## UNIT -1

**UNIT-I:**

**Introduction and Data Foundation**: Basics - Relationship between Visualization and Other Fields- The Visualization Process - Pseudo code Conventions - The Scatter plot. Data Foundation - Types of Data - Structure within and between Records - Data Preprocessing - Data Sets

**References:**

1.Matthew Ward, Georges Grinstein and Daniel Keim, : Interactive Data Visualization Foundations, Techniques, Applications ",2010

# 1.1Basics:

### Visualization

- **We define visualization as the communication of information using graphical representations.**
- Pictures have been used as a mechanism for communication since before the formalization of written language
- A single picture can contain a wealth of information that can be processed much more quickly than a comparable page of words
- This is because image interpretation is performed in parallel within the human perceptual system, while the speed of text analysis is limited by the sequential process of reading.

### Visualization in everyday life:

It is an interesting exercise to consider the number and types of data and information visualization that we encounter in our normal activities

- a table in a newspaper, representing data being discussed in an article
- a train and subway map with times used for determining train arrivals and departures
- a map of the region, to help select a route to a new destination
- a weather chart showing the movement of a storm front that might influence your weekend activities
- a graph of stock market activities that might indicate an upswing (or downturn) in the economy

- a plot comparing the effectiveness of your pain killer to that of the leading brand

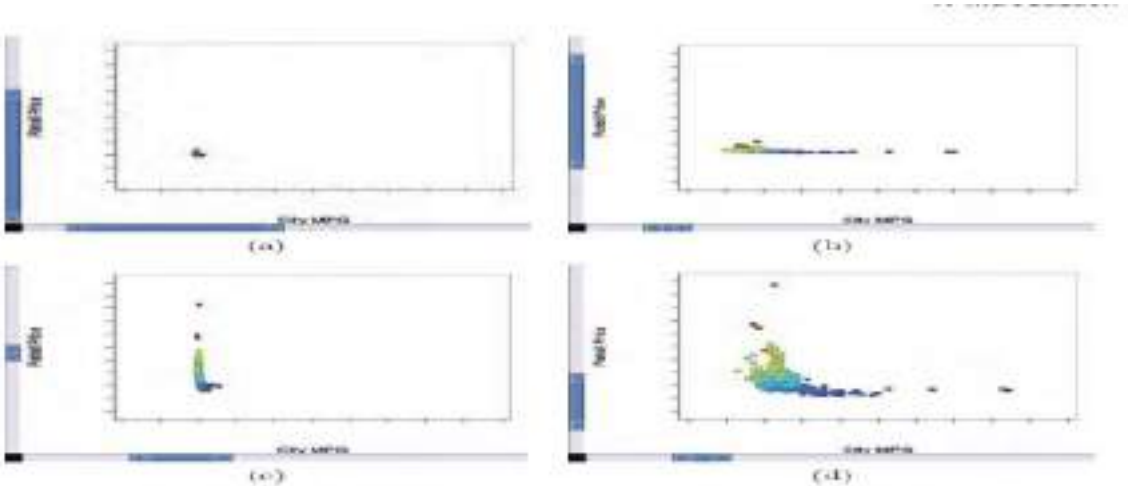
an instruction manual for putting together a bicycle, with views specific to each part as it is added;

- a highway sign indicating a curve, merging of lanes, or an intersection.
- Visualization is used on a daily basis in many areas of employment as well, such as
- the result of a financial and stock market analysis;
- a mechanical and civil engineering rotary bridge design and systems analysis;
- a breast cancer MRI for diagnosis and therapy;
- a comet path data and trend analysis;
- the analysis of human population smoking behaviors;
- the study of actuarial data for confirming and guiding quantitative analysis;
- the simulation of a complex process
- the analysis of a simulation of a physical system;
- a visual representation of Facebook friends and their connections;
- marketing posters and advertising

**Importance of Visualization:**

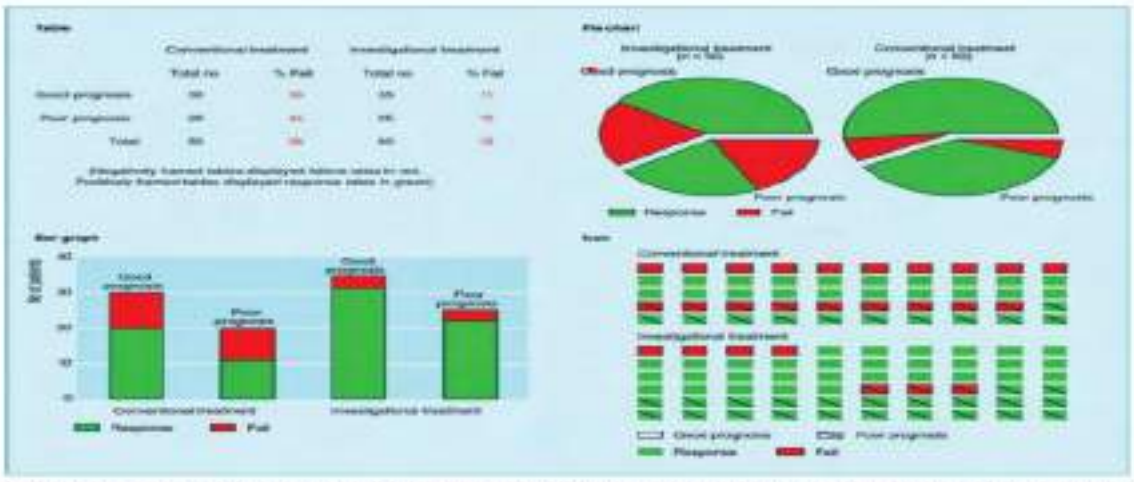
- There are many reasons why visualization is important.
- Perhaps the most obvious reason is that we are visual beings who use sight as one of our key senses for information understanding.
- The two examples below highlight why visualization is so important in decision making, and the role of human preferences and training.

One example focuses on data distortion, and the other on human interpretation



The same data plotted with different scales is perceived dramatically differently: (a) Equally (uniformly) large scale in both $x$ and $y$. (b) Large scale in $y$. (c) Large scale in $x$. (d) Scale determined by range of $x$- and $y$-values.

