

Cápsula 3: Separación matricial

Hola, bienvenidxs a una cápsula del curso Visualización de Información. Continuaré con la idea de separación espacial de la cápsula anterior, y hablaré sobre separación matricial.

La separación matricial del espacio bidimensional suele ocuparse en conjuntos de datos donde **hay dos atributos que funcionan como llave en conjunto**. Uno de los atributos se despliega en las filas y otro en las columnas, de forma que las celdas que resultan como combinaciones de valores son las regiones para mostrar o codificar información.

Los mapas de calor o *heatmaps* son unos de los *idioms* más simples que utilizan separación matricial, donde las celdas se rellenan completamente de una marca rectangular cuyo color codifica un atributo, por lo general, cuantitativo.

Una de sus ventajas es que son compactos, las áreas de las celdas pueden ser bastante pequeñas. La escala está limitada por miles de celdas, que corresponde a cientos de valores en los atributos llave. Por otro lado, la cantidad de niveles diferenciables por color es más limitada, dependiendo del tipo de *colormap* utilizado.

Otro ejemplo de uso de separación matricial son las **matrices de gráficos de dispersión**. Estas son matrices donde en cada celda hay un gráfico de dispersión. La matriz se forma por combinaciones de atributos del *dataset*, y a partir de las combinaciones de atributos se genera un gráfico de dispersión en cada celda.

A diferencia de los mapas de calor, los valores en filas y columnas **no son valores de atributos, si no que son listas de nombres de atributos**. Esto podría considerarse un dato derivado del *dataset*, donde es de interés ver emparejamiento de múltiples atributos, y no solo dos.

Suelen omitirse las celdas que corresponden a combinaciones de atributos idénticos, ya que los gráficos de dispersión resultantes son diagonales que no entregan mucha información. También muchas veces se omiten las repeticiones de combinaciones, por razones similares.

Ya revisamos separación de regiones para una y dos llaves. Podemos extender ese patrón a tres llaves y generar grillas 3D para codificar combinaciones. Pero como podrán intuir, por el uso de 3D se introducen problemas de distorsión de perspectiva y oclusión, **por lo que es mejor optar por una traducción en dos dimensiones**.

También es posible realizar subdivisiones recursivas cuando se trata con múltiples atributos llave. Es decir, manteniendo separación y alineamiento en listas o matrices, pero extendiendo las regiones con la idea de contención y agregando más separaciones y alineamientos por región.

Con eso termina el contenido de esta cápsula. Recuerda que si tienes preguntas, puedes dejarlas en los comentarios del video para responderlas en la sesión en vivo de esta temática. ¡Chao!