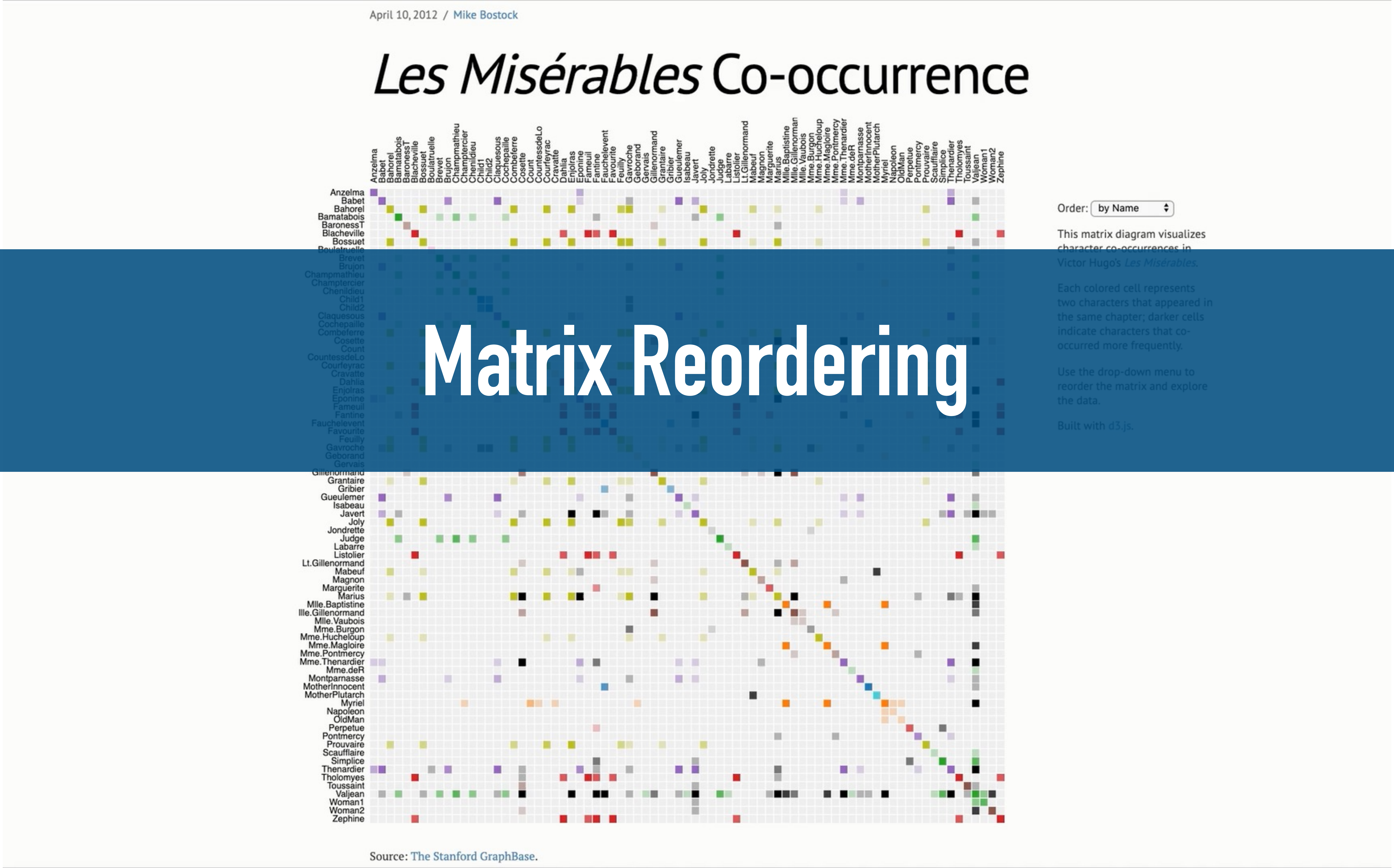
(a) Sorted by structure. (b) Sorted by attribute similarity.



Adjacency Matrix

Berger et al, 2019

Matrix Reordering



Reorder.js

Code | Issues 2 | Pull requests 1 | Projects 0 | Wiki | Security | Insights

Home

Edit | New Page

Jean-Daniel Fekete edited this page on Apr 23, 2015 · 2 revisions

Reorder.js is a library to reorder tables and graph/networks.

Resources

- Introduction
- API Reference

Browser / Platform Support

Reorder.js is mainly developed on Chrome and Node.js. Use `npm install reorder.js` to install, and `require("reorder")` to load.

Installing

Download the latest version here:

- https://github.com/jdfekete/reorder.js/releases

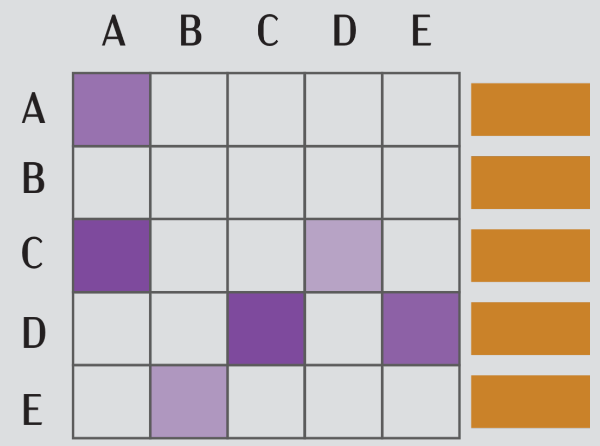
+ Add a custom footer

Pages 12

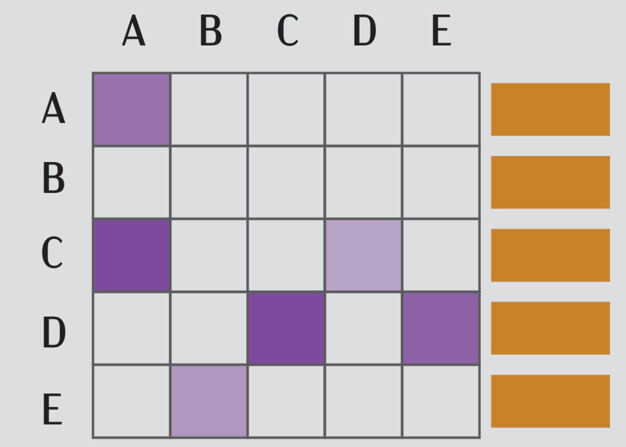
Find a Page...

- Home
- API Reference
- Conversion
- Core
- Gallery
- Graph
- Introduction
- LinearAlgebra
- Matrix
- Measure
- Permutation
- Reordering

+ Add a custom sidebar



Adjacency
Matrix



Adjacency Matrix

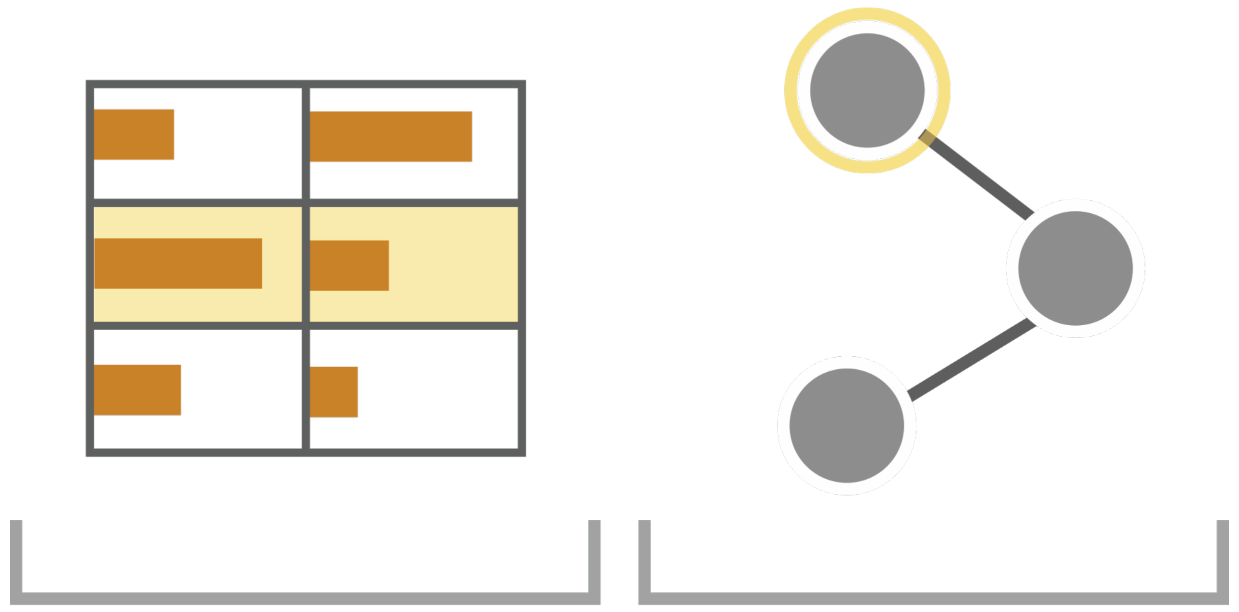
Ideal for dense and completely connected networks

Requires quadratic space with respect to the number of nodes.

