Data Exploration & Visualization

Module 1

Data Model

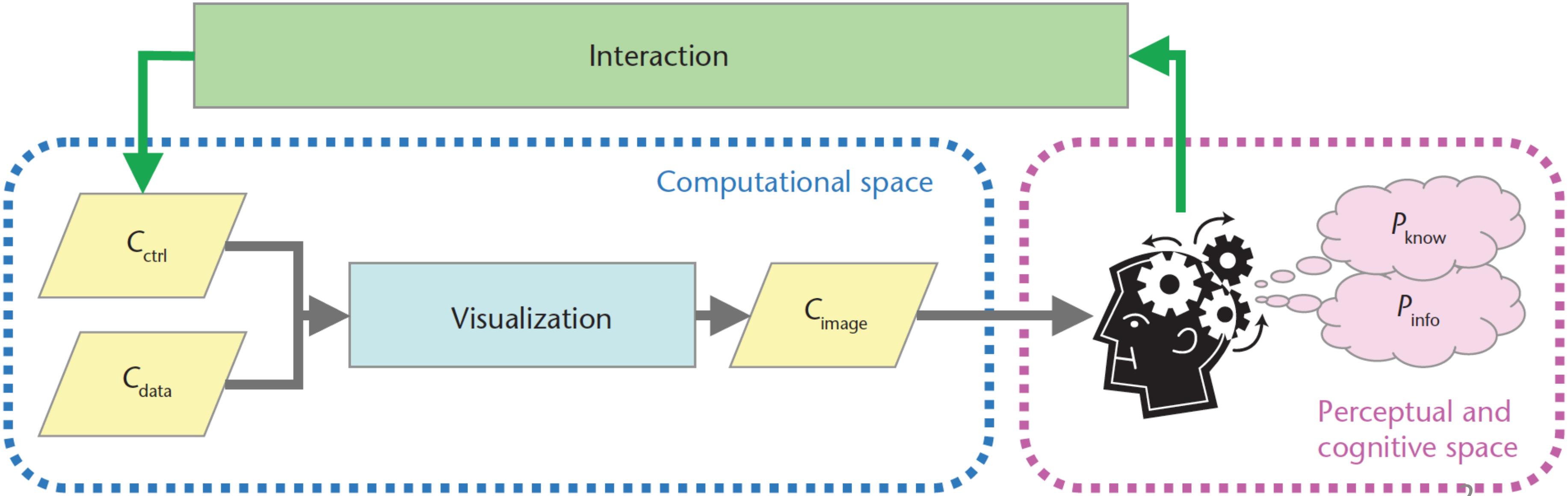
Dr. ZENG Wei

DSAA 5024

The Hong Kong University of Science and Technology (Guangzhou)

Visualization process

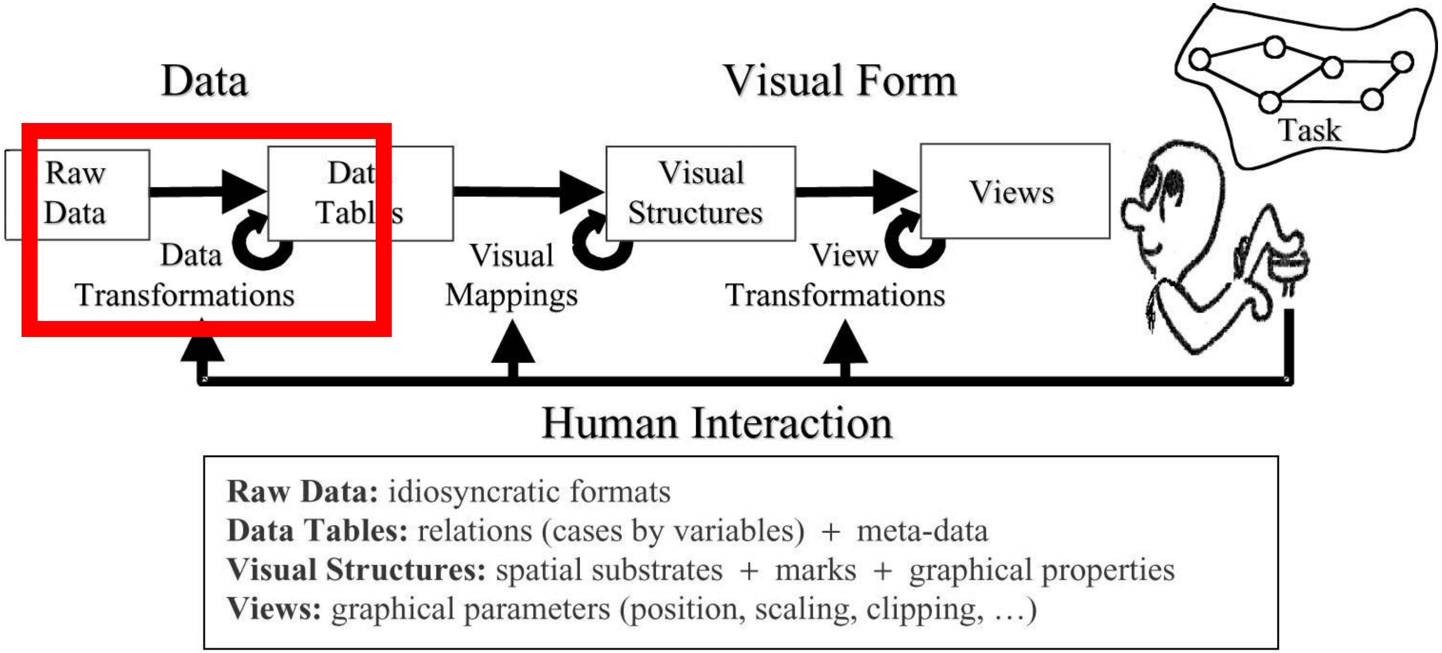
- A typical visualization process maps from *data* and *control parameters* to *images*.
  - Data: symbols
  - Control parameters: viewpoint, filter, etc.
  - Images: visual representations



Credit: Chen et al., 2009

Visualization process

- Information visualization reference model
  - Data transformations
  - Visual mappings
  - View transformations



Credit: Stuart, Mackinlay, Shneiderman, 1999

Data Exploration & Visualization

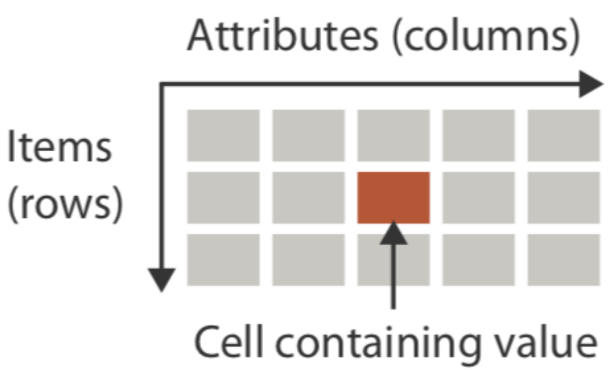
Module 1: Data Model

- Data Model
- Data Types
  - Nominal, ordinal, quantitative
- Data structure
  - Table
  - Data cube

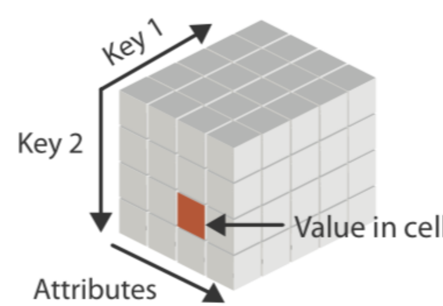
Data & dataset

- **Data** (plural) are observations or measurements represented as text, numbers, or multimedia.
  - Singular form: a datum
- A **dataset** is a structured collection of data generally associated with a unique body of work.

➔ Tables



➔ *Multidimensional Table*



Credit: Tamara Munzner

Data model

- Data model: an abstraction of how elements of a dataset relate to each other
  - This is more or less synonymous with “data structure”
- Don’t be confused with other ‘models’
  - Statistics and ML model
    - an algorithm trained with data, resulting in learned parameters or weights.
  - Workflow or analysis model
    - Abstraction of a process for a workflow or a sequence of analysis processes
  - Cognitive and mental model
    - An abstract representation of how a person thinks

Data model

- Shneiderman, 1996:
  - 1D, 2D, …, high-dimensional
  - 3D (spatial)
  - Temporal
  - Tree (Hierarchy)
  - Network (Graph)
- Keim, 2002:
  - Text
  - Algorithms/software/processes
- Others:
  - Maps, geospatial
  - Trajectory (geospatial temporal)
  - Sequences
  - Images, audio, and video (multi-media)
  - Relational
  - Sets
  - Lists (ranking)
  - Streaming data
  - :
  - :

Not a clear categorization

- There are some clear overlaps between model types
  - For example, how is ***Temporal*** data different from ***2D*** data?
  - In most cases, they *could* be the same
  - But there are technical differences, e.g.,
    - In a 2D dataset, (1.0, 1.0), (2.0, 2.0), (2.0, 3.0) is allowed, but this would be very confusing for temporal dataset

  - But what analysis would you use for temporal data that you wouldn’t use for the other (and vice versa)?

