Part IV Arranging Space

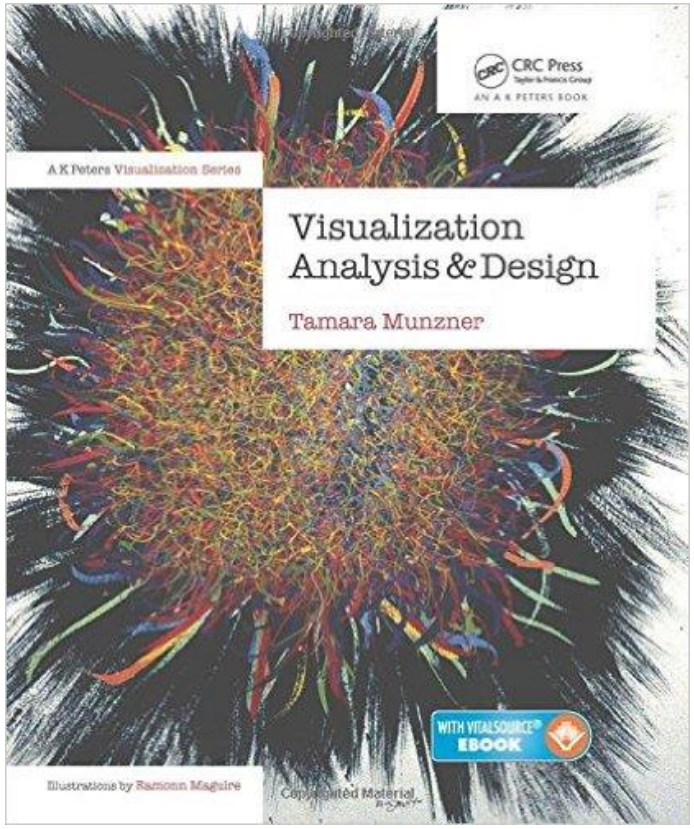
Tables

The bearing of a child takes nine months,
no matter how many women are assigned.

Frederick P. Brooks Jr.

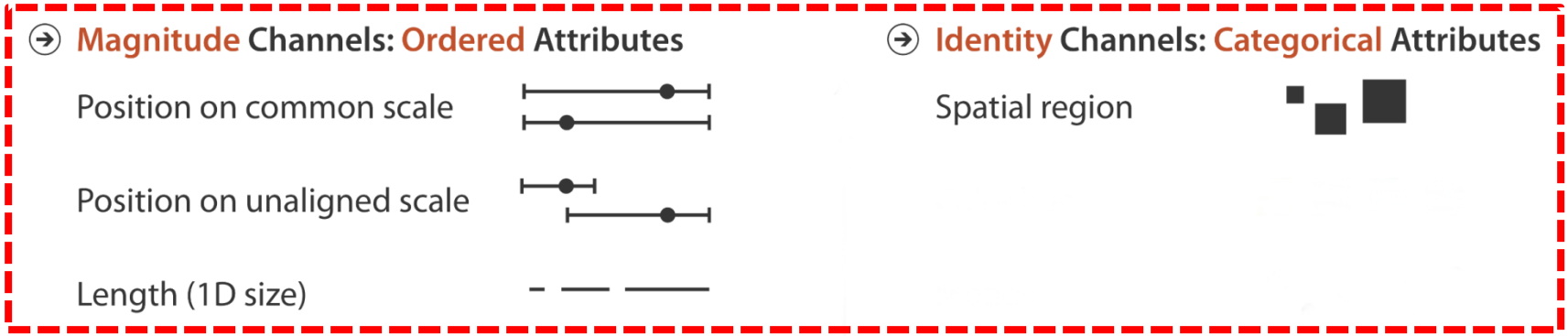
Source of This Unit

- **Material of this unit is based on Chapter 7 of Tamara Munzner, *Visualization Analysis and Design*, AK Peters/CRC Press, 2014.**



Channel Ranking

- **The three highest ranked effectiveness channels for quantitative and ordered attributes are all related to spatial position: planar position on common scale, planar position along a magnitude scale, and length.**
- **The highest ranked effectiveness channel for categorical attributes (e.g., grouping) is also about the use of space.**



Keys and Values

- A **key** is an independent attribute that can be used as a unique index to look up items in a table.
- A **value** is a dependent attribute, the value of a cell in a table.
- Key attributes can be categorical or ordinal, whereas values can be all three of the types: categorical, ordinal, or quantitative.
- The unique values for a categorical or ordered attribute are called **levels**.

Keys and Values

- **The visual encoding choices directly relate to the semantics of the table's attributes: how many keys and how many values does it have?**
  - **An idiom could only show values, with no key – scatter plots can show two value attributes**
  - **An idiom shows one key and one value – bar charts**
  - **An idiom shows two keys and many values – heatmaps**
  - **Idioms that show many keys and many values often recursively subdivide space into many regions, as with scatter plot matrices.**

Express: Quantitative Values 1/2

- Using space to express quantitative attributes is a simple use of the spatial position channel.
- The attribute is mapped to spatial position along an axis.
- The simplest case is encoding a single attribute, each item is encoded with a mark at some position along the axis.
- Additional attributes may also be encoded on the same mark with other nonspatial channels such as color and size.

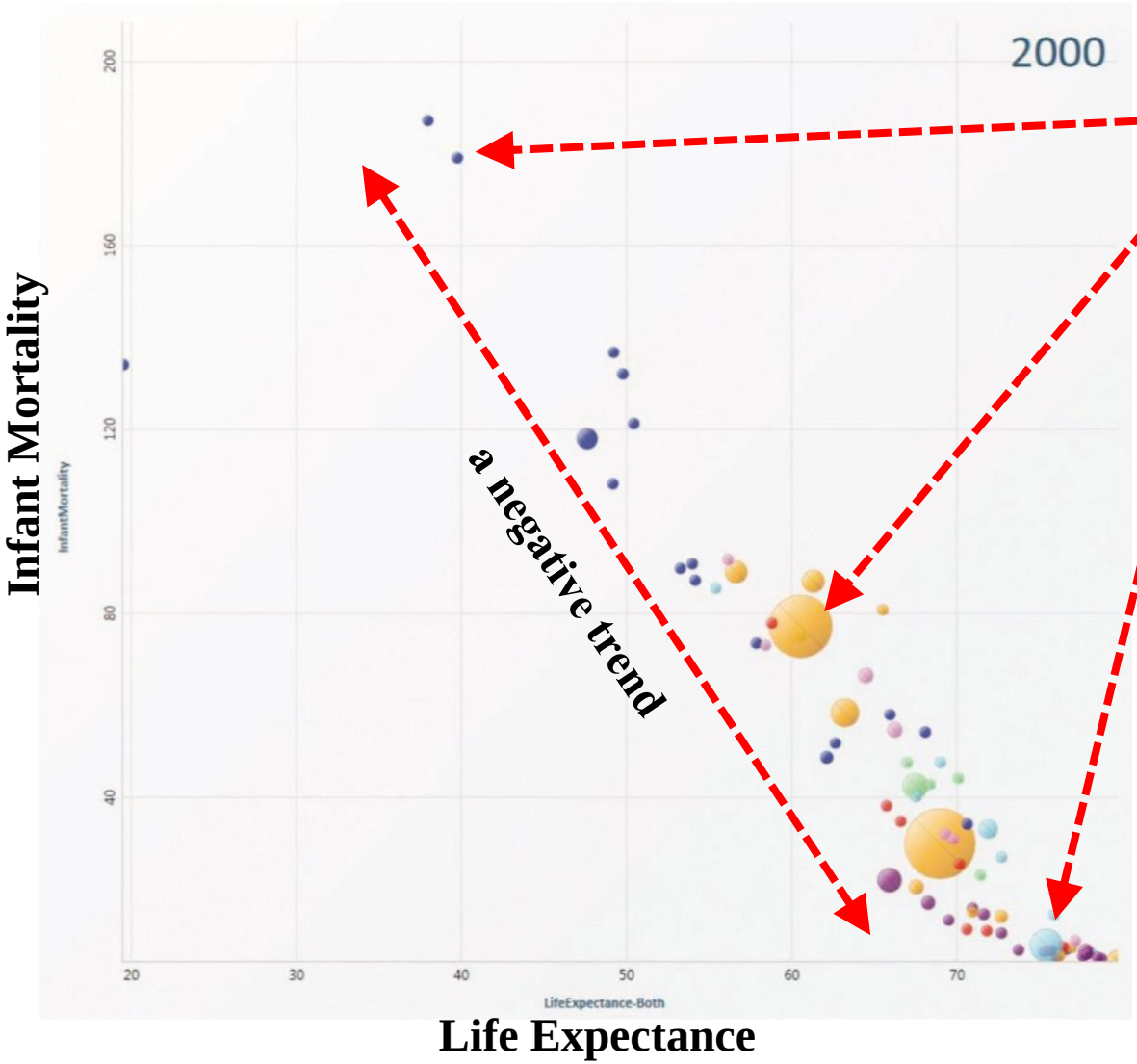
Express: Quantitative Values 2/2

- A **glyph** is an object with internal structure that arises from multiple marks.
- In a more complex situation, a composite glyph object is drawn, with internal structure that arises from multiple marks.
- Each mark lies within a subregion in the glyph that is visually encoded differently, so the glyph can show multiple attributes at once.

Example: Scatter plots 1/4

- **The idiom of scatter plots encodes two quantitative value variables using both the *horizontal* and *vertical* spatial position channels, and the mark type is a point.**
- **Scatter plots are effective for the abstract tasks of providing *overviews* and characterizing *trends*, *distributions*, and specifically for finding *outliers* and *extreme values*.**
- **Scatter plots are also very effective for the abstract task of judging the *correlation* between two attributes.**

Example: Scatter plots 2/4



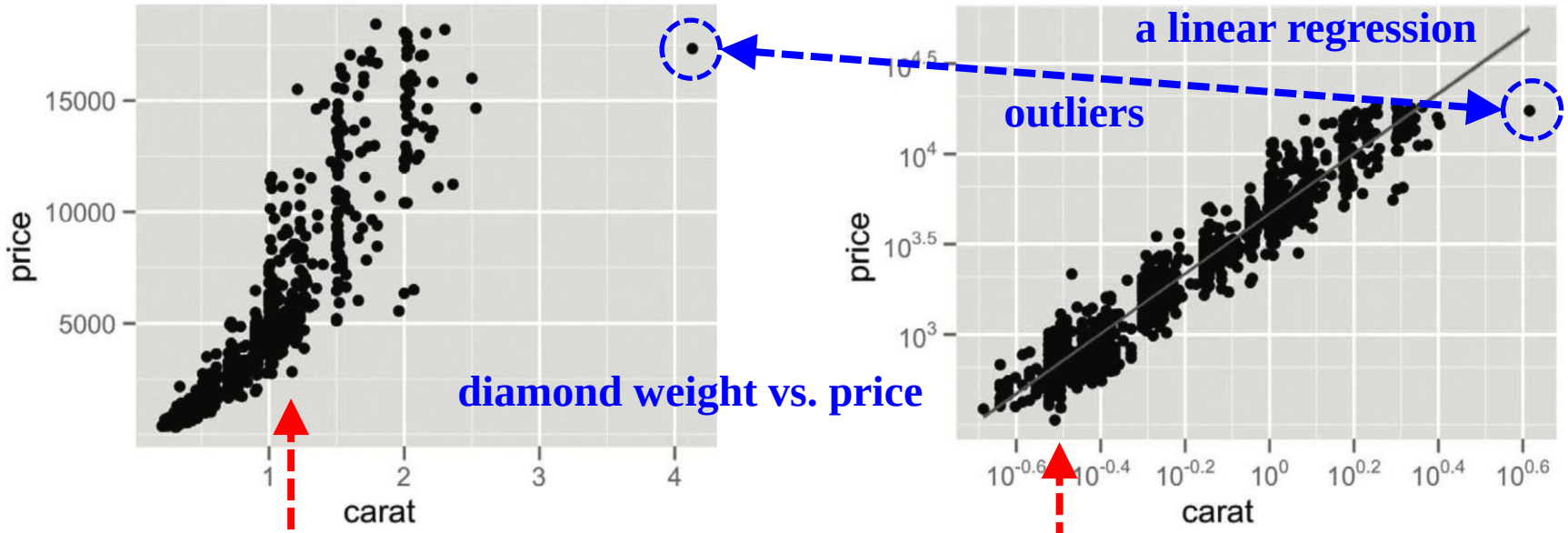
Each point represents a country

The color channel is used for the categorical country attribute

The size channel is for quantitative population attribute

Example: Scatter plots 3/4

- **Transformations may be applied to quantitative data value to reveal more insight.**



Original data shows an exponential trend

Transforming both attributes (i.e., weight and price) with logarithm, the new plots appears to have a linear correlation.

