4. Basic Mapping Techniques

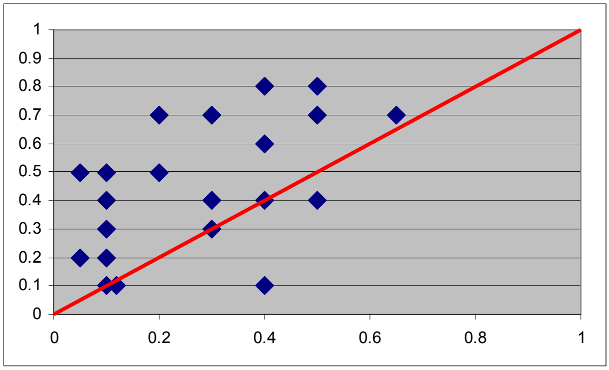
- Mapping from (filtered) data to renderable representation
- Most important part of visualization
- Possible visual representations:
  - Position
  - Size
  - Orientation
  - Shape
  - Brightness
  - Color (hue, saturation)
  - ….

4. Basic Mapping Techniques

- Efficiency and effectiveness depends on input data:
  - Nominal
  - Ordinal
  - Quantitative
- Psychological investigations to evaluate the appropriateness of mapping approaches

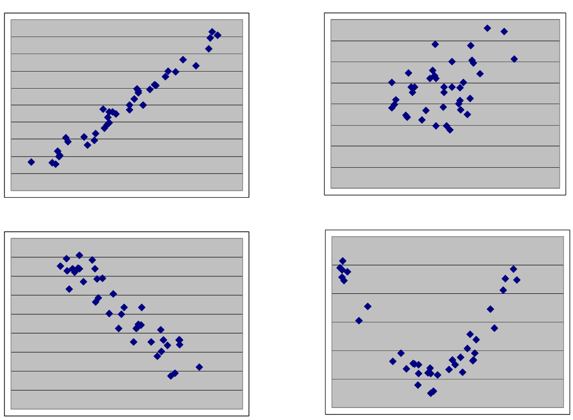
4.1. Diagram Techniques

- Scatter plots:
  - Quantitative data
  - Accuracy in recognizing positions



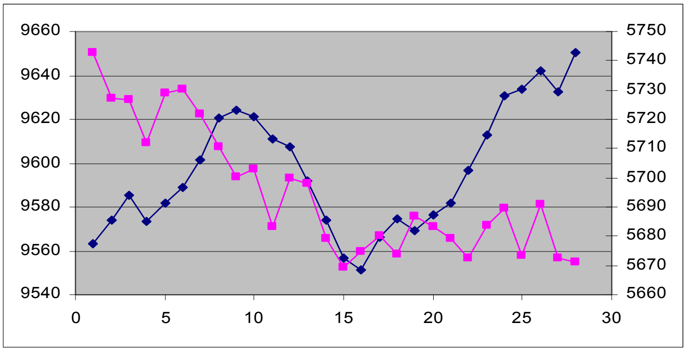
4.1. Diagram Techniques

- Scatter plots: visual recognition of correlations



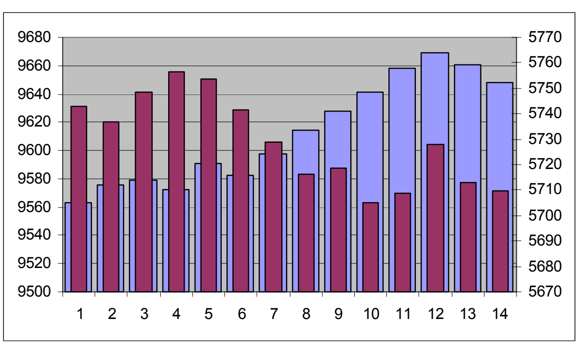
4.1. Diagram Techniques

- Line graph:
  - Connection between points



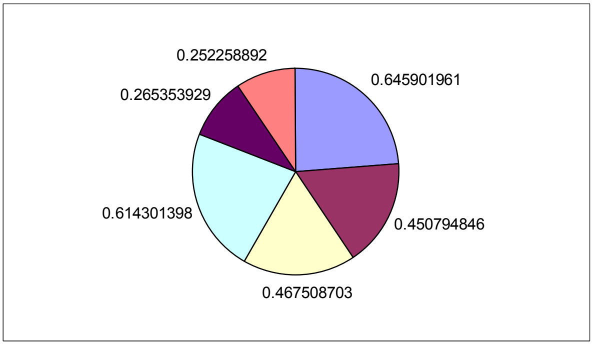
4.1. Diagram Techniques

- Bar graph:
  - Discrete independent variable (domain): nominal/ordinal/quantitative
  - Quantitative dependent variable (data)



4.1. Diagram Techniques

- Pie charts:
  - Quantitative data that adds up to a (fixed?) number