Not a clear categorization

- There are some clear overlaps between model types
  - For example, how is ***Temporal*** data different from ***2D*** data?
  - In most cases, they *could* be the same
  - But there are technical differences, e.g.,
    - In a 2D dataset, (1.0, 1.0), (2.0, 2.0), (2.0, 3.0) is allowed, but this would be very confusing for temporal dataset

  - The types of analyses would be different. The following are common for temporal data but not for general 2D data
    - For temporal data
      - Harmonic (period) analysis
      - Analysis based on (sliding) windows
    - For 2D data
      - Clustering

Choose the right data model

- The **quality of the analysis** depends on the **quality of the (data) modeling.**

- Example: Google vs. early-days of Search Engines (e.g. Yahoo)

Internet Search, the Early Days (1993 – 1997)

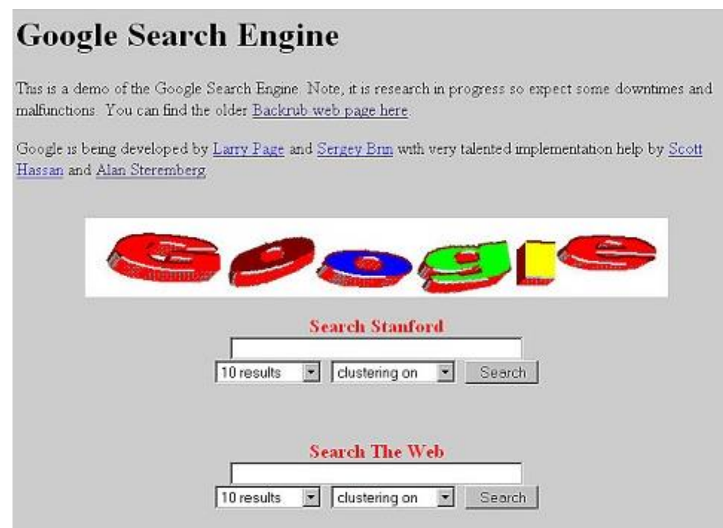
- Web pages are "modeled" as ***texts***

- What can you do with texts?
  - Frequency of words
  - TFIDF (tf-idf)
  - Sequencing of words (e.g. n-grams)

- This led to some hilarious ways that people use to increase traffic to their websites

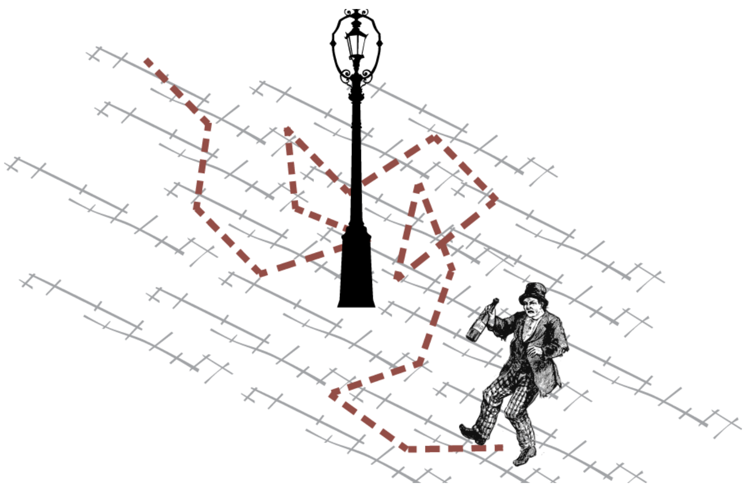
Internet Search, How Google Took Over the World

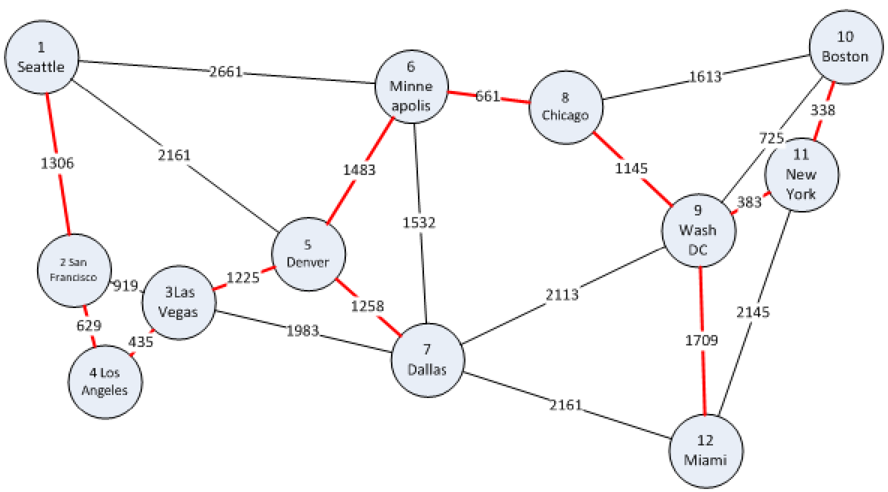
- Web pages are "modeled" as ***graphs***
- What can you do with graphs?
- Turns out, Google won the world because of the PageRank algorithm, which is essentially a measure of "eigencentrality"



Eigencentrality: as a Graph

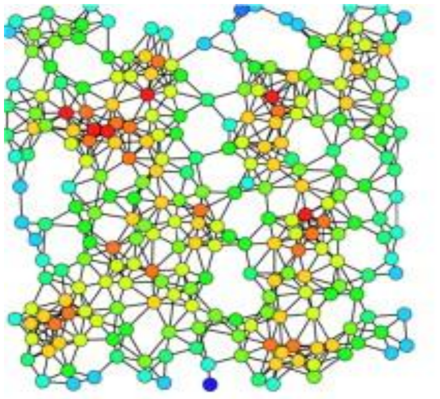
- Eigencentrality is an algorithm that aims to measure the “importance” of a node in a graph
- In the graph sense, Eigencentrality can be thought of as “taking infinite random walks”



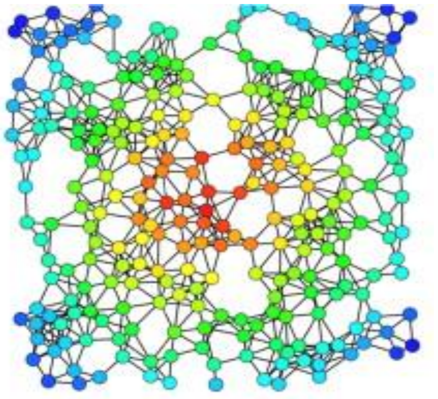


Eigencentrality: as a Matrix

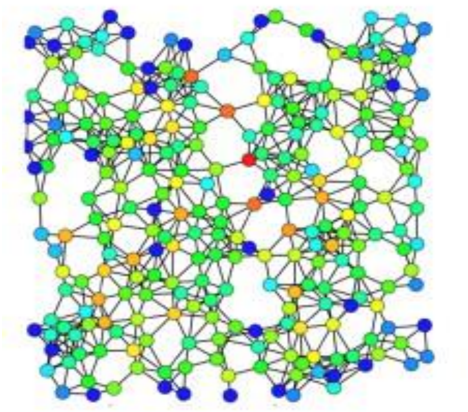
- Turns out that taking "infinite random walks" on the internet is not practically computable.
- Approach: convert a graph into a (weight) adjacency matrix and do eigen-decomposition



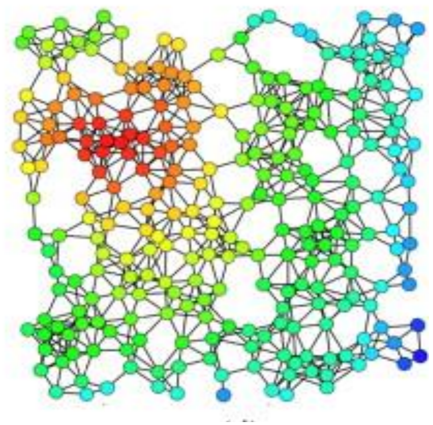
Degree Centrality



Closeness Centrality



Betweenness Centrality



Eigencentrality

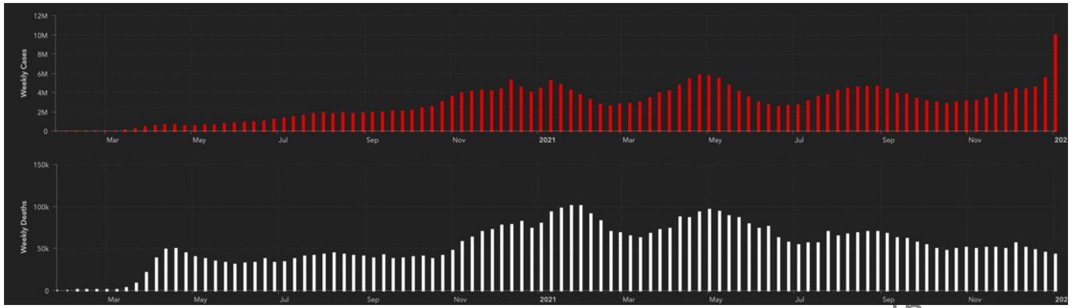
Summary

- The “modeling of data” dictates what “types” of analyses one can do.
- Modeling the data to fit the problem is arguably the most important (and therefore difficult) task in visual analytics.
- Reason is that the ***types of analyses & visualizations*** become apparent (or restricted) once the data model is decided.
    - We will learn what visualization methods are there for different data models

For example

- Consider an analysis of the spread of COVID
  - We can think about it geospatially
  - We can think about it as time series
  - We can think about it as a graph

- For your final project, do not automatically assume a data model
  - For example just because the dataset contains location information, you don't have to immediately think "maps"
  - You can consider other models/representations



Tips for the final project

- For your final project, do not automatically assume a data model!!
  - First, think about what your research hypothesis is
  - Second, think about how you would need to test the hypothesis
  - Third, think about whether your data model and the associated data analysis methods are appropriate
  - Lastly, think about how you can design appropriate visualizations to help a user interact with the data and the model

Example: Interchange pattern

- ***interchange patterns*** in human movements.
  - how moving objects redistribute when passing through a junction node in the network
- Interchange pattern happens in our daily lives
  - Road junctions, subway stations, etc.