Example: Scatter plots 4/4

- **The scalability of a scatter plot is limited by the need to distinguish points from each other.**
- **Thus, it is good for hundreds of items.**

Idiom	Scatter Plots
Data Type	Two quantitative value attributes (Table)
Encoding	Express values with horizontal and vertical spatial position with point marks
Task	Find trends, outliers, distribution, correlation; local clusters
Scale	Hundreds items

Separate, Order and Align Categorical Regions: 1/4

- **The use of space to encode categorical attributes is more complex.**
- **This is because spatial position is an *ordered* magnitude visual channel, but categorical attributes have *unordered* identity semantics.**
- **If categorical attributes are encoded with spatial position, the principle of expressiveness would be violated.**

Separate, Order and Align Categorical Regions: 2/4

- **The semantics of categorical attributes does match up well with the idea of a spatial region: regions are contiguous bounded areas that are distinct from each other.**
- **Drawing all items with the same values for a categorical attribute within the same region uses spatial *proximity* to encode *similarity*.**
- **These regions must be given spatial positions in the plane in order to draw any specific picture.**

Separate, Order and Align Categorical Regions: 3/4

- **The problem in drawing regions can be broken down into three operations: separating into regions, aligning the regions, and ordering the regions.**
- **The separation and the ordering must happen, but the alignment is usually optional.**
- **The separation should be done according to a *categorical* attribute, the ordering and alignment should be done by other *ordered* attribute.**

Separate, Order and Align Categorical Regions: 4/4

- **The attribute used to order the regions must have ordered semantics, and hence it cannot be the categorical one that was used to do the separation.**
- **If alignment is done, the ordered attribute used to control the alignment between regions may be the same one used to encode the spatial position of items *within* the region.**

List Alignment: One Key

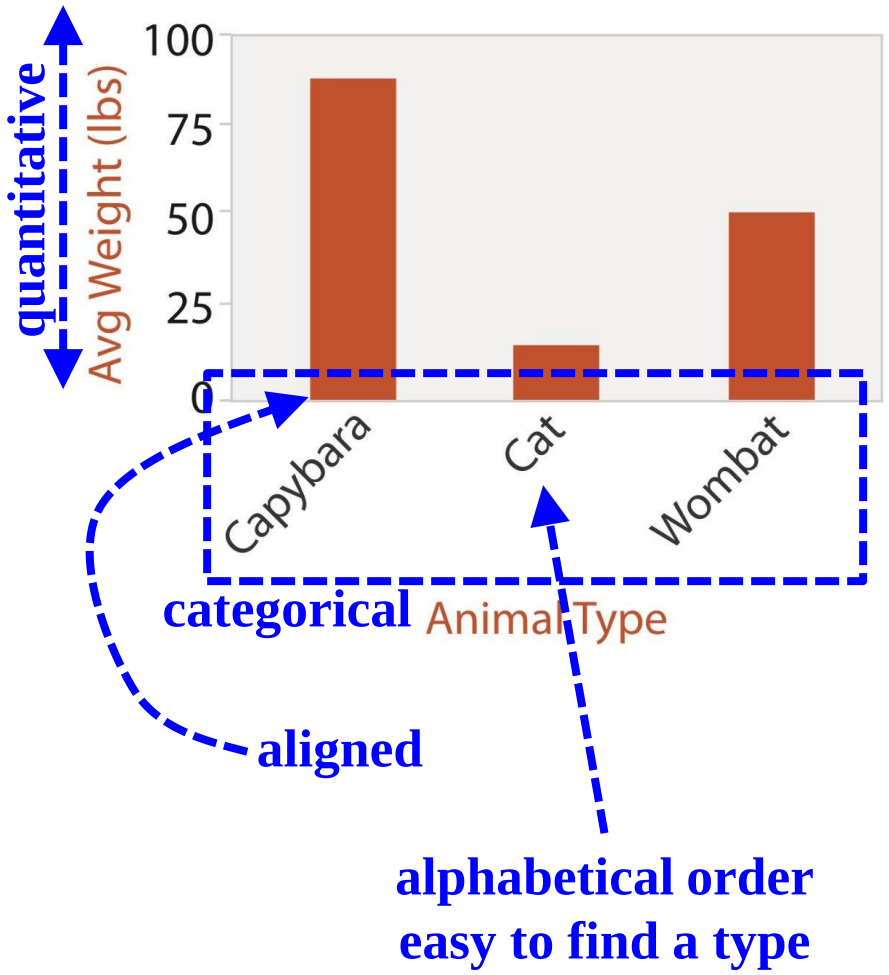
- **If we have only one key, using the key yields one region per item separates the data into regions.**
- **The regions are frequently arranged in a one-dimensional list alignment, either horizontal or vertical.**
- **The view itself is a two-dimensional area: the aligned list of items uses one of the spatial dimension, and the region to show the values uses the other.**

Examples

- **The following examples will be discussed to illustrate the List Alignment of One Key**
  - **Bar Charts**
  - **Stacked Bar Charts**
  - **Stream Graphs**
  - **Dot and Line Charts**

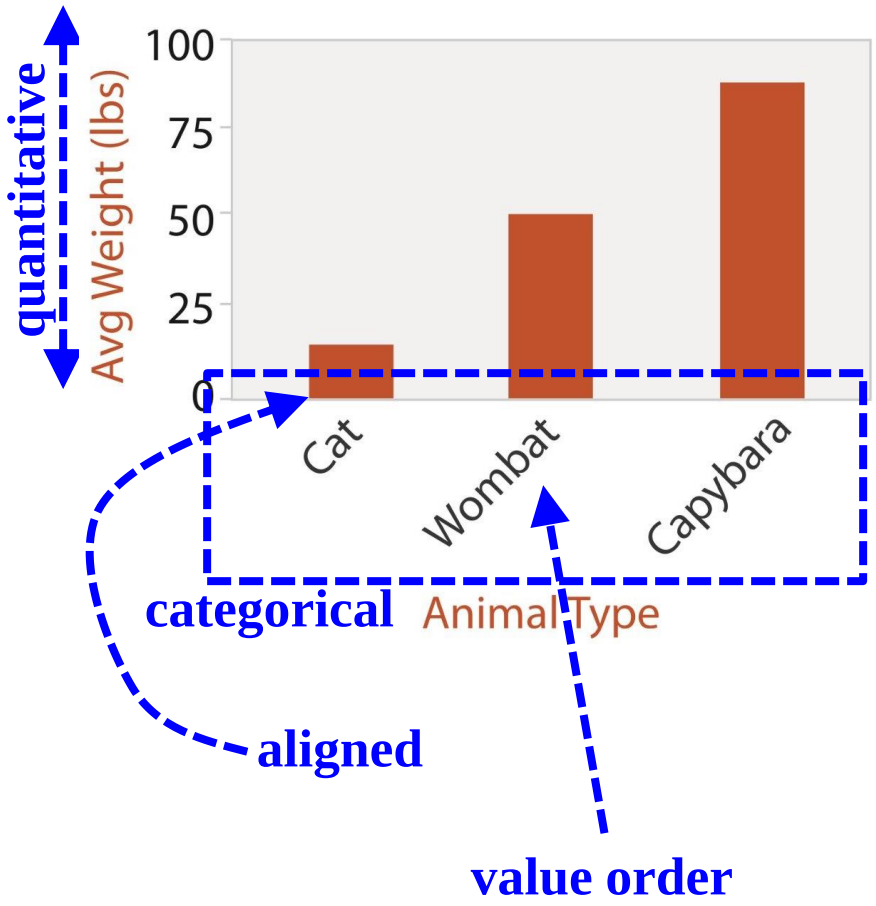
Example: Bar Charts 1/3

- The bar chart idiom uses a line mark and encode a quantitative value attribute with one spatial channel.
- The other attribute is a categorical key attribute.
- Each line mark forms a separate region of space.
- The line marks are all aligned position channel.



Example: Bar Charts 2/3

- A default arrangement may hide what could be meaningful patterns in the dataset.
- The right chart uses animal weight order that is encoded by the bar height.
- The scalability with bar charts are the available space for all the bars.



Example: Bar Charts 3/3

- **The following table is a summary of the use of bar charts.**

Idiom	Bar Charts
Data Type	Table: One quantitative attribute, one categorical key attribute
Encoding	Line marks, express value attribute with aligned vertical position, separate key attribute with horizontal position
Task	Good for lookup and compare values
Scale	Dozens to hundreds of levels

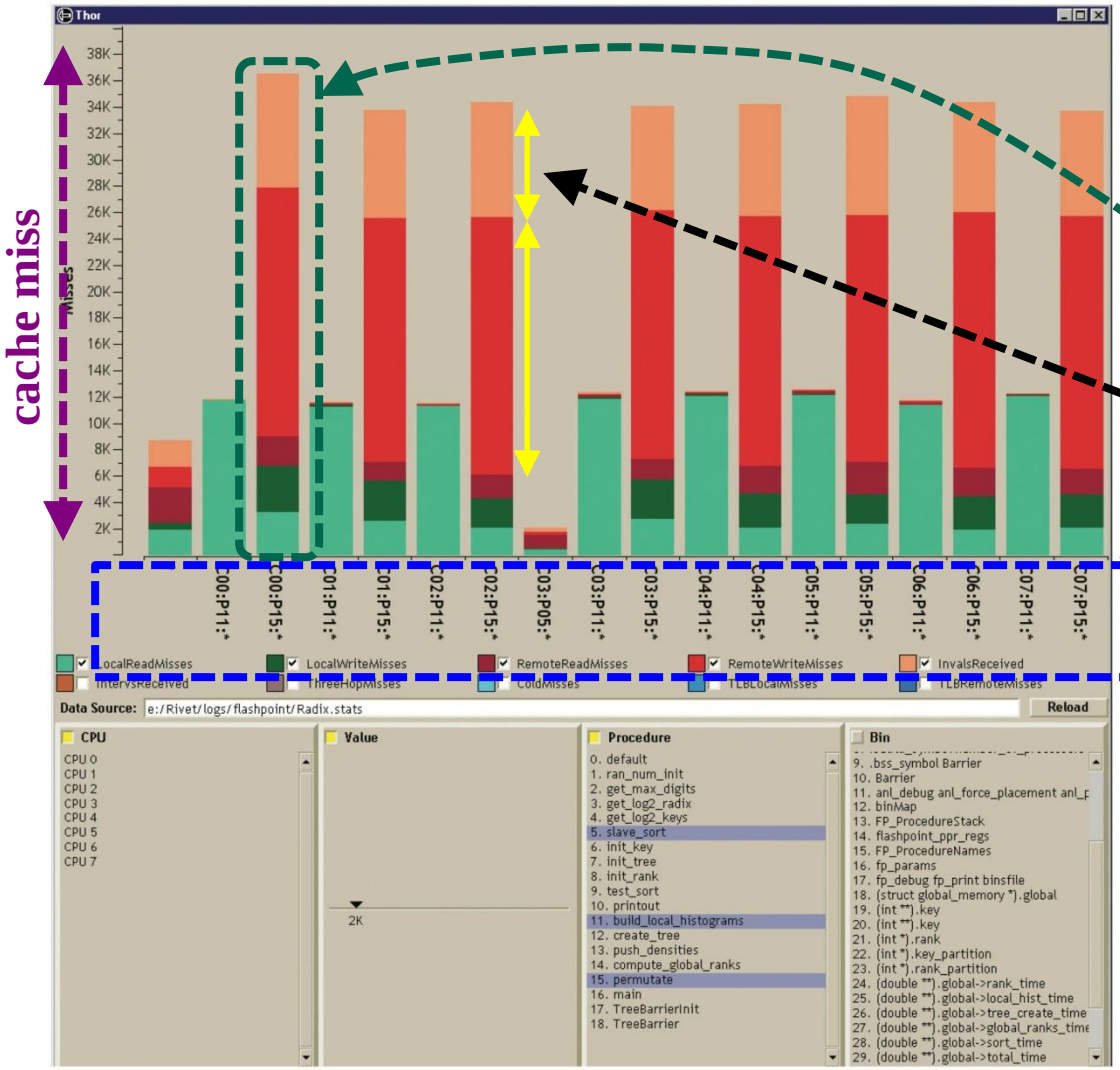
Example: Stacked Bar Charts 1/5

- The **stacked bar chart** is a more complex glyph for each bar, where multiple sub-bars are stacked vertically.
- The length of the composite glyph still encode a value as in a standard bar chart, but each component also encode a length-encoded value.
- Stacked bar charts show information about multidimensional tables, specifically a two-dimensional table with two keys.