## 1.4 Various Visual representations:



Various visual representations of a hypothetical clinical trial. The icon display (lower right) was the most effective for the decision to stop the clinical trial. The bar and pie charts were the least effective. (Image courtesy [98], © 1999 BMJ.)

- it is obvious that Marketing has the most consultants and that the Driver has the longest chain of command
- The flood of data, its democratization, and the web have brought about an increasing use of both static and interactive visualizations that are much more aesthetic and understandable to the user
- The exploration and analysis of large marketing, financial, security, medical, and biological data sets has led to results needing to be explained,

**An organizational chart.:**


3. An organizational chart. Patterns often require a great deal of words to describe.

**Early Visualizations:**

- Perhaps the first technique for graphically recording and presenting information was that used by early man.
- An example is the early Chauvet-Pont-d'Arc Cave, located near Vallon-Pont-d'Arc in southern France [421].
- The Chauvet Cave contains over 250 paintings, created approximately 30,000 years ago.
- These were likely meant to pass on information to future generations.

**Cave painting:**

One of the Lascaux cave paintings on the northern slopes of the French Pyrenees on the banks of the Vézère river [329].

**Mesopotamia Graphics:**

Early graphical writing. The Kish limestone tablet from Mesopotamia [338].

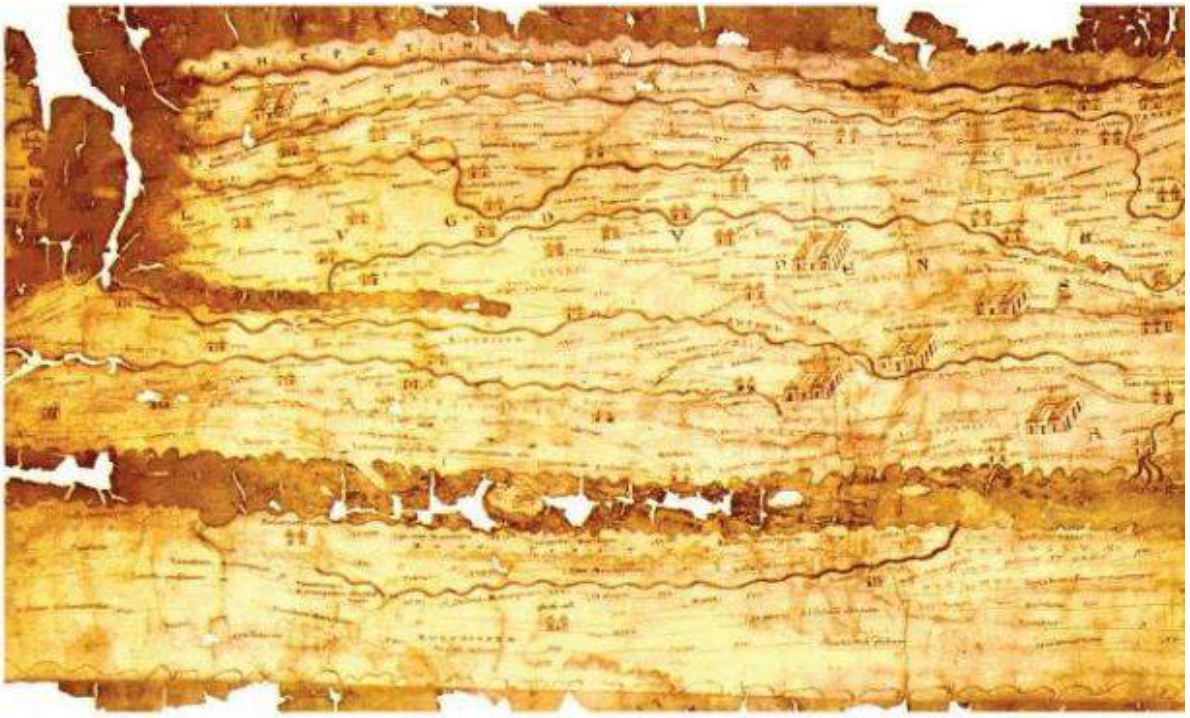


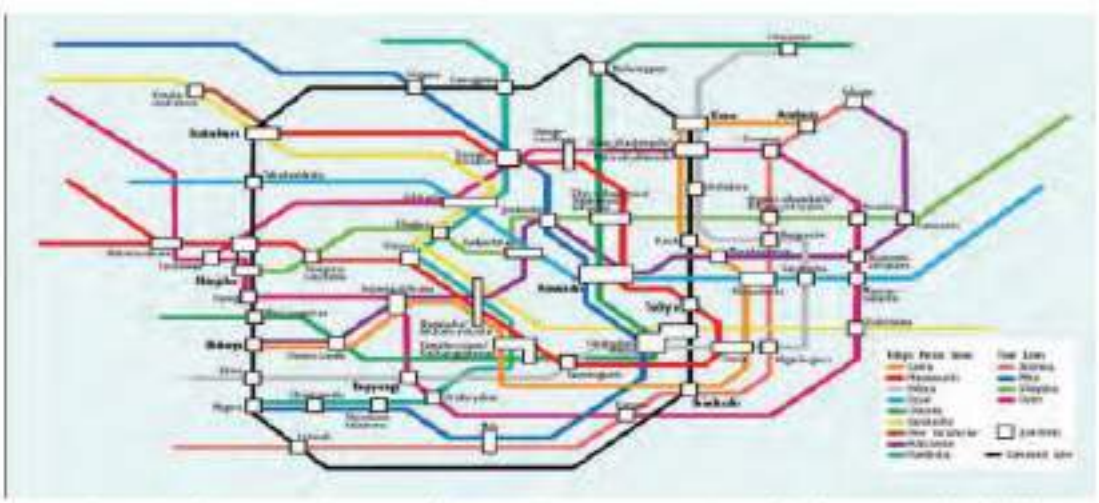
Figure 1.6. A copy of one of the 12 pages of the Peutinger Map set, showing the roads of the Roman Empire. (Image courtesy http://www.livius.org/pen-pg/peutinger/map.html.)

**Visualization today:**

- **Visualization most often provides different levels of both qualitative and quantitative views of the information being communicated**
- For example, Figure below shows a map of the Tokyo underground
- It provides an easy-toread yet logically distorted view of the criss-crossing network of the subway system to facilitate interpretation

- similar techniques have been used for a number of subway and other transportation systems as well as for processes and their flows

**Visualization Today:**



The Tokyo Underground map. A logical representation of the metro highlighting qualitative relationships between the stops. (Image courtesy Wikimedia Commons.)