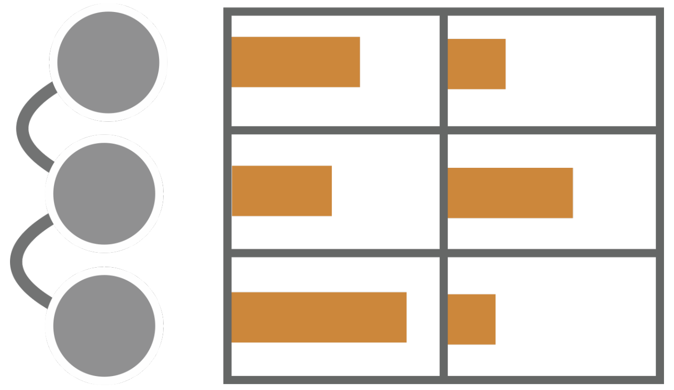
Complexity of choosing the right reordering algorithm

Recommended for smaller, complex and dense networks with rich node and/or edge attributes, for all tasks except for those involving paths

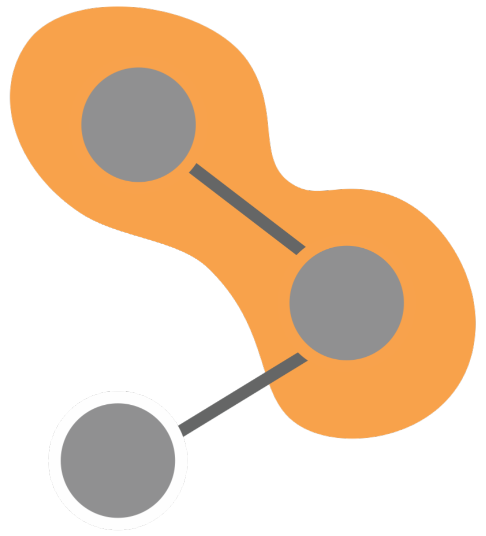
View Operations



Juxtaposed

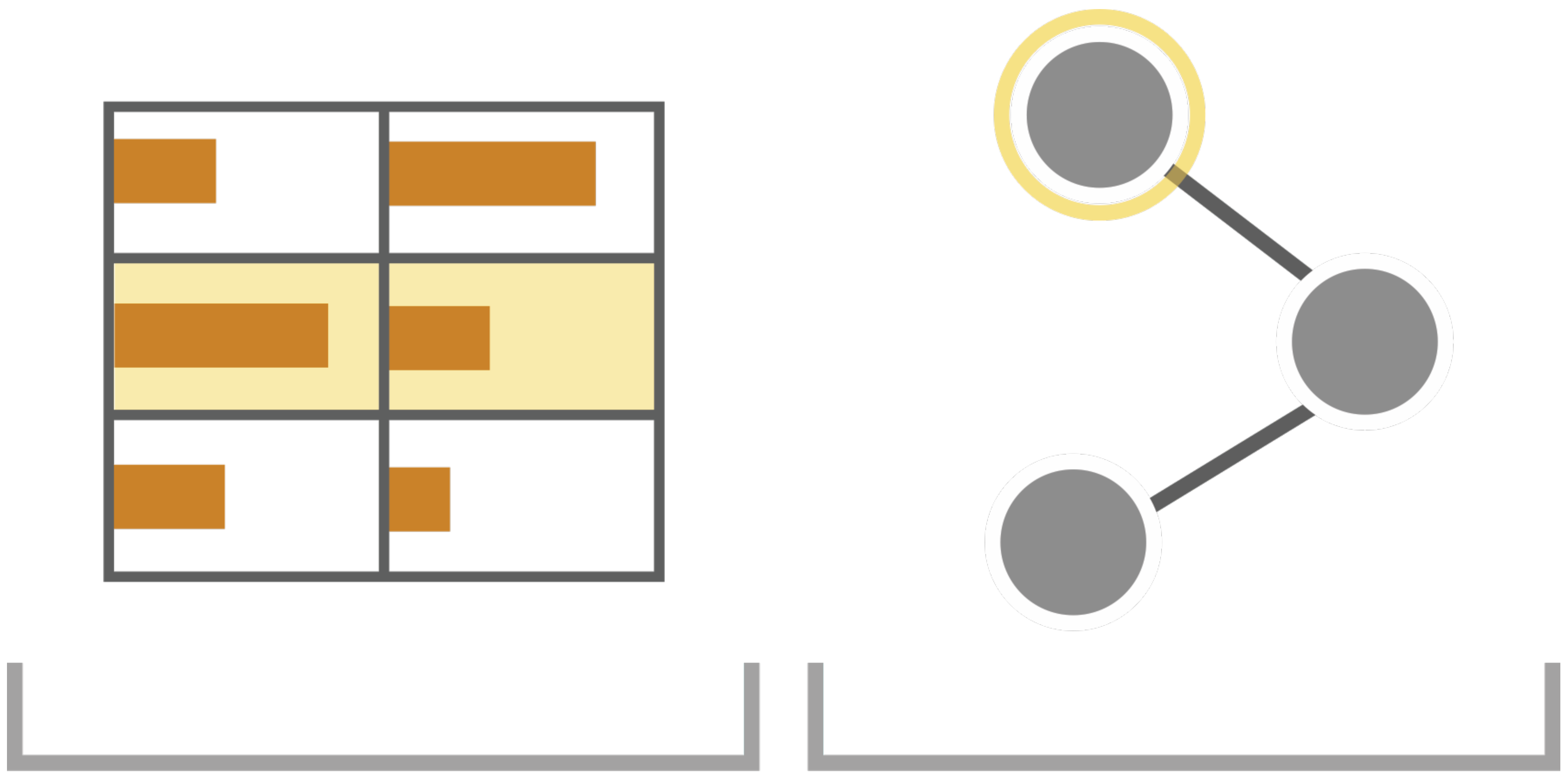