Example: Stacked Bar Charts 2/5

- **Stacked bar charts are an example of a list alignment used with more than one key attribute and support the task of lookup according to either of the two keys.**
- **Stacked bar charts typically use color and length coding.**
- **Each subcomponent is colored according to the same key that is used to determine the vertical ordering.**

Example: Stacked Bar Charts 3/5



Processor/Procedure vs. cache miss

a glyph

type of cache miss

aligned

categorical: processor/procedure

Example: Stacked Bar Charts 4/5

- **The cache miss bars can easily be compared because they are aligned.**
- **However, each sub-component in a glyph is difficult to compare because they are not aligned to a common scale.**
- **The order of stacking is significant for the kinds of patterns that are most easily visible.**
- **Scalability is similar to standard bar charts in terms of categories, but it is more limited for the key used to stack the subcomponents.**

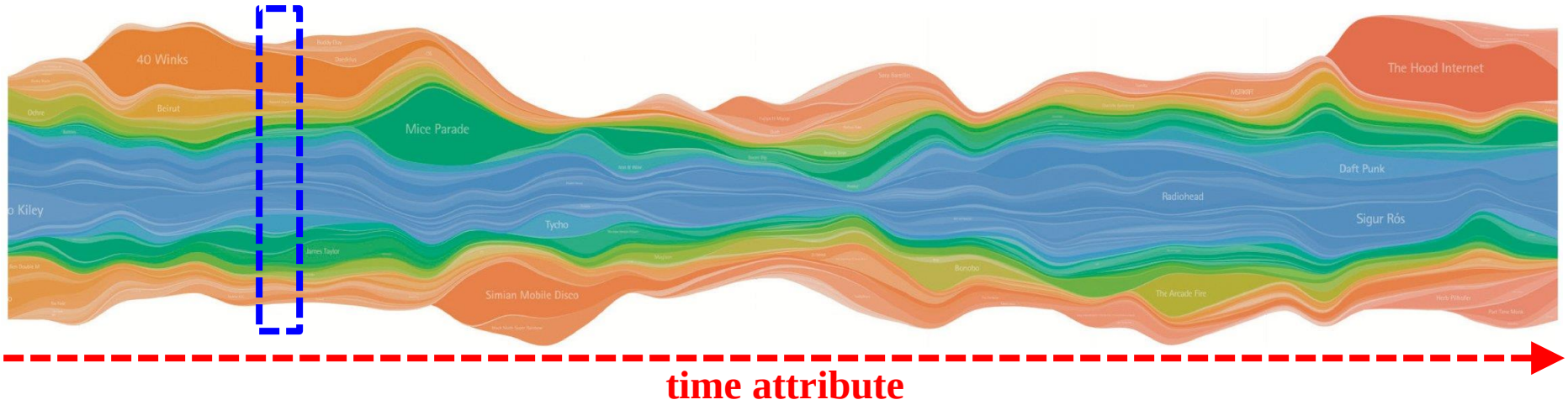
Example: Stacked Bar Charts 5/5

Idiom	Stacked Bar Charts
Data Type	Multidimensional Table: One quantitative attribute, two categorical key attributes
Encoding	Bar glyph with length-coded subcomponents of value attribute for each category of secondary key attribute. Separate bars by category of primary key attribute.
Task	Part-to-Whole relationship, lookup values, find trends
Scale	Key attribute (main axis): dozens to hundred of levels. Key attribute (stacked glyph axis): several to one dozen.

Example: Stream Graphs 1/6

- A **stream graph** is a type of stacked area graph which is displaced around a central axis, resulting in a flowing, organic shape.

glyph: the number of times a musician's music was listened to each week

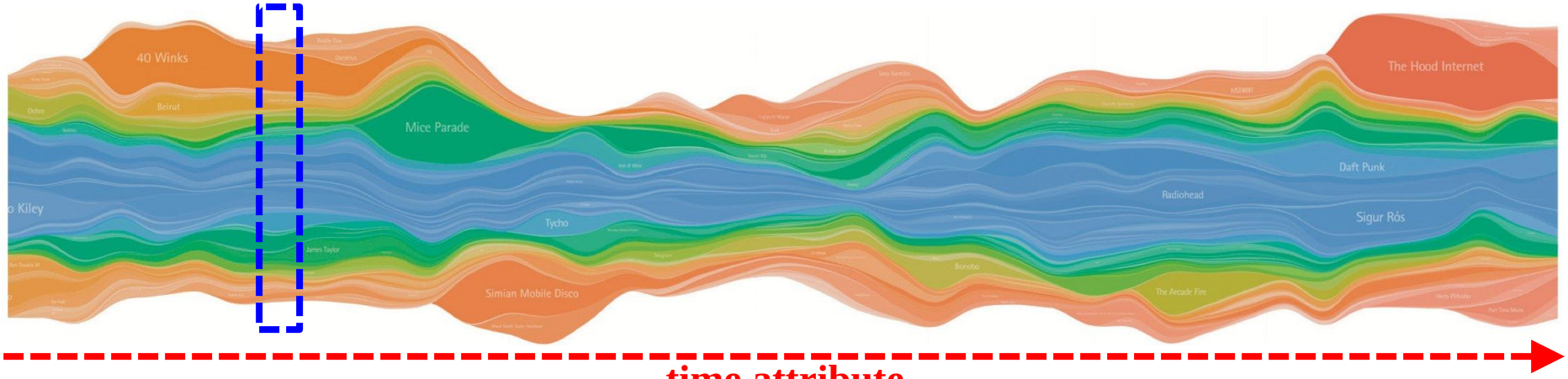


time attribute

Example: Stream Graphs 2/6

- The **stream graph** idiom shows derived geometry that emphasizes the continuity of the horizontal layers that represented the artists, rather than showing individual vertical glyphs that would emphasize listening behavior at a specific time.

glyph: the number of times a musician's music was listened to each week



time attribute

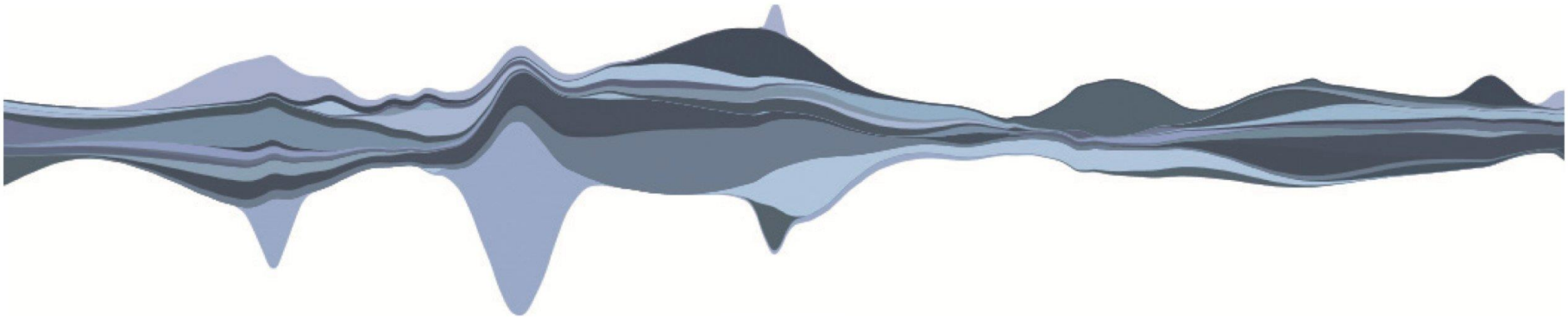
Example: Stream Graphs 3/6

- **The derived geometry is the result of a global computation, whereas individual glyphs can be constructed using only calculations about their own local region.**
- **The stream graph idiom emphasizes the legibility of the individual streams with a deliberately organized silhouette, rather than using the horizontal axis as the baseline.**

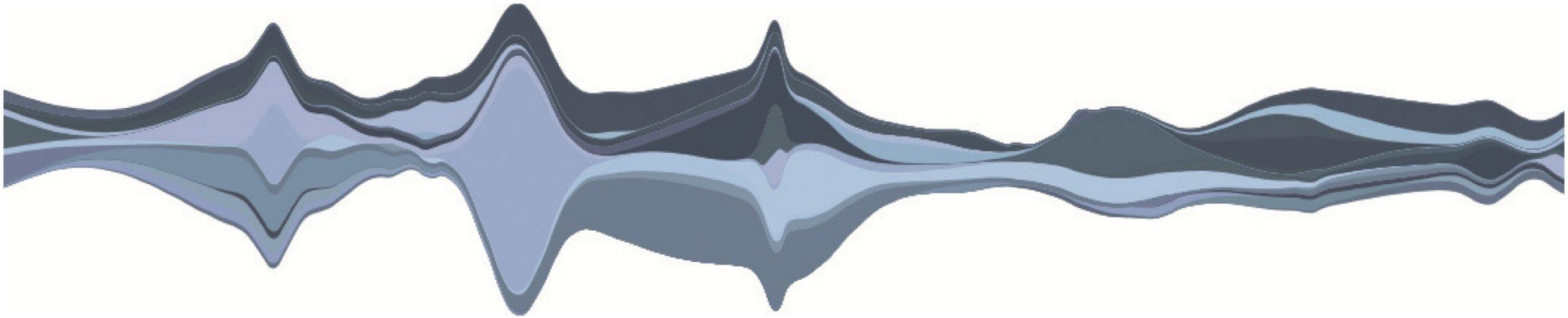
Example: Stream Graphs 4/6

- **The shape of the layout is optimized as a trade-off between multiple factors:**
  - **A. the external silhouette of the entire shape**
  - **B. the deviation of each layer from the baseline**
  - **C. the amount of wiggle in the baseline**
- **The order of the layers is computed with an algorithm that emphasize a derived value.**

Example: Stream Graphs 5/6



Stream graph with layers ordered by the volatility of artist's popularity



Stream graph with layers ordered by onset time when artist's music of first gain attention

Example: Stream Graphs 6/6

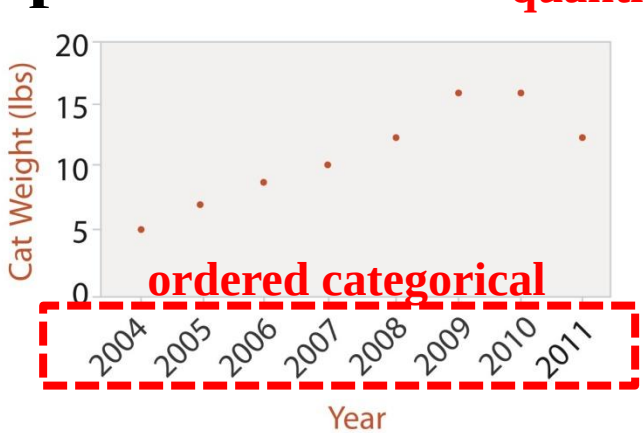
Idiom	Stream Graphs
Data Type	Multidimensional Table: One quantitative attribute (count), one ordered key attribute (time), one categorical key attribute (artist)
Encoding	One quantitative attribute (for layer ordering)
Task	Use derived geometry showing artist layers across time, layer height encodes counts.
Scale	Key attribute (time, main axis): hundreds of time points. Key attribute (artist, short axis): dozens to hundreds

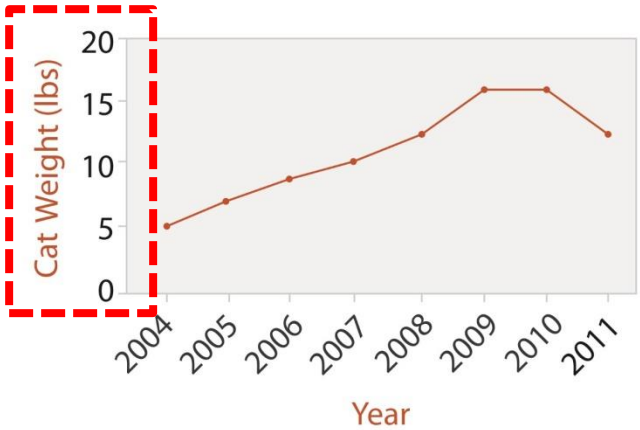
Example: Dot and Line Charts 1/10

- The **dot chart** idiom is a visual encoding of one quantitative attribute using spatial position against one categorical attribute using point marks.
- The **line charts** idiom extends dot charts with line connection marks running between the points.