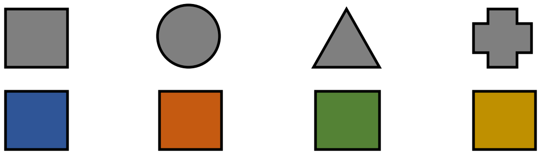
Data Exploration & Visualization

Module 1: Data Model

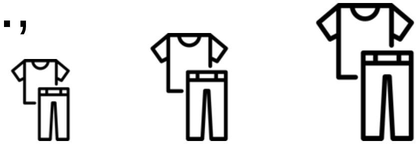
- Data Model
- Data Types
  - Nominal, ordinal, quantitative
- Data structure
  - Table
  - Data cube

Data types

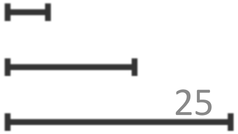
- There are different types of data attribute
  - Nominal/categorical: an unordered set, e.g.,
    - {“John Smith”, “Jane Doe”, …}
    - {Apple, orange, pear}
    - {Red, blue, green}



  - Ordinal: an ordered set (a tuple), e.g.,
    - Numerical <0, 1, 2, 3>
    - non-numerical <S, M, L>



  - Quantitative
    - Interval: ordered numeric elements that can be mathematically manipulated, but not compared as ratios, e.g.,
      - Calendar dates, current time
    - Ratio: where there exists an absolute zero e.g.,
      - length, temperature



Example: Iris sample

- Many of the exploratory data techniques are illustrated with the Iris flower data set
  - Can be obtained from the UCI ML Repository https://archive.ics.uci.edu/ml/datasets/Iris
  - Introduced by statistician and biologist Douglas Fisher
  - Three flower types (classes)
    - Setosa
    - Virginica
    - Versicolour
  - Four attributes
    - Sepal (萼片) width and length
    - Petal (花瓣) width and length

Data attribute

Sepal Length	Sepal Width	Petal Length	Petal Width	Species
	…	….	…..	
4.9	3.0	1.4	0.2	I. setosa
4.7	3.2	1.3	0.2	I. setosa
	…	….	…..	
5.7	2.8	4.5	1.3	I. versicolor
6.3	3.3	4.7	1.6	I. versicolor
	…	….	…..	
6.1	2.6	5.6	1.4	I. virginica
6.3	3.4	5.6	2.4	I. virginica
	…	….	…..	

Quantitative

Categorical

Attribute types

Value	Format	Attribute types
Setosa	String	Categorical
Virginica	String	Categorical
6.3	Numeric	Quantitative
3.0	Numeric	Quantitative

- Does data format determine attribute type?
  - No. Zip code (e.g., 100049, 518055) is stored in numeric format, but it is categorical attribute.
- What matters?
  - What kinds of mathematical operations are meaningful for it.
  - Categorical → separate, identify…
  - Ordered → compare, sort… quantitative → addition, scale…

Attribute types

- The type of a data attribute depends on which of the following properties it possesses:
  - Distinctness: $= \quad \neq$
  - Order: $< \quad >$
  - Addition: $+ \quad -$
  - Multiplication: $\times \quad \div$

  - Categorical attribute:
  - Ordinal attribute:
  - Interval attribute:
  - Quantitative (ratio) attribute:

Attribute types

Attribute type	Description	Operations
Nominal	The values of a nominal attribute are just different names, i.e., nominal attributes provide only enough information to distinguish one object from another. (=, ≠)	mode, entropy, contingency correlation, χ^2 test
Ordinal	The values of an ordinal attribute provide enough information to order objects. (<, >)	median, percentiles, rank correlation, sign tests
Interval	The differences between values are meaningful, i.e., a unit of measurement exists. (+, −)	mean, standard deviation, Pearson‘s correlation, t and F tests
Quantitative (ratio)	Both differences and ratios are meaningful. (×, /)	geometric mean, harmonic mean, percent variation

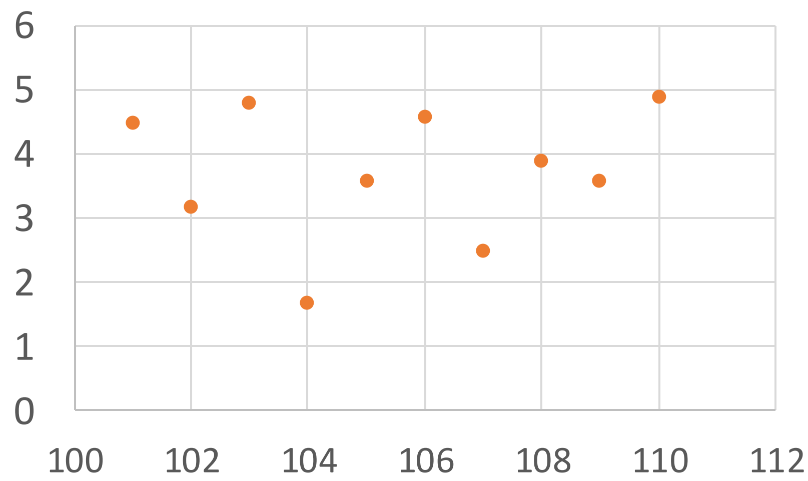
What is the attribute type?

- ID: 3429 2342, 3452 3234, …
- Zip code: 495214, 454245, …
- Grade in A, A-, B+, B, …
- Grade in 100, 94, 45, …

Name	Student ID	GPA
Ana	101	4.5
Bob	102	3.2
Cindy	103	4.8
Dider	104	1.5
…	…	…

Given the above table of student name, ID, and GPA, an analyst tried to examine the correlation between student ID and GPA. He draw the scatterplot as below.

What went wrong?



COMMENTARIES

Commentaries are informative essays dealing with viewpoints of statistical practice, statistical education, and other topics considered to be of general interest to the broad readership of *The American Statistician*. Commentaries are similar in spirit to Letters to the Editor, but they involve longer discussions of background, issues, and perspectives. All commentaries will be refereed for their merit and compatibility with these criteria.

Nominal, Ordinal, Interval, and Ratio Typologies Are Misleading

PAUL F. VELLEMAN and LELAND WILKINSON*

The psychophysicist S. S. Stevens developed a measurement scale typology that has dominated social statistics methodology for almost 50 years. During this period, it has generated considerable controversy among statisticians. Recently, there has been a renaissance in the use of Stevens's scale typology for guiding the design of statistical computer packages. The current use of Stevens's terminology fails to deal with the classical criticisms at the time it was proposed and ignores important developments in data analysis over the last several decades.

KEY WORDS: Data analysis; Data types; Measurement scales; Scaling.

In the early 1940s, the Harvard psychologist S. S. Stevens coined the terms *nominal*, *ordinal*, *interval*, and *ratio* to describe a hierarchy of measurement scales used in psychophysics, and classified statistical procedures according to the scales for which they were "permissible." This taxonomy was subsequently adopted by several important statistics textbooks and has thus influenced the statistical reasoning of a generation. Although criticized by statisticians, Stevens's categories still persist in some textbooks.

Recent interest in artificially intelligent computer programs that automate statistical analysis has renewed attention to Stevens's work. Computer programs designed to assist in the selection of data analysis methods have been based on his prescriptions. Even some general-purpose programs have used them to structure their interaction with the user.

Unfortunately, the use of Stevens's categories in selecting or recommending statistical analysis methods is inappropriate and can often be wrong. They do not describe the attributes of real data that are essential to good statistical analysis. Nor do they provide a classification scheme appropriate for modern data analysis methods. Some of these points were raised even at the time of Stevens's original work. Others have become clear with the development of new data analysis philosophies and methods.

In the following sections, we review Stevens's taxonomy and provide definitions; many have used these terms without clarifying their exact meaning. We discuss their use in statistics and in applications, and consider some of the classical criticisms of this work. Throughout our account, we provide references for interested readers who may wish to learn more. We then describe some of the failures of Stevens's taxonomy to classify data, and examine the nature of these failures. Similarly, we consider whether modern statistical methods can be classified according to the types of data appropriate for them. Finally, we consider what ideas from Stevens's work are still useful for modern computer-based statistical analysis.

1. STEVENS'S TYPOLOGY OF DATA

In his seminal paper, "On the Theory of Scales of Measurement" (1946), Stevens presented a hierarchy of data scales based on invariance of their meaning under different classes of transformations. Measurement scales that preserve meaning under a wide variety of transformations in some sense convey less information than those whose meaning is preserved by only a restricted class of transformations. For example, assume a scale, s, is used to assign real numbers in $\Re$ to the elements of a set, P, of observed judgments so that for all i and j in P, $s(i) > s(j)$ iff i is preferred to j. That is, if we let the symbol "$\rangle$" stand for "is preferred to," then

$$P \xrightarrow{s} \Re \quad \text{such that}$$
$$i \rangle j \leftrightarrow s(i) > s(j), \quad \text{for all } i, j \in P. \qquad (1)$$

*Paul F. Velleman is Associate Professor, Department of Economic and Social Statistics, Cornell University, Ithaca, NY 14851 and President, Data Description, Inc. Leland Wilkinson is President, SYSTAT, Inc., Evanston, IL 60201 and Adjunct Professor, Department of Statistics, Northwestern University. The authors thank Sergei Adamov, Ingwer Borg, Laszlo Engelman, Pat Fleury, David Hoaglin, and John Tukey for helpful comments.

Data Exploration & Visualization

Module 1: Data Model

- Data Model
- Data Types
  - Nominal, ordinal, quantitative
- Data structure
  - Table
  - Data cube

Data structure

- A **data structure** is a particular way of organizing data in a computer so that it can be used effectively.
  - Array, list, tree, stack, queue, etc.
  - An essential topic in computer science.
- Today we learn two basic formats of data structure, from a visualization perspective
  - **Tabular data**
  - **Data cube**

Tabular data

- An attribute is a property or characteristic of an object
  - Examples: eye color of a person, temperature, etc.
  - Attribute is also known as variable, field, characteristic, or feature
- A collection of attributes describe an object
  - Object is also known as record, point, case, sample, entity, or instance

Attributes

Objects

Tid	Refund	Marital Status	Taxable Income	Cheat
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	$\boxed{\text{120K}}$	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes

Cell

Example: Iris sample

Sepal Length	Sepal Width	Petal Length	Petal Width	Species
	…	….	…..	
4.9	3.0	1.4	0.2	I. setosa
4.7	3.2	1.3	0.2	I. setosa
	…	….	…..	
5.7	2.8	4.5	1.3	I. versicolor
6.3	3.3	4.7	1.6	I. versicolor
	…	….	…..	
6.1	2.6	5.6	1.4	I. virginica
6.3	3.4	5.6	2.4	I. virginica
	…	….	…..	

