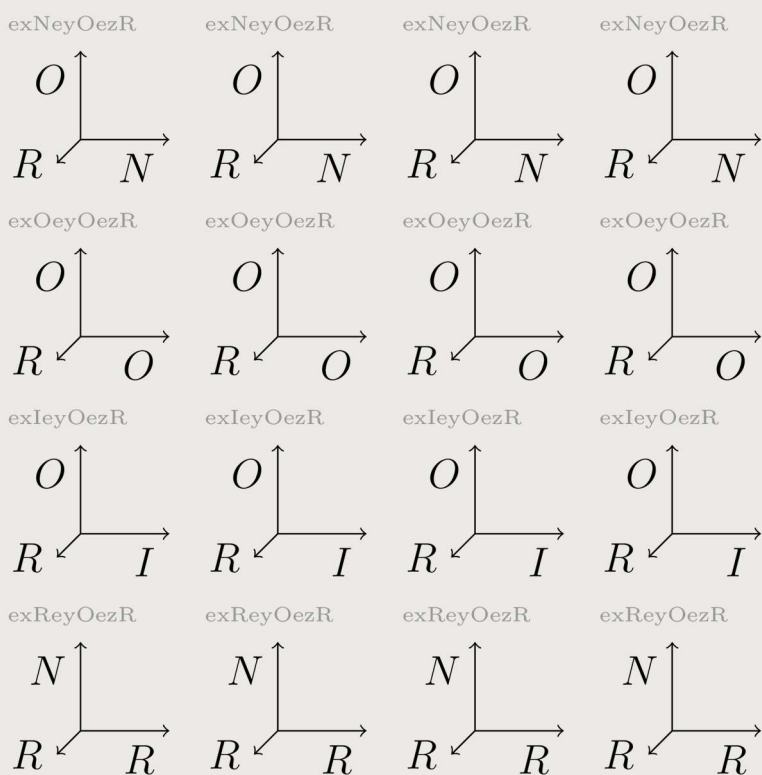


Un sistema de recomendación de gráficas estadísticas basado en las características de los datos

Pere Millán Martínez



Un sistema de recomendación de gráficas estadísticas basado en las características de los datos

Pere Millán Martínez

Director de tesi: Ramon Oller Piqué

Programa de Doctorat en Cures Integrals i Serveis de Salut

2022

... penso en les nostres bessones Ainhoa i Sofía ...

AGRADECIMIENTOS

Son varias las personas, sin cuyo esfuerzo, esta tesis no se estaría proyectando en esta pantalla. Los nombres de todas ellas no están necesariamente representados a continuación.

En lo académico, agradezco a Ramon Oller Piqué haber dirigido y escrutado cada oración de esta tesis con un rigor desacostumbrado. A Pedro Valero Mora por haberme acompañado en mi primera experiencia como doctorando y, especialmente, por dedicar su tiempo en orientar mis primeros pasos en la escritura de artículos científicos. A Michael Friendly por facilitarme desarrollar parte del trabajo en su laboratorio *datavis*, por su estima y por su empeño en que acabara defendiendo esta tesis. A Joan Carles Cases Baroy y Joan Carles Martori Cañas sus comentarios en las comisiones de seguimiento del proyecto.

En lo personal, agradezco especialmente a mis padres José Millán García y Caridad Martínez Millán haberme concebido y luego costeado todos los estudios, siempre confiados de que sacaría provecho de ellos. A mi esposa, Paola Galbany Estragués, por escuchar atentamente mis divagaciones aún sin argumentos que refutar, dado que su ámbito de especialización es otro. A Gloria Gallego Caminero y Clara Juando Prats por facilitar la estancia de toda la familia en Toronto. A mi cuñado Roger Papasseit Borrell y a mis hermanos Mateo, Marco Antonio, Juan José y Silvia por interesarse de vez en cuando por el estado de desarrollo de la tesis. A mis cuñados Antonio Galbany Estragués y André Moreira Santos por dormir junto a un tractor en el Canigó. A Virtudes Serrano Claramonte por alojarme los días que tenía que asistir a cursos diversos. A mi suegra Ana María Estragués Cornejo por identificar y resaltar la importancia de cada pequeño

logro y a mi suegro Antonio Galbany Viñamata por observar, siempre antes que nadie, que el F.C. Barcelona no está jugando bien.

ABSTRACT

We are immersed in a data explosion that makes it necessary to expand and improve the methods that allow us to extract information from them. One of the first processes to convert data into information is known as EDA (or exploratory data analysis) which consists of observing the characteristics of a data set, without emphasizing data modeling or testing preconceived hypotheses. If this exploration uses graphs that represent the data, then it is known as GEDA (or graphical exploratory data analysis).

Observing the data through graphs, without preconceived hypotheses, and that these graphs make us discover aspects of the data that lead to the emergence of new hypotheses, leads us to what is known as the graph problem: among the range of possible graphs... which one we choose? This is where statistical graph recommenders and autoGEDA systems (or automated graphical exploratory data analysis systems) come into service.

The recommendation of statistical graphs can be done following different strategies. On the one hand, based on the characteristics of the data, such as the number of variables to be related, the characteristics of the variables separately, the characteristics of the relationships between them, the way in which the data is structured and its origin or utility for which they have been collected. On the other hand, we have the characteristics of the receiving users, that is, the characteristics of human perception, the task to be done by the user, the memory of previous selections and social conventions. Graphics may also be recommended based on the characteristics of the communication channel, for example, due to limitations in data transmission, processing, or the size of the screen where the graphic

is projected. Finally, graphs can also be recommended based on the more or less specific characteristics of the type of graph desired.

Among the strategies that can be followed to recommend statistical graphs, the number of variables to be related and the characteristics of the variables separately are especially relevant. Among the characteristics that can be described for each of the variables and that have an impact on the selection of one or another statistical graph, we find aspects such as, for example, the scale of measurement of the variables, their consideration as predictors or response, the number of observations, or the count of unique values observed. Given a limited selection of variables in a data set, the more detailed the characterization of these variables is, the fewer the number of statistical graphs that may be of interest to the user.

Based on this premise, this work proposes a multidimensional characterization of the variables separately, which is useful for choosing which graphs to show to a user based on the characteristics of the variables selected by the user. The proposed characterization considers the graphic measurement scale, the data aggregation method, the cyclicity of the sample space, the convenience of explicitly showing the scale of the variable and its length. From the proposed characterization of the variables separately and the statistical graphs to which each combination of variables can be associated, a framework is established with which the statistical graphs can be classified.

The proposed characterization of the variables, despite the possible improvements to which it may be subjected, may be the seed of a graph grammar that, instead of being based on representation models, would be based on the properties of the variables. This would translate, for example, in defining a variable as ambiguous to eliminate a certain coordinate axis or a certain legend, defining a variable as cyclic to convert an orthogonal coordinate axis into a circular one, or defining a variable as gridded type to convert, for example, an uniaxial dot plot into a histogram or a scatterplot into

a heatmap.

However, we must not lose sight of the fact that the data sets are generally stored in computer systems that have already characterized the variables with other criteria that, instead of pursuing the best visualization, seeks to minimize storage space. Given this preset characterization, expecting a user to characterize the variables again before getting a plot is possibly a naive approach. Having to re-characterize the variables raises a barrier between the data and the user, especially if we take into account that users are not necessarily familiar with the data.

Overcoming the barrier of having to characterize the data has three possible solutions. The first solution is to take advantage of the pre-established characterization of the data to, based on it, suggest the statistical graphs. The second solution involves making assumptions in relation to the data, so that the characterization of the variables is transparent to the user and, if it is wrong, he can modify it. The third solution is to store the data prioritizing possible graphic exploitation instead of prioritizing the necessary space on a hard drive or any other support.

When proposing a recommendation system for statistical graphs, based on the characteristics of the data, among the possible solutions to prevent the user from having to re-characterize the variables, we have chosen the first one. In our case, we have taken advantage of the pre-established characterization in the specific scope of the **R** statistical programming environment. The result of this implementation is the **brinton** package for **R** that includes the functions `wideplot()`, `longplot()`, `matrixplot()` and `plotup()` which automatically present statistical plots, assist the user in exploring data sets using univariate and bivariate plots, while also facilitating the choice, edition and representation of a given graph by the user.

Each function of the **brinton** package adds a new alternative within the scope of automated graphical exploration of data and the

set of functions, facilitates and speeds up the process of generating information from a data set. In the near future, the usefulness of the `brinton` package will be enhanced by the addition of new species to the univariate and bivariate graph specimens as well as the addition of a new trivariate graph specimen and new functions to complement the existing ones.

Given the range of graphs that the `brinton` package provides and the ease with which users can choose between one graph or another, a future line of research is to discover the relationship between the chosen graphs and the utility they represent for the users. This relationship would allow adding precision to the recommendation of statistical graphs, since the range of graphs to be displayed could be reduced to those that are compatible with the selected data and that have the best expectation to satisfy the utility that the user expects.

RESUM

Estem immersos en una explosió de dades que fa necessari ampliar i millorar els mètodes que permeten extreure'n informació. Un dels primers processos per convertir les dades en informació es coneix com a EDA (o anàlisi exploratòria de dades) que consisteix a observar les característiques d'un conjunt de dades, sense posar l'accent en el modelatge de les dades o el contrast d'hipòtesis preconcebudes. Si aquesta exploració se serveix de gràfiques que representen les dades, aleshores es coneix com a GEDA (o anàlisi gràfica exploratòria de dades).

Observar les dades mitjançant gràfiques, sense hipòtesis preconcebudes, i que aquestes gràfiques ens facin descobrir aspectes de les dades que facin emergir noves hipòtesis, ens condueix a allò que es coneix com el problema gràfic: d'entre el ventall de gràfiques possibles... quina triar? Aquí entren en servei els recomanadors de gràfiques estadístiques i els sistemes autoGEDA (o sistemes automatitzats d'anàlisi gràfica exploratòria de dades).

La recomanació de gràfiques estadístiques es pot fer seguint diferents estratègies. D'una banda, a partir de les característiques de les dades, com ara el nombre de variables a relacionar, les característiques de les variables per separat, les característiques de les relacions entre aquestes, la manera com s'estructuren les dades i la seva procedència o utilitat per a la qual s'han recollit. D'altra banda, tenim les característiques dels usuaris receptors, és a dir, les característiques de la percepció humana, la tasca a realitzar per l'usuari, el record de seleccions prèvies i les convencions socials. També es poden recomanar gràfiques en funció de les característiques del canal de comunicació, per exemple, a causa de limitacions en la transmissió

de dades, de processament o de la mida de la pantalla on es projecta la gràfica. Finalment, també es poden recomanar gràfiques a partir de les característiques, més o menys concretes, del tipus de gràfica desitjada.

Entre les estratègies que es poden seguir per recomanar gràfiques estadístiques, tenen especial rellevància el nombre de variables a relacionar i les característiques de les variables per separat. Entre les característiques que es poden descriure de cadascuna de les variables i que tenen incidència en la selecció d'una gràfica estadística o una altra, trobem aspectes com, per exemple, l'escala de mesura de les variables, la consideració d'aquestes com a predictores o de resposta, el nombre d'observacions o el recompte de valors diferents observats. Donada una selecció limitada de variables d'un conjunt de dades, com més detallada és la caracterització d'aquestes variables, menor és el nombre de gràfiques estadístiques que poden ser interessants per a l'usuari.

A partir d'aquesta premissa, aquest treball proposa una caracterització multidimensional de les variables per separat que és útil per escollir quines gràfiques mostrar a un usuari a partir de les característiques de les variables seleccionades per aquest. La caracterització proposada considera l'escala de mesura gràfica, el mètode d'agregació de les dades, la ciclicitat de l'espai mostral, la conveniència de mostrar explícitament l'escala de la variable i la longitud d'aquesta. A partir de la caracterització proposada de les variables per separat i de les gràfiques estadístiques a què cada combinació de variables es pot associar, s'estableix un marc amb què es poden classificar les gràfiques estadístiques.

La caracterització de les variables proposada, malgrat les possibles millores a què es pugui sotmetre, pot ser la llavor d'una gramàtica de les gràfiques que, en comptes d'estar basada en models de representació, estaria basada en les propietats de les variables. Això es traduiria, per exemple, en definir una variable com a ambigua

per eliminar un determinat eix de coordenades o una determinada llegenda, definir una variable com a cíclica per convertir un eix de coordenades ortogonal en un circular, o definir una variable com de tipus tamisat per convertir, per exemple, un diagrama uniaxial de punt en un histograma o un diagrama de dispersió en un mapa de calor.

No cal perdre de vista, però, que els conjunts de dades es troben emmagatzemats, generalment, en sistemes informàtics que ja tenen caracteritzades les variables amb un altre criteri que, en comptes de perseguir la millor visualització, persegueix minimitzar l'espai d'emmagatzematge. Donada aquesta caracterització preestablerta, esperar que un usuari torni a caracteritzar les variables novament abans d'obtenir una gràfica és, possiblement, un plantejament naïf. Haver de tornar a caracteritzar les variables aixeca una barrera entre les dades i l'usuari, més si tenim en compte que els usuaris no estan necessàriament familiaritzats amb les dades.

Superar la barrera que suposa haver de caracteritzar les dades té tres possibles solucions. La primera solució passa per aprofitar la caracterització preestablerta de les dades per, en base a aquesta, suggerir les gràfiques estadístiques. La segona solució passa per fer suposicions en relació a les dades, de manera que la caracterització de les variables sigui transparent per a l'usuari i que, en cas de ser errònia, aquest pugui modificar-la. La tercera solució passa per emmagatzemar les dades primant la possible explotació gràfica en comptes de primar l'espai necessari en un disc dur o qualsevol altre suport.

A l'hora de proposar un sistema de recomanació de gràfiques estadístiques, en base a les característiques de les dades, entre les possibles solucions per evitar que l'usuari hagi de tornar a caracteritzar les variables, hem escollit la primera. En el nostre cas, hem aprofitat la caracterització preestablerta en l'àmbit específic de l'entorn de programació estadística R. El fruit d'aquesta implementació és el paquet

`brinton` per a **R** que inclou les funcions `wideplot()`, `longplot()`, `matrixplot()` i `plotup()` que presenten de manera automàtica gràfiques estadístiques, assisteixen a l'usuari en l'exploració dels conjunts de dades mitjançant gràfiques univariades i bivariades, alhora que faciliten l'elecció, edició i representació d'una gràfica determinada per part de l'usuari.

Cada funció del paquet `brinton` afegeix una alternativa nova dins l'àmbit de l'exploració gràfica automatitzada de dades i el conjunt de les funcions, facilita i accelera el procés de generació d'informació a partir d'un conjunt de dades. En un futur proper, la utilitat del paquet `brinton` serà reforçada mitjançant la incorporació de noves espècies als espècimens de gràfiques univariades i bivariades així com la incorporació d'un nou espècimen de gràfiques trivariades i noves funcions que complementin les existents.

Donat el ventall de gràfiques que el paquet `brinton` proporciona i la facilitat amb què els usuaris poden triar entre una gràfica o una altra, una futura línia de recerca és conèixer la relació entre les gràfiques escollides i la utilitat que aquestes representen per als usuaris. Aquesta relació permetria afegir precisió a la recomanació de gràfiques estadístiques atès que, el ventall de gràfiques a mostrar, es podria reduir a aquelles que són compatibles amb les dades seleccionades i que millor expectativa tenen de satisfer la utilitat que l'usuari n'espera.

RESUMEN

Estamos inmersos en una explosión de datos que hace necesario ampliar y mejorar los métodos que permiten extraer información de ellos. Uno de los primeros procesos para convertir los datos en información se conoce como EDA (o análisis exploratorio de datos) que consiste en observar las características de un conjunto de datos, sin hacer hincapié en el modelado de los datos o el contraste de hipótesis preconcebidas. Si esta exploración se sirve de gráficas que representan los datos, entonces se conoce como GEDA (o análisis gráfico exploratorio de datos).

Observar los datos mediante gráficas, sin hipótesis preconcebidas, y que estas gráficas nos hagan descubrir aspectos de los datos que hagan emerger nuevas hipótesis, nos conduce a lo que se conoce como el problema gráfico: de entre el abanico de gráficas posibles... cuál elegir? Aquí entran en servicio los recomendadores de gráficas estadísticas y los sistemas autoGEDA (o sistemas automatizados de análisis gráfico exploratorio de datos).

La recomendación de gráficas estadísticas se puede realizar siguiendo diferentes estrategias. Por un lado, a partir de las características de los datos, como el número de variables a relacionar, las características de las variables por separado, las características de las relaciones entre éstas, la forma en que se estructuran los datos y su procedencia o utilidad para la que se han recogido. Por otro lado, tenemos las características de los usuarios receptores, es decir, las características de la percepción humana, la labor a realizar por el usuario, el recuerdo de selecciones previas y las convenciones sociales. También se pueden recomendar gráficas en función de las características del canal de comunicación, por ejemplo, debido a limitaciones en la transmisión

de datos, procesamiento o tamaño de la pantalla donde se proyecta la gráfica. Por último, también se pueden recomendar gráficas a partir de las características, más o menos concretas, del tipo de gráfica deseada.

Entre las estrategias que pueden seguirse para recomendar gráficas estadísticas, tienen especial relevancia el número de variables a relacionar y las características de las variables por separado. Entre las características que se pueden describir de cada una de las variables y que tienen incidencia en la selección de una u otra gráfica estadística, encontramos aspectos como, por ejemplo, la escala de medida de las variables, la consideración de éstas como a predictoras o de respuesta, el número de observaciones o el recuento de valores distintos observados. Dada una selección limitada de variables de un conjunto de datos, cuanto más detallada es la caracterización de estas variables, menor es el número de gráficas estadísticas que pueden resultar interesantes para el usuario.

A partir de esta premisa, este trabajo propone una caracterización multidimensional de las variables por separado que es útil para elegir qué gráficas mostrar a un usuario a partir de las características de las variables seleccionadas por éste. La caracterización propuesta considera la escala de medida gráfica, el método de agregación de los datos, la ciclicidad del espacio muestral, la conveniencia de mostrar explícitamente la escala de la variable y su longitud. A partir de la caracterización propuesta de las variables por separado y de las gráficas estadísticas a las que cada combinación de variables se puede asociar, se establece un marco con el que se pueden clasificar las gráficas estadísticas.

La caracterización de las variables propuesta, a pesar de las posibles mejoras a las que pueda someterse, puede ser la semilla de una gramática de las gráficas que, en vez de estar basada en modelos de representación, estaría basada en las propiedades de las variables. Esto se traduciría, por ejemplo, en definir una variable como ambigua

para eliminar un determinado eje de coordenadas o una determinada leyenda, definir una variable como cíclica para convertir un eje de coordenadas ortogonal en un circular, o definir una variable como de tipo tamizado para convertir, por ejemplo, un diagrama uniaxial de punto en un histograma o un diagrama de dispersión en un mapa de calor.

Sin embargo, no hay que perder de vista que los conjuntos de datos se encuentran almacenados, generalmente, en sistemas informáticos que ya tienen caracterizadas las variables con otro criterio que, en vez de perseguir la mejor visualización, persigue minimizar el espacio de almacenamiento. Dada esta caracterización preestablecida, esperar que un usuario vuelva a caracterizar las variables nuevamente antes de obtener una gráfica es, posiblemente, un planteamiento naíf. Tener que volver a caracterizar las variables levanta una barrera entre los datos y el usuario, más si tenemos en cuenta que los usuarios no están necesariamente familiarizados con los datos.

Superar la barrera que supone tener que caracterizar los datos tiene tres posibles soluciones. La primera solución pasa por aprovechar la caracterización preestablecida de los datos para, en base a ésta, sugerir las gráficas estadísticas. La segunda solución pasa por realizar suposiciones en relación a los datos, de modo que la caracterización de las variables sea transparente para el usuario y que, en caso de ser errónea, éste pueda modificarla. La tercera solución pasa por almacenar los datos primando la posible explotación gráfica en vez de primar el espacio necesario en un disco duro o cualquier otro soporte.

En el momento de proponer un sistema de recomendación de gráficas estadísticas, en base a las características de los datos, entre las posibles soluciones para evitar que el usuario deba volver a caracterizar las variables, hemos escogido la primera. En nuestro caso, hemos aprovechado la caracterización preestablecida en el ámbito específico del entorno de programación estadística R. El fruto de esta implementación es el paquete `brinton` para **R** que incluye

las funciones `wideplot()`, `longplot()`, `matrixplot()` y `plotup()` que presentan de forma automática gráficas estadísticas, asisten al usuario en la exploración de los conjuntos de datos mediante gráficas univariadas y bivariadas, a la vez que facilitan la elección, edición y representación de una gráfica determinada por parte del usuario.

Cada función del paquete **brinton** añade una nueva alternativa dentro del ámbito de la exploración gráfica automatizada de datos y el conjunto de las funciones, facilita y acelera el proceso de generación de información a partir de un conjunto de datos . En un futuro próximo, la utilidad del paquete **brinton** será reforzada mediante la incorporación de nuevas especies a los especímenes de gráficas univariadas y bivariadas así como la incorporación de un nuevo espécimen de gráficas trivariadas y nuevas funciones que complementen las existentes.

Dado el abanico de gráficas que el paquete **brinton** proporciona y la facilidad con la que los usuarios pueden elegir entre una gráfica u otra, una futura línea de investigación es conocer la relación entre las gráficas escogidas y la utilidad que éstas representan para los usuarios. Esta relación permitiría añadir precisión a la recomendación de gráficas estadísticas dado que, el abanico de gráficas a mostrar, podría reducirse a aquellas que son compatibles con los datos seleccionados y que mejor expectativa tienen que satisfacer la utilidad que el usuario espera.

ABREVIATURAS

- EDA: Análisis exploratorio de datos
- GEDA: Análisis gráfico exploratorio de datos
- autoEDA: Análisis exploratorio de datos automatizado
- autoGEDA: Análisis gráfico exploratorio de datos automatizado
- VisRec: Sistema de recomendación de gráficas
- VV: Variables visuales
- DataVis: Visualización de datos
- InfoVis: Visualización de información

ÍNDICE GENERAL

Abstract	VII
Resum	XI
Resumen	XV
Abreviaturas	XIX
Índice general	XXI
Índice de figuras	XXIII
Índice de cuadros	XXIX
1 Introducción	1
1.1. Motivación	4
1.2. Objetivos de investigación	4
1.3. Contribución	5
1.4. Organización de la disertación	6
2 Generalidades	9
2.1. Variantes de la DataVis	9
2.2. El propósito de la gráfica	16
2.3. Elementos de la gráfica estadística	22
3 Automatización de gráficas estadísticas	43
3.1. Los datos	45
3.2. Los usuarios receptores	73
3.3. El canal	83

3.4.	La gráfica	83
3.5.	Conclusiones	88
4	Caracterización de los datos	91
4.1.	Escala gráfica de medida (M)	93
4.2.	Método de agregación de los datos (A)	99
4.3.	Ciclicidad (C)	105
4.4.	Explicitud (E)	106
4.5.	Longitud de las variables (L)	108
4.6.	Clasificación de las gráficas basadas en la caracterización	109
4.7.	Ejemplos basados en un conjunto de datos	116
4.8.	Discusión	128
4.9.	Limitaciones	131
4.10.	Conclusiones y trabajos futuros	131
5	brinton para GEDA univariado	133
5.1.	Antecedentes	134
5.2.	El panorama de autoGEDA en R	137
5.3.	Gráficas multipanel en R	142
5.4.	El paquete brinton	144
5.5.	Conclusiones	170
6	brinton para GEDA bivariado	173
6.1.	Preámbulo	173
6.2.	Análisis bivariado en los paquetes de autoEDA de R .	175
6.3.	El espécimen de gráficas bivariadas	177
6.4.	Gráficas bivariadas en las funciones	183
6.5.	La nueva función matrixplot	186
6.6.	Ejemplos de exploración	198
7	Conclusiones y futura línea de investigación	225
	Bibliografía	233

ÍNDICE DE FIGURAS

2.1. Diagram de Venn de términos de InfoVis	10
2.2. Visualización científica contemporánea	11
2.3. Visualización científica clásica	12
2.4. Arte de datos	13
2.5. Arte generativo	14
2.6. Diagrama de barras físico	14
2.7. Analítica visual con ViSta	15
2.8. Infografía	16
2.9. Gráfica de registro de información	17
2.10. Gráfica de tratamiento de información	17
2.11. Gráfica de comunicación divulgativa	18
2.12. Gráfica de comunicación científica	19
2.13. Rueda de tensiones	20
2.14. Gráfica de control de procesos	21
2.15. Nomograma	22
2.16. Despiece de una gráfica unipanel	25
2.17. Estructuras sintácticas en las gráficas	26
2.18. Mapa con una componente geográfica	27
2.19. Ábaco de roseta	29
2.20. Las variables visuales de Bertin	30
2.21. Propiedades perceptivas básicas	32
2.22. Propiedades perceptivas elementales	33
2.23. Propiedades perceptivas elementales ordenadas	34
2.24. Ranking de las variables visuales	35
2.25. Despiece de una gráfica multipanel	36
2.26. Panel de control	37
2.27. SpreadPlot implementado en ViSta	38

2.28. Gráfica multipanel condicionada	39
2.29. Diagrama de pares	40
2.30. Diagrama generalizado de pares	41
3.1. Mapa conceptual de las estrategias	45
3.2. Gráfica producida por el sistema APT	49
3.3. Diagrama de Gantt	54
3.4. Sistema Dominó para el análisis gráfico	55
3.5. Tabla semigráfica producida por el sistema CHART	56
3.6. Árbol de decisión del sistema BHARAT	60
3.7. Reglas heurísticas utilizadas por Show Me	65
3.8. Gráfica apta para correspondencia unívoca	66
3.9. Gráfica apta para correspondencia no unívoca	67
3.10. Diagrama inapropiado para variables sin cobertura relacional	68
3.11. Diagrama apropiado producido por el sistema SAGE	68
3.12. Diagrama apto para diferentes relaciones de cardinalidad	69
3.13. Diagrama de intervalos	70
3.14. Diagrama de barras que muestra dependencia algebraica	70
3.15. Matriz de adyacencia y grafo que la representa	71
3.16. Anatomía macroscópica del ojo	73
3.17. Longitudes de onda del espectro visible	74
3.18. Ilusión óptica	76
3.19. Interficie de usuario del sistema BHARAT	78
3.20. Operadores perceptuales utilizados por el sistema BOZ	80
3.21. Diagrama de bloques producido por el sistema BOZ	81
3.22. Explorador SageBook	82
3.23. Editor SageBrush	86
4.1. Escala referenciada a dos extremos	97
4.2. Escala referenciada a un extremo	98
4.3. Escala arbitrariamente referenciada	98
4.4. Escala ordenada	99
4.5. Escala ordenable	99

4.6. Valores dispersos	101
4.7. Tamiz formado por dos variables tamizadas	102
4.8. Recuento de valores únicos de tres variables tamizadas. Fuente: Elaboración propia.	103
4.9. Representación de tres variables tamizadas	103
4.10. Recodificación de valores dispersos a tamizados	104
4.11. Recodificación de valores secuenciales a tamizados	105
4.12. Recodificación de valores secuenciales a dispersos	105
4.13. Recodificación de valores cíclicos en acíclicos	106
4.14. Recodificación de valores acíclicos en cíclicos	106
4.15. Variables de retina ambigua	108
4.16. <i>Treemap</i> con diferentes valores de luminosidad	110
4.17. Gráfica de barras apiladas	110
4.18. Matriz de escalas gráficas y métodos de agregación	112
4.19. Gráficas que codifican los valores de una variable	112
4.20. Gráficas que codifican los valores de dos variables	113
4.21. Gráficas que codifican los valores de tres variables	114
4.22. Gráficas que codifican los valores de una variable cíclica	114
4.23. Gráficas que codifican los valores de dos variables, una de ellas cíclica	114
4.24. Gráficas que codifican los valores de tres variables, una de ellas cíclica	115
4.25. Gráficas que codifican los valores de tres variables, una de ellas ambigua	115
4.26. Gráficas que incluyen variables de una longitud específica	116
4.27. Matriz de valores ordenados por filas y columnas	118
4.28. Gráficas de puntos con líneas descendientes	118
4.29. Gráficas de áreas superpuestas y de puntos con líneas descendientes	119
4.30. Gráficas de punto, de barras y lista ordenada	119
4.31. Tabla semigráfica	120
4.32. Gráficas de áreas superpuestas	121

4.33. Gráfica de puntos con líneas descendentes	121
4.34. Gráfica de áreas superpuestas	122
4.35. Matriz de gráficas de área superpuestas	122
4.36. Gráficas uniaxiales de puntos y de tiras	123
4.37. Tabla semigráfica	124
4.38. Gráfica de puntos con líneas descendentes	125
4.39. Gráficas de puntos y de tiras	125
4.40. Diagrama de dispersión	126
4.41. Diagramas de violín, de caja e histograma	126
4.42. Diagramas multipanelados de violín, de caja e histograma	127
4.43. Diagrama de espagueti	127
5.1. Gráfica estructural <i>Table Lens</i>	138
5.2. Gráfica estructural <i>tableplot</i>	138
5.3. Gráfica estructural que muestra los tipos de vectores . . .	139
5.4. Gráfica estructural <i>spine plot</i>	139
5.5. Diagramas de densidad superpuestos	140
5.6. Gráfica estructural	141
5.7. Embudo de correlaciones	141
5.8. Cuadro de mando simple	143
5.9. Gráficas multipanel condicionada	143
5.10. Gráfica generalizada de pares	144
5.11. Gráfica wideplot	147
5.12. Gráfica longplot	150
5.13. Gráfica de línea	152
5.14. 1er grado gráfico de libertad	153
5.15. 2o grado gráfico de libertad	154
5.16. 3er grado gráfico de libertad	155
5.17. 4o grado gráfico de libertad	155
5.18. 5o grado gráfico de libertad	156
5.19. 6o grado gráfico de libertad	156
5.20. 7o grado gráfico de libertad	157

5.21. 8o grado gráfico de libertad	157
5.22. 9o grado gráfico de libertad	158
5.23. Gráfica wideplot con etiquetas	160
5.24. Gráfica wideplot con tipos específicos	161
5.25. Gráfica wideplot con etiquetas	162
5.26. Gráfica wideplot	163
5.27. Gráfica wideplot con etiquetas	164
5.28. Gráfica wideplot para identificar variables clave	164
5.29. Gráfica de barras	166
5.30. Histograma y gráfica de barras	167
5.31. Gráfica de puntos mejorable	168
5.32. Gráfica de puntos mejorada	169
5.33. Gráfica multipanel	170
6.1. Gráficas del espécimen para dos variables numéricas . . .	178
6.2. Gráficas del espécimen para una variable de tipo factor y otra factor ordenado	179
6.3. Gráficas del espécimen para dos variables de tipo factor .	179
6.4. Gráficas del espécimen para una variable de tipo carácter	181
6.5. Gráficas del espécimen para dos variables numéricas . . .	181
6.6. Gráfica del espécimen para dos variables de tipo factor ordenado	182
6.7. Gráficas del espécimen para dos variables numéricas . . .	182
6.8. Gráfica longplot para una combinación bivariada	184
6.9. Gráfica de línea bivariada	185
6.10. Matriz de diagramas de dispersión	188
6.11. Diagrama generalizado de pares	189
6.12. Diagrama generalizado de pares	190
6.13. Diagrama de pares de variables de tipo cruzado	191
6.14. Comparación entre diagramas de pares	192
6.15. Diagrama generalizado de pares	193
6.16. Diagrama de pares de variables de tipo cruzado	194

6.17. Diagrama de pares de variables monotipo	195
6.18. Diagrama de pares de variables monotipo	198
6.19. Diagrama de pares de variables de tipo cruzado	199
6.20. Gráfica wideplot	201
6.21. Gráfica wideplot	202
6.22. Diagrama de pares de variables de tipo cruzado	203
6.23. Sección de gráfica longplot	203
6.24. Diagrama de pares de variables de tipo cruzado	204
6.25. Gráfica de barras	205
6.26. Gráfica multipanel de barras	206
6.27. Partes de un diamante	208
6.28. Gráfica wideplot	209
6.29. Diagrama de pares de variables monotipo	210
6.30. Diagrama de pares de variables monotipo	212
6.31. Salida parcial de gráfica longplot	213
6.32. Gráfica de curvas de nivel	213
6.33. Gráfica de curvas de nivel	214
6.34. Gráfica wideplot	217
6.35. Diagrama de pares de variables de tipo cruzado	218
6.36. Histograma estratificado	219
6.37. Diagrama de pares de variables de tipo cruzado	219
6.38. Histograma estratificado	220
6.39. Diagrama de pares de variables monotipo	221
6.40. Diagrama de dispersión por grupos	222
6.41. Diagrama de dispersión por grupos	223

ÍNDICE DE CUADROS

3.1. Tabla estructurada	47
3.2. Tabla no estructurada	48
3.3. Tabla de casos	48
3.4. Tabla de frecuencias	49
3.5. Tabla de contingencia	50
5.1. Valores posibles del argumento <code>group</code>	149
6.1. Combinaciones entre tipos de variables	178

CAPÍTULO 1

INTRODUCCIÓN

“Graphs carry the message home. A universal language, graphs convey information directly to the mind.” — Henry D. Hubbard

Estamos inmersos en una explosión de datos. Antes, la adquisición de datos era más a menudo planificada y los datos más estructurados de acuerdo con las necesidades que el usuario final había previsto. Ahora, en cambio, la gran cantidad de datos almacenados y la facilidad para consultarlos recomienda hacer una prospección de los datos disponibles antes de planificar cualquier campaña de adquisición. Sin embargo, los datos no producen información por si solos y se hace cada vez más necesario utilizar métodos de análisis y comunicación que maximicen la cognición de éstos.

Es en este contexto en el que toma importancia lo que se conoce como Análisis Exploratorio de Datos (o EDA, acrónimo de *exploratory data analysis*). Este análisis consiste en explorar los datos, que generalmente otros han recogido, para entender aspectos como su estructura, las variables que incluyen, la naturaleza de estas variables (como por ejemplo, si son categóricas o numéricas, continuas o discretas), la distribución de los valores de las variables o las relaciones entre los valores de diferentes variables. Para llevar a cabo el análisis, se suelen observar sumarios en forma de tablas que recogen estadísticos de las diferentes variables, o que relacionan variables entre sí. Otra posibilidad es observar las características mediante gráficas, esto se conoce como Análisis Gráfico Exploratorio de Datos (o GEDA, acrónimo de *graphical exploratory data analysis*).

Escoger, sin embargo, qué gráficas estadísticas representar en base a un conjunto de datos no es una tarea evidente. Jaques Bertin (1967, p. 100), el principal precursor de la investigación en la comunicación gráfica de datos y quien llevó al terreno gráfico las teorías de Ferdinand de Saussure en lingüística (Palsky, 2017), hizo la siguiente reflexión: ¿Es necesario hacer una gráfica? La comunicación gráfica, en contraposición con las cadenas de texto o la representación en tablas, se beneficia especialmente de las aproximadamente veinte mil millones de neuronas del cerebro dedicadas a analizar información visual. Una de las mayores ventajas de la comunicación gráfica es la gran cantidad de información que puede ser rápidamente interpretada si está bien representada (Ware, 2004, p.2). Pero escoger una buena representación gráfica no es trivial y para ilustrarlo, Bertin (1967, parte 1, cap. 3) construye, en un ejercicio de virtuosismo, más de cien gráficas a partir de tres variables: la población activa en cada departamento de Francia según los tres sectores económicos principales. De esta forma tan práctica Bertin describe el “problema gráfico”, o expresado de otro modo, de entre el abanico de gráficas posibles. . . ¿Cuál elegir? ¿Existe una gráfica óptima?

Bertin (1967, p.139) pensaba que era posible encontrar gráficas óptimas si se buscaba minimizar el esfuerzo mental necesario para interpretarla, esto es, minimizar el esfuerzo de llevar a cabo tareas perceptivas. Este punto de vista tiene la gran desventaja de que el autor de la gráfica ha de tomar decisiones respecto a ésta presumiendo el receptor sabe cómo interpretarla y no siempre se conoce de antemano quién es el receptor. Facilitar una tarea perceptiva requiere además saber cuál es el propósito de la gráfica dado que la importancia de elegir una gráfica de datos u otra reside en que éstas permiten responder, con mayor o menor eficacia, preguntas que un usuario puede plantearse.

Estas preguntas que se hace el usuario pueden ser recurrentes pero en ningún caso automatizables y puede resultar, especialmente

durante el EDA, que un usuario llegue incluso a explorar datos sin una pregunta predefinida. La generación de gráficas sí que es, en cambio, automatizable, pudiendo derivar el GEDA en lo que podemos llamar Análisis Gráfico Exploratorio de Datos Automatizado (o autoGEDA, acrónimo de *automated graphical exploratory data analysis*) que no es otra cosa que el GEDA llevado a cabo mediante gráficas generadas de forma automática, independientemente de si éstas se benefician de técnicas dinámicas o interactivas.

El encuentro entre unas preguntas no automatizables y la posibilidad de automatizar la generación y presentación de gráficas estadísticas presenta entonces el presente reto: ¿es posible seleccionar de manera automática una gráfica óptima para presentar a un usuario, antes incluso de conocer la pregunta que este usuario se formula?

Los sistemas ideados para presentar gráficas a un usuario, de manera automática, para que éste pueda llevar a cabo la tarea que tiene en mente, se conocen como sistemas de recomendación de gráficas (o VisRec, abreviatura de *visualization recommendation*). A partir de las diferentes estrategias que siguen estos sistemas para abordar el “problema gráfico”, se puede obtener una visión general de los aspectos que sugieren la elección de una u otra gráfica que, a grandes rasgos, se basan en las características de los datos, en las características del usuario o en la tarea a llevar a cabo por éste, en las características del canal de comunicación y/o, finalmente, en las características más o menos concretas del tipo de gráfica buscada.

Entre las estrategias llevadas a cabo por los VisRec, las características de los datos tienen un papel fundamental en el momento de sugerir una determinada gráfica y, entre éstas, aquellas características que se pueden describir de cada una de las variables por separado, pueden ser una clave para mejorar la selección de la gráfica o gráficas a presentar al usuario.

1.1. MOTIVACIÓN

La principal motivación de esta investigación es aportar otra solución al “problema gráfico” con la que enriquecer el conjunto de estrategias de los sistemas VisRec. Entre las diferentes estrategias posibles que pueden conducir a la preferencia de una u otra gráfica, la solución que se busca no está basada en las características de la gráfica estadística porque se pretende encontrar una solución para usuarios no necesariamente expertos. Tampoco está basada en un ámbito de conocimiento concreto, porque se presume que las técnicas que son válidas en un determinado ámbito pueden ser igualmente útiles en otros ámbitos. Tampoco se basa en las características del usuario, ni en el canal de comunicación que suponemos que es una pantalla de un ordenador personal.

Buscamos otra solución no basada en una única gráfica pretendidamente óptima, dado que esto requeriría, entre otras cosas, conocer la pregunta que el usuario tiene en mente y esto supondría una barrera para el propósito de presentar de manera automática las gráficas al usuario. La solución que buscamos tampoco se basa en las relaciones entre los valores de las diferentes variables porque son, precisamente estas relaciones, lo que las gráficas estadísticas hacen aflorar. Así pues, la solución buscada se basa únicamente en las características de los datos y, más concretamente, en las características de las variables por separado.

1.2. OBJETIVOS DE INVESTIGACIÓN

El principal objetivo de esta investigación es presentar un sistema para la recomendación de gráficas estadísticas basado en el número de variables seleccionadas y en las características de estas variables por separado, que facilite al usuario el análisis exploratorio de datos y la selección y elaboración de una gráfica estadística ajustada a su necesidad.

1.3. CONTRIBUCIÓN

Las principales contribuciones de esta investigación son:

- Un compendio de generalidades de la visualización de datos que relaciona los diferentes campos y variantes de la visualización de información, describe los elementos de las gráficas estadísticas y los diferentes propósitos que llevan a producirlas.
- Un compendio de las estrategias que se pueden utilizar para producir una gráfica estadística adecuada con ejemplos de cómo estas estrategias han sido implementadas en diferentes sistemas de automatización de gráficas estadísticas. Se incluyen estrategias basadas en las características de los datos, en las de los usuarios, en las de la gráfica buscada y en las del soporte de ésta.
- Un marco con el que clasificar las gráficas estadísticas a partir de las características de las variables por separado. Las características consideradas son las escalas de medición de las variables, si se trata de observaciones diseminadas o tamizadas, si la sucesión en la que se encuentran las observaciones es o no relevante, si el espacio muestral es cíclico o acíclico, si se requiere o no incluir una escala gráfica de cada variable en particular o si se trata o no de variables dicotómicas.
- La implementación en R de un sistema de recomendación de gráficas estadísticas para asistir al usuario en el análisis gráfico exploratorio de datos. El sistema de recomendación está basado también en las características de las variables por separado, concretamente, en la clase de variable: lógica, carácter, factor o factor ordenado, numérica o temporal. Se incluyen además ejemplos de la utilidad de este sistema de recomendación de gráficas estadísticas.

1.4. ORGANIZACIÓN DE LA DISERTACIÓN

En el capítulo 2 hacemos un repaso de las variantes de la visualización de datos para acotar el campo de las gráficas estadísticas de que son objeto el sistema de recomendación propuesto. Luego introduce los diferentes propósitos de la gráfica estadística para acotar también el uso analítico para el que está pensado dicho sistema de recomendación. Finalmente hace un repaso de los elementos que componen una gráfica estadística, con especial hincapié en la relación entre las variables en los datos y las variables visuales que las representan, para introducir conceptos a los que más adelante nos referiremos.