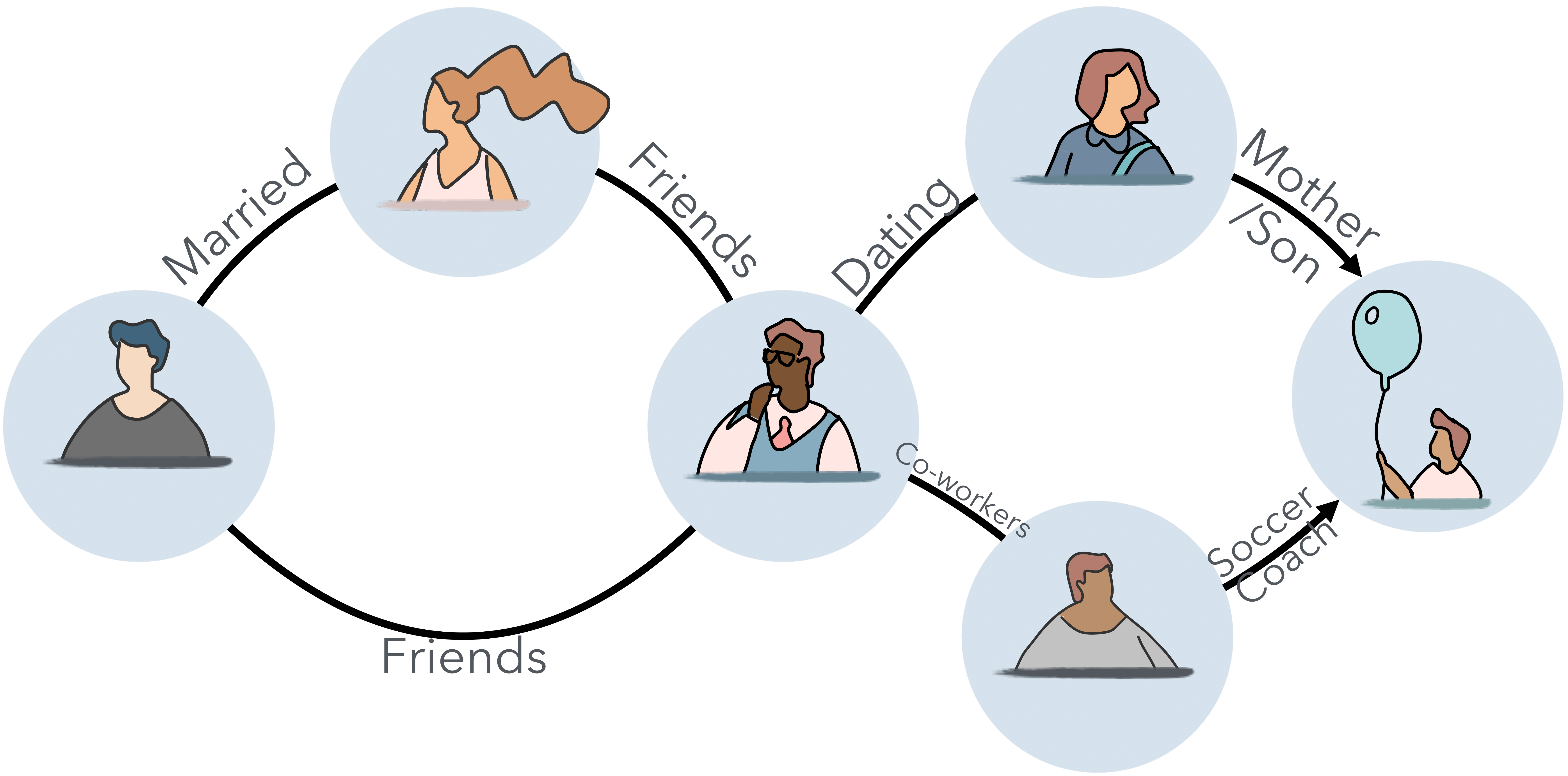
Integrated



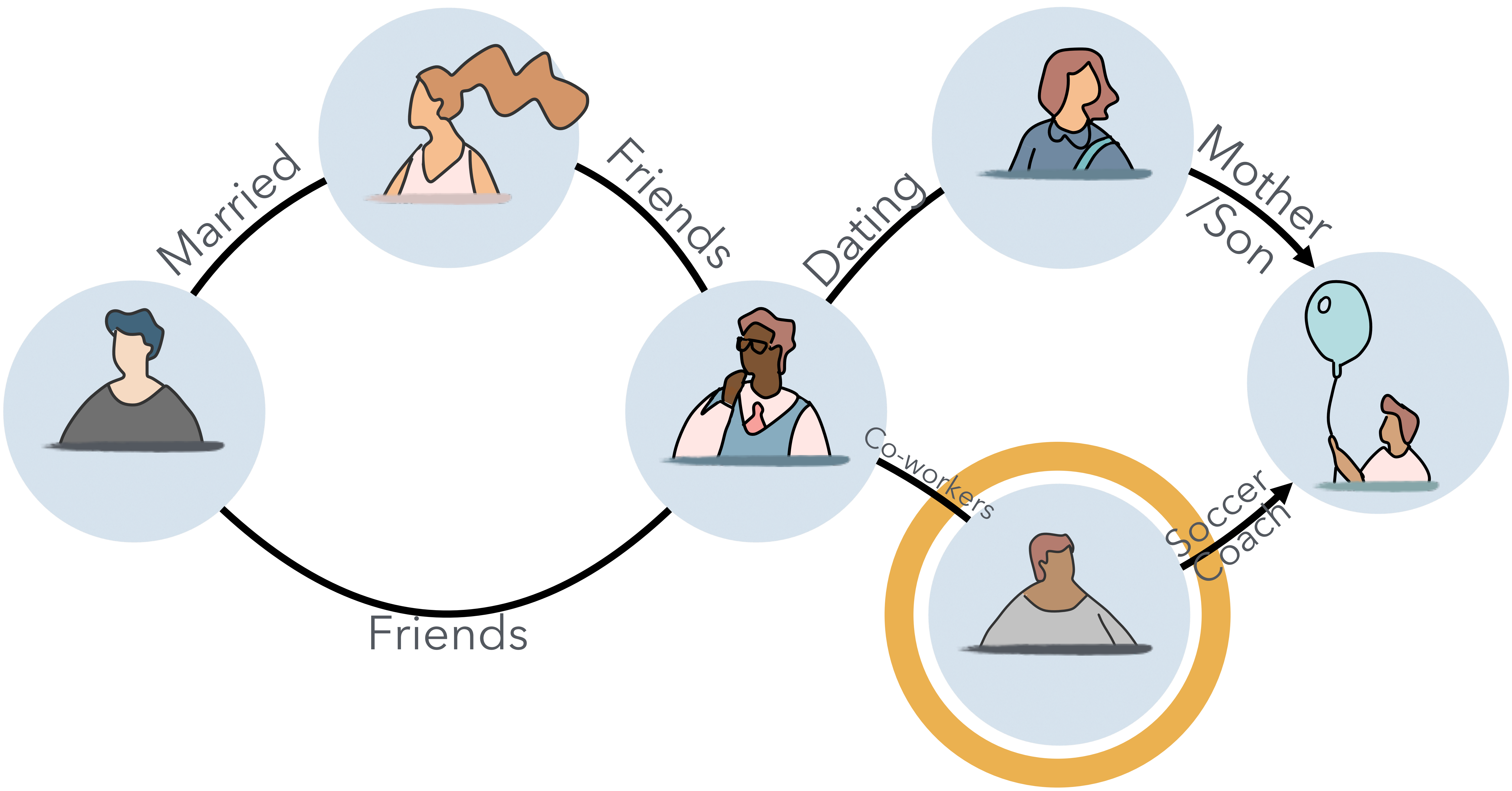
Overloaded

Juxtaposed

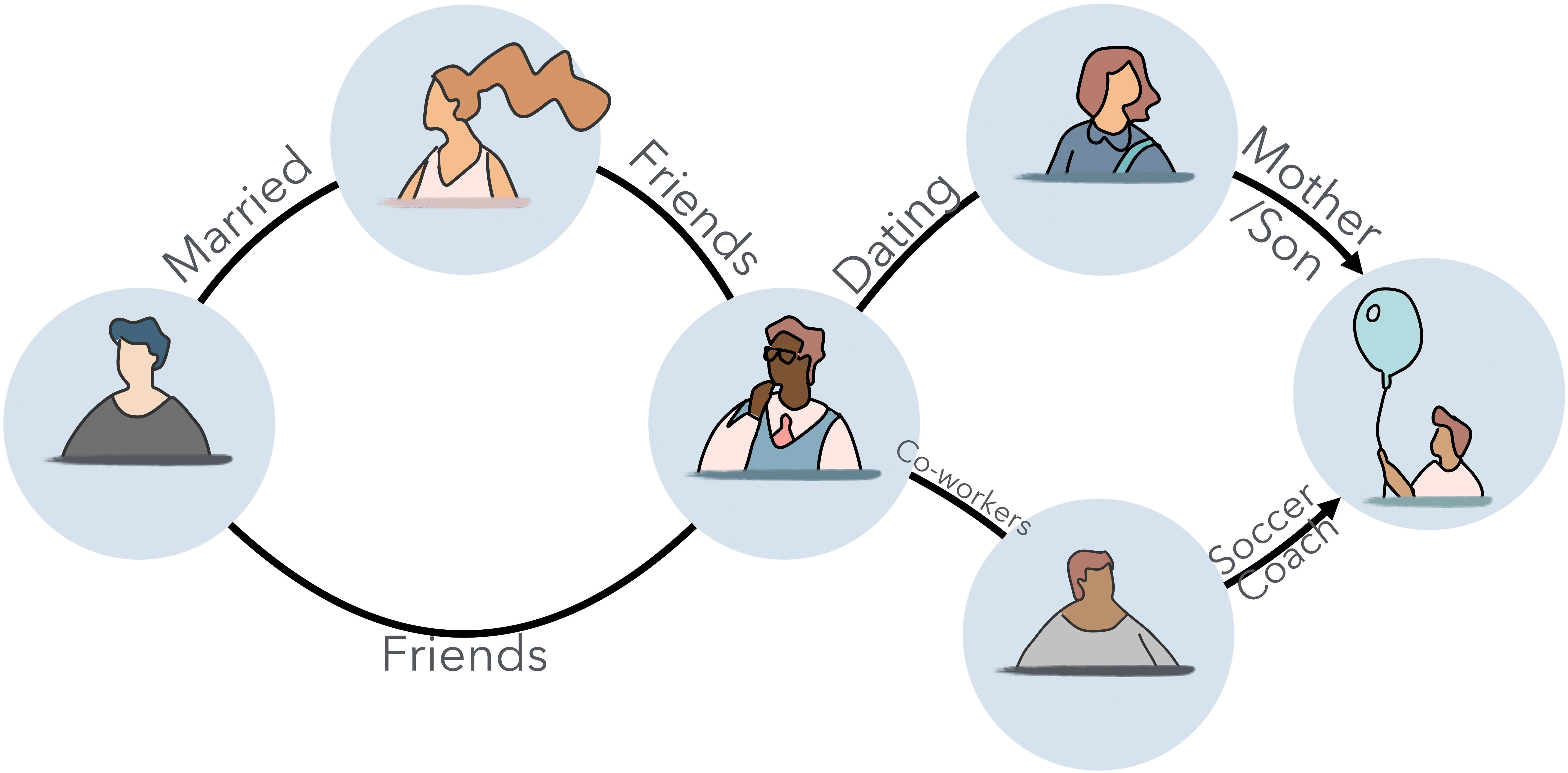




Married
Friends
Dating
Mother
/Son
Co-workers
Soccer
Coach
Friends

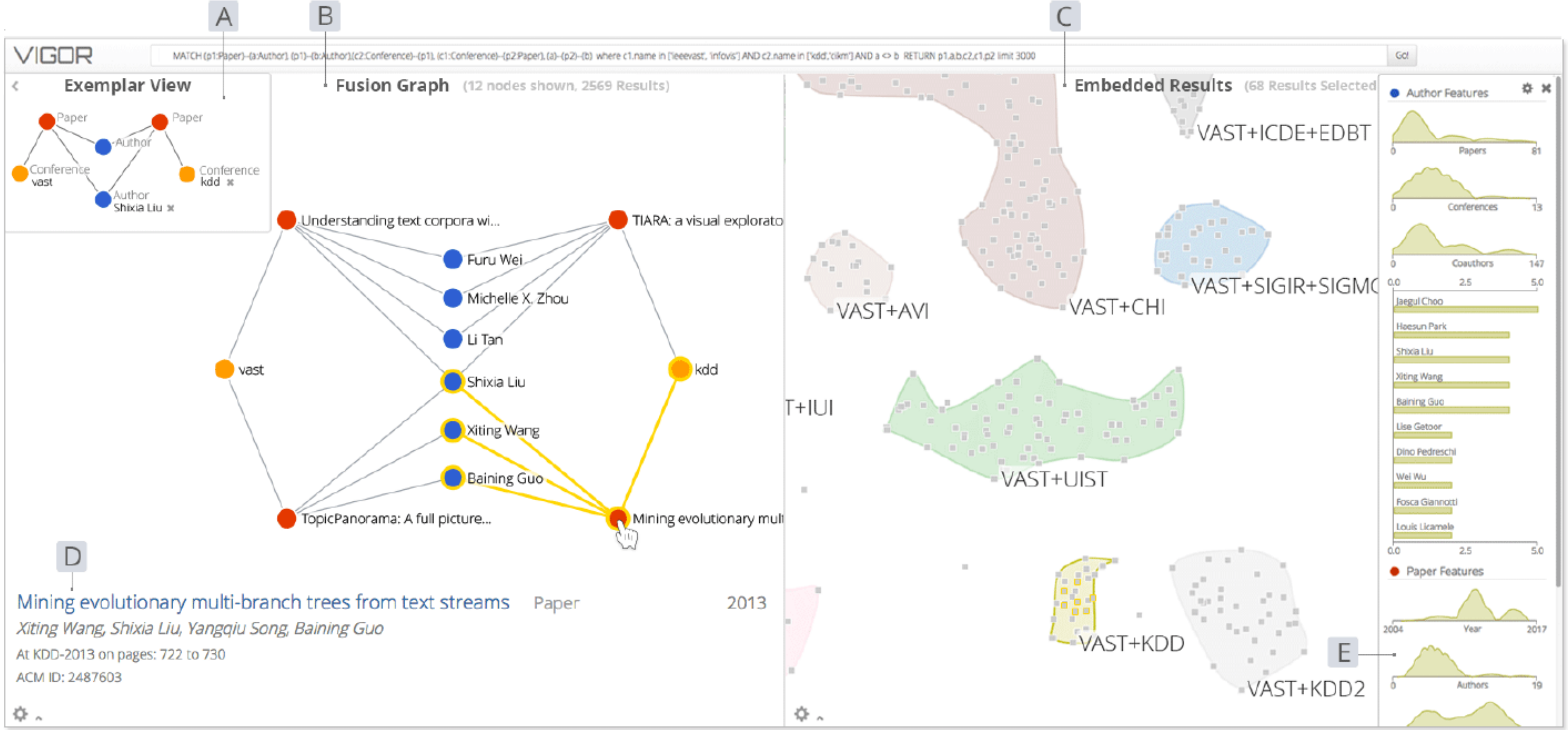


Name	Beverage	Day 1
Mark	Beer	1
Sue	Coke	0
Cole	Port	4
Jon	Coke	5
Tom	Beer	2
Abby	Port	3