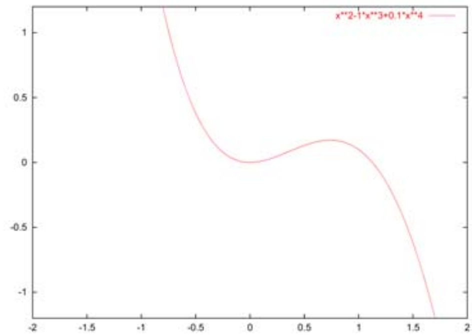


4.2. Function Plots and Height Fields

- Visualization of 1D or 2D scalar fields
  - 1D scalar field: $\Omega \subset \mathbb{R} \rightarrow \mathbb{R}$
  - 2D scalar field: $\Omega \subset \mathbb{R}^2 \rightarrow \mathbb{R}$

4.2. Function Plots and Height Fields

- Function plot for a 1D scalar field
  - Points $\{(s, f(s)) \mid s \in \mathbb{R}\}$
  - 1D manifold: line
  - Is interpolation meaningful?
  - Axis labeling
  - Annotations
  - Error bars



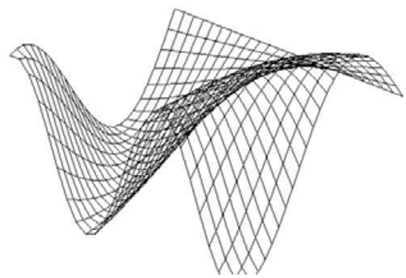
Gnuplot example

4.2. Function Plots and Height Fields

- Function plot for a 2D scalar field
  - Points $\{(s, t, f(s,t) \mid (s,t) \in \mathbb{R}^2\}$
  - 2D manifold: surface
- Surface representations
  - Wireframe
  - Hidden lines
  - Shaded surface

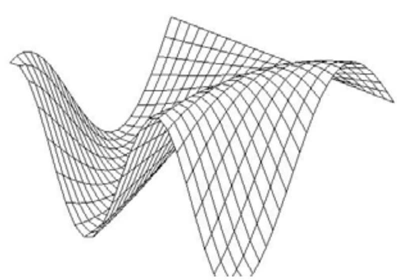
4.2. Function Plots and Height Fields

- Wireframe representation
  - Requires specification of viewing parameters
  - Draw each edge of the grid



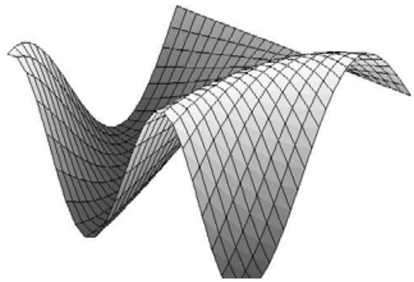
4.2. Function Plots and Height Fields

- Hidden line representation
  - Remove edges that belong to hidden faces
  - Better spatial orientation due to occlusion