El capítulo 3 sirve para construir un marco general en el que situar el sistema de recomendación propuesto. El valor de este marco general es que desgrana las estrategias que podemos seguir para obtener una gráfica adecuada, estrategias que generalmente se encuentran entremezcladas en los sistemas de automatización de gráficas estadísticas. Este capítulo incluye referencias a teorías y sistemas que buscan conducir al usuario hacia una gráfica pretendidamente óptima y que se basan en aspectos de lo más variado, como por ejemplo, las características de los datos, las de los usuarios receptores de la gráfica, las del canal de comunicación o las de la propia gráfica buscada.

El capítulo 4 se basa en un artículo publicado (Millán-Martínez y Valero-Mora, 2018) y desgrana en diferentes dimensiones y niveles una serie de características de las variables por separado que pueden ser útiles para acotar el abanico de gráficas a presentar a un usuario. Las características de las variables que utilizamos para clasificar las gráficas son las escalas de medición, el modo de agregación de los datos, la ciclicidad del espacio muestral, la conveniencia de representar la escala de las variables en la propia gráfica y la longitud de la variable. Esta clasificación conduce a un sistema teórico de recomendación de gráficas estadísticas del que también se expone un ejemplo práctico.

El capítulo 5 se basa también en un artículo publicado (Millán-Martínez y Oller, 2020) y presenta, a partir del marco teórico introducido en el capítulo anterior, el paquete ‘brinton’ para la recomendación de gráficas estadísticas. Este paquete está orientado al autoGEDA y está implementado en el sistema de programación estadística **R**. Dado que el artículo se presentó cuando el paquete incorporaba únicamente gráficas univariadas, este capítulo se limita a presentar sus ventajas para el nivel de análisis univariado. Concretamente, se presenta un espécimen de gráficas univariadas y también las funciones `wideplot()` que muestra una matriz de gráficas univariadas que representan diferentes variables de un conjunto de datos, `longplot()` que muestra todo el catálogo de gráficas disponibles para representar una selección de variables de un conjunto de datos y `plotup()` que puede devolver una gráfica específica elegida de manera nominal así como el código de **R** necesario para reproducirla o editarla.

El capítulo 6 expone las mejoras introducidas en el paquete ‘brinton’ desde la publicación del artículo al que hace referencia el capítulo anterior. Concretamente, presenta un espécimen de gráficas bivariadas que amplía la utilidad de las funciones `longplot()` y `plotup()` ya presentadas, y presenta también la nueva función `matrixplot()` que genera gráficas multipanel que combinan variables por pares según la clase de éstas (factor, factor ordenado, numéricas o temporales) y que son específicas para el análisis exploratorio bivariado.

Finalmente, el capítulo 7 concluye la tesis con una discusión sobre los hallazgos presentados y la utilidad de éstos, así como una posible futura línea de investigación en la que puede derivar el trabajo realizado.

CAPÍTULO 2

GENERALIDADES DE LA VISUALIZACIÓN DE DATOS

“La graphique n’est plus seulement la re-présentation de la simplification finale, c’est aussi, c’est surtout, le point de départ exhaustif et l’instrument qui permet de découvrir et de défendre cette simplification.” — Jaques Bertin

El campo de las gráficas estadísticas coexiste con otros múltiples campos que se relacionan y solapan entre ellos. En este capítulo primero repasamos, como antes han hecho Friendly y Denis (2006), diferentes términos que se entrelazan dentro del ámbito de la visualización de datos (o DataVis, abreviatura de *data visualization*), para acotar el campo de las gráficas estadísticas que son el objeto de esta tesis. El compendio de términos que aparecen en este capítulo y que se encuentran, a modo de guía, en la figura 2.1, no pretende ser exhaustivo pero si suficiente para orientarnos en el campo de la DataVis. Luego, en base a la obra de Bertin (1967), repasamos los diferentes propósitos para los que puede servir una gráfica estadística, hacemos un despiece de sus elementos para poner en común los términos que más adelante se utilizarán y describimos el recorrido que se siguió para relacionar por primera vez las propiedades en los datos con las propiedades de las gráficas.

2.1. VARIANTES DE LA DATAVIS

Las gráficas estadísticas de que son objeto esta tesis son una parte de lo que se conoce como “visualización de información” (o InfoVis, abreviatura de *information visualization*) . Este término engloba muchos otros relacionados entre sí que se diferencian entre ellos en

función de las técnicas utilizadas (como por ejemplo la posibilidad de incluir el tiempo o la interactividad), el soporte de la gráfica (por ejemplo el papel contrapuesto a la pantalla), los tipos de datos (por ejemplo, estructurados en una hoja de cálculo o en forma de crónica de un suceso), el carácter figurativo o abstracto, el número de variables implicadas, las características del público al que se dirige o el propósito de las gráficas.

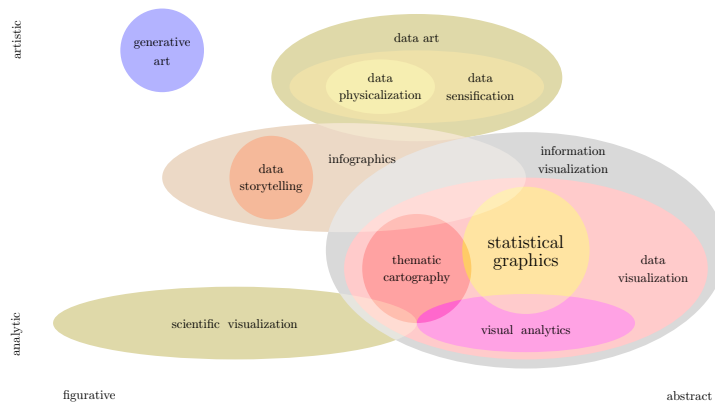


Figura 2.1: Diagrama de Venn en el que se sitúan diferentes términos relacionados con las gráficas estadísticas. El eje de las abcisas representa el grado de abstracción o figuración mientras que el eje de las ordenadas representa el propósito artístico o analítico. Fuente: Elaboración propia.

El sentido del término InfoVis no ha permanecido invariable con el tiempo. Card et al. (1999, p.6), por ejemplo, lo define como “*the use of computer-supported, interactive, visual representations of data to amplify cognition*”, más adelante en el tiempo, el mismo autor escribe “*Information visualization is a set of technologies that use visual computing to amplify human cognition with abstract information*” (Card, 2007, p.542), de modo que la referencia a la interactividad decae de la definición. Hoy en día, el término InfoVis representa cualquier forma de representación gráfica abstracta que se ayuda de las computadoras para hacer llegar una información.

Un subconjunto de la InfoVis es lo que conocemos como “visualización de datos” (o DataVis, abreviatura de *data visualization*) que Friendly (2009, p.2) define como “*the science of visual representation of data, defined as information which has been abstracted in some schematic form, including attributes or variables for the units of information*”. Esta definición no incluye cualquier tipo de representación gráfica sino a la que codifica información bajo variables visuales (veremos en la sección 2.3 a qué se refiere este término) y tampoco se refiere a cualquier tipo de información sino a la que ha sido codificada mediante variables. Asimismo, el autor subdivide la DataVis entre “cartografía temática” que codifica información de datos necesariamente sobre una base cartográfica y “gráficas estadísticas” que se refiere aquellas con el propósito de asistir en el análisis estadístico, cualquiera que sea el método gráfico utilizado.

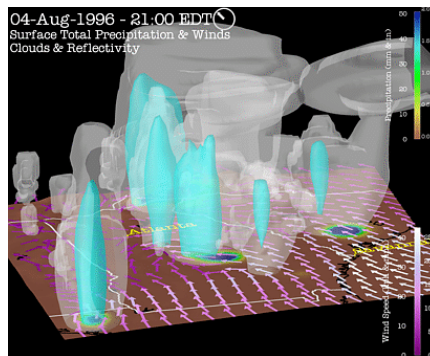


Figura 2.2: Visualización científica de Lloyd Treinish para el pronóstico del tiempo en la clausura de los Juegos Olímpicos de Atlanta en 1996. Esta visualización científica es una captura de una animación que representa un mapa sobre el que se proyecta la dirección y velocidad del viento al mismo tiempo que se recrea en las tres dimensiones de la escena, la evolución de las nubes y las precipitaciones previstas. Fuente: Treinish y Rothfusz (1997).

Existen otros términos cercanos a los comentados, por ejemplo, en el registro científico con una importante componente figurativa encontramos el término “visualización científica” que, a diferencia de

la InfoVis, pone el acento en la gráfica figurativa, en vez de abstracta, para ilustrar fenómenos no visibles y también en la recreación de fenómenos complejos en 3D siempre dentro de un registro científico o de divulgación científica (ver figura 2.2).

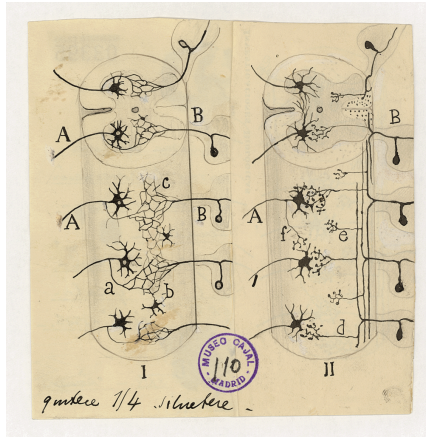


Figura 2.3: Visualización científica de Ramón y Cajal para explicar la diferencia entre las teorías reticular, que entendía la red neuronal como una retícula continua de fibras, y la teoría neuronal que presentaba redes discontinuas. Fuente: Ramón y Cajal (1923).

La visualización científica hoy en día se refiere, básicamente, a las gráficas generadas mediante computadora pero éstas son herederas de las técnicas sobre papel con el mismo propósito, como por ejemplo los dibujos del neurocientífico Ramón y Cajal (ver figura 2.3).

Cercano también a los dibujos de Cajal, pero en el ámbito artístico, encontramos el término “arte de datos” que engloba las creaciones artísticas a partir de datos y que busca impactar, provocar emociones o reflexiones a los espectadores. Amparados por este término solemos encontrar gráficas generadas por ordenador o trabajos derivados de datos procesados por ordenador (ver figura 2.4).

Un término próximo al arte de datos pero con un origen completamente diferente es el “arte generativo” que a diferencia del primero, en vez de utilizar datos, utiliza sistemas autónomos que siguen unas re-

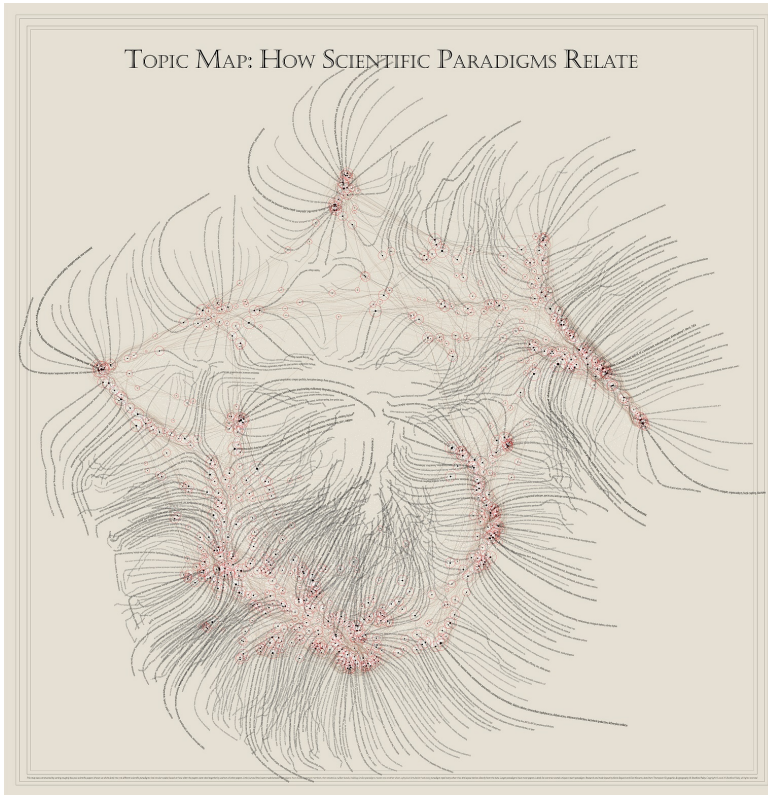


Figura 2.4: Arte de datos a partir del análisis de unos 800.000 artículos científicos agrupados según 776 paradigmas científicos representados por los puntos negros. Los círculos rojos representan la frecuencia de las citas por otros autores y las líneas que enlazan los puntos representan los vínculos entre paradigmas por tener autores en común. La posición relativa de los puntos en la gráfica se establece creando repulsión entre ellos atenuada por los vínculos existentes. Fuente: Paley (2006).

glas con un cierto grado de aleatoriedad para generar obras artísticas (ver figura 2.5).

Otras obras entre el arte de datos y la artesanía se conoce como “fiscalización de los datos” (ver figura 2.6) que consiste en codificar datos en objetos físicos en vez de imágenes (Jansen, 2014). Este término se engloba en lo que podemos llamar “sensificación de los

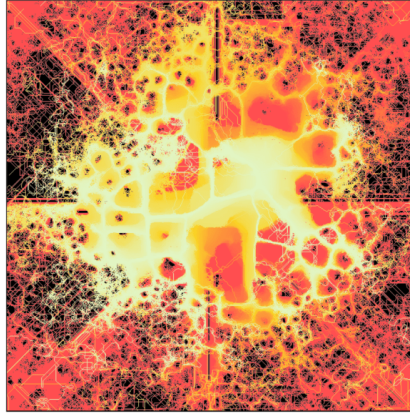


Figura 2.5: Arte generativo en base al modelo Physarum que simula la evolución de una colonia de organismos muy simple que, bajo condiciones específicas, pueden presentar comportamientos complejos. Título: *The Death of a Red Dwarf*. Autor: Alberto Sánchez Chinchón. Fuente: fronkonstin.com. Consultada el 6 de agosto de 2021.

datos” que consiste en codificar datos en objetos, olores, sonidos, gustos o relieves.

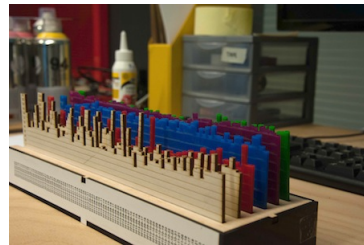


Figura 2.6: Diagrama de barras físico construido con paneles de madera cortados con láser. Experimento llevado a cabo por Aviz (INRIA). Fuente:aviz.fr. Consultada el 28 de agosto de 2022.

En el registro científico-técnico, con importante componente abstracta e interactiva y con un propósito analítico se encuentra el término “analítica visual” que incluye la investigación en campos multidisciplinarios entre los que se encuentran la DataVis y la interacción hombre-máquina (Keim et al., 2010). Una imagen representativa de

este campo son las ventanas interactivas que proporciona el sistema Vista (ver figura 2.7) cuyo desarrollo detallan Valero-Mora et al. (2012).

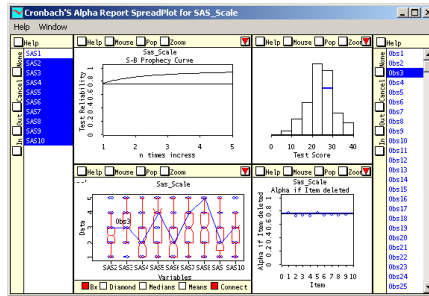


Figura 2.7: Captura de pantalla del software de analítica visual ViSta. Se pueden observar diferentes paneles con gráficas y listados interactivos que facilitan la observación de diferentes aspectos de los datos en su conjunto y, en este contexto, la situación de registros individuales seleccionadas. Fuente: visualstats.org. Consultada el 28 de agosto de 2022.

En el registro divulgativo, en cierto modo artístico, con el propósito de comunicar información por medio de imágenes estáticas o dinámicas, se encuentra el término “infografía” (ver figura 2.8). Este término se encuentra próximo al de *data storytelling* que Riche et al. (2018) define como “*stories that are data-driven in that they start from a narrative that either is based on or contains data and incorporates this data evidence, often portrayed by data graphics, data visualizations, or data dynamics, to confirm or augment a given story*”.

En todo este amalgama de términos y acepciones, las gráficas estadísticas son un subconjunto de la DataVis, que a su vez es un subconjunto de la InfoVis. Las gráficas estadísticas son generalmente generadas mediante computadora, utilizadas con el propósito de asistir en el análisis estadístico ya sea en el registro científico-técnico o divulgativo y para cuyo propósito puede utilizar tanto técnicas estáticas como dinámicas y/o interactivas.

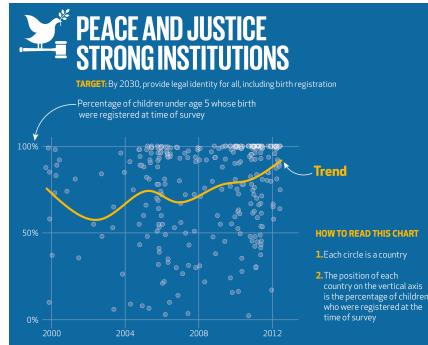


Figura 2.8: Infografía del objetivo núm. 16 de las Naciones Unidas: Promover sociedades pacíficas e inclusivas para el desarrollo sostenible, facilitar el acceso a la justicia para todos y construir a todos los niveles instituciones eficaces e inclusivas que rindan cuentas. Esta infografía representa la evolución en el tiempo del porcentaje de niños menores de 5 años cuyo nacimiento ha sido registrado. La infografía incluye una explicación sobre cómo interpretarla dado que la audiencia a la que se dirige no está necesariamente familiarizada con la interpretación gráfica de datos. Autor: Alberto Cairo. Fuente: ar.pinterest.com . Consultada el 28 de agosto de 2022.

2.2. EL PROPÓSITO DE LA GRÁFICA

Las DataVis tiene diferentes propósitos en función de los cuales las gráficas adoptan diferentes características. Bertin (1967, p.160) distingue los propósitos de registrar, tratar y comunicar la información. En el primer caso tenemos gráficas que se generan automáticamente y que pueden ser consultadas en caso de ser requerido (ver figura 2.9) sin el propósito de ser constantemente analizadas ni memorizadas.

El propósito de tratar la información se refiere a analizarla para así poder extraer información relevante. En este grupo se encuentran las gráficas para el EDA, las gráficas de sistemas para analítica visual y en gran medida las gráficas estadísticas (ver figura 2.10).

El propósito de comunicar surge una vez analizada la información y seleccionada aquella susceptible de ser memorizada (Bertin, 1967, p.160). En este grupo encontramos el campo del arte de datos

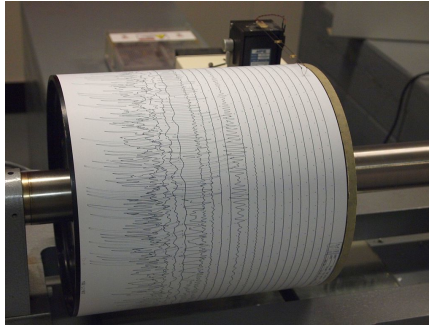


Figura 2.9: Registro de un sismo en el Weston Observatory de Massachusetts como ejemplo de gráfica cuyo propósito es el registro de información. Fuente: commons.wikimedia.org. Consultada el 28 de agosto de 2022.

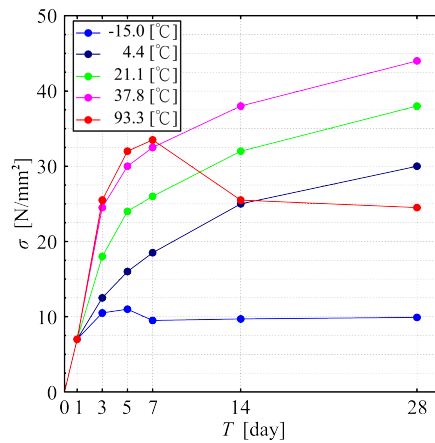


Figura 2.10: Diagrama de evolución de la resistencia a compresión del hormigón con diferentes temperaturas de fraguado. El eje X muestra la edad en días de las muestras de hormigón, el color las diferentes temperaturas de conservación de las muestras y el eje Y la resistencia a compresión hasta el colapso de las muestras, Fuente: commons.wikimedia.org. Consultada el 28 de agosto de 2022.

y el de las infografías, e incluso gráficas de tipos específicos para la comunicación de datos como por ejemplo las gráficas llamadas *Isotype* desarrolladas por Otto Neurath, Marie Neurath y Gerd Arntz hacia 1924 con el objetivo de facilitar la comprensión de los datos estadísticos por una audiencia no experta ni familiarizada con la estadística (ver figura 2.11). Sin embargo, el propósito de comunicar no se ciñe únicamente al registro divulgativo sino que se encuentra también en el registro científico, como ejemplo sirve la figura 2.12, conocida como *Hockey stick graph*, que muestra la variación de la temperatura media en la Tierra durante el pasado milenio.

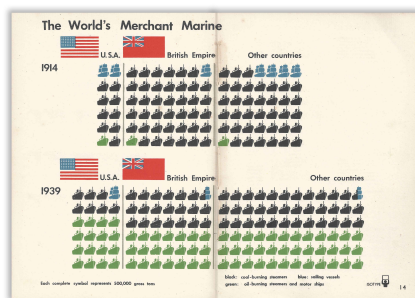


Figura 2.11: Gráfica de comunicación de tipo Isotype. La gráfica compara las flotas de los Estados Unidos, el Imperio Británico y el resto de países, en dos años concretos (1914 y 1939) según la fuente de energía (de vela, carbón o gasóleo) utilizada para su desplazamiento. Estas gráficas permiten comparar valores absolutos a la vez que proporciones entre diferentes recuentos. Fuente: thomwhite.co.uk. Consultada el 28 de agosto de 2022.

Una gráfica adecuada adquiere diferentes características en función de su propósito pero también de su registro de comunicación. Para poder evaluar las cualidades de una gráfica y su adecuación a un determinado destinatario, Costa (1998) propuso descomponer las cualidades de una gráfica según cuatro 4 dimensiones:

- el grado de abstracción o, recíprocamente, de iconicidad.
- el grado de información o, recíprocamente, de redundancia.
- el grado de inteligibilidad o, recíprocamente, de complejidad.

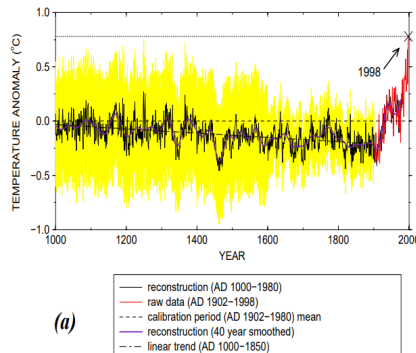


Figura 2.12: Gráfica conocida como *Hockey stick graph* para comunicar la variación de la temperatura media durante el pasado milenio. El eje X muestra la evolución del tiempo mientras que el eje Y muestra la variación respecto a la temperatura media entre 1902 y 1980. El color se utiliza principalmente para representar el intervalo de confianza y la diferente fuente de los datos: aproximaciones a partir de las propiedades del coral y mediciones climáticas. Fuente: Mann et al. (1999).

- el grado de semanticidad, o recíprocamente, de estética.

A partir de este esquema, Cairo (2011) incorporó nuevas dimensiones con las que caracterizar las gráficas según los diferentes ejes de la rueda de tensiones (ver figura 2.13). Una vez descompuestas las características de una gráfica y situados los puntos en cada uno de los ejes, la rueda de tensiones permite obtener una imagen con la que comparar diferentes gráficas y su adecuación a un determinado público.

Los usos de la gráfica están ligados a los avances tecnológicos pero también a los fenómenos culturales. Un ejemplo de recién aparición es el arte de datos (ver figura 2.4), que florece en los años 90, unos 30 años después de la clasificación de Bertin (1967). Por otro lado, el uso de la gráfica como registro de información va perdiendo vigencia porque la fuente de los datos es, cada vez con mayor frecuencia, digital y el coste de almacenar datos digitales es cada vez menor. Los datos se almacenan ahora en forma digital y se presentan en forma gráfica, a

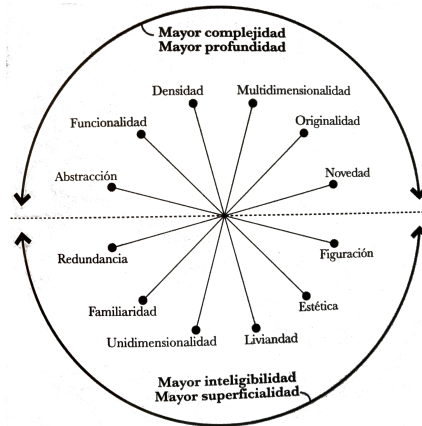


Figura 2.13: Rueda de tensiones de Alberto Cairo con la que mesurar diferentes aspectos de una gráfica y evaluar así su adecuación para diferentes registros de comunicación. Fuente: Cairo (2011).

requerimiento, cuando se hace necesario consultarlos o interpretarlos. La gráfica para el registro de información ha perdido su función pero las características de éstas perduran en las gráficas para el control o monitorización de procesos, especialmente en aquellos que tienen una componente temporal. Un ejemplo de este tipo de gráficas lo encontramos en los monitores de constantes vitales como el de la figura 2.14.

Otra utilidad que ha caído en desuso a partir de la irrupción de la informática (Escribano, 2003) es el cálculo de ecuaciones mediante nomogramas como el de la figura 2.15 que permite deducir el valor de una variable de una ecuación. La nomografía tiene por objeto la construcción de sistemas gráficos planos, llamados “nomogramas”, que corresponden a una relación determinada entre varias variables y permiten la resolución gráfica de ecuaciones cuando se conoce el valor de todas las variables menos una, considerada la incógnita (Belgrano et al., 1953, p.1). Los nomogramas comparten elementos con las gráficas estadísticas si bien las únicas marcas dependientes de los datos son las líneas rectas que se trazan para hallar el valor de la



Figura 2.14: Monitor de signos vitales modelo CMS8000VET de Contec Medical Systems. El monitor muestra las observaciones de diferentes signos vitales como el pulso cardíaco, la respiración o la saturación de oxígeno durante un breve espacio de tiempo. Los registros se almacenan digitalmente de modo que se pueden recuperar y graficar nuevamente si se precisa. Fuente: contecmed.com. Consultada el 28 de agosto de 2022.

variable incógnita. En estas gráficas la posición relativa entre los ejes y su deformación condensan la información de la ecuación de forma análoga a como un mapa de curvas de nivel utiliza estas curvas para representar un tercer eje espacial.

Las gráficas estadísticas sirven especialmente para el tratamiento de la información porque codifican esta información de manera abstracta. También se utilizan para comunicar información en el registro científico-técnico, como por ejemplo, mediante gráficas estadísticas que se editan para acompañar los artículos en revistas científicas. Finalmente, a medida que aumentan las competencias gráficas de una población, las gráficas estadísticas van ocupando cada vez más espacios en la comunicación de información en un registro divulgativo mediante infografías estáticas o dinámicas.

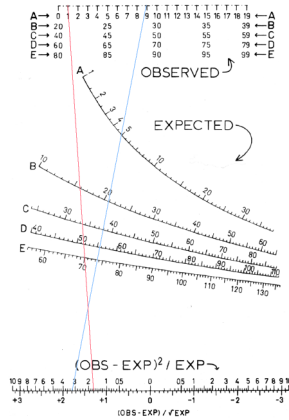


Figura 2.15: Nomograma para el cálculo de la probabilidad χ^2 del test de Pearson. Para calcular la bondad de ajuste entre un valor observado y otro teórico, se fija primero un punto para el valor observado en la parte superior de la figura, se fija luego un segundo punto en cualquiera de los ejes A, B, C, D o E con el valor esperado. La intersección de la línea que une ambos puntos con el eje inferior del nomograma determina el valor buscado. Fuente: commons.wikimedia.org. Consultada el 28 de agosto de 2022.

2.3. ELEMENTOS DE LA GRÁFICA ESTADÍSTICA

Como hemos comentado anteriormente, la gráfica estadística se inscribe dentro de la DataVis. Se requieren necesariamente datos para construir una gráfica estadística ya que sin éstos, la gráfica se convierte en un lienzo vacío sin la posibilidad de aportar información alguna. Bertin (1967, p.16) definía la información así: “*Une information est une série de correspondences observée entre un ensemble fini de concepts de variation ou composantes. Toutes les correspondences doivent répondre à une définition invariable*”. La definición de Bertin, a diferencia de la definición según la Teoría de la información, no se basa en el bit y en la selección entre dos alternativas igualmente probables, sino que define la información a partir una observación.

Según Bertin, la observación de una serie de relaciones entre un conjunto limitado de variables es lo que proporciona información,

ahora bien, estas variables se refieren a las representadas en la gráfica, que Bertin llama “componentes”, el número de componentes ha de ser limitado (Bertin escribe “finito” pero quizás es más adecuado traducirlo como “limitado”) y éstas, a pesar de que tienen su origen en datos adquiridos en una circunstancia concreta, pueden haber sufrido transformaciones. Precisamente es el contexto en el que se inscriben los datos y sus transformaciones (generalmente definido en el título, la leyenda o el pie de figura) lo que Bertin llama “invariable”. La invariable enmarca las relaciones entre las variables representadas en la gráfica y es necesaria para convertir una observación en información.

Esta sección, despieza la gráfica estadística en los diferentes elementos que la componen, tomando como base teórica la obra de Bertin (1967) considerada un tratado fundacional de la comunicación gráfica de datos (Palsky, 2017).

La información

Para que una gráfica pueda contener información se requiere una o más variables en unos datos. Estos datos típicamente se encuentran estructurados en una tabla rectangular y singularmente en formato *tidy data* (Wickham, 2014) según el cual cada variable forma una columna, cada observación forma una fila y cada tipo de unidad observacional forma una tabla. Los datos, sin embargo, pueden encontrarse estructurados en otros formatos como, por ejemplo, en ficheros de texto de lenguaje marcado (como el formato html, xml o json) que contiene información sobre la jerarquía de los datos, o simplemente en cadenas de texto sin marcas, que luego tienen que ser interpretadas a partir de otras características.

A partir de los datos se definen las variables a representar en la gráfica. Las variables en los datos no coinciden necesariamente con las representadas en la gráfica porque las segundas pueden ser variables calculadas a partir de las primeras, y además, porque la gráfica no suele representar todas las variables en los datos sino una

selección limitada de éstas. Cada una de las variables representadas en la gráfica debe poder adquirir, necesariamente, más de un único valor porque, de otro modo, formarían parte del contexto en el que se enmarcan los datos, esto es, la invariable.

Por otro lado, Bertin (1967) define como “longitud de la componente”, el número de valores diferentes que interesa distinguir de la variable representada en la gráfica. La observación de las relaciones establecidas entre los diferentes valores de una o varias variables representadas en la gráfica es lo que proporciona información al observador en el momento de interpretar la gráfica estadística, pero en cualquier caso un número limitado de “componentes” y bajo un contexto común.

Áreas de una gráfica

El espacio físico en el que se proyecta una gráfica se puede dividir en dos áreas: el área interna, o “*Meaningful space*” según von Engelhardt (2002, p.54), y el área externa (ver figura 2.16). El área interna es el área donde se sitúan las marcas que relacionan los valores de las variables representadas y se encuentra en todas las gráficas estadísticas. En el área interna la posición de las marcas está sujeta a un significado independientemente del resto de marcas. El área externa es el área que no puede ser ocupada por las marcas que relacionan los valores de las variables representadas. Generalmente en el área externa encontramos los ejes de coordenadas, las etiquetas de las variables, el título de la gráfica, las leyendas o el pie de figura. Una gráfica estadística podría llegar a prescindir del área externa si no fuera necesario especificar los ejes de las escalas ni el contexto ni las transformaciones de los datos para interpretarla correctamente.

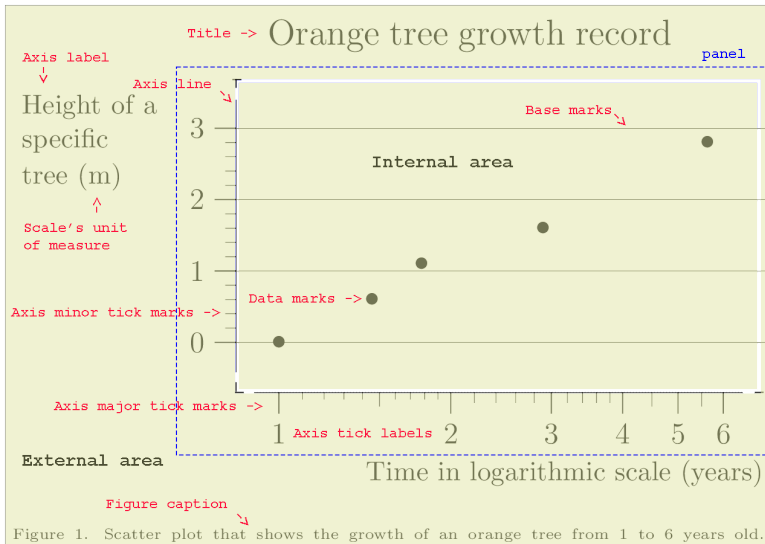


Figura 2.16: Despiece de los principales elementos de una gráfica estadística. Fuente: Elaboración propia

Los grupos de imposición

Bertin (1967) hace una primera clasificación de las gráficas según tres principales grupos (que llama “de imposición”) y distingue entre diagramas, redes y mapas. Los grupos de imposición se pueden entender mejor como si nos referimos a ellos como tipos de correspondencias entre valores de variables representadas en la gráfica. Los valores de variables representadas en la gráfica pueden responder a atributos que describen hechos u objetos, pero también pueden responder a relaciones entre un mismo atributo y pueden responder a relaciones entre atributos o valores de un mismo atributo con respecto a su posición en el espacio. Esta distinción también la explora von Engelhardt (2002, p.30) en lo que llama “estructuras sintácticas” que clasifica las relaciones según si se basan en los atributos de los objetos, la posición de éstos en el espacio o bien en la relación entre los diferentes objetos (ver figura 2.17).

Los diagramas se caracterizan por relacionar valores de una o

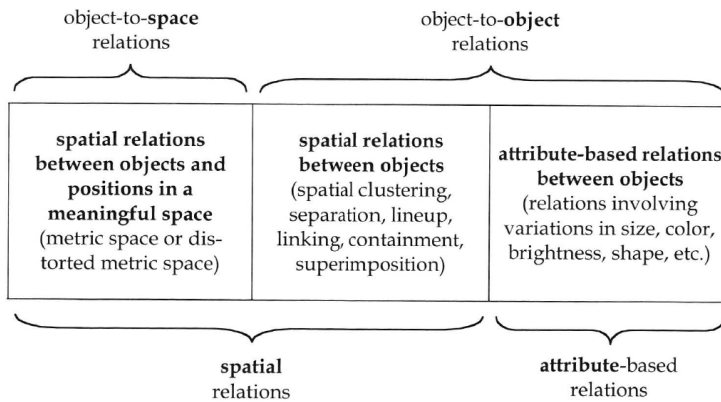


Figura 2.17: Esquema de las diferentes estructuras sintácticas de las representaciones gráficas. El eje superior distingue las relaciones entre las marcas y el espacio, de las relaciones entre las diferentes marcas. El eje inferior distingue entre las relaciones espaciales y las relaciones basadas en atributos de las marcas. En el centro se muestran los tres tipos de estructuras sintácticas que combinan los ejes superior e inferior. Fuente: von Engelhardt (2002, p.30).

diferentes variables representadas en la gráfica (Bertin, 1967, p. 193) como por ejemplo en la figura 2.10 en la que se relaciona la resistencia del hormigón en el tiempo con la temperatura de fraguado. Las redes se caracterizan por relacionar los valores de una misma variable (Bertin, 1967, p. 269), como por ejemplo ocurre con las líneas que unen los puntos negros en la figura 2.4 que enlazan los paradigmas científicos que comparten autores. Los mapas se caracterizan porque las relaciones en el plano se establecen entre los elementos de una misma componente que sigue un orden geográfico (Bertin, 1967, p. 285), esto es, que los valores de la componente mantienen entre ellos una relación espacial como en la figura 2.18. A parte de estos tres tipos de correspondencias, Bertin (1967, p. 51) también identifica los símbolos, como por ejemplo las señales de tráfico, en los que las correspondencias se establecen entre un único elemento del plano y el lector, de modo que las correspondencias son exteriores a la representación gráfica y tiene que ver con el valor simbólico del

elemento.

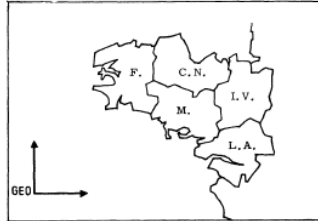


Figura 2.18: Mapa con una componente geográfica que ocupa las dos dimensiones del plano y que representa la situación de diferentes departamentos de la Bretaña en Francia. Fuente: Bertin (1967, p.60).

La implantación

En el área interna, se pueden situar tres tipos elementales de marcas. Bertin (1967) define la “implantación” como cada una de las tres significaciones que una mancha visible puede recibir en relación con las dos dimensiones del plano: puntos, líneas y zonas. Las gráficas estadísticas, se componen de puntos, líneas y zonas, o cualquier combinación de éstas, e incluso, pueden incluir símbolos complejos (también llamados glifos estadísticos (Borner, 2015, p.33)) cuya deformación codifica información. Un ejemplo de lo anterior lo tenemos en el diagrama de caja como los del panel inferior izquierdo de la figura 2.7 que se compone de zonas, líneas y puntos formando un símbolo complejo. No hay que confundir estos símbolos complejos (que podrían considerarse un tipo de implantación adicional junto con los puntos, líneas y zonas) con otros símbolos, como por ejemplo una señal de tráfico, que Bertin (1967) considera un grupo de imposición adicional que hemos comentado anteriormente.

Marcas según su origen

El área gráfica incluye típicamente marcas, cualquiera que sea su implantación, que representan los valores de las variables. El área gráfica no incluye únicamente marcas relativas a los datos sino que puede incluir también otras que asisten en la interpretación de éstos (ver figura 2.16). Tenemos, por ejemplo, las marcas de base que en el caso de implantación puntual podría ser, por ejemplo, el punto que señala el origen de coordenadas. Un ejemplo de líneas de base lo tenemos en las cuadrículas que dividen el área gráfica según las escalas de los ejes de coordenadas. También tenemos las familias de curvas acotadas y redes acotadas (Belgrano et al., 1953) que son marcas más habituales en gráficas de ingeniería que en gráficas estadísticas y que tampoco dependen de los datos, sino que representan soluciones a ecuaciones que se pueden graficar de antemano, y que dividen el área gráfica en zonas mediante curvas que representan unos ejes diferentes a los de coordenadas.

Un ejemplo de familias de curvas acotadas lo tenemos en el ábaco de roseta (ver figura 2.19) y otro ejemplo lo tenemos en el nomograma de la figura 2.15. Finalmente, en el área gráfica encontramos anotaciones que contextualizan los resultados o las anotaciones que un analista ha podido incluir para asistir a los demás en la interpretación de los datos, una vez analizados éstos.

Las variables visuales

Bertin (1967, p.42) propuso delimitar este sistema de signos mediante lo que llamó “variables visuales” (de ahora en adelante VV) en un intento de analizar la gráfica como un sistema de marcas. Hay que considerar que Bertin se refería en este caso a las gráficas estáticas imprimibles en una hoja de papel blanco de un formato *medio* que permitiera percibir la gráfica de un vistazo, con luz normal y constante y a una distancia también constante. En este marco, las

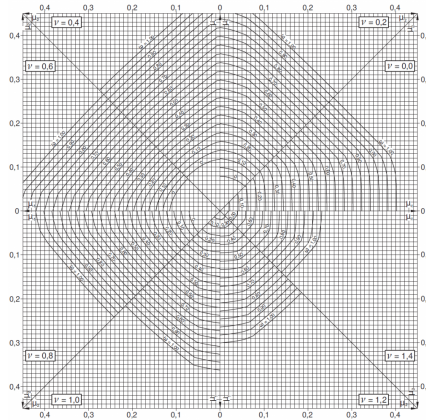


Figura 2.19: Ábaco de roseta para una sección de hormigón armado sometida a flexión compuesta, esto es, en dos direcciones. Este ábaco permite deducir la cuantía de acero necesaria (w) para armar una sección rectangular de hormigón sometida a compresión y flexión esviada. Fuente: estructurando.net. Consultada el 28 de agosto de 2022.

herramientas de que dispone el diseñador para representar unos datos son las manchas que se pueden someter a diferentes variaciones. Estas posibles variaciones Bertin las categoriza en dos grandes grupos: las VV espaciales y las VV de elevación o de retina.

Las VV espaciales son dos y coinciden con las dos dimensiones del plano sobre el que se proyecta la gráfica y que, generalmente, coinciden con la posición de las marcas respecto a los ejes X e Y. Las VV de elevación, en cambio, son las variaciones a las que puede estar sometida cualquier mancha de un tamaño visible situada en un punto del plano entre las que Bertin identifica 6: tamaño, luminosidad, grano, tono de color, orientación y forma (ver figura 2.20).