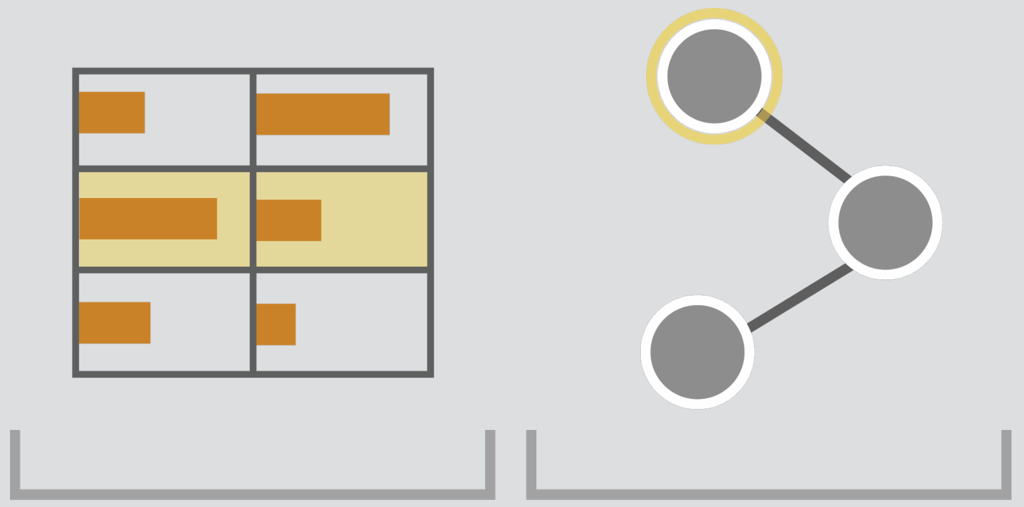


Name	Beverage	Day 1
Mark	Beer	1
Sue	Coke	0
Cole	Port	4
Jon	Coke	5
Tom	Beer	2
Abby	Port	3

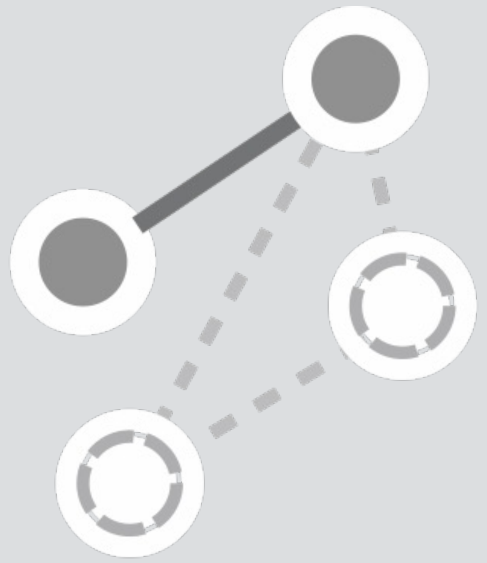
Relationship	Years
Dating	4
Mother / Son	12
Co-workers	3
Soccer Coach	2
Friends	8
Friends	3
Married	4