quantitative attribute

dot chart





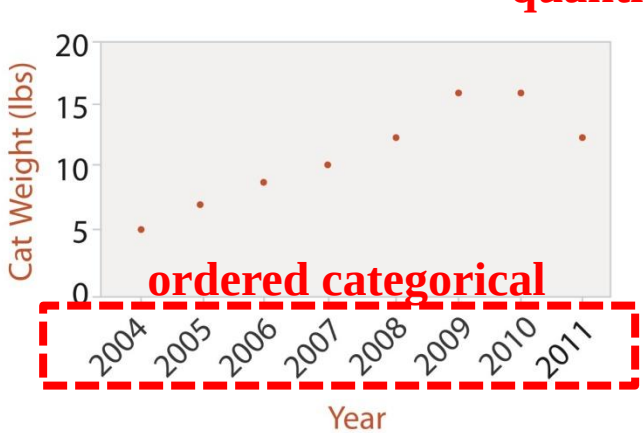
line chart

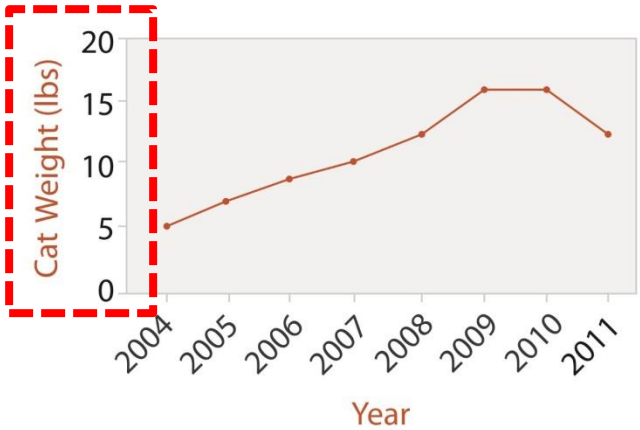
Example: Dot and Line Charts 2/10

- Dot charts, line charts and bar charts all show one value attribute and one key attribute with a rectilinear spatial layout.
- All these chart types are often extended to show a second categorical attribute using color and size channels.

quantitative attribute

dot chart



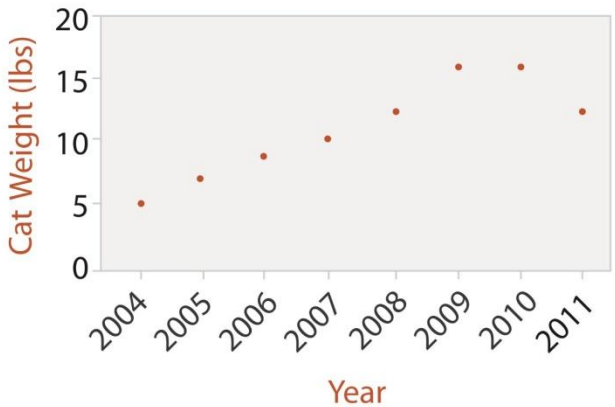


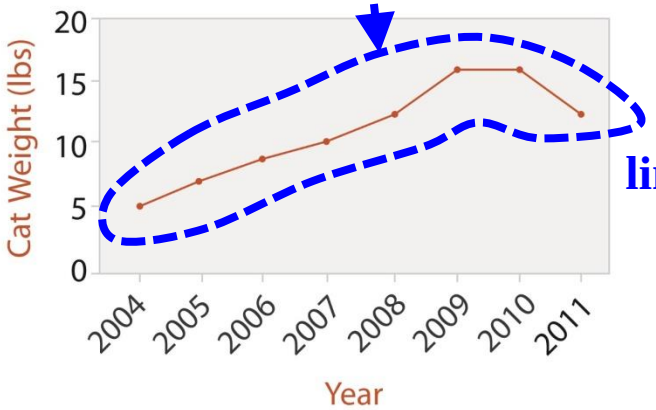
line chart

Example: Dot and Line Charts 3/10

- **Both charts use one spatial position channel to express a quantitative key attribute.**
- **However, line charts also use connections marks to emphasize the ordering of the items and the next one. This implies a *trend* relationship.**

dot chart



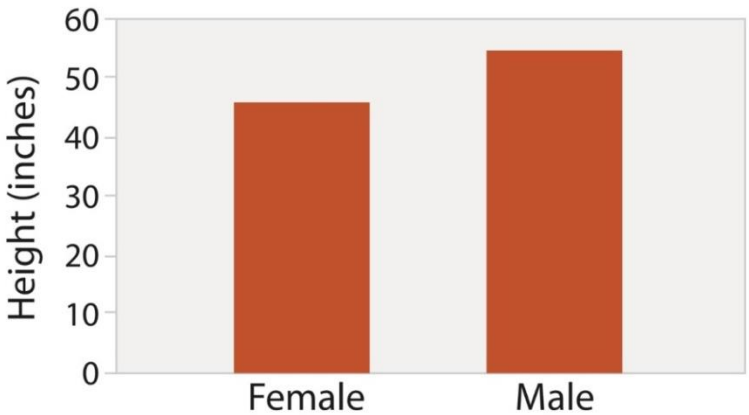


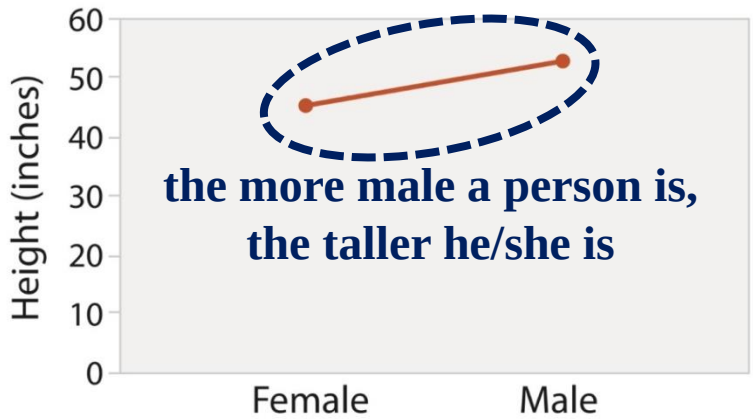
line chart

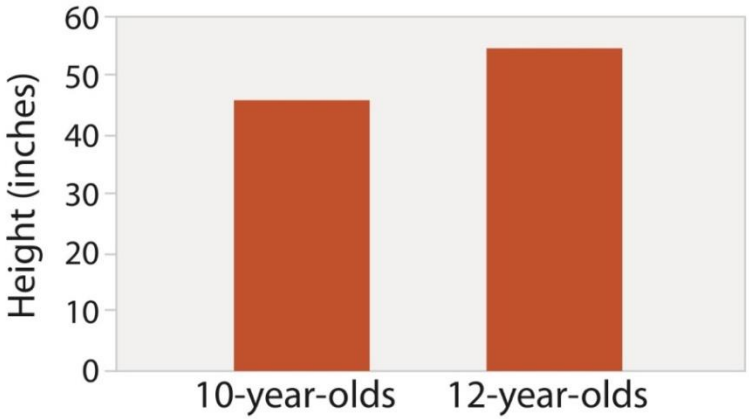
Example: Dot and Line Charts 4/10

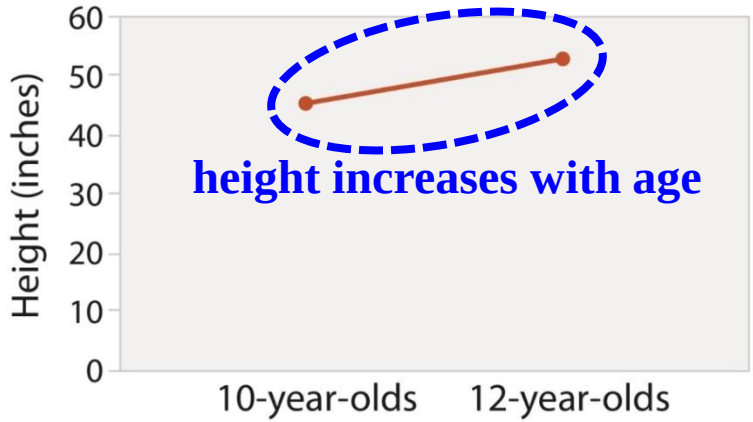
- **Line charts should be used for ordered keys but not categorical keys. A line chart used for categorical data violates the expressiveness principle, because it implies a trend that does not exist.**
- **This implication can be so strong that override common knowledge.**

Example: Dot and Line Charts 5/10









Bar charts encourage discrete comparisons

Line charts encourage trend assessments

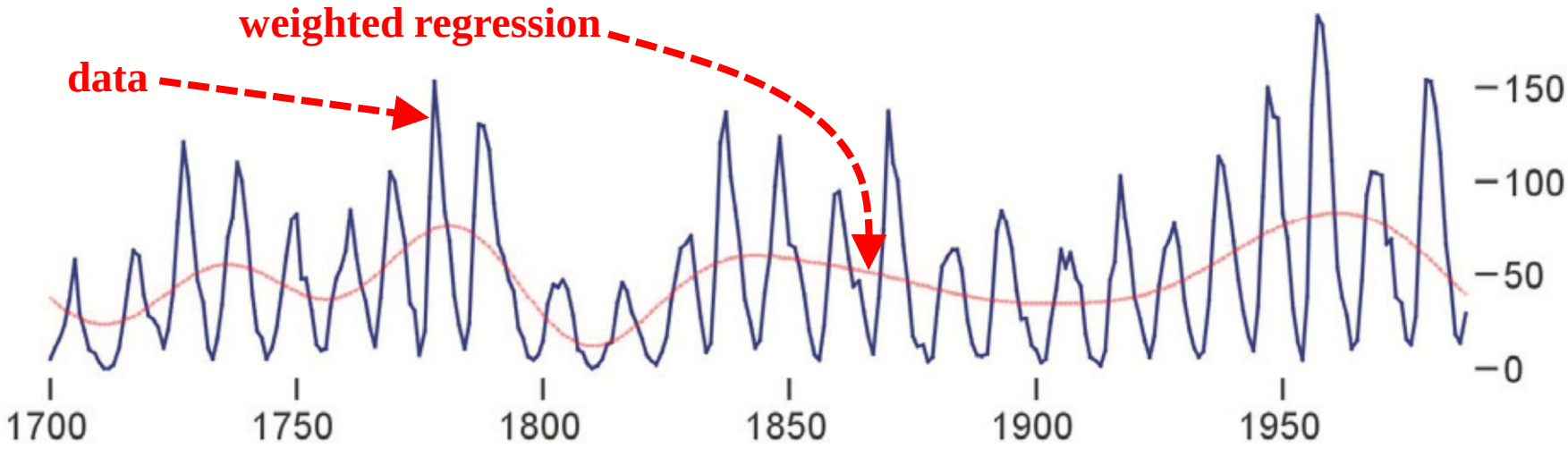
Example: Dot and Line Charts 6/10

- When designing a line chart, an important question to consider is its **aspect ratio**: the ratio of width to height of the entire plot.
- Note that our ability to judge angles is more accurate at exact diagonals (i.e., 45°) than at arbitrary directions.
- The **banking to 45°** idiom computes the best aspect ratio for a chart in order to maximize the number of line segments that fall close to the diagonal.

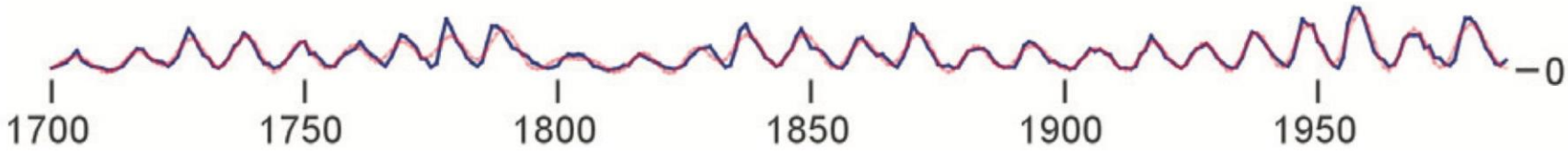
Example: Dot and Line Charts 7/10

- **The multiscale banking to 45° technique, which comes from signal processing, automatically finds a set of informative aspect ratios to analyze the line graph in the frequency domain, with the derived variable of the power spectrum.**

Example: Dot and Line Charts 8/10



Aspect Ratio ≈ 4: Shows the classic low-frequency oscillations in the maximum values of each sunspot cycle



Aspect Ratio ≈ 22: Shows that many cycles have a steep onset followed by more gradual decay

Example: Dot and Line Charts 9/10

Idiom	Dot Charts
Data Type	Table: One quantitative attribute, one ordered key attribute
Encoding	Express value attribute with aligned vertical position and point marks. Separate/order into horizontal regions by key attribute

Example: Dot and Line Charts 10/10

Idiom	Line Charts
Data Type	Table: One quantitative attribute, one ordered key attribute
Encoding	Express value attribute with aligned vertical position and point marks. Separate/order into horizontal regions by key attribute
Why	Show trends
Scale	Key attribute: hundreds levels

