



Tema 2: Técnicas de Visualización – Parte 2/2

2.1 Introducción

2.2 Datos de 2 y 3 dimensiones

2.3 Técnicas de Visualización de datos multivariados

2.3.1 Proyección Geométrica

2.3.2 Basadas en Píxeles

2.3.3 Jerárquicas

2.3.4 Basadas en Iconos

Máster Universitario en Ingeniería Informática
Pablo.Bermejo@uclm.es

Técnicas de Visualización

1



2.3 Datos multivariados

- Registros representados con n variables, siendo n cualquier valor mayor que 3. Si es 3 ó menos, conviene utilizar las técnicas anteriores ya que son más intuitivas.
- 4 categorías de Técnicas de Visualización para datos multivariados de n dimensiones y m registros:
 - **Proyección Geométrica:** proyección de esas n dimensiones a un espacio bidimensional; u organización de las dimensiones en pares de dos.
 - **Basadas en Píxeles:** habrá n marcos y m píxeles en cada marco, donde el color del documento representado por un pixel dependerá del valor de la variable que representa el marco.
 - **Jerárquicas:** el espacio bidimensional donde se realiza la representación se particiona en tantos subespacios como sean necesarios para representar las n dimensiones.
 - **Basadas en Iconos:** las variables se presentan con pictogramas

Técnicas de Visualización

2



2.3.1 Datos multivariados – Proyección Geométrica

PROYECCIÓN GEOMÉTRICA

- Se **transforma el espacio** multivariado a otro de menor dimensiones:
 - Espacio cartesiano
 - Espacio donde las coordenadas se representan de forma no convencional
- Muy **eficiente** cuando se dispone de muchas variables.
- El **orden** de las variables afecta a la representación
- Los registros pueden aparecer **superpuestos** en la representación
- **Técnicas:**
 - Matriz de dispersión
 - Matriz de proyecciones
 - HyperSlice
 - Hyperbox
 - Coordenadas Paralelas y Paralelas Circulares
 - Curva de Andrew
 - Table Lens



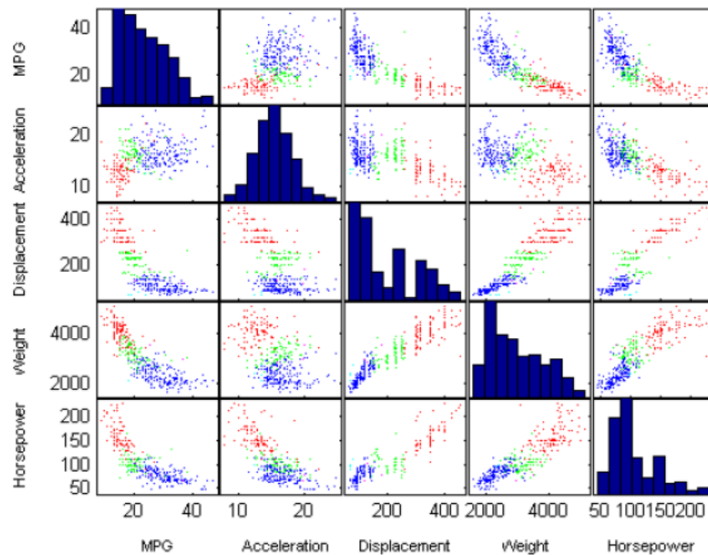
2.3.1 Datos multivariados – Proyección Geométrica

Matriz de dispersión

- Un diagrama de dispersión por cada **par de variables**, formando una matriz
- La **diagonal principal** representa la distribución de cada variable. Si la variable es discreta, entonces es un histograma.
- **Cada punto** dentro de un diagrama representa un registro:
 - Poco eficiente cuando la base de datos es muy grande
- El color de cada punto puede representar el valor de otra variable (categorizada): **brushing**
- Ejemplo:
 - Base de datos con 6 variables:
 - MPG (Miles per Galon)
 - Aceleración
 - Displacement (volumen de aire por ciclo del motor)
 - Peso
 - Horsepower
 - Cilindros

2.3.1 Datos multivariados – Proyección Geométrica

- Hacemos una matriz de dispersión para las primeras 5 variables
- El número de cilindros se discretiza en 3 intervalos de valores, a los que se les asigna los colores azul, verde y rojo respectivamente.



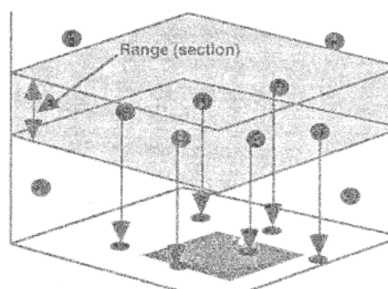
Técnicas de Visualización

5

2.3.1 Datos multivariados – Proyección Geométrica

Matriz de proyecciones [Furnas and B uja. Prosection Views: Dimensional Inference through Sections and Projections. Journal of Computational and Graphical Statistics. Vol 3, N 4. 1994]

- Matriz de proyecciones por cada par de las n variables
- Para cada par de variables, una proyección es una **sección+proyección** de otra variable de interés:
 - Sección: definición de un **rango de valores** en el que queremos que caiga la variable de interés
 - Proyección: proyección a 2D de **únicamente los registros que cumplan** las condiciones de la sección, donde el color del punto depende del grado de satisfacción de las condiciones indicadas para la sección.

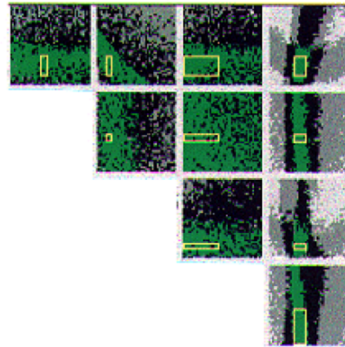


Técnicas de Visualización

6

2.3.1 Datos multivariados – Proyección Geométrica

- La diferencia práctica con la matriz de dispersión es que en este caso no se proyectan todos los registros sino solo los que cumplen los criterios de la sección.
- Rectángulo amarillo: zona de resistencia a diferentes valores; es decir, los registros que aún se seguirían proyectando aunque la sección cambiara mucho: **enhancement**



2.3.1 Datos multivariados – Proyección Geométrica

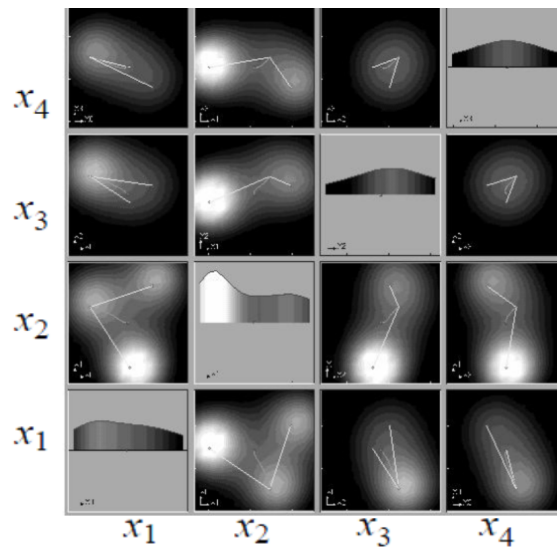
HyperSlice [J.J. van Wijk and R. van Liere. HyperSlice: Visualization of Scalar Functions of Many Variables. Proceedings of the 4th IEEE Conference on Visualization. 1993]

- Variables numéricas
- Hay que **definir una función escalar** f tal que
$$f(x_1, x_2, \dots, x_n) \rightarrow \mathbf{R}$$
- De nuevo, **matriz** de casillas para cada par de variables
- En cada casilla, la **tonalidad** del color en un punto (i,j) es proporcional al valor de f para dichos valores, y teniendo un valor fijo del resto de variables.
- La **diagonal principal** representa, para una variable x , la distribución de f conforme varía x y el resto de variables tienen un valor fijo.

2.3.1 Datos multivariados – Proyección Geométrica

Ejemplo:

-Hay pares de variables (x_4 , x_3) cuyos rangos de valores no parecen cambiar el valor de la función más que en sus valores intermedios.

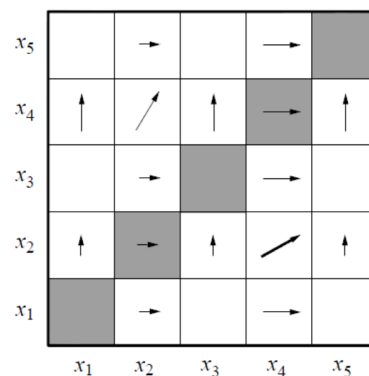


Técnicas de Visualización

9

2.3.1 Datos multivariados – Proyección Geométrica

- El valor fijo de las variables es el valor que tengan en un registro seleccionado.
- Interactuando con el ratón encima de cada casilla, se puede cambiar el registro y ver cómo varían los tonos (valores de la función).