Objects/ records/ items

Attributes

Example: Shenzhen taxi

- A set of facts arranged in rows and columns
  - Rows: items (GPS records)
  - Columns: attributes (taxi id, position, time, etc.)

```
粤B263YS,114.037300,22.705099,2016-01-01 00:05:11,1454778,2,0,0,,,0,蓝色,
粤B6HR48,114.107880,22.611349,2016-01-01 00:05:11,1372113,0,126,0,,,1,蓝色,
粤BL6C35,114.145485,22.716700,2016-01-01 00:04:09,1228931,0,57,1,,,0,蓝色,
粤B5WR39,113.945747,22.523899,2016-01-01 00:04:26,1433675,15,90,0,,,1,蓝色,
粤B5WR39,113.947365,22.523733,2016-01-01 00:04:41,1433675,28,90,0,,,1,蓝色,
粤B4K1S2,113.928146,22.492018,2016-01-01 00:05:12,1608266,0,135,1,,,0,蓝色,
粤BQ74Q5,114.173218,22.603050,2016-01-01 00:05:12,1519417,11,90,0,,,0,蓝色,
粤SQZ583,113.734352,23.019917,2016-01-01 00:05:06,1198307,51,282,0,,,0,,
粤BF7644,113.814880,22.610283,2016-01-01 00:05:11,1467026,70,90,0,,,1,蓝色,
粤BR5127,113.887779,22.561642,2016-01-01 00:04:05,1344821,0,0,0,,,0,黄色,
粤B4V1Q2,114.074364,22.531767,2016-01-01 00:05:06,1571096,75,45,0,,,1,蓝色,
粤SQS507,113.934853,23.091600,2016-01-01 00:05:11,1197523,55,152,0,,,0,,
粤SQM455,113.819901,22.818916,2016-01-01 00:05:11,1197425,69,190,0,,,0,,
粤SYP417,114.163002,22.840567,2016-01-01 00:05:12,1294012,0,0,0,,,0,,
```

Taxi ID

Position
(long, lat)

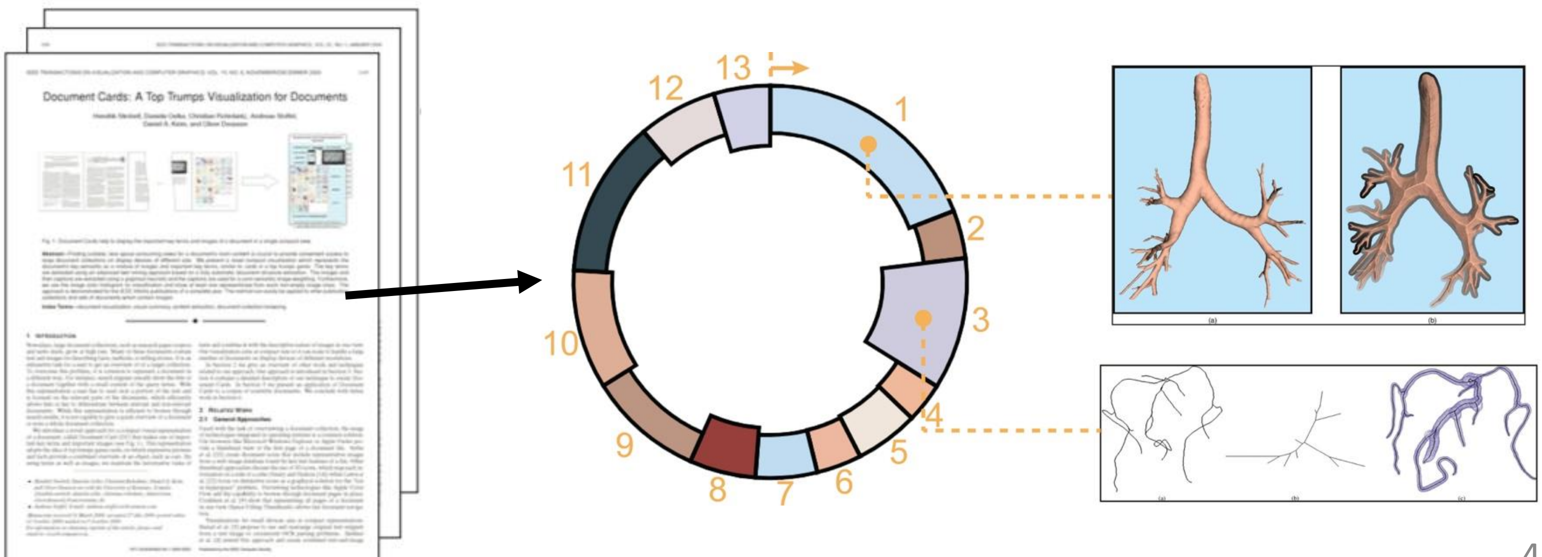
Time

Tables are Still the Most Common

- The majority of the data is represented in tabular form
  - How to "vectorize" non-tabular data is a very active area of research
  - For example, how to convert each word in the English language into a vector of numbers?
    - Naively, we can have a vector of length k, where k is the number of all the words in the English dictionary. Then each word is a vector of 0's, except for a single entry of 1.
    - This is very expensive and doesn't really afford analysis over the data. Is there a more compact representation?
  - Once the data is in tabular (vector / matrix) form, we can apply all the common data analysis methods!

Nested table

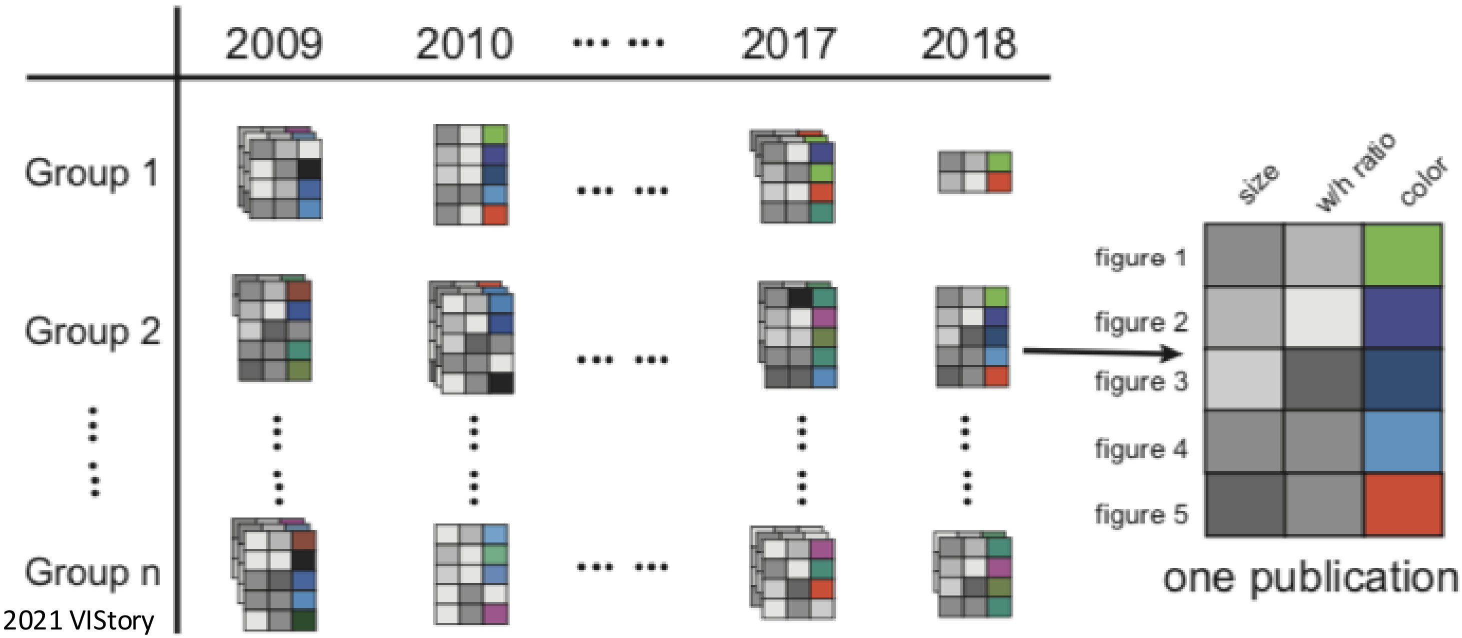
- Nested table: one can place all data attributes into a table, which may then be decomposed into smaller tables as needed (Database analysis and design. 1984)
- An example:
  - Papers with attributes of publication year, venue, authors, etc.
  - Figures in each paper with attributes of size, main color, WH ratio, etc.