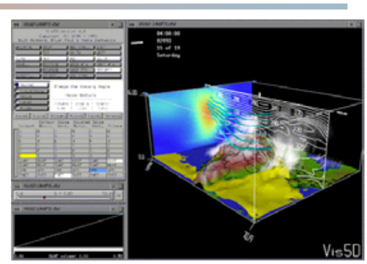


4.2. Function Plots and Height Fields

- Shaded surface
  - Requires specification of lighting/shading model



4.3. Isolines

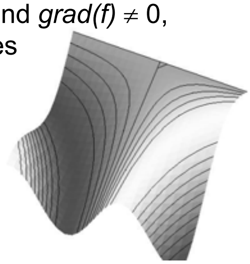


- Visualization of 2D scalar fields
- Given a scalar function $f : \Omega \mapsto \mathbb{R}$

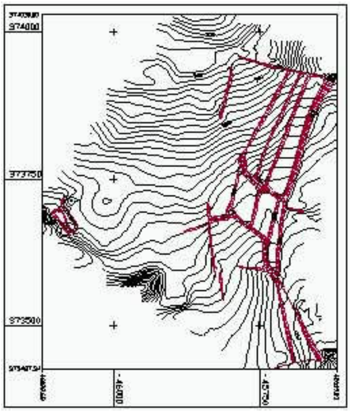
  and a scalar value $c \in \mathbb{R}$
- Isoline consists of points

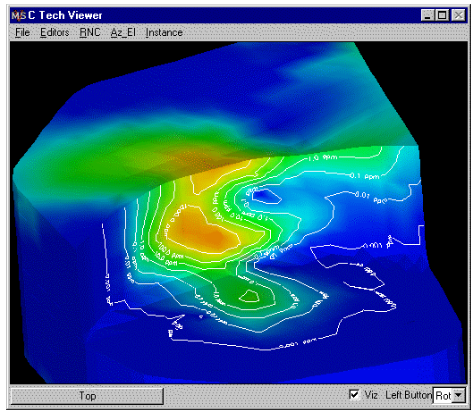
$$\{(x,y) \mid f(x,y) = c\}$$

- If *f()* is differentiable and *grad(f)* $\neq 0$, then isolines are curves
- Contour lines



4.3. Isolines





4.3. Isolines

- Pixel by pixel contouring
- Straightforward approach: scanning all pixels for equivalence with isovalue
- Input
  - $f : (1,\ldots,x_{max}) \times (1,\ldots,y_{max}) \rightarrow \mathbf{R}$
  - Isovalues $I_1,\ldots, I_n$ and isocolors $c_1,\ldots,c_n$
- Algorithm

  for all $(x,y) \in (1,\ldots,x_{max}) \times (1,\ldots,y_{max})$ do

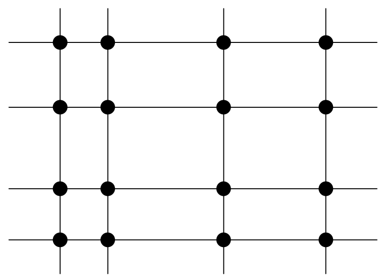
      for all $k \in \{1,\ldots,n\}$ do

          if $|f(x,y)-I_k| < \varepsilon$ then

              draw (x,y,c_k)
- Problem: Isoline can be missed if the gradient of *f()* is too large (despite range ε)

4.3. Isolines

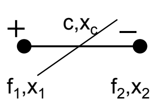
- Marching squares
  - Representation of the scalar function on a rectilinear grid
  - Scalar values are given at each vertex $f \leftrightarrow f_{ij}$
  - Take into account the interpolation within cells
  - Isolines cannot be missed
  - Divide and conquer: consider cells independently of each other

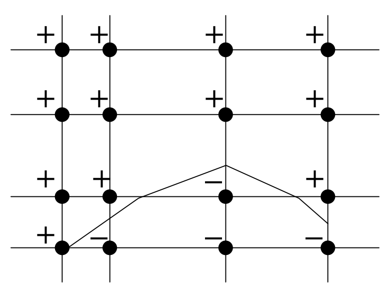


4.3. Isolines

- Which cells will be intersected ?
  - Initially mark all vertices by + or – , depending on the conditions $f_{ij} \geq c$, $f_{ij} < c$
- No isoline passes through cells (=rectangles) which have the same sign at all four vertices
  - So we only have to determine the edges with different signs
  - And find intersection by linear interpolation