- El camino recorrido por el hiperespacio queda marcado con unas líneas en cada casilla (ver imagen del ejemplo)

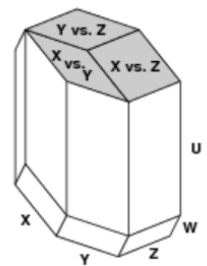
Técnicas de Visualización

10

2.3.1 Datos multivariados – Proyección Geométrica

HyperBox. [Alpern and Carter. Hyperbox. Proceedings of the 2nd IEEE Conference on Visualization. 1991]

- Una caja *n-dimensional*.
- Cada cara, como en las matrices, representa un par de variables.
- La caja tendrá n^2 segmentos y $n(n-1)/2$ caras.
- Al dibujar sobre una superficie de 2D, **no todas las variables se tratan igual** (el orden aquí es aún más importante que en las técnicas anteriores) ya que no todas las caras son iguales:
 - Se pueden enfatizar variables
- Un **conjunto direccional** es un conjunto de segmentos con la misma dirección
- Se asigna un conjunto direccional a cada variable



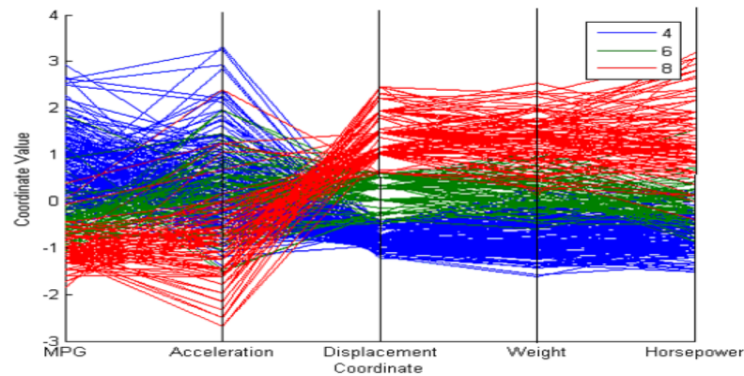
2.3.1 Datos multivariados – Proyección Geométrica

- **Coordenadas Paralelas** [A. Inselberg. "Multidimensional Detective". Proceedings of the IEEE Symposium on Information Visualization. 1997]
- n ejes verticales equidistantes, uno para cada variable y dividido en los posibles valores que la variable puede tomar.
- Hay que escalar los posibles valores en cada eje para que todos los ejes tengan la **misma altura** (escala).
- Cada **registro** se representa por una **línea poligonal** que corta cada eje vertical por el valor que toma la variable correspondiente en dicho registro.

Volvemos al ejemplo de los cilindros:

- MPG (Miles per Galon)
- Aceleración
- Displacement (volumen de aire por ciclo del motor)
- Peso
- Horsepower
- Cilindros (usando brushing)

2.3.1 Datos multivariados – Proyección Geométrica



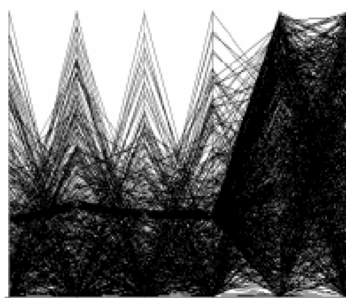
- Muy efectivo para mostrar relaciones entre atributos
- Problemas:
 - El orden de las variables es extremadamente importante para encontrar patrones
 - Al tener muchas variables, se necesita mucho espacio horizontal

2.3.1 Datos multivariados – Proyección Geométrica

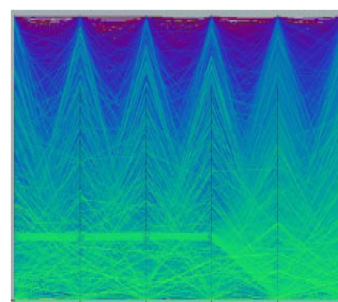
- Representación con 15000 registros



- Soluciones:
 - (1) Selección de registros

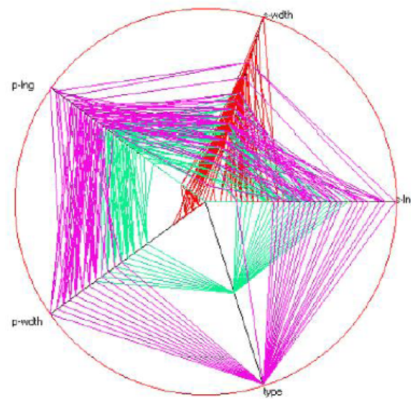


- (2) Brushing



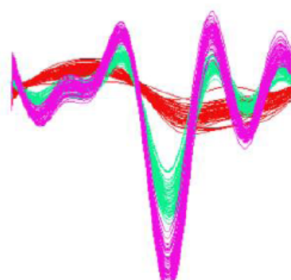
2.3.1 Datos multivariados – Proyección Geométrica

- **Coordenadas Paralelas Circulares** [P.E.Hoffman. Table Visualizations: A Formal Model and Its Applications. Doctoral Dissertation, University of Massachusetts at Lowell. 1999]
- Igual que las anteriores pero los ejes son radios de un círculo
- Ejemplo (usando brushing):



2.3.1 Datos multivariados – Proyección Geométrica

- **Curva de Andrew** [D.F. Andrews. Plots of High-Dimensional Data". Biometrics. 1972]
- Cada registro se transforma en una línea curva, mediante una transformación matemática (ej, transformada de Fourier).
- Registros con valores similares en sus variables forman curvas similares:
 - detección de clusters
 - detección de outliers
- Ejemplo (usando brushing):

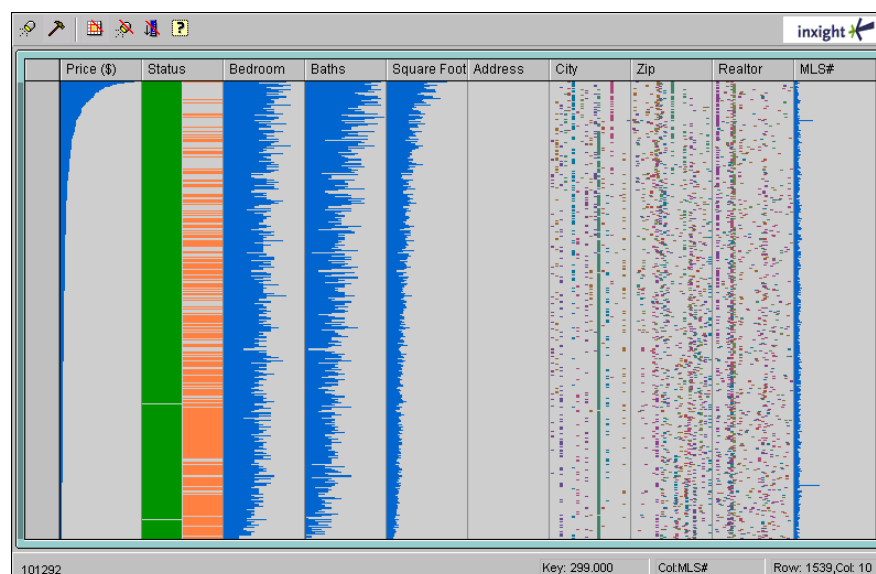


2.3.1 Datos multivariados – Proyección Geométrica

- **Table Lens** [R. Fao and S. K. Card, "The Table Lens: Merging Graphical and Symbolic Representations in an Interactive Focus + Context Visualization for Tabular Information", *Proceedings of the SIGCHI Conference on Human Factors in Computer Systems: Celebrating Interdependence. 1994*]
- Utiliza una distribución de los datos que nos resulta muy familiar: **tabla con los registros en filas y las variables en columnas**
- Pero cada posición (i,j) no es un valor numérico sino una representación gráfica de dicho valor, en forma de **columna horizontal** (si es valor real) o **color** (si es categórico).
- Se puede **interaccionar**, pero esto lo veremos en los temas de Interacción y Distorsión.