**Google maps**

The google.com map directions from 198 Riverside St., Lowell, MA (UMass Lowell, North Campus) to 883 Broadway St., Lowell, MA (UMass Lowell, South Campus). Google.com maps provide graphical cues drawn on top of road maps to indicate driving directions from point A to point B. (Image © 2009 Google.)

## 1.2. Relationship between Visualization and Other Fields:

Visualization, as a field, has significant relationships with several other fields due to its multidisciplinary nature

- Here are some key relationships between visualization and other fields:

**Data Science and Analytics**:

- Visualization plays a crucial role in data science and analytics by helping to understand and communicate complex data patterns and insights
- Data visualization techniques enable analysts to explore large datasets, identify trends, outliers, and correlations, and present their findings effectively to both technical and non-technical audiences

**Computer Science and Information Technology**

- Visualization is closely linked to computer science and information technology, particularly in terms of developing algorithms and software tools for creating interactive and immersive visual representations.
- Computer graphics, image processing, and human-computer interaction are among the subfields that contribute to visualization techniques and technologies.
- Design and User Experience (UX): Visualization heavily relies on design principles and user experience considerations to create effective and aesthetically pleasing visual representations.
- Designers play a crucial role in developing visually engaging and intuitive interfaces for data visualization applications, ensuring that the information is presented in a user-friendly and accessible manner.

**Difference between Visualization and Computer Graphics**

**Originally, visualization was considered a subfield of computer graphics, primarily because visualization uses graphics to display information via images.** As illustrated by any of the computer-generated images shown earlier, visualization applies graphical techniques to generate visual displays of data. Here, graphics is used as the communication medium.

In all visualizations, one can clearly see the use of the graphics primitives (points, lines, areas, and volumes). Beyond the use of graphics, the most important aspect of all visualizations is their connection to data. Computer graphics focuses primarily on graphical objects and the organization of graphic primitives; visualizations go one step further and are based on the underlying data, and may include spatial positions, populations, or physical measures. Consequently, visualization is the application of graphics to display data by mapping data to graphical primitives and rendering the display.

However**, visualization is more than simply computer graphics**. The field of visualization encompasses aspects from numerous other disciplines, including human-computer interaction, perceptual psychology, databases, statistics, and data mining, to name a few. While computer graphics can be used to define and generate the displays that are used to communicate the information, the sources of data and the way users interact and perceive the data are all important components to understand when presenting information,

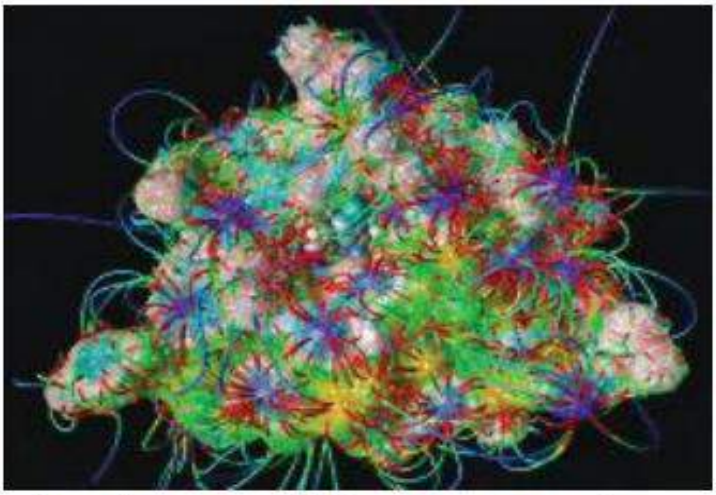
Our view is **that computer graphics is predominantly focused on the creation of interactive synthetic images and animations of three-dimensional objects, most often where visual realism is one of the primary goals.** A secondary application of computer graphics is in art and entertainment, with video games, cartoons, advertisements, and movie special effects as typical examples. Visualization, on the other hand, does not emphasize visual realism as much as the effective communication of information. Many types of visualizations do not deal with physical objects, and those that do are often communicating attributes of the objects that would normally not be visible, such as material stress or fluid flow patterns. Thus, while computer graphics and visualization share many concepts, tools, and techniques, the underlying models (the information to be visualized) and goals (what the viewer hopes to extract) are fundamentally different.

**Thus, computer graphics consists of the tools that display the visualizations . This includes the graphics-programming language (OpenGL, DirectX, Processing, Java3D), the underlying graphics hardware (NVidia or ATI/AMD graphics cards), the rendering process (flat, Gouraud, Phong, ray tracing, or radiosity), the output format (JPEG, TIFF, AVI, MPEG), and more.** We consider computer graphics to be the underpinning of visualization and thus need to keep abreast of it.

**Scientific Data Visualization vs. Information Visualization**

- Although during the 1990s and early 2000s the visualization community differentiated between scientific visualization and information visualization, we do not.
- Both provide representations of data.

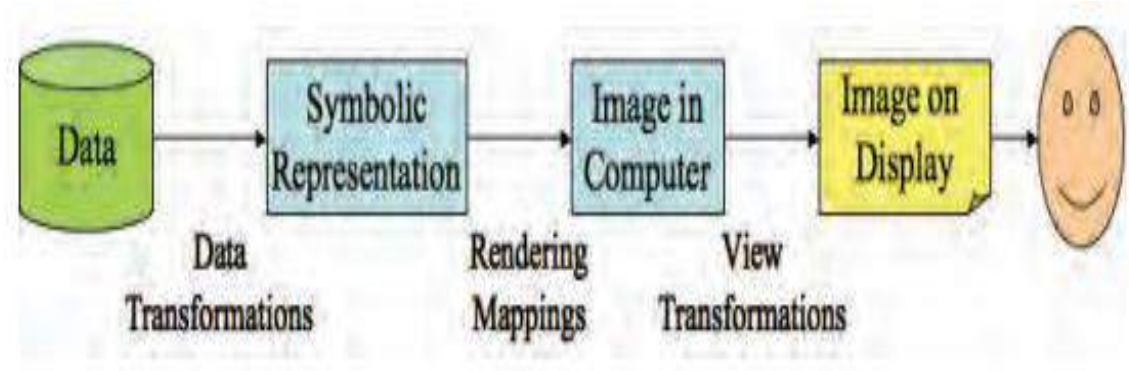
- However the data sets are most often different. Figure 1.25 highlights the importance and value of having both.
- Biomolecular chemistry, which once only considered the visual representation of molecules as stick and balls, has migrated over time



An example of a drug that targets HIV-I reverse transcriptase. (Image courtesy IBM OpenDX Highlights.)