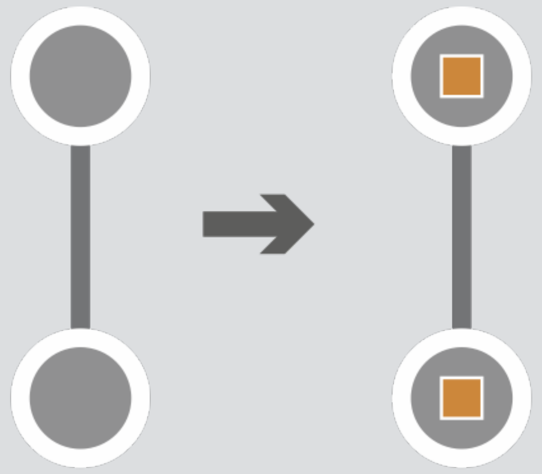
VIGOR *Pienta et al. 2018*



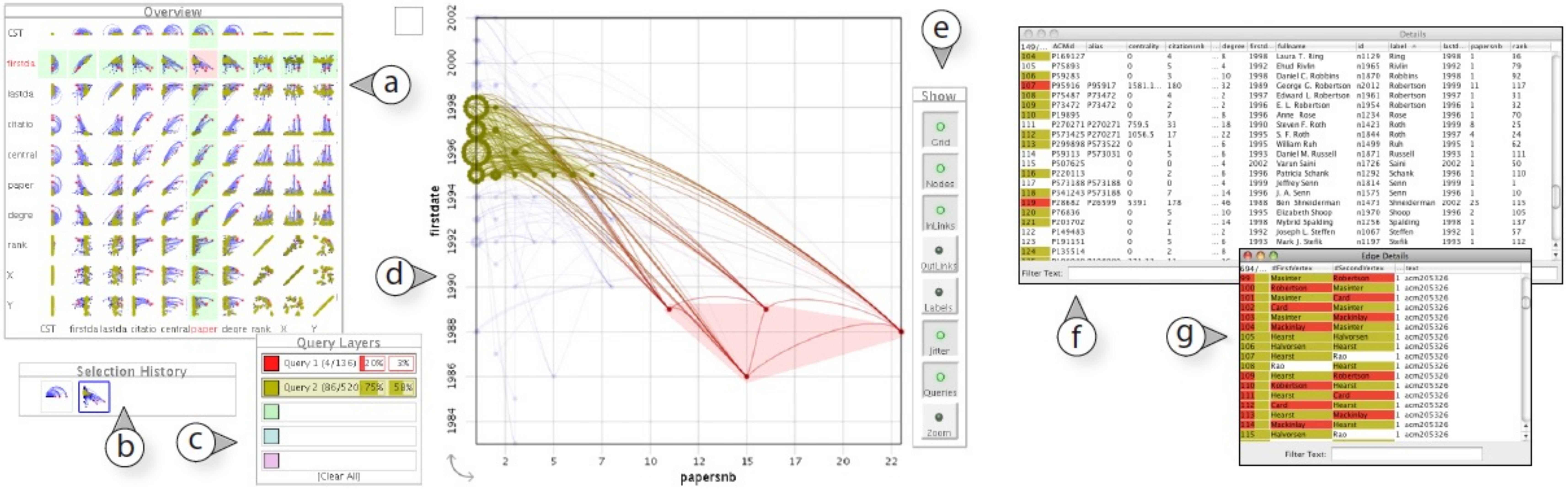
Juxtaposed

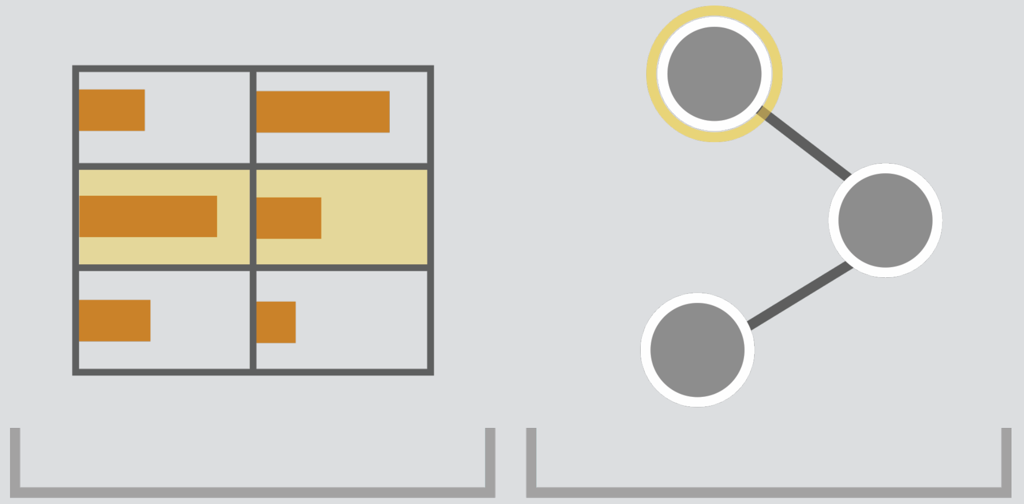


Querying and Filtering



Deriving New Attributes

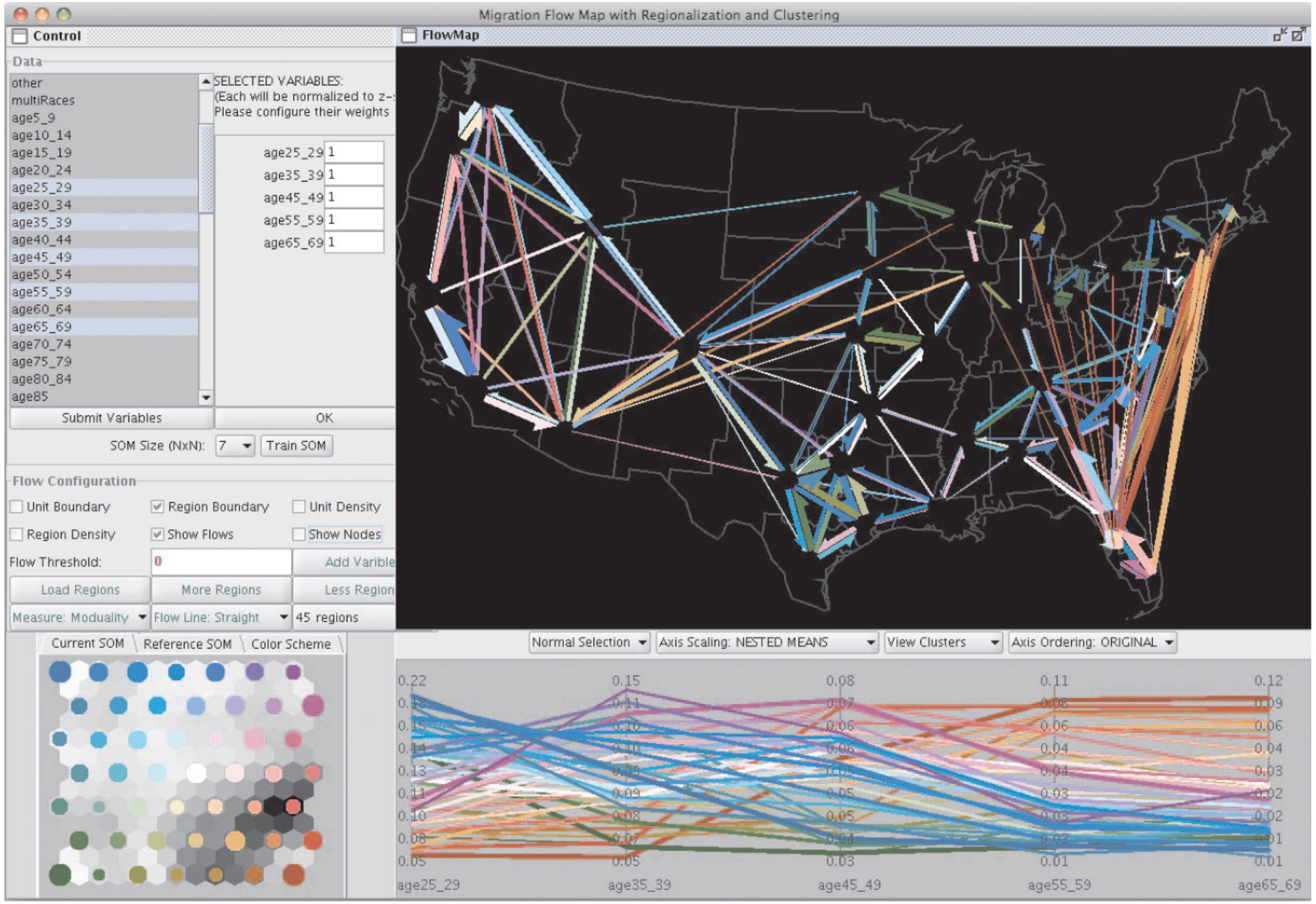


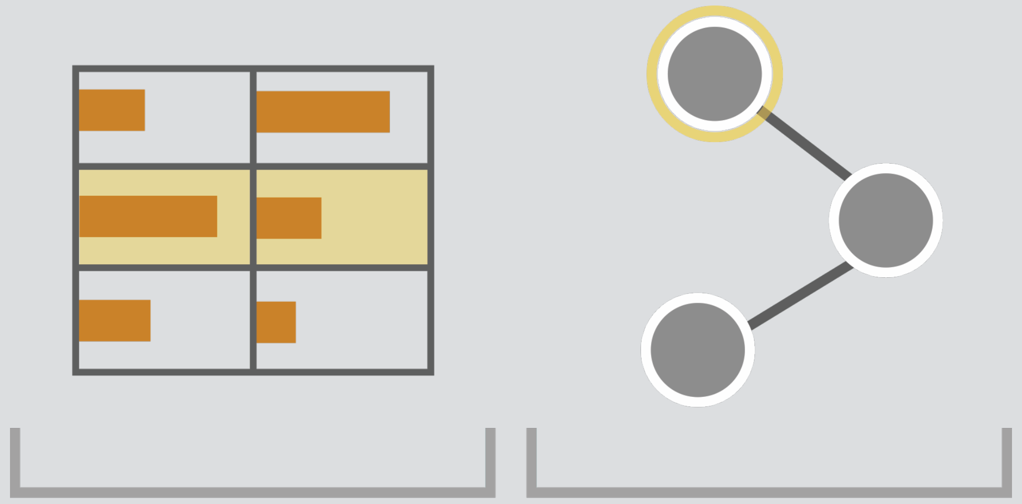


Juxtaposed

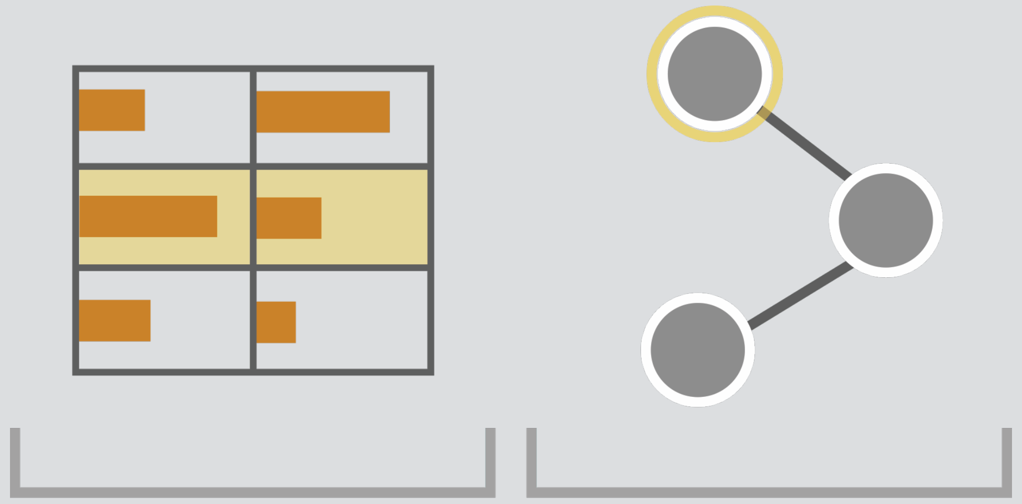
Graph Dice *Bezerianos et al. 2010*

Guo, 2009

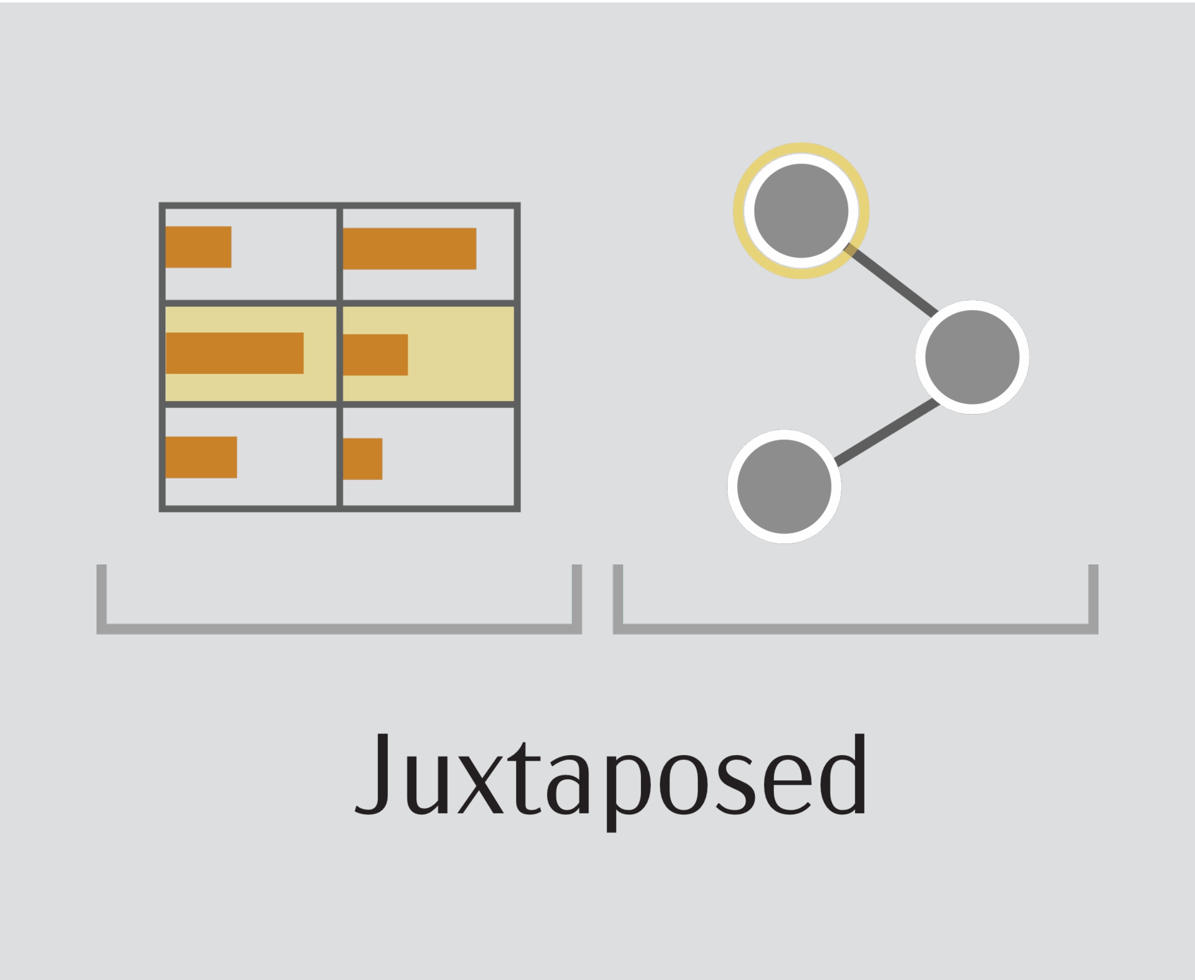




Juxtaposed



Juxtaposed

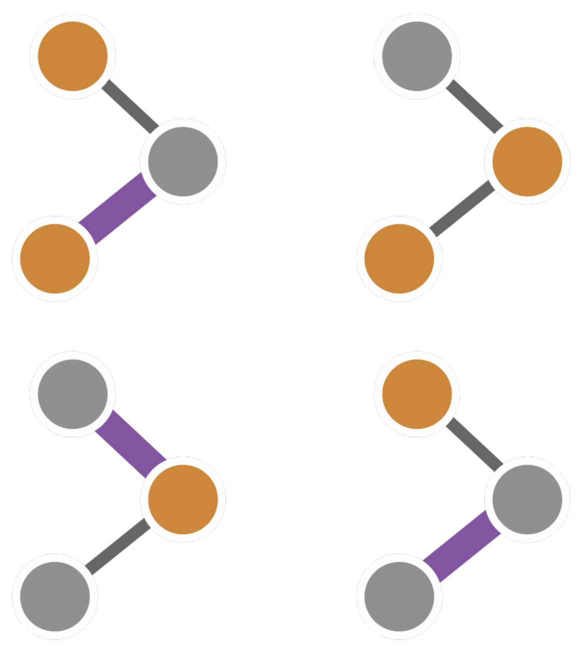


Independent views can optimize for topology and attribute independently.

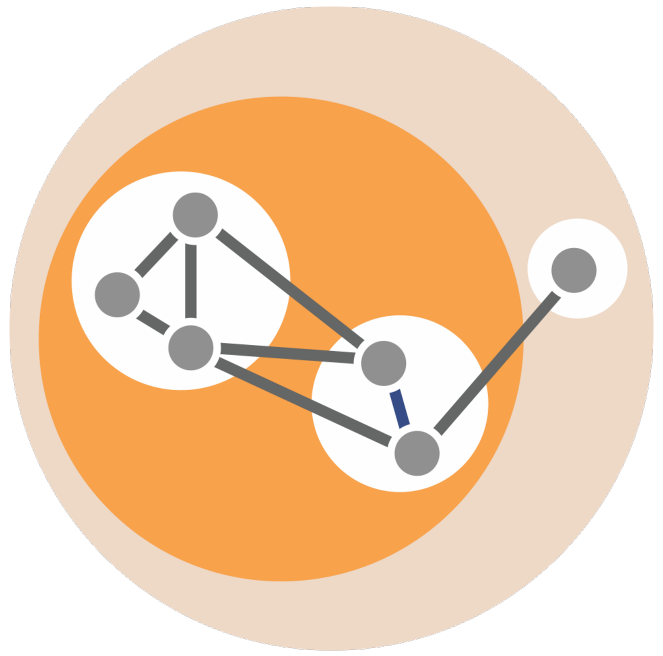
Not great for tasks on topological structures beyond a single node or edge.