2.3.1 Datos multivariados – Proyección Geométrica

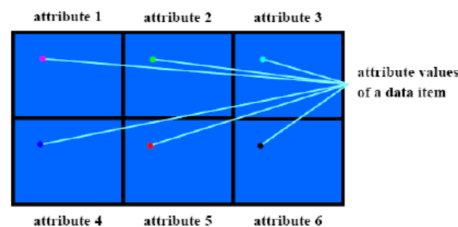
- Table Lens from *InXight*



2.3.2 Basadas en Píxeles

[Daniel A. Keim. Designing Pixel-Oriented Visualization Techniques: Theory and Applications. IEEE Transactions on Visualization and Computer Graphics. Vol 6, Número 1. 2000]

- Si tenemos n variables, se utilizarán n ó $n+1$ ventanas.
- En cada ventana n_i correspondiente a la variable i , se dibuja **un pixel por cada registro**. El color de este pixel dependerá del valor de dicha variable en el registro.
- La decisión de dónde ha de dibujarse cada pixel dentro de una ventana da lugar a diferentes técnicas.
- Las relaciones entre variables se detectan **comparando las mismas zonas entre distintas ventanas**.
- Puesto que a cada variable le corresponde una ventana, y a cada registro una posición:
 - **no hay solapamiento** de variables ni registros.
 - muy útil **para bases de datos masivas** (miles de registros)



Técnicas de Visualización

19

2.3.2 Basadas en Píxeles

- Con lo cual, hay que tomar 3 decisiones:
 - Color de un pixel
 - Tamaño y orden de las ventanas
 - Localización de un pixel
- La técnica de **mapeado de un registro a un color** debe cumplir:
 - Brillo decreciente de forma monótona



- Que genere una escala con todos los colores
- Que la saturación sea fija (viveza en el color). Cuanto mayor es la saturación, da más impresión de movimiento.

Técnicas de Visualización

20



2.3.2 Basadas en Píxeles

- **El tamaño de las ventanas** dependerá de cuántas hay que dibujar.
- El **orden de las ventanas** es muy importante: variables relacionadas deben estar próximas para detectar patrones → selección de variables!
- Las diferentes técnicas para decidir dónde dibujar un píxel pueden dividirse en 2 grupos:
 - *Orientadas a consultas* (query-dependent):
 - los registros (píxeles) siguen un orden basado en una condición indicada por el usuario, y el color dependerá del grado de satisfacción de la consulta.
 - Útil cuando queremos explorar la base de datos de forma interactiva
 - Se calculan distancias entre la consulta y los valores de cada registro
 - *No orientadas a consultas* (query-independent): los registros (píxeles) siguen un orden natural basados en el valor de los atributos (ej: tiempo),



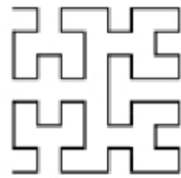
2.3.2 Basadas en Píxeles

- [Query-Independent](#)
- Hay que encontrar un orden apropiado para los píxeles. Este orden debe mostrar un orden natural en el valor de los atributos, para así mostrar tendencias o clusters.
- Y encima el orden será el mismo en todas las ventanas!
- Ordenar de izquierda a derecha o arriba abajo según el valor de un atributo no vale para los otros atributos, ya que el orden debe ser igual.
- Distintas formas de solucionar este **problema matemático de optimización**:
 - Curva de Peano-Hilbert
 - Morton
 - Patrón recursivo

2.3.2 Basadas en Píxeles (query-independent)

- **Space Filling Curve**

- Curva de Peano-Hilbert



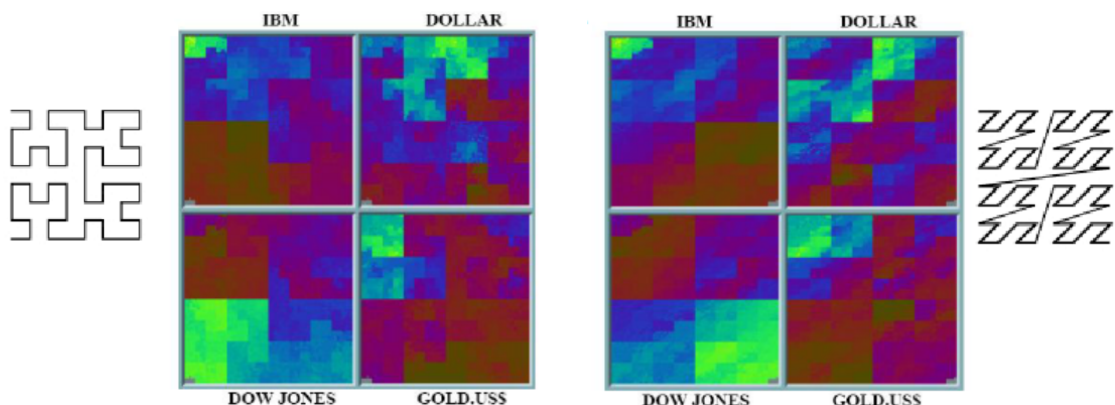
- Curva de Morton (más sencilla de seguir)



- **EJEMPLO** de base de datos con

- 4 variables (más 1 variable de tiempo):
 - IBM: valor de las acciones de IBM
 - DOW JONES: índice de la bolsa de EEUU
 - DOLLAR: cambio del dólar a libras
 - GOLD.US\$: precio del oro
- 7 registros por día
 - Desde Septiembre de 1987 a Febrero 1995
 - Total: 64800 registros

2.3.2 Basadas en Píxeles (query-independent)



- Los píxeles se colocan en **orden según la variable temporal**
- El **lugar** se indica **por el prototipo de curva** seleccionado
- El **color** del píxel indica el **valor del atributo** en el correspondiente registro
- Con atención, los métodos nos permiten encontrar relaciones entre variables.
- ¿El valor de las acciones de IBM siguieron el índice DOW JONES?
- ¿El precio del dólar siguió el precio del oro?
- ¿En qué fechas tuvo su máximo el valor del Dólar? ¿Y el Dow Jones?

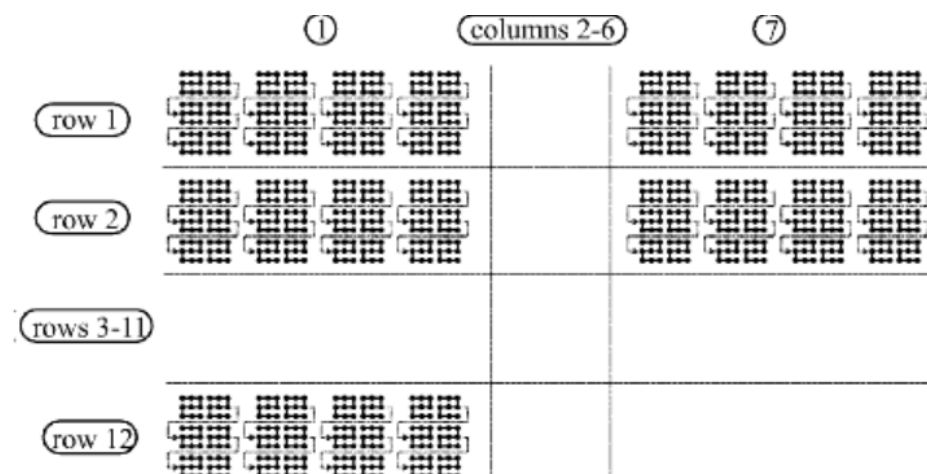
2.3.2 Basadas en Píxeles (query-independent)

- **Patrón Recursivo**
- De izquierda a derecha y de arriba a abajo.
- Agrupar **registros relacionados** entre sí **en bloques**. Por ejemplo, datos recogidos en un mismo día.
- Bloques de bajo nivel construyen bloques de niveles superiores: un bloque con los datos de un día, junto a otros bloques, forman el bloque de los datos de una semana.
- El usuario debe definir:
 - cuántos niveles
 - cuántos registros por nivel.
 - En cuántas filas y columnas se distribuyen los registros de un nivel.

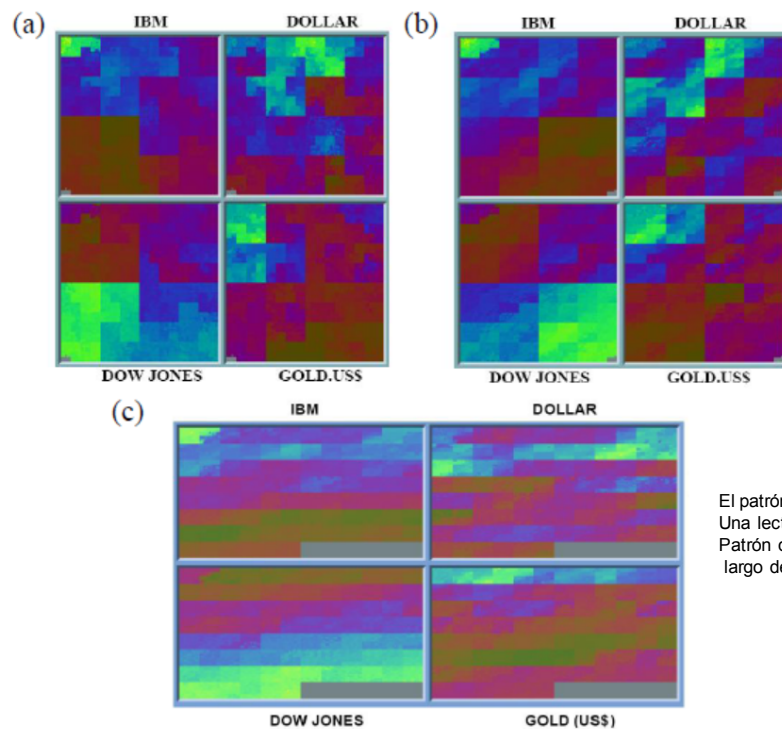
2.3.2 Basadas en Píxeles (query-independent)

- Ejemplo de distribución cuando el usuario selecciona:
 - 5 niveles
 - w: columnas por nivel
 - h: filas por nivel
 - Se divide la pantalla desde el nivel más alto hacia abajo

$[(w_1, h_1) = (3, 3), (w_2, h_2) = (2, 3), (w_3, h_3) = (4, 1), (w_4, h_4) = (1, 12), (w_5, h_5) = (7, 1)]$



2.3.2 Basadas en Píxeles (query-independent)



El patrón recursivo ofrece
Una lectura más intuitiva del
Patrón que siguen los datos a lo
largo del tiempo

27

2.3.2 Basadas en Píxeles

- **Query-Dependent**
- Una **consulta** es un conjunto de condiciones o valores requeridos para las variables.
- La **distancia global** es la distancia entre la condición para todas las variables y el valor de todas las variables de cada registro.