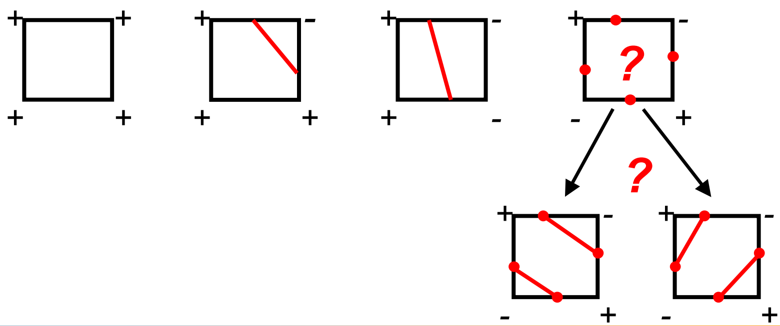
$$x_c = [(f_2 - c)x_1 + (c - f_1)x_2] / (f_2 - f_1)$$





4.3. Isolines

- Only 4 different cases (classes) of combinations of signs
- Symmetries: rotation, reflection, change + ↔ -
- Compute intersections between isoline and cell edge, based on linear interpolation along the cell edges



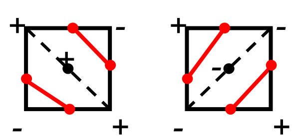
4.3. Isolines

- We can distinguish the cases by a decider

- Mid point decider
  - Interpolate the function value in the center

$$f_{\text{center}} = \frac{1}{4}(f_{i,j} + f_{i+1,j} + f_{i,j+1} + f_{i+1,j+1})$$

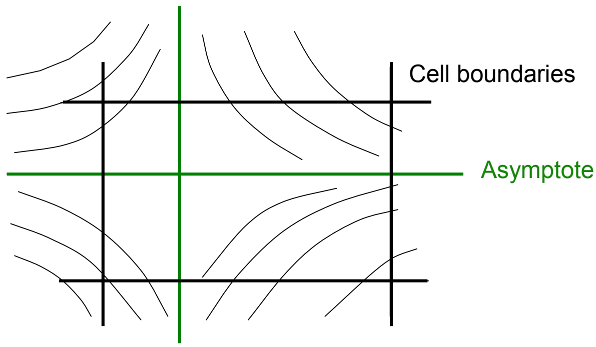
  - If $f_{center} < c$ we chose the right case, otherwise we chose the left case



  - Not always correct solution

4.3. Isolines

- Asymptotic decider
  - Consider the bilinear interpolant within a cell
  - The true isolines within a cell are hyperbolas



4.3. Isolines

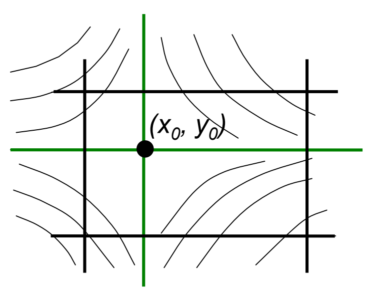
- Interpolate the function bilinearly

$$f(x,y) = f_{i,j}(1-x)(1-y) + f_{i+1,j}x(1-y) + f_{i,j+1}(1-x)y + f_{i+1j+1}xy$$

- Transform *f()* to

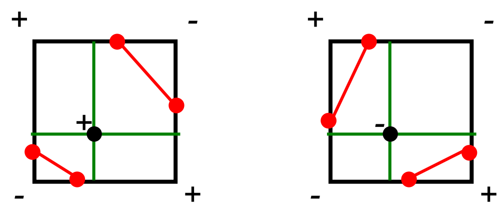
$$f(x,y) = \eta(x-x_0)(y-y_0) + \gamma$$