Zeng et al., 2021 VIStory

Nested table

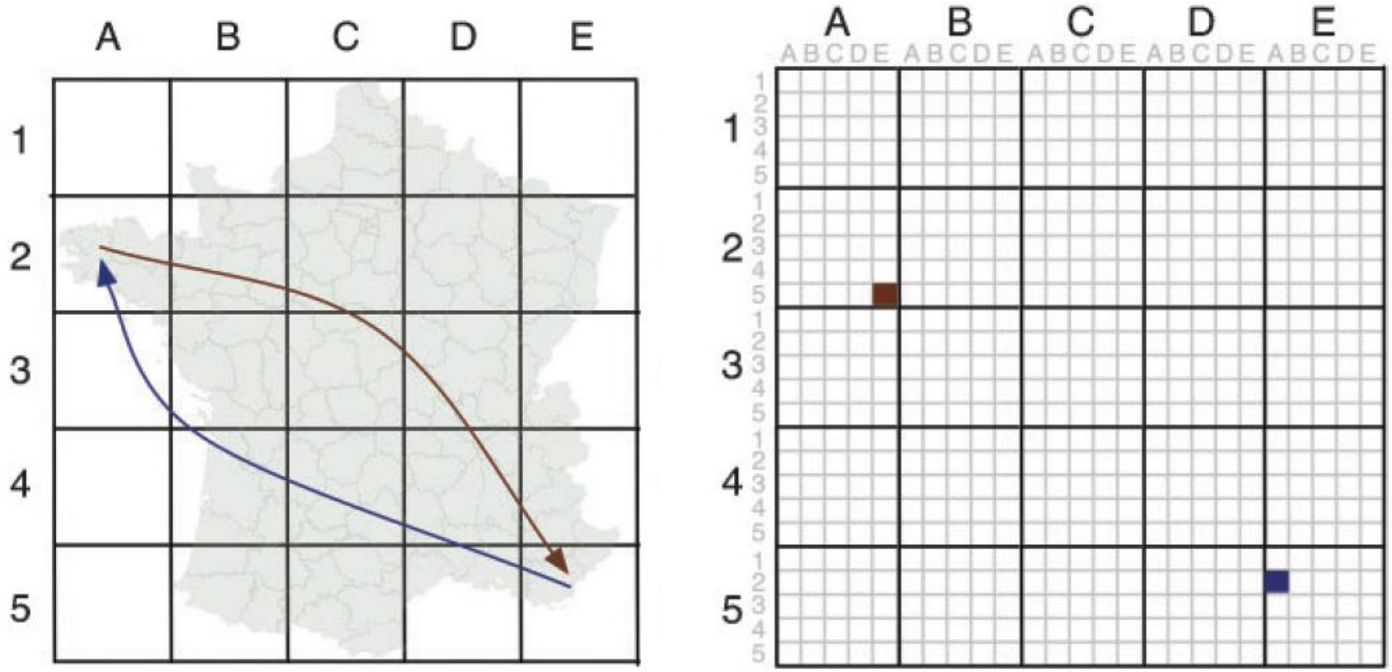
- Nested table: one can place all data attributes into a table, which may then be decomposed into smaller tables as needed (Database analysis and design. 1984)
- An example:
  - Papers with attributes of publication year, venue, authors, etc.
  - Figures in each paper with attributes of size, main color, WH ratio, etc.



Zeng et al., 2021 VIStory

Nested table

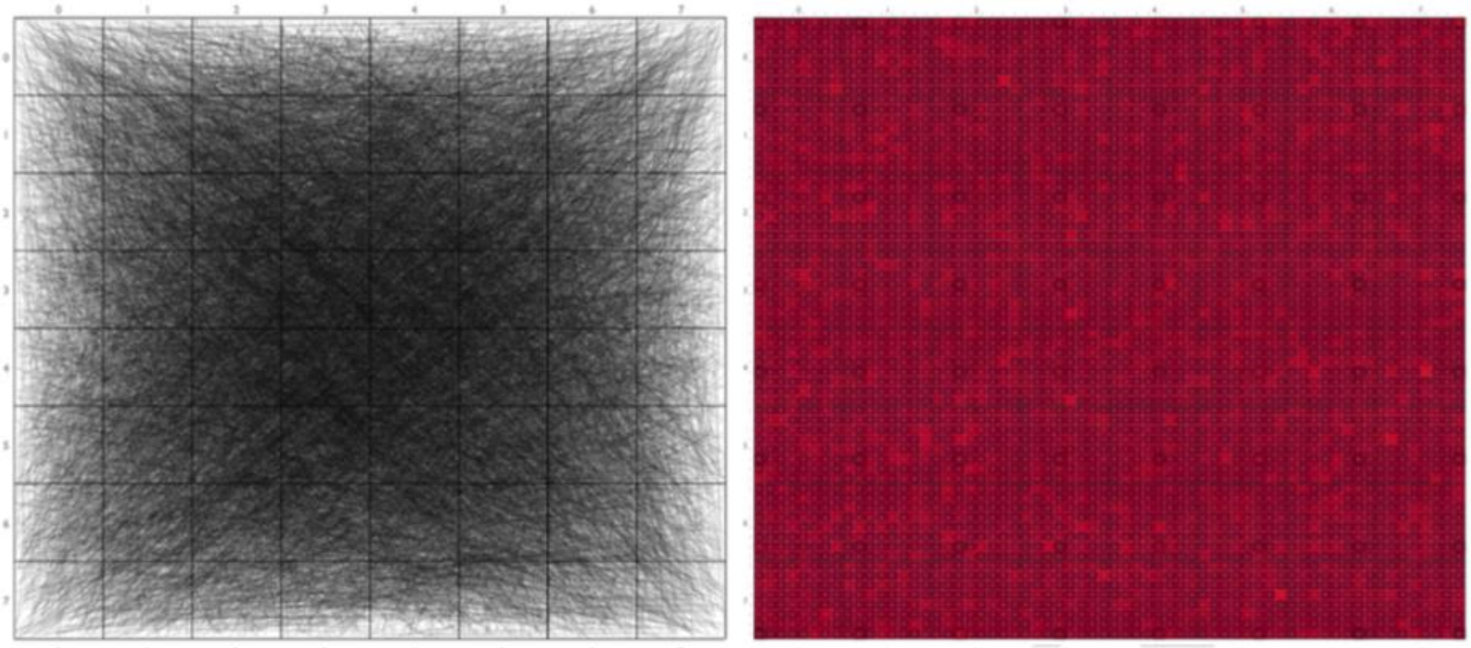
- OD movements: model as a nested table
  - Each grid is an origin (whole 2D space divided into a table)
  - In each grid, movements to destinations are further divided as a table (nested table)



J. Wood, J. Dykes, and A. Slingsby, "Visualization of origins, destinations and flows with OD maps," The Cartographic Journal, vol. 47, no. 2, pp. 117-129, 2010.

Nested table

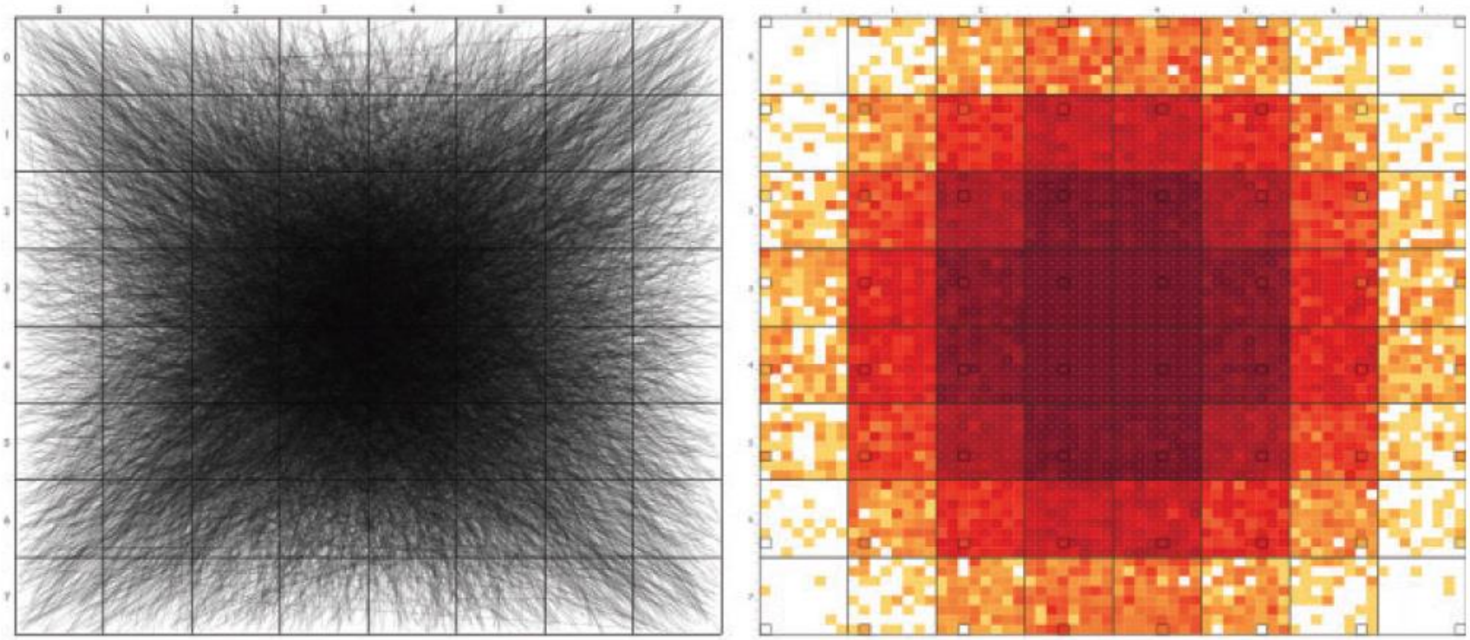
- OD movements: random
  - Each grid is an origin (whole 2D space divided into a table)
  - In each grid, movements to destinations are further divided as a table (nested table)



J. Wood, J. Dykes, and A. Slingsby, "Visualization of origins, destinations and flows with OD maps," The Cartographic Journal, vol. 47, no. 2, pp. 117-129, 2010.

Nested table

- OD movements: Gaussian distribution
  - Each grid is an origin (whole 2D space divided into a table)
  - In each grid, movements to destinations are further divided as a table (nested table)



J. Wood, J. Dykes, and A. Slingsby, "Visualization of origins, destinations and flows with OD maps," The Cartographic Journal, vol. 47, no. 2, pp. 117-129, 2010.

Demo: Nested Table

OLAP

- **OnLine Analytical Processing** (OLAP) was proposed by E. F. Codd, the father of the relational databases
- Relational databases put data into tables, while OLAP uses a multidimensional array representation
  - Such representations of data previously existed in statistics and other fields
- There are a number of data analysis and data exploration operations that are easier with such a data representation.

Multidimensional Array

- Two key steps in converting tabular data into a multidimensional array.
  - First, identify which attributes are to be the dimensions and which attribute is to be the target attribute whose values appear as entries in the multidimensional array.
    - Attributes used as dimensions *must have discrete values*
    - The target value is typically *a count or continuous value*, e.g., the cost of an item
    - Can have no target variable at all except the count of objects that have the same set of attribute values
  - Second, find the value of each entry in the multidimensional array by summing the values (of the target attribute) or count of all objects that have the attribute values corresponding to that entry.