Recommended for large networks and/or very large numbers or heterogeneous types of node and link attributes

Layout Operations

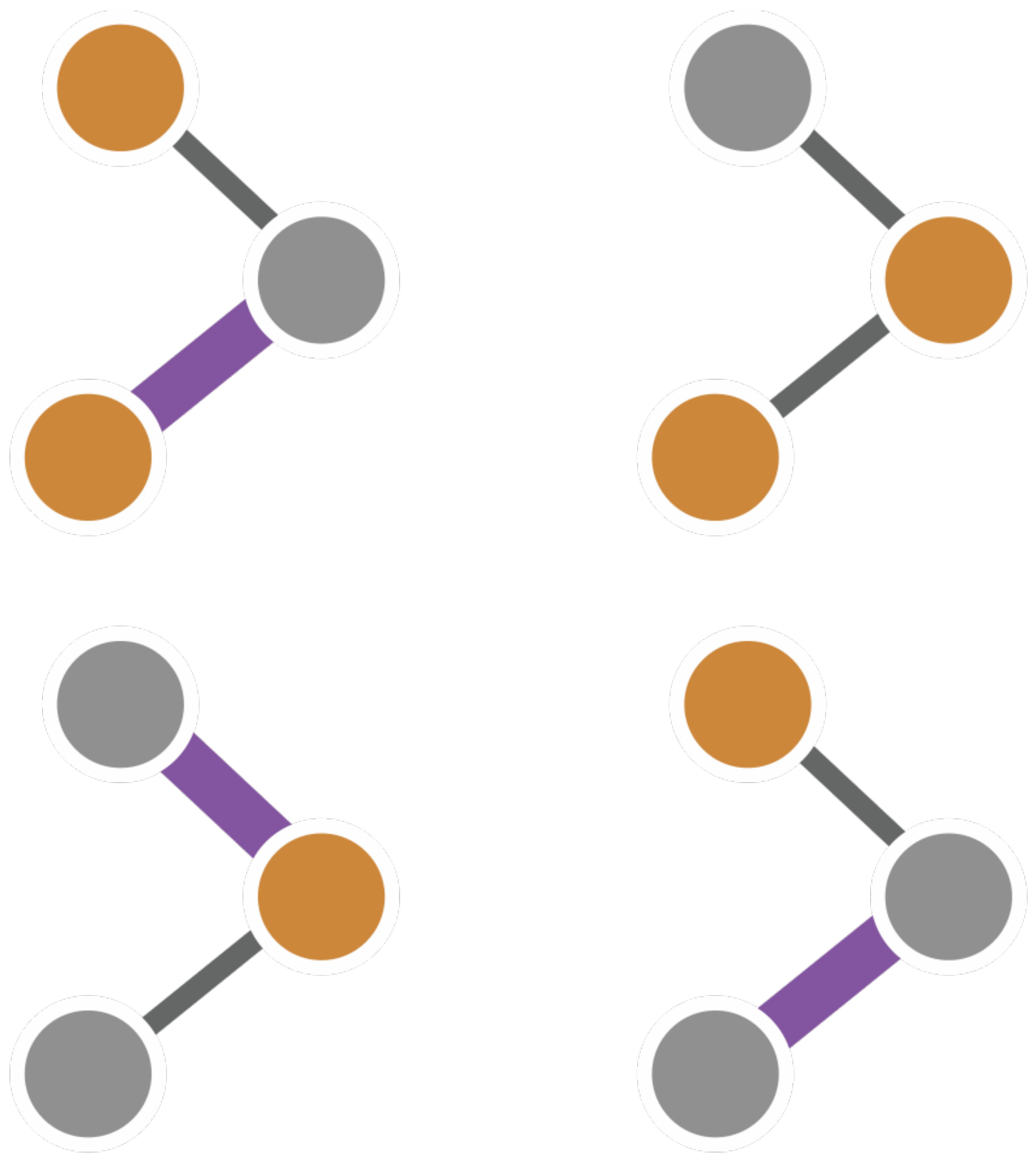


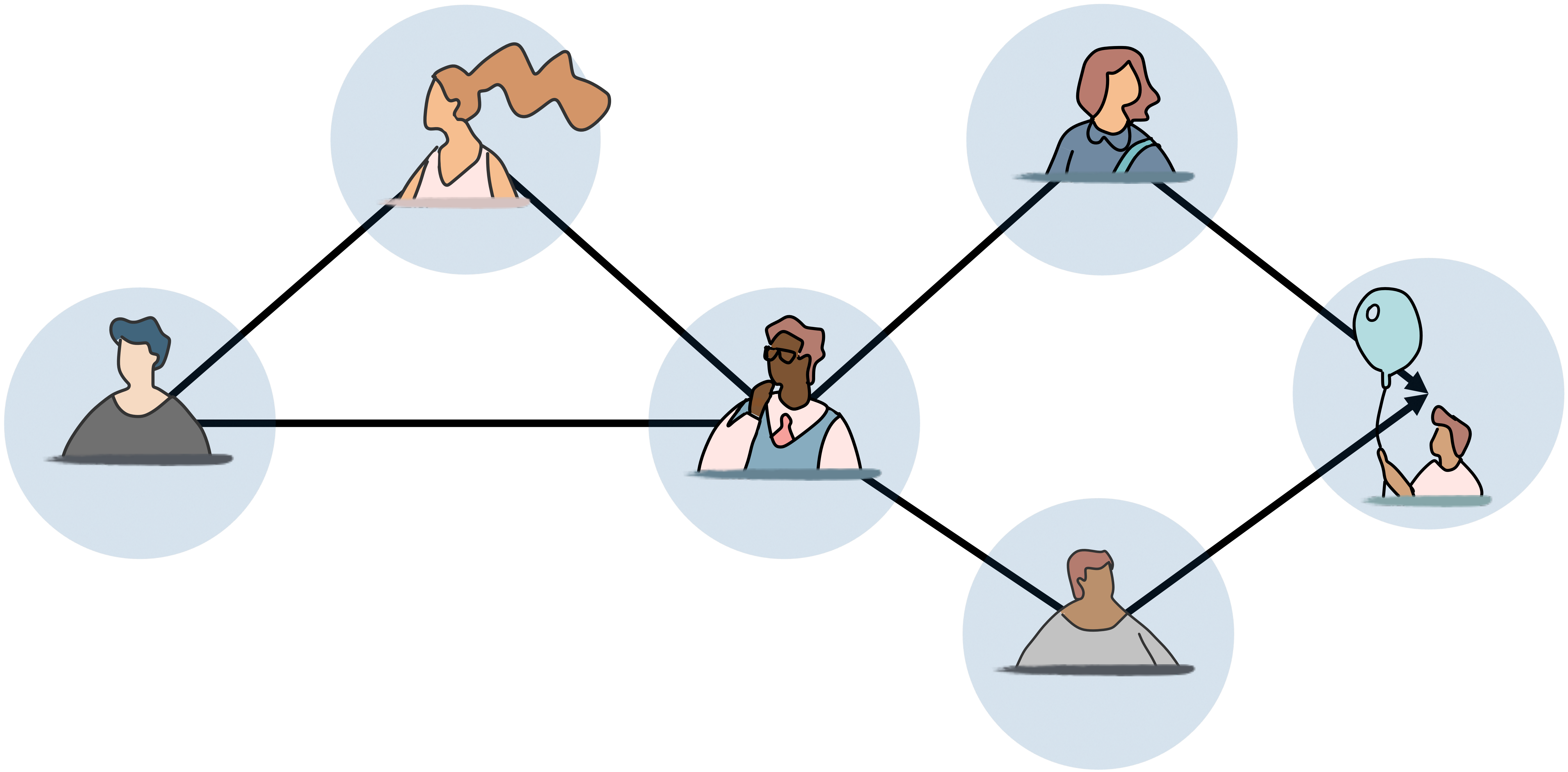
Small Multiples



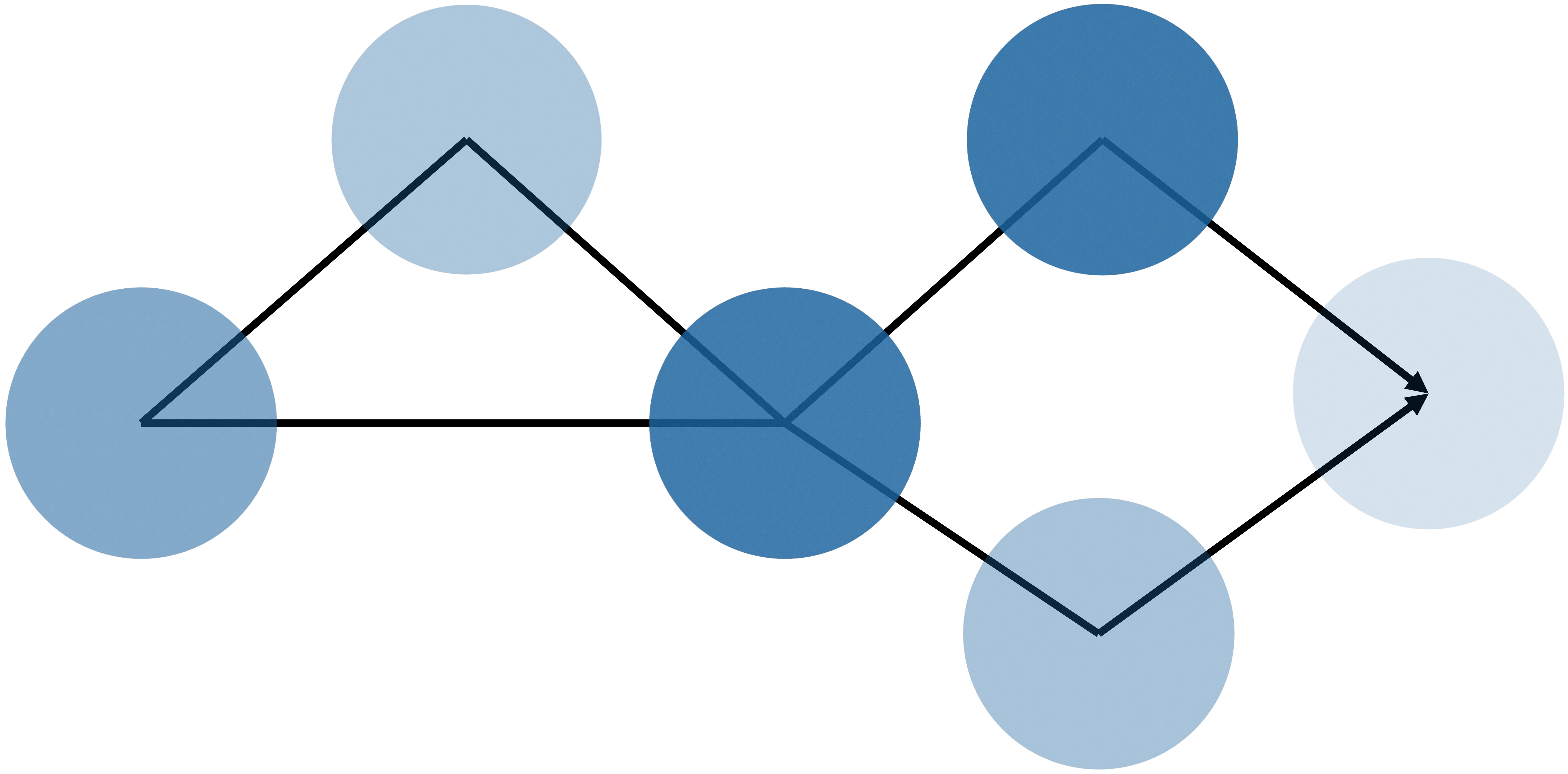
Hybrids

Small Multiples





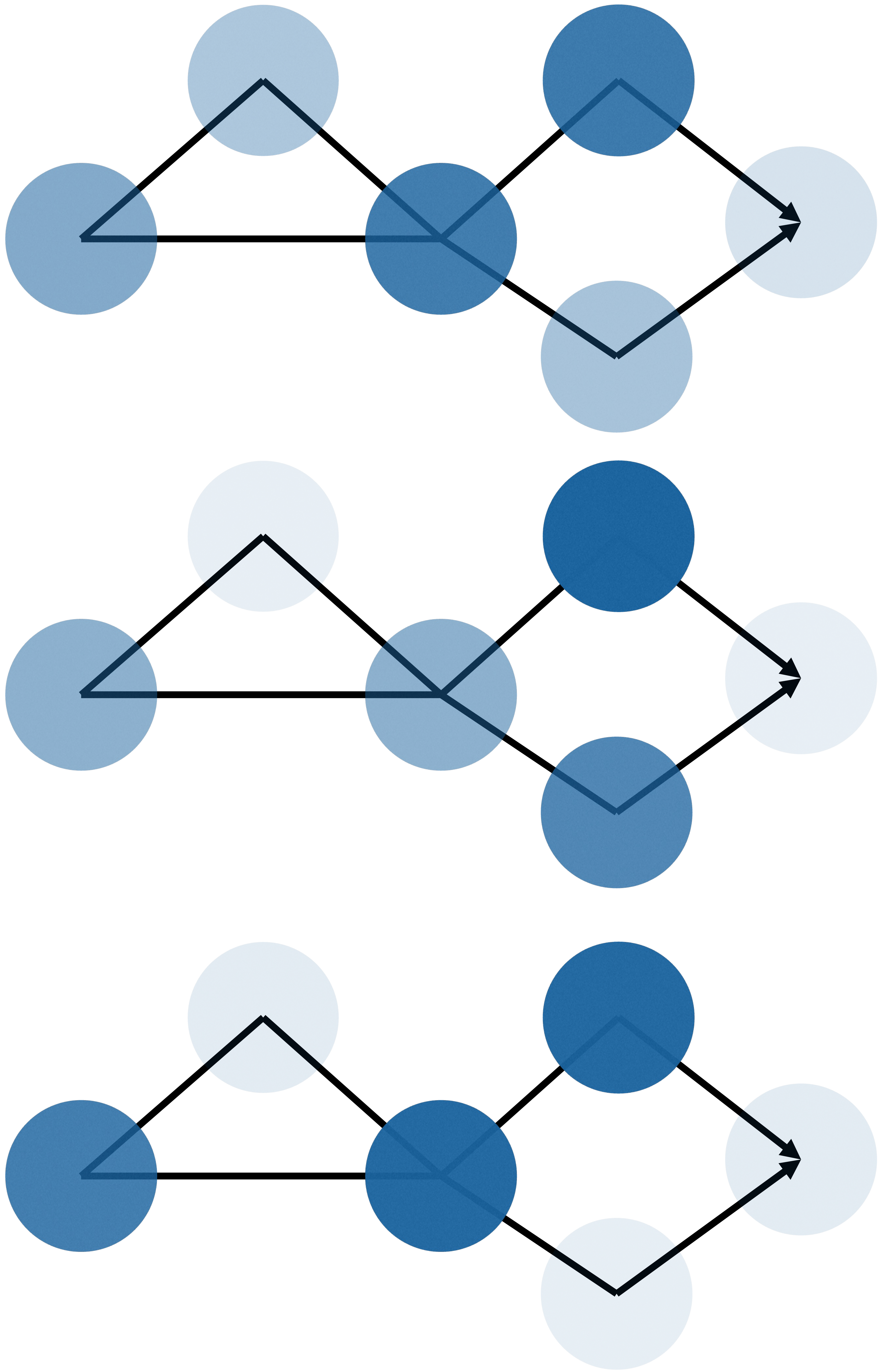
Day 1

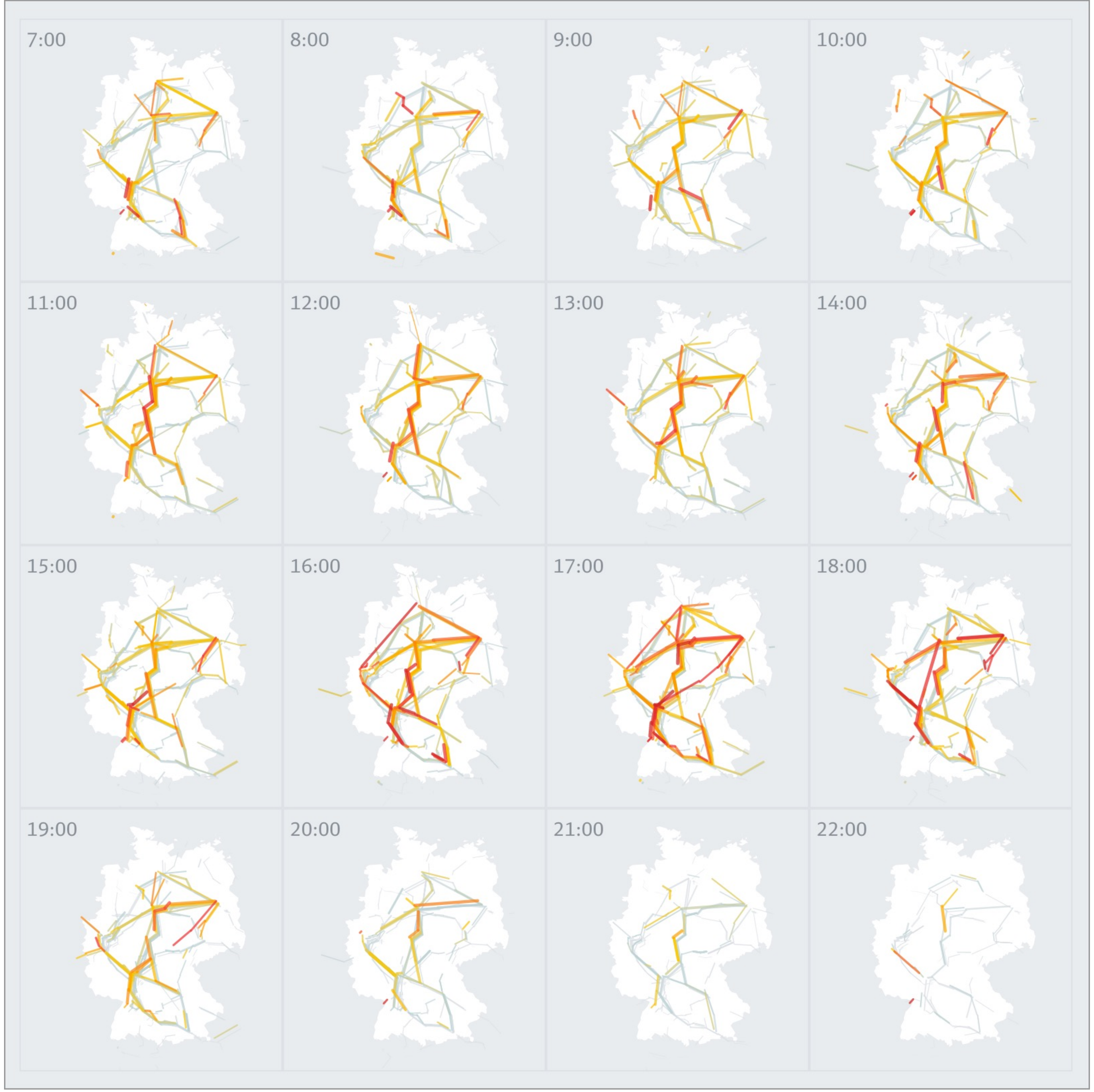


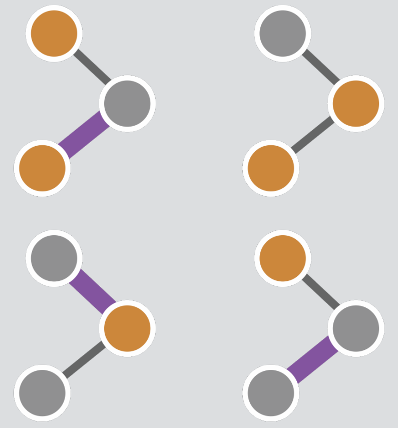
Day 1

Day 2

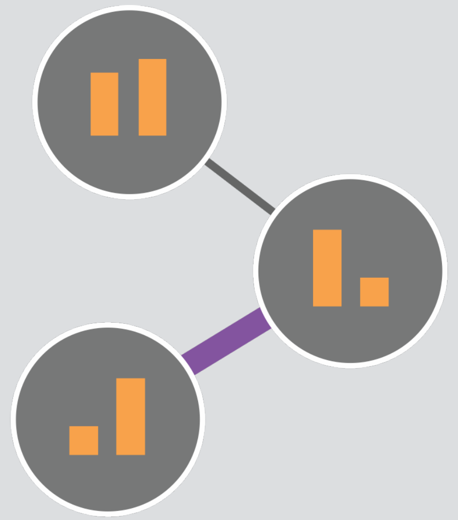
Day 3







Small Multiples

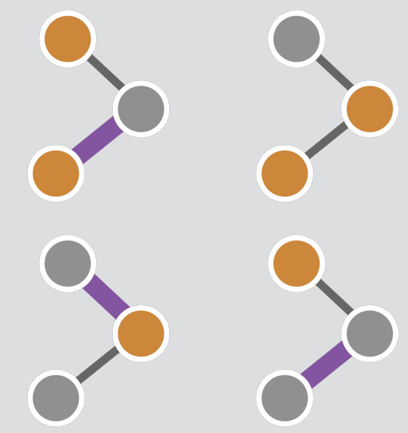


On-Node / On-Edge Encoding

Peakspotting - https://truth-and-beauty.net/projects/peakspotting

Small Multiples

Common layout facilitates attribute comparisons in specific topological features

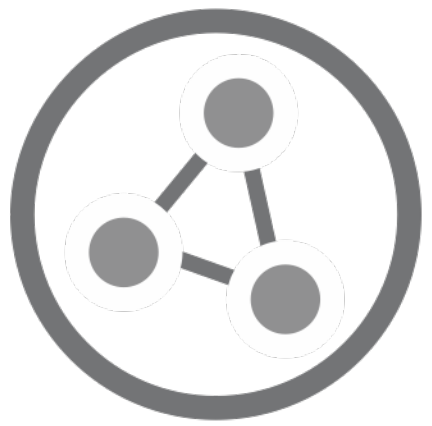


Small Multiples

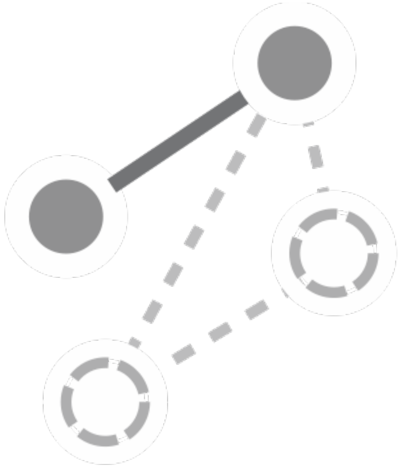
Not ideal for large networks, or tasks on clusters

Recommended for small networks where the tasks are focused on attribute comparison

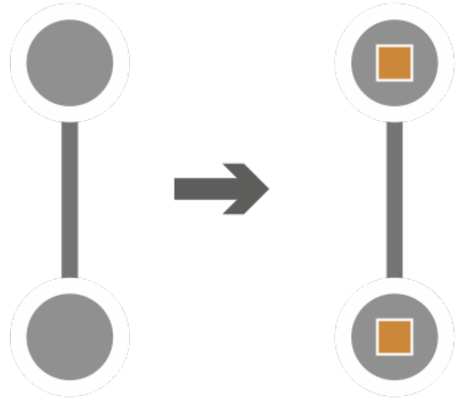
Data Operations