Registro (a_1, a_2, \dots, a_n)

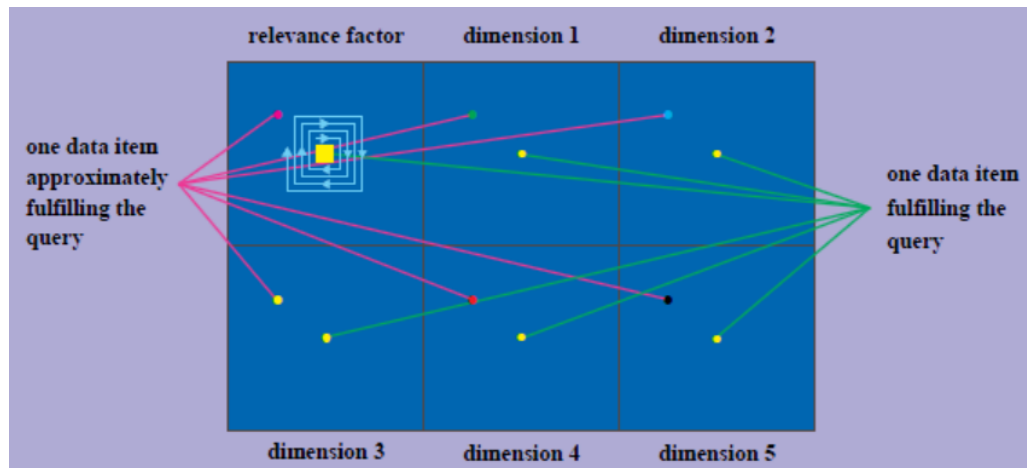
Query (q_1, q_2, \dots, q_n)

→ Distancias (d_1, d_2, \dots, d_n) → distancia global d_{n+1}

- Hay $n+1$ ventanas. La ventana extra se utiliza para la distancia global.
- La **posición** de un píxel en la ventana extra depende de la distancia global.
- La posición de un píxel en el resto de ventanas será la misma que en la ventana extra.
- El color de un píxel para una atributo a_i depende de la distancia d_i

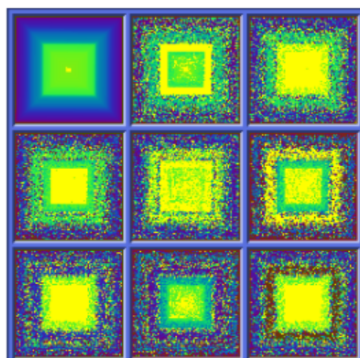
2.3.2 Basadas en Píxeles (query-dependent)

- **Espiral** [D. A. Keim and H.-P. Kriegel, "VisDB: Database Exploration using Multidimensional Visualization", *IEEE Transactions on Computer Graphics and Applications*, vol. 14, no. 5, pp. 40-49, 1994.]
- Partiendo desde el centro, los píxeles se dibujan en espiral.
- La primera ventana representa la distancia global



2.3.2 Basadas en Píxeles (query-dependent)

- Ejemplo de Técnica de Espiral, con 8 variables y 7000 registros.
- El color amarillo significa una distancia pequeña.
- En la ventana de distancia global, el color amarillo solo está en el centro ya que representa d_{n+1} .
- En el resto de ventanas, el color amarillo se usa cuando el valor de la variable a_i es similar al indicado en la consulta según q_i .



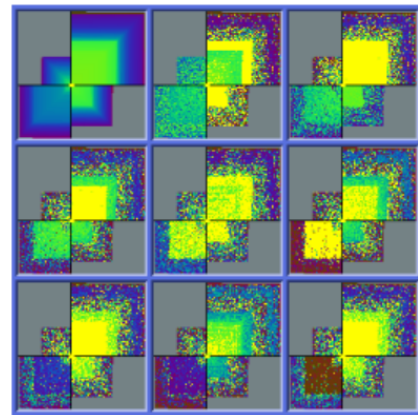
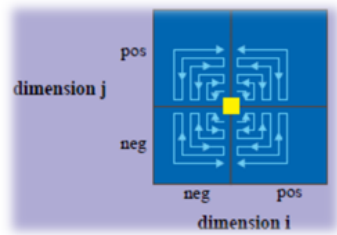
-Casi los mismo registros cumplen las condiciones exigidas para las variables 2, 3, 6 y 8

-La variable 7 (octavo marco) podría eliminarse.

-Visualmente e interaccionando, podemos librarnos de miles de registros a nuestra elección.

2.3.2 Basadas en Píxeles (query-dependent)

- **Ejes o 2D** [D. A. Keim and H.-P. Kriegel, "VisDB: Database Exploration using Multidimensional Visualization", *IEEE Transactions on Computer Graphics and Applications*, vol. 14, no.5, pp.40-49, 1994.]
- Las distancias pueden ser positivas o negativas, y es posible que nos interese centrarnos en alguna de las 2 (ej: casas más caras que x\$).
- Se seleccionan 2 atributos i y j , y la base de datos se particiona en 4 según el signo de las distancias de las variables a_i y a_j a las consultas q_i y q_j .
- Para cada partición, se realiza una distribución es Espiral.

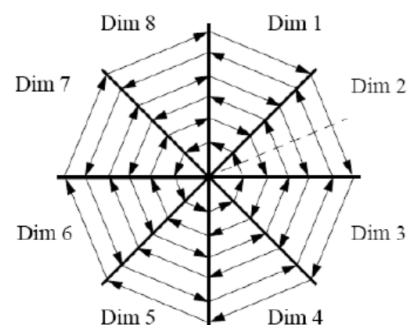
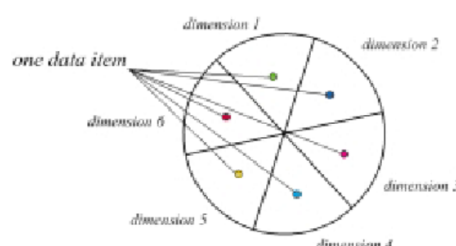


Técnicas de Visualización

31

2.3.2 Basadas en Píxeles (query-dependent)

- **Segmento Circular** [M. Ankerst, D. A. Keim, and H.-P. Kriegel, "Circle Segments: A Technique for Visually Exploring Large Multidimensional Data Sets", *Proceedings of the IEEE Conference on Visualization '96, Hot Topic Session*, 1996.]
- Se divide el círculo **en tantos sectores como variables tengamos**.
- Cada registro se coloca en la **misma posición en cada sector**.
- Esta **posición depende de la distancia global** a la consulta del usuario:
 - Desde el centro los registros más cercanos
 - Rebotando en las paredes del sector
- El **color** del píxel dentro de cada sector i dependerá de la **distancia d_i**
- **Conforme hay más dimensiones**, esta distribución es mejor que la división del espacio con una matriz de cuadrados.