Example: Iris Dataset

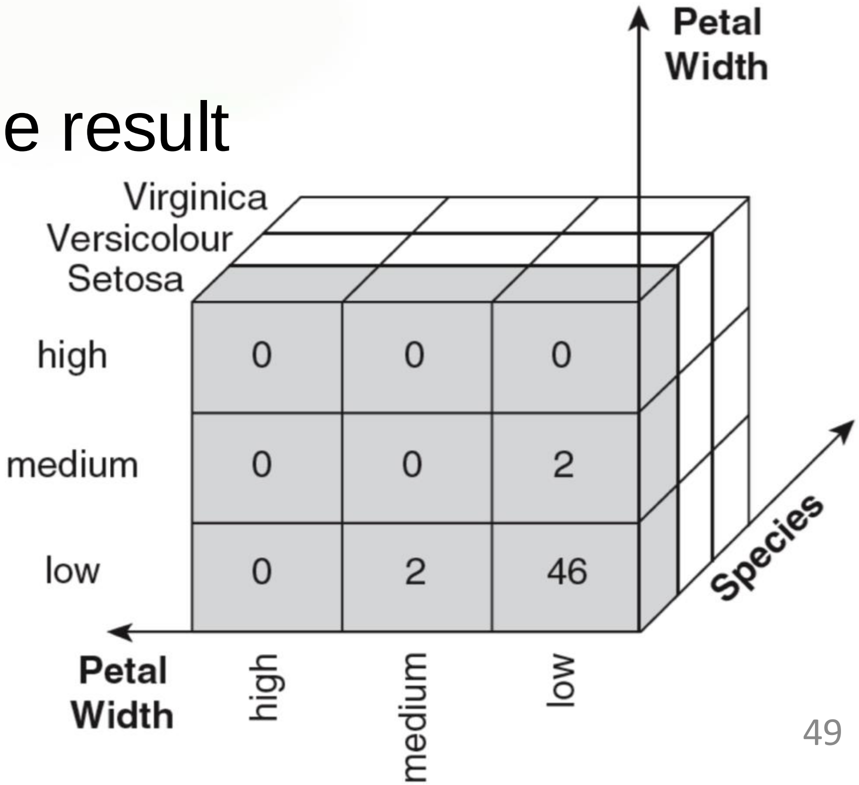
- How the attributes (petal length, petal width, and species type) can be converted to a multidimensional array
  - First, we *discretized* the petal width and length to have categorical values: *low*, *medium*, and *high*
  - We get the following table — note the *count* attribute

Petal Length	Petal Width	Species Type	Count
low	low	Setosa	46
low	medium	Setosa	2
medium	low	Setosa	2
medium	medium	Versicolour	43
medium	high	Versicolour	3
medium	high	Virginica	3
high	medium	Versicolour	2
high	medium	Virginica	3
high	high	Versicolour	2
high	high	Virginica	44

Example: Iris Dataset (cont'd)

- Each unique tuple of petal width, petal length, and species type identifies one element of the array
- This element is assigned the corresponding count value
- The figure illustrates the result
- All non-specified

  tuples are 0



Example: Iris Dataset (cont'd)

- Slices of the multidimensional array are shown by the following cross-tabulations
- What do these tables tell us?

		Petal Width		
		low	medium	high
Length	low	46	2	0
	medium	2	0	0
	High	0	0	0

		Petal Width		
		low	medium	high
Length	low	0	0	0
	medium	0	43	3
	High	0	2	2

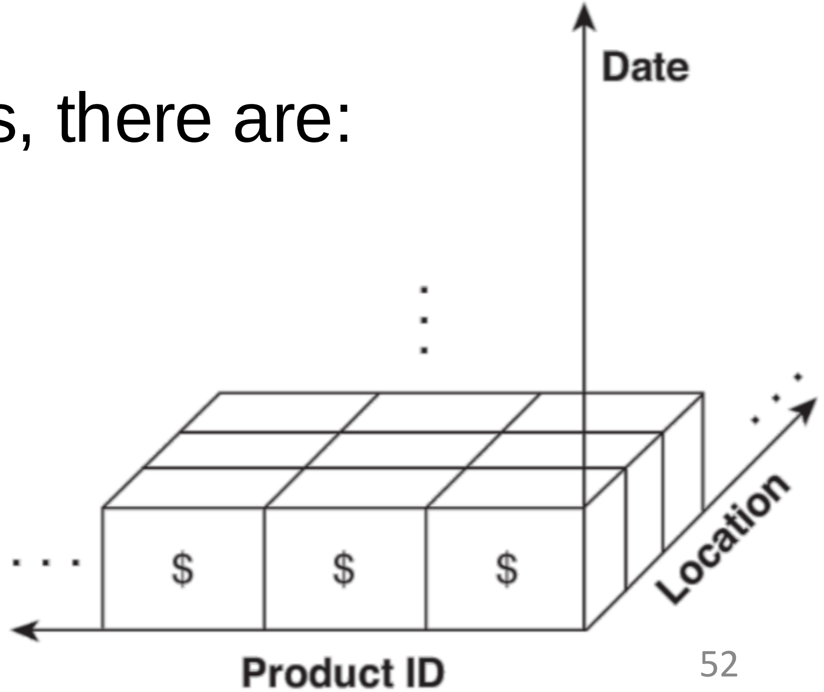
		Petal Width		
		low	medium	high
Length	low	0	0	0
	medium	0	0	3
	High	0	3	44

Data cube

- The key operation of OLAP is the formation of a data cube
- A data cube is a multidimensional representation of data, together with all possible aggregates
- By all possible aggregates, we mean the aggregates that result by selecting a proper subset of the dimensions and summing over all remaining dimensions
- For example, if we choose the species type dimension of the Iris data and sum over all other dimensions, the result will be a one-dimensional entry with three entries, each of which gives the number of flowers of each type

Data cube

- Consider a data set that records the sales of products at a number of company stores at various dates.
- This data can be represented as a 3 dimensional array
- Using binomial coefficients, there are:
  - 3 of 2D aggregates
  - 3 of 1D aggregates
  - 1 of 0D aggregates (the overall total)



Data cube

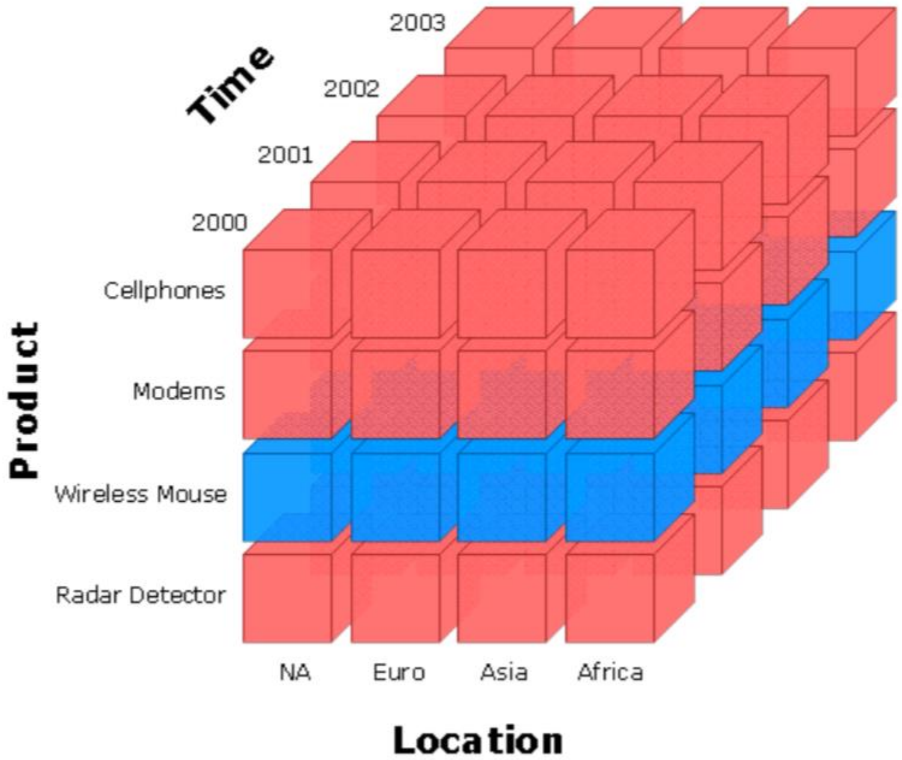
- The following figure table shows one of the *2D aggregates*, along with two of the *1D aggregates*, and the overall total (the *0D aggregate*)

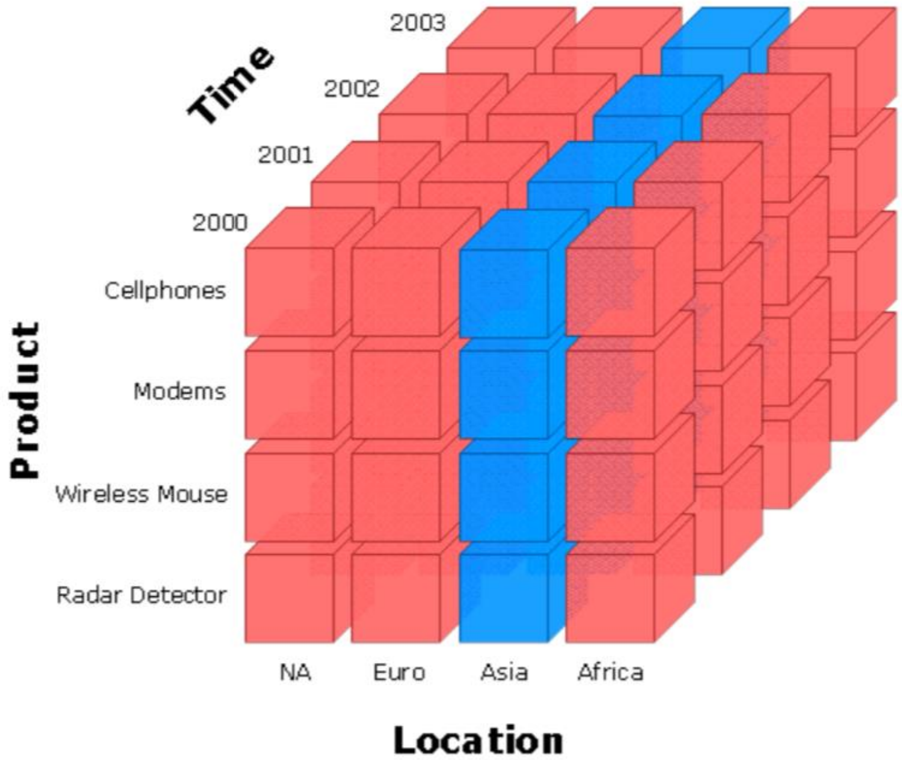
		Date				*total*
		1/1/2017	2/1/2017	...	31/12/2017	
Product ID	1	$1,001	$987	...	$891	**$370,000**
	⋮	⋮			⋮	⋮
	27	$10,265	$10,225		$9,325	**$3,800,020**
	⋮	⋮			⋮	⋮
total		**$527,362**	**$532,953**	**...**	**$631,221**	**$227,352,127**