- γ is the function value in the intersection point of the asymptotes



4.3. Isolines

- If $\gamma \leq c$ we chose the right case, otherwise we chose the left one

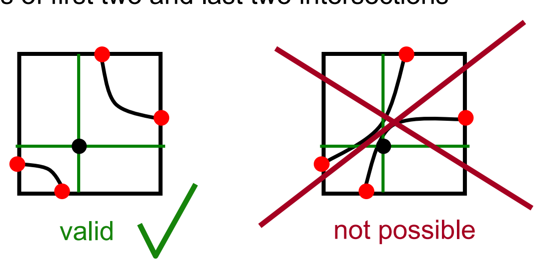


4.3. Isolines

- Explicit transformation *f()* to

$$f(x,y) = \eta(x - x_0)(y - y_0) + \gamma$$

  can be avoided
- Idea: investigate the order of intersection points either along x or y axis
- Build pairs of first two and last two intersections



4.3. Isolines

- Cell order approach for marching squares
  - Traverse all cells of the grid
  - Apply marching squares technique to each single cell
- Disadvantage of cell order method
  - Every vertex (of the isoline) and every edge in the grid is processed twice
  - The output is just a collection of pieces of isolines which have to be post-processed to get (closed) polylines

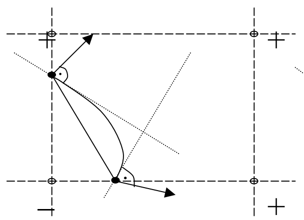
4.3. Isolines

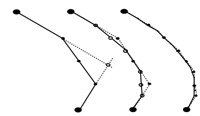
- Contour tracing approach
  - Start at a seed point of the isoline
  - Move to the neighboring cell into which the isoline enters
  - Trace isoline until
    - Bounds of the domain are reached or
    - Isoline is closed
- Problem: How to find seed points efficiently?
  - In a preprocessing step, mark all cells which have a sign change
  - Remove marker from cells which are traversed during contour tracing (unless there are 4 intersection edges!)

4.3. Isolines

- How to smooth isolines
  - Evaluate the gradient at vertices by central differences
  - Estimate tangents at the intersection points by linear interpolation (Note that the gradient is perpendicular to the isoline !)
  - Draw a parabolic arc which is tangential to the estimated tangents at the intersections
    - Quadratic Bezier curve
    - Approximation with 2-3 subdivision steps is sufficient

Quadratic Bezier





Quadratic Bezier approximation with subdivision

4.4. Color Coding

- Issues:
  - What kind of data can be color-coded?
  - What kind of information can be efficiently visualized?

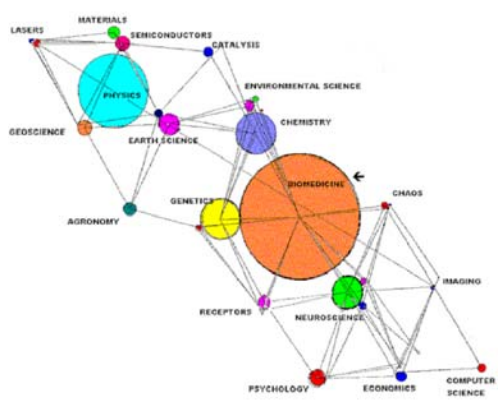
- Areas of application
  - Provide information coding
  - Designate or emphasize a specific target in a crowded display
  - Provide a sense of realism or virtual realism
  - Provide warning signals or signify low probability events
  - Group, categorize, and chunk information
  - Convey emotional content
  - Provide an aesthetically pleasing display

4.4. Color Coding

- Possible problems:
  - Distract the user when inadequately used
  - Dependent on viewing and stimulus conditions
  - Ineffective for color deficient individuals (use redundancy)
  - Results in information overload
  - Unintentionally conflict with cultural conventions
  - Cause unintended visual effects and discomfort