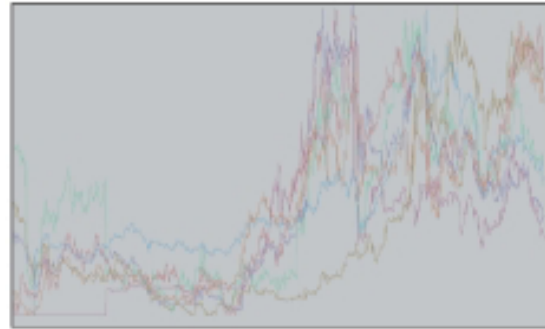
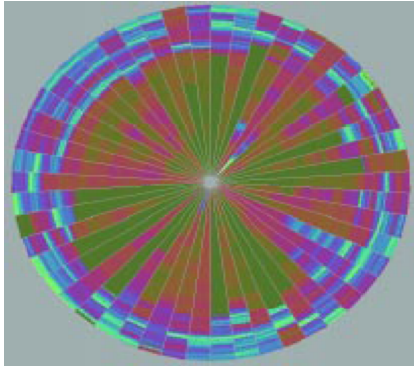


Técnicas de Visualización

32

2.3.2 Basadas en Píxeles (query-dependent)

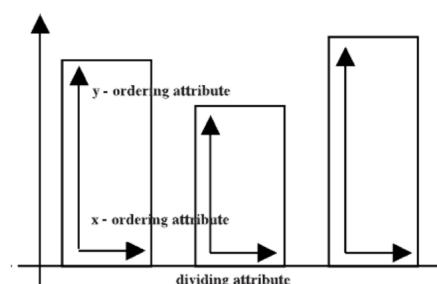
- Ejemplo:
 - Base de datos del índice de bolsa de Frankfurt FAZ
 - 50 variables, una por cada tipo de acción
 - Datos recogidos durante 20 años
 - 265000 registros



- Representación de los datos con Segmento Circular Vs. Diagrama de Líneas
 - Los datos de las 50 variables no se superponen con el Segmento Circular

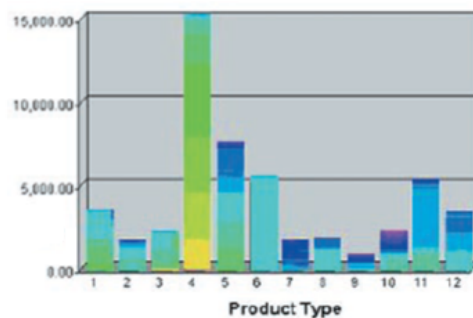
2.3.2 Basadas en Píxeles (query-dependent)

- **Diagrama de Barras Pixeladas** [D. Keim, M. Hao, U. Dayal, M. Hsu and J. Ladisch, "Pixel Bar Charts: A New Technique for Visualizing Large Multi-Attribute Data Sets without Aggregation", *Proceedings of the IEEE Symposium on Information Visualization 2001 (INFOVIS'01)*, pp.113-120, 2001.]
- Parte de un histograma creado para una variable nominal, que tiene tantas barras como posibles valores tiene la variable: **atributo divisor**
- La textura de cada barra no es uniforme sino que está formada por píxeles que representan un registro.
- La posición (x,y) del píxel dentro de una barra depende del valor de ese registro para dos variables:
 - **Atributo de orden-x**
 - **Atributo de orden-y**
- El color del píxel depende del valor de otra variable.



2.3.2 Basadas en Píxeles (query-dependent)

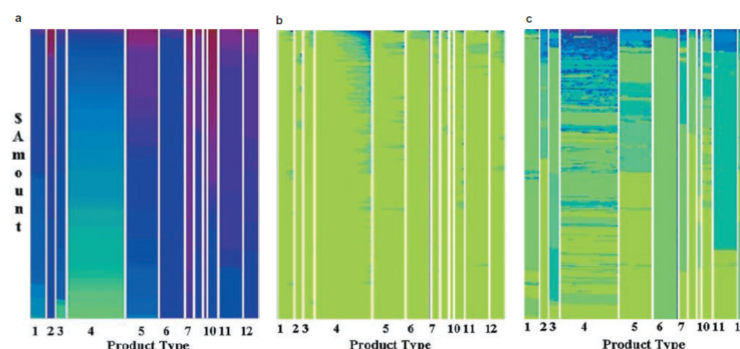
- Ejemplo:
 - Base de datos de miles de ventas on-line
 - Cada registro de comprado se describe con 3 variables:
 - Tipo de producto (12 posibles valores) → *atributo divisor*
 - Número de visitas del comprador → *atributo de orden x*
 - Precio de la venta → *atributo orden y*
 - El color también representa el precio de venta
 - Ojo!: la altura que se indica con el *atributo de orden y* es distinto en cada barra: tiene que ser proporcional a la altura de la barra en el histograma.



Al coincidir el atributo orden-y con el color, se consiguen efectos de degradación.

2.3.2 Basadas en Píxeles (query-dependent)

- El problema del anterior es que solo muestra 3 ó 4 variables:
 - la divisora y las 2 de orden (x,y).
 - más una cuarta si se quiere con el color.
- **Diagrama de Barras Multi-Pixel**
- Se repite el diagrama de barras pixeladas por cada atributo extra que se quiera mostrar:
 - El atributo extra cambia la codificación de colores
 - Los registros están en la misma posición en los diferentes diagramas



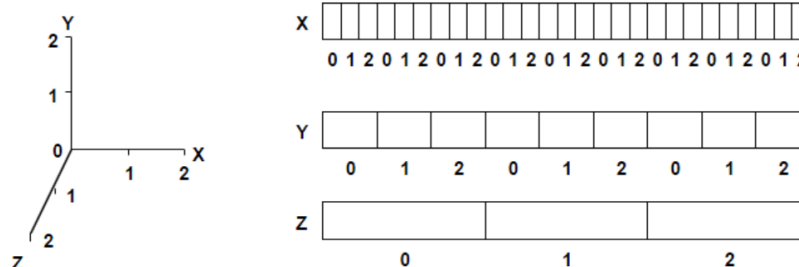
(a) Color=dollar amount. (b) Color=no. of visits. (c) Color=quantity

2.3.3 Jerárquicas

- Dividen el espacio bi-dimensional en subespacios de forma jerárquica.
- Los atributos se tratan de forma diferente: algunos tienen una representación mayor que otros.
- Estas técnicas son útiles cuando hay atributos que nos resultan más interesantes que otros.
- Vamos a ver:
 - Ejes Jerárquicos
 - Dimensiones Apiladas
 - Mundos dentro de Mundos
 - Treemap

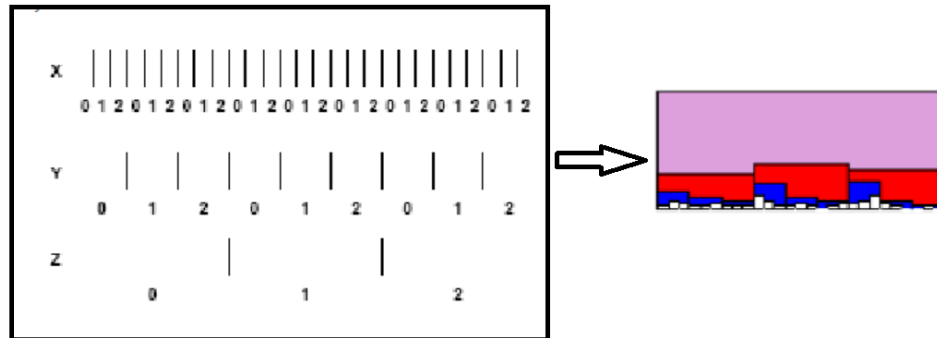
2.3.3 Jerárquicas

- Ejes Jerárquicos [T. Mihalisin, J. Timlin and J. Schwegler, "Visualization and Analysis of Multi-Variate Data: A Technique for All Fields", Proceedings of the 2nd IEEE Conference on Visualization '91, pp.171-178, 1991.]
- Las variables deben ser discretas.
- Los ejes no parten de un mismo centro como en el sistema cartesiano, sino que se dibujan uno debajo del otro.
 - Fir 3 variables x y z. Cada una con valores {0 1 2}



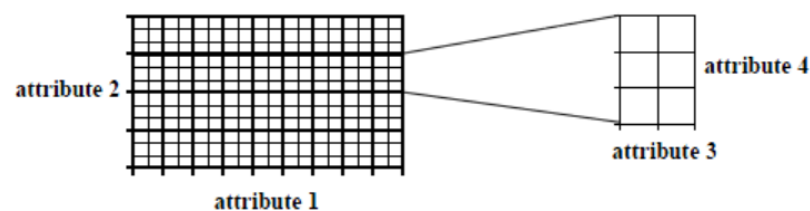
2.3.3 Jerárquicas

- Los rectángulos blancos son el histograma de la frecuencia de aparición en la base de datos de la combinación correspondiente de los valores de las 3 variables.
- Los rectángulos azules son una función (suma, media,...) de los rectángulos blancos para cada valor de y .
- Los rojos función de los azules, para cada valor de z .