Slicing and dicing

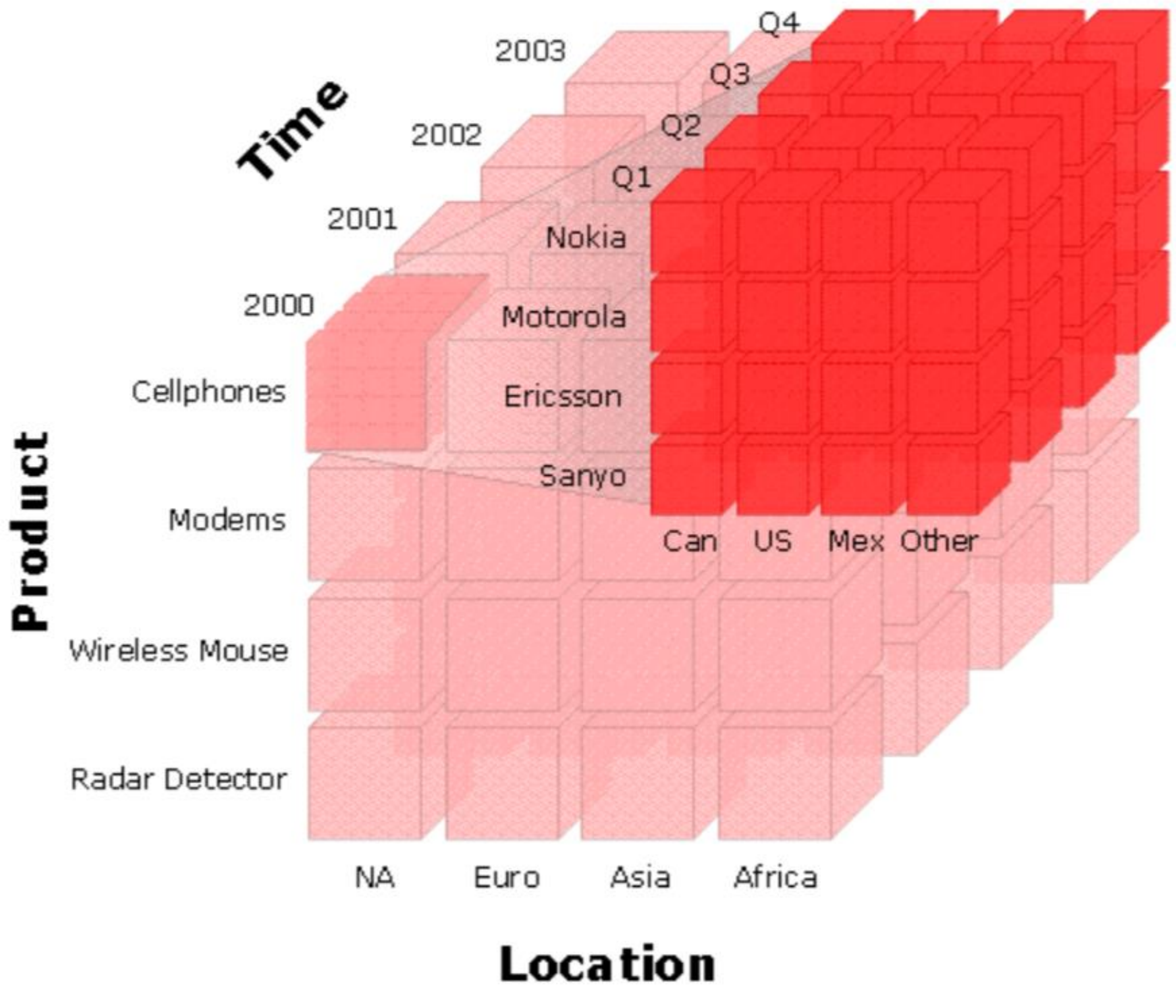
- **Slicing** is selecting a group of cells from the entire multidimensional array by specifying a specific value for one or more dimensions.
- **Dicing** involves selecting a subset of cells by specifying a range of attribute values.
  - This is equivalent to defining a subarray from the complete array.
- In practice, both operations can also be accompanied by aggregation over some dimensions.

Slicing examples





Dicing examples



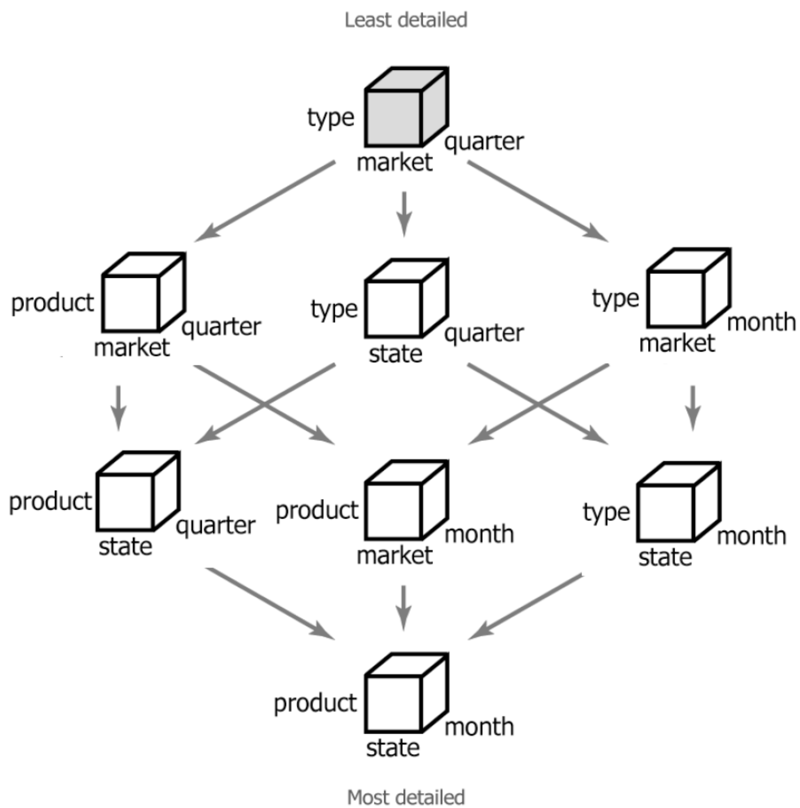
Roll-up & drill-down

- Attribute values often have a hierarchical structure
  - Each *date* is associated with a *year*, *month*, and *week*
  - A *location* is associated with a *continent*, *country*, and *city*
  - Products can be divided into various categories, such as *clothing*, *electronics*, and *furniture*
- Note that these categories often nest and form a **tree** or **lattice**
  - A *year* contains *months* which contains *days*
  - A *country* contains a *region* which contains a *city*

Roll-up & drill-down

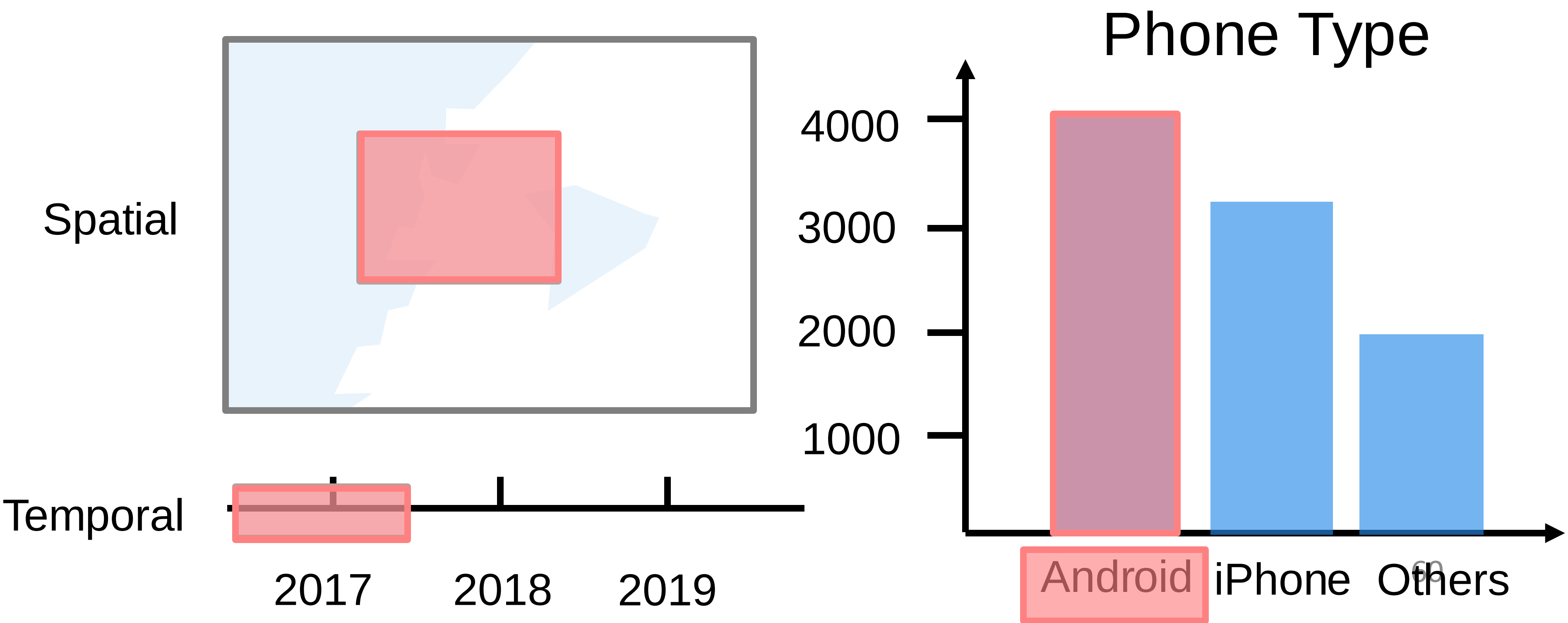
- This hierarchical structure gives rise to the **roll-up** and **drill-down** operations
  - For sales data, we can aggregate (**roll-up**) the sales across all the dates in a month
  - Likewise, given the data where the time is broken into months, we can split the monthly sales totals (**drill-down**) into daily sales totals.
- Similarly, we can drill-down or roll-up on the *location* or *product ID* attributes