- to representations as spheres with rods, to more realistic ones, to now including information visualizations (scatterplots and other visualizations).

## 1.3. The Visualization Process

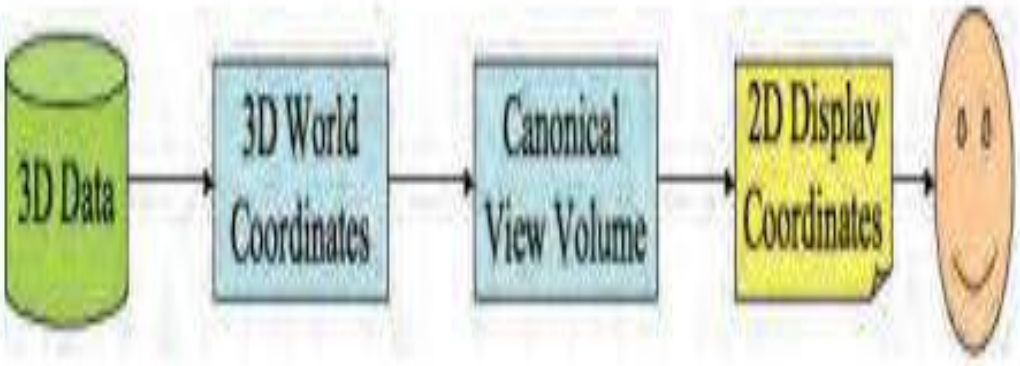



The visualization process at a very high or primitive level view.

- To visualize data, one needs to define a mapping from the data to the display (see Figure ). There are many ways to achieve this mapping.
- The user interface consists of components, some of which deal with data needing to be entered, presented, monitored, analyzed, and computed.
- These user interface components are often input via dialog boxes, but they could be visual representations of the data to facilitate the selections required by the user.
- Visualizations can provide mechanisms for translating data and tasks into more visual and intuitive formats for users to perform their tasks.
- This means that the data values themselves, or perhaps the attributes of the data, are used to define graphical objects, such as points, lines, and shapes; and their attributes, such as size, position, orientation, and color. Thus, for example, a list of numbers can be plotted by mapping each number to the y-coordinate of a point and the number's index in the list to the xcoordinate. Alternatively, we could map the number to the height of a bar or the color of a square to get a different way to view the data.
- Visualization is often part of a larger process, which may be exploratory data analysis, knowledge discovery, or visual analytics. In this discovery process, the preparation of data depends upon the task and often requires massaging erroneous or noisy data. Visualization and analysis go hand in hand with the goal of building a model that represents or approximates the data. Visualization in data exploration is used to convey information, discover new knowledge, and identify structures, patterns, anomalies, trends, and relationships.
- **The process of starting with data and generating an image, a visualization, or a model via the computer is traditionally described as a pipeline—a sequence of stages that can be studied independently in terms of algorithms, data structures, and coordinate systems.** These processes or pipelines are different for graphics, visualization, and knowledge discovery, but overlap a great deal. All start with data and end with the user.

**Computer Graphics Pipeline**

- **Modeling.** A three-dimensional model, consisting of planar polygons defined by vertices and surface properties, is generated using a world coordinate system.
- **Viewing.** A virtual camera is defined at a location in world coordinates, along with a direction and orientation (generally given as vectors).
- All vertices are transformed into a viewing coordinate system based on the camera parameters




The graphics pipeline.

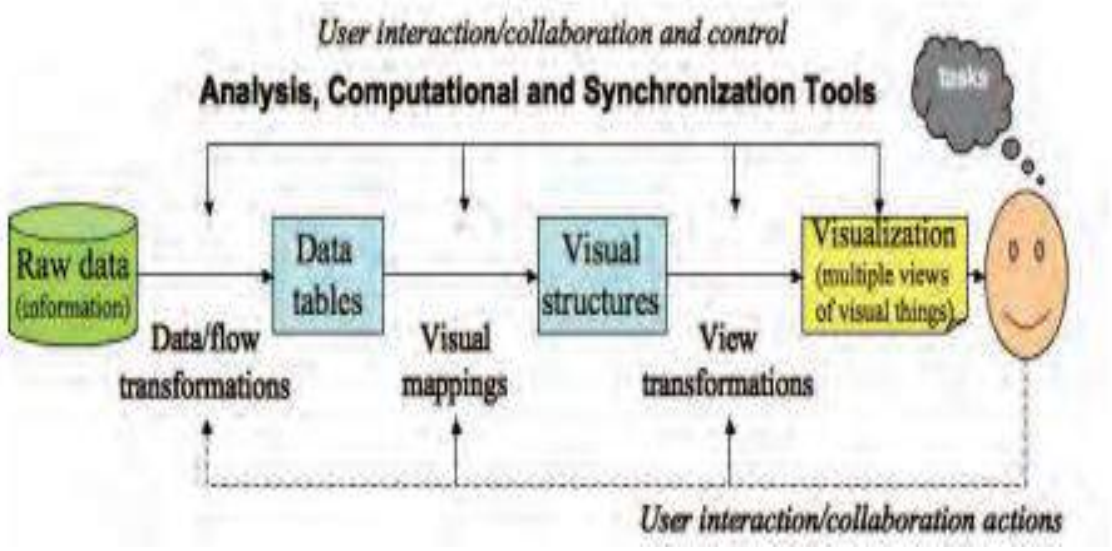
- **Clipping:-** By specifying the bounds of the desired image (usually given by corner positions on a plane of projection placed in front of the
- camera), objects out of view can be removed, and those that are partially visible can be clipped.
- Objects may be transformed into a normalized viewing coordinates to simplify the clipping process.
- Clipping can actually be performed at many different stages of the pipeline.
- **Hidden surface removal:-** Polygons facing away from the camera, or those obscured by others, are removed or clipped.
- This process may be integrated into the projection process.