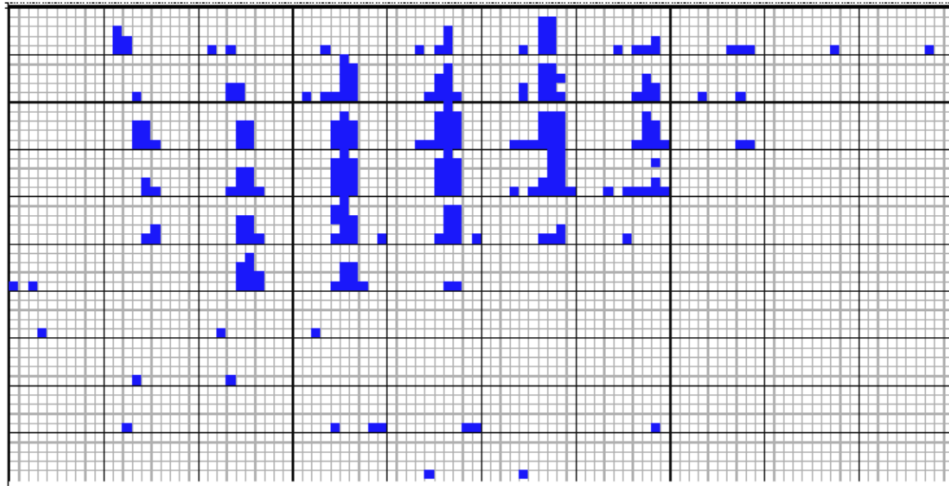
2.3.3 Jerárquicas

- **Apilado Dimensional** [J. LeBlanc, M. O. Ward and N. Wittels, "Exploring N-Dimensional Databases", *Proceedings of the 1st IEEE Conference on Visualization '90*, pp.230-237, 1990.]
- Se reparten las n variables partiendo el espacio bi-dimensional en varios **espacios bi-dimensionales anidados** uno dentro de otro.
- Las **variables más importantes** deberían estar en **niveles más externos**.



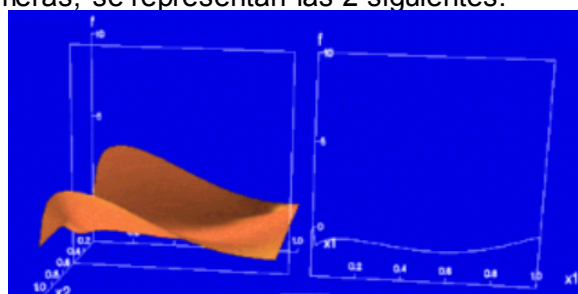
2.3.3 Jerárquicas

- Ejemplo con base de datos sobre pozos petrolíferos, con 4 variables:
 - Longitud y Latitud: ejes x,y externos
 - Pureza y profundidad: ejes x,y internos



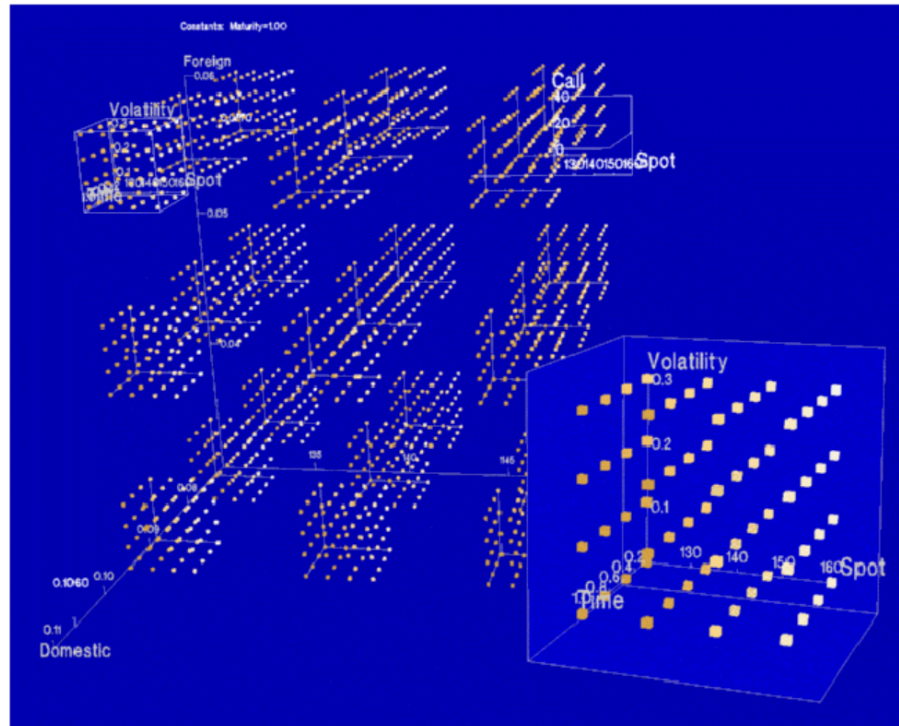
2.3.3 Jerárquicas

- **Mundos dentro de Mundos (N-vision)** [S. Feiner and C. Beshers, "Visualizing n-Dimensional Virtual World with n-Vision", *Computer Graphics*, vol.24, no.2, pp.37-38, 1990]
- Similar al Apilado Dimensional, pero la representación se hace **de 3 en 3 dimensiones: renderizado en tiempo real**.
- Cuando **se fija** el valor para las **3 dimensiones externas**, se renderiza una nueva representación para las siguientes 3 dimensiones interiores.
- La representación visual puede ser:
 - Superficie 3D función de las 3 dimensiones del nivel actual
 - Matriz de dispersión 3D.
- Ej: 5 dimensiones con *representación de Superficie*: al seleccionar un valor para las 3 primeras, se representan las 2 siguientes:



2.3.3 Jerárquicas

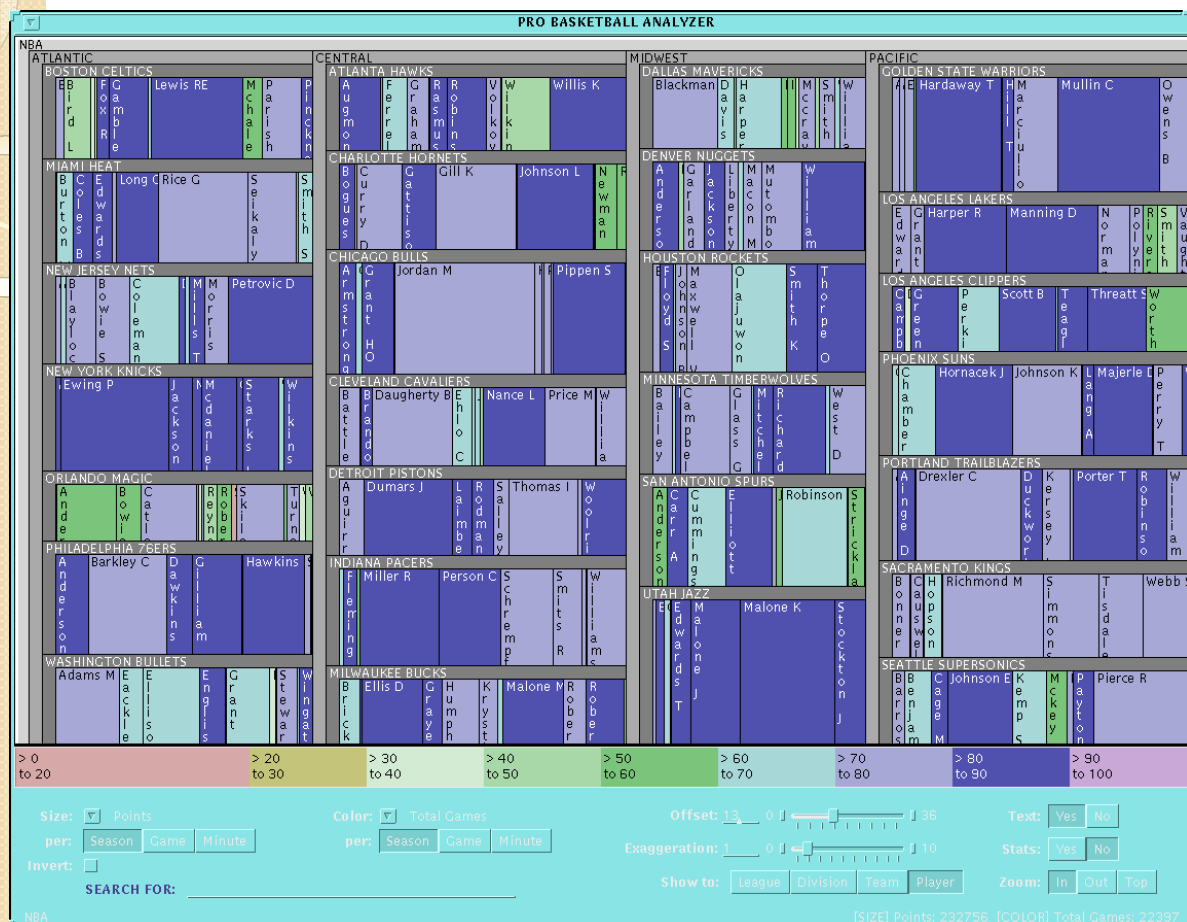
- Ej: 6 dimensiones ((Domestic, Foreign,Z),(Tiime,Volatility,Spot)) con representación de matriz de dispersión



43

2.3.3 Jerárquicas

- Treemap** [B. Shneiderman, "Tree Visualization with Treemaps: A 2D Space-Filling Approach", ACM Transactions on Graphics, vol.11, no.1, pp.92-99, 1992.]
- Toda su historia: <http://www.cs.umd.edu/hcil/treemap-history/>
- Las dimensiones x,y de la pantalla o papel se **van dividiendo alternativamente según los posibles valores de una variable** discreta.
- El **tamaño** de cada región representa el valor de la variable: especialmente adecuado para variables discretas con carácter **ordinal**.
- El **color** de las regiones puede representar un atributo adicional.
- El **orden de las particiones** es indicado por el usuario.
- Ejemplo:
 - Base de datos de 453 jugadores de la NBA en la liga 1992-1993
 - 42 variables: puntos, tiros libres, faltas,...
 - En la página siguiente:
 - Primer nivel: Zona geográfica (4 valores)
 - Segundo nivel: Nombre del equipo (27 valores)
 - Tercer nivel: Puntos del jugador
 - Color: con el color se va eligiendo de forma interactiva qué variable queremos mostrar. En este ejemplo se codifican el número de partidos jugados.