Roll-up & drill-down



Example: large ST dataset

- Imagine to query a mobile phone sales dataset
  - Spatial: city, province, country
  - Temporal: day, week, month, year
  - Phone type: android, iPhone, etc.



Example: large ST dataset

- Common approaches
  - Sampling
    - Park et al., “Visualization-Aware Sampling for Very Large Databases”, ICDE, 2016.
    - Samples may not be representative.
  - Parallel computing
    - Requires powerful computing resources
  - Pre-aggregation
    - imMens [EuroVis, 2013], NanoCubes [IEEE Vis, 2013], SmartCubes [IEEE Vis, 2019]

Example: large ST dataset

- **Data Cubes**: all aggregates of dimension combinations
- **Cuboid**: the aggregates of a combination of dimensions

Count
5

Country	Count
US	3
CAN	2

Phone	Count
iPhone	3
Android	2

Year	Count
2019	3
2018	2

Country	Phone	Count
US	iPhone	2
US	Android	1
CAN	iPhone	1
CAN	Android	1

Country	Year	Count
US	2019	1
US	2018	2
CAN	2019	1

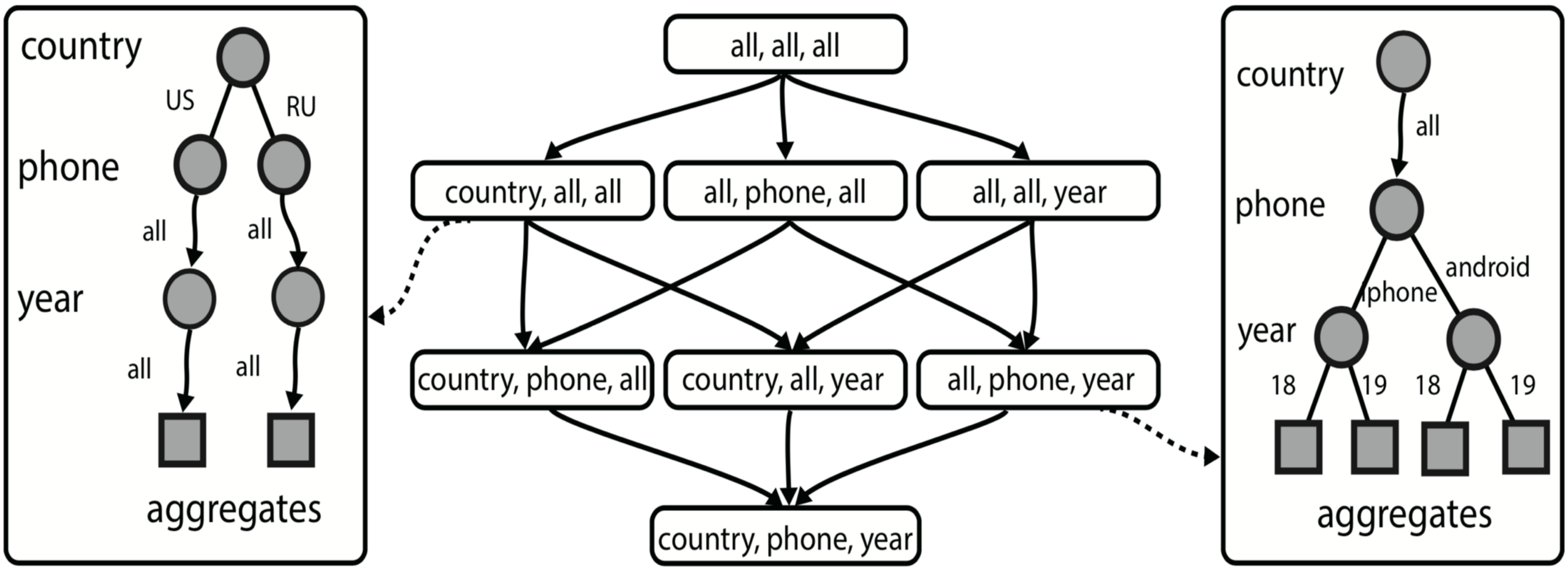
Phone	Year	Count
iPhone	2019	2
iPhone	2018	1
Android	2018	1
Android	2019	1

Country	Phone	Year	Count
US	iPhone	2019	1
US	iPhone	2018	1
US	Android	2018	1
CAN	iPhone	2019	1
CAN	Android	2019	1

Jim et al., 1997

Example: large ST dataset

- **Data Cubes**: all aggregates of dimension combinations
- **Cuboid**: the aggregates of a combination of dimensions



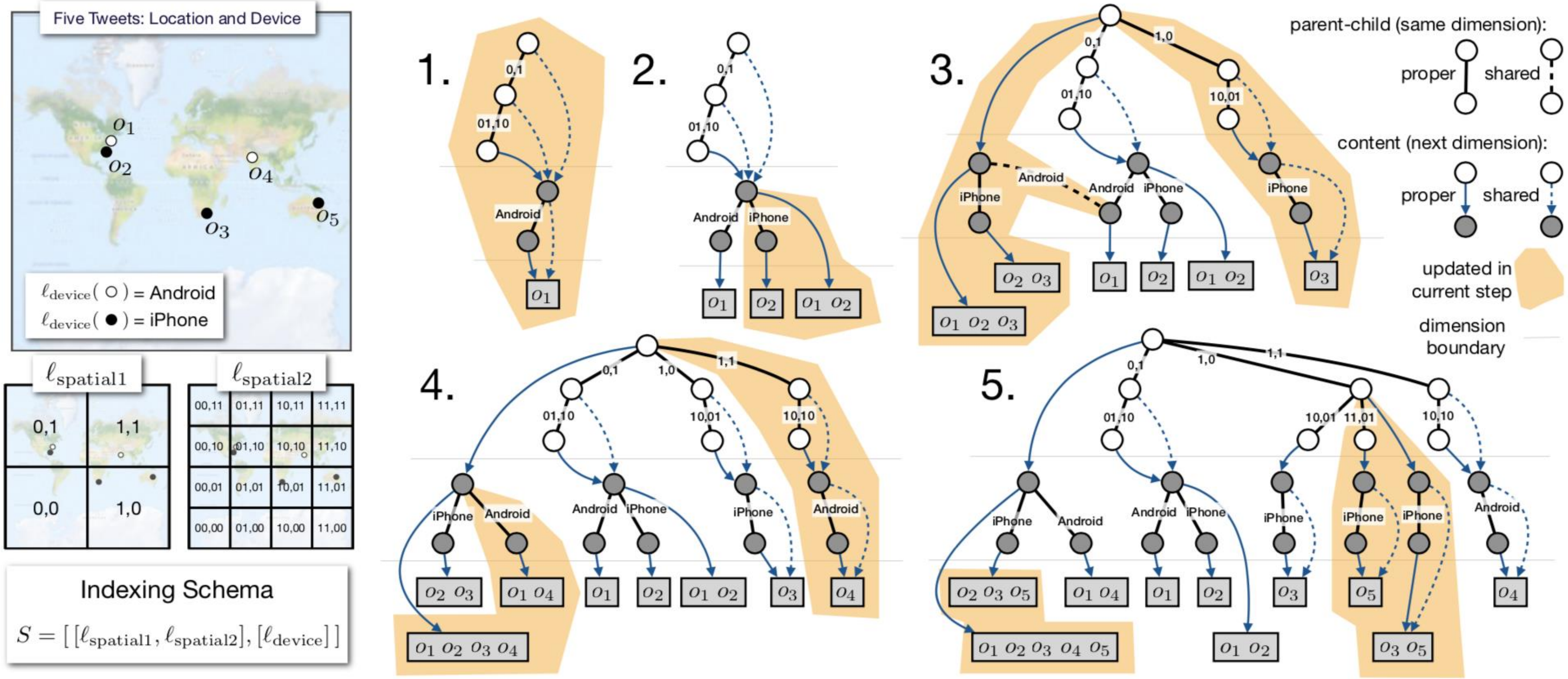
Data Cubes

A Cuboid

Jim et al., 1997

Example: large ST dataset

- Effective way to construct a data cube that fits in a modern laptop's main memory



Nanocubes [Lins et al., 2013]

Summary

- The choice of **how to model the data** affects what kinds of analyses can be applied
  - This will directly impact the kind of findings your analysis can result in
- The same consideration applies to **data attributes**
  - Depending on the data “types” there are different (statistical) analyses and visualizations that can be applied
- A proper **data structure** also facilitates data analysis and visualization
  - Rooted in database