4.4. Color Coding

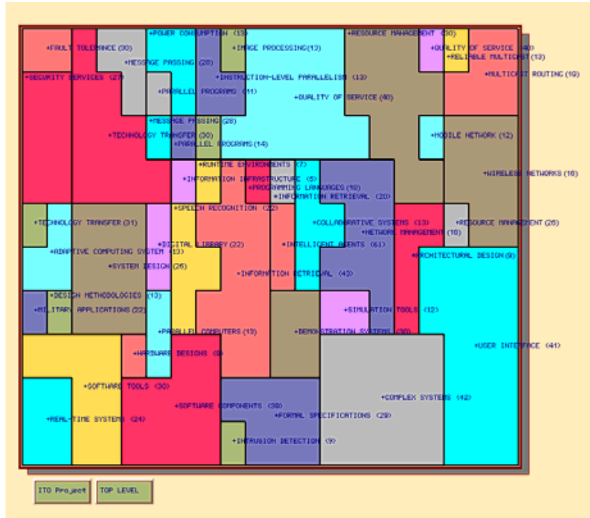
- Nominal data
  - Colors need to be distinguished
  - Localization of data
  - Around 8 different basis colors



co-citation analysis

4.4. Color Coding

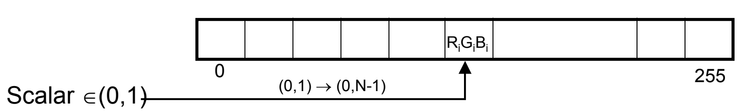
- Nominal data



4.4. Color Coding

- Ordinal and quantitative data
  - Ordering of data should be represented by ordering of colors
  - Psychological aspects
  - $x_1 < x_2 < \ldots < x_n \Rightarrow E(c_1) < E(c_2) < \ldots < E(c_n)$
- Color coding for scalar data
  - Assign to each scalar value a different color value
  - Assignment via transfer function T

    $T : scalarvalue \rightarrow colorvalue$
  - Code color values into a color lookup table



4.4. Color Coding

- Pre-shading vs. post-shading
  - Pre-shading
    - Assign color values to original function values (e.g. at vertices of a cell)
    - Interpolate between color values (within a cell)
  - Post-shading
    - Interpolate between scalar values (within a cell)
    - Assign color values to interpolated scalar values
- Linear transfer function for color coding
  - Specify color for f_{min} and for f_{max}
    - $(R_{min}, G_{min}, B_{min})$ and $(R_{max}, G_{max}, B_{max})$
  - Linearly interpolate between them

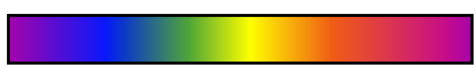
$$f \mapsto \frac{f - f_{min}}{f_{max} - f_{min}}(R_{min}, G_{min}, B_{min}) + \frac{f_{max} - f}{f_{max} - f_{min}}(R_{max}, G_{max}, B_{max})$$

4.4. Color Coding

- Different color spaces lead to different interpolation functions
- In order to visualize (enhance/suppress) specific details, non-linear color lookup tables are needed

- Gray scale color table
  - Intuitive ordering

- Rainbow color table
  - Less intuitive
  - HSV color model



- Temperature color table

4.4. Color Coding

- Bivariate and trivariate color tables are not very useful:
  - No intuitive ordering
  - Colors hard to distinguish

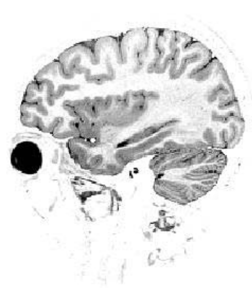
- Many more color tables for specific applications
- Design of good color tables depends on psychological / perceptional issues

- Often interactive specification of transfer functions to extract important features

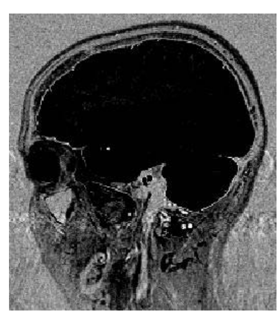
4.4. Color Coding

- Example
  - Special color table to visualize the brain tissue
  - Special color table to visualize the bone structure

Original



Brain



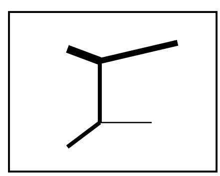
Tissue

4.5. Glyphs and Icons

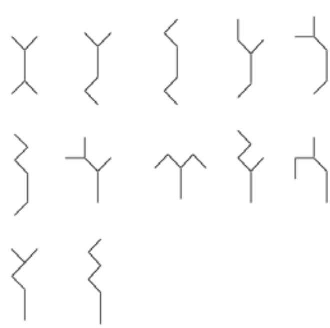
- Glyphs and icons
- Features should be easy to distinguish and combine
- Icons should be separated from each other
- Mainly used for multivariate data