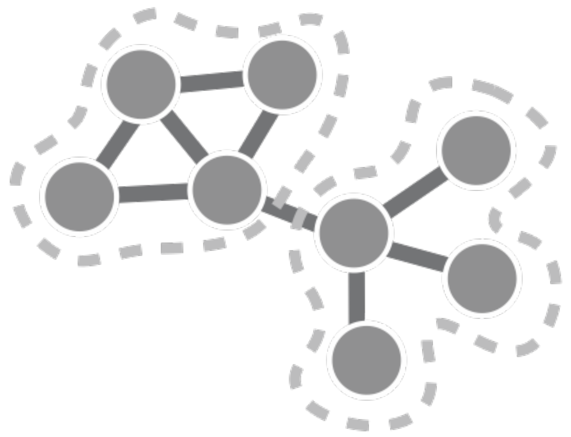
Aggregating Nodes/Edges



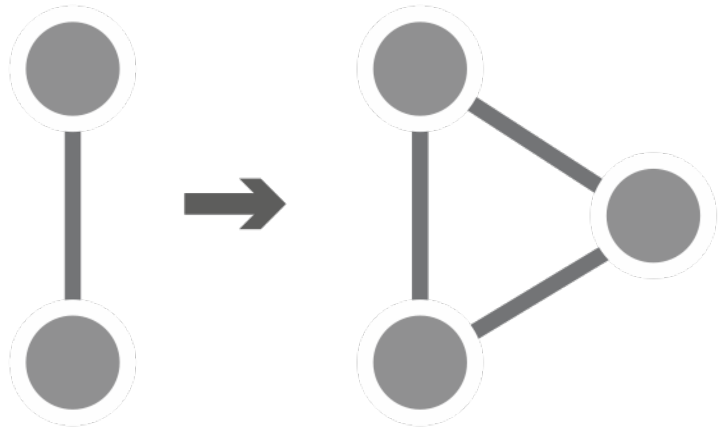
Querying and Filtering



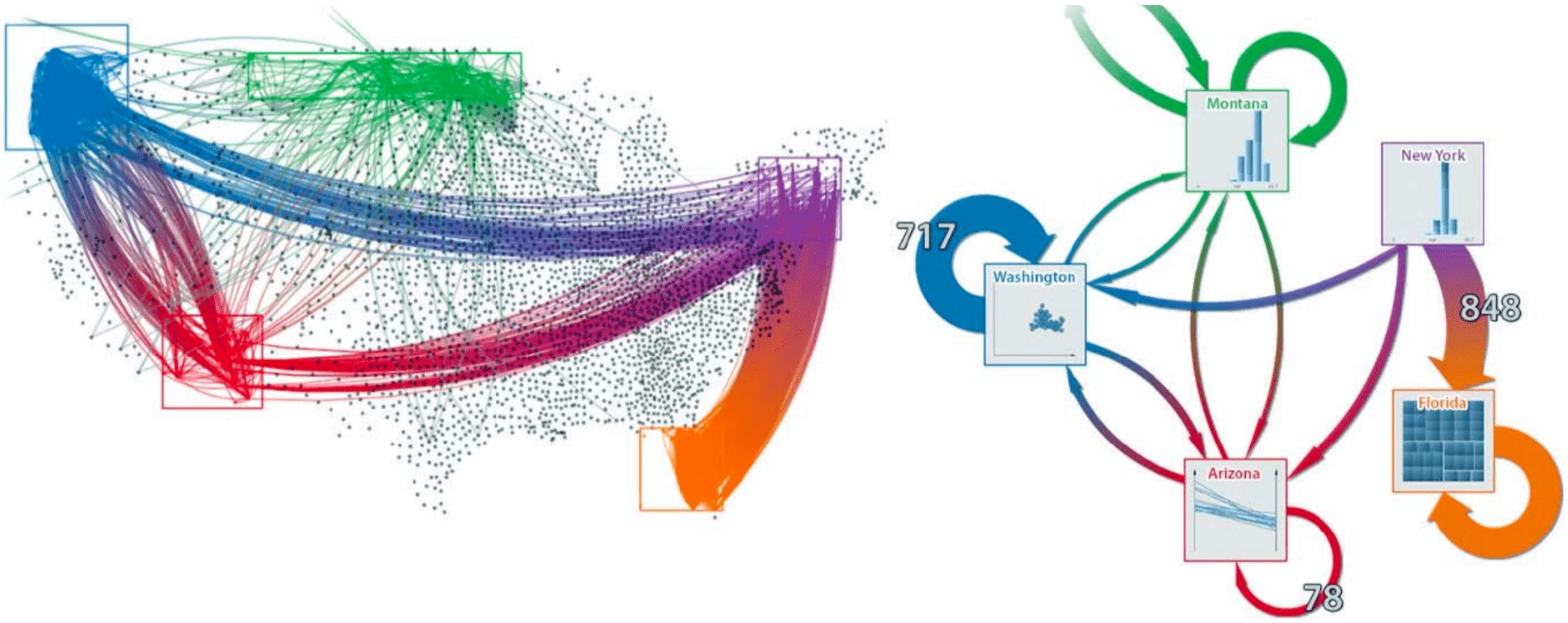
Deriving New Attributes



Clustering



Converting Attributes/Edge to Nodes



Elzen and Wijk, 2014

Aggregating Nodes/Edges

Multivariate Network Visualization Techniques

A companion website for the STAR Report on Multivariate Network Visualization Techniques.

HOME TECHNIQUES WIZARD

About

This is a companion website for a review article on multivariate network visualization techniques.

Multivariate networks are networks where both the structure of the network and the attributes of the nodes and edges matter. It turns out, these are very common. Every person in a social network, for example, has both, relationships and lots of other characteristics, such as their age, the school they went to, or the city they live in. Multivariate network visualization techniques are designed to be able to show both, these attributes and the structure. Using these visualization techniques, we can analyze, for example, if a network of friends predominantly went to the same high school.

The visualization research community has developed many techniques to visualize these kinds of networks, and our review article – and this website – are designed to help you sort through these options.

Browse through the techniques illustrated below, or use our wizard to find the right multivariate network visualization technique for your datasets and tasks!

Get in touch if you have questions or comments.

Use the Wizard

Technique recommendations to fit your needs!

Navigate to the wizard tab and select your specific network characteristics, such as the size of the network and its type, and what tasks are relevant for your analysis and receive technique recommendations that are best suited to your selection.

Read the Review Article

The State of the Art in Visualizing Multivariate Networks
Carolina Nobre, Miriah Meyer, Marc Streit, and Alexander Lex
To appear in Computer Graphics Forum (EuroVis 2019)

vdl.sci.utah.edu/mvnv/