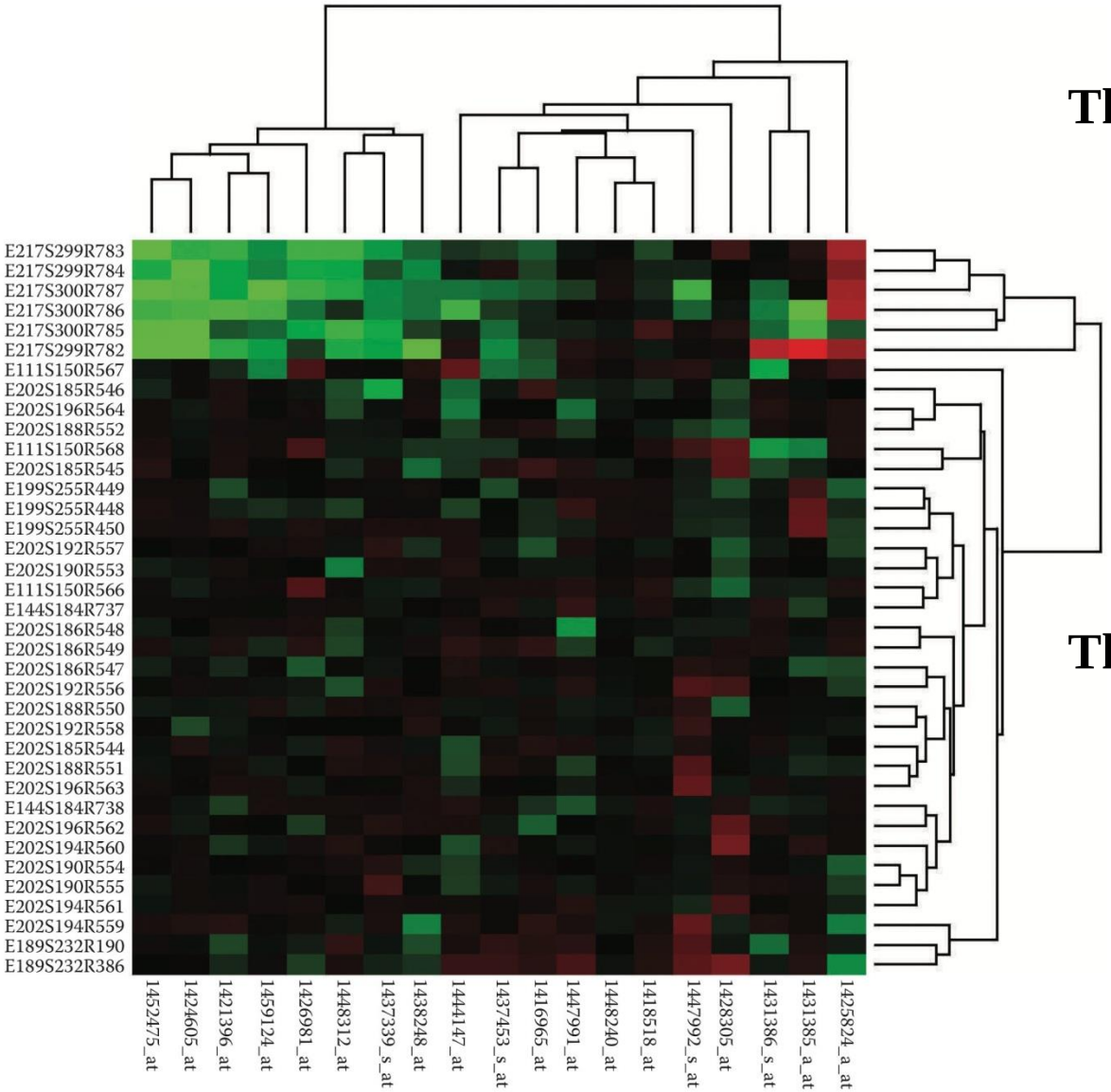
Matrix Alignment: Two Keys

- **Datasets with two keys are often arranged in a two-dimensional matrix alignment, where one key is distributed along the rows and the other along the columns.**
- **Commonly seen arrangements are**
  - **Cluster Heatmaps**
  - **Scatter Plot Matrix**

Example: Cluster Heatmaps 1/9

- **The idiom of heatmap is one of the simplest uses of the matrix alignment.**
- **Each cell is fully occupied by an area mark encoding a single quantitative value attribute with color.**
- **The benefit of heatmaps is that visually encoding quantitative data with color using small area marks is very compact, so they are good for providing *overviews* with *high information density*.**

Example: Cluster Heatmaps 2/9



The keys are genes and experimental conditions, and the quantitative value attribute is the activity level of a particular gene in a particular experimental condition as measured by a microarray.

The heatmap uses a diverging red-green colormap, a common but not necessary good approach (i.e., color blindness) in the genomics domain.

Example: Cluster Heatmaps 3/9

- **The area marks in a heatmap are often several pixels on a side for easy distinguishability.**
- **The scalability limits are hundreds of levels for each of the two categorical key attributes.**
- **On the other hand, only a small number of different levels of the quantitative attribute can be distinguishable, because of the limits on color perception in small noncontiguous regions.**
- **Usually, only 3 to 11 bins can be used.**

Example: Cluster Heatmaps 4/9

- The **cluster heatmap** idiom combines the basic heatmap with **matrix reordering**, where two attributes are reordered in combination.
- The goal of matrix reordering is to group similar cells in order to check for large-scale patterns between both attributes.

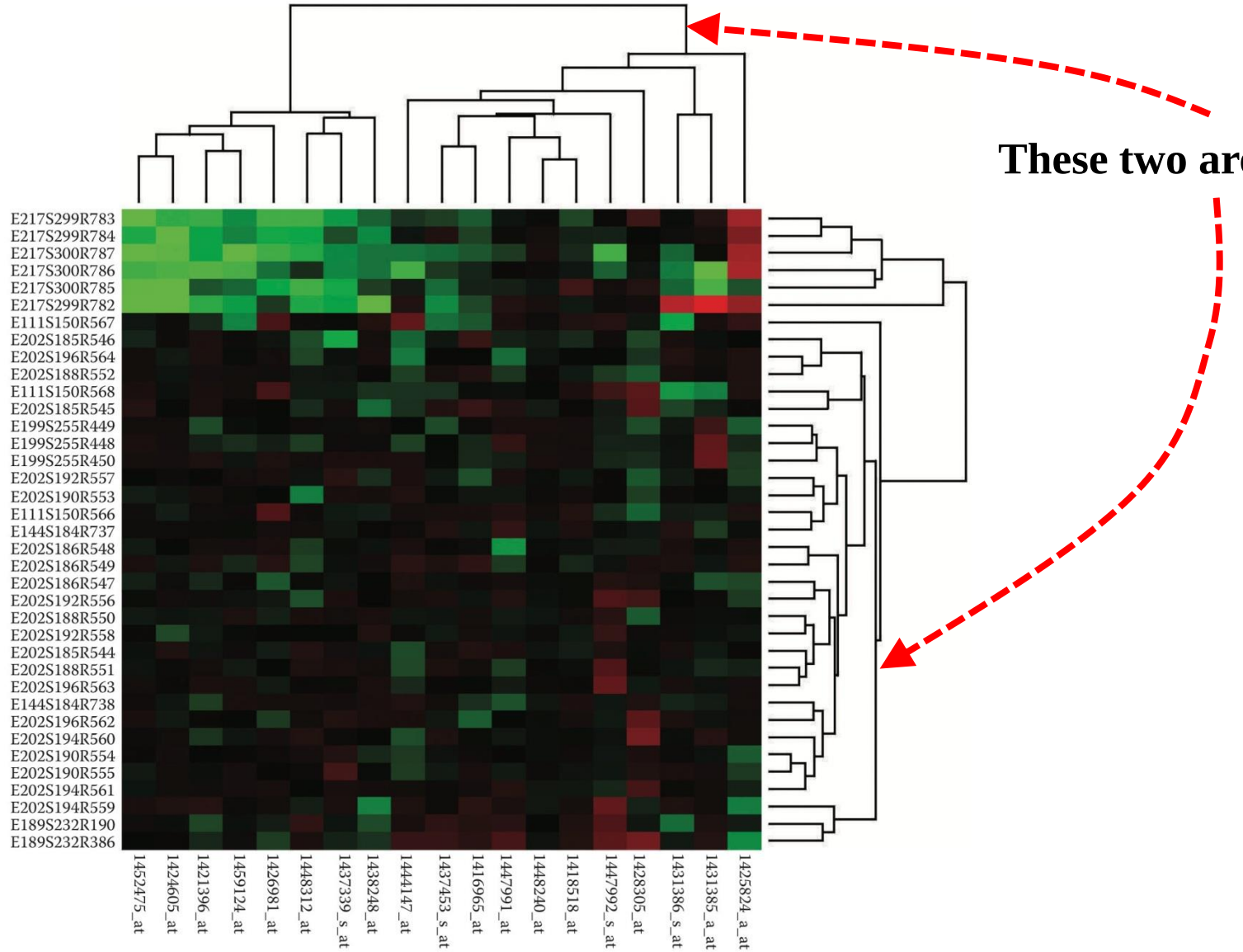
Example: Cluster Heatmaps 5/9

- A **cluster heatmap** is the juxtaposed combination of a heatmap and two dendrograms showing the derived data of the cluster hierarchies used in the reordering.
- A **dendrogram** (from Greek *dendro* "tree" and *gramma* "drawing") is a tree diagram used to illustrate the arrangement of the clusters produced by hierarchical clustering.

Example: Cluster Heatmaps 6/9

- The **cluster hierarchy** encapsulates the complete history of how a clustering algorithm operates iteratively.
- Each leaf represents a cluster of a single item; the interior nodes record the order in which clusters are merged together based on similarity, with the root representing the single cluster of all items.

Example: Cluster Heatmaps 7/9



These two are dendrograms

Example: Cluster Heatmaps 8/9

Idiom	Heatmaps
Data Type	Table: Two categorical key attributes (genes, conditions), one quantitative value attribute (activity level)
Encoding	2D Matrix alignment of area marks, diverging colormap
Why Task	Find clusters, outliers; summarize
Scale	Items: as high as one million. Categorical levels: hundreds. Quantitative attribute levels: 3-11.

Example: Cluster Heatmaps 9/9

Idiom	Heatmaps
Derived Data Type	Two hierarchies for table rows and columns
Encoding	Heatmap: 2D Matrix alignment, ordered by both cluster hierarchies. Dendrogram: connection line marks for parent-child relationships in tree

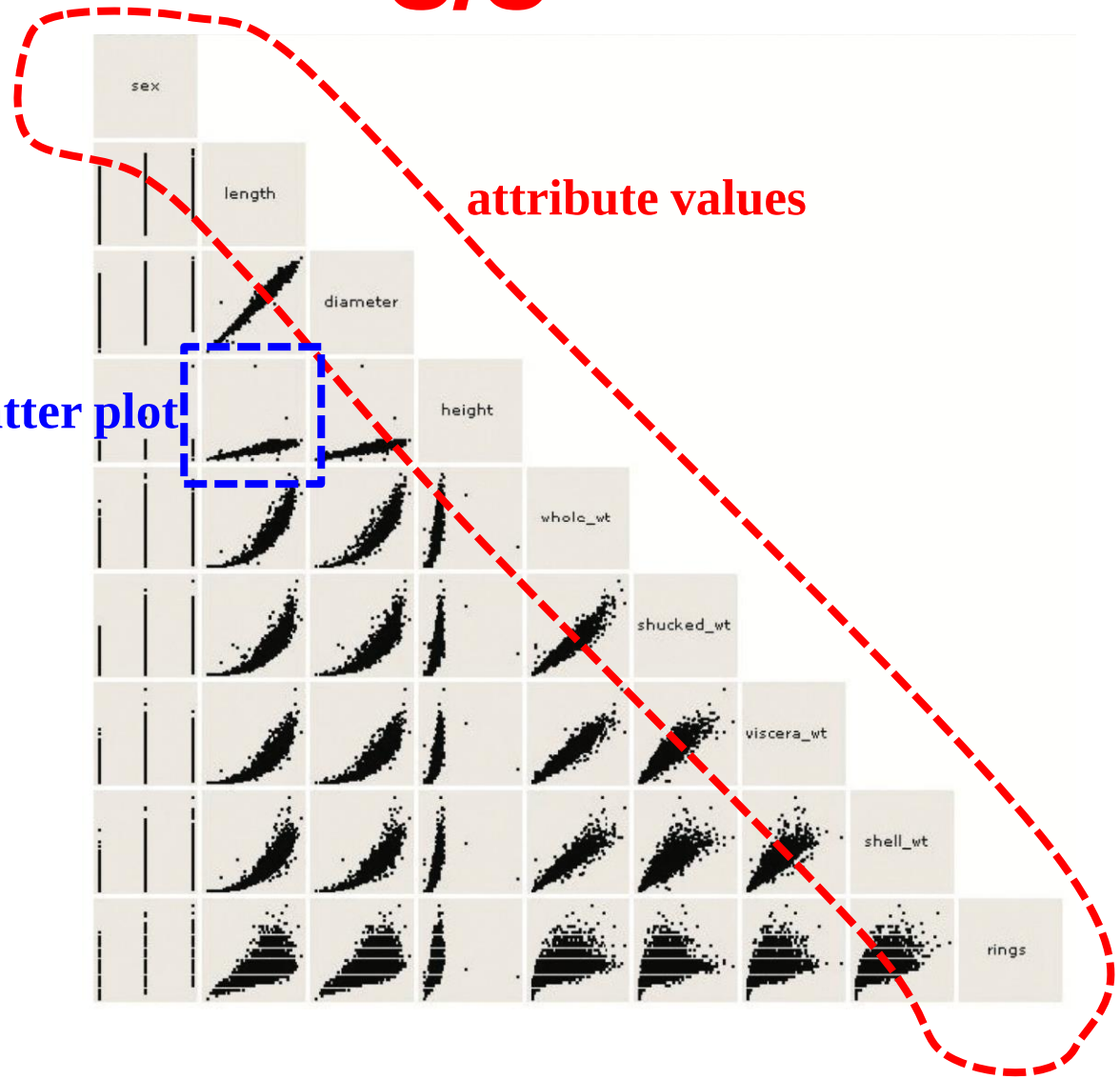
Example: Scatter Plot Matrix 1/5

- A **scatter plot matrix** (**SPLOM**) is a matrix where each cell contains an entire scatter plot, with the original attributes as the rows and columns.
- A **SPLOM** is an example of a more complex matrix where each cell shows a complete chart.