Dado que la clasificación de Bertin se ceñía al caso de gráficas estáticas e impresas sobre papel, después de él, otros autores han considerado otras VV, por ejemplo, para el caso de gráficas dinámicas e interactivas (MacEachren (2004), Roth (2017), Borner (2015, p.34)) e incluso su mismo planteamiento ha servido para relacionar las

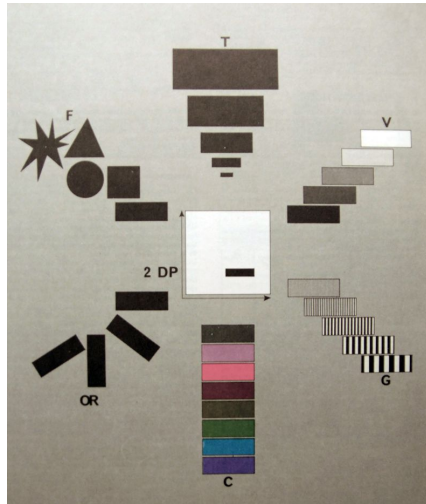


Figura 2.20: VV consideradas en la teoría de Bertin. En el centro encontramos las VV espaciales (ejes X e Y del plano) y alrededor las VV de elevación (en el sentido de las agujas del reloj: tamaño, luminosidad, grano, tono, orientación y forma). Fuente: Bertin (1967, p. 43).

propiedades de los datos con variaciones de los sonidos (MacEachren, 2004, p.289) en vez de variaciones de las marcas sobre un trozo de papel.

PROPIEDADES PERCEPTIVAS BÁSICAS DE LAS VV Bertin (1967) vincula la eficiencia de una gráfica con las capacidades cognitivas del usuario para lo que lleva al campo de la gráfica la noción de eficiencia de Zipf (1935), según la cual, se mantiene un equilibrio entre la frecuencia en la que se observan las palabras en un texto y su longitud. En base a esa noción de eficiencia, una gráfica eficiente es aquella que minimiza el coste mental de realizar diferentes tareas cognitivas en el proceso de *lectura* de la gráfica. Bertin (1967, p.65-69) inicia el camino hacia una gráfica eficiente identificando 4 “niveles de organización de las VV” y que aquí preferimos llamar “propiedades perceptivas básicas”:

- Asociativa: Permite asociar todos los valores de una variable visual sin que unos valores estimulen más la atención que otros.
- Selectiva: Permite aislar todas las marcas de un determinado valor de la variable visual de un vistazo sin tener que fijar la atención en cada una de las marcas para evaluar su valor.
- Ordenada: Permite establecer relaciones de mayor o menor entre diferentes marcas de una variable visual.
- Cuantitativa: Permite establecer relaciones de proporcionalidad entre diferentes marcas de una variable visual.

RELACIÓN ENTRE LAS VV Y LAS PROPIEDADES PERCEPTIVAS BÁSICAS

Una vez identificadas las 4 propiedades perceptivas básicas, Bertin clasifica las VV según las tareas perceptivas básicas que éstas facilitan, relación que queda plasmada en figura 2.21. Según la clasificación de Bertin, las variables espaciales posibilitan las 4 tareas perceptivas básicas, el tamaño no facilita el nivel asociativo, la luminosidad solo facilita los niveles selectivo y ordenado, el grano no facilita el nivel cuantitativo, el tono tan solo facilita los niveles asociativo y selectivo, la orientación solo facilita el nivel asociativo y parcialmente el selectivo (en el caso de implantación en punto y línea) y finalmente, la forma facilita únicamente el nivel asociativo. Esta primera relación entre las VV y las propiedades perceptivas básicas para “leer” una gráfica, no está basada en experimentos científicos sino en la propia experiencia e intuición de Bertin, pero establece una base teórica con la que iniciar el camino hacia una gráfica eficiente.

RELACIÓN ENTRE LAS PROPIEDADES PERCEPTIVAS BÁSICAS Y LAS

ESCALAS DE MEDICIÓN Después de identificar las tareas perceptivas básicas y clasificar las VV según las tareas que posibilitan, Bertin (1967) relaciona las tareas perceptivas básicas con las escalas de medición de las variables representadas en la gráfica (en vez de

**NIVEAUX D'ORGANISATION
DES VARIABLES VISUELLES**

DIMENSIONS DU PLAN	≡	≠	○	Q
TAILLE	≠	≠	○	Q
VALEUR	≠	≠	○	
GRAIN	≡	≠	○	
COULEUR	≡	≠		
ORIENTATION	≡	≠		Implantations P et L
FORME	≡			

Figura 2.21: Relación entre las VV y las propiedades perceptivas básicas que facilitan. Las filas ordenan las VV (de arriba a abajo: VV espaciales, tamaño, luminosidad, grano, tono, orientación y forma) según las propiedades perceptivas básicas que facilitan y que se encuentran representadas en las columnas, de izquierda a derecha: asociativa (equivalencia), selectiva (desigualdad), ordenada (O) y cuantitativa (Q). Fuente: Bertin (1967, p.69).

escalas de medición utiliza el término *niveles de organización de las componentes*), entre las que distingue tres escalas: cualitativa, ordenada y cuantitativa. Para la escala cualitativa elige el mismo símbolo que para la tarea selectiva (desigualdad), para la escala ordenada el mismo símbolo que para la tarea ordenada (O) y para la escala cuantitativa, el mismo símbolo que para la tarea cuantitativa (Q), de modo que la tarea asociativa queda huérfana

Puede sorprender que Bertin (1967) no utilice las escalas de medición de Stevens (1946) (nominal, ordinal, intervalar y de razón) sino tres únicos niveles (cualitativo, ordenado y cuantitativo). El nivel cualitativo equivale a la escala nominal, el cuantitativo equivale a la escala de razón y el ordinal agrupa las escalas ordinal e intervalar.

RELACIÓN ENTRE LAS TAREAS PERCEPTIVAS ELEMENTALES Y LAS VV La relación entre las tareas perceptivas básicas y las escalas de

medición inspiró más adelante a Cleveland y McGill (1984) quienes evaluaron el acierto en la decodificación de cantidades a partir de diferentes codificaciones gráficas de los datos y que ponían a prueba diferentes “tareas perceptivas elementales”, como por ejemplo, comparar la posición de dos puntos en una escala común, o en escalas no alineadas, comparar dos ángulos o la longitud de dos segmentos de recta (ver figura 2.22).

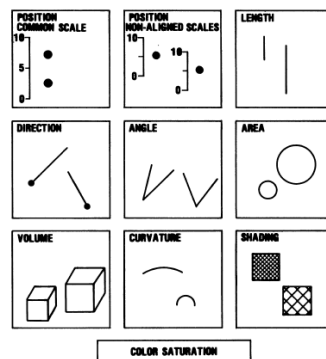


Figura 2.22: Clasificación de las tareas perceptivas elementales que Cleveland y McGill utilizaron para medir la precisión en la decodificación de variables cuantitativas. Cada panel identifica una tarea perceptiva diferente entre las incluidas en el estudio. De izquierda a derecha y de arriba a abajo: comparar la posición de dos puntos en una escala común o en escalas no alineadas, comparar la longitud o la orientación de dos segmentos de recta, los ángulos entre dos pares de segmentos de recta, el área de dos superficies, el volumen de dos cuerpos proyectados en perspectiva isométrica, la curvatura de dos segmentos, el sombreado o la saturación de color de dos objetos. Fuente: Cleveland y McGill (1984).

RELACIÓN ENTRE LAS VARIABLES EN LOS DATOS Y LAS VV A partir de los resultados de los estudios de Cleveland y McGill (1984), Mackinlay y Genesereth (1985) reeditaron la figura 2.22 de modo que las diferentes tareas perceptivas elementales quedaban ordenadas según la precisión con la que se pueden decodificar los valores de las

variables cuantitativas representadas. El resultado puede observarse en la figura 2.23.

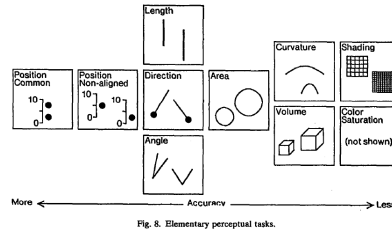


Fig. 8. Elementary perceptual tasks.

Figura 2.23: Reinterpretación de la figura 2.22 que de Mackinlay y Genesereth utilizaron para mostrar las diferentes tareas perceptivas elementales evaluadas en sus experimentos. De izquierda a derecha y de arriba a abajo: comparar la posición de dos puntos en una escala común o en escalas no alineadas, comparar la longitud o la orientación de dos segmentos de recta, los ángulos entre dos pares de segmentos de recta, el área de dos superficies, la curvatura de dos segmentos, el volumen de dos cuerpos proyectados en perspectiva isométrica, el sombreado y la saturación de color de dos objetos. Fuente: Mackinlay y Genesereth (1985).

Una vez establecido el orden de precisión de diferentes codificaciones gráficas para las variables cuantitativas, Mackinlay (1986) se aventuró a presentar un orden de precisión que relaciona las diferentes codificaciones, con las escalas de medición de las variables en los datos (ver figura 2.24) que se derivan de las escalas de medición de las variables representadas en la gráfica de Bertin (1967). Esta clasificación de Mackinlay carecía de evidencia científica pero su interés radica en que por primera vez se utiliza una relación directa entre la variable visual y la escala de medición de la variable en los datos con el objetivo de automatizar una gráfica estadística (Mackinlay, 1986).

Los ejes

Los ejes son el soporte de las escalas. En el caso de las variables espaciales, los ejes que les dan soporte son los ejes de coordenadas. En el caso de las variables de elevación, el soporte de sus escalas se

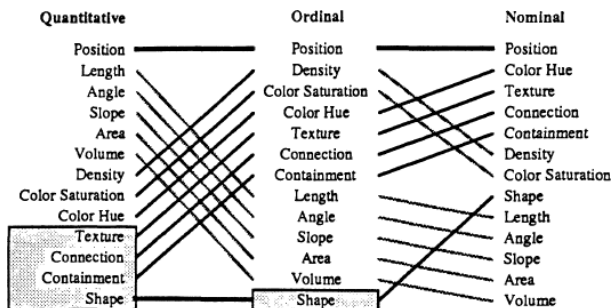


Figura 2.24: Ranking de las VV en función de la eficacia con la que se decodifican, para cada escala de medición de las componentes. Las VV en recuadros no son relevantes para estas determinadas escalas. Esta figura ejemplifica que si bien el tono de color no se recomienda para codificar variables cuantitativas (se sitúa como octava mejor elección), resulta más apropiado para representar variables ordinales e intervalares (se sitúa como tercera mejor elección) y muy recomendable si se trata de codificar variables nominales (se sitúa como segunda mejor opción justo detrás de la posición en el plano). Fuente: Mackinlay (1986).

encuentra en las leyendas. A veces las variables de elevación se substituyen por familias de puntos, de curvas o redes acotadas (Belgrano et al., 1953) que proyectan las variables de elevación sobre el área gráfica y que constituyen a su vez el soporte de la escala. En este caso, las familias de puntos, curvas o redes acotadas incluyen etiquetas en el área gráfica que fijan los valores de la escala (ver la figura 2.19).

Los elementos de los ejes de coordenadas son las etiquetas de los ejes y la escala gráfica. Las etiquetas de los ejes suelen indicar la variable que codifican y la unidad de medición considerada. La escala gráfica del eje incluye una sucesión de marcas de graduación, junto con etiquetas de graduación de la escala (ver figura 2.16) y también otras marcas secundarias de graduación de las escalas que no llevan asociada ninguna etiqueta de graduación dado que su valor se deduce por interpolación entre los valores de las etiquetas contiguas. Los ejes suelen incluir también la curva del eje que une los puntos sobre los que deben interpolarse las lecturas de las marcas de escala.

Gráficas multipanel

La combinación del área interna junto con los ejes de coordenadas se conoce como “panel” y, como gráfica multipanel, aquella que incluye diferentes áreas gráficas, cada una de ellas con unos ejes de coordenadas que pueden o no ser compartidos entre los diferentes paneles (ver figura 2.25).

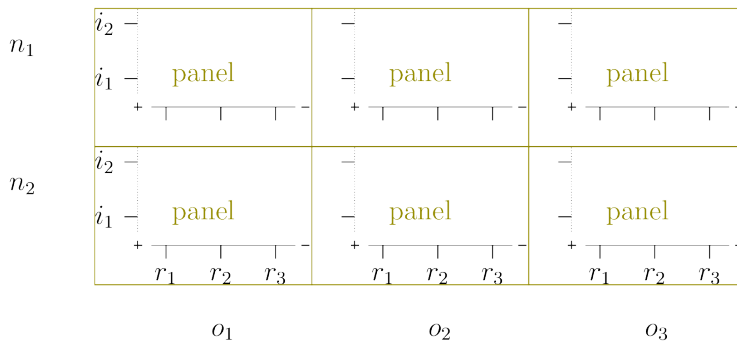


Figura 2.25: Despiece de una gráfica multipanel en sus diferentes paneles. Fuente: Elaboración propia.

Existen diferentes tipos de gráficas multipanel en función del número de tipos de gráficas diferentes que incluyen y el origen de los datos que utilizan. Por un lado tenemos los paneles de mando (o *dashboards*), como el de la figura 2.26 que generalmente incluyen diferentes gráficas y que utilizan diferentes tablas de datos. Los paneles de mando se utilizan para controlar procesos complejos.

Si los diferentes paneles con diferentes gráficas se construyen a partir de un mismo conjunto de datos, entonces obtenemos unas gráficas multipanel equivalentes a las *SpreadPlots* del sistema Vista (Valero-Mora et al., 2012) como el de la figura 2.27. En el caso de Vista, además, las gráficas de los diferentes paneles son interactivas, se encuentran enlazadas (de modo que las acciones realizadas en un panel tienen incidencia en los demás) y algunos de los paneles tienen

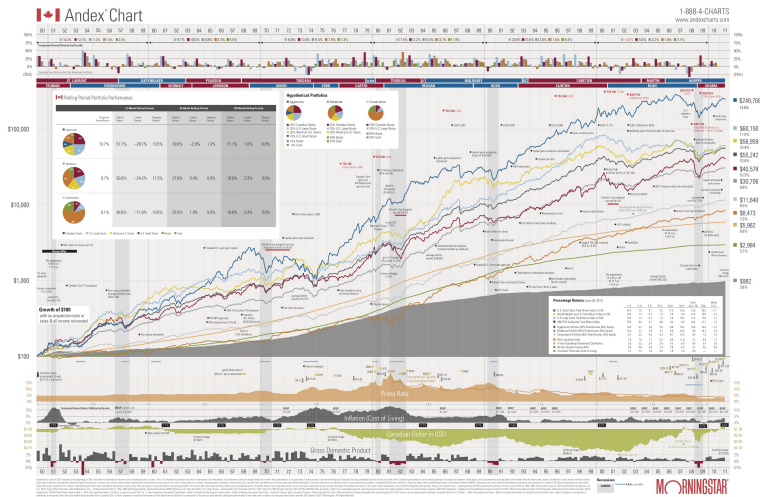


Figura 2.26: Los paneles de control Andex Chart son útiles para ayudar a los consultores a explicar las tendencias, las expectativas de interés y los riesgos asociados a las inversiones en diferentes grupos de acciones en bolsa. Fuente: datavis.ca. Consultada el 28 de agosto de 2022.

propiedades dinámicas.

Otro tipo de gráfica multipanel es la que se conoce como gráfica condicionada (o *cooplot*) como el de la figura 2.28 (Theus, 2016). Estas gráficas se componen de diversos paneles que comparten las escalas de las variables gráficas y el tipo de gráfica, cada uno referido a un subconjunto de registros según los diferentes niveles de una tercera o incluso cuarta variable. Las gráficas representadas en los diferentes paneles se nutren de un solo conjunto de datos. No existe un término comúnmente aceptado para este tipo de gráficas; Bertin (1967, p.26) se refiere a *séries homogènes*, Tufte (1983) se refiere a *small multiples* y Cleveland (1985, p.200) los llama *juxtaposed panels*, mientras que en el entorno de S-plus se conocen como *trellis displays* (Becker et al., 1996) que se adaptan luego a R como *lattice graphics* (Sarkar, 2008) o *facet plots* (Wickham, 2009, sec. 2.5).

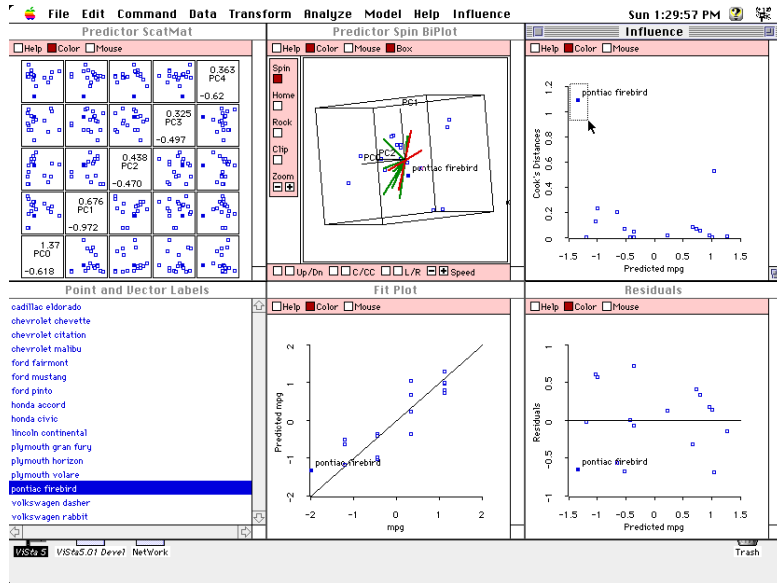


Figura 2.27: SpreadPlot para modelos de regresión del sistema Vista. En el panel superior derecho se puede observar el puntero que, en seleccionar una marca, se relaciona ésta con el Pontiac Firebird. Automáticamente, las marcas de este registro quedan identificadas en el resto de paneles. Fuente: visualstats.org. Consultada el 28 de agosto de 2022.

Otro grupo de gráficas multipanel es el conocido como diagrama de pares, *matrix of plots* o *pairs plot*, que se compone generalmente de paneles que forman una matriz cuadrada en la que cada celda representa una gráfica que combina las variables dos a dos, dispuestas tanto en las filas como en las columnas. La diagonal de esta matriz suele contener el nombre de la variable, una gráfica univariada, o se encuentra simplemente vacía. Originariamente, este tipo de gráficas se limitaba a combinar variables numéricas mediante diagramas de dispersión (Hartigan, 1975), pero más tarde se fueron añadiendo nuevos elementos como por ejemplo líneas de tendencia (ver figura 2.29) e implementando nuevos tipos de gráficas útiles también para pares de variables numéricas. Cabe destacar que los diagramas de pares pueden llegar a tratar las variables categóricas como si fue-

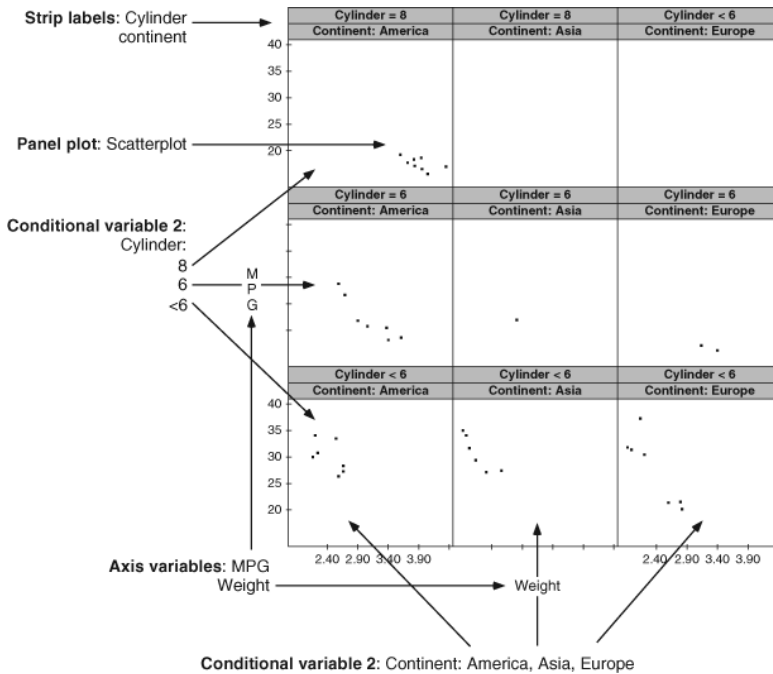


Figura 2.28: Esquema de gráfica multipanel condicionada, en este caso, a dos variables: número de cilindros y continente. Fuente: Theus (2016).

ran numéricas con el objetivo de representarlas todas en un único diagrama de pares.

Una evolución de los diagramas de pares la encontramos cuando los pares de variables, en una misma gráfica, pueden ser de cualquier tipo. En este caso, los tipos de gráficas resultantes son también diferentes según sea la combinación de tipos de variables y la gráfica multipanel resultante se conoce como diagrama generalizado de pares (o *generalized pairs plot*) como por ejemplo el de la figura 2.30 (Emerson et al., 2013, Friendly (2014)).

Finalmente, si las combinaciones son entre variables de dos tipos diferentes y siempre entre estos mismos tipos, y si las relaciones entre pares de variables se representan mediante un único tipo de gráfica repetida en cada celda de la matriz, obtenemos una matriz rectangular

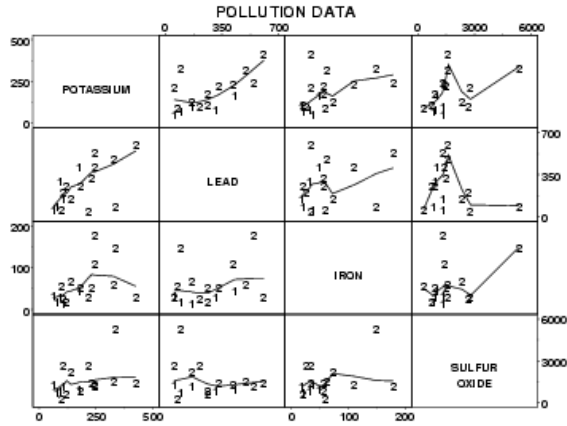


Figura 2.29: Diagrama de pares producido por el software Dataplot a partir de datos de polución. La matriz, en este caso, combina por pares cuatro variables numéricas y representa cada combinación mediante dos diagramas de dispersión con los ejes transpuestos. Fuente: itl.nist.gov. Consultada el 28 de agosto de 2022.

de gráficas como las que produce la función `brinton::matrixplot()` cuando relaciona variables de diferente tipo (ver el capítulo 6) y que llamamos “diagramas de pares de variables de tipo cruzado”.

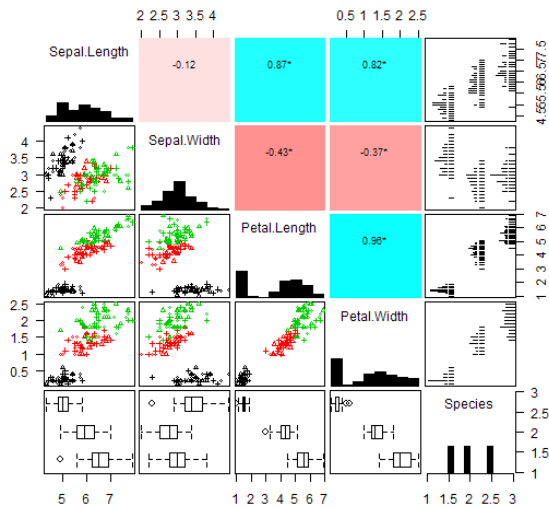


Figura 2.30: Diagrama generalizado de pares de variables producida por la función 'gpairs()' del paquete homónimo de R. La gráfica combina hasta 6 tipos diferentes de gráficas a partir de 4 tipos diferentes de combinaciones entre variables de tipo numérico y de tipo factor. Fuente: r-bloggers.com. Consultada el 28 de agosto de 2022.

AUTOMATIZACIÓN DE GRÁFICAS ESTADÍSTICAS

“A graph can answer the important, though often implicit, question, *compared to what?* It can convey a sense of uncertainty of evidence for the validity of a claim.” —
Michael Friendly

Antes de proponer cualquier sistema de recomendación de gráficas estadísticas como los que veremos en los capítulos 4 y 5, conviene revisar las diferentes estrategias que se han planteado para la automatización de las gráficas estadísticas, para procurar tener una visión general del problema. La revisión de diferentes sistemas y teorías permite anticipar las ventajas y las limitaciones que puede tener un sistema de recomendación de gráficas estadísticas en función de los mecanismos utilizados para recomendar una u otra gráfica.

Los sistemas de recomendación de gráficas estadísticas se pueden clasificar en dos grandes grupos según si éstos interpretan qué datos mostrar o si en cambio interpretan aspectos de la gráfica a mostrar (Wongsuphasawat et al., 2016).

En el primer grupo encontramos sistemas que utilizan estadísticos para evaluar qué datos tienen características que podrían interesar al usuario, como por ejemplo HCE 3.0 (Seo y Shneiderman, 2005), AutoVis (Wills y Wilkinson, 2010) o SeeDB (Vartak et al., 2015). Encontramos también sistemas especializados en hallar relaciones entre pares de variables continuas como Scagexplorer (Dang y Wilkinson, 2014) o el paquete de R `scagnostics` (Wilkinson et al., 2005, Wilkinson y Anand (2018)) que basan la selección de las gráficas en estadísticos que evalúan, por ejemplo, la posible presencia de valores

atípicos, la densidad o la forma que adquieren los contornos de las observaciones en diagramas de dispersión.

Otro caso de recomendadores de qué datos mostrar lo encontramos en sistemas que, en vez de filtrar las variables a mostrar, filtran los registros a mostrar, como por ejemplo Discovery-Driven OLAP (Sarawagi et al., 1998) para explorar valores atípicos en cubos de datos¹ sobre las que dirigir la atención. Otro ejemplo más reciente es VisPilot (Lee et al., 2019) que asiste en la exploración de subconjuntos de datos mostrando diferentes y sucesivos desgloses para guiar el análisis. El propósito de este trabajo no se dirige hacia la recomendación de qué datos mostrar por lo que este grupo de recomendadores de gráficas estadísticas no lo describimos con mayor detalle.

El segundo grupo de recomendadores de gráficas estadísticas, al que nos referimos a partir de ahora cuando hablamos de automatización de gráficas estadísticas, pone el acento en los aspectos de la gráfica a mostrar. Dado que el propósito de este trabajo es precisamente la recomendación de tipos de gráficas estadísticas, las diferentes estrategias que siguen estos sistemas las describimos con detalle en este capítulo. En este grupo encontramos sistemas que seleccionan automáticamente gráficas estadísticas a partir de una selección de datos que previamente ha hecho el usuario. La gráfica o gráficas presentadas, a diferencia de lo que ocurre con los recomendadores de datos a mostrar, son una representación de los datos seleccionado por el usuario y no una selección automática de un subconjunto de éste.

La automatización de gráficas estadísticas requiere de un algoritmo que interprete unos datos y que, en función de éstos, aplique una serie de normas para producir una representación gráfica. Las

¹Un cubo OLAP (*OnLine Analytical Processing*) es una base de datos multi-dimensional que puede asimilarse a una generalización de una tabla de calculo para n dimensiones. De este modo, las dimensiones del cubo se corresponden con las dimensiones de la tabla y el valor almacenado en cada celda del cubo equivale a la métrica o métricas almacenadas en la tabla.

diferentes normas a aplicar determinan qué estrategias de automatización se llevan a cabo para presentar una gráfica estadística. Como hemos apuntado en el capítulo 1, podemos clasificar las estrategias según si se basan en las características de los datos (sección 3.1, en las características del usuario receptor de la gráfica (sección 3.2), en las del canal de comunicación (sección 3.3) o, finalmente, en las características más o menos concretas del tipo de gráfica buscada (sección 3.4). La figura 3.1 extraída de (Millán-Martínez y Valero-Mora, 2017) sirve de esquema de las diferentes estrategias y subaspectos de éstas que se desarrollan a continuación.

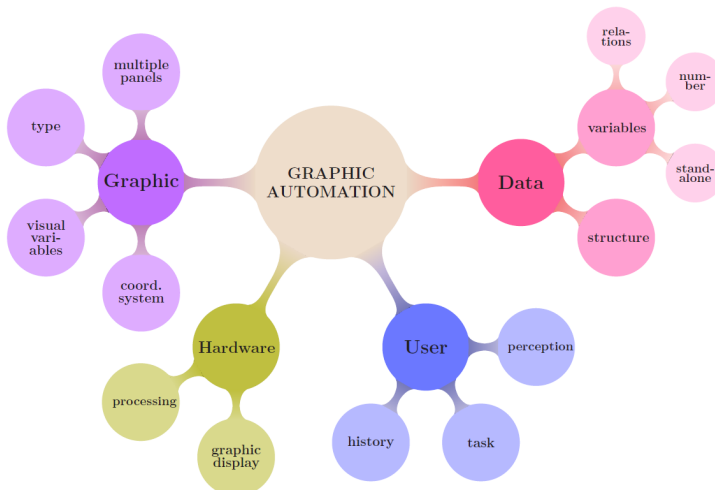


Figura 3.1: Mapa conceptual de las estrategias de recomendación de tipos de gráficas estadísticas. Fuente: Millán-Martínez y Valero-Mora (2017).

3.1. LOS DATOS

Todos los sistemas recomendadores de aspectos de la gráfica a mostrar, tienen en cuenta alguna característica de los datos en el momento de presentar una u otra gráfica. Kamps (1999) utiliza el término *functional design* para referirse a los métodos que determinan

el tipo de gráfica a presentar a partir de estas características de los datos que, como se detalla a continuación, se pueden referir al número de variables que considerar en los datos, a las características de cada variable en particular, a las relaciones entre las variables, a la estructura de los datos o a la procedencia de éstos.

El número de variables

El número de variables en los datos ha dado paso a una clasificación clásica de las gráficas entre univariadas, bivariadas o multivariadas, siendo las gráficas univariadas aquellas que requieren una única variable en los datos, las bivariadas las que requieren dos y las multivariadas las que requieren tres o más.

Aunque generalmente los sistemas de automatización de gráficas se refieren al número de variables en los datos que sirven de entrada para generar la gráfica, a nivel teórico se han presentado otras aproximaciones. Bertin (1967), por ejemplo, en su recorrido para hallar la gráfica más eficiente para cada caso, considera, en vez del número de variables en los datos, el número de variables representadas en la gráfica y clasifica las gráficas según este criterio. Bertin utiliza el término “número de variables” cuando se refiere a las VV y el término “número de componentes”, ya introducido en la sección 2.3, cuando se refiere a las variables representadas en la gráfica. Otro ejemplo es el trabajo de Bachi (1968) que clasifica las gráficas en función del número de variables en los datos pero distingue el papel de las variables cuantitativas, a las que llama “variables”, del de los demás tipos de variables. Esta concepción de Bachi es parecida a la distinción que hacen los cubos de datos entre dimensiones y medidas. En este caso las dimensiones son los valores cualitativos como nombres, fechas o agrupaciones geográficas en las que se pueden desglosar las medidas (o valores numéricos cuantitativos que Bachi llama “variables”).

Por otro lado, si identificamos el número de variables en los datos con el número de columnas en un conjunto de datos, tenemos que tener en cuenta que una misma información puede estructurarse de manera diferente, como por ejemplo ocurre en las tablas de los cuadros 3.1 y 3.2 en las que número de columnas varía entre 4 y 5 pero encontramos 4 únicas variables en los datos que corresponden a las 4 columnas que vemos en el cuadro 3.1.

Otro ejemplo de cómo unos mismos datos pueden adoptar diferentes estructuras con diferente número de variables, lo encontramos en la diferencia entre la tabla de casos, en la que cada fila representa un caso, y la tabla de frecuencias en la que cada fila representa una combinación posible entre variables y una nueva columna representa el recuento de filas agrupadas según esa combinación de variables (Friendly y Meyer, 2015, p.5). Si el recuento de frecuencias se hace respecto de dos variables, otra solución posible es mediante una tabla de contingencia en la que la primera columna y la primera fila contienen los diferentes valores de cada una de las variables y en el resto de celdas las frecuencias concomitantes.

Para ilustrar las diferencias entre la tabla de casos, de frecuencias y de contingencia vamos a utilizar el conjunto de datos `TwinsLungs` del paquete `Stat2Data`. El cuadro 3.3 muestra todo el conjunto de datos y es un ejemplo de tabla de casos que cuenta con 16 registros de 3 variables, una numérica y dos de tipo factor. A partir de esta tabla de casos, podemos obtener un recuento de la frecuencia concomitante

	country	year	cases	population
1	Afghanistan	1999	745	19987071
2	Afghanistan	2000	2666	20595360
3	Brazil	1999	37737	172006362
4	Brazil	2000	80488	174504898
5	China	1999	212258	1272915272
6	China	2000	213766	1280428583

Cuadro 3.1: Tabla estructurada en la que cada columna representa una variable diferente.

	Country	cases_1999	cases_2000	population_1999	population_2000
1	Afghanistan	745	2666	19987071	20595360
2	Brazil	37737	80488	172006362	174504898
3	China	212258	213766	1272915272	1280428583

Cuadro 3.2: Tabla no estructurada en la que diferentes columnas representan una única variable.

de los valores de las variables `Pair` y `Environ` y obtener, por ejemplo, la tabla de frecuencias del cuadro 3.4, o bien en forma de tabla de contingencia como la del cuadro 3.5 .

	Pair	Environ	Percent
1	A	Rural	10.1
2	B	Rural	51.8
3	C	Rural	33.5
4	D	Rural	32.8
5	E	Rural	69.0
6	F	Rural	38.8
7	G	Rural	54.6
8	A	Urban	28.1
9	B	Urban	36.2
10	C	Urban	40.7
11	D	Urban	38.8
12	E	Urban	71.0
13	F	Urban	47.0
14	G	Urban	57.0

Cuadro 3.3: Tabla de casos en la que cada registro observado está representado en una fila.

Si nos desplazamos del ámbito teórico al práctico, tenemos la herramienta *A Presentation Tool* (Mackinlay, 1986) que hace uso de la “expresividad” como criterio en la selección de una gráfica. Para Mackinlay, un conjunto de hechos es expresable en un lenguaje si éste contiene una frase que codifica todos los hechos del conjunto y solamente estos hechos. Este criterio lleva a seleccionar gráficas que no utilizan más VV (ver figura 2.20) que las estrictamente necesarias, de modo que el número de variables en los datos resulta determinante en la selección de la gráfica adecuada (ver figura 3.2).

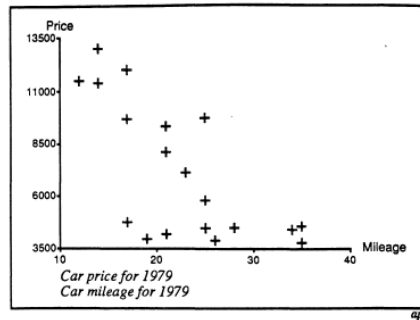


Figura 3.2: Diagrama de dispersión producido por el sistema APT que, en base a un conjunto de datos, representa la sentencia *Present the Price and Mileage relations. The details about the set of Cars can be omitted.* Fuente: Mackinlay (1986).

Las características intravariabile

Después del número de variables, el elemento más utilizado en la selección de gráficas estadísticas, son las características de las variables por separado, estas son las características que se pueden atribuir a

	Pair	Environ	Freq
1	A	Rural	1
2	B	Rural	1
3	C	Rural	1
4	D	Rural	1
5	E	Rural	1
6	F	Rural	1
7	G	Rural	1
8	A	Urban	1
9	B	Urban	1
10	C	Urban	1
11	D	Urban	1
12	E	Urban	1
13	F	Urban	1
14	G	Urban	1

Cuadro 3.4: Tabla de frecuencias en la que cada fila representa una combinación diferente de las variables **Pair** y **Environ**, y una nueva columna representa el recuento de filas agrupadas según esa combinación de variables.

cada variable en los datos. Algunas características, como por ejemplo el número de observaciones, el número de valores únicos observados o el número de valores perdidos, se pueden deducir implícitamente de cada columna de datos, pero otras características, como por ejemplo si el cero es una referencia o implica ausencia de magnitud, el carácter cíclico (que al crecer vuelve al punto de origen) de las variables o su consideración como variable predictora o de respuesta, requiere una declaración expresa o bien una suposición.

Una primera característica de las variables por separado que puede interesar determinar es su escala de medición y, por la implicación que tienen en el campo de la estadística, la referencia de partida son las escalas de medición de Stevens que describimos a continuación. Conviene aclarar que la escala de medición de una variable, que es la relación entre los valores de una variable, no se deben confundir con la escala gráfica que se encuentra dibujada en la propia gráfica y que permite decodificar los valores graficados de una variable.

CARACTERIZACIÓN DE STEVENS La clasificación más utilizada de las escalas de medición es la planteada por Stevens (1946), en función del orden y las distancias o proporciones que se puede establecer entre los valores de las variables. Stevens diferenció inicialmente entre

Environ		
Pair	Rural	Urban
A	1	1
B	1	1
C	1	1
D	1	1
E	1	1
F	1	1
G	1	1

Cuadro 3.5: Tabla de contingencia en la que cada fila representa una valor diferente de las variables **Pair** y cada columna (a excepción de la primera) un valor diferente de la variable **Environ**. El resto de celdas muestra la frecuencia concomitante entre valores de ambas variables.

cuatro tipos de escalas de medición: nominal, ordinal, intervalar y racional.

- **NOMINAL** si toma valores de un conjunto de palabras o números no ordenados, por lo que intercambiar el orden no altera la información y en el que carece de sentido realizar operaciones aritméticas. Aunque los conjuntos estén compuestos de números, éstos identifican igualmente categorías y no valores numéricos. Entre valores de una variable nominal sólo caben comparaciones de igualdad y diferencia: $\forall a, b \in N; a = a, a \neq b$. Pyle (1999) hace además la distinción entre variables nominales y variables categóricas; se refiere a nominales si asignan un nombre a un elemento individual y a categóricas si se refiere a un conjunto de elementos o una propiedad de éstos. Ejemplos de variables nominales son el número de identificación fiscal o los nombres de personas o, referidas a agrupaciones de elementos, el código postal es una variable categórica.
- **ORDINAL** si toma valores de un conjunto de palabras o números ordenados. En este caso no cabe la posibilidad de intercambiar el orden ni de realizar operaciones aritméticas. La variable ordinal categoriza un elemento a la vez que le asigna un orden dentro del conjunto: $\forall a, b, c \in O; a < b, b < c \Rightarrow a < c$. Un ejemplo es la clasificación de dureza de los minerales de Mohs de más blando **1-talco** a más duro **10-diamante**. En esta clasificación se puede utilizar tanto la escala numérica como la alfanumérica y al estar ordenada sí cabe preguntarse qué material es mas duro.
- **INTERVALAR** si el conjunto continuo o discreto de valores, además de etiquetar y ordenar, permite establecer intervalos iguales entre sus valores y en que el cero no implica ausencia de magnitud sino simplemente una referencia. Estas variables permiten

realizar operaciones de adición y sustracción. Ejemplos de variables intervalares son la temperatura en grados Celsius, las coordenadas geográficas o las cotas sobre el nivel del mar.

- **RACIONAL** si tiene las características de la intervalar pero además el cero implica ausencia de magnitud. Estas variables permiten realizar operaciones aritméticas más complejas y gozan de proporcionalidad. Ejemplos de variables de esta escala son el coste de un producto o la velocidad de un vehículo, de modo que si el coste es 0, el producto es gratuito y si la velocidad es 0, el vehículo permanece inmóvil.

La distinción automática de las variables nominales y ordinales respecto de las variables intervalares y racionales puede normalmente deducirse de forma implícita en los datos dado que las primeras suelen estar almacenadas como cadenas de texto y las segundas como numéricas. No es inusual, sin embargo, encontrar categorías codificadas como numéricas como por ejemplo la variable `status` del conjunto de datos `prostateSurvival` del paquete `asaur` que adquiere valores del conjunto $\{0, 1, 2\}$. Más difícil resulta diferenciar implícitamente si una variable es nominal u ordinal porque, por poner un ejemplo, los elementos del conjunto $\{\text{costa, interior, montaña}\}$ pueden ser simplemente tres destinos turísticos o pueden considerarse ordenados si nos estamos refiriendo a cuencas hidrográficas. Finalmente, para deducir implícitamente si una variable tiene una escala de medición intervalar o racional, la dificultad es parecida, por poner otro ejemplo, los valores de la variable `diag` del conjunto de datos `Aids2` del paquete `MASS` están codificados como números enteros lo que conduce a pensar que se trata de recuentos, pero en realidad se refieren al número de días transcurridos desde el 1 de enero de 1960.

CARACTERIZACIÓN DE BERTIN Stevens sentó un precedente pero no basa la anterior clasificación en los diferentes tipos de gráficas a

los que cada escala puede conducir. Tuvieron que pasar veinte años hasta que, ya en el ámbito de la representación gráfica de datos, Bertin (1967) adoptara las escalas de medición según su utilidad para escoger entre unas u otras gráficas, adopción que, en vez de “escalas de medición de las variables”, nombró “niveles de organización de las componentes” (ver sección 2.3) dado que se refería a las escalas de medición de las variables representadas en la gráfica. Bertin considera únicamente las siguientes tres escalas de medición:

- NIVEL CUALITATIVO, el de las variables cuyos valores no guardan una relación de orden universal sino que pueden ser ordenados de formas diversas.
- NIVEL ORDENADO, el de las variables cuyos valores guardan una relación de orden universal y las distancias entre valores sucesivos es constante.
- NIVEL CUANTITATIVO, el de las variables entre cuyos valores se pueden precisar la variación de distancia.

Cabe destacar que, a diferencia de Stevens que inicialmente clasificó las escalas en cuatro categorías, Bertin distingue únicamente tres porque une las escalas ordinal e intervalar en un único nivel ordenado. Si la escala ordinal de Stevens se caracterizaba por establecer un orden entre los valores pero no una distancia, las mismas variables pasan a ser incluidas en el nivel ordenado de Bertin, que se caracteriza por mantener constante la distancia entre sus valores. Esta diferencia se debe a que Bertin define las escalas de medición a partir de las variables graficadas en una hoja de papel y esto implica una posición en el plano y, por consiguiente, una distancia entre sus valores. La figura 3.3 muestra como la variable “tiempo” en el que se suceden acontecimientos, que es típicamente una medida intervalar en la clasificación de Stevens, se encuentra clasificada en el nivel ordenado en el esquema de Bertin.