4.5. Glyphs and Icons

- Stick-figure icon [Picket & Grinstein 88]

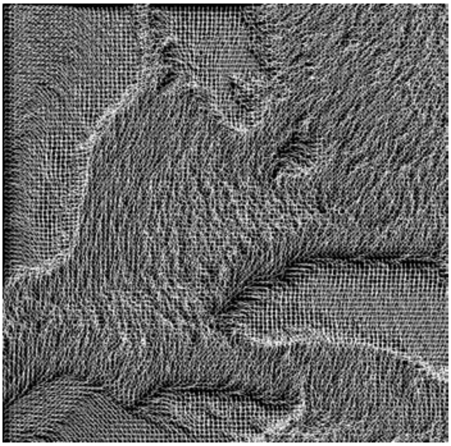


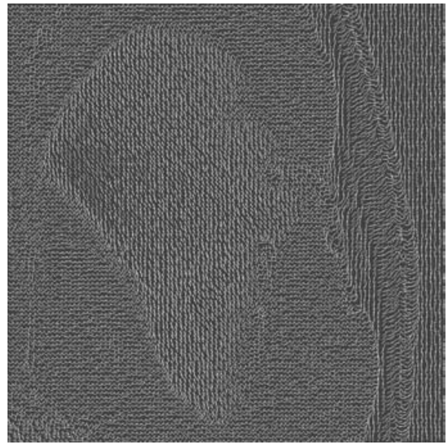
- 2D figure with 4 limbs
- Coding of data via
  - Length
  - Thickness
  - Angle with vertical axis
- 12 attributes
- Exploits the human capability to recognize patterns/textures



4.5. Glyphs and Icons

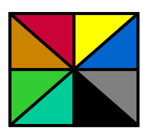
- Stick-figure icon

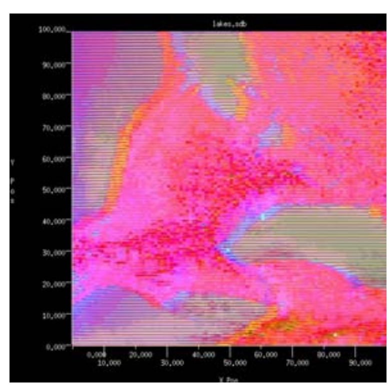




4.5. Glyphs and Icons

- Color icons [Levkowitz 91]
- Subdivision of a basic figure (triangle, square, …) into edges and faces
- Mapping of data to faces via color tables
- Grouping by emphasizing edges or faces

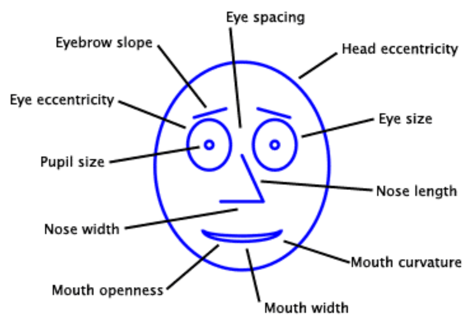




4.5. Glyphs and Icons

- Chernoff faces/icons
  [H. Chernoff (1973). The use of faces to represent points in k-dimensional space graphically. *Journal of the American Statistical Association* , *68* :361-368]

- Each facial feature represents one variable
- Human ability to distinguish small features in faces