- **Projection:-** Three-dimensional polygons are projected onto the twodimensional plane of projection, usually using a perspective transformation.
- The results may be in a normalized 2D coordinate system or device/screen coordinates
- **Rendering:-** The actual color of the pixels associated with a visible polygon depends on a number of factors, including the material properties being synthesized (base color, texture, surface roughness, shininess), the type(s), location(s), color, and intensity of the light source(s), the degree of occlusion from direct light exposure, and the amount and color of light being reflected off of other objects onto the polygon
- This process may also be applied at different stages of the pipeline (e.g., vertex colors can be assigned during the modeling process); however, due to its computational complexity, it is usually performed in conjunction with projection.

**The Visualization Pipeline**

The data/information visualization pipeline has some similarities to the graphics pipeline, at least on an abstract level. The stages of this pipeline (see Figure ) are as follows:

**Data modeling**. The data to be visualized, whether from a file or a database, has to be structured to facilitate its visualization. The name, type, range, and semantics of each attribute or field of a data record must be available in a format that ensures rapid access and easy modification




**Data selection**. Similar to clipping, data selection involves identifying the subset of the data that will be potentially visualized. This can occur totally under user control or via algorithmic methods, such as cycling through time slices or automatically detecting features of potential interest to the user

**Data to visual mappings.** The heart of the visualization pipeline is performing the mapping of data values to graphical entities or their attributes. Thus, one component of a data record may map to the size of an object, while others might control the position or color of the object. This mapping often involves processing the data prior to mapping, such as scaling, shifting, filtering, interpolating, or subsampling

Scene parameter setting (view transformations). As in traditional graphics, the user must specify several attributes of the visualization that are relatively independent of the data. These include color map selection (for different domains, certain colors have clearly defined meaning), sound map selection (in case the auditory channels will be conveying information as well), and lighting specifications (for 3D visualizations).

**Rendering or generation of the visualization**. The specific projection or rendering of the visualization objects varies according to the mapping being used; techniques such as shading or texture mapping might be involved, although many visualization techniques only require drawing lines and uniformly shaded polygons. Besides showing the data itself, most visualizations

also include supplementary information to facilitate interpretation, such as axes, keys, and annotations

**The Knowledge Discovery Pipeline**

**The knowledge discovery (also called data mining) field has its own pipeline.** As with the graphics and visualization pipelines, we start with data; in this case we process it with the goal of generating a model, rather than some graphics display.

Note that the visualization pipeline can be overlaid on this knowledge discovery (KD) pipeline. If we were to look at a pipeline for typical statistical analysis procedures, we would find the same process structure:

**Data.** In the KD pipeline there is more focus on data, as the graphics and visualization processes often assume that the data is already structured to facilitate its display.

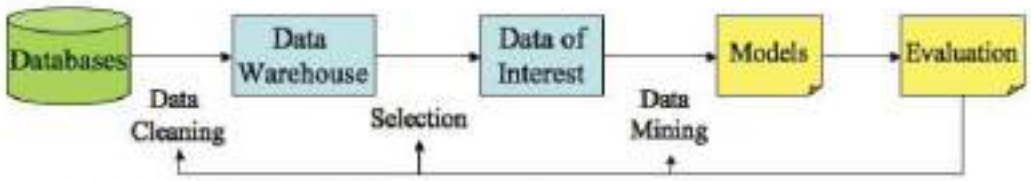
**Data integration, cleaning, warehousing and selection**. These involve identifying the various data sets that will be potentially analyzed. Again, the user may participate in this step. This can involve filtering, sampling, subsetting, aggregating, and other techniques that help curate and manage the data for the data mining step.

**Data mining.** The heart of the KD pipeline is algorithmically analyzing the data to produce a model.

**Pattern evaluation.** The resulting model or models must be evaluated to determine their robustness, stability, precision, and accuracy.

**Rendering or visualization**. The specific results must be presented to the user. It does not matter whether we think of this as part of the graphics or visualization pipelines; the fact is that a user will eventually need to see the results of the process.

Interactive visualization can be used at every step of the KD pipeline. One can think of this as computational steering.




One view of the knowledge discovery pipeline or process.

## 1.4. Pseudo-code Conventions

- **In our pseudo-code, we aim to convey the essence of the algorithms at hand, while leaving out details required for user interaction, graphics nuances, and data management**
- We therefore assume that the following global variables and functions exist in the environment of the pseudo-code
- data—The working data table
- This data table is assumed to contain only numeric values. In practice, dimensions of the original data table that contain non-numeric values must be somehow converted to numeric values
- When visualizing a subset of the entire original data table, the working data table is assumed to be the subset
- m—The number of dimensions (columns) in the working data table. Dimensions are typically iterated over using j as the running dimension index
- n—The number of records (rows) in the working data table. Records are typically iterated over using i as the running record index
- Normalize(record, dimension), Normalize(record, dimension, min, max)—A function that maps the value for the given record and dimension in the working data table to a value between min and max, or between zero and one if min and max are not specified
- The normalization is typically linear and local to a single dimension
- However, in practice, code must be structured such that various kinds of normalization could be used (logarithmic or square root, for example) either locally (using the bounds of the current dimension), globally

(using the bounds of all dimensions), or local to the active dimensions (using the bounds of the dimensions being displayed)

- Also, in practice, one must accommodate multiple kinds of normalization within a single visualization. For example, a scatter plot may require a linear normalization for the x-axis and a logarithmic normalization for the y-axis.
- Color(color)—A function that sets the color state of the graphics environment to the specified color (whose type is assumed to be an integer containing RGB values)
- MapColor(record, dimension)—A function that sets the color state of the graphics environment to be the color derived from applying the global color map to the normalized value of the given record and dimension in the working data table
- Circle(x, y,radius)—A function that fills a circle centered at the given (x, y)-location, with the given radius, with the color of the color state of the graphics environment. The plotting space for all visualizations is the unit square. In practice, this function must map the unit square to a square in pixel coordinates
- Polyline(xs, ys)—A function that draws a polyline (many connected line segments) from the given arrays of x- and y-coordinates.
- Polyline(xs, ys)—A function that draws a polyline (many connected line segments) from the given arrays of x- and y-coordinates.