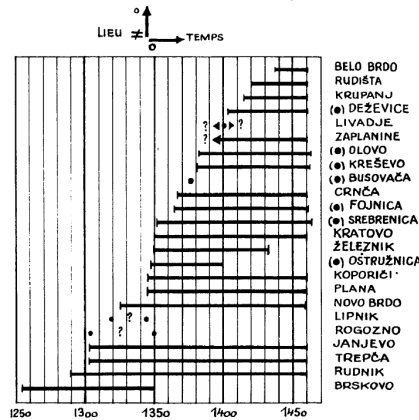


Figura 3.3: Diagrama de Gantt con los periodos de explotación de diferentes minas de plata de Yugoslavia. En el eje Y encontramos las diferentes minas que están clasificadas como una variable cualitativa y, dado que carecen de un orden natural, las minas han sido aquí ordenadas según la fecha aproximada de inicio de la explotación. En el eje X encontramos el paso de tiempo que ha sido clasificado como una variable ordenada. Las barras gruesas horizontales representan el periodo entre el inicio y el fin de la explotación. Fuente: Bertin (1967, p.198)

Otra característica que diferencia las escalas de medición ordinal e intervalar de Stevens es que una escala ordinal suele estar compuesta de valores alfanuméricos que tienen que estar todos representados en los ejes de la gráfica (ver sección 2.3). En cambio, una escala intervalar se compone de valores numéricos o de tipo fecha y los valores observados se suelen interpolar entre los valores mostrados por las etiquetas de graduación de las escalas. Esta diferencia implica que los ejes de las gráficas requieran más o menos espacio en función de si soportan un tipo de variable u otra.

Bertin (1967) une, como hemos comentado, las escalas ordinal e intervalar de Stevens en un único nivel ordenado pero tiene en cuenta parcialmente el diferente comportamiento de los ejes cuando introduce una nueva característica intravariante que resulta útil en la elección de una gráfica eficaz. Esta nueva característica, que llama “longitud

de las componentes”, se refiere al número de valores diferentes de una variable representada en la gráfica que interesa poder identificar. Bertin distingue, en el caso de variables discretas, dos niveles en función del número de valores que es útil diferenciar:

- Las componentes “cortas”, que cuentan como máximo con 4 valores representados.
- Las componentes “largas”, que cuentan con más de 15 valores representados.
- Las componentes que tienen entre 5 y 15 valores no forman un grupo diferenciado sino que pueden corresponder tanto a las construcciones gráficas estándar para componentes largas, como a los casos especiales para componentes cortas.

El primer sistema que adopta las teorías de Bertin para producir gráficas de manera automática es el sistema CHART (Benson y Kitous, 1977). El sistema CHART genera tablas de contingencia (ver sección 3.1) semigráficas que guardan semejanza con las matrices físicas ordenables impulsadas por Bertin (1977) (ver figura 3.4) pero sin las características interactivas.

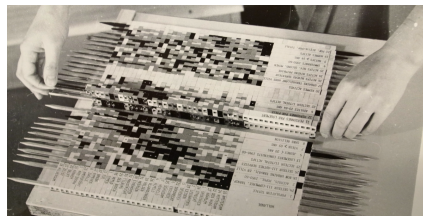


Figura 3.4: Sistema Dominó para el análisis gráfico de matrices de datos. El sistema Dominó representa una tabla de contingencia construido a partir de piezas de madera que representan cada celda. Mediante un sistema de agujas, se pueden permutar tanto filas como columnas y así agrupar los valores de las variables categóricas en función de sus características observadas. Fuente: dataphys.org. Consultada el 28 de agosto de 2022.

En la figura 3.5 podemos ver un ejemplo de una de las tablas producida por el sistema CHART en la que podemos identificar hasta 5 variables representadas en la gráfica:

- Área geográfica. Variable cualitativa desplegada en el eje Y.
- Productos o servicios. Cualitativa desarrollada en el eje X.
- Desviación del coste respecto al coste medio. Cuantitativa desarrollada en el eje X pero sin escala gráfica.
- Signo de la desviación. Cualitativa desarrollada por la luminosidad (rectángulo sin trama si se encuentra por encima de la media y con trama si se encuentra por debajo).
- Existencia o no de valores. Cualitativa desarrollada por la forma (rectángulo en el caso de valores informados y asterisco en el caso de valores perdidos).

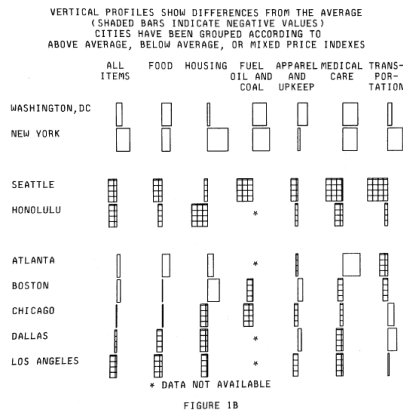


Figura 3.5: Tabla semigráfica producida por el sistema CHART que codifica 5 variables: Área geográfica, productos o servicios, desviación respecto al coste medio, signo de la desviación y la existencia o no de valores informados. Fuente: Benson y Kitous (1977).

Otras iniciativas que derivan de las matrices ordenables de Bertin son los sistemas MATRIX, AMADO o Voyager (Perin et al., 2018) entre otros hasta acabar con Bertifier (Perin et al., 2014) que reelabora

la técnica de Bertin pero esta vez emulando también por computadora las técnicas dinámicas e interactivas del sistema físico original.

CARACTERIZACIÓN DE BACHI Otra forma de clasificar las relaciones intravariante según las gráficas que sugieren, es la propuesta por Bachi (1968), que clasifica los datos según la “naturaleza” de la variable (por ejemplo cuantitativa, cronológica o geográfica) y según el “orden de la secuencia”, es decir, el modo como los valores se suceden unos a otros. Esta doble consideración le lleva a identificar tipos y subtipos de características:

- Variables con características “lineales”, en las que los valores se suceden unos a otros en un orden natural o secuencial. Entre éstas distingue las siguientes variables:
 - “Cuantitativas”, que se equipara a las escalas intervalar y racional de Stevens.
 - “Temporales” que se refieren a intervalos más o menos extendidos de tiempo.
 - “Cualitativas lineales” que se refiere a la escala ordinal de Stevens.
- Variables “circulares” en las que los valores se suceden unos a otros en un orden natural pero no tienen valores extremos. Estas variables se refieren a variables cuyo espacio muestral es cíclico como, por ejemplo, las coordenadas geográficas o la dirección del viento.
- Variables “geográficas” cuyos valores tienen correspondencia con entidades geográficas.
- Variables “cualitativas no ordenadas” cuyos valores no guardan ninguna relación de orden ni tampoco relación con entidades geográficas.

Esta clasificación de Bachi presenta tres novedades interesantes respecto a la clasificación de Bertin. Por un lado divide el nivel ordenado de Bertin en dos categorías, las variables temporales y las cualitativas lineales. Esta división acerca más las escalas de medición a la propuesta por Stevens (1946) dado que la escala nominal estaría representada por las variables cualitativas no ordenadas, la escala ordinal por las variables cualitativas lineales, la escala intervalar se asociaría en parte con las variables temporales y en parte con las variables cuantitativas junto con la escala de razón. Por otro lado introduce las variables cuyo dominio es cíclico y, por último, distingue entre las variables cualitativas no ordenadas y las variables geográficas.

CARACTERIZACIÓN DE MOSTELLER Y TUKEY A pesar de que tampoco ha sido implementada en ningún sistema de representación gráfica de datos, la clasificación de las variables de Mosteller y Tukey (1977) resulta interesante porque relaciona las escalas de medición con diferentes conjuntos de números que son habituales en los conjuntos de datos. La escala que proponen es la siguiente:

- “Números”
 - “Recuentos”, es decir, números naturales.
 - “Cantidades” que son números reales positivos.
 - “Balances” o números reales de signo positivo o negativo.
 - “Fracciones contadas” del tipo ‘trece de cada cien’.
 - “Rankings” en los que 1 es el mayor o menor número y 2 el mayor o menor que sigue.
 - “Grados”, es decir, categorías ordenadas pero en este caso no numéricas.
- “Nombres” que, a diferencia de los grupos anteriores, son categorías alfanuméricas que no guardan relación de orden.

No resulta evidente relacionar estas variables con las escalas de Stevens dado que aquí se introduce el signo que pueden adquirir los valores y si los valores son enteros o reales. Las variables que parecen relacionarse unívocamente con las escalas de Stevens son los “nombres” que corresponden a la escala nominal, los “grados” a la escala ordinal y los rankings con la escala intervalar. Por otro lado se introducen las fracciones contadas que corresponderían a la escala “absoluta” definida por Stevens posteriormente que tiene las propiedades de la racional pero además se encuentra acotada inferior y superiormente.

CARACTERIZACIÓN DE GNANAMGARI El sistema BHARAT (Gnanamgari, 1981) se presenta 4 años después del sistema CHART y se basa en una clasificación totalmente diferente para determinar qué gráfica escoger. El sistema escoge la gráfica a presentar según un árbol de decisión (ver figura 3.6) a partir de la información que el usuario introduce acerca de las siguientes características intravariante:

- Continuidad: Un valor booleano que indica si las observaciones de valores ordenados tienen que representarse formando una secuencia ordenada. Esta propiedad da lugar a gráficas como series temporales.
- Totalidad: Un valor booleano que indica si un conjunto de valores representa la totalidad de valores de partes de un objeto o de un concepto abstracto. Esta propiedad da lugar a gráficas que muestran proporciones respecto de un total.
- Cardinalidad: El número de valores de un conjunto de etiquetas. Esta propiedad da lugar a gráficas para variables dicotómicas.
- Multiplicidad: El número de valores asignados a cada elemento en un conjunto de etiquetas (en una tabla de doble entrada, la multiplicidad sería 2). Esta propiedad da lugar a gráficas multipanel.

(cualitativo, ordinal y cuantitativo), las series temporales las incluye en el nivel cuantitativo en vez del ordinal.

- El carácter como “coordenadas o cantidades” que se refiere a la división de la escala cuantitativa en dos niveles, las *coordenadas* que son valores con carácter de referencia temporal o espacial, mientras que las *cantidades* guardan referencia respecto al valor cero.
- El “dominio de pertenencia” reconoce el sustrato cultural que aconseja la utilización de unos ejes de coordenadas u otros en función de las magnitudes de las variables. Requiere determinar las magnitudes de las diferentes variables tales como temporales, espaciales, de temperatura o de masa.
- La “aridad” que indica el número de valores de una variable que distinguen las marcas de una gráfica. Roth y Mattis (1990) establecen tres niveles de relación diferentes: la relación unaria en la que las marcas representan un solo valor de una variable, la relación binaria en la que las marcas representan dos valores y la relación n-aria en la que las marcas representan más de dos valores.

La principal novedad de esta caracterización es la correspondencia de las 4 escalas de Stevens con el orden del conjunto más las coordenadas o cantidades en que se subdividen las variables cuantitativas. El dominio de pertenencia es una característica que puede resultar útil, por ejemplo, para asignar a una variable el eje de las ordenadas o de las abscisas pero su utilidad decae si se requiere que el usuario caracterice las variables de antemano. En lo que respecta a la aridad, esta característica guarda relación con las componentes cortas o largas de Bertin pero, igual que el sistema BHARAT, SAGE lo utiliza, básicamente, para diferenciar las variables dicotómicas del resto.

CARACTERIZACIÓN DE KAMPS La clasificación de Kamps (1999) se diferencia de las demás en que toma prestados las clases estándar de datos, a partir de las cuales establece operaciones específicas para cada clase. Los tipos estándar de datos que utiliza son los siguientes:

- Números enteros
- Números reales
- Booleanos
- Cadenas de texto
- OID (Identificadores de objeto)
- Conjunto vacío

Esta aproximación tiene la ventaja evidente de que, si las variables se encuentran correctamente caracterizadas en el conjunto de datos, se evita tener que consultar al usuario acerca de las características de las variables, por lo que se puede subir un peldaño en la automatización de la gráfica. Si la gráfica propuesta no es la adecuada porque la caracterización de las variables en el conjunto de datos no es óptima, entonces, más allá de modificar el tipo de gráfica, se presenta una oportunidad para modificar la caracterización de las variables mal codificadas y volver a solicitar la generación de la gráfica.

CARACTERIZACIÓN DE THEUS Martin Theus (Unwin et al., 2006, cap. 3) divide las variables categóricas en tres grupos según el conocimiento que se tiene acerca de la longitud de las variables. Esta distinción, aunque no se tiene constancia de que haya sido implementada en ningún sistema de automatización de gráficas, resulta útil para establecer las dimensiones de los ejes X~Y si estos ejes soportan escalas categóricas, de modo que todas las etiquetas de la escala tengan cabida.

- Variables “fijas” cuyo número de categorías es conocido, independientemente de su número (por ejemplo *grupo sanguíneo* o *continente*).

- Variables “limitadas a priori” cuyo número de categorías no se conoce de antemano pero es acotado y puede ser estimado (por ejemplo “fabricante de automóvil” o “país de origen”).
- Variables “sin límites conocidos”, lo conforman las variables cuyo número de valores crece a medida que se incrementan las observaciones (por ejemplo “empresa empleadora”, “ciudad de origen” o “vino favorito”).

CARACTERIZACIÓN DE MACKINLAY, HANRAHAN Y STOLTE Si bien el sistema APT de Mackinlay (1986) implementa tal cual las escalas nominal, ordinal y cuantitativa de Bertin (1967), más adelante renueva esta aproximación. Mackinlay et al. (2007) presentan Show Me, germen del software de exploración gráfica de datos Tableau, que caracteriza las variables según las siguientes cuatro dimensiones:

- El *tipo de dato* entre los que incluye texto, fecha, fecha y hora, numérico o booleano. Como en la clasificación de Kamps (1999), si el conjunto de datos se encuentra correctamente caracterizado, no requiere que el usuario aporte información adicional dado que se trata de clases de variables comúnmente utilizadas por los sistemas informáticos.
- El rol como *dimensión o medida* de la variable, que se relaciona con su consideración como variable predictora o de respuesta. Igual que pasa con el tipo de dato, es posible aproximar el rol estableciendo por defecto las variables numéricas como medidas y el resto como dimensiones, de modo que sea el usuario quien recodifique el rol si es que prefiere otro.
- La consideración como variable *continua o discreta* que puede ser inferido también a partir del tipo de dato, el número de observaciones y el número de observaciones únicas. Esta distinción afecta especialmente, más que a la selección del tipo de gráfica, a la construcción de las escalas gráficas.

- El *tipo de marca* a utilizar entre los que distingue: texto, barra, línea, forma o diagrama de Gantt.

Para preseleccionar el tipo de gráfica Mackinlay et al. (2007) utilizan principalmente el tipo de dato y su rol como dimensión o medida. Para acabar de determinar el tipo de gráfica, requiere además la selección de un tipo de marca. A diferencia del tipo de dato y del rol como dimensión o medida, el tipo de marca no resulta fácilmente predecible, por lo que establecen unas reglas heurísticas con las que automatizar la selección del tipo de marca a partir de la naturaleza categórica o cuantitativa de las variables del conjunto de datos y su consideración como dependiente o independiente (ver figura 3.7). De este modo se evita el inconveniente de tener que preguntar a un usuario, no necesariamente experto, sobre el tipo de marca a utilizar.

Esta clasificación resulta novedosa por dos motivos. Por un lado porque utiliza el rol de variable como dimensión o medida, términos prestados de los cubos de datos (ya comentados al inicio de este capítulo), para recomendar una u otra gráfica. Las variables de tipo dimensión son variables cuyos valores se emparejan con una observación o a un estadístico de la variable de tipo medida. Las variables de tipo medida son generalmente variables numéricas cuantitativas. Si para cada valor de la variable de tipo dimensión existe más de un valor de la variable de tipo medida, entonces se calculan sumatorios, recuentos, promedios, etc. Ejemplos de variables de tipo dimensión en un conjunto de datos de inmuebles a la venta pueden ser: municipio, distrito postal, ascensor {si, no}, tipo de vivienda, negocio {venta, alquiler}. Ejemplos de variables de tipo medida del mismo conjunto de datos pueden ser: superficie en metros cuadrados, número de habitaciones, número de baños, precio deseado.

Otra característica novedosa de esta clasificación es la consideración como continua o discreta de las variables de tipo medida. Esta

característica modifica la construcción de los ejes de las variables de tipo medida.

Table 1: Automatic marks rules

Pane Type		Mark Type	View Type
Field	Field		
C	C	Text	Cross-tab
Qd	C	Bar	Bar view
Qd	Cdate	Line	Line view
Qd	Qd	Shape	Scatter plot
Qi	C	Gantt	Gantt view
Qi	Qd	Line	Line view
Qi	Qi	Shape	Scatter plot

Figura 3.7: Reglas heurísticas utilizadas por Show Me para definir el tipo de gráfica a presentar. La selección se lleva a cabo en función de la caracterización como categórica o cualitativa con el caso particular de las fechas {C, Q, Qdate}, el carácter dependiente o independiente {d, i} que se deduce del rol como dimensión o medida y el tipo de marca seleccionada. Fuente: Mackinlay et al. (2007).

Las características intervariable

Después del número de variables y de las características de las variables por separado, se puede caracterizar también las relaciones entre pares de variables aunque su utilidad no está exenta de dificultades. Una primer problema radica en que, a medida que se añaden variables a un conjunto de datos, el número de relaciones entre sus variables crece exponencialmente. Un segundo problema radica en que, si ya resulta difícil preguntar a un usuario, no necesariamente experto, acerca de las características de las variables por separado, más difícil puede resultar que un usuario no familiarizado con un conjunto de datos, pueda aportar información acerca de las relaciones que se establecen entre pares de variables. Finalmente, hay que considerar que existen relaciones que no se observan en los datos pero que son potencialmente observables y éstas pueden determinar también

el tipo de gráfica adecuada con la que mostrar las relaciones entre una selección de variables.

A pesar de lo expuesta en párrafo anterior, existen sistemas de recomendación de gráficas que utilizan las relaciones intervariable, entre los cuales destacan el sistema APT (Mackinlay, 1986) y SAGE (Roth y Mattis, 1990). A continuación detallamos las características identificadas en sendos sistemas.

CARACTERIZACIÓN DE MACKINLAY Mackinlay (1986) identifica una primera relación entre las variables que llama “dependencia funcional”, término prestado de la notación de bases de datos (Ullman, 1980), y es la relación por la cual cada elemento de una variable se corresponde, como máximo, con un único elemento de otra variable, en otros términos, se refiere a la correspondencia unívoca. Esta relación es útil para seleccionar gráficas como, por ejemplo, un diagrama de barras que relaciona diferentes modelos de vehículos con diferentes variables con las que el modelo de vehículo guarda una correspondencia unívoca (ver figura 3.8).

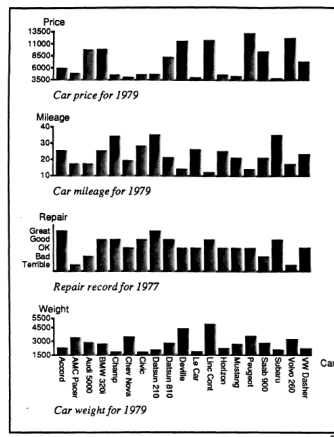


Figura 3.8: Gráfica producida por el sistema APT a partir de las variable 'Car' y la correspondencia unívoca de esta respecto a la variables 'Price', 'Mileage', 'Repair' y 'Weight'. Fuente: Mackinlay (1986, p.32).

Una correspondencia no unívoca la encontramos por ejemplo en la relación “se subdivide” en la que un elemento se relaciona con más de un elemento del conjunto imagen. Un ejemplo de esta relación lo tenemos en los años que se subdividen en estaciones, de modo que una construcción posible es la gráfica de barras acumuladas de la figura 3.9.

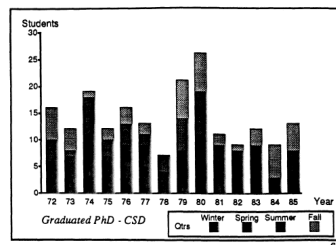


Figura 3.9: Gráfica producida por el sistema APT a partir de las variables *Students*, *Year* y *Qtrs*. La variable *Year* se divide entre los 4 valores de la variable *Qtrs* y la suma del número de estudiantes en los 4 cuatrimestres guarda una relación unívoca con la variable *Year*. Fuente: Mackinlay (1986, p.28).

CARACTERIZACIÓN DE ROTH Y MATTIS En el desarrollo del sistema SAGE, Roth y Mattis (1990) identifican tres tipos de relaciones adicionales para discriminar también entre tipos de gráficas:

- La “cobertura relacional” por la cual cada elemento de una variable guarda relación con al menos un elemento de otra variable. Distingue luego tres tipos diferentes de falta de cobertura: falta relación porque se trata de datos perdidos, la relación no tiene razón de ser, y finalmente, la falta de relación que no debiera darse por lo que es indicio de una anomalía. Para ayudar a entender estas relaciones, Roth and Mattis ponen como ejemplo la relación entre las variables ‘coste’ y ‘actividad’ representadas en el diagrama de barras de la figura 3.10 y en la que se representan inapropiadamente los valores perdidos porque se confunden con valores próximos a cero. En cambio,

el diagrama de puntos de la figura 3.11 sí que muestra apropiadamente los valores perdidos de la relación entre las variables ‘servidor’ y ‘actividad que lleva a cabo’ dado que permite añadir el valor ‘None’ entre los diferentes servidores para identificar aquellos con valores perdidos.

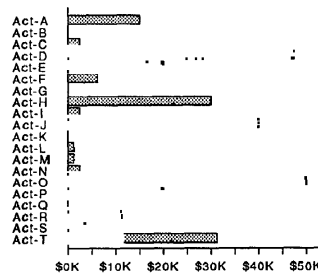


Figura 3.10: Diagrama de barras producido por el sistema SAGE que resulta inapropiado para representar variables sin cobertura relacional, dado que ésta puede ser confundida con valores próximos a 0. Fuente: Roth y Mattis (1990, p.195).

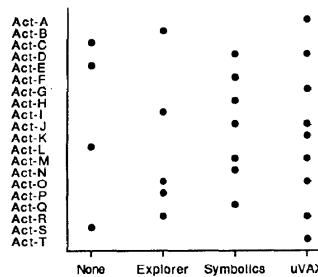


Figura 3.11: Diagrama de puntos apropiado para representar variables sin cobertura relacional. Fuente: Roth y Mattis (1990, p.196).

- La “cardinalidad” que expresa el número de elementos de una variable que pueden estar relacionados con un elemento de otra variable. Establece tres niveles de cardinalidad si es un único elemento, múltiples elementos en un número predeterminado, o

un número variable de elementos. Ejemplos de representaciones gráficas que expresan efectivamente una cardinalidad variable son los diagramas jerárquicos como el de la figura 3.12, las listas sangradas o los diagramas de puntos, mientras que los diagramas de barras son ineficaces y los mapas de calor requieren cardinalidad de valor único.

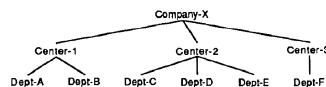


Figura 3.12: Diagrama jerárquico apropiado para representar diferentes relaciones de cardinalidad. Estas relaciones no se mostrarían adecuadamente, por ejemplo, en un diagrama de barras. Fuente: Roth y Mattis (1990, p.196).

- La “unicidad” es la característica por la que los valores de dos variables mantienen una relación jerárquica o carecen de esta estructura. Para el primer caso ponen como ejemplo la relación entre las variables **centro** y **departamentos** en el que un diagrama jerárquico de la figura 3.12 o una lista sangrada puede resultar eficaz y, para el segundo, la relación entre las variables **fecha de inicio** y **actividad** que puede representarse mejor mediante un diagrama de puntos dado que dos actividades pueden tener una misma fecha de inicio.

Además de estos tres tipos de relaciones, Roth y Mattis (1990) describen relaciones entre relaciones:

- los “tipos de datos complejos” que relacionan una variable con un vector como por ejemplo la relación entre un hotel y las coordenadas geográficas donde se encuentra o con una tupla entre **empleado** y **periodo de contrato**.

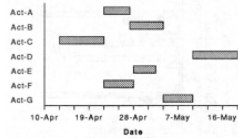


Figura 3.13: Diagrama de intervalos producido por el sistema SAGE que representa datos complejos en los que una variable actividad se relaciona con la dupla de variables fecha inicial y fecha final. Fuente: Roth y Mattis (1990).

- las *dependencias algebraicas* entre variables que pueden ser agragadas y transformadas en una nueva variable, relación que debería evidenciar su representación gráfica. Como ejemplo se describe la relación entre las variables ‘actividad’, ‘coste del material’, ‘coste del personal’ y ‘coste total’ que puede representarse mediante un diagrama de barras apiladas (ver figura 3.14).

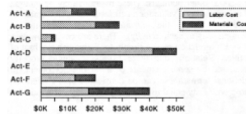


Figura 3.14: Gráfica producida por el sistema SAGE que representa la dependencia algebraica en la que el coste total se compone del coste en personal más el coste de los materiales. Fuente: Roth y Mattis (1990, p.193).

La estructura de los datos

La mayor parte de los conjuntos de datos en estadística son rectangulares, que significa que contienen el mismo número de observaciones, representadas por las filas, para cada una de las variables, representadas en las columnas (Wickham, 2014). Unos mismos datos pueden, sin embargo, adoptar diferentes formas con el objetivo de facilitar diferentes tareas, por ejemplo, facilitar el registro de los datos por

un encuestador, facilitar el intercambio de datos entre computadoras mediante una API (acrónimo de *Application Programming Interface*) o facilitar el procesamiento estadístico de esos datos. En los conjuntos de datos en estadística adquiere especial relevancia la distinción entre la estructura en forma de tabla de casos, tabla de frecuencias y tabla de contingencia ya comentados en la sección 3.1.

Otra distinción puede hacerse entre los conjuntos de datos en formato ancho (por ejemplo la tabla de contingencia), en el que existen encabezados de columna que corresponden a diferentes valores de una variable, o en formato largo en el que las observaciones de cada variable se estructuran en una columna.

Del mismo modo como un sistema puede requerir los datos con una determinada estructura para realizar un análisis estadístico, un sistema puede obtener información acerca de la estructura de los datos para presentar una gráfica u otra. Hay estructuras de datos para un número y tipo específico de variables, como por ejemplo, las matrices de adyacencia, que son matrices cuadradas que se utilizan para representar relaciones binarias entre un conjunto de objetos y que pueden ser representadas mediante grafos (ver figura 3.15). Otro ejemplo son las tablas de contingencia que pueden ser representadas mediante gráficas de mosaico.

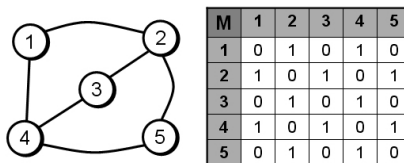


Figura 3.15: Matriz de adyacencia y grafo que la representa. Fuente: commons.wikimedia.org. Consultada el 28 de agosto de 2022.

Los lenguajes de programación orientada a objetos como **R** tienen asociados los términos “clases” y “métodos”. Las clases son tipos de objetos que cuentan con una estructura determinada y los métodos son funciones cuya definición cambia según la clase de objeto sobre

la que aplican. Como ejemplo de lo anterior podemos comprobar que la misma función `plot()` produce un diagrama de dispersión si se utiliza con el objeto `cars` de clase `data.frame` con dos variables numéricas, pero si utilizamos esta misma función con el conjunto de datos `airmiles` de clase `ts`, entonces la gráfica presentada será de línea y si la función `plot()` se aplica al objeto `Titanic` de clase `table`, entonces produce un diagrama de mosaico. Otro ejemplo es la librería `ggplot2`, también de **R**, que supone que los datos están estructurados en formato largo o *tidy data* (Wickham, 2014).

La procedencia de los datos

Entre las características de los datos que resultan útiles en el análisis de éstos, Schulz et al. (2016) incluye la procedencia y la utilidad de los datos. Aspectos como las limitaciones de los aparatos de medida o las circunstancias en las que los datos fueron adquiridos tiene relevancia en el momento evaluar la confiabilidad de los datos y modular las conclusiones derivadas de su análisis, pero pueden también determinar la selección de una gráfica si, por ejemplo, se describe un método de adquisición en el que el orden en el que se suceden las observaciones pudiera tener implicaciones en el análisis a pesar de que ninguna variable recogiera explícitamente este orden.

Otro aspecto de la procedencia de los datos que puede determinar la selección de una u otra gráfica es el carácter teórico o empírico de los datos. Es común, por ejemplo, utilizar geometrías diferentes para diferenciar los datos teóricos, de los empíricos y también para diferenciar los datos empíricos y los modelos a partir de éstos. Otra particularidad de las gráficas estadísticas es que a menudo representan el residuo entre los valores empíricos y los teóricos respecto a los que se comparan.

3.2. LOS USUARIOS RECEPTORES

Son varias las características de los receptores que pueden ayudar en la selección de una u otra gráfica.

Características de la percepción humana

La percepción humana dista mucho de mantenerse neutral respecto a las imágenes que procesa. La evolución ha especializado la visión de manera que ésta busca distinguir objetos, seguir sus trayectorias o identificar amenazas.

Los humanos podemos ver porque la luz se proyecta sobre la retina después de atravesar la córnea, el cristalino y el humor vítreo (ver figura 3.16). En la retina se encuentran precisamente los fotorreceptores que son las células capaces de transformar la luz en impulsos eléctricos. Los fotorreceptores, sin embargo, no se encuentran distribuidos uniformemente en la retina, sino que se concentran en la región opuesta al cristalino, conocida como mácula y más intensamente en la fovea. También encontramos una región alrededor del nervio óptico que carece de fotorreceptores, lo que provoca un punto ciego que se resuelve gracias a la visión binocular. A mayor concentración de fotorreceptores, mayor detalle en la visión, mayor contraste y mejor apreciación de los colores.

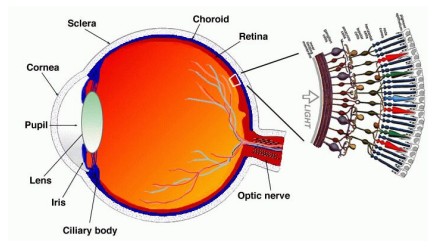


Figura 3.16: Anatomía macroscópica del ojo en la que puede verse en detalle las diferentes capas de la retina, en particular los bastones (*rods*) y los conos (*cones*). Fuente: webvision.med.utah.edu. Consultada el 28 de agosto de 2022.

Los fotorreceptores tampoco tienen todas las mismas características. En la capa externa de la retina se encuentran dos tipos. Por un lado encontramos los bastones de los que hay unos 120 millones, son monocromáticos y se especializan en la visión nocturna. Por otro lado encontramos los conos que se concentran en la fovea y de los que hay solamente unos 8 millones. Los conos se subdividen a su vez en tres tipos en función de la longitud de onda alrededor de la cual transforman la luz en impulsos (ver figura 3.17):

- *S-short*. Pico de sensibilidad: 440 nm. (en realidad violeta aunque es identificado como ‘azul’).
- *M-medium*. Pico de sensibilidad: 550 nm. (entre verde y amarillo aunque identificado como ‘verde’)
- *L-long*. Pico de sensibilidad: 570 nm. (amarillo aunque identificado como ‘rojo’)

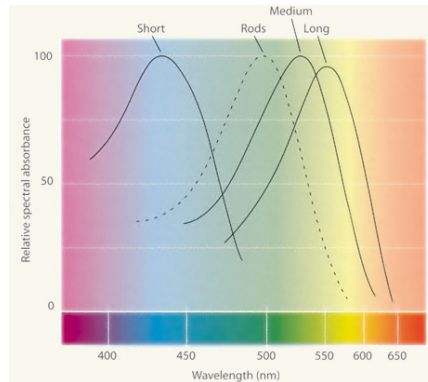


Figura 3.17: Distribución relativa de las longitudes de onda del espectro visible de la luz, que los diferentes tipos de conos y los bastones transforman en impulsos eléctricos. Fuente: webexhibits.org. Consultada el 28 de agosto de 2022.

La construcción del color en base a los 4 tipos de células fotosensibles ha dado lugar a diferentes modelos y escalas de color para facilitar diferentes propósitos como por ejemplo, maximizar el número

de colores distinguibles entre dos tonos (Levkowitz y Herman, 1987), igualar la distancia percibida entre los diferentes colores de la escala (Levkowitz, 1997) o mejorar la correcta interpretación por personas con deficiencias en la interpretación del color (Nuñez et al., 2018).

Más allá de la conversión de la luz en impulsos eléctricos y de éstos en un color percibido, está el lugar y la forma en la que se reflejan la luz que emiten los objetos en la retina y su procesamiento por el cerebro. Costa (1998) enumera 20 principios y leyes gestálticas² entre las que encontramos, por ejemplo, la ley dialéctica por la que toda forma se desprende del fondo sobre la que está establecida, la ley de completación por la que si un contorno no está completamente cerrado, la mente tiende a completar o continuar dicho contorno, o el principio de similitud, por el cual, en un campo de elementos equidistantes aquellos que tienen mayor similitud se perciben ligados entre ellos. En illusionsindex.org se pueden consultar multitud de ejemplos de ilusiones ópticas basadas en las leyes gestálticas.

La figura 3.18 muestra el Triángulo de Kanizsa, una ilusión óptica que hace uso de la ley dialéctica, la de completitud y del principio de similitud). Otro ejemplo es la ley de cierre por la que una forma será mejor en la medida en que su contorno esté mejor cerrado, o la ley de simplicidad por el que las figuras menos complejas tienen una mayor pregnancia.

Costa (1998) enumera además otras leyes de *infra-lógica visual* por las que se establecen mecanismos de la visión que se sitúan en el umbral inferior a los de la percepción gestáltica. Entre estas leyes se encuentran, por ejemplo, la ley de centralidad por la cual los elementos que se presentan en el centro son más importantes, o mejores, que los de la periferia. Otro ejemplo es la ley de la correlación por la cual la correlación es siempre una presunción de causalidad.

²Las leyes gestálticas son una lista exhaustiva de sesgos de la percepción visual que fueron recopilados por miembros de la corriente psicológica conocida como Gestalt.

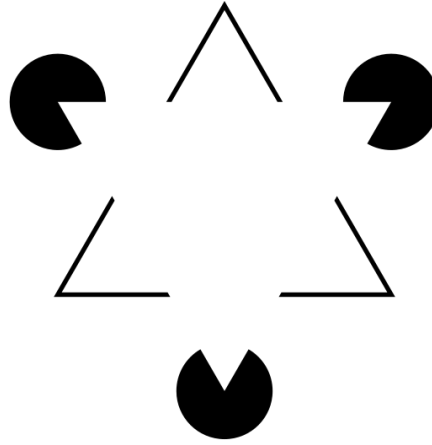


Figura 3.18: Ilusión óptica Triángulo de Kanizsa. Esta ilusión óptica hace emerger un triángulo en el centro de la figura con un vértice en la parte inferior. Fuente: commons.wikimedia.org. Consultada el 28 de agosto de 2022.

Las características de la percepción humana pueden determinar la elección de una gráfica u otra. Un ejemplo lo tenemos en las series temporales que, para unos mismos datos, unas veces se pueden representar mediante gráficas de puntos, otras veces mediante gráficas de líneas y otras mediante gráficas de puntos y líneas. Otro ejemplo lo tenemos en la elección entre diferentes paletas de color, o entre diferentes formas para identificar con menor esfuerzo los valores de una variable. También, en función del número de valores a diferenciar, un sistema puede elegir una u otra variable visual. En este sentido, el paquete `ggplot2` de **R** por ejemplo, asigna diferentes paletas de color en función de si éstas representan variables categóricas o numéricas.

Tarea a realizar por el usuario

La tarea a realizar por el usuario es, después de las características de los datos, el elemento más determinante para acertar en la presentación de una gráfica adecuada a un usuario. A continuación hacemos un recorrido cronológico de diferentes puntos de vista sobre

cómo incorporar la tarea a realizar por el usuario en la selección de gráficas estadísticas.

La tarea está intrínsecamente relacionada con la pregunta que un usuario espera responder mediante la observación de una gráfica. Según Bertin (1967, p. 141), una misma gráfica puede responder a tres niveles de preguntas. Por un lado hay un nivel elemental que permite responder a preguntas sobre valores puntuales codificados en una gráfica. Éstas, en una gráfica de evolución quinquenal del tipo de cambio €-US\$, serían preguntas del tipo ¿Cuál era el tipo de cambio €-US\$ tal día? El nivel intermedio de lectura responde a preguntas que se formulan en relación a un subconjunto de valores representados en la gráfica, por ejemplo, ¿Qué caracteriza la tendencia del cambio €-US\$ durante el primer año? Finalmente, el nivel superior de lectura responde a preguntas que requieren la observación de todas las correspondencias representadas en la gráfica, como por ejemplo, ¿Qué caracteriza la evolución del tipo de cambio durante el periodo representado en la gráfica?

La utilidad para la que se recogen los datos, aunque también se recogen datos sin una utilidad específica, determina, por ejemplo, las transformaciones estadísticas que se prevé aplicar a los datos y consecuentemente, las gráficas asociadas a los diferentes métodos estadísticos.

Por otro lado, Bertin también identifica diferentes propósitos de la gráfica como registrar, tratar o comunicar la información (ya presentados en la sección 2.2) que, para una misma pregunta, pueden llevar a diferentes soluciones y los diferentes registros de comunicación pueden, a su vez y también para una misma pregunta, sugerir gráficas con diferentes características.

Los sistemas de recomendación de gráficas estadísticas han ido incorporando las tareas a realizar por los usuarios desde diferentes perspectivas. El sistema BHARAT (Gnanamgari, 1981) es uno de los primeros sistemas, si no el primero, que incorpora aspectos de la

tarea a realizar. Este sistema solicita al usuario cuál es su “objetivo”, como por ejemplo si se pretende hacer un análisis de la tendencia o si se pretende hacer una comparación en términos absolutos (ver figura 3.19). La respuesta a estas preguntas, junto con las características de las variables, permiten seleccionar un tipo u otro de gráfica.

```

Main Menu:
0. Terminate the session
1. Request for initial display
2. Change the objectives
3. Display
4. Table Manipulation
5. Change the format
6. Change the attributes
7. Save the display
8. Help

Please enter your selection number: 2

Is this data for TREND analysis (Y/N) ? N

Do you prefer ABSOLUTE comparison (Y/N) ? Y

Continue ?

```

Figura 3.19: Interficie de usuario del sistema BHARAT que requiere, por primera vez en un sistema de automatización de gráficas, que el usuario explicita la intención de éste con la gráfica a producir. Fuente: Gnanamgari (1981).

En el campo experimental, Cleveland y McGill (1984) identifican un conjunto de tareas perceptivas elementales que llevan a cabo los usuarios para extraer información cuantitativa a partir de las gráficas y ordenan estas tareas en función de la precisión con la que los usuarios extraen la información. Las tareas perceptivas elementales, como por ejemplo, comparar la posición de dos puntos en una escala común, comparar dos ángulos o comparar la longitud de dos segmentos de recta, que ya se han recogido en la sección 2.3.

El sistema SAGE (Roth y Mattis, 1990) incluye, además de una caracterización de las variables para la selección de gráficas, una selección de tareas perceptuales útiles para presentar una gráfica pretendidamente óptima al usuario. Entre las tareas que identifican se encuentran las siguientes:

- Valor de búsqueda preciso. Esta tarea favorece la selección de tablas de texto en vez de gráficas estadísticas porque la tabla puede incorporar el valor preciso con menos elementos accesorios que la gráfica.
- Comparación de valores entre dos variables, pero no más. Esta tarea favorece la desagregación de las variables en gráficas multipanel, cada uno dedicado a representar las relaciones entre dos variables.
- Comparación de relaciones entre tres o más variables. Esta tarea favorece la superposición de variables comparables en un mismo panel.
- Distribuciones de valores. Favorece la selección de gráficas de recuentos de frecuencias.
- Correlaciones funcionales entre atributos. Se refiere correlaciones entre variables cuantitativas que favorece la selección de gráficas con marcas referidas a cada registro en particular en vez de las gráficas que muestran recuentos.
- Indexación respecto a una o más variables. Esta tarea facilita establecer un orden entre las categorías observadas, por ejemplo, el orden alfabético, según la frecuencia de observaciones o el orden de aparición.

El sistema BOZ (Casner, 1991), como secuela de los estudios de Cleveland y McGill (1984) presenta un nuevo planteamiento que el autor mismo resume de la siguiente manera: “Dado que la utilidad de una representación gráfica es una función de la tarea a la que la gráfica ha de dar soporte, el diseño gráfico debe centrarse en diseñar procedimientos que sean eficientes para llevar a cabo por la percepción humana. Las decisiones que se tomen sobre cómo codificar y estructurar la información en una gráfica deben basarse principalmente en respaldar un desempeño eficiente y preciso del procedimiento de percepción”. Siguiendo este planteamiento, el sistema BOZ imple-

menta un catálogo de operadores perceptuales que intervienen en la elección de la gráfica a presentar, como por ejemplo: “determinar la distancia horizontal”, “determinar la posición horizontal” o “buscar objeto según grado de sombra” (ver figura 3.20).