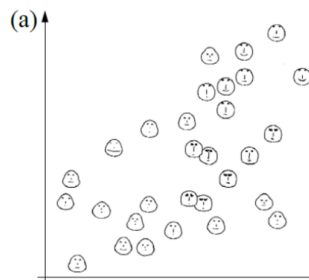
Técnicas de Visualización

2.3.4 Basadas en Iconos

- Cada registro se transforma un icono o pictograma.
- Interpretación intuitiva.
- La percepción varía entre personas, así que hay sesgo en la interpretación de los resultados.
- Vamos a ver:
 - Caras de Chernoff
 - Pictograma de Estrella
 - Figura de Palos
 - Codificación de Forma
 - Icono de colores
 - Textura

2.3.4 Basadas en Iconos

- **Caras de Chernoff** [H. Chernoff, "The Use of Faces to Represent Points in k-Dimensional Space Graphically", *Journal American Statistical Association*, vol. 68, pp.361-368, 1973.]
- Técnica basada en iconos más famosa.
- Se seleccionan **2 atributos** para acordar la posición (**x,y**) de la cara en un sistema cartesiano.
- Las características de la cara representan los valores del **resto de atributos: forma de la nariz, ojos, boca,...**
- No se pueden mostrar muchos registros



(a) Chernoff faces in various 2D positions



(b) Different facial features

Técnicas de Visualización

47

2.3.4 Basadas en Iconos

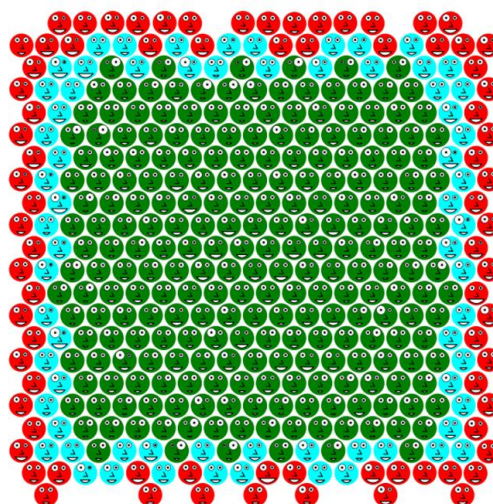
[https://www-](https://www-304.ibm.com/connections/blogs/predictiveanalytics/entry/funny_faces_visualizing_many_variables1?lang=en_us)

304.ibm.com/connections/blogs/predictiveanalytics/entry/funny_faces_visualizing_many_variables1?lang=en_us

Left Eye Size: Acceleration **Right Eye Size:** Horsepower **Mouth Size:** Miles Per Gallon

US cars: green

"Sadly for U.S. drivers, they have some of the smallest smiles as their fuel economy is relatively bad."

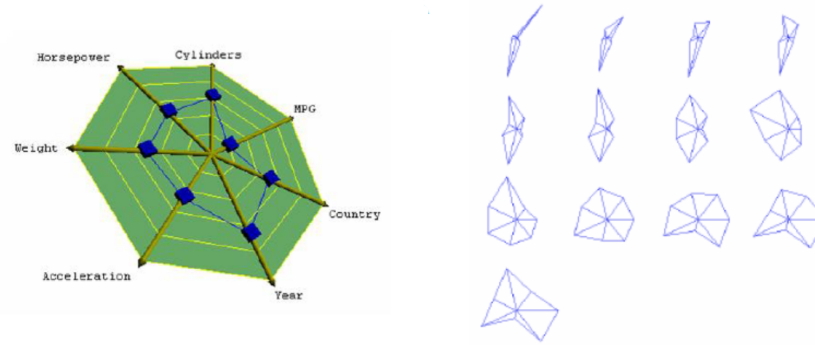


Técnicas de Visualización

48

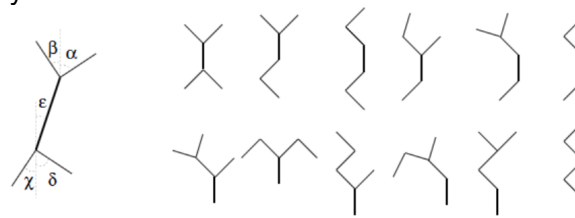
2.3.4 Basadas en Iconos

- **Pictograma de Estrella** [V. Bategelj and A. Mrvar, "Visualization of Multivariate Data Using 3D and VR Presentations", *Indo-French Workshop on Symbolic Data Analysis and its applications*, vol. 1, pp.66-76, 1997, available at: <http://vlado.fmf.uni-lj.si/vrml/paris.97/>]
- Construcción de Diagrama de Coordenadas Paralelas Circulares (ver sección 2.3.1)
- Se extrae cada estrella dibujada por registro y se muestran todas juntas
- No es útil cuando crece el número de registros o variables.



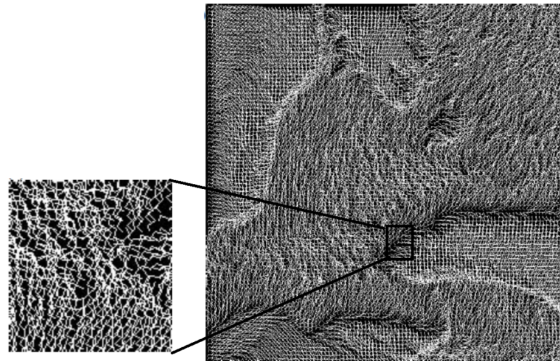
2.3.4 Basadas en Iconos

- **Figura de Palos** [R. M. Pickett and G. G. Grinstein, "Iconographic Displays for Visualizing Multidimensional Data", *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics*, vol. 1, pp.514-519, 1988]
- Se seleccionan **2 atributos** para acordar la posición (**x,y**) del palo en un sistema cartesiano.
- El **resto de atributos** se utilizan para dar **longitud, ángulo, brazos,... a un palo por registro.**
- Es útil para bases de datos con muchos registros y no muchos atributos.
- Se pueden observar **texturas** que muestran **patrones.**
- La selección de las propiedades del palo da lugar a diferentes representaciones, y por lo tanto hay que dedicar mucho tiempo a ello.
- 1 cuerpo y hasta 4 extremidades.



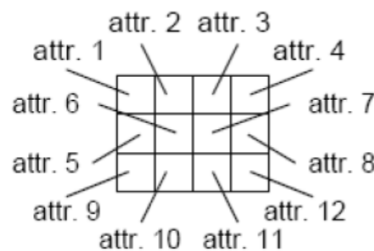
2.3.4 Basadas en Iconos

- Ej: base de datos sobre la región de Grandes Lagos de EEUU.
 - 5 dimensiones
 - (x,y) en el gráfico se mapean directamente a las variables longitud y latitud
 - Las otras 3 variables muestran: extensión, profundidad,...



2.3.4 Basadas en Iconos

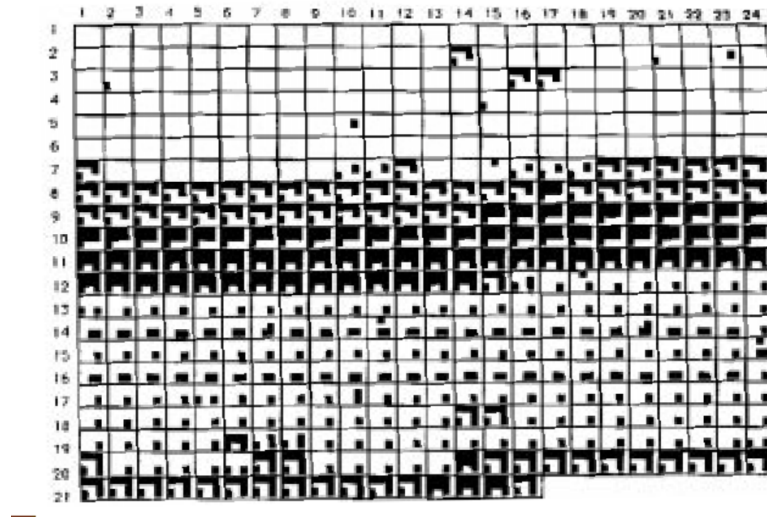
- **Codificación de Forma** [J. Beddow, "Shape Coding of Multidimensional Data on a Microcomputer Display", *Proceedings of the 1st IEEE Conference on Visualization '90*, pp.238-246, 1990.]
- Cada registro se representa con un array dividido en tantas celdas como variables tengamos.
- A cada celda le corresponde un color según el valor que tome en el correspondiente registro.