For geographic visualizations, the following functions are assumed to exist in the environment:

- GetLatitudes(record), GetLongitudes(record)—Functions that retrieve the arrays of latitude and longitude coordinates, respectively, of the geographic polygon associated with the given record. For example, these polygons could be outlines of the countries of the world.
- ProjectLatitudes(lats, scale), ProjectLongitudes(longs, scale) —Functions that project arrays of latitude values to arrays of y values, and arrays of longitude values to arrays of x values, respectively.
- For graph and 3D surface data sets, the following is provided:

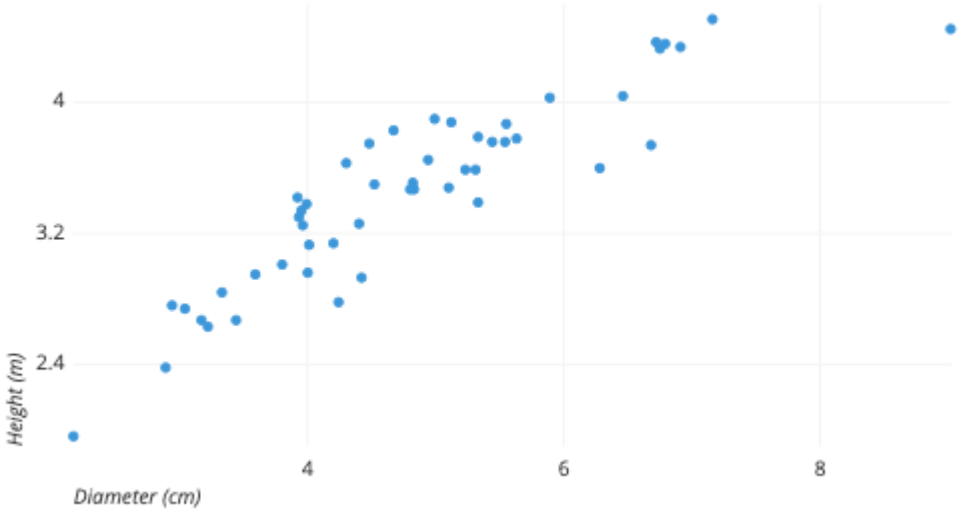
• GetConnections(record)—A function that retrieves an array of record indices to which the given record is connect.

Arrays are indexed starting at zero.

## 1.5. The Scatter plot:

**The scatterplot is one of the earliest and most widely used visualizations developed. It is based on the Cartesian coordinate system.**

**A scatter plot uses dots to represent values for two different numeric variables**. The position of each dot on the horizontal and vertical axis indicates values for an individual data point. Scatter plots are used to observe relationships between variables.

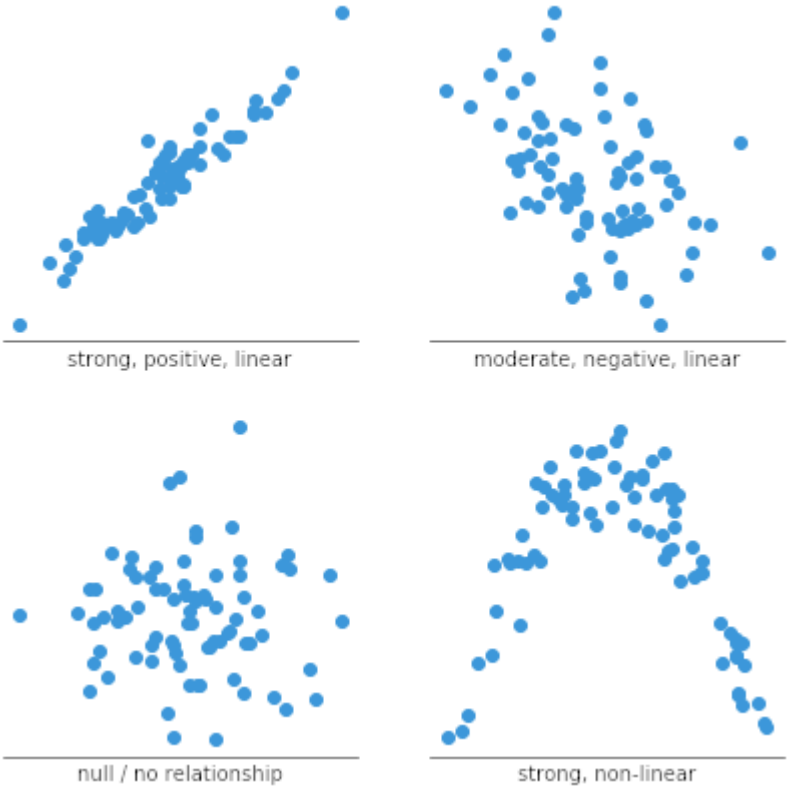



The example scatter plot above shows the diameters and heights for a sample of fictional trees.

Scatter plots' primary uses are to observe and show relationships between two numeric variables. The dots in a scatter plot not only report the values of individual data points, but also patterns when the data are taken as a whole.

Identification of correlational relationships are common with scatter plots. In these cases, we want to know, if we were given a particular horizontal value, what a good prediction would be for the vertical value. You will often see the variable on the horizontal axis denoted an independent variable, and the variable on the vertical axis the dependent variable.

Relationships between variables can be described in many ways: positive or negative, strong or weak, linear or nonlinear.




- The following pseudo-code renders a scatter plot of circles.
- Records are represented in the scatter plot as circles of varying location, color, and size
- The x- and y-axes represent data from dimension numbers xDim and yDim, respectively
- The color of the circles is derived from dimension number cDim
- The radius of the circles is derived from dimension number rDim, as well as from the upper and lower bounds for the radius, rM in and rM ax
- Scatter plot(xDim, yDim, cDim, rDim, rM in, rM ax)
- Step1:- for each record i ✂ For each record,
- Step2:- do x ← Normalize(i, xDim) ✂ derive the location,
- Step3:- y ← Normalize(i, yDim)

- Step4:- r ← Normalize(i, rDim, rM in, rM ax) ✂ radius,
- Step5:- MapColor(i, cDim) ✂ and color, then
- Step6:- Circle(x, y, r) ✂ draw the record as a circle.