Example: Scatter Plot Matrix 2/5

- **The key is a simple derived attribute that is the same for both the rows and the columns: the index listing of all the attributes in the original dataset.**
- **The matrix may also be reordered according to any ordered attribute.**
- **Usually only the lower or upper triangle of the matrix is shown due to symmetry, and the diagonal cells are also omitted.**

Example: Scatter Plot Matrix 3/5



Example: Scatter Plot Matrix 4/5

- **SPLOM**s are used for the abstract tasks of *finding correlations*, *trends*, and *outliers*, in keeping with the usage of their constituent scatter plot components.
- Each scatter plot cell in the matrix requires enough room to plot a dot for each item discernibly. Thus, the scalability is limited to dozen attributes and hundreds items.

Example: Scatter Plot Matrix 5/5

Idiom	Scatter Plot Matrix
Data Type	Table
Derived Data	Ordered key attributes: list of original attributes
Encoding	Scatter plots in 2D matrix alignment
Why Task	Find correlations, trends, outliers
Scale	Attributes: one dozen Items: Dozens to hundreds

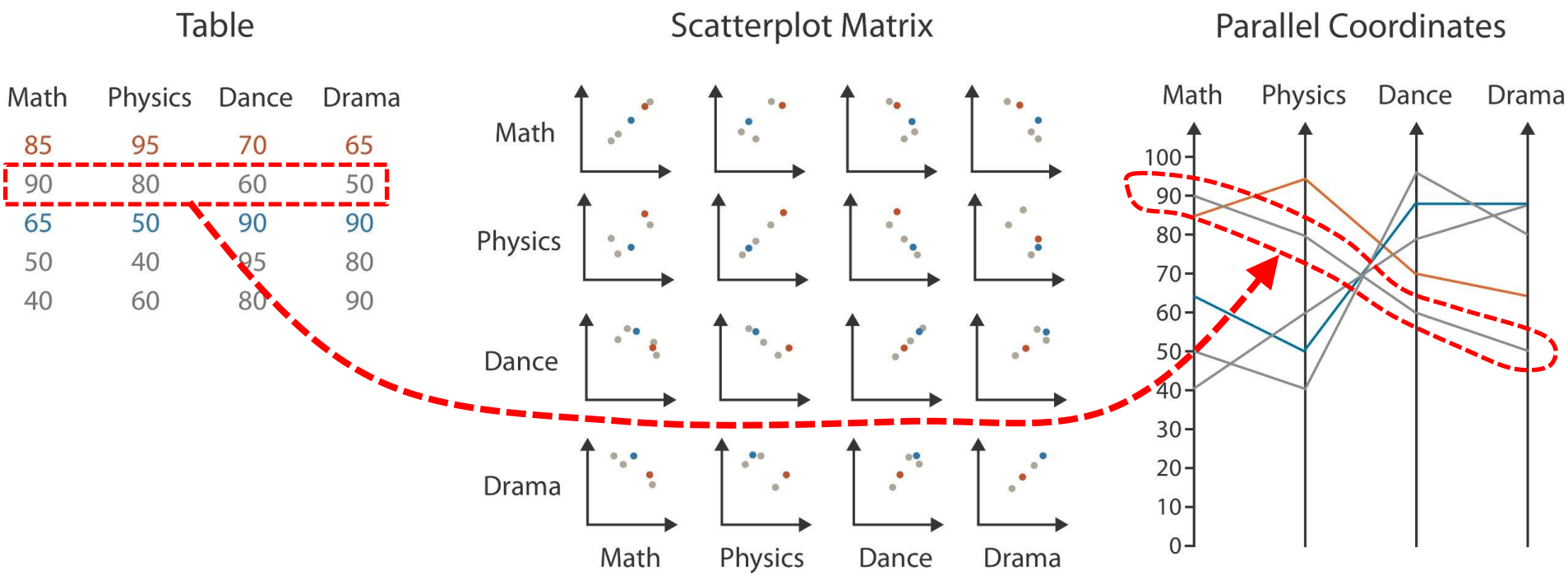
Spatial Axis Orientation

- An additional design choice with the use of space is how to orient the spatial axes: **rectilinear**, **parallel** or **radial**.
- A **rectilinear** layout is easy, just the horizontal and vertical axes as we are used to so far.
- Commonly seen arrangements are
  - **Parallel** Layout: Parallel Coordinates
  - **Radial** Layouts: Radial bar charts, pie charts

Example: Parallel Coordinates 1/10

- **The idiom of parallel coordinates is an approach for visualizing many quantitative attributes at once using spatial position.**
- **The coordinate axes are no more perpendicular to each other. Instead, they are *parallel* to each other as the name suggests.**

Example: Parallel Coordinates 2/10

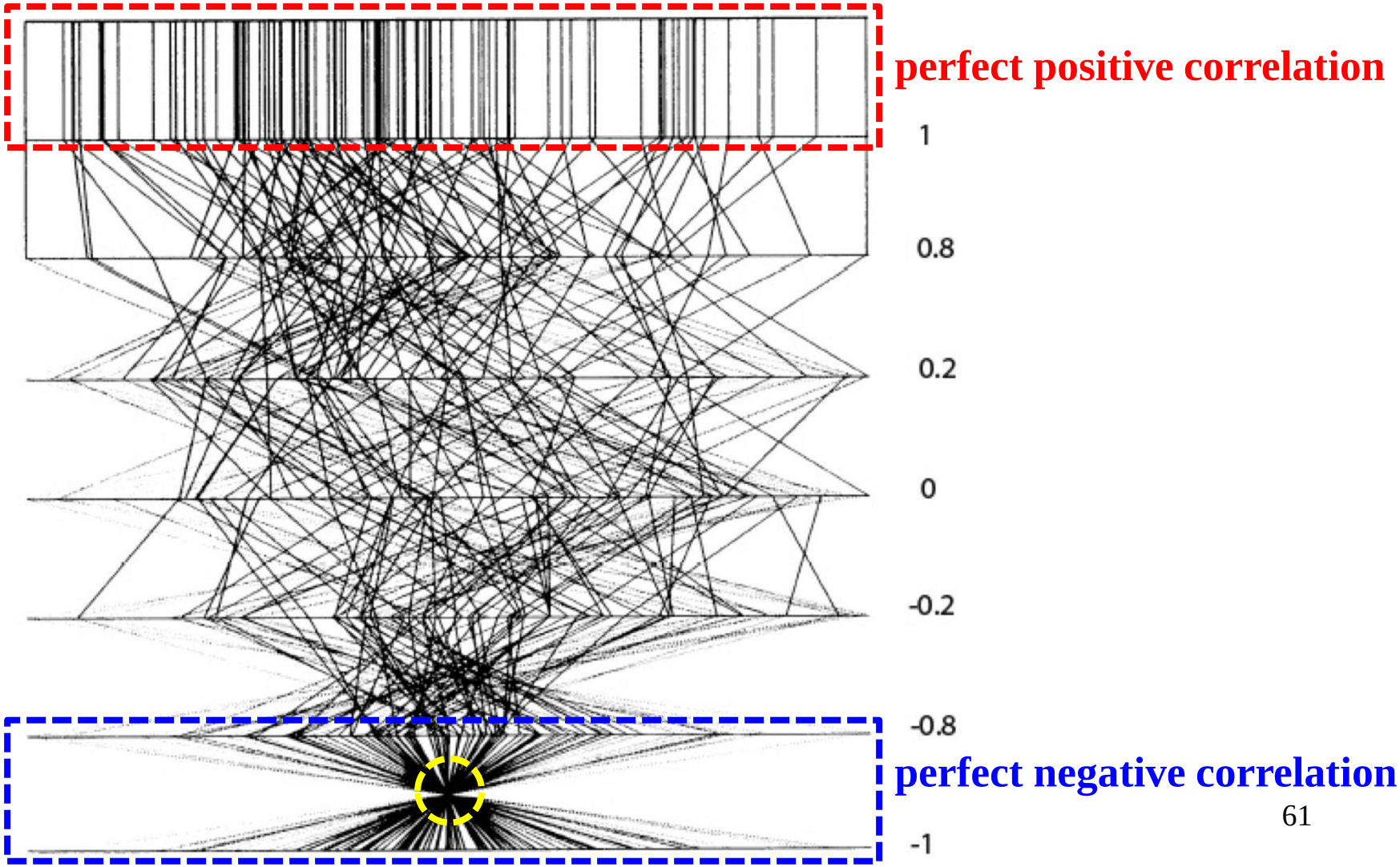


Each row is considered a coordinate in 4D space.
(90,80,60,50) is recorded as a polyline.

Example: Parallel Coordinates 3/10

- **One original motivation of parallel coordinates was that they can be used for the abstract task of checking for *correlation* between attributes.**
- **If two neighboring axes have high *positive correlation*, the line segments are mostly *parallel*.**
- **If two axes have high *negative correlation*, the line segments mostly *cross over each other* at a single spot between axes.**

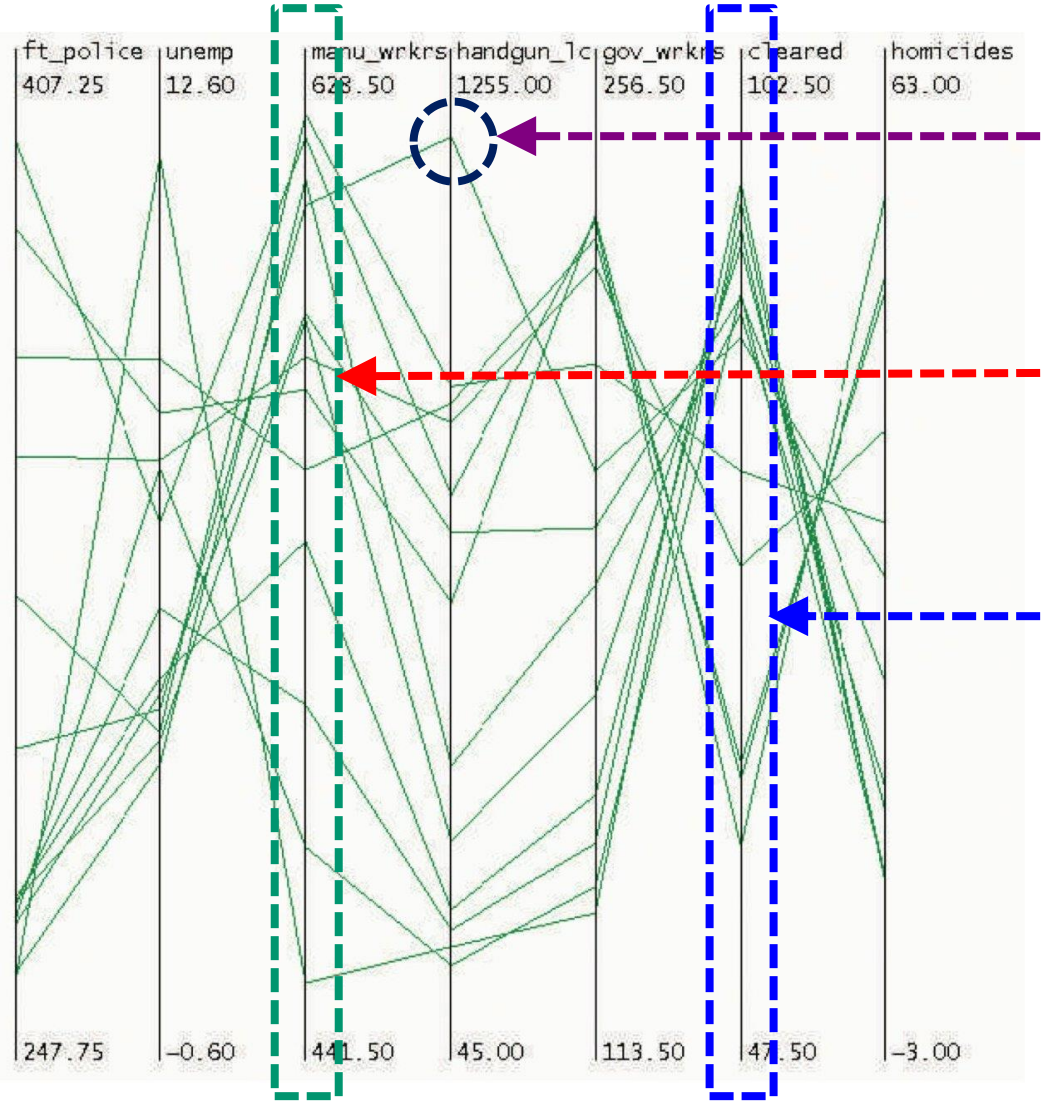
Example: Parallel Coordinates 4/10



Example: Parallel Coordinates 5/10

- **In practice, SPLOMs are typically for findings correlation between attribute pairs easily.**
- **Parallel coordinates are more often used for other tasks, such as *overview* over all attributes, finding the *range of individual attributes*, selecting a *range of items*, and *outlier* detection.**