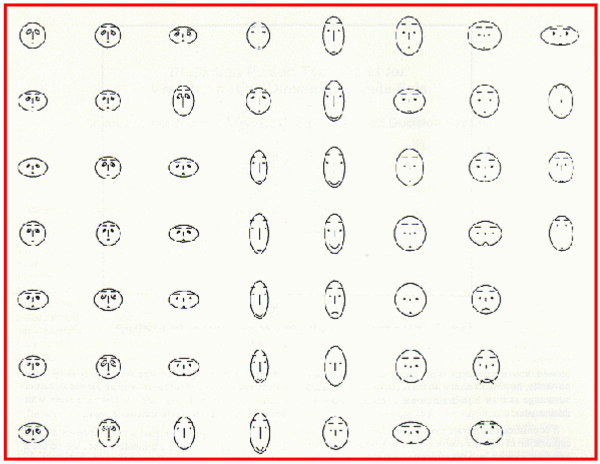


4.5. Glyphs and Icons

- Chernoff faces/icons
- Possible assignment in the decreasing order of importance:
  - Area of the face
  - Shape of the face
  - Length of the nose
  - Location of the mouth
  - Curve of the smile
  - Width of the mouth
  - Location, separation, angle, shape, and width of the eyes
  - Location of the pupil
  - Location, angle, and width of the eyebrows
- Coding of 15 attributes
- Additional variables could be encoded by making faces asymmetric

4.5. Glyphs and Icons

- Chernoff faces/icons



4.5. Glyphs and Icons

- Face morphing [Alexa 98]

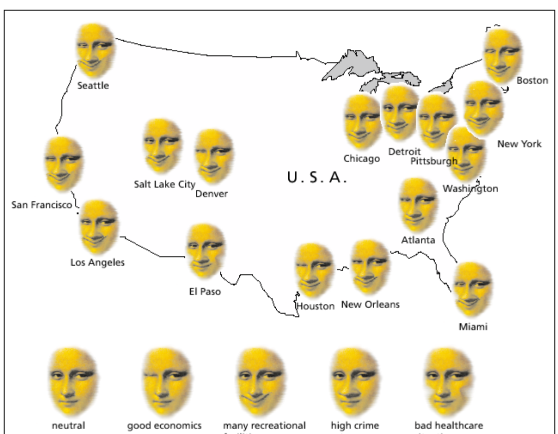


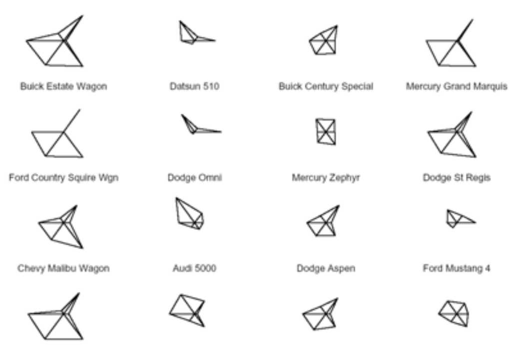
Figure 1: Distorted Mona Lisa faces as visual representations of city ratings in the U.S.

4.5. Glyphs and Icons

- Circular icon plots:
  - Star plots
  - Sun ray plots
  - etc…
- Follow a "spoked wheel" format
- Values of variables are represented by distances between the center ("hub") of the icon and its edges

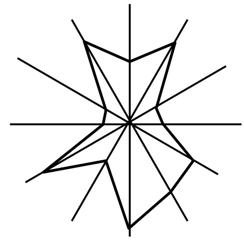
4.5. Glyphs and Icons

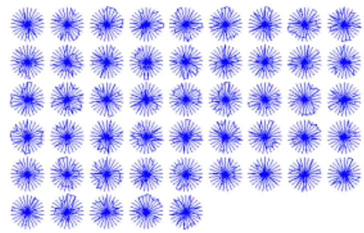
- Star glyphs
  [S. E. Fienberg: Graphical methods in statistics.
  The American Statistician, 33:165-178, 1979]
  - A star is composed of equally spaced radii, stemming from the center
  - The length of the spike is proportional to the value of the respective attribute
  - The first spike/attribute is to the right
  - Subsequent spikes are counter-clockwise
  - The ends of the rays are connected by a line



4.5. Glyphs and Icons

- Sun ray plots
  - Similar to star glyphs/plots
  - Underlying star-shaped structure





4.6. Multiple Attributes

- Other approach to visualizing multivariate data:
  Map data values to different visual primitives
- Multiple attributes
- Typical combination of attributes:
  - Geometric position, e.g., height field
  - Color: saturation, intensity, tone
  - Texture