## 1.6. Data Foundations:

Since every visualization starts with the data that is to be displayed**, a first step in addressing the design of visualizations is to examine the characteristics of the data.** Data comes from many sources; it can be gathered from sensors or surveys, or it can be generated by simulations and computations. Data can be raw (untreated), or it can be derived from raw data via some process, such as smoothing, noise removal, scaling, or interpolation. It also can have a wide range of characteristics and structures.

A typical data set used in visualization consists of a list of n records,

(r1, r2,...,rn)

Each record ri consists of m (one or more) observations or variables,

(v1, v2,...vm)

- An observation may be a single number/symbol/ string or a more complex structure (discussed in more detail later in this chapter)
- A variable may be classified as either independent or dependent. An independent variable ivi is one whose value is not controlled or affected by another variable, such as the time variable in a time-series data set
- A dependent variable dvj is one whose value is affected by a variation in one or more
- associated independent variables
- Temperature for a region would be considered a dependent variable, as its value could be affected by variables such as date, time, or location
- Thus we can formally represent a record as
- ri = (iv1, iv2, . . . ivmi , dv1, dv2, . . . dvmd ),
- where mi is the number of independent variables and md is the number of dependent variables
- With this notation we have, m = mi + md

- In many cases, we may not know which variables are dependent or independent

**Types of Data**

- In its simplest form, each observation or variable of a data record represents a single piece of information. We can categorize this information as being ordinal (numeric) or nominal (nonnumeric). Subcategories of each can be readily defined.
- **Ordinal**. The data take on numeric values:

  **binary**—assuming only values of 0 and 1;

  **discrete**—taking on only integer values or from a specific subset (e.g., (2, 4, 6)); **continuous**—representing real values (e.g., in the interval [0, 5]).

  **Nominal**

  The data take on nonnumeric values:

- **Categorical**—a value selected from a finite (often short) list of possibilities (e.g., red, blue, green);
- **Ranked**—a categorical variable that has an implied ordering (e.g., small, medium, large);
- **Arbitrary**—a variable with a potentially infinite range of values with no implied ordering (e.g., addresses)
- Another method of categorizing variables is by using the mathematical concept of scale.

  **Scale**

  Three attributes that define a variable's measure are as follows:

- **Ordering relation**, with which the data can be ordered in some fashion By definition, ranked nominal variables and all ordinal variables exhibit this relation
- **Distance metric**, with which the distances can be computed between different records. This measure is clearly present in all ordinal variables, but is generally not found in nominal variables.

- **Existence of absolute zero**, in which variables may have a fixed lowest value. This is useful for differentiating types of ordinal variables. A variable such as weight possesses an absolute zero, while bank balance does not. A variable possesses an absolute zero if it makes sense to apply all four mathematical operations (+, −, ×, ÷) to it [129].

## 1.7.Structure within and between Records

- Data sets have structure, both in terms of the means of representation (syntax ), and the types of interrelationships within a given record and between records (semantics)

**Scalars, Vectors, and Tensors**

- Scalars and vectors are simple variants on a more general structure known as a tensor
- A tensor is defined by its rank and by the dimensionality of the space within which it is defined. It is generally represented as an array or matrix
- A scalar is a tensor of rank 0, while a vector is a tensor of rank 1. One could use a 3 × 3 matrix to represent a tensor of rank 2 in 3D space, and in general, a tensor of rank M in D-dimensional space requires DM data values
- An example of a tensor that might be found in a data record would be a transformation matrix to specify a local coordinate system.

**Geometry and Grids:**

Geometric structure can commonly be found in data sets, especially those from scientific and engineering domains. The simplest method of incorporating geometric structure in a data set is to have explicit coordinates for each data record. Thus, a data set of temperature readings from across the country might include the longitude and latitude associated with the sensors, as well as the sensor values. In modeling of 3D objects, the geometry constitutes the majority of the data, with coordinates given for each vertex.

There are many different coordinate systems that are used for gridstructured data, including Cartesian, spherical, and hyperbolic

coordinates. Often, the choice of coordinate system is domain-specific, and is partially dependent on how the data is acquired/computed, and on the structure of the space in which the data resides.

Nonuniform, or irregular , geometry is also common.

**Other Forms of Structure:**

A timestamp is an important attribute that can be associated with a data record. Time perhaps has the widest range of possible values of all aspects of a data set, since we can refer to time with units from picoseconds to millennia. It can also be relative or absolute in terms of its base value. Data sets with temporal attributes can be uniformly spaced, such as in the sampling of a continuous phenomenon, or nonuniformly spaced, as in a business transaction database.

The following are examples of various structured data:

MRI (magnetic resonance imagery). Density (scalar), with three spatial attributes, 3D grid connectivity;

CFD (computational fluid dynamics). Three dimensions for displacement, with one temporal and three spatial attributes, 3D grid connectivity (uniform or nonuniform)

Financial. No geometric structure, n possibly independent components, nominal and ordinal, with a temporal attribute;

CAD (computer-aided design). Three spatial attributes with edge and polygon connections, and surface properties.

Remote sensing. Multiple channels, with two or three spatial attributes, one temporal attribute, and grid connectivity;

Census. Multiple fields of all types, spatial attributes (e.g., addresses), temporal attribute, and connectivity implied by similarities in fields;

Social Network. Nodes consisting of multiple fields of all types, with various connectivity attributes that could be spatial, temporal, or dependent on other attributes, such as belonging to the same group or having some common computed values.

•

## 1.8. Data Preprocessing

In most circumstances, it is preferable to view the original raw data. In many domains, such as medical imaging, the data analyst is often opposed to any sort of data modifications, such as filtering or smoothing, for fear that important information will be lost or deceptive artifacts will be added.

Viewing raw data also often identifies problems in the data set, such as missing data, or outliers that may be the result of errors in computation or input.

Depending on the type of data and the visualization techniques to be applied, however, some forms of preprocessing might be necessary.

**common methods for preprocessing data:**

**Metadata and Statistics**

Information regarding a data set of interest (its metadata) and statistical analysis can provide invaluable guidance in preprocessing the data.

Metadata may provide information that can help in its interpretation, such as the format of individual fields within the data records.

It may also contain the base reference point from which some of the data fields are measured, the units used in the measurements, the symbol or number used to indicate a missing value (see below), and the resolution at which measurements were acquired.