Example: Parallel Coordinates 6/10



maybe an outlier of that attribute

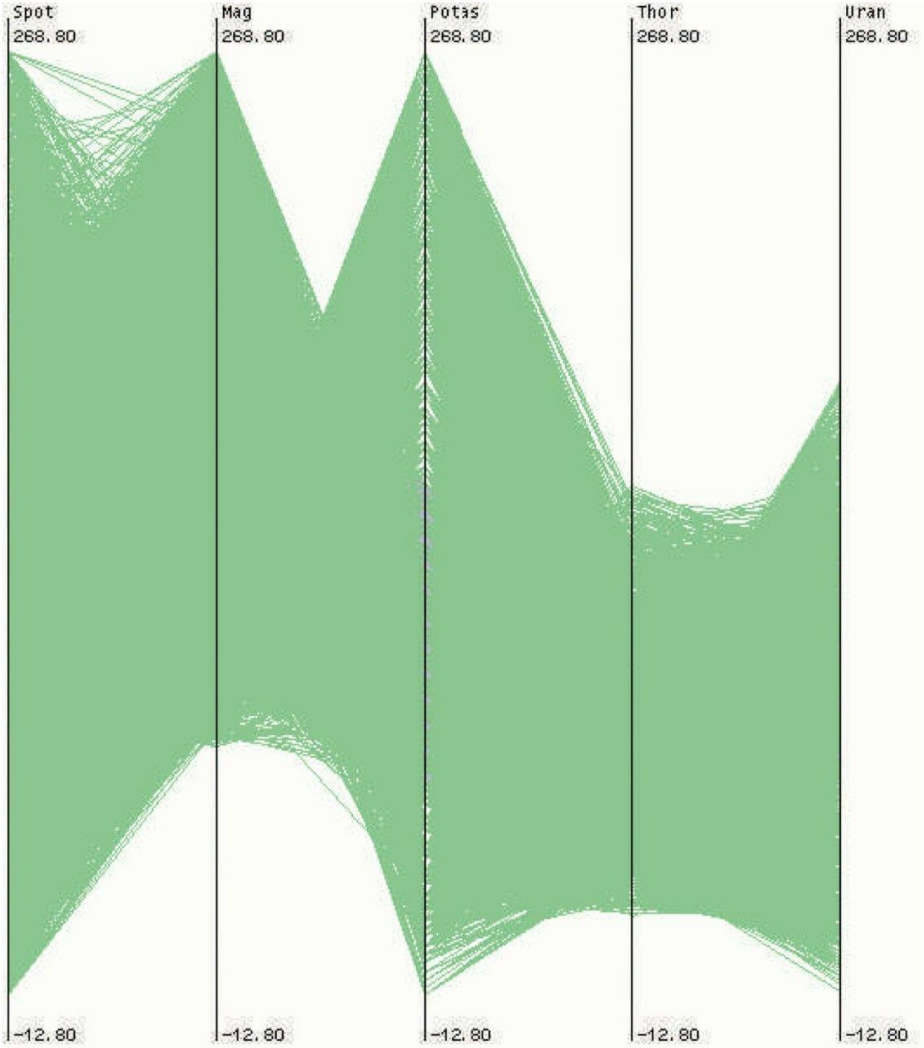
a broad range

a narrower range

Example: Parallel Coordinates 7/10

- **Parallel coordinates visually encode data using two dimensions of spatial position.**
- **The scalability is high in terms of the *number of quantitative attribute values* that can be discriminated because of high-precision channel of planar spatial position.**
- **The scalability is moderate in terms of *number of attributes* that can be displayed.**

Example: Parallel Coordinates 8/10



The left plot has 5 attributes and more than 16000 items in each attribute.

Only the minimum and maximum values can be read easily.

Trends, correlations, outliers can not be seen easily.

The transparency channel may be used to reduce this occlusion problem.

Example: Parallel Coordinates 9/10

- **The patterns easily visible by parallel coordinates are those pairwise relationships between neighboring axes.**
- **Therefore, how to determine the order of the axes is a limitation of using parallel coordinates.**
- **Beginners may not have intuitions about the meaning of the patterns they see, which must be taught explicitly.**
- **The combination of more familiar views (e.g., scatter plots) can be helpful.**

Example: Parallel Coordinates 10/10

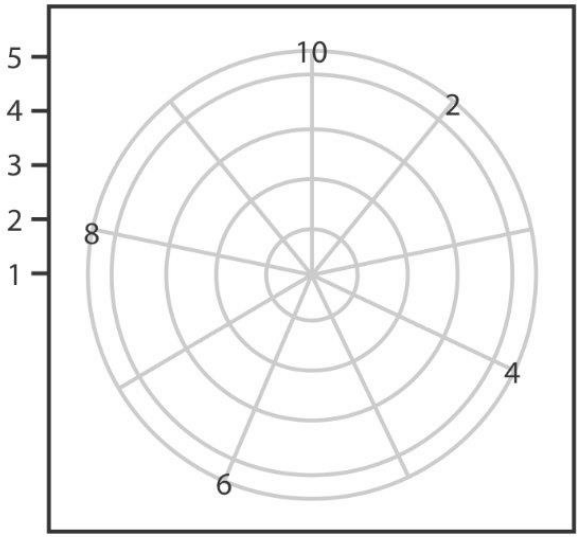
Idiom	Parallel Coordinates
Data Type	Table: many value attributes
Encoding	Parallel layout: horizontal spatial position used to separate axes, vertical spatial position used to express value along each aligned axis with connection line marks as segments between them
Why Task	Find extremes, correlations, trends, outliers
Scale	Attributes: dozens along secondary axis Items: hundreds

Radial Layouts: 1/4

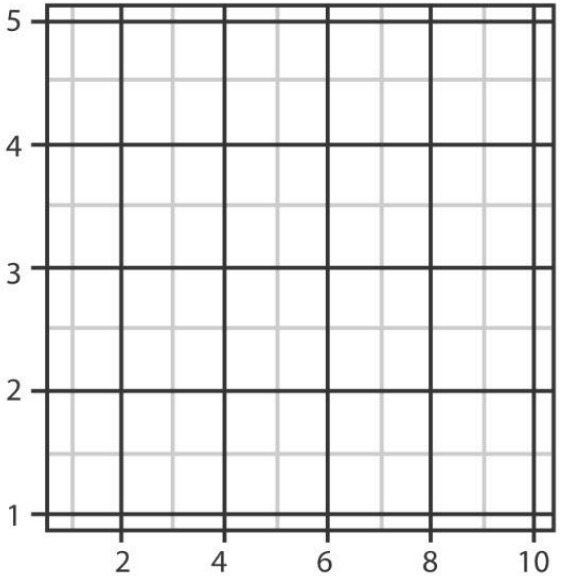
- In a **radial** spatial layout, items are distributed around a circle using the angle channel in addition to one or more linear spatial channels.
- The rectilinear layouts only use two spatial channels.
- A natural coordinate system in radial layouts is the polar coordinates, where one dimension is measured as an angle from a starting line and the other is measured as a distance from a center point.

Radial Layouts: 2/4

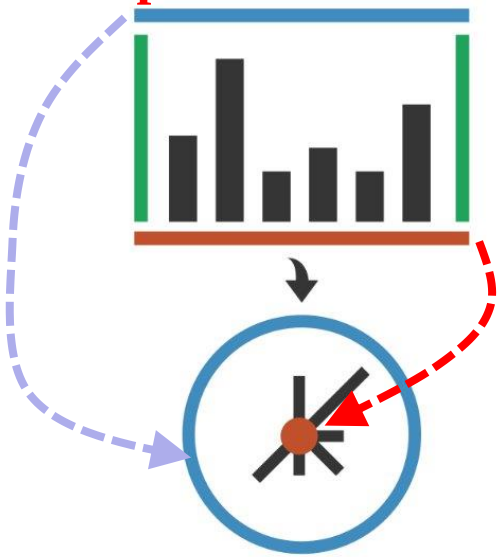
Polar Coordinates



Rectilinear Coordinates



Transforming rectilinear to radial layouts maps two parallel lines to a point at the center and a circle at the perimeter



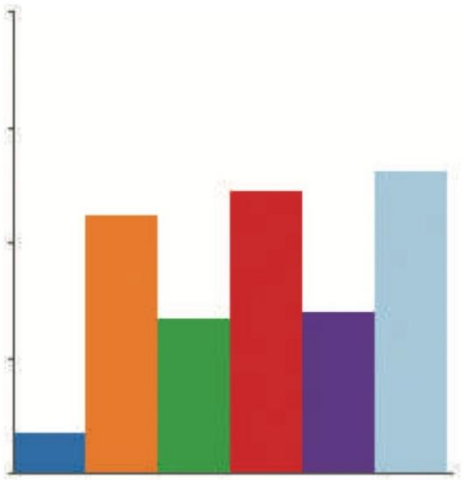
Radial Layouts: 3/4

- **Rectilinear and radial layouts are not equivalent perceptually.**
- **The change of visual channel has two major consequences:**
  - **A. The angle channel is less accurately perceived than a rectilinear spatial position channel.**
  - **B. The angle channel is inherently cyclic, because the start and end points are the same, as oppose to the inherently linear nature of a position channel.**

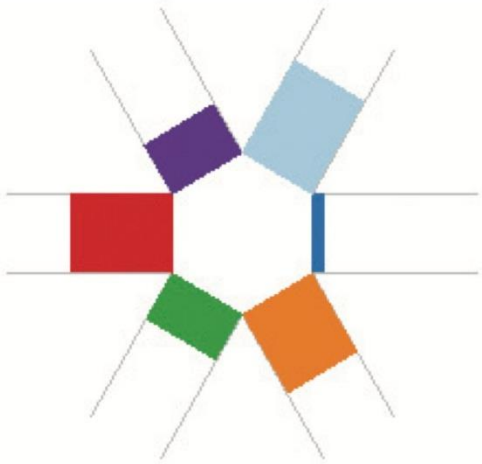
Radial Layouts: 4/4

- **The expressiveness and effectiveness principles suggest some guidelines:**
  - **A. Radial layouts may be more effective than rectilinear ones in showing the *periodicity* of patterns, but encoding nonperiodic data with the periodic channel of angle may be misleading.**
  - **B. Radial layouts imply an *asymmetry* of importance between two attributes and would be inappropriate when the two attributes have equal importance.**

Example: Radial Bar Charts 1/1



Bar Chart



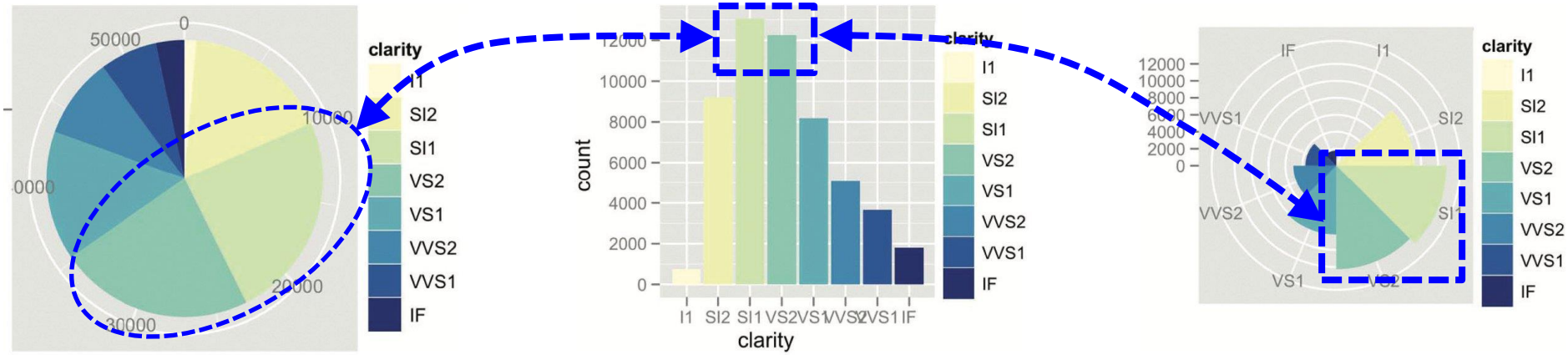
Radial Bar Chart

Idiom	Radial Bar Charts
Data Type	Table: one quantitative attribute, one categorical attribute
Encoding	Length coding of line marks; radial layout

Example: Pie Charts 1/8

- **Pie charts** encode a single attribute with *area* marks and the *angle* channel.
- *Angle* judgments on *area* marks are less accurate than *length* judgments on *line* marks.
- The wedges vary in width along the radial axis, from narrow near the center to wide near the outside, making the area judgment particularly difficult.