- Los arrays se ordenan según el valor de un atributo seleccionado (ej: tiempo)

2.3.4 Basadas en Iconos

- Ejemplo:
- observaciones de la Tierra recogidas por la Nasa a lo largo del tiempo.
- Se ordenan los registros por tiempo.

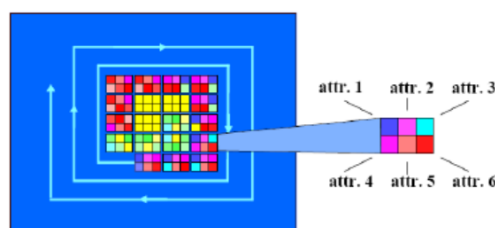


Técnicas de Visualización

53

2.3.4 Basadas en Iconos

- **Icono de colores** [H. Levkowitz, "Color Icons: Merging Color and Texture Perception for Integrated Visualization of Multiple Parameters", *Proceedings of the 2nd IEEE Conference on Visualization '91*, pp.164-170, 1991]
- Los registros se representan con arrays como en la técnica anterior (Codificación de Forma)
- Se distribuyen como en la técnica basada en píxeles de Espiral, cambiando los píxeles por arrays.
- Cada celda se representa con varios píxeles.
- El array debe estar limitado para distinguirse del resto de arrays.

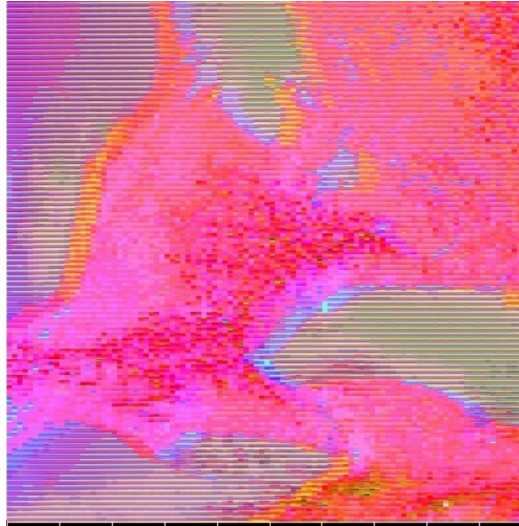


Técnicas de Visualización

54

2.3.4 Basadas en Iconos

- Mismo ejemplo de la Región de Grandes Lagos de EEUU, pero esta vez usando Iconos de Color.



2.3.4 Basadas en Iconos

- **Texture** [V. Interrante, "Harnessing Natural Textures for Multivariate Visualization", *IEEE Transactions on Computer Graphics and Applications*, vol.20, no.6, pp.6-11, 2000.]
- Las técnicas anteriores crean texturas que hay que interpretar
- Actualmente, es posible seleccionar las texturas directamente para que la interpretación sea más directa.
- Propiedades más distintivas de una textura, además del color:
 - Orientación
 - Tamaño
 - Contraste
- Otras:
 - Luz
 - Transparencia
 - Difusa

2.3.4 Basadas en Iconos

- Ejemplo: datos meteorológicos.
 - Se inventa una textura con forma de pincelada
 - La forma de la pincelada representa 5 dimensiones (además de las posiciones x,y mostradas por la posición de la pincelada en el mapa)
 - Temperatura (azul, verde, rosa y rojo de menos a más)
 - Lluvia (recta poca lluvia, y se inclina cuantas más lluvias haya)
 - Presión (color en gradiente)
 - Viento (pinceladas muy juntas es viento fuerte)
 - Heladas (regularidad en la distribución de las pinceladas es que no hay heladas)

2.3.4 Basadas en Iconos

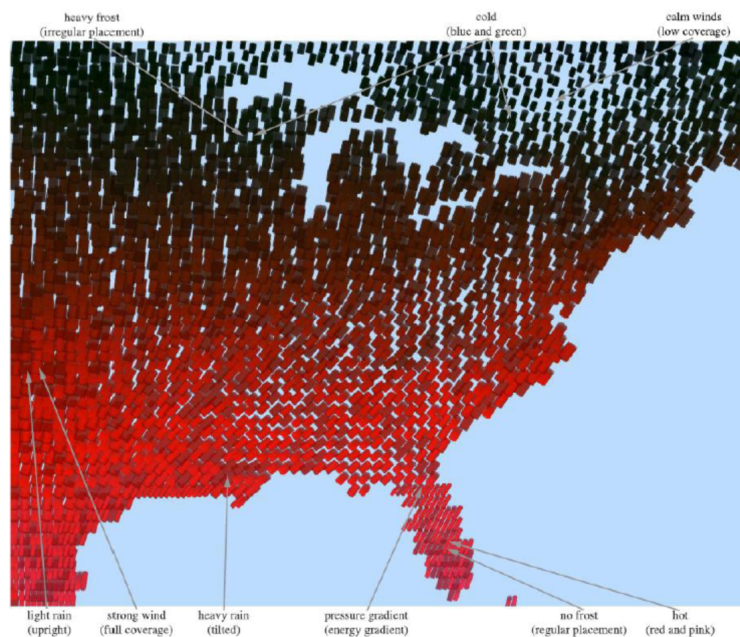
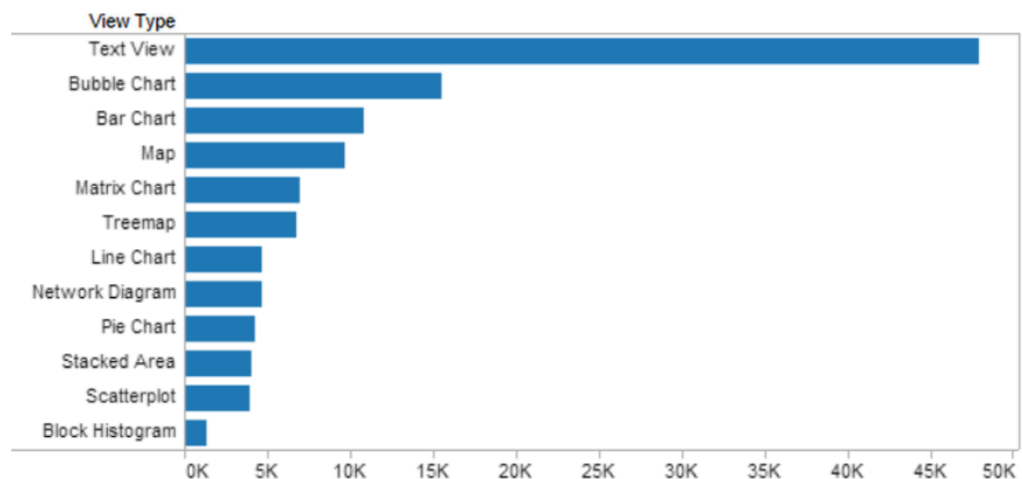


Tabla orientativa

- ¿En qué caso conviene usar cada tipo de técnica?

		cluster- ing	multi- variate hot spot	no. of variates	no. of data items	cate- gorical data	visual overlap	learning curve
Geometric Techniques	Scatterplot Matrices	++	++	+	+	-	o	++
	Landscapes	+	+	-	o	o	+	+
	Prosection Views	++	++	+	+	-	o	+
	Hyperslice	+	+	+	+	-	o	o
	Parallel Coordinates	o	++	++	-	o	--	o
Icon-based Techniques	Stick Figure	o	o	+	-	-	-	o
	Shape Coding	o	-	++	+	-	+	-
	Color Icon	o	-	++	+	-	+	-
Pixel-oriented Techniques	Query-Independent	+	+	++	++	-	++	+
	Query-Dependent	+	+	++	++	-	++	-
Hierarchical Techniques	Dimensional Stacking	+	+	o	o	++	o	o
	Worlds-within-Worlds	o	o	o	+	o	o	o
	Treemap	+	o	+	o	++	+	o
	Cone Trees	+	+	o	+	o	+	+
	InfoCube	o	o	-	-	o	o	+

Daniel A. Keim



Gráficos obtenido de: <http://eagereyes.org/criticism/quo-vadis-many-eyes>