This information may be important in selecting the appropriate preprocessing operations, and in setting their parameters.

**Missing Values and Data Cleansing**

One of the realities of analyzing and visualizing “real” data sets is that they often are missing some data entries or have erroneous entries.

Missing data may be caused by several reasons, including, for example, a malfunctioning sensor, a blank entry on a survey, or an omission on the part of the person entering the data.

Erroneous data is most often caused by human error and can be difficult to detect. In either case, the data analyst must choose a strategy for dealing with these common events.

**Normalization**

**Normalization is the process of transforming a data set so that the results satisfy a particular statistical property.**

A simple example of this is to transform the range of values a particular variable assumes so that all numbers fall within the range of 0.0 to 1.0.

Other forms of normalization convert the data such that each dimension has a common mean and standard deviation.

Normalization is a useful operation since it allows us to compare seemingly unrelated variables.

**Segmentation**

In many situations, the data can be separated into contiguous regions, where each region corresponds to a particular classification of the data.

For example, an MRI data set might originally have 256 possible value point, and then be segmented into specific categories, such as bone, muscle, fat, and skin.

Simple segmentation can be performed by just mapping disjoint ranges of the data values to specific categories. However, in most situations, the assignment of values to a category is ambiguous.

**Sampling and Subsetting**

Often it is necessary to transform a data set with one spatial resolution into another data set with a different spatial resolution.

For example, we might have an image we would like to shrink or expand, or we might have only a small sampling of data points and wish to fill in values for locations between our samples.

In each case, we assume that the data we possess is a discrete sampling of a continuous phenomenon, and therefore we can predict the values at another location by examining the actual data nearest to it.

The process of interpolation is a commonly used resampling method in many fields, including visualization.

**Dimension Reduction**

In situations where the dimensionality of the data exceeds the capabilities of the visualization technique, it is necessary to investigate ways to reduce the data dimensionality, while at the same time preserving, as much as possible, the information contained within.

This can be done manually by allowing the user to select the dimensions deemed most important, or via computational techniques, such as principal component analysis (PCA) , multidimensional scaling (MDS) , Kohonen self-organizing maps (SOMs) , and Local Linear Embedding (LLE)

**Mapping Nominal Dimensions to Numbers**

In many domains, one or more of the data dimensions consist of nominal values.

We may have several alternative strategies for handling these dimensions within our visualizations, depending on how many nominal dimensions there are, how many distinct values each variable can take on, and whether an ordering or distance relation is available or can be derived.

The key is to find a mapping of the data to a graphical entity or attribute that doesn't introduce artificial relationships that don't exist in the data.

**Aggregation and Summarization**

In the event that too much data is present, it is often useful to group data points based on their similarity in value and/or position and represent the group by some smaller amount of data.

This can be as simple as averaging the values, or there might be more descriptive information, such as the number of members in the group and the extents of their positions or values.

**Smoothing and Filtering**

A common process in signal processing is to smooth the data values, to reduce noise and to blur sharp discontinuities.

A typical way to perform this task is through a process known as convolution, which for our purposes can be viewed as a weighted averaging of neighbors surrounding a data point.

**Raster-to-Vector Conversion**

In computer graphics, objects are typically represented by sets of connected, planar polygons (vertices, edges, and triangular or quadrilateral patches), and the task is to create a raster (pixel-level) image representing these objects, their surface properties, and their interactions with light sources and other objects.

In spatial data visualization, our objects can be points or regions, or they can be linear structures, such as a road on a map.

It is sometimes useful to take a raster-based data set, such as an image, and extract linear structures from it.

## 1.9 Data Sets

- **djia-100.xls**. A univariate, nonspatial data set consisting of 100+ years of daily Dow Jones Industrial Averages

  Source—http://www.analyzeindices.com/dow-jones-history.shtml

  Format—Excel spreadsheet. After the header, each entry is of the form YYMMDD followed by the closing value

  Code—file can be viewed with Excel

- **colorado elev.vit**. A two-dimensional, uniform grid, scalar field representing the elevation of a square region of Colorado

  Source—included with the distribution of OpenDX (http://www.opendx.org/)

  Format—binary file with a 268-byte header followed by a 400 × 400 array of 1-byte elevations

Code—file can be rendered with Topo-Surface, a Processing program included in Appendix C and on the book's web site

- **cars.xls, detroit.xls, cereal.xls**. Several multivariate, non-spatial data sets commonly used in multivariate data analysis

  Source—http://lib.stat.cmu.edu

  Format—Excel spreadsheets, mostly with no headers

  Code—files can be viewed with Excel

- **Health-related multivariate data sets**. Subsets from UNICEF's basic indicators by country and the CDC's obesity by state; several multivariate, spatial data sets used with Geospatial Information Systems

  Source—http://OpenIndicators.org

  Format—Excel spreadsheets

  Code—files can be viewed with Excel, with Weave, and with many other visualization systems.

- **iris.csv. Size** data for Iris plants classified by type

  Source—http://archive.ics.uci.edu/ml/datasets/Iris

  Format—comma-separated values

  Code—The file can be viewed with Excel or as text

**city temp.xls**. A two-dimensional, nonuniform, geo-spatial, scalar data set containing the average January temperature for 56 U.S. cities. Source—Peixoto, J.L. (1990), "A Property of Well-Formulated Polynomial Regression Models." American Statistician, 44, 26–30. Also found in: Hand, D.J., et al. (1994) A Handbook of Small Data Sets, London: Chapman & Hall, 208–210. Downloaded from http://lib.stat .cmu.edu

**uvw.dat.** A three-dimensional uniform grid vector field representing a simulated flow field. The data shows one frame of the unsteady velocity field in a turbulent channel flow, computed by a finite volume method. The streamwise velocity (u) is much larger than the secondary velocities in the transverse direction (v and w).

Source—Data courtesy of Drs. Jiacai Lu and Gretar Tryggvason, ME Department, Worcester Polytechnic Institute (http://www.me.wpi .edu/Tryggvason).

Format—plain text. After the header, each entry is a set of 6 float values, 3 for position, 3 for displacement. There is roughly a 20:1:1 ratio between the 3 displacements.

Code—file can be rendered with FlowSlicer, a Processing program, and FlowView, a Java program (also need Voxel.java)