Example: Pie Charts 2/8



Pie Chart Bar Chart Polar Area Chart

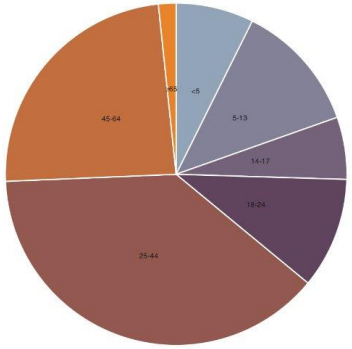
Which of these is perceptually better?

- **The polar area chart also encodes a single quantitative attribute, but varies the length of the wedge just as a bar chart varies the length of the bar, rather than varying the angle as in a pie chart.**
- **Each mark is encoded with color for easier legibility, but these idioms could be used without color encoding.**

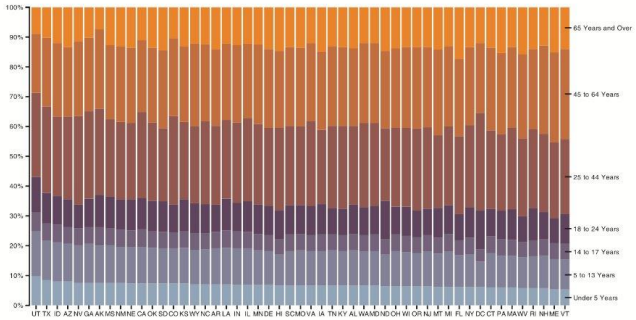
Example: Pie Charts 3/8

- **The most useful property of pie charts is to show the *relative contribution* of parts to a whole.**
- **However, this property is not unique to pie charts: a single bar in a normalized stacked bar chart can also be used with the more accurate length channel.**
- **A normalized stacked bar chart stretches each bar to the maximum, showing the percentages rather than absolute counts.**

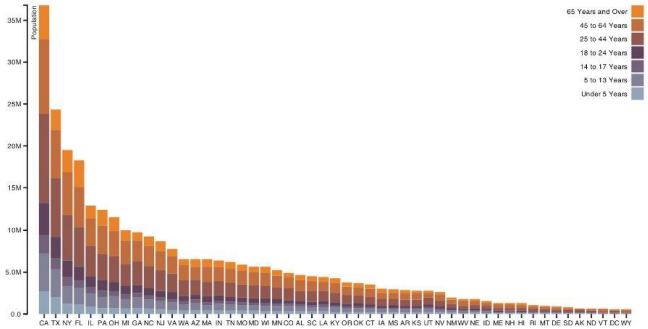
Example: Pie Charts 4/8



Aggregate Population of 50 states



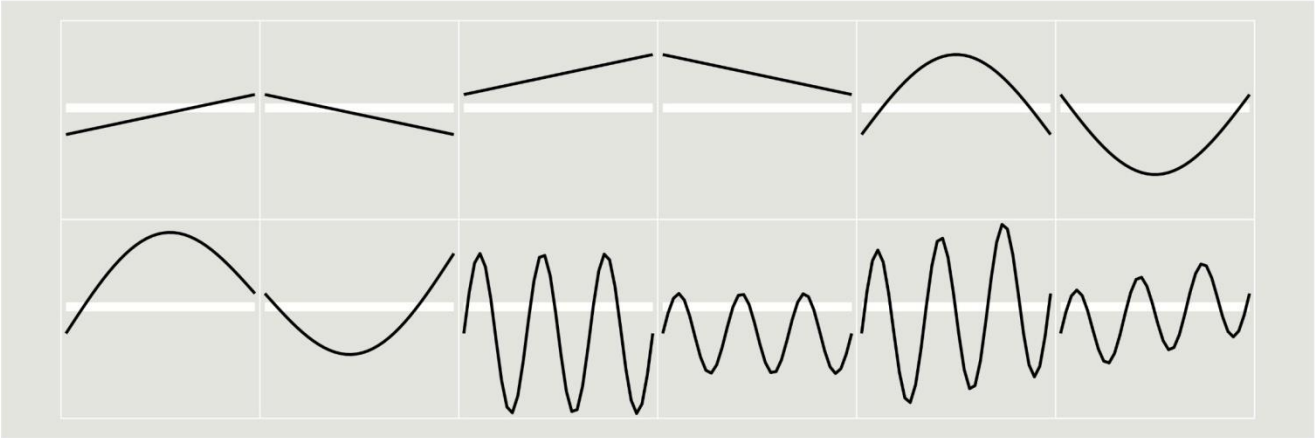
Population of 50 states Normalized Stacked Bar Relative Values



Population of 50 states Stacked Bar Chart Absolute Counts

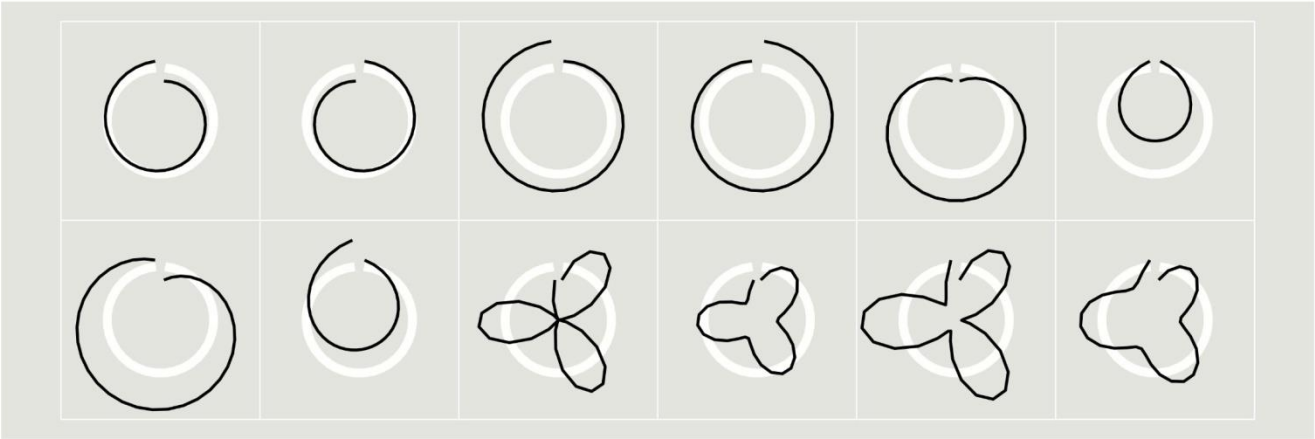
- **Pie charts require somewhat more screen area than normalized stacked bar charts because the angle channel has lower precision than the length channel.**
- **The aspect ratio also differs, where a pie chart requires a square, whereas a bar chart requires a rectangle.**

Example: Pie Charts 5/8



12 iconic time series dataset in rectilinear layout

A study shows that rectilinear layouts outperformed radial layout:
fast perception speed and better accuracy.



12 iconic time series dataset in radial layout

When one attribute is more important than the other, the use of radial layouts is justified .

Example: Pie Charts 6/8

Idiom	Pie Charts
Data Type	Table: one quantitative attribute, one categorical attribute
Encoding	Area marks (wedges) with **angle** channel; radial layout
Why Task	Part-Whole relationship
Scale	One dozen categories

Example: Pie Charts 7/8

Idiom	Polar Area Charts
Data Type	Table: one quantitative attribute, one categorical attribute
Encoding	Area marks (wedges) with **length** channel; radial layout
Why Task	Part-Whole relationship
Scale	One dozen categories

Example: Pie Charts 8/8

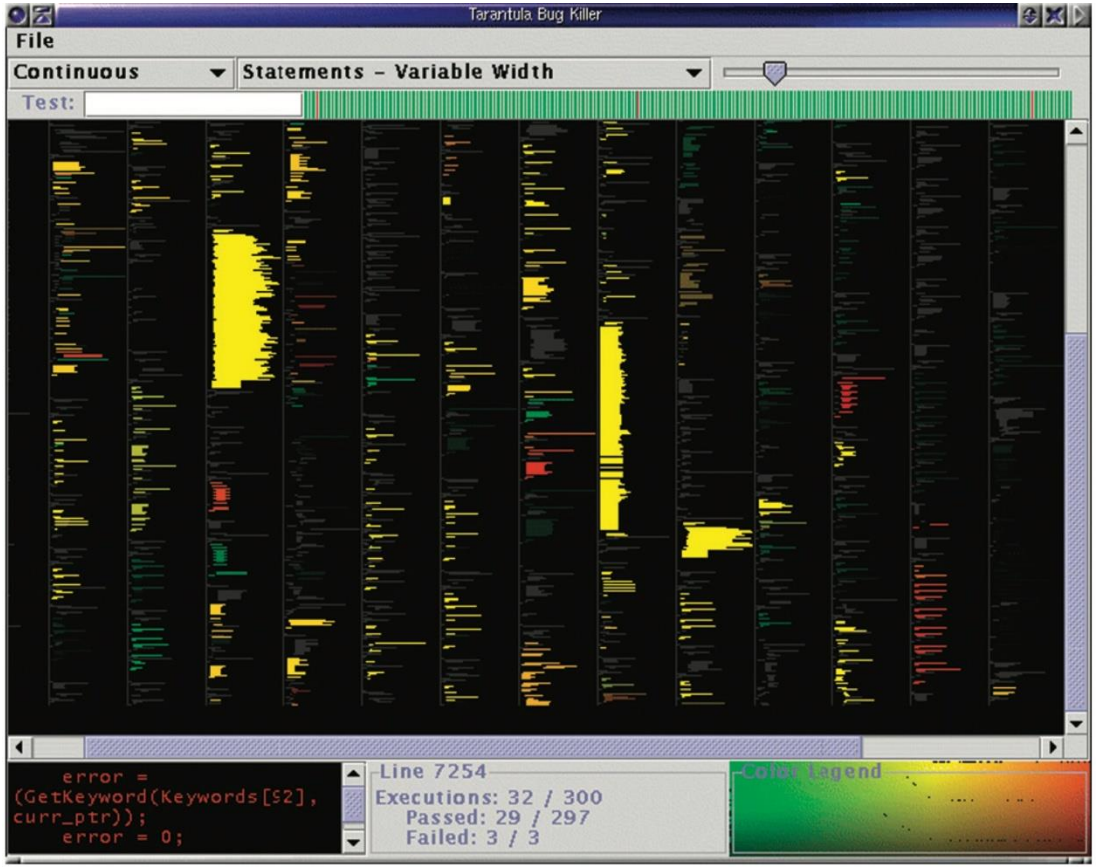
Idiom	Normalized Stacked Bar Charts
Data Type	Multidimensional Table: one quantitative value attribute, two categorical key attributes
Derived Data Type	One quantitative value attribute (normalized version of original attribute)
Encoding	Line marks with **length** channel; rectilinear layout
Why Task	Part-Whole relationship
Scale	One dozen categories for stacked attributes. Several dozen categories for axis attribute

Spatial Layout Density: 1/3

- **Another design choice with spatial visual encoding is whether a layout is *dense* or *sparse*.**
- **A related, but not identical, choice is whether a layout is *space-filling*.**
- **A dense layout uses small and densely populated marks to provide an overview of as many items as possible with very high information density.**
- **A maximally dense layout has point marks that are only a single pixel is size and line marks that are a single pixel in width.**

Spatial Layout Density: 2/3

- **Tarantula is a software engineering tool for visualizing test coverage.**



Tarantula shows a dense overview of source code with lines colored by execution status of a software suite.

Spatial Layout Density: 3/3

- A *space-filling* layout has the property that it fills all available space in the view.
- Space-filling layouts typically use area marks for items or containment marks for relationships, rather than line of connection marks.
- One advantage of space-filling approaches is that they maximize the amount of room available for color coding, increasing the chance that the colored regions will be large enough.
- One disadvantage is that the design cannot make use white space in the layout.

The End