Horizontal Position	Shading
determine-horz-pos	determine-shade
search-object-at-horz-pos	search-object-with-shade
search-any-horz-pos-object	search-object-and-shade
verify-object-at-horz-pos	verify-object-and-shade
horz-coincidence?	darker?
left-of?	lighter?
right-of?	same-shade?
horz-forward-projection	
horz-backward-projection	
determine-horz-distance	

Figura 3.20: Catálogo de los diferentes operadores perceptuales utilizados por el sistema BOZ para seleccionar una gráfica adecuada a partir de la posición horizontal y el sombreado. Fuente: Casner (1991).

Como ejemplo de los tipos de gráficas que produce el sistema BOZ a partir de las características de las variables, las VV y los operadores perceptuales mostramos en la figura 3.21 una gráfica que muestra el origen y destino de una serie de vuelos, el horario de salida y la duración prevista, la disponibilidad de asientos y el precio. Esta gráfica facilita la identificación de los próximos vuelos con asientos disponibles por un operador de ventas. Los operadores perceptuales a utilizar serían, por ejemplo, “buscar objeto según grado de sombra” o “determinar la posición horizontal” entre otros.

Shneiderman (1996) presenta una taxonomía basada en los 7 tipos de datos que identifica: datos en una, dos o tres dimensiones, datos temporales, multidimensionales, en árbol o en red. En función de estos tipos de datos, la taxonomía incluye después una clasificación de las acciones a llevar a cabo. Estas acciones se resumen en obtener una visión general y luego más específica, excluir datos no relevantes, obtener detalles de los datos relevantes, relacionar los datos relevantes con otros elementos, guardar un registro de las acciones llevadas a cabo que permita corregirlas o iterarlas, y extraer información. Estas

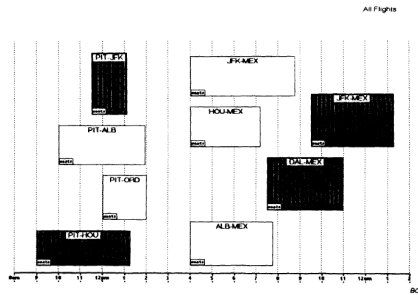


Figura 3.21: Diagrama de bloques producido por el sistema BOZ que presenta objetos que heredan cuatro propiedades gráficas: posición horizontal que codifica el horario previsto de salida, posición vertical que evita superposiciones, sombreado que codifica la disponibilidad de asientos, altura que codifica el precio y etiquetas que codifican el origen y el destino. Fuente: Casner (1991)

acciones no se ciñen a los sistemas que producen gráficas estáticas sino que es válido especialmente para sistemas que incorporan técnicas dinámicas e interactivas para el análisis de datos.

La definición de las acciones o tareas a realizar por el usuario, sin embargo, no tienen porqué definirse a tan bajo nivel sinó que las gráficas pueden también determinarse en función de, por ejemplo, un determinado análisis estadístico para el se suele utilizar una o unas gráficas específicas.

El recuerdo de selecciones previas

Existen dos aproximaciones en la recomendación basada en preferencias previas. Una se conoce como “filtrado basado en el contenido” y recomienda un producto a un usuario a partir de las elecciones anteriores o preferencias de este mismo usuario. La otra estrategia se conoce como “filtrado colaborativo” y promueve sugerencias a partir de las elecciones de otros usuarios con los que el sistema asocia el usuario objeto de la recomendación.

Un primer e incipiente ejemplo de filtrado basado en el contenido lo encontramos en SageBook (Roth et al., 1994) que presenta la

primera herramienta informática que permite guardar modelos de gráficas en una biblioteca que complementa el sistema SAGE (ver figura 3.22). Esta biblioteca se utiliza luego para buscar gráficas que anteriormente el usuario ha recogido. La búsqueda que puede realizarse desde dos perspectivas: filtrar gráficas a partir de modelos de representación (por ejemplo gráficas de barras, de líneas, redes, mapas, etc.), o bien filtrar gráficas producidas a partir de unos datos con las mismas características de los que tenemos entre manos. Otros ejemplos más recientes los tenemos en DBVR (Gotz y Wen, 2009) es Dziban (Lin et al., 2020).

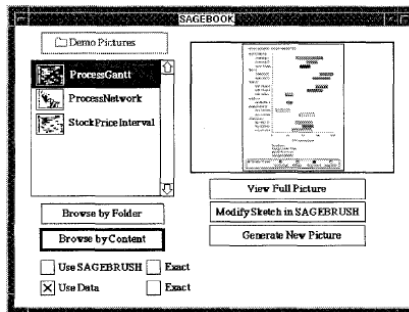


Figura 3.22: Explorador SageBook de modelos de representación que se combina con el editor de gráficas SageBrush, los dos incorporados en el sistema SAGE para la presentación automática de gráficas. Fuente: Roth y Mattis (1990).

Un ejemplo de filtrado colaborativo lo tenemos en VizRec (Mutlu et al., 2015) que utiliza las valoraciones previas de otros usuarios que han puntuado según 9 criterios como {saturado, organizado, confuso, fácil de entender, aburrido, divertido, útil, efectivo, satisfactorio}. Otro ejemplo lo tenemos también en VizML (Hu et al., 2019) que utiliza redes neuronales para sugerir gráficas a partir de gráficas realizadas anteriormente por otros usuarios en la plataforma Chart Studio Community Feed.

Convenciones sociales

Si se considera no únicamente las preferencias de un grupo reducido de usuarios sino las convenciones sociales de un conjunto amplio de usuarios, se pueden establecer reglas heurísticas como por ejemplo la de utilizar el eje de las abscisas para representar la variable tiempo, etiquetar los ejes de los paneles en sus lados derecho o izquierdo, o superior e inferior, ordenar los valores de las leyendas en orden creciente o decreciente, utilizar una paleta de colores u otra, etc.

3.3. EL CANAL

Las limitaciones del canal pueden clasificarse en, al menos, tres tipos según si estas limitaciones son de transmisión de datos, si se trata de limitaciones de procesamiento o bien si se trata de limitaciones en el tamaño o resolución de la pantalla (o soporte sobre el que se proyecta la gráfica). Estas limitaciones tienen más implicaciones en la estética de la gráfica o en la usabilidad que en la selección de uno u otro tipo de gráfica.

3.4. LA GRÁFICA

Si el usuario es inexperto o bien le interesa recibir sugerencias, entonces se suele encontrar con “el problema gráfico” (comentado en el capítulo 1), que consiste en qué gráfica elegir de entre las numerosas representaciones gráficas posibles.

Otra posibilidad es que un usuario, más o menos experto, conozca exactamente el tipo de gráfica con el que pretende visualizar un conjunto de datos o conozca aspectos concretos de la gráfica que pretende obtener. En este caso los sistemas que permiten producir una gráfica concreta o definir explícitamente aspectos de la gráfica a buscada utilizan la estrategia basada en modelos de representación.

Los elementos de la gráfica

Seleccionar una gráfica u otra, implica elegir la composición y características de las marcas dependientes de los datos dentro del área gráfica. Una vez seleccionado el tipo de gráfica, para materializarla, se requiere asignar unos datos específicos para que así, el sistema, pueda transformarlos en una imagen. Así, los elementos que subyacen en una representación gráfica son el conjunto de datos del que bebe, las variables que intervienen, las transformaciones estadísticas a las que se someten las variables, las relaciones algebraicas entre las variables y los metadatos.

Las marcas dependientes de los datos dentro del área gráfica, pueden ser un reflejo de los datos en bruto o pueden ser datos procesados que han sufrido transformaciones estadísticas. En cualquier caso, estas marcas pueden variar según el tipo de implantación (como punto, línea, área o símbolo compuesto), las VV que intervienen (entre las que cabe diferenciar las variables espaciales y las variables de retina), el sistema de coordenadas, las transformaciones de las escalas, la posible descomposición de paneles (mediante operaciones de traslación o simetría) y los ajustes de posición de las marcas (por ejemplo barras apiladas o intercaladas).

Aproximación de Bertin

Bertin (1967, Part Two) descompone el *problema gráfico* en diferentes subproblemas con los que acotar el abanico de gráficas a escoger. Algunos de estos subproblemas afectan las características de la gráfica:

- El grupo de imposición, que descompone en cuatro niveles, a partir de la naturaleza de las correspondencias manifestadas en el plano: diagramas, redes, mapas y símbolos.

- El número de *VV* necesarias, que descompone según sean necesarias 1, 2, 3 o más *VV*.
- El tipo de implantación entre los que distingue el punto, la línea o el área.

Bertin presenta un catálogo de soluciones para diferentes combinaciones de estas tres características en el que, además, añade otras característica para acotar el tipo de gráfica como son el *tipo de imposición* (o sistema de coordenadas), las *VV* y la diferente asignación de las variables en los datos a los diferentes ejes en un mismo sistema de coordenadas.

Aproximación de Roth

El editor SageBrush, que complementa el sistema de presentación automática de gráficas SAGE, permite a los usuarios hacer un esbozo o conjugar elementos gráficos para crear prototipos sobre los que vincular diferentes conjuntos de datos. La creación de una nueva gráfica se realiza mediante la combinación de diversos elementos: la organización espacial, grafemas y codificadores (escalas con las que interpretar las propiedades de los grafemas). Diferentes combinaciones de estos elementos que resultan especialmente frecuentes, y que guardan relación con los grupos de imposición de Bertin, se presentan como prototipos que el usuario puede luego editar. SageBrush considera los siguientes prototipos y grafemas (ver figura 3.23) que recuerdan a los tipos de marcas utilizados posteriormente por (Mackinlay et al., 2007):

- Prototipos: Diagrama, red, mapa, tabla, matriz, lista no ordenada y lista jerarquizada.
- Grafemas: Marca puntual, barra, línea, gálibo y texto.

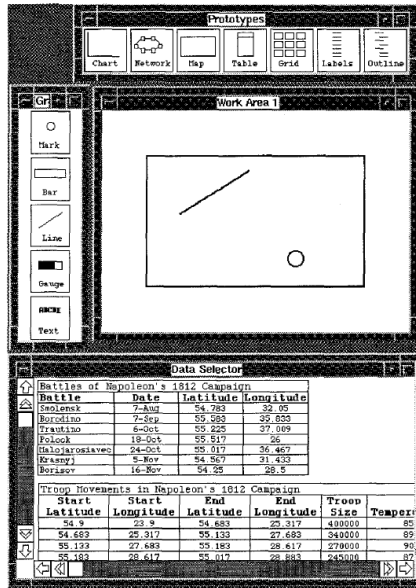


Figura 3.23: Editor SageBrush que facilita la creación de gráficas a partir de los elementos de ésta que se vinculan a variables en los datos. Fuente: Roth et al. (1994).

Aproximación de Wilkinson

Wilkinson (1999) descompone los elementos de la gráfica y presenta una gramática con la que se organizan y combinan estos elementos. Traslada al espacio digital, como antes había hecho Mackinlay (1986), algunos preceptos que Bertin (1967) había estipulado para las gráficas sobre papel, pero esta vez, el esfuerzo no lo dirige hacia la selección de una gráfica óptima sino hacia la definición de un lenguaje con el que poder construir cualquier gráfica. Una implementación de esta gramática la presenta el mismo Wilkinson (2005) en la segunda edición del mismo título, concretamente con el lenguaje GPL (acrónimo de *Graphics Production Language*).

Los principales elementos de la gráfica que Wilkinson identifica en su gramática son los siguientes:

- Los datos son representaciones simbólicas de observaciones o pensamientos acerca del mundo.
- Las variables otorgan métodos para asociar conceptos con observaciones en los datos.
- El álgebra se refiere a la restauración y reunión de conjuntos de variables para formar conjuntos de datos de los que se nutrirán las gráficas.
- Las escalas son las funciones que utilizamos para transformar variables representadas en las gráficas (*varsets*) en dimensiones.
- La estadística referida a las transformaciones de los datos que, en este caso, Wilkinson sitúa bajo el control del procedimiento para crear gráficas.
- Geometría entendida como las funciones geométricas producidas por el objeto **Grapher** para crear grafos que pueden ser representados por magnitudes en un espacio.
- Las coordenadas que son esquemas sobre los que proyectar elementos de conjuntos de datos sobre objetos geométricos.
- La estética que transforma grafos en gráficas de modo que son percibibles sin ser percepciones.
- Las facetas con las que se obtienen gráficas multipanel condicionadas por una o más variables.
- Las guías que determinan aspectos de anotación en los ejes, marcas y anotaciones auxiliares.

Como ejemplo más recientes, se puede incluir en este apartado la librería de creación de gráficas en dos dimensiones `ggplot2` (Wickham, 2009) en el caso del lenguaje de programación R que implementa, con modificaciones, la gramática presentada por Wilkinson (1999). Esta librería presenta al usuario, en principio, la gráfica que el mismo usuario determina, y a falta de determinación, propone una gráfica o a partir de elementos acotados por éste y otros asumidos por defecto.

3.5. CONCLUSIONES

De los diferentes caminos que conducen a una gráfica estadística adecuada, la tarea a realizar por el usuario, el número de variables a relacionar y las características de estas variables por separado son los factores clave.

Entre las características intravariante hemos identificado las escalas de medición (aunque diferentes autores consideran diferentes clasificaciones de estas escalas (Bertin, 1967; Gnanamgari, 1981; Mackinlay, 1986; Bachi, 1968; Roth y Mattis, 1990; Mackinlay et al., 2007)), el carácter cíclico del espacio muestral del que toman valores (Bachi, 1968), la longitud de las variables (Bertin, 1967; Roth y Mattis, 1990), la secuencia entre los valores (Gnanamgari, 1981) y el tipo de objeto elegido para almacenar la variable (Kamps, 1999; Mackinlay et al., 2007). Otros aspectos que han aparecido es la consideración de la variable como dimensión o medida (o como predictora o de respuesta) y la procedencia o utilidad de los datos.

La utilidad de los datos guarda relación con la tarea a realizar por el usuario y adquiere especial importancia en entornos de programación como **R** en el momento de sugerir una u otra gráfica. **R** permite definir nuevas clases de objetos y, sobre estos nuevos objetos, definir métodos que permiten producir gráficas específicas si se utiliza la función genérica `plot()` sobre estos objetos. Por ejemplo, la función `plot(BOD)` presente un diagrama de punto, `plot(Nile)` presente un diagrama de línea y `plot(Titanic)` un diagrama de mosaico siendo el primer objeto un `data.frame`, el segundo de clase `ts` y el tercero de clase `table`.

Las características de la percepción humana y las convenciones sociales ayudan en la elección de las VV y en la asignación de variables a los ejes de coordenadas. Estos dos caminos no sirven para elegir entre un gran abanico de gráficas posibles pero sí que pueden complementar cualquier otra estrategia.

Finalmente, en tanto que la alfabetización gráfica va en aumento, la elección de una gráfica estadística a partir de las características de ésta, es probable que cada vez sea más utilizada.

CAPÍTULO 4

CARACTERIZACIÓN DE LOS DATOS

“Graphic thinking is primary, and design is the vehicle which carries the graphic thought to its destination as a graphic statement.” — William J. Bowman

El presente capítulo incluye el artículo de Millán-Martínez y Valero-Mora (2018) publicado en la revista *Information Visualization* en 2018 que presenta un marco teórico con el que clasificar, separadamente, las variables en un conjunto de datos, así como las gráficas estadísticas según la manera como representan las propiedades de las variables por separado.

Como ya hemos comentado anteriormente en el capítulo 3, este trabajo presenta un sistema de recomendación de gráficas estadísticas que interpreta qué gráficas presentar a diferencia de otros sistemas que interpretan qué datos presentar. Entre las diferentes estrategias que pueden conducir hacia un tipo de gráfica adecuado, la estrategia funcional se refiere a los métodos que determinan el tipo de gráfica a partir de las características de los datos, y entre estos métodos, encontramos los que utilizan el número de variables de entrada a combinar para construir la gráfica, los que utilizan las características intravariante, las características intervariable, la estructura de los datos, su procedencia, la utilidad o los metadatos.

Se ha escogido la estrategia funcional porque ésta es una técnica que, en mayor o menor medida, utilizan todos los sistemas de recomendación de datos analizados (Millán-Martínez y Valero-Mora, 2018, tabla 1). La información acerca de los datos suele ser luego complementada con otra información como la tarea a realizar, las características de la percepción humana, las convenciones sociales, el

modelo de representación buscado, las preferencias del usuario o las limitaciones del hardware.

Entre las características de los datos, el método propuesto descarta utilizar las características intervariable por varios motivos. En primer lugar porque el número de relaciones entre pares de variables aumenta exponencialmente a medida que se añaden variables a partir de las cuales construir la gráfica. En segundo lugar porque una cosa son las relaciones observadas y otra las relaciones posibles que no tienen porqué reconocerse *a priori* y sería complicado para un usuario inexperto caracterizar cada relación. En tercer lugar porque son precisamente estas relaciones lo que la gráfica permite deducir fácilmente. También descarta determinar la gráfica a partir de la procedencia de los datos porque se aboga precisamente por ofrecer alternativas que resultan de utilidad independientemente del campo de conocimiento del que se nutren los datos, se descarta también utilizar los metadatos porque requeriría una estructura estandarizada de la que generalmente carecen los conjuntos de datos. Una alternativa también interesante hubiera sido un recomendador de gráficas a partir de la utilidad que se pretende de los datos pero, en estadística, la utilidad de los datos está muy ligada a métodos estadísticos específicos para los que se han desarrollado métodos gráficos también específicos de los que generalmente hacen uso usuarios expertos.

El método propuesto se fundamenta en el número de variables de entrada, las características intravariante y en una característica de la estructura de los datos, concretamente, el orden que secuencia las observaciones. Otras características de la estructura de los datos no se tienen en consideración porque se presume que los datos se encuentran estructurados como *tidy data* (Wickham, 2014) en forma de tabla rectangular en la que cada variable está representada en una columna, cada observación está representada por una fila y cada tipo de unidad de observación está representada en una tabla. La selección de las variables de entrada equivale consecuentemente a seleccionar las

columnas del conjunto de datos. A continuación se detallan diferentes dimensiones que permiten caracterizar separadamente las relaciones intravariante de un conjunto de datos y se describe de qué manera la caracterización como uno y otro nivel de estas dimensiones puede conducir a una gráfica con características diferentes en cada caso.

4.1. ESCALA GRÁFICA DE MEDIDA (M)

Una primera dimensión con la que se puede caracterizar una columna de valores de una variable es la escala de medición. A diferencia de las escalas de medición propuestas por Stevens (1946) que clasifica las escalas en función de las transformaciones estadísticas admisibles, en este caso podemos clasificar las escalas de medida en función de la relación que guardan entre sí las marcas de los diferentes valores observados y la relación entre estas marcas y otras que representan los extremos teóricos no necesariamente observados. A continuación se detallan los diferentes valores que esta dimensión puede tomar en función de las citadas relaciones.

Escala ordenable (Un)

Los valores ordenables son aquellos valores que no guardan una relación de orden obvia entre ellos de modo que no pueden clasificarse de menor a mayor. Un ejemplo de valores de este tipo lo podemos encontrar, por ejemplo, en una columna en la que se encuentra informado el número de dorsal de las jugadoras de un equipo de hockey, a pesar de llamarse “número” podría tratarse de un código alfanumérico.

Los valores de las variables ordenables pueden verse representados en el área gráfica, en las leyendas, en las escalas de un panel o en las escalas multipanel. Cualquier orden en el que se representan los valores de variables ordenables no obedece a una relación de orden obvia entre los valores sino a elementos ajenos como pueden ser la

secuencia de los valores observados, la frecuencia de observaciones, el orden alfabético, el valor promedio de alguna otra variable o cualquier otro criterio. De este modo, el orden en el que se pueden representar los valores ordenables representa una herramienta para acometer el propósito de la gráfica.

A parte de la oportunidad de ordenar los valores ordenables de una u otra manera en el momento de representar estos valores en la gráfica, la cualidad de ordenable puede aconsejar relacionar estos valores con una variable visual u otra, como por ejemplo la forma en vez del tamaño o el tono de color en vez de la luminosidad. También puede aconsejar, por ejemplo, la identificación de los valores mediante etiquetas que identifican puntos en el área gráfica antes que dar soporte a esos valores en un eje espacial en el que éstos se ven forzados a seguir algún tipo de orden.

Escala ordenada (Or)

El orden de los valores de la escala no debe confundirse con el orden en el que suceden las observaciones. Los valores ordenados son aquellos que guardan una relación de mayor a menor sin que tenga sentido una operación aritmética entre los valores de la escala. Ejemplos de valores ordenados los tenemos en las escalas de tipo Likert que presentan un abanico de categorías entre las que se hace escoger a un informador para medir aptitudes, opiniones o percepciones, con valores como por ejemplo “totalmente en desacuerdo”, “en desacuerdo”, “ni de acuerdo ni en desacuerdo”, “de acuerdo” y “totalmente de acuerdo”.

Los valores ordenados suelen determinar el orden en el que se sitúan los valores de la escala en un eje de coordenadas o en la leyenda. También puede aconsejar el uso de una variable visual con la que relacionar los valores en vez de otra, como por ejemplo la luminosidad del color en vez del tono de color o la forma.

Escala arbitrariamente referenciada ($0i$)

Cuando las observaciones se componen de valores escalares entre los que sí tiene sentido realizar operaciones aritméticas, entonces, una posibilidad es que la escala tenga un origen o valor cero arbitrario, es decir, que el valor cero en vez de representar ausencia de cantidad, represente simplemente una referencia que facilita la lectura de las observaciones. En este caso, una única observación de estas variables puede resultar insuficiente dado que el interés se encuentra en las distancias entre observaciones que se mantienen constantes cualquiera que sea el valor cero. En este caso la escala de medición recibe el nombre de escala intervalar. Un ejemplo de valores arbitrariamente referenciados puede ser la temperatura atmosférica medida en grados Celsius en la que el cero no representa ausencia de calor.

Las observaciones de una variable escalar arbitrariamente referenciada pueden aconsejar, por ejemplo, el uso de implantación en forma de punto o de línea en vez de área, su soporte en una variable espacial en vez de una de retina, o una escala de tonos de color divergentes en vez una escala de un mismo tono de color pero con diferente luminosidad.

Escala referenciada a un extremo ($1i$)

Si las observaciones se componen de valores escalares que se referencian respecto de un origen o cero no arbitrario, de modo que el origen de la escala denota ausencia de cantidad, entonces puede tener sentido representar cada observación como un intervalo entre el origen y el valor observado. En este caso la escala de medición recibe el nombre de escala de razón. Un ejemplo de valores referenciados a un extremo puede ser la altura de un conjunto de personas.

Las observaciones de valores de una variable escalar referenciados a un extremo pueden aconsejar la implantación en forma de línea o

área en vez de la de punto, dado que las distancias entre los puntos observados y el origen representan cantidades.

Escala referenciada a dos extremos (2i)

Otro caso lo encontramos cuando las observaciones de escalares se referencian respecto a dos extremos representando uno la ausencia de cantidad y el otro extremo la completitud. Un ejemplo de valores referenciados a dos extremo puede ser la probabilidad de éxito de un experimento que se encuentra acotada entre cero y uno.

Las observaciones de valores escalares referenciados a dos extremos pueden aconsejar también la implantación en forma de línea o área en vez de la de punto, líneas o áreas que pueden cubrir los intervalos entre uno u otro extremo.

La escala gráfica de medida no se encuentra implícita en los valores de las variables. Hay aspectos de la escala gráfica de medida que son más o menos fáciles de deducir, por ejemplo si una variable es numérica o categórica, pero hay otros aspectos que son más difíciles de deducir. Ejemplos de lo dificultad para caracterizar valores categóricos como ordenables u ordenados puede ser el conjunto de planetas del sistema solar que pueden ordenarse según la distancia entre éstos y el sol, según su tamaño, o cualquier otra característica sin que exista una relación de orden evidente entre ellos. Una cosa parecida ocurre con los valores de variables de intervalo, de razón y doblemente acotadas, como por ejemplo, las mediciones de profundidad de un lago de montaña que tienen un mínimo y máximo teóricos pero cuyas oscilaciones pueden ser tan ínfimas que conviertan en irrelevantes la relación de éstas con el máximo y el mínimo teóricos.

Recodificaciones entre escalas gráficas de medida

La caracterización de las variables según una escala gráfica de medida se puede recodificar siempre en el sentido de eliminar res-

tricciones, esto es, ganando grados de libertad. De este modo, los valores escalares acotados por dos extremos pueden ganar un grado de libertad si pasan a ser representados en relación solamente a uno de los extremos; pueden ganar dos grados de libertad si no se representan referenciados a ninguno de sus extremos; tres grados de libertad si los valores se representan, por ejemplo, en forma de lista ordenada de mayor a menor o viceversa; y finalmente, pueden ganar cuatro grados de libertad si los valores observados pasan a ser meramente categorías sin relación de orden entre ellas. Siguiendo esta secuencia, los valores escalares referenciados a un extremo podrían adquirir hasta tres grados de libertad, los valores escalares arbitrariamente referenciados dos y los valores ordenados uno. Por otro lado, los valores ordenables, si se soportan en una variable espacial, tienen que ser ordenados de uno u otro modo por lo que pueden añadir una restricción a cambio de facilitar un determinado propósito de la gráfica.

Un ejemplo de lo expuesto en el anterior párrafo puede ser la representación de una variable que recoge la mortalidad infantil como porcentaje de bebés nacidos vivos que mueren durante el primer año de vida. El porcentaje se encuentra acotado entre 0 y 1 por lo que una posibilidad podría ser listar los acontecimientos y acompañar cada uno de ellos con un diagrama de punto acotado entre el cero y el cien como el de la figura 4.1.

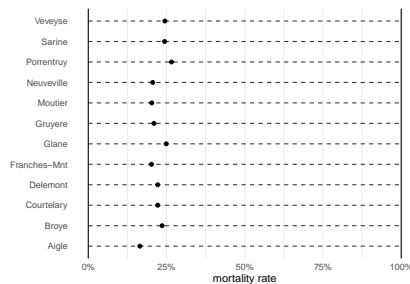


Figura 4.1: Escala referenciada a dos extremos. Fuente: Elaboración propia.

Pudiera suceder que los porcentajes de las diferentes observaciones fueran cercanos por lo que no se apreciara la diferencia entre la posición de algunos de los puntos, entonces, una opción podría ser cambiar el diagrama uniaxial de punto acotado entre dos extremos por un diagrama de punto acotado únicamente a un extremo como el de la figura 4.2.

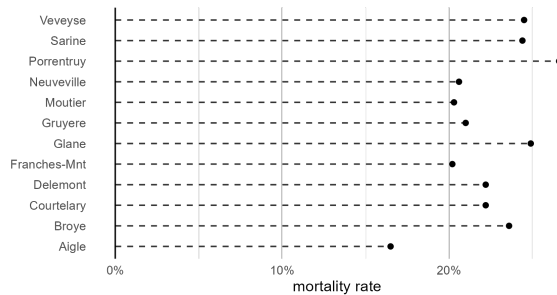


Figura 4.2: Escala referenciada a un extremo. Fuente: Elaboración propia.

Una vez observado el diagrama de puntos acotados a un extremo, pudiera pasar que se prefiriera facilitar la diferencia de lectura de las diferentes observaciones por lo que se modificara la escala gráfica a una de valores escalares acotados arbitrariamente, mediante un diagrama simple de punto como el de la figura 4.3.

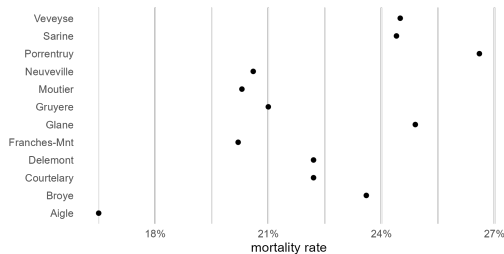


Figura 4.3: Escala arbitrariamente referenciada. Fuente: Elaboración propia.

Otra posibilidad es que se quisiera dar énfasis al orden que siguen las observaciones sin reparar en la diferencia aritmética entre éstas,

por lo que la escala gráfica se podría recodificar a una de valores ordenados como en la figura 4.4.

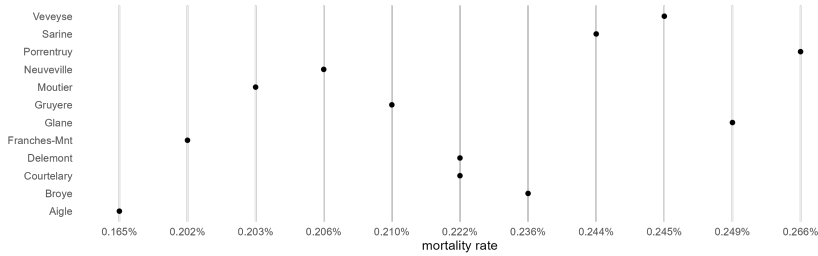


Figura 4.4: Escala ordenada. Fuente: Elaboración propia.

Finalmente, las observaciones se podrían tratar como meras categorías sin relación de orden entre ellas y ser representadas como valores ordenables como en la figura 4.5.

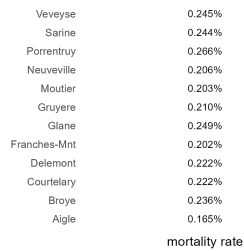


Figura 4.5: Escala ordenable. Fuente: Elaboración propia.

4.2. MÉTODO DE AGREGACIÓN DE LOS DATOS (A)

La segunda dimensión con la cual se propone caracterizar las variables es el método de agregación de los datos. En esta dimensión diferenciamos en primer lugar las variables secuenciales, las de observaciones dispersas y las de observaciones tamizadas.

Valores secuenciales (seq)

Las variables secuenciales establecen el orden en el que se suceden las observaciones. A menudo los conjuntos de datos llevan implícita la

variable secuencial en el orden en el que se suceden los registros, esto es las filas. Otras veces, en cambio, los conjuntos de datos pueden incorporar variables que explicitan el orden en el que se suceden los registros, especialmente si el orden de éstos tiene relevancia. Variables de este tipo se encuentran típicamente en conjuntos de datos que incluyen una variable temporal que marca el momento en el que se produce el registro, en conjuntos de datos agregados según la fecha u otros conjuntos de datos en los que se registra simplemente el orden de las observaciones dado que puede resultar de interés para su análisis. Ahora bien, dado que los datos se tienen que estructurar de uno u otro modo, la variable secuencial puede estar latente en los conjuntos de datos sin llegar a ser representada en una columna de forma explícita.

Por otro lado, los conjuntos de datos se componen también de variables no secuenciales, que no guardan información acerca de la secuencia de las observaciones. Las variables no secuenciales las dividimos en las dos otras categorías del método de agregación de los datos, según si se trata de observaciones dispersas (*scattered data*) u observaciones tamizadas (*gridded data*).

Valores dispersos (sam)

Lo que diferencia las variables de observaciones dispersas y las tamizadas, es la relación entre la cardinalidad de los valores únicos observados y la cardinalidad de todos los posibles valores que podrían llegar a ser observados (o cardinalidad del espacio muestral si se trata de un experimento aleatorio) dentro de un rango de interés. Las variables de tipo disperso se caracterizan por tener una cardinalidad de los valores únicos observados que se encuentra muy por debajo de la de los valores potencialmente observables dentro de un rango de interés. El conjunto de datos `faithful`, por ejemplo, se compone de 272 observaciones dos variables numéricas (`waiting` y `eruptions`). El rango de valores de la variable `waiting` se encuentra entre 43'0 y

96'0, siendo la cardinalidad de los valores observables 531, y la de los valores únicos observados 51. En el caso de la variable `eruptions`, el rango se encuentra entre 1'600 y 5'100, la cardinalidad de los valores observables en este rango de 3.501 y la de los valores únicos observados de 126. La relación entre ambas cardinalidades permite caracterizar ambas variables como de tipo disperso y una gráfica adecuada para representarlas podría ser el diagrama de dispersión como el de la figura 4.6.

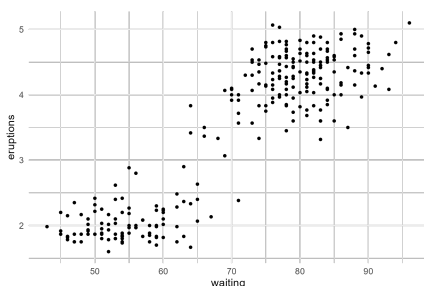


Figura 4.6: Valores dispersos. Fuente: Elaboración propia.

Valores tamizados (pop)

Las variables de tipo tamizado, en cambio, se caracterizan por tener la cardinalidad de los valores únicos observados y la de los potencialmente observables dentro de un rango de interés, coincidente, o tan próxima, que la no observación de una serie de valores resulta relevante. Ejemplos de variables de tipo tamizado los tenemos en variables dicotómicas o en las de tipo factor aunque también en las variables en las que la cardinalidad de los valores posibles es muy elevada, pero el también elevado número de observaciones únicas, convierte en relevante la no observación de ciertos valores o el diferente recuento de frecuencias de observaciones de valores únicos. La matriz `volcano` implementada en R por ejemplo, se compone de 87 filas que corresponden a intervalos de 10 metros de sur a norte y 61 columnas que corresponden a intervalos de 10 metros de este a oeste,

formando un total de 5.307 celdas que representan la cota de una superficie de 100m² del perfil topográfico del volcán Maunga Whau de Auckland. En este caso carece de sentido representar las filas y columnas mediante un diagrama de dispersión dado que las dos variables son de tipo tamizado y el resultado es una matriz de puntos como los de la figura 4.7.

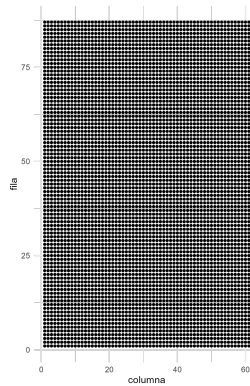


Figura 4.7: Tamiz que representa la observación, o no observación, de pares de valores de dos variables tamizadas. El tamiz permite comprobar que todas las combinaciones de pares de valores tienen como mínimo un valor observado. Fuente: Elaboración propia.

La lectura de la cota tiene un mínimo de 94 y un máximo de 195 siendo la cardinalidad de los valores potencialmente observables de 102 y la de los valores únicos observados en este rango también de 102 que se observan repetidamente (ver figura 4.8 derecha), por lo que esta variable también se puede caracterizar como tamizada.

Una gráfica adecuada para representar las tres variables puede ser el mapa de calor como el de la figura 4.9.

Recodificaciones entre métodos de agregación de los datos

Las variables de valores dispersos son fácilmente recodificables como tamizadas. Simplemente hay que establecer intervalos equidistantes a partir de los cuales hacer el recuento de casos que se sitúan

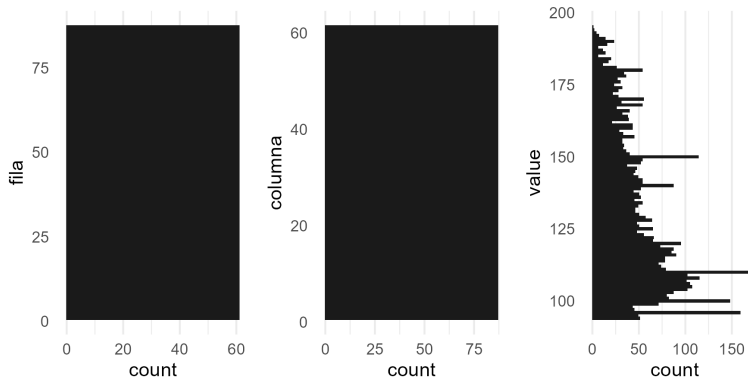


Figura 4.8: Recuento de valores únicos de tres variables tamizadas.
Fuente: Elaboración propia.

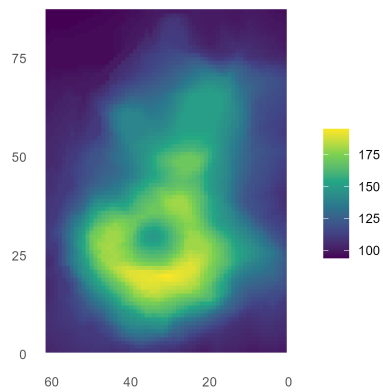


Figura 4.9: Representación de tres variables tamizadas. Fuente:
Elaboración propia.

en cada intervalo. Un ejemplo sencillo lo encontramos en el conjunto de datos `faithful`, cuyas dos variables de valores dispersos se han representado en la figura 4.6 mediante un diagrama de dispersión. Estas dos mismas variables, transformadas en otras de tipo tamizado, permiten igualmente representar el conjunto de datos mediante un mapa de calor como el de la figura 4.10. En el otro sentido resulta imposible recodificar variables de tipo tamizado en otras de tipo disperso porque se carece de los valores atómicos de las observaciones.

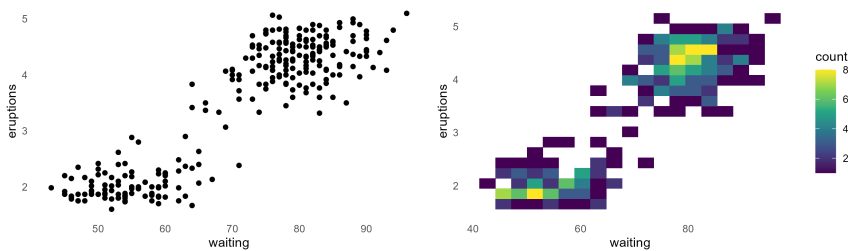


Figura 4.10: Recodificación de valores dispersos a tamizados. Fuente: Elaboración propia.

Otra posible recodificación es de variables secuenciales en variables tamizadas o dispersas. El primer caso se produce si los registros se producen en intervalos pero interesa obtener una representación sin vacíos. Un ejemplo de cómo una variable secuencial puede ser recodificada como de tipo tamizado lo encontramos en el figura 4.11 en la que encontramos a la izquierda una gráfica de puntos conectados y a la derecha un mapa de color.

Recodificar una variable secuencial en otra dispersa se justifica especialmente si las observaciones se toman en intervalos no regulares, si no interesa conectar las observaciones según su secuencia y no interesa rellenar el espacio sin observaciones. Un ejemplo de esto lo tenemos en la figura 4.12 que muestra a la izquierda una secuencia de puntos conectados y a la derecha un diagrama de punto.

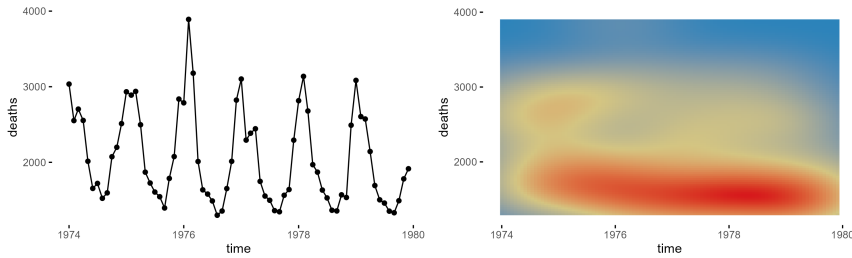


Figura 4.11: Recodificación de valores secuenciales a tamizados.
Fuente: Elaboración propia.

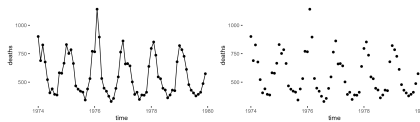


Figura 4.12: Recodificación de valores secuenciales a dispersos.
Fuente: Elaboración propia.

4.3. CICLICIDAD (C)

La ciclicidad se refiere al posible carácter cíclico de las variables. Las variables cíclicas son aquellas cuyos valores, necesariamente ordenados, se encuentran acotados entre un máximo y un mínimo que resultan ser colindantes. Las variables de valores ordenables no pueden ser al mismo tiempo caracterizadas como cíclicas pero sí las del resto de escalas.

Él carácter cíclico de una variable puede aconsejar el uso de sistemas de coordenadas polares, esféricas o cilíndricos en los que cada variable cíclica se desarrolla en una de las dimensiones polares. La ciclicidad es independiente al método de agregación de los datos de modo que encontramos, por ejemplo, ciclos de días de la semana, que es una variable ordenada tamizada, o lecturas dispersas de dirección del oleaje medido en grados o estas mismas lecturas tamizadas según intervalos equidistantes.

Esta dimensión distingue, por consiguiente, entre dos categorías según si las variables son cíclicas (*cycl*) o acíclicas (*ncycl*). Pudiendo

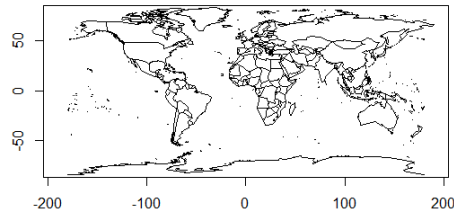


Figura 4.13: Recodificación de valores cíclicos en acíclicos. Fuente: Elaboración propia.

las primeras ser recodificadas como acíclicas (figura 4.13) y otras veces, las variables acíclicas pueden recodificarse como cíclicas, por ejemplo el tiempo que a pesar de no poder dar marcha atrás, puede recodificarse en valores cíclicos como los días de un año, meses, días de la semana, (ver figura 4.14) etc. No hay que confundir, sin embargo, la ciclicidad de la variable con el posible carácter cíclico que se puede observar en una secuencia de valores, dado que el primero hace referencia al dominio de la variable y el segundo a la secuencia de las observaciones.

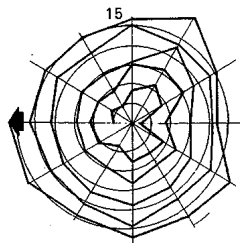


Figura 4.14: Recodificación de valores acíclicos en cíclicos. Fuente: Bertin (1967, p.214)

4.4. EXPLICITUD (E)

La cuarta dimensión de la caracterización de las variables es la explicitud de su escala. Las variables pueden adquirir, según esta dimensión, dos categorías: explícitas (*exp*) o ambiguas (*amb*). Las

variables explícitas son aquellas cuyas escalas se encuentran representadas de manera explícita y las variables ambiguas son aquellas que son consideradas por la gráfica aunque éstas no muestren los valores de los que se componen sus escalas.

Una variable en los datos o calculada que se desarrolla en una variable espacial de la gráfica, resulta explícita si la gráfica incluye un eje con etiquetas que acota y da soporte a la escala. Si, en cambio, la gráfica no incluye las etiquetas en el eje que da soporte a la escala, esta variable resulta ambigua.

Si la variable se desarrolla mediante una variable de retina y la gráfica incluye en la leyenda el soporte de esta escala con sus valores específicos, entonces la variable resulta explícita. Si por el contrario, una variable en los datos o calculada se desarrolla mediante una variable de retina pero la gráfica no incluye una leyenda que de soporte a la escala de la variable de retina, entonces ésta se torna ambigua. Las variables con escala ambigua son aquellas cuya escala no se encuentra explicitada mediante etiquetas en los ejes o leyendas o etiquetas en el área gráfica, pero cuyos valores son necesarios para construir la gráfica.

Una variable ambigua permite conocer otros aspectos de la variable que no son los valores de su escala, por ejemplo, el número de registros o el número de valores únicos. Un diagrama de dispersión permite averiguar el número de registros que incluye un conjunto de datos sin necesidad de identificar las observaciones mediante etiquetas. Una pirámide de población permite comparar recuentos según intervalos de edad de dos grupos de una población sin necesidad de explicitar qué parte de la pirámide pertenece a cada grupo. Otras veces la variable explícita desarrolla variables de retina y permite, por ejemplo, representar diferentes grupos en una gráfica mediante diferentes tonos de color sin que la gráfica incluya una leyenda que indique qué color corresponde a cada grupo (ver figura 4.15).

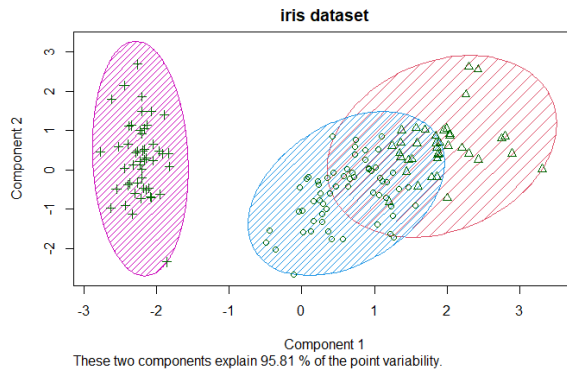


Figura 4.15: Diagrama de dispersión por grupos con las VV forma y tono de color ambiguas. Fuente: Elaboración propia.

4.5. LONGITUD DE LAS VARIABLES (L)

La quinta dimensión de la caracterización que proponemos de las variables es la longitud de la variable que ya definió Bertin (1967)(p.9) entendida como el número de valores únicos de una variable que resulta útil identificar y, por consiguiente, representar. No se debe confundir la longitud de las variables con la cardinalidad de la variable en los datos o calculada que es el número de valores únicos, o la cardinalidad de su escala que es el número de valores únicos que pueden ser observados potencialmente. Existe otra longitud de las variables pero, en este caso, de las variables visuales que es el número de valores únicos que una variable visual permite representar de manera que puedan decodificarse con facilidad.

Los motivos por los que la longitud de las variables puede no coincidir su cardinalidad son variados. En el caso de variables categóricas puede ocurrir que todos los valores de la escala sean observados pero que interese explicitar que no existen valores no informados. Puede ocurrir también que no todos los valores de la escala sean observados e interese explicitar qué valores no han sido observados. Puede ocurrir también que no resulte útil diferenciar entre unos determinados valores en los datos de modo que se agrupen en una

sola categoría. En el caso de variables cuantitativas es posible reducir su longitud simplemente despreciando decimales si la solución no requiere un nivel de precisión tan elevado. Otra posibilidad para el caso de variables cuantitativas es recodificar los valores dispersos en valores tamizados entre distancias equidistantes o mediante intervalos variables si la precisión necesaria no es homogénea.

La longitud de la variable resulta especialmente crítica en el momento de seleccionar una gráfica u otra a consecuencia de las limitaciones en las longitudes de las variables visuales. Por ejemplo, desarrollar mediante el tono de color una variable categórica de longitud 20 puede resultar inadecuado porque entre 20 tonos diferentes de color puede resultar un reto no confundir ninguno de ellos. Adicionalmente, una misma variable visual puede contar con una longitud u otra en función de si desarrolla una variable dispersa (ver figura 4.16) u otra tamizada (ver figura 4.17). La longitud de las variables puede determinar también las dimensiones de la gráfica en el caso de variables categóricas cuando éstas se soportan en variables espaciales y puede convertir una gráfica útil en inútil si, por ejemplo, se superpuebla con etiquetas las observaciones en un diagrama de dispersión de modo que las marcas colisionen y se hagan ilegibles. Otra consideración es que la longitud de la variable puede condicionar el uso de traslaciones, rotaciones o reflexiones hasta el punto que existen construcciones gráficas aptas para una determinada longitud de la variable, como por ejemplo, las pirámides de población que son específicas para variables dicotómicas.

4.6. CLASIFICACIÓN DE LAS GRÁFICAS BASADAS EN LA CARACTERIZACIÓN

En las secciones anteriores hemos visto cómo es posible caracterizar, por separado, las variables de un conjunto de datos según diferentes dimensiones y niveles. Por otro lado, cabe clasificar las gráficas según la adecuación a las características previamente descritas

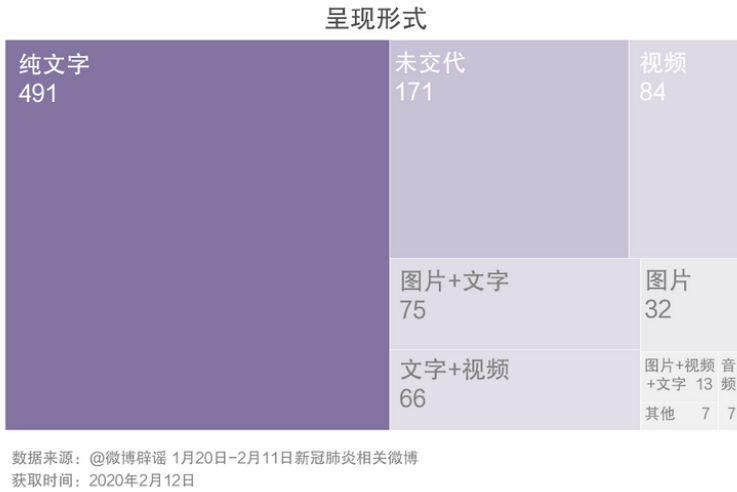


Figura 4.16: *Treemap* en el que la luminosidad representa una variable dispersa, concretamente, el recuento de 946 rumores sobre el COVID agrupados según el canal de divulgación. Fuente: covic-archive.org. Consultada el 23 de diciembre de 2021

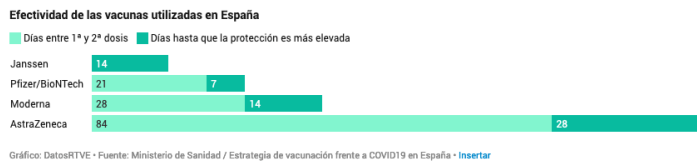


Figura 4.17: Gráfica de barras apiladas que representan los días necesarios para alcanzar la plena efectividad de diferentes vacunas. El tono de color representa una variable tamizada, concretamente, dos intervalos diferentes de días entre sucesos. Fuente: covic-archive.org. Consultada el 23 de diciembre de 2021

de las variables. Una estrategia para la clasificación de las gráficas es, primero, clasificarlas según el número de variables representadas y que pueden asociarse a una variable en los datos o calculada. Luego según la escala gráfica de las variables, luego según el método de agregación de los datos, la ciclicidad, la explicitud y finalmente la longitud de las variables. La idea que subyace en esta estrategia de clasificación es que cuanto más detallada es la caracterización de las variables por separado, menor es el conjunto de gráficas estadísticas que se adecuan a dicha caracterización.

Una limitación de esta estrategia de clasificación es que tiene que existir una correspondencia identificable entre las variables en los datos y las variables visuales que codifican la información, por lo que, por ejemplo, métodos estadísticos de reducción de dimensionalidad como el análisis de componentes principales (PCA) mediante diagramas de dispersión no están considerados al no poder identificar las variables desplegadas en cada uno de los ejes espaciales con una única variable en los datos.

Otra limitación de esta estrategia es que resulta útil para gráficas que representan un número reducido de variables dado que a medida que aumenta el número de variables, el número de combinaciones de estas variables con sus respectivas caracterizaciones aumenta exponencialmente y resulta costoso encontrar gráficas específicamente adecuadas para cada una de las combinaciones, aunque, visto de otro modo, identificar estas gráficas puede convertirse en un reto en vez de una limitación.

Matriz de escalas gráficas y métodos de agregación de los datos

Para clasificar cada una de las variables a representar por una gráfica según su escala gráfica (M) y su método de agregación de los datos (A), puede resultar útil la matriz $M \times A$, que contiene todos los posibles emparejamientos de los valores de ambas dimensiones. Si a esta matriz le añadimos las posibles recodificaciones entre escalas

gráficas de medida y entre métodos de agregación de los datos, obtenemos la matriz de escalas gráficas y métodos de agregación de los datos representada en la figura 4.18.

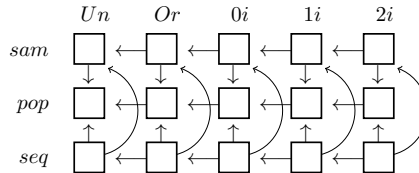


Figura 4.18: Matriz de escalas gráficas y métodos de agregación de los datos. Fuente: Elaboración propia.

Con una sola matriz ya podríamos clasificar las gráficas de una única variable según los tres primeros criterios antes comentados, es decir, el número de variables (en este caso una), la escala gráfica y el método de agregación. Por ejemplo, podemos asociar las siguientes gráficas a las combinaciones de la figura 4.19.

$0i^{sam}$		simple point graph, one-axis point graph
$1i^{sam}$		simple bar graph, bar graph with added random jitter
$2i^{sam}$		dichotomous pie chart
Un^{pop}		reorderable matrix
Or^{pop}		semi-reorderable matrix
$0i^{pop}$		point graph
$1i^{pop}$		bar graph
$2i^{pop}$		bar graph bounded on both ends
Un^{seq}		sequentially ordered list, arc diagram with orderable nodes
Or^{seq}		check-list in a sequentially ordered Likert-type scale, arc diagrams
$0i^{seq}$		line graph
$1i^{seq}$		area graph
$2i^{seq}$		area graph bounded on both ends

Figura 4.19: Gráficas que codifican los valores de una variable. Fuente: Elaboración propia.

Si en vez de una matriz, utilizamos dos, entonces podemos clasificar las gráficas que requieren dos variables de entrada, o bien una variable de entrada y otra calculada y así sucesivamente. La figura 4.20 recoge algunos ejemplos.

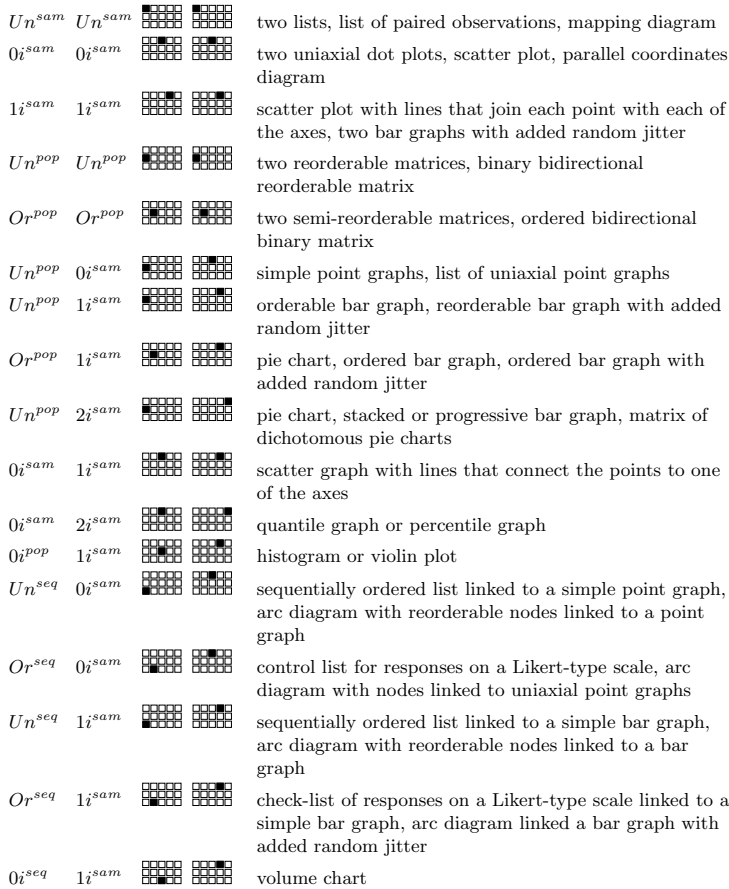


Figura 4.20: Gráficas que codifican los valores de dos variables.

Fuente: Elaboración propia.

En el caso de gráficas que codifican 3 variables necesitamos tres matrices. La 4.21 recoge también algunos ejemplos.

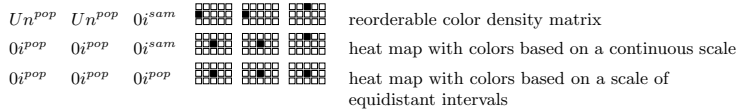


Figura 4.21: Gráficas que codifican los valores de tres variables. Fuente: Elaboración propia.

Ventajas de la ciclicidad en la caracterización

Las variables cíclicas sugieren en uso de sistemas de coordenadas polares, cilíndricos o esféricos. Se pueden caracterizar como cíclicas las variables cuya escala no sea ordenable y se requiere conocer el rango de la escala de la variable cíclica para poder graficarla correctamente. Los ejemplos de la figura 4.22 incluyen gráficas que codifican una variable cíclica, la figura 4.23 muestra ejemplos de gráficas que codifican 2 variables, una de ellas cíclica y la figura 4.24 muestra ejemplos de gráficas que codifican 3 variables, una de ellas cíclica.

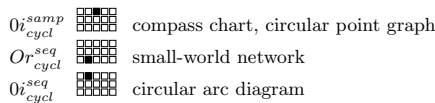


Figura 4.22: Gráficas que codifican los valores de una variable cíclica. Fuente: Elaboración propia.

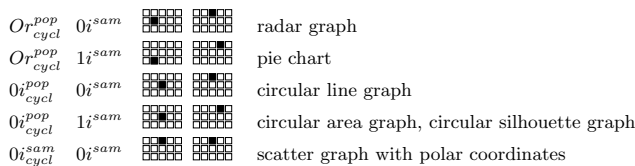


Figura 4.23: Gráficas que codifican los valores de dos variables, una de ellas cíclica. Fuente: Elaboración propia.

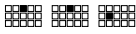
$0i_{cycl}^{sam}$ $0i^{sam}$ Un^{pop}  grouped scatter graph

Figura 4.24: Gráficas que codifican los valores de tres variables, una de ellas cíclica. Fuente: Elaboración propia.

Ventajas de la explicitud en la caracterización

La caracterización de las variables como ambiguas implica diferentes cambios en las gráficas. Si la variable ambigua se soporta en una VV espacial, simplemente basta con suprimir las etiquetas de escala. Si se soporta en una VV de retina, puede bastar con eliminar la leyenda. Si la variable que se pretende mostrar de manera ambigua identifica las observaciones mediante etiquetas en el área gráfica, basta con eliminar estas etiquetas. La ambigüación de variables puede producir gráficas específicas como por ejemplo el *spaghetti plot* como el de la figura 4.43 en la que la variable **Seed** del conjunto de datos **Loblolly** se encuentra presente pero de forma ambigua. En la figura 4.25 se muestran otras combinaciones que se pueden relacionar con gráficas con variables ambiguas.

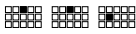
$0i^{sam}$ $0i^{sam}$ Un_{amb}^{pop}  scatter graph, parallel coordinate graph

Figura 4.25: Gráficas que codifican los valores de tres variables, una de ellas ambigua. Fuente: Elaboración propia.

Ventajas de la longitud de la variable en la caracterización

La longitud de la variable es determinante en el momento de seleccionar una gráfica u otra. Son múltiples las gráficas específicas para variables de longitud 1 como por ejemplo el velocímetro, el manómetro o el termómetro analógico. Otras gráficas son específicas para variables de longitud 2 como por ejemplo la pirámide de población (ver figura 4.26) o el *slope diagram*. La longitud incide también en la conveniencia de utilizar una o otra variable espacial dado que éstas permiten diferenciar con facilidad un número limitado

y diferente de valores. También determina el tamaño de gráfica si se pretende facilitar la lectura de las etiquetas de escala y pueden sugerir el uso de gráficas multipanel condicionadas.

$0i^{pop}$ $1i^{sam}$ $Un^{pop(2)}$  population pyramid

Figura 4.26: Gráficas que incluyen variables de una longitud específica. Fuente: Elaboración propia.

4.7. EJEMPLOS BASADOS EN UN CONJUNTO DE DATOS

Una vez descrita la caracterización de gráficas basada en la caracterización multidimensional de datos, ahora presentamos los resultados que un sistema automatizado de gráficas estadísticas podría sugerir en base a esta estrategia. Para ello, utilizamos el conjunto de datos `Loblolly`, limitado a cuatro variables, que relaciona el crecimiento del pino piñonero en 84 plantaciones. Para cada plantación, el conjunto de datos incluye el promedio de la altura de los árboles medida en pies, la edad de la plantación en años, y la fuente de la semilla de los árboles. A pesar del número limitado de variables de este conjunto de datos, los resultados también serían válidos para combinaciones de variables con las mismas características de otros conjuntos de datos.

Caracterización de las variables

La variable `Id` de la plantación está compuesta por categorías. El modo de agregación de los datos, suponiendo que todos los valores de interés estén presentes, puede caracterizarse como tamizada, pero dado que el orden en que aparecen estos códigos numéricos no es estrictamente ascendente, es preferible caracterizar la variable como de tipo secuencial para no perder información que podría ser de interés. En cuanto al resto de dimensiones, esta variable se caracteriza como acíclica, explícita (porque se pretende mostrar sus valores en la gráfica), y de longitud 84.

La variable `height` se compone de escalares limitado en un extremo y de tipo de disperso (dado que los 84 valores de la variable representan una pequeña muestra de la valores potencialmente observables). Su dominio es acíclico, la escala es explícita, y su longitud es cercana a 650 si consideramos que una décima de pie es suficientemente precisa para la decodificación de la gráfica.

La variable `age` también se compone de escalares delimitados por un extremo. El número de valores únicos es 6, cada uno de ellos con una frecuencia de 14, por lo que esta variable se caracteriza como tamizada, acíclica, explícita y con una longitud de 6.

La variable `seed` está compuesta por 14 categorías ordenadas según los resultados obtenidos en la variable `height`. Es una variable tamizada, de dominio acíclico, de escala explícita y de longitud 14, igual al número de categorías.

Para organizar las posibles gráficas en función de las variables seleccionadas y sus posibles recodificaciones, primero describimos las gráficas que pueden representar cada variable por separado. Luego combinamos dos o más variables caracterizadas *a priori*. Finalmente, identificamos otras posibles representaciones gráficas a partir de una selección de variables específicas sobre las que se aplica una recodificación en los niveles de al menos una de sus variables.

Combinaciones

UNIVARIADAS Si seleccionamos la variable `Id` por separado, dado que la longitud de la variable es 84, una posible representación es una lista de estos códigos ordenados por su posición en el conjunto de datos. La lista podría presentarse como una matriz de valores ordenados por filas y columnas (ver figura 4.27), una sola fila o columna con una barra de desplazamiento, o varios paneles ordenados entre los que el usuario puede pasar presionando una tecla.

Id											
1.	1	15	31	29	61	43	8	57	38	71	68
2.	15	16	45	30	75	44	22	58	52	72	32
3.	29	17	59	31	6	45	36	59	66	73	13
4.	43	18	73	32	20	46	50	60	80	74	27
5.	57	19	4	33	34	47	64	61	11	76	41
6.	71	20	18	34	48	48	78	62	25	76	55
7.	2	21	32	35	62	49	9	63	39	77	69
8.	16	22	46	36	76	50	23	64	53	78	83
9.	30	23	60	37	7	51	37	65	67	79	14
10.	44	24	74	38	21	52	51	66	81	80	28
11.	58	25	5	39	35	53	65	67	12	81	42
12.	72	26	19	40	49	54	79	68	26	82	56
13.	3	27	33	41	63	55	10	69	40	83	70
14.	17	28	47	42	77	56	24	70	54	84	84

Figura 4.27: Matriz de valores ordenados en filas y columnas.
Fuente: Elaboración propia.

Si seleccionamos la variable `height`, ésta podría presentarse en una gráfica de puntos agitados con líneas descendentes o un gráfico de puntos con líneas descendentes (ver figura 4.28).

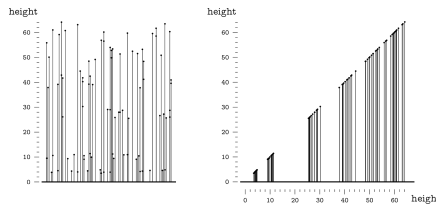


Figura 4.28: Gráficas de puntos agitados con líneas descendentes (izquierda) y de puntos con líneas descendentes (derecha). Ambas gráficas representan la variable `height` limitada por un extremo.
Fuente: Elaboración propia.

Si seleccionamos la variable `age`, ésta se podría presentar en una gráfica de áreas superpuestas con rectángulos que tienen una dimensión proporcional a la edad y otra al conteo de frecuencia de los valores o un gráfico de puntos con líneas de caída (ver figura 4.29).

Finalmente, si solo seleccionamos la variable `seed`, ésta se podría volver a presentar en una lista ordenada, una gráfica de puntos con líneas descendentes o un gráfico de barras con la longitud de cada línea o barra proporcional a la frecuencia de conteo, y ordenados según el orden asignado a esta variable cualitativa (ver figura 4.30).

BIVARIADAS Y MULTIVARIADAS Cualquier combinación de la variable `Id` con las demás se puede representar a través de una tabla

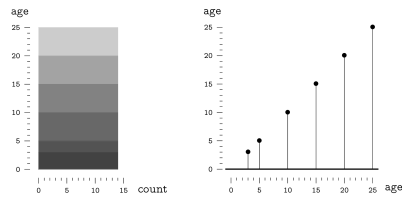


Figura 4.29: Gráficas de áreas superpuestas (izquierda) y de puntos con líneas descendientes (derecha). Ambas gráficas representan la variable `age` limitada por un extremo. Fuente: Elaboración propia.

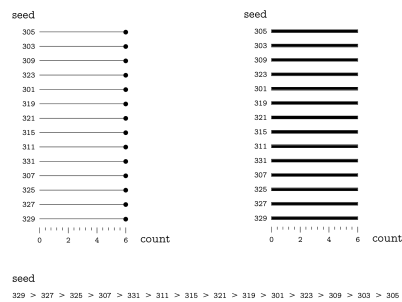


Figura 4.30: Gráficas de punto (panel superior izquierda), de barras (panel superior derecha) y lista ordenada. Las tres gráficas representan la variable `seed`. Fuente: Elaboración propia.

semigráfica. Si combinamos la variable `Id` con `height` o con `age`, podemos presentar una lista ordenada en la que se puede ubicar la columna que corresponde a `height` o `age` a través de un gráfico de barras simple con barras proporcionales a la altura o la edad. Dada la longitud de la variable `Id`, se pueden utilizar las mismas técnicas mencionadas anteriormente para mostrar la lista. Si incluimos la variable `seed`, cada fila puede incluir una marca rellena con diferentes densidades de color en un aumento secuencial de acuerdo con los 14 valores ordenados de la variable `seed`. Si se seleccionan las cuatro variables, la tabla puede incluir todas las columnas mencionadas (ver figura 4.31).

La combinación `height` y `age` se puede representar a través de una gráfica de áreas superpuestas con rectángulos que tienen dimensiones

Id	height	age	seed
1	—	—	■
15	—	—	■
29	—	—	■
43	—	—	■
57	—	—	■
71	—	—	■
2	—	—	■
16	—	—	■
30	—	—	■
44	—	—	■
58	—	—	■
72	—	—	■
3	—	—	■
17	—	—	■
31	—	—	■
45	—	—	■
59	—	—	■
73	—	—	■
4	—	—	■
18	—	—	■
32	—	—	■
46	—	—	■
60	—	—	■
74	—	—	■
5	—	—	■
19	—	—	■
33	—	—	■
47	—	—	■
61	—	—	■
75	—	—	■
6	—	—	■
20	—	—	■
34	—	—	■
48	—	—	■
62	—	—	■
76	—	—	■
7	—	—	■
21	—	—	■
35	—	—	■
49	—	—	■
63	—	—	■
77	—	—	■
8	—	—	■
22	—	—	■
36	—	—	■
50	—	—	■
64	—	—	■
78	—	—	■
9	—	—	■
23	—	—	■
37	—	—	■
51	—	—	■
65	—	—	■
79	—	—	■
10	—	—	■
24	—	—	■
38	—	—	■
52	—	—	■
66	—	—	■
80	—	—	■
11	—	—	■
25	—	—	■
39	—	—	■
53	—	—	■
67	—	—	■
81	—	—	■
12	—	—	■
26	—	—	■
40	—	—	■
54	—	—	■
68	—	—	■
82	—	—	■
13	—	—	■
27	—	—	■
41	—	—	■
55	—	—	■
69	—	—	■
83	—	—	■
14	—	—	■
28	—	—	■
42	—	—	■
56	—	—	■
70	—	—	■
84	—	—	■

Figura 4.31: Tabla semigráfica que combina las variables Id, height, age y seed. Fuente: Elaboración propia.

proporcionales a estas dos variables. Dado que la variable `age` se describe como tamizada, los rectángulos pueden agruparse por edad y luego representarse en una matriz de gráficos de áreas superpuestas (ver figura 4.32).

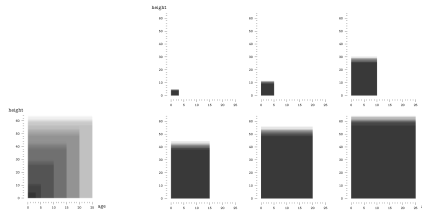


Figura 4.32: Gráfica de áreas superpuestas (izquierda) y gráfica multipanel de áreas superpuestas (derecha) que combinan las variables `height` y `age`. Fuente: Elaboración propia.

La combinación de `height` y `seed` se puede representar con una matriz de gráficos de puntos agitados con líneas descendentes ordenadas según el tipo de semilla (ver figura 4.33).

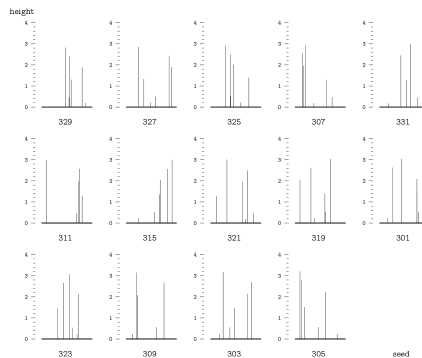


Figura 4.33: Gráfica de puntos agitados con líneas descendentes ordenadas según el tipo de semilla. Se encuentran representadas las variables `height` y `seed`. Fuente: Elaboración propia.

La combinación `age` y `seed` se puede representar mediante una serie de gráficas de área superpuestas también ordenadas según la semilla (ver figura 4.34).

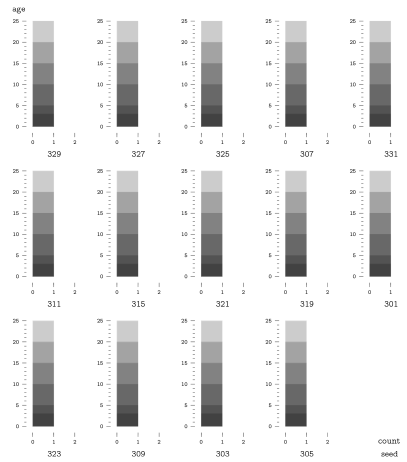


Figura 4.34: Gráfica de áreas superpuestas ordenadas según el tipo de semilla. Se encuentran representadas las variables `age` y `seed`. Fuente: Elaboración propia.

Si se seleccionan las variables `height`, `age` y `seed`, éstas se pueden representar mediante una matriz de gráficas de área superpuestas ordenadas por tipo de semilla (ver figura 4.35).

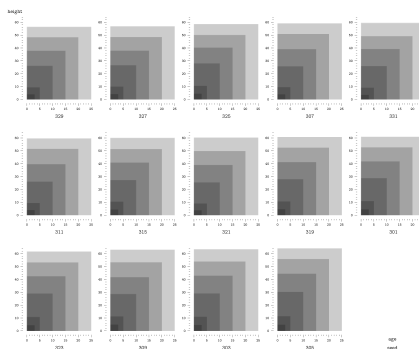


Figura 4.35: Matriz de gráficas de área superpuestas ordenadas según el tipo de semilla. Se encuentran representadas las variables `height`, `age` y `seed`. Fuente: Elaboración propia.

Combinaciones con variables recodificadas

Una posible recodificación es considerar la `height` como una variable escalar no acotada. Esto tiene sentido si nos interesa más la relación entre los valores observados que su relación con el origen o cero. Si en este caso solo se selecciona esta variable, se podría traducir en una gráfica de puntos uniaxial que puede utilizar varias técnicas para evitar colisiones de puntos, o un gráfica de tiras (ver figura 4.36).

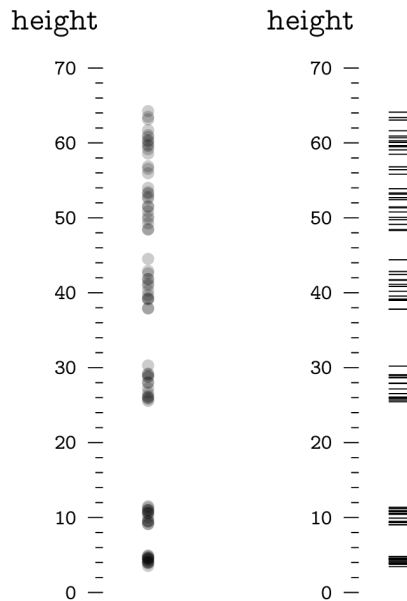


Figura 4.36: Gráfica uniaxial de puntos (izquierda) y gráfica uniaxial de tiras que representan la variable `height` una vez recodificada como no acotada. Fuente: Elaboración propia.

Si se selecciona junto con la variable `Id`, la columna de la tabla semigráfica que corresponde a la variable `height` mostrará una gráfica de puntos simple en lugar de una gráfica de barras simple (ver figura 4.37).

Una posible representación de la variable `height` combinada con la variable `age` es una gráfica que traza la edad en su eje X y la altura

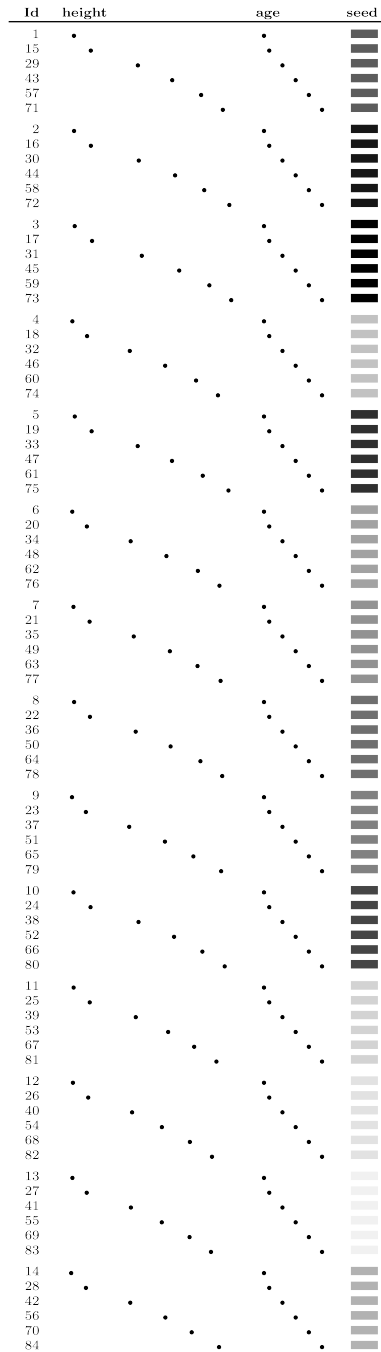


Figura 4.37: Tabla semigráfica que incluye, entre otras, la variable `Id` y la variable `height` una vez recodificada como no acotada. Fuente: Elaboración propia.

en su eje Y y conecta los puntos al eje y con líneas (ver figura 4.38).

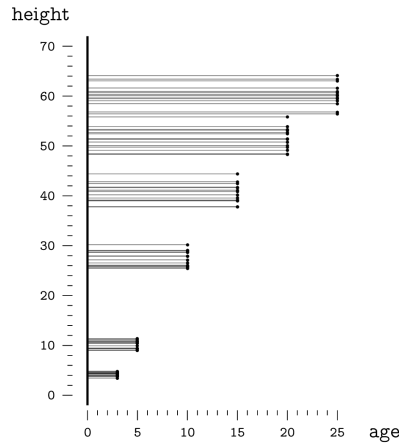


Figura 4.38: Gráfica de puntos con líneas descendientes hacia el eje Y que representa la variable `age` y la variable `height` una vez recodificada como no acotada. Fuente: Elaboración propia.

Finalmente, la combinación de las variables `height` y `seed` ahora produciría una serie de gráficas de puntos o gráficas de tiras ordenadas por tipo de semilla según el orden asignado para esta variable (ver figura 4.39).

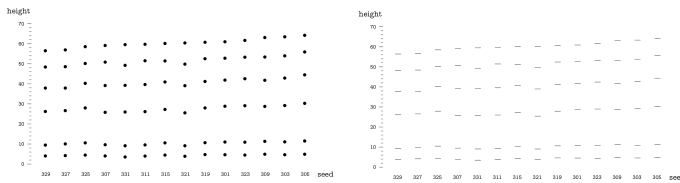


Figura 4.39: Gráficas de puntos (izquierda) y de tiras (derecha) que representa la variable `seed` y la variable `height` una vez recodificada como no acotada. Fuente: Elaboración propia.

Si además de la recodificación mencionada en el párrafo anterior, también consideramos la variable `age` como una variable escalar no acotada, la selección de las variables `height` y `age` daría como resultado un diagrama de dispersión que no necesariamente incluiría cero en cualquiera de sus ejes (ver figura 4.40).

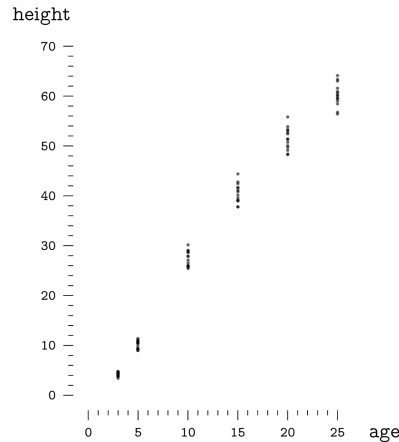


Figura 4.40: Diagrama de dispersión que representa las variables `height` y `age` una vez recodificadas ambas como no acotadas.

Si luego recodificamos la variable `height` como una variable tamizada y seleccionamos solo esta variable, terminaríamos con un diagrama de violín, un diagrama de caja o un histograma (ver figura 4.41).

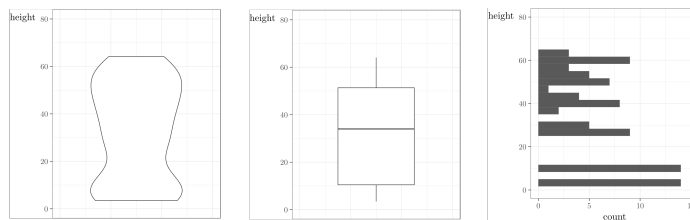


Figura 4.41: Diagramas de violín, de caja e histograma. Los 3 representan la variable `height`. Fuente: Elaboración propia.

Si lo combinamos con `age`, obtendríamos una sucesión de cualquiera de los diagramas anteriores ordenados por grupos de edad (ver figura 4.42).

Recodificar cualquiera de las variables como ambigua resultaría en la omisión de la escala para esa variable en el gráfico. Esto incluye la superposición de paneles en lugar de su yuxtaposición, la exclusión

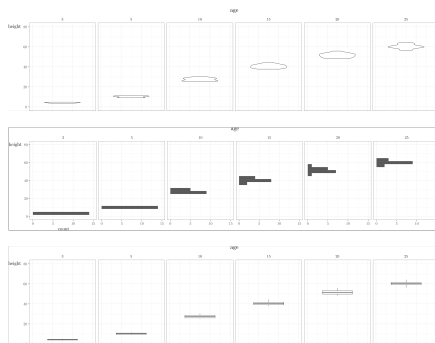


Figura 4.42: Diagramas multipanelados de violín, de caja e histograma. Los 3 representan la variable `height` combinada con `age`. Fuente: Elaboración propia.

de etiquetas de escala en los ejes en el caso de variables espaciales, así como la exclusión de la leyenda en el caso de variables de retina. Por ejemplo, la combinación de las variables `height` y `age` registradas como escalares ilimitados, además de la variable `seed` codificada como ambigua, produciría un diagrama de espagueti (4.43) en lugar de una matriz de gráficas de línea ordenadas por tipo de semilla.

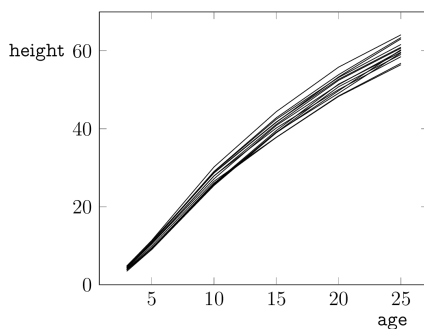


Figura 4.43: Diagrama de espagueti representa las variables `height` y `age` recodificadas como no acotadas más la variable `seed` recodificada como ambigua. Fuente: Elaboración propia.

4.8. DISCUSIÓN

En esta sección, compararemos nuestra propuesta de caracterización de variables con las otras soluciones que revisamos y mostraremos cómo nuestro nuevo enfoque genera resultados diferentes y algo más precisos. Cabe señalar que los sistemas con los que estamos comparando nuestro marco son, de alguna manera, sistemas pioneros con limitaciones en términos del conjunto de gráficas presentados a los usuarios. El sistema CHART, por ejemplo, solo muestra matrices de gráficas de barras que pueden tener diferentes sombreados y matrices de gráficas circulares de diferentes tamaños. El sistema BHARAT solo presenta gráficos circulares, gráficos de barras y gráficos de líneas, así como combinaciones de estos. Los sistemas APT, SAGE, BOZ y EAVE muestran diagramas y redes, pero solo con dos dimensiones espaciales. Los sistemas NSP y Vista también incluyen gráficos tridimensionales, mientras que los sistemas Polaris y Tableau incluyen mapas. Sin embargo, el catálogo de tipos de gráficas en estos cuatro sistemas también es limitado. Finalmente, el sistema ViSta enfatiza la interacción dinámica con gráficas vinculadas dinámicamente, pero la cantidad de tipos diferentes de gráficas que ofrece también es limitada.

Anteriormente, describimos las diferentes estrategias que permiten refinar la selección de gráficas en función de las características de los datos, el usuario, el hardware y los modelos de representación. La caracterización de los datos presentados sigue la estrategia funcional basada en las características de los datos. Esto permite caracterizar las representaciones gráficas a partir de la caracterización de los datos para presentar al usuario una gama de gráficas posibles para un conjunto de datos determinado. La doble caracterización de datos y gráficos ha sido implementada por sistemas como SAGE con SageBook y Tableau con Show ME.

Mackinlay consideró que este enfoque estaba demasiado simplificado porque no había garantía de que existiera un diseño apropiado para una variedad tan grande de situaciones. Por lo tanto, era necesario considerar la lista completa de soluciones *ad hoc*, aunque solo unas pocas alternativas podrían ser aceptables. Desde nuestro punto de vista, el argumento de que no hay garantía de encontrar un método apropiado para una gran variedad de combinaciones sirve, primero, como un desafío para encontrar estas combinaciones y, segundo, como una oportunidad para que los creadores de visualizaciones propongan métodos gráficos apropiados para estas combinaciones. Con respecto a la necesidad de considerar la lista completa de soluciones *ad hoc*, creemos que es necesario clasificar la mayor cantidad de métodos gráficos precisamente para descartar aquellas alternativas que no son aceptables.

La presentación de gráficas sin determinar previamente la tarea a realizar da como resultado gráficas adecuadas para una determinada tarea con diversos grados de efectividad. Los sistemas como APT, Vista y EAVE no preguntan sobre la tarea y proporcionan solo una representación gráfica supuestamente óptima. Por el contrario, la estrategia que proponemos presenta al usuario varias posibilidades gráficas para un conjunto de datos, al igual que el sistema ViSta, que también incluye otras consideraciones para sugerir gráficas, como la distribución teórica, que compara con la empírica, y el tipo de análisis estadístico seleccionado. Sin embargo, en cuanto a la estrategia presentada, tiene el inconveniente de que ofrece una gama limitada de gráficas, seleccionadas *ad hoc*, para que el usuario elija. Para mejorar la selección automática de gráficas de acuerdo con la estrategia presentada, sería conveniente realizar estudios cognitivos que clasifiquen conjuntos de gráficos posibles para cada combinación de variables en función de la facilidad con la que permitan ejecutar una serie de tareas perceptivas.

La caracterización de los datos presentados deriva del trabajo

de Jaques Bertin, quien, sin embargo, no consideró las diferentes escalas de medida para las variables cuantitativas; en consecuencia, su caracterización de datos agrupa, en una sola combinación, gráficas tan diversas como gráficas de barras, gráficas circulares y gráficas de barras apiladas. Adicionalmente, en el nivel de variables ordenadas, Bertin reúne variables cualitativas que mantienen una relación de mayor a menor y variables secuenciales, lo que da como resultado una combinación única para gráficas tan diversas como un diagrama de Gantt y una matriz semireordenable.

Con el sistema CHART, solo es posible graficar variables cuantitativas. Por ello y por ser un sistema pionero en la automatización de gráficos estadísticos, su gama de gráficas es muy limitada. Otros sistemas, como APT, NSP, BOZ, Vista, EAVE, Polaris, Tableau y VizRec, consideran entre dos y seis niveles en una sola dimensión. Las combinaciones posibles con hasta tres variables con dos niveles son nueve. Con tres niveles aumenta a 19, con cuatro a 34, con cinco a 55 y con seis a 83. El sistema SAGE utiliza una caracterización bidimensional, pero la segunda dimensión, el dominio de pertenencia, no convence dado que no considera otras magnitudes físicas fundamentales, como la intensidad de una corriente eléctrica o de una fuente de luz, ni magnitudes derivadas de magnitudes físicas fundamentales. El sistema BHARAT tiene hasta cinco dimensiones; los dos primeros son dicotómicos, pero el sistema no establece dimensiones predeterminadas para los demás y parece que el algoritmo se ve obligado a utilizar límites *ad hoc* a la hora de evaluar cada gráfica posible, lo que hace imposible saber el número de combinaciones que permite esta caracterización. La caracterización presentada, considerando sólo las dos primeras dimensiones, permite un total de 815 combinaciones de hasta tres variables, por lo que la gama de gráficas posibles se reduce necesariamente.

4.9. LIMITACIONES

Si bien este marco de clasificación y presentación automática de gráficos es válido para gráficos que representan un gran número de variables, este estudio se limita a representaciones gráficas de un máximo de tres variables. Esto se debe a que, a medida que aumenta la cantidad de variables seleccionadas de un conjunto de datos, se produce un aumento exponencial en la cantidad de combinaciones posibles y en la gama de gráficos posibles para cada combinación. Esto se debe a que cada variable en un conjunto de datos se puede representar de varias formas (como puntos, líneas o áreas), con varias variables visuales y varios sistemas de coordenadas. Adicionalmente, se puede utilizar una yuxtaposición o superposición de paneles. El estudio tampoco considera variables compuestas por valores de tipo vector y tensor.

4.10. CONCLUSIONES Y TRABAJOS FUTUROS

Hemos presentado una caracterización multidimensional para variables individuales que puede servir como marco para la clasificación de gráficos estadísticas y permitir reducir notablemente la gama de posibilidades gráficas para un conjunto de datos dado. El método propuesto se puede utilizar para automatizar la presentación de gráficos estadísticas en función de las características de los datos y también para encontrar nuevas combinaciones que actualmente no tienen métodos gráficos asociados, creando así nuevas oportunidades para diseñar visualizaciones novedosas.

El siguiente paso en esta línea de trabajo sería crear una base de datos de gráficos que se caractericen según la tipología de fuente de datos con la que cada gráfica sea compatible. Esta base de datos se puede construir a partir de los gráficos mencionadas, por ejemplo, en la literatura científica. Para cada combinación de variables, podemos crear un árbol de gráficos compatibles que además considere las

posibles recodificaciones entre niveles para cada variable. Una segunda tarea complementaria sería la creación y distribución de un paquete R que implementaría esta caracterización y posibilitaría la presentación de una gama de gráficas compatibles con un conjunto de datos. Finalmente, para mejorar la selección de gráficas adecuadas a cada situación, sería necesario incluir la tarea perceptiva como criterio en la selección de la representación gráfica; esto requeriría estudios cognitivos para evaluar la efectividad de los gráficos asociados a cada combinación con una taxonomía de tareas perceptivas.

CAPÍTULO 5

BRINTON PARA GEDA UNIVARIADO

“Well-designed charts are empowering. They enable conversations. They imbue us with X-ray vision, allowing us to peek through the complexity of large amounts of data.”
— Alberto Cairo

El presente capítulo de esta monografía adapta el artículo de Millán-Martínez y Oller (2020) publicado en la revista *The R Journal* en 2020 que presenta el paquete `brinton` del entorno de programación estadística R. Éste mismo paquete ha incorporado, después de la publicación del artículo, nuevas utilidades que se describen en el capítulo 6.

En el momento de desarrollar el paquete, la primera idea fue implementar el marco teórico presentado en el capítulo 4, pero esto hubiera derivado en un sistema de recomendación de gráficas estadísticas que hubiera obligado a los usuarios a caracterizar, antes que nada, las variables en los conjuntos de datos. El paquete, sin embargo, se ha desarrollado con la estrategia de presentar diferentes abanicos de gráficas desde un primer momento, con el menor contratiempo para el usuario. Esta estrategia pasa por aprovechar al máximo la información implícita de los datos, esto es, la caracterización que las variables ya tienen incorporada en los conjuntos de datos de tipo `data.frame` del entorno de programación estadística **R**. Esta caracterización, a diferencia de la propuesta en el capítulo 4, es unidimensional, pero aún así, ayuda en la selección de una gráfica adecuada.

El paquete `brinton` presenta una serie de funciones útiles para el análisis gráfico exploratorio de datos. Las primeras tres funciones que implementa son `wideplot()`, `longplot()` y `plotup()`. La primera

permite explorar de manera gráfica la estructura de un conjunto de datos, la segunda presenta un catálogo exhaustivo de gráficas a partir de la selección de una variable de un conjunto de datos y la tercera representa una gráfica en particular a partir de una variable de un conjunto de datos. Estas tres funciones buscan ayudar al usuario en su búsqueda de una gráfica que represente características en los datos que resulten de su interés.

5.1. ANTECEDENTES

El análisis exploratorio de datos (EDA) es un enfoque del análisis de datos orientado a observar las características de un conjunto de datos, sin poner el acento en el modelado de los datos o el contraste de hipótesis preconcebidas. Este enfoque fue impulsado por Tukey (1977) (p.iv) quien declara *The greatest value of a picture is when it 'forces' us to see what we never expected to see*. Esta declaración pone el acento en la visualización de los datos y se alinea con la *expectation disconfirmation theory* (Oliver, 1977) que relaciona la satisfacción con la expectativa. El EDA, precisamente por no basarse en hipótesis o expectativas preestablecidas, premia con especial satisfacción a los usuarios cuando hallan aspectos inesperados en los conjuntos de datos. Para completar la tríada de referencias que trae a la mente el Cinturón de Orión, ese mismo año se alinea Bertin (1977, p.2), quien señala que el proceso de definir un problema o la correspondiente hipótesis no es automatizable y aquí radica el principal reto: automatizar la representación gráficas de datos de modo que los usuarios puedan examinarlos, plantear hipótesis y seleccionar la gráfica estadística apropiada que les permita responder satisfactoriamente a expectativas recién creadas.

La automatización de procedimientos orientada a facilitar el EDA recibe el nombre de autoEDA, (Staniak y Biecek, 2019). Si el EDA se lleva a cabo mediante la observación de gráficas, entonces se conoce como “análisis gráfico exploratorio de datos” (GEDA) y, si las gráficas

que hacen posible esta exploración se generan automáticamente, entonces podríamos hablar de “análisis gráfico exploratorio de datos automatizado” (autoGEDA). El paquete `brinton` que presenta este capítulo se enmarca dentro de las herramientas de autoGEDA, grupo al que también pertenecen herramientas como GGobi (Cook et al., 2007) o Mondrian (Theus y Urbanek, 2008) y más concretamente en el subgrupo de herramientas que interpretan qué gráfica a mostrar (*visual encoding recommendations*) referidos en el capítulo 3.

En el entorno de programación estadística R conviven dos sistemas gráficos (Friendly, 2018). Por un lado tenemos el sistema de gráficos básico de R que proporciona el paquete `graphics` con funciones de bajo nivel que controlan aspectos específicos de una gráfica, como por ejemplo `lines()`, `points()` o `legend()` que añaden segmentos de línea, puntos o la leyenda a una gráfica, y también funciones de alto nivel que producen gráficas completas, como por ejemplo `plot()` capaz de producir diferentes tipos de gráficas en función de sus argumentos, `pie()` que es una función específica para diagramas de sectores o `barplot()` que produce diagramas de barras. Por otro lado encontramos el sistema de gráficos `grid` que suplementa el sistema `graphics` con funciones de bajo nivel como `grid.lines()`, `grid.points()` o `legendGrob()` para añadir segmentos de línea, puntos o la leyenda a una gráfica. El sistema `grid` no incluye funciones de alto nivel sino que deja que sean otros desarrollos basados en éste los que implementen funciones de alto nivel como por ejemplo `ggplot()` o `xypplot()` de los paquetes `ggplot2` (Wickham, 2016) y `lattice` (Sarkar, 2008) respectivamente. El sistema `grid` se complementa, a su vez, con otros paquetes con funciones de bajo nivel como por ejemplo `gridExtra` (Auguie, 2017). El paquete `brinton` se sustenta en el sistema `grid`, `gridExtra` y `ggplot2` con funciones de más alto nivel que permiten producir múltiples gráficas a la vez.

En ambos sistemas `graphics` y `grid` y en los paquetes basados en éstos, se utilizan diferentes estrategias de las descritas en el capítulo

3 para producir gráficas estadísticas automáticamente. La estrategia funcional se observa por ejemplo en la función `plot()` que, si la aplicamos al conjunto de datos `cars`, produce un diagrama de dispersión dado que el objeto `cars` es de la clase `data.frame` e incluye dos variables numéricas. Si se aplica sobre el objeto `airmiles` produce en cambio un diagrama de línea porque es de clase `ts` (*time series*) y contiene un rango de años, una frecuencia de observación y una secuencia de números. El diseño según tarea (*task design*) lo observamos, por ejemplo en la función `ggsurvplot()` del paquete `survminer` (Therneau, 2015) que produce gráficas específicas utilizadas para el análisis de supervivencia. La estrategia basada en modelos de representación se observa por ejemplo en funciones básicas como `barplot()`, `hist()` o `pie()` que producen respectivamente una gráfica de barras, un histograma o diagrama de porciones respectivamente, o también en funciones como `geom_point()` o `geom_line()`, de `ggplot2` que reducen el abanico de gráficas posibles a las de implantación de tipo punto o línea respectivamente. Las características de la percepción humana también son tenidas en cuenta por las gráficas producidas por defecto mediante la función `ggplot()`. Aspectos como por ejemplo el color del fondo del área gráfica, el tamaño de los puntos de un diagrama de dispersión o el espaciado de las etiquetas en los ejes de coordenadas buscan facilitar la percepción de la gráfica siguiendo criterios como por ejemplo los de Carr (1994).

A pesar de la gran variedad de aproximaciones diferentes de automatización de gráficas estadísticas implementadas en R, existe un vacío que rellenar basado en funciones de más alto nivel que, en vez de mostrar una única gráfica, sugieran un abanico de gráficas a partir del cual ir acotando la gráfica que mejor responde a las necesidades del usuario o mejor satisface su recién creada expectativa. La presentación de múltiples gráficas que representen las relaciones entre unos mismos valores facilita la observación polifacética y con ella, la posibilidad de que emerjan preguntas que pueden ser respondidas mediante las

gráficas presentadas, algunas de esas gráficas ligeramente modificadas u otras gráficas de nueva creación. El paquete `brinton` viene a ocupar un hueco en este vacío y añadir una alternativa para el análisis gráfico exploratorio de datos.

5.2. EL PANORAMA DE AUTOGEDA EN R

Entre los más de 18.000 paquetes disponibles en el CRAN en octubre de 2022, encontramos tan solo unos 40 que incluyen la palabra *exploratory* en el título y unos 125 que la contienen en su descripción. Sólo una pequeña porción de estos paquetes tienen una orientación eminentemente gráfica, siendo lo más habitual encontrar paquetes que combinan resultados textuales o tabulados con gráficas auxiliares. A continuación se agrupan someramente diferentes paquetes de autoEDA según el tipo de soluciones que promueve, aunque un mismo paquete suele contener funciones con orientaciones también diversas.

Por un lado tenemos paquetes que presentan *structure plots* que son gráficas que concentran en una imagen la información contenida en todo un conjunto de datos (Unwin et al., 2006, p. 51). En este grupo encontramos paquetes como `tabplot` (Tennekes et al., 2013), que ha sido recientemente archivado, cuya función `tableplot()` produce *tableplots*, una versión estática (ver figura 5.2) de *Table Lens* (Rao y Card, 1994) que es un tipo de visualización pionero para extraer información de tablas con numerosas filas y columnas (ver figura 5.1), o funciones como `visdat::vis_dat()` (Tierney, 2017) que produce una versión también estática y aún más simplificada de *Table Lens* (ver figura 5.3). Por otro lado encontramos funciones como `inspectdf::show_plot()` (Rushworth, 2019) que puede generar, en función de su argumento, diferentes gráficas como por ejemplo la gráfica de proporciones *spine plot* de la figura 5.4, que a diferencia del *table lens*, incluye sólo las variables categóricas.

Otro tipo de funciones que encontramos en los paquetes de autoEDA en R, agrupan las variables según sea su clase, y también

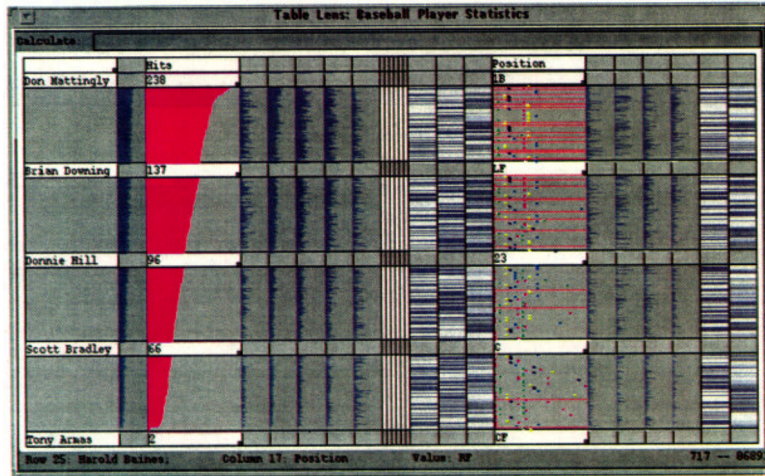


Figura 5.1: Captura de pantalla de una gráfica dinámica Table Lens que muestra la estructura de un conjunto de datos estadísticos de jugadores de béisbol así como detalles de 5 jugadores en particular. Fuente: Rao y Card (1994).

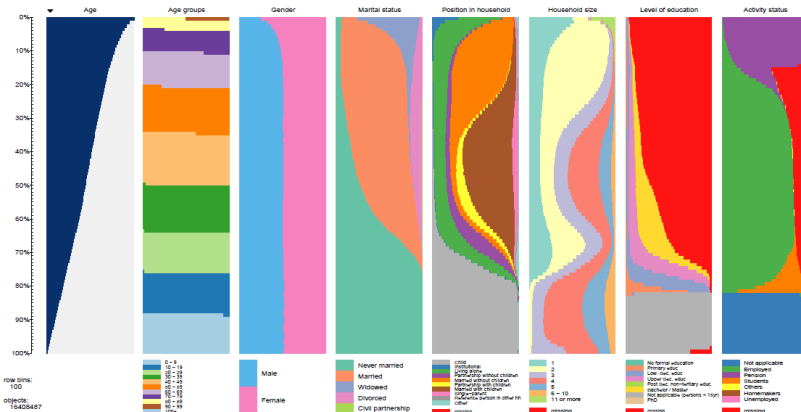


Figura 5.2: Gráfica Tableplot que muestra la estructura de un conjunto de datos de prueba de integración de datos censales. Fuente: Tennekes et al. (2013).

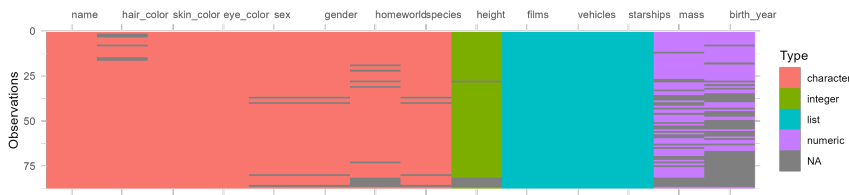


Figura 5.3: Gráfica Tableplot simplificada producida con el paquete `visdat` que muestra el conjunto de datos `starwars` del paquete `dplyr`. Función: `visdat::vis_dat(starwars)`. Fuente: Elaboración propia.

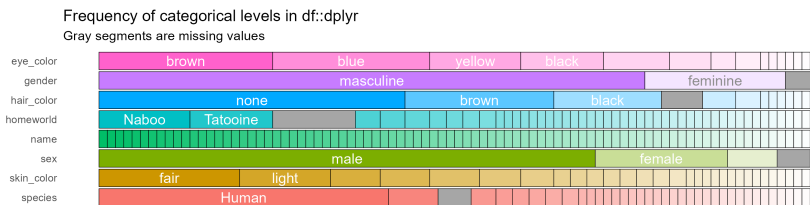


Figura 5.4: Gráfica de proporciones *spine plot* producida con el paquete `inspectdf` que muestra el conjunto de datos `starwars` del paquete `dplyr`. Función: `show_plot(inspect_cat(starwars))`. Fuente: Elaboración propia.

muestran diferentes estadísticos y/o gráficas que representan cada variable según sea su clase. Los hay, incluso, que presentan gráficas que relacionan diferentes variables, también según sea su clase. En este grupo de funciones encontramos, por ejemplo, `xray::distributions()` (Seibelt, 2017) que produce una tabla y tantas gráficas como variables tiene un conjunto de datos, `DataExplorer::create_report()` (Cui, 2019) que genera un informe que contiene tablas, una selección adhoc de gráficas de variables por separado y otras gráficas para su exploración, la función `SmartEDA::ExpReport()` (Dayanand Ubrangala et al., 2019) que produce también un informe con una selección de gráficas adhoc o la función `ExPanDar::ExPanD()` (Gassen, 2020) que produce también un informe, pero esta vez, interactivo.

Otro grupo de funciones como `dataMaid::makeDataReport()`

(Petersen y Ekstrøm, 2019) o `summarytools::dfSummary()` (Comtois, 2019) ofrecen otra manera de explorar las variables. Estas funciones generan informes que incluyen estadísticos de cada una de las variables acompañados de una gráfica que las describe, generalmente un histograma si la variable es numérica y un diagrama de barras si es categórica. Por otro lado existen multitud de paquetes con diferentes funciones que proporcionan una gráfica específica para un cometido concreto, este es el caso por ejemplo de las funciones `dlookr::plot_qq_numeric()` (Ryu, 2021) para explorar si alguna de las variables numéricas de un conjunto de datos sigue una distribución normal, `explore::explain_tree()` (Krasser, 2021) que produce un árbol de decisión para clasificar según una variable objetivo o `explore::explore_density()` que produce un diagrama de densidad por grupos (ver figura 5.5).

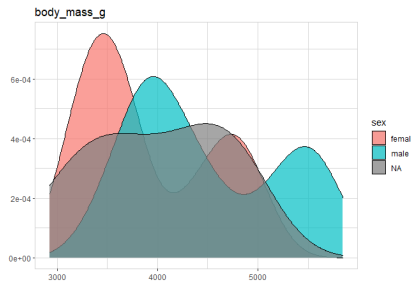


Figura 5.5: Diagramas de densidad por grupos producido con el paquete `explore` respecto de la variable `male` del conjunto de datos `penguins` del paquete `palmerpenguins`. Función: `explore_density(penguins , body_mass_g, target = sex)`. Fuente: Elaboración propia.

Los paquetes de autoEDA suelen tener una doble representación tabulada y gráfica de los datos y cada vez es más frecuente que incluyan funciones para generar automáticamente informes que describen las variables por separado y algunas relaciones entre las variables mediante análisis de correlaciones o de componentes principales. Son pocos los paquetes que incluyen funciones para representar *structure*

plots que muestren la estructura de un conjunto de datos en una única gráfica unipanel o multipanel. En cambio, es común encontrar funciones que producen gráficas unipanel o multipanel a partir de variables de un determinado tipo de un conjunto de datos, como la de la figura 5.6 producida por la función `DataExplorer::plot_histogram()` que presenta histogramas de todas las variables numéricas de un conjunto de datos. También es común encontrar gráficas que combinan las variables por pares, generalmente de un mismo tipo como por ejemplo el “embudo de correlaciones” de la figura 5.7.

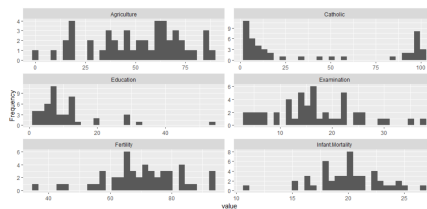


Figura 5.6: Gráfica estructural producida con el paquete `DataExplorer` que muestra las distribuciones de las variables numéricas del conjunto de datos `swiss`. Función: `plot_histogram(swiss, ncol = 2L)`. Fuente: Elaboración propia.

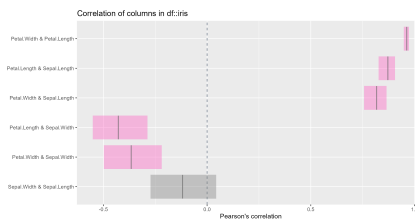


Figura 5.7: Embudo de correlaciones producido con el paquete `inspectdf` que muestra las correlaciones entre pares de variables del conjunto de datos `iris` ordenadas en el eje Y según su valor absoluto. El color codifica la superación de un cierto umbral del p-valor (por defecto 0.5) y la amplitud de las bandas el intervalo de confianza de cada correlación entre pares de variables. Función: `show_plot(inspect_cor(iris))`. Fuente: Elaboración propia.

A pesar de la gran utilidad de los paquetes existentes, éstos suelen

presentar un número limitado de tipos de gráficas que, por otro lado, son muy comunes. Echamos de menos soluciones que presenten, en una única función de más alto nivel, un mayor abanico de gráficas en el que poder explorar las variables de un conjunto de datos.

El abanico de gráficas puede presentarse, esencialmente, en dos formatos: una posibilidad es mediante un informe como los que presentan los paquetes `dataMaid` y `summarytools` que listan las variables analizadas y combinan texto, tablas y gráficas; otra posibilidad es mediante gráficas multipanel, ya introducidas en la sección 2.3 que facilitan la comparación entre los valores de las diferentes gráficas, especialmente si los ejes colindantes dan soporte a la misma variable y comparten la escala. Dado que el paquete `brinton` que se presenta más adelante, saca ventaja de las gráficas multipanel, a continuación hacemos un repaso escueto de diferentes implementaciones en **R** para generar gráficas multipanel.

5.3. GRÁFICAS MULTIPANEL EN R

Un primer grupo de gráficas multipanel lo representan los cuadros de mando (o *dashboards*) que en el entorno de programación **R** se pueden generar mediante paquetes específicos tales como `shinydashboard` (Chang y Borges Ribeiro, 2018) o `flexdashboard` (Iannone et al., 2018). Esta agrupación de gráficas de tipo, origen y tamaño diverso, también se puede conseguir, aunque sin capacidades interactivas, con otras funciones como por ejemplo `layout()` de *base graphics* (ver la figura 5.8) u otras de *grid graphics* como `cowplot::plot_grid()` (Wilke, 2020), `egg::grid.arrange()` (Auguie, 2019) o `patchwork` (Pedersen, 2020).

Un segundo grupo de gráficas multipanel lo conforman las gráficas condicionadas (*conditioning plots* o *coplots*). Este tipo de gráficas multipanel se pueden generar mediante funciones concretas de `ggplot2` (Wickham, 2016) tales como `ggplot2::facet_wrap()` o `ggplot2::facet_grid()` y también mediante el paquete `lattice`

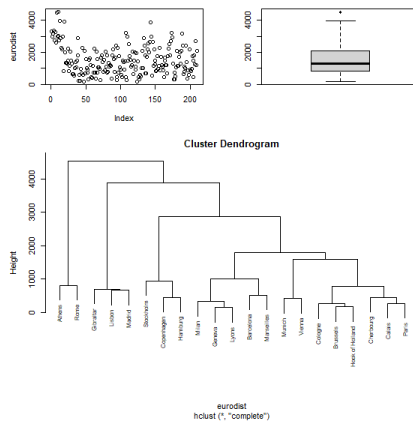


Figura 5.8: Cuadro de mando simple producido con la función `layout()`, el conjunto de datos `eurodist` y diferentes funciones de *base graphics* para generar las gráficas de cada panel. Fuente: Elaboración propia.

(Sarkar, 2008) que es más específico para este tipo de gráficas multipanel y que genera gráficas como la de la figura 5.9.

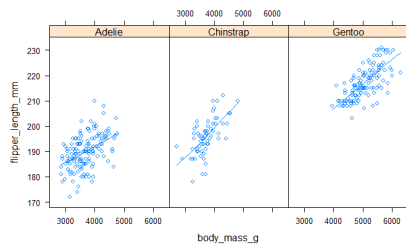


Figura 5.9: Gráficas multipanel condicionada producida con el paquete `lattice` y el conjunto de datos `penguins`. Función: `xyplot(flipper_length_mm ~ body_mass_g | species, data = penguins)`. Fuente: Elaboración propia.

Otro grupo de gráficas multipanel es el conocido como diagramas de pares, *matrix of plots* o *pairs plot*. En el entorno de R se pueden producir diagramas de pares, por ejemplo mediante la función de *base graphics* `pairs()` o en base al sistema de gráficos *grid*, mediante la función `gpairs::gpairs()` que utiliza el paquete `lattice`

(ver gráfica 5.10).

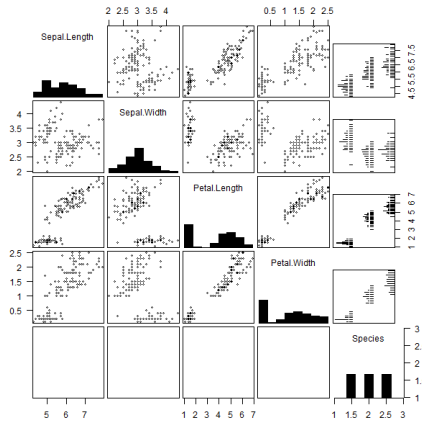


Figura 5.10: Gráfica generalizada de pares producida con el paquete `gpairs` y el conjunto de datos `iris`. Función: `gpairs(iris)`. Fuente: Elaboración propia.

En el entorno de programación **R** no hemos encontrado paquetes específicos que permitan la producción de *SpreadPlots* que cuentan con múltiples paneles vinculados que muestran diferentes aspectos de un mismo conjunto de datos. Es posible, sin embargo producir estas gráficas mediante paquetes genéricos para la creación de páginas web interactivas como **Shiny**.

5.4. EL PAQUETE BRINTON

La librería `brinton` la hemos creado para facilitar el análisis exploratorio de datos siguiendo el mantra de búsqueda de información visual *Overview first, zoom and filter, then details on demand* (Shneiderman, 1996). La idea principal es acompañar al usuario en estas tres fases mediante tres funciones (`wideplot()`, `longplot()` y `plotup()`) cada una de ellas con argumentos y utilidad bien diferenciada pero todas orientadas a facilitar el análisis exploratorio de datos y la selección de una gráfica adecuada.

La función `wideplot()` está pensada para explorar un conjunto de datos mediante la presentación de una matriz de gráficas en la que cada variable se encuentra representada mediante múltiples gráficas. Una vez observado el conjunto, es posible explorar otras gráficas para una variable determinada. Para esto se ha diseñado la función `longplot()` que también presenta una matriz de gráficas pero a diferencia del caso anterior, en vez de mostrar una selección de gráficas para cada variable, presenta todo el abanico de gráficas disponibles en la librería para representar una única variable. Una vez que fijamos nuestra atención en una determinada gráfica, podemos utilizar la función `plotup()` que presenta los valores de una variable en una única gráfica. Tenemos la posibilidad también de recuperar el código de la gráfica resultante para adaptarla a unas necesidades particulares. Estas tres funciones vienen a aumentar la riqueza de tipos de gráficas que son presentadas automáticamente por las librerías de *autoGEDA* en el entorno **R**.

La librería `brinton` se apoya esencialmente en la gramática de las gráficas introducida por Wilkinson (2005) e implementada en **R** por la librería `ggplot2`. También por la librería `gridExtra` (Aguie, 2017) para la composición de las gráficas multipanel y en `rmarkdown` (Xie et al., 2018) para componer dinámicamente los resultados.

En el contexto de las librerías gráficas de **R** basadas en el sistema `grid`, la librería `lattice` permite elaborar un abanico de gráficas limitado a unos trece tipos distintos pero adaptables hasta un nivel de detalle muy sofisticado. Por otra parte, la librería `ggplot2` permite elaborar hasta el más mínimo detalle de una gráfica pero a costa de tener que aprender su gramática y su sistema por capas. La librería `brinton` se orienta, en cambio, a facilitar la selección de gráficas estadísticas, presentando a los usuarios un amplio abanico de gráficas posibles y permitiendo que éstos las puedan seleccionar de manera nominal y, si se conoce la gramática de `ggplot2`, adaptar a sus necesidades. Para crear una gráfica estadística en **R**, si ésta se

encuentra implementada en la librería **brinton**, ya no se requiere más que especificar la fuente de los datos y el tipo de gráfica a producir.

La instalación de la librería se realiza fácilmente desde el *Comprehensive R Archive Network* (CRAN) mediante la consola de **R**:

```
install.packages("brinton")
library(brinton)
```

Al cargar en memoria la librería, ésta presenta un mensaje de bienvenida que pretende rendir un homenaje a la entusiasta introducción de Henry D. Hubbard del libro *Graphic Presentation* (Brinton, 1939):

```
## M a G i C I N G R a P H S
```

La función wideplot

En el momento de cargar un conjunto de datos en **R** la siguiente función a utilizar suele ser `str()`. Esto se debe a que, si no lo determinamos de manera explícita, las funciones para cargar en memoria conjuntos de datos hacen suposiciones acerca de la naturaleza de las variables. La función `str()` muestra en la consola el tipo de objeto sobre el que se aplica la función, el número de filas, el número de columnas, el nombre de éstas, su clase (esto se refiere a si es numérica, factor, etc. . .) y las primeras observaciones de cada una de las variables. La función `wideplot()` se inspira en esta función, pero en vez de describir el conjunto de datos de manera textual o tabulada lo hace de manera gráfica. Es fácil comparar los resultados entre estas dos funciones, por ejemplo, con el conjunto de datos *esoph* que recoge datos de un estudio de cáncer de esófago en Ille-et-Vilaine, Francia, y que cuenta con tres variables de tipo factor ordenado y dos numéricas:

```
str(esoph)
```

```
'data.frame': 88 obs. of 5 variables:
 agegp: Ord.factor w/ 6 levels "25-34"<"35-44"<...: 1 1 1 1 1 1 1 1 1 1 ...
 alcgp: Ord.factor w/ 4 levels "0-39g/day"<"40-79"<...: 1 1 1 1 2 2 2 2 3 3 ...
 tobgp: Ord.factor w/ 4 levels "0-9g/day"<"10-19"<...: 1 2 3 4 1 2 3 4 1 2 ...
 ncases: num 0 0 0 0 0 0 0 0 0 0 ...
 ncontrols: num 40 10 6 5 27 7 4 7 2 1 ...
```

```
wideplot(data = esoph)
```

```
wideplot graphic
```

```
by brinton R package
```

```
## # esoph dataframe
```

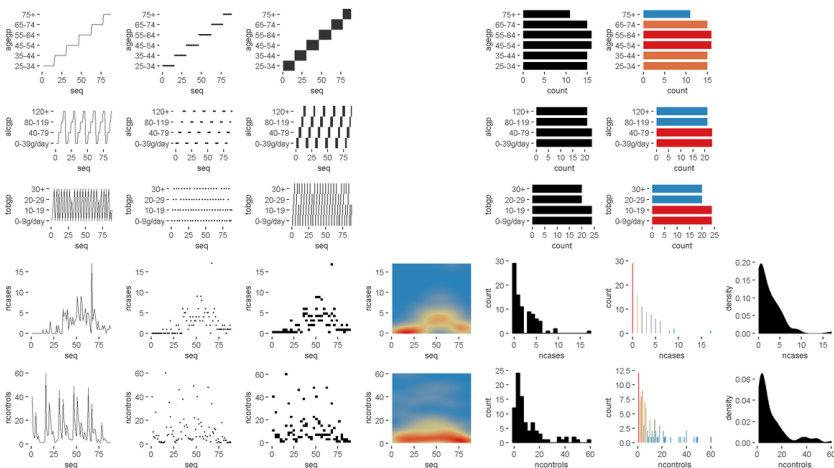


Figura 5.11: Gráfica wideplot. Función: `wideplot(esoph)`. Fuente: Elaboración propia.

La función `wideplot()` devuelve un resumen gráfico (ver figura 5.11) de las variables incluidas en el conjunto de datos sobre el cual se aplica. Primero agrupa las variables según esta secuencia: `logical`, `ordered`, `factor`, `character`, `datetime`, `numeric`. Luego crea una gráfica multipanel en formato `html` en la que cada variable del conjunto de datos se representa en cada fila de la matriz mientras que en cada columna se representan diferentes gráficas de esa misma variable. El tipo de gráfica resultante recibe el nombre de *wideplot*

porque muestra un abanico de gráficas para todas las columnas del conjunto de datos. La estructura de la función, los argumentos que admite y los valores por defecto son como siguen:

```
## wideplot(data, dataclass = NULL, logical = NULL,  
##   ordered = NULL, factor = NULL, character = NULL,  
##   datetime = NULL, numeric = NULL, group = NULL, ncol = 7,  
##   label = 'FALSE')
```

El único argumento necesario para obtener un resultado es `data` que espera un objeto de clase `data-frame`; `ncol` filtra las n primeras columnas de la matriz, entre 3 y 7, que se mostrarán. Cuanto menor es el número de columnas a mostrar, mayor es el tamaño de las gráficas lo que resulta especialmente interesante si las etiquetas de las escalas empequeñecen el área gráfica; `label` añade a la matriz un vector debajo de cada grupo de filas según el tipo de variable con los nombres y el orden de las gráficas representadas; `logical`, `ordered`, `factor`, `character`, `datetime` y `numeric` permiten determinar qué gráficas y en qué orden debe mostrar la matriz para cada tipo de variable en particular. Finalmente, `group` cambia la selección de las gráficas que se muestran por defecto según los criterios de la tabla 5.1.

Si no se especifican el orden y los tipos de gráfica a mostrar para cada tipo de variable, y tampoco se filtran los tipos de gráficas a mostrar mediante el argumento `group`, entonces, la gráfica que se muestra por defecto contiene una selección opinada de gráficas para cada tipo de variable, organizadas espacialmente para facilitar la comparación entre gráficas de una misma fila y también entre las de una misma columna. Cada usuario puede sobrescribir esta selección opinada de gráficas según su propia preferencia, utilidad o necesidad, mediante los argumentos `logical`, `ordered`, `factor`, `character`, `datetime` y `numeric`.

group	Tipo de gráficas
sequence	que incluyen la secuencia en la que se observan los valores de manera que un eje desarrolla esta secuencia. p.e. <i>*line graph*</i> o el <i>*point-to-point graph*</i> .
scatter	cuyas marcas representan observaciones individuales p.e. <i>*point graph*</i> o el <i>*stripe graph*</i> .
bin	cuyas marcas representan observaciones agregadas según intervalos de clase. p.e. <i>*histogram*</i> o el <i>*bar graph*</i> .
model	que representan modelos a partir de las observaciones. p.e. <i>*density plot*</i> o el <i>*violin plot*</i> .
symbol	cuyas marcas se componen de símbolos complejos y no solo puntos, líneas o áreas. p.e. <i>*box plot*</i> .
GOF	que representan la bondad de ajuste (<i>*goodness of fit*</i>) de unos valores respecto a un modelo. p.e. <i>*qq plot*</i> .
random	seleccionadas de manera aleatoria.

Cuadro 5.1: Valores posibles del argumento **group** de la función `wideplot()`. Fuente: Elaboración propia.

La función longplot

Por una cuestión de economía de cálculo la función `wideplot()` tiene limitado el número de gráficas que puede presentar en cada fila. Si se quiere ampliar el abanico de gráficas sugeridas para una variable determinada, tenemos que utilizar la función `longplot()` que devuelve una matriz con todas las gráficas consideradas por la librería (ver figura 5.12) para esa variable en particular. La estructura de la función es muy sencilla `longplot(data, vars, label = TRUE)` y es fácil comprobar el resultado de aplicar esta función sobre la variable `alcgp` del conjunto de datos `esoph`:

```
longplot(data = esoph, vars = "alcgp")
```

El tipo de gráfica resultante recibe el nombre de *longplot* porque muestra todo el abanico de gráficas disponibles para representar las relaciones entre los valores de una selección limitada de variables (aunque en esta librería, por ahora, solo se han incluido gráficas para una única variable).

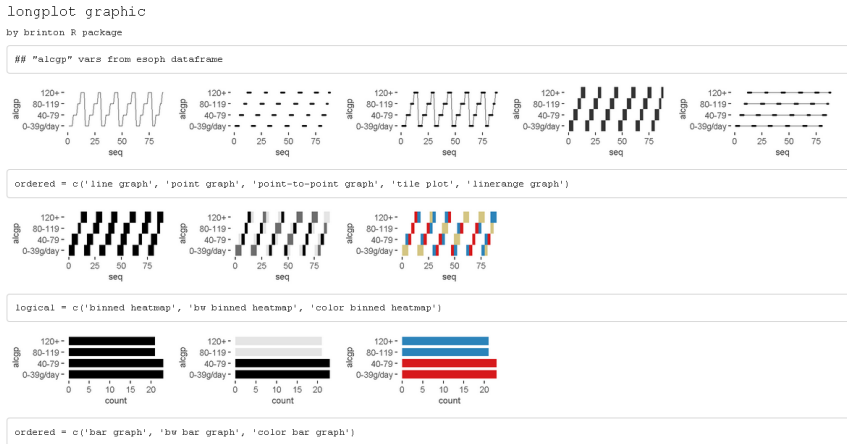


Figura 5.12: Gráfica longplot. Función: `longplot(data = esoph, vars = 'alcgp')`. Fuente: Elaboración propia.

```
## longplot(data, vars, label = 'TRUE')
```

Los argumentos de la función son: `data` que necesariamente debe ser un objeto de clase `data-frame`; `vars` que requiere el nombre de una variable específica del conjunto de datos y `label` que no es obligado definir y que añade un vector debajo de cada fila de la matriz con indicación del nombre de cada gráfica. A diferencia de la matriz de la función `wideplot()`, la función `longplot()` no incluye parámetros para limitar el abanico de gráficas a presentar porque consideramos que la principal utilidad de esta función es precisamente la de presentar todas las posibles representaciones gráficas disponibles para una variable en particular. No descartamos, sin embargo, añadir filtros que limiten el número de gráficas a mostrar si al ir ampliando el catálogo de gráficas se demuestra conveniente. Cada una de las gráficas que presenta pueden ser llamadas de manera explícita por su nombre por las funciones `wideplot()` y `plotup()`, razón por la cual el argumento `label` se ha predeterminado como `TRUE` en este caso.

El abanico de gráficas que devuelve la función `longplot()` se encuentra ordenado de modo que en las filas se encuentran básicamente

diferentes tipos de gráficas y en las columnas diferentes variaciones de un mismo tipo de gráfica. Este orden, sin embargo, no es inviolable y en algunos casos para compactar el resultado, en las columnas de una misma fila se encuentran diferentes tipos de gráficas.

La función plotup

La función `plotup()` tiene la siguiente estructura `plotup(data, vars, diagram, output = "plots pane")`. Esta función devuelve, por defecto, un objeto de clase `gg` y `ggplot2` que se muestra graficado en el *plots pane* de RStudio. El objeto graficado es una única gráfica a partir de una variable de un conjunto de datos particular y el nombre de la gráfica deseada, de entre los nombres que incluye el espécimen que presentamos en la siguiente subsección. Es fácil comprobar el resultado (ver figura 5.13) de aplicar esta función para producir un diagrama de línea a partir de la variable `ncases` del conjunto de datos `esoph`:

```
plotup(data = esoph,  
       vars = "ncases",  
       diagram = "line graph",  
       output = "html")
```

Esta función requiere 3 argumentos: `data`, `vars` y `diagram`. El cuarto argumento opcional `output` tiene establecido por defecto el valor `plots pane` pero que si se iguala a `html`, genera la gráfica en una página `html` y si se iguala a `console`, la función devuelve el código utilizado por la librería para generar esta precisa gráfica. Esto es especialmente útil para adaptar la gráfica producida por defecto a las necesidades o gustos particulares de los usuarios.

```
plotup(data = esoph,  
       vars = "ncases",  
       diagram = "line graph",  
       output = "console")
```

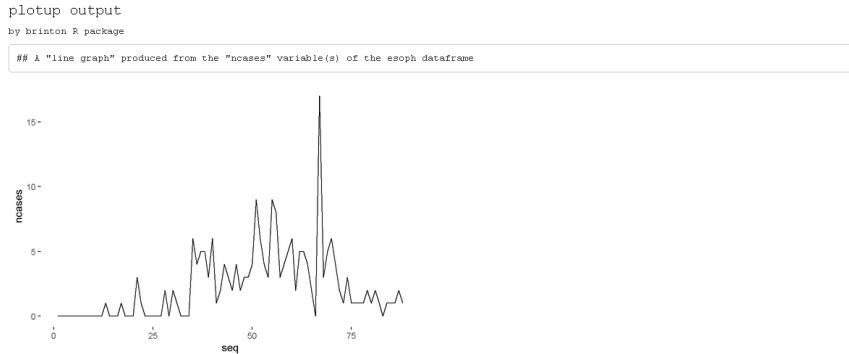


Figura 5.13: Gráfica de línea. Función: `plotup(data = esoph, vars = 'ncases', diagram = 'line graph')`. Fuente: Elaboración propia.

```
##
## ggplot(esoph, aes(x=seq_along(ncases), y=ncases)) +
##   geom_line() +
##   labs(x='seq') +
##   theme_minimal() +
##   theme(panel.grid = element_line(colour = NA),
##         axis.ticks = element_line(color = 'black'))
```

El espécimen

La documentación de la librería incluye la *vignette* “1v specimen” que contiene un espécimen con imágenes de todos los tipos de gráficas de una sola variable, incorporadas en la librería según sea el tipo de la variable. Estas gráficas sirven a modo de ejemplo para que los usuarios puedan comprobar rápidamente si una gráfica está incorporada, el tipo o tipos de variable para los que ha sido incorporada y la etiqueta con la que se ha identificado. Cabe decir que la idoneidad de una gráfica en particular, se tiene que contrastar con los conjuntos de datos de interés y variables de cada usuario en particular. Este espécimen, en su versión actual, ha sido incorporado a este artículo como material suplementario.

GRADOS GRÁFICOS DE LIBERTAD La utilidad de esta librería se basa en el hecho de que diferentes representaciones gráficas de unos mismos datos permiten, no solo observar diferentes características de estos datos, sino además evidenciar de manera más o menos efectiva una determinada característica. Es por este motivo por el que las gráficas consideradas por esta librería gozan de un número elevado de grados gráficos de libertad, que permite incluir en su catálogo tanto gráficas comúnmente utilizadas como gráficas todavía por desarrollar. El concepto de *graphical degrees of freedom* ya ha sido utilizado por Bengler y Hege (2006) para referirse a las variables visuales de Bertin (p.43, 1967) pero con alguna modificación. Aquí utilizamos este concepto como detallamos a continuación.

- Tipo de gráfica.** El principal grado de libertad del catálogo de gráficas es el tipo de gráfica. Los diferentes tipos de gráfica no se refieren exclusivamente a tipos muy alejados entre ellos, al contrario, existen gráficas muy similares que coexisten porque un elevado número de usuarios prefiere cada uno de los tipos. Este es el caso, por ejemplo, del *density plot* y el *violin plot* representados en la figura 5.14.

```
wideplot(data = esoph[5],
         numeric = c('filled violin plot', 'filled density plot'))
```

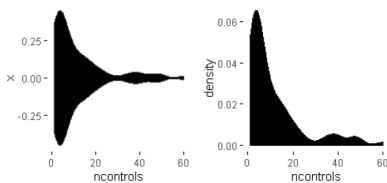


Figura 5.14: Primer grado gráfico de libertad: tipo de gráfica.
Fuente: Elaboración propia.

- Escala cromática.** Una misma gráfica puede tener diferentes versiones según la escala cromática asociada a una variable en los datos o transformada a partir de éstos. Un ejemplo de esto mismo se puede observar en la siguiente figura 5.15. A pesar de que el color podría llegar a descomponerse en tres variables visuales tales como el tono, la saturación y la luminosidad, para los propósitos de esta librería hemos considerado únicamente el tono en el caso de la escala de color y la luminosidad en el caso de la escala de grises siguiendo la clasificación de las variables visuales de Bertin (p.43, 1967).

```
wideplot(data = esoph[5],
          numeric = c('histogram', 'bw histogram', 'color histogram'))
```

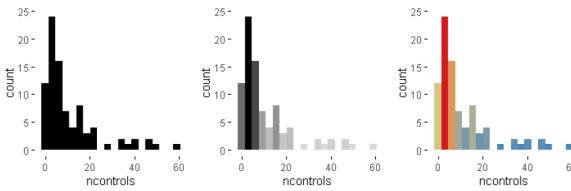


Figura 5.15: Segundo grado gráfico de libertad: escala cromática.
Fuente: Elaboración propia.

- Modo de agregación: diseminado o tamizado.** Unos mismos valores se pueden representar de modo que cada marca represente una valores únicos o bien de modo agregado. Un ejemplo de esto mismo se puede observar en la siguiente figura 5.16.

```
wideplot(data = esoph[5],
          numeric = c('stripe graph', 'binned stripe graph', 'bar graph',
                    'histogram'))
```

- Paneles anidados.** Una posibilidad, aunque poco explorada, es la de subdividir en diferentes paneles las celdas de gráficas

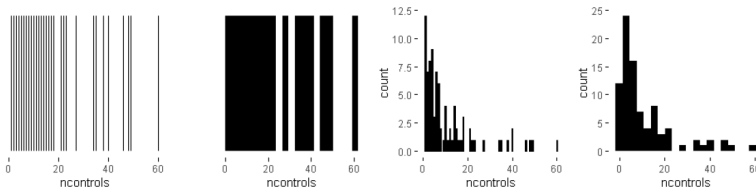


Figura 5.16: Tercer grado gráfico de libertad: modo de agregación.
Fuente: Elaboración propia.

multipanel, a modo de sistemas de coordenadas dentro de sistemas de coordenadas. Una solución parecida a la utilizada en los *treemap*. En el ejemplo de la figura 5.17, la gráfica de la derecha tiene tres paneles que pueden substituir las tres primeras gráficas.

```
wideplot(data = esoph[5],
         numeric = c('violin plot', 'stripe graph', 'box plot', '3 uniaxial'))
```

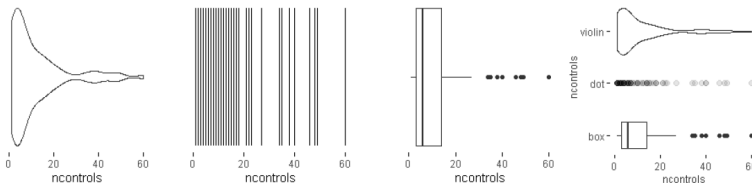


Figura 5.17: Cuarto grado gráfico de libertad: paneles anidados.
Fuente: Elaboración propia.

- Forma.** Una misma información puede representarse con marcas de diferentes formas. Esta posibilidad se ejemplifica en la figura 5.18, que compara dos gráficas con una construcción similar pero una con marcas en forma de circunferencia y la otra en forma cuadrada.

```
wideplot(data = esoph[5],
         numeric = c('color binned point graph', 'color binned heatmap'))
```

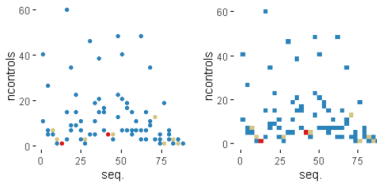


Figura 5.18: Quinto grado gráfico de libertad: forma. Fuente: Elaboración propia.

- **Implantación.** Unos mismos valores puede representarse con marcas con diferente tipo de implantación como punto, línea, área o una combinación de éstos. Un ejemplo se muestra en la figura 5.19, que compara las gráficas de punto, de línea, y de puntos conectados por una línea.

```
wideplot(data = esoph[5],
         numeric = c('point graph', 'line graph', 'point-to-point graph'))
```

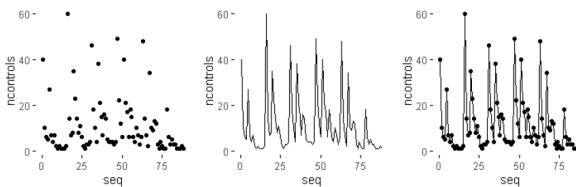


Figura 5.19: Sexto grado gráfico de libertad: implantación. Fuente: Elaboración propia.

- **Transición.** La transición o itinerario entre dos puntos puede ayuda a reflejar el carácter discreto de los cambios en los valores observados. La figura 5.20 compara dos gráficas de línea con diferente transición entre puntos.


```
wideplot(data = esoph[5],
         numeric = c('line graph', 'stepped line graph'))
```

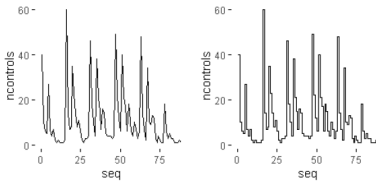


Figura 5.20: Séptimo grado gráfico de libertad: transición. Fuente: Elaboración propia.

- Orden.** Los valores de variables, especialmente aquellos que no guardan una relación de orden, son susceptibles de ser ordenados según diferentes criterios. Esta librería, como se puede ver en la figura 5.21, considera tres: según el orden de aparición en la secuencia de observaciones, la frecuencia con la que se observan los valores y el orden alfabético.

```
wideplot(data = data.frame("Region" = state.region),
         factor = c('tile plot', 'freq. reordered tile plot',
                   'alphab. reordered tile plot'))
```

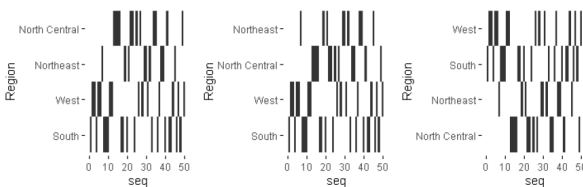


Figura 5.21: Octavo grado gráfico de libertad: orden. Fuente: Elaboración propia.

- Superposición.** Un último grado de libertad considerado es la posibilidad de incluir gráficas que superponen marcas cuya

fuente de los datos es la misma pero con diferentes grados de transformación (ver figura 5.22).

```
wideplot(data = esoph[5],
         numeric = c('point graph', 'point graph with trend line'))
```

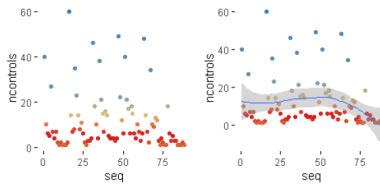


Figura 5.22: Noveno grado gráfico de libertad: superposición. Fuente: Elaboración propia.

Después de especificar los grados gráficos de libertad de que goza el espécimen de gráficas de la que se nutre *brinton*, queremos dejar constancia de que para la construcción del espécimen hemos descartados grados de libertad, tales como por ejemplo, el grupo de imposición (Bertin, 1967, p.52) o la permutación de variables espaciales (Bertin, 1967, p.43). Dicho de otro modo, la librería *brinton* presenta únicamente diagramas y no presenta redes ni mapas, así como tampoco muestra alternativas cuya única diferencia sea el intercambio del eje x por el y .

Aplicación en conjuntos de datos

La principal aplicación de una librería para el análisis exploratorio de datos no puede ser otra que la de facilitar la comprensión de los datos. Esta comprensión incluye la descripción del número y naturaleza de las variables, el número de observaciones y ejemplos de éstas – esto es precisamente lo que hace la función `str()` –. Incluye también evaluar la validez o calidad de los datos y las propiedades de los valores hallados.

En el caso que nos ocupa, el número de variables se deduce del número de filas de la matriz de la gráfica *wideplot*. Los nombres de las variables se encuentran en cada una de las gráficas con que cuenta por ahora el catálogo. La naturaleza – referida a la escala de medida – de las variables se puede conocer observando el abanico de gráficas seleccionadas o especificando el valor `label = TRUE` para las matrices de gráficas *wideplot* o *longplot*. En cuanto al número de observaciones, se deduce al observar las gráficas que incluyen la secuencia de las observaciones o, en el caso de variables categóricas, mediante el recuento de categorías y número de observaciones de cada una. Las gráficas *wideplot*, a diferencia del sumario textual de la función `str()`, no sólo muestran ejemplos de las primeras observaciones sino que las muestran todas. Para evaluar la validez de los datos se pueden observar gráficas específicas que facilitan la identificación de valores atípicos, valores perdidos o la discontinuidad en las observaciones. Lo mismo ocurre con las propiedades de los valores hallados, hay un enorme abanico de gráficas, cada una de las cuales permite con mayor o menor grado, resaltar diferentes propiedades. A continuación enumeramos una serie de tareas para las que las funciones incluidas en *brinton* han resultado útiles y mostramos el proceso que hemos llevado a cabo para completarlas.

IDENTIFICAR VARIABLES QUE ORDENAN EL CONJUNTO Este primer ejemplo muestra como utilizar la función `wideplot()` para determinar si las observaciones del conjunto de datos `aids` de la librería `KMsurv` se encuentran o no ordenadas según alguna de las variables. Este conjunto de datos cuenta con tres variables, `infect` (*infection time for AIDS in years*), `induct` (*induction time for AIDS in years*) y `adult` (*indicator of adult (1=adult, 0=child)*). Para llevar a cabo la tarea, primero instalamos la librería, luego la cargamos en memoria y ejecutamos la función `wideplot` con su salida por defecto.

```
install.packages("KMsurv")
data(aids, package = "KMsurv")
wideplot(data = aids, label = TRUE)
```

wideplot graphic

by brinton R package

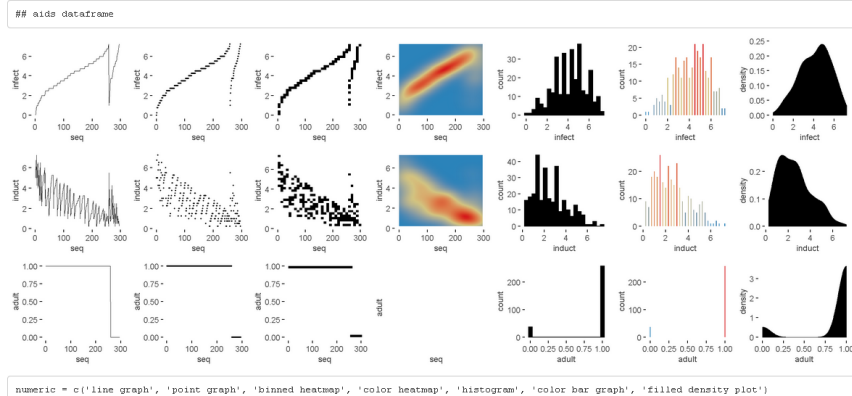


Figura 5.23: Gráfica wideplot con etiquetas. Función: `wideplot(data = aids, label = T)`. Fuente: Elaboración propia.

A partir de este primer resultado observamos que la gráfica de línea es la que mejor evidencia que el conjunto de datos se encuentra ordenado primero por la variable `adult` y luego por la variable `infect`. Para acabar de afinar la selección de la gráfica más adecuada podemos entonces ejecutar la misma función pero limitando los tipos de gráfica a mostrar a dos variaciones de gráficas de línea y limitando el número de columnas, por ejemplo, a 5 para que las gráficas resulten de mayor tamaño.

```
wideplot(data = aids,
         numeric = c('line graph', 'stepped line graph'),
         ncol = 5)
```

El resultado son dos variaciones de la gráfica de línea para cada una de las tres variables en las cuales se observa claramente que el conjunto de datos se encuentra ordenado primero por la variables

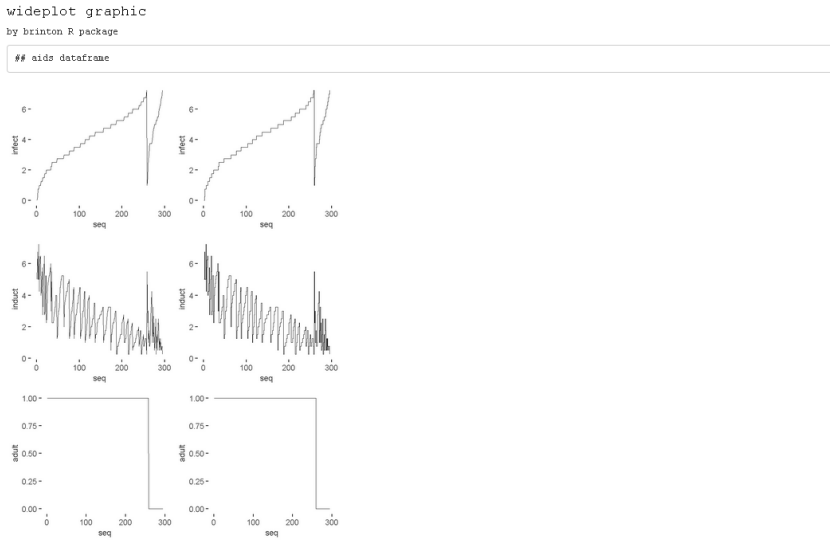


Figura 5.24: Gráfica wideplot con tipos de gráficas específicos. Función `wideplot(data = aids, numeric = c('line graph', 'stepped line graph'))`, `ncol = 5`). Fuente: Elaboración propia.

`adult` y luego por la variable `infect`. En este caso, puede haber argumentos tan válidos para utilizar las gráficas de la primera como para las de la segunda columna.

Este mismo ejemplo sirve también para el caso de conjuntos de datos con variables categóricas como es el caso del conjunto de datos `MentalHealth` de la librería `Stat2Data`. Este conjunto de datos se compone de tres variables: `Month` – mes del año –, `Moon` – relación con la luna llena: `After`, `Before`, or `During` – y `Admission` – número de ingresos en urgencias –, las dos primeras categóricas y la última numérica.

```
install.packages("Stat2Data")
data(MentalHealth, package = "Stat2Data")
wideplot(data = MentalHealth, label = TRUE)
```

Para este otro caso con variables categóricas, si observamos las

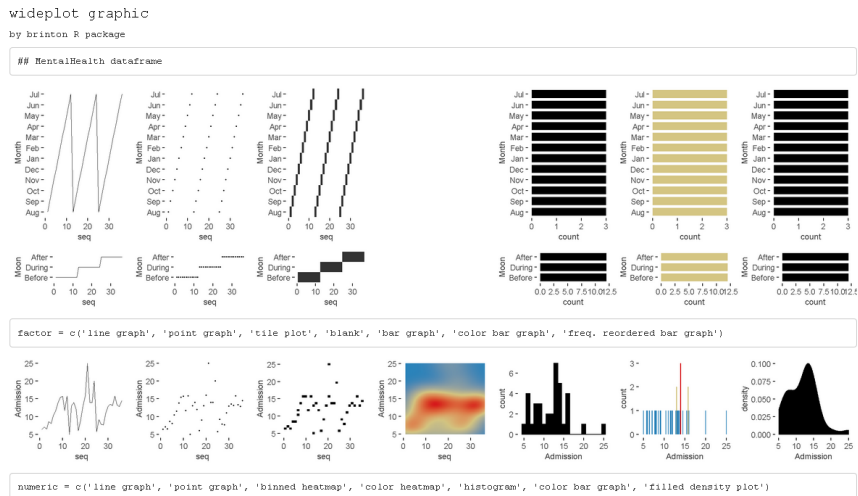


Figura 5.25: Gráfica wideplot con etiquetas. Función: `wideplot(data = MentalHealth, label = T)`. Fuente: Elaboración propia.

gráficas *line graph* y además, la gráfica *tile plot* para las variables de tipo factor y la gráfica *binned heatmap* para las numéricas, se puede identificar fácilmente que el conjunto de datos está primero ordenado por la variable Moon y luego por la variable Month (ver figura 5.25).

IDENTIFICAR VARIABLES QUE PUEDEN SER RECODIFICADAS Cuando se carga un conjunto de datos es conveniente comprobar qué asunciones ha hecho la función y qué variables son susceptibles de ser reclasificadas. Un ejemplo de esto lo podemos ver en la figura 5.24 que muestra como la variable `adult` del conjunto de datos `aids` puede mejor ser tratada como una variable de tipo `logical` en vez de `integer`. Si recodificamos el tipo de una variable hacia uno más conveniente, en aplicar la función `wideplot()` de nuevo, las gráficas que ésta muestra suelen resultar también más convenientes. Como ejemplo, en la figura 5.26 muestra el resultado una vez reclasificada la variable `adult`.

```
aids$adult <- as.logical(aids$adult)
wideplot(data = aids)
```

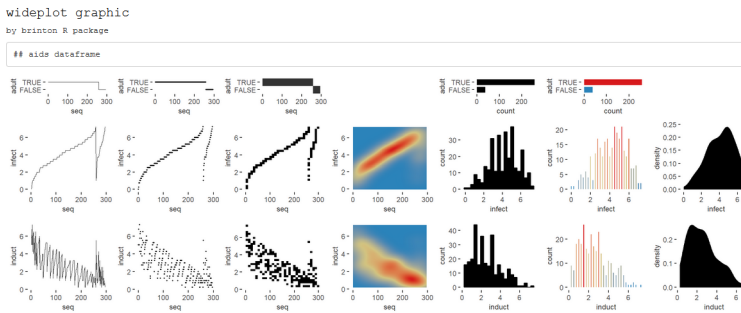


Figura 5.26: Gráfica wideplot. Función: `wideplot(data = aids)`.
Fuente: Elaboración propia.

IDENTIFICAR VARIABLES ÍNDICE La mejor manera de identificar variables clave es mediante gráficas complementarias. La figura 5.27 permite, por ejemplo, identificar rápidamente la variable `patient` del conjunto de datos `azt` de la librería `KMsurv`, como una variable clave dado que asigna un número secuencial a cada registro, cada uno de los cuales se observa una única vez. Estas dos conclusiones se deducen de las gráficas *line graph* y *color bar graph*.

```
data(azt, package = "KMsurv")
wideplot(data = azt, label = TRUE)
```

En el caso de variables clave categóricas, las mismas gráficas *line graph* y *color bar graph* nos ayudarían igualmente a identificar la variable clave. La figura 5.28 muestra estas dos gráficas para la variable de tipo factor del conjunto de datos `SpeciesArea` de la librería `Stat2Data` que permiten identificar rápidamente la variable `Name` como variable clave.

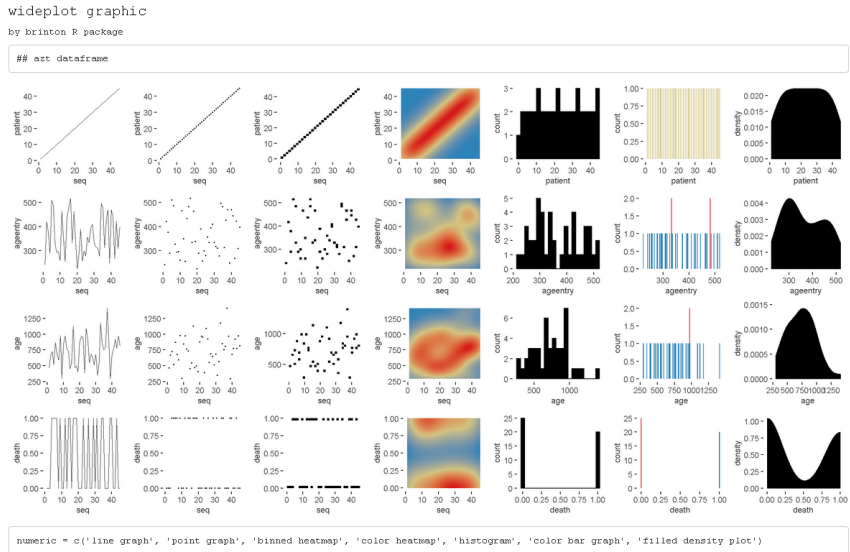


Figura 5.27: Gráfica wideplot con etiquetas. Función: `wideplot(data = azt, label = TRUE)`. Fuente: Elaboración propia.

```
data(SpeciesArea, package = "Stat2Data")
wideplot(data = SpeciesArea,
         dataclas = c("factor"),
         factor = c('line graph', 'color bar graph'),
         ncol = 5)
```

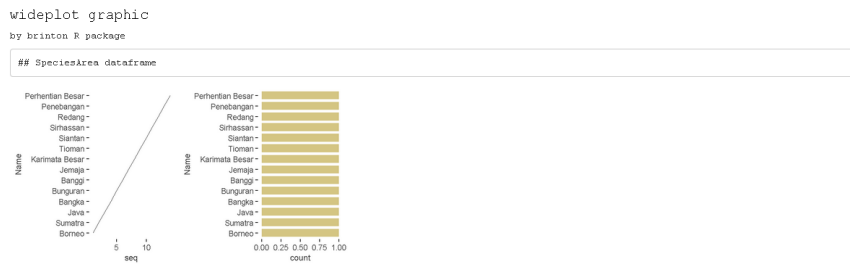


Figura 5.28: Gráfica wideplot para identificar variables clave. Fuente: Elaboración propia.

DEJARSE SORPRENDER POR LA SERENDIPIA A continuación mostramos casos aislados en los que nos dejamos sorprender por los valores que dibujan los datos. El procedimiento que hemos llevado a cabo para hallar aspectos inesperados en los datos ha sido siempre el mismo: primero obtenemos una vista general del conjunto de datos mediante la función `wideplot()`; luego focalizamos nuestra atención en alguna variable en particular y exploramos todas las gráficas compatibles mediante la función `longplot()`; finalmente utilizamos la función `plotup()` para obtener la gráfica que mejor permite identificar, acotar o comunicar el aspecto en los datos que hemos hallado.

- Un primer ejemplo de hallazgo inesperado lo tenemos en la variable `experience` del conjunto de datos HI de la librería `Ecdat`. Este conjunto de datos cuenta con 22.272 registros de trece variables que relacionan los seguros de salud con las horas semanales trabajadas por las esposas de los tomadores del seguro, mientras que la variable `experience` se refiere a los años de experiencia laboral en potencia de las esposas. Si observamos la gráfica de barras aplicada sobre esta variable numérica (ver figura 5.29), vemos que la frecuencia con que se observan los valores enteros es sistemáticamente mayor que la frecuencia de los valores reales no enteros. Este comportamiento puede ser un indicio de que la variable admite ser informada con una precisión excesiva y que, quien quiera que haya informado la variable `experience` ha tendido a redondear a la unidad. Otra posibilidad es que el conjunto de datos se halla construido juntando dos fuentes de datos con diferente precisión¹.

¹En los siguientes ejemplos hemos preferido limitar el número de registros a 5.000 para reducir el tiempo de cálculo y agilizar la reproducción de estos mismos ejemplos.

```
data(HI, package = 'Ecdat')
HI_sam <- HI[sample(nrow(HI), 5000), ]
wideplot(data = HI_sam)
longplot(data = HI_sam, vars = 'experience')
plotup(data = HI_sam, vars = 'experience', diagram = 'bar graph')
```

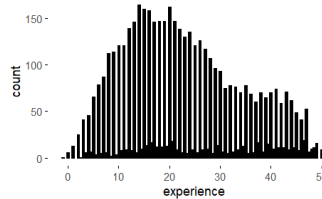


Figura 5.29: Gráfica de barras producida. Función: `plotup(data = HI_sam, vars = 'experience', diagram = 'bar graph')`. Fuente: Elaboración propia.

- En el mismo conjunto de datos podemos observar también que podríamos llegar a conclusiones equivocadas sobre la distribución de la variable `husby` (ingresos del marido en miles de dólares) si observáramos solamente un histograma. Como se puede observar en la figura 5.30 la distribución, y en particular el valor cero, adquiere un valor diferente si comparamos el histograma (a la derecha de la figura) con otra gráfica, no tan común para variables numéricas, como es la gráfica de barras (a la izquierda de la figura) que muestra el recuento de valores únicos. La gráfica de barras permite diferenciar claramente dos grupos: el de las informantes cuyos maridos no tienen ingresos y el de cuyos maridos sí que tienen y, por consiguiente, tiene más sentido preguntar el valor aproximado de los ingresos.

COMBINAR LAS GRÁFICAS QUE MEJOR REPRESENTAN UNA CARACTERÍSTICA ESPECÍFICA DE LOS DATOS Del mismo modo como las gráficas multipanel permiten revelar diferentes aspectos de los datos,

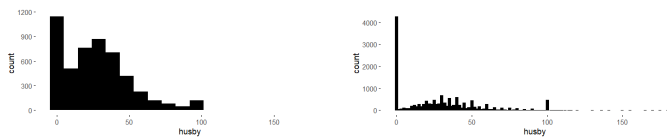


Figura 5.30: Histograma (izquierda) y gráfica de barras (derecha) que permite identificar el significado especial del valor “cero”. Fuente: Elaboración propia.

puede ser conveniente también utilizar una selección de gráficas para presentar una cierta característica de los datos. A continuación mostramos un ejemplo de cómo la librería **brinton** puede ayudarnos a mejorar las gráficas obtenidas por defecto para luego combinarlas para evidenciar una característica.

- Un problema recurrente cuando tratamos conjuntos de datos con muchos registros es la colisión entre las marcas que impide interpretar correctamente el conjunto de las observaciones. La presentación de múltiples gráficas para representar unos mismos valores nos permite identificar estas colisiones y mejorar la representación que la librería muestra por defecto. Por ejemplo, en la figura 5.31 podemos observar cómo el diagrama de puntos para la misma variable `husby` resulta poco esclarecedor debido a la superposición de las marcas.

```
plotup(data = HI_sam, vars = 'husby', diagram = 'point graph')
```

- El usuario no tiene porqué conformarse con el resultado por defecto sino que puede recuperar la función de `ggplot2` que utiliza la librería mediante el argumento `output = 'console'` para luego mejorarla:

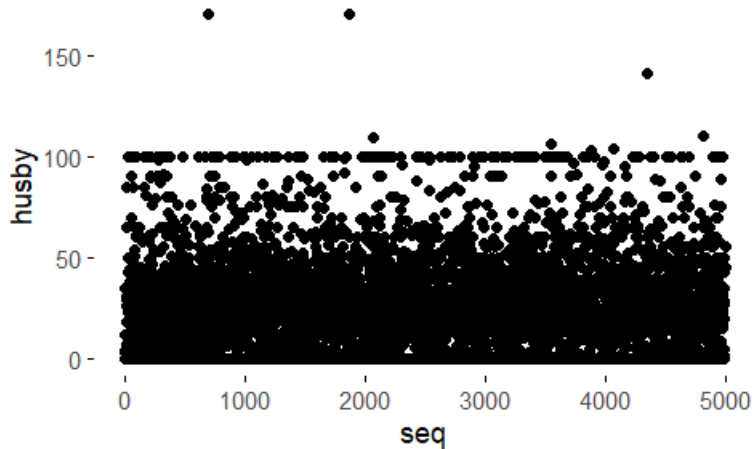


Figura 5.31: Gráfica de puntos francamente mejorable. Función `plotup(data = HI_sam, vars = 'husby', diagram = 'point graph')`. Fuente: Elaboración propia.

```
plotup(data = HI_sam,
       vars = "husby",
       diagram = "point graph",
       output = "console")
```

```
## ggplot(HI_sam, aes(x=seq_along(husby), y=husby)) +
##   geom_point() +
##   labs(x='seq') +
##   theme_minimal() +
##   theme(panel.grid = element_line(colour = NA),
##         axis.ticks = element_line(color = 'black'))
```

- En este caso podemos, por ejemplo, mejorar la gráfica reduciendo el tamaño de los puntos y añadiendo un canal alfa (ver figura 5.32).

```
newpointgraph <- ggplot(HI_sam, aes(x=seq_along(husby), y=husby)) +
  geom_point(size = 0.3, alpha = 0.15) +
  labs(x='seq') +
  theme_minimal() +
  theme(panel.grid = element_line(colour = NA),
        axis.ticks = element_line(color = 'black'))
```

```
newpointgraph
```

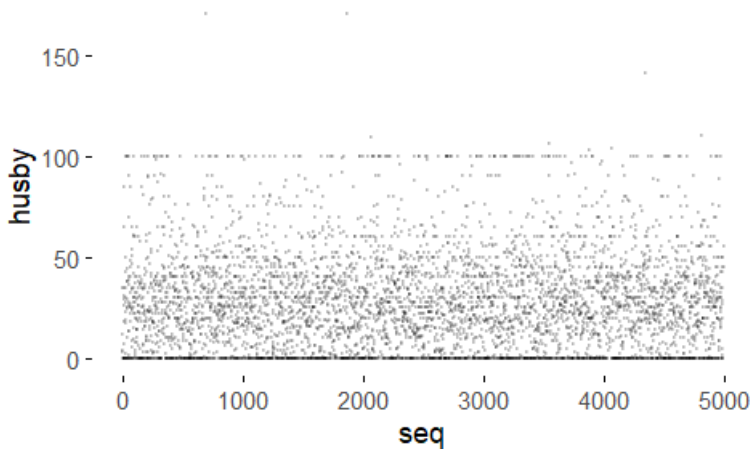


Figura 5.32: Gráfica de puntos mejorada. Fuente: Elaboración propia.

- Una alternativa a esta gráfica que no se ve afectada por la colisión de marcas es el mapa de calor y otra alternativa es la gráfica de barras que representa la frecuencia con la que se observan los valores únicos. La combinación de las tres gráficas ayuda a destacar diferentes aspectos de modo que se facilita, en definitiva, la comprensión de los datos. La figura 5.33 muestra la manera de combinar las tres gráficas. Queremos hacer notar que la gráfica de barras se ha rotado 90 grados a modo de gráfica marginal, siguiendo la gramática implementada en `ggplot2`, para facilitar la correspondencia entre las observaciones individuales, la densidad que se deduce de éstas y la frecuencia de valores únicos. Asimismo, se han retocado las etiquetas de los ejes para evitar reiteraciones innecesarias.

```

newpointgraph +
  labs(y = "husband's income * 1000$") +
  plotup(data = HI_sam, vars = 'husby', diagram = 'color heatmap') +
  labs(y = '') +
  plotup(data = HI_sam, vars = 'husby', diagram = 'bar graph') +

```

```
labs(x = "'') +
coord_flip()
```

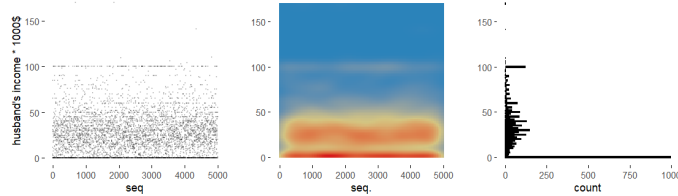


Figura 5.33: Gráfica multipanel compuesta a partir de tres gráficas producidas con la función `plotup()`. Fuente: Elaboración propia.

La gráfica multipanel resultante muestra que a lo largo del conjunto de datos, la distribución de los ingresos se mantiene esencialmente constante, en la que destaca el número de maridos sin ingresos y el redondeo de los valores informados alrededor de números redondos como 25, 30, 40, 50 y 100 (aunque en realidad, el valor que dibuja una línea horizontal alrededor de 100 es, sorprendentemente, 99.999). Así obtenemos otra incógnita que dilucidar.

5.5. CONCLUSIONES

Este artículo ha presentado la librería `brinton`, una herramienta de autoGEDA, diseñada para facilitar la presentación, selección y edición de gráficas estadísticas edificadas sobre `ggplot2`. Esta librería explota hasta el máximo nivel la estrategia determinística de selección de gráficas, primero mediante la presentación de abanicos de gráficas, entre las que los usuarios pueden seleccionar de manera nominal, automatizar su construcción e incluso recuperar la función de `ggplot2` empleada y adaptar a sus necesidades. La utilización de esta librería facilita el conocimiento de los conjuntos de datos y estimula la formulación de hipótesis en base a éstos.

Esta primera versión de la librería representa la semilla sobre la que los autores quieren edificar un más amplio catálogo que incluya

gráficas para combinaciones de hasta tres variables, mejorar la estética de las gráficas por defecto y añadir nuevas funciones para el autoGEDA.

CAPÍTULO 6

BRINTON PARA GEDA BIVARIADO

“Graphical methods tend to show datasets as a whole, allowing us to summarize the general behaviour and to study detail.” — William S. Cleveland

El primera versión 0.1.0 del paquete `brinton` fue publicada en el CRAN el 30 de noviembre de 2019. Las principales características de este paquete fueron ya presentadas por Millán-Martínez y Oller (2020). Desde entonces, el paquete se ha beneficiado de mejoras entre las que podemos destacar el incremento de gráficas disponibles en los especímenes, la posibilidad de producir gráficas bivariadas (a partir de la versión 0.1.4 de febrero de 2020) o la introducción de la nueva función `matrixplot()` para producir gráficas del tipo *pairs plot* (a partir de la versión 0.2.0 de junio de 2020). Este capítulo presenta en detalle la utilidad de las mejoras introducidas que están orientadas al análisis bivariado.

6.1. PREÁMBULO

Las primeras versiones del paquete `brinton` incluían tres funciones básicas para el análisis gráfico exploratorio de datos que, en resumen y obviando las diferentes opciones, se describen a continuación. La función `wideplot()` introdujo una nueva gráfica multipanel que recibe el mismo nombre que la función. Ésta función genera una matriz de gráficas en la que cada variable de un conjunto de datos está representado por una fila de gráficas, y cada columna representa diferentes tipos de gráficas construidas a partir de la misma variable. La función `longplot()` produce, a partir de la especificación de un conjunto de datos y una variable en particular, un abanico con

todas las gráficas consideradas per el paquete para esa variable en particular. Finalmente, la función `plotup()` produce una gráfica concreta a partir de la especificación de un conjunto de datos, una variable y el nombre de la gráfica deseada con la capacidad, además, de recuperar el código de **R** utilizado para graficarla y así facilitar su edición. Estas primeras versiones del paquete incluyen además un espécimen con todas las gráficas univariadas consideradas hasta la fecha por el paquete. El espécimen muestra los diferentes tipos de gráficas, a partir de unas variables escogidas *ad hoc* a título de ejemplo, de unos conjuntos de datos de diferentes paquetes de **R** disponibles en el CRAN.

El análisis univariado suele ser el primero que se lleva a cabo en el momento de realizar un EDA, este primer análisis permite conocer aspectos como por ejemplo la distribución de las variables, el recuento de valores únicos, el recuento de registros, la identificación de valores no informados o atípicos, o la existencia de variables índice o de variables que ordenan el conjunto de datos. Resulta del todo útil, sin embargo, complementar el análisis univariado con el análisis bivariado que permite, por ejemplo, observar si dos variables están correlacionadas, cuantificar esa correlación, comparar la distribución de dos o más grupos o hacer recuentos respecto a pares de observaciones (frecuencias bivariadas). Más allá del análisis bivariado se encuentra el análisis trivariado, que implica específicamente tres variables, y el análisis multivariado que se suele referir al análisis simultáneo de las relaciones entre tres o más variables. El análisis multivariado se puede abordar de dos maneras diferentes, una posibilidad es intentar conseguir una gráfica unipanel que combine el mayor número de variables en él. Otra posibilidad es construir gráficas multipanel que combinen pares o grupos de variables de manera sistemática. Si cada panel combina pares de variables, cada uno de ellos facilita el análisis bivariado y el conjunto de paneles, el análisis multivariado aunque desagregado por pares.

A continuación explicamos cómo se ha incorporado el análisis bivariado en las funciones `longplot()` y `plotup()` ya presentadas para el análisis univariado. También presentamos la nueva función `matrixplot()` para el análisis exploratorio de relaciones entre pares de variables de un conjunto de datos.

6.2. ANÁLISIS BIVARIADO EN LOS PAQUETES DE AUTOEDA DE R

Staniak y Biecek (2019) analizan hasta 14 paquetes dedicados al análisis exploratorio de datos automatizado (autoEDA). En el capítulo 5 ya hemos hecho referencia a estos paquetes que, de un modo u otro, incluyen funciones útiles tanto para el análisis univariado como el bivariado. Dado que este capítulo está dedicado a describir la extensión para el análisis bivariado del paquete `brinton`, antes resulta útil explorar cómo los diferentes paquetes de autoEDA de R abordan el análisis bivariado.

Entre los 14 paquetes, no todos incluyen funciones para el análisis bivariado, por ejemplo, los paquetes `visdat`, `inspectdf` (Tierney, 2017) o `dataMaid` (Petersen y Ekstrøm, 2019) están especialmente pensados para la exploración de la estructura, la completitud o los errores de los conjuntos de datos. Otros paquetes limitan el análisis bivariado a describir de forma tabulada las relaciones entre pares de variables. Ejemplos de estos paquetes los tenemos en `summarytools::ctable()` (Comtois, 2019) que produce tablas de contingencia, o el paquete `arsenal` (Heinzen et al., 2021) con funciones como `freqlist()`, `paired()` o `tableby()` que producen tablas de frecuencia bivariadas o multivariadas que pueden incluir, además, una variedad de estadísticos que relacionan las variables consideradas.

Entre los paquetes que incluyen gráficas que relacionan pares de variables encontramos de dos tipos. Por un lado tenemos paquetes que incluyen funciones que generan gráficas bivariadas pero que requieren que el usuario especifique las dos variables a relacionar. Ejemplos de esto los tenemos en el paquete `ExPanDaR` (Gassen, 2020) con funciones

como `prepare_scatter_plot()` o `prepare_trend_graph()`, o el paquete `dlookr` (Ryu, 2021) cuya función `relate()` crea objetos de una clase homónima que tienen asociados los métodos `summary.relate` o `plot.relate` que devuelven básicamente tablas o diagramas (de densidad, de mosaico, de caja o de dispersión), en función del tipo de variables y el rol de éstas como de respuesta o predictoras. También el paquete `funModeling` (Casas, 2020) incluye las funciones `cross_plot()` o `plotar()` que producen diagramas de barras, de cajas o de densidad entre pares de variables concretas.

Por otro lado tenemos paquetes que incluyen funciones para crear tablas y gráficas que relacionan las variables por pares de forma más generalizada. Paquetes como `DataExplorer` (Cui, 2019), `explore` (Krasser, 2021) o `SmartEDA` (Dayanand Ubrangala et al., 2019), por ejemplo, incluyen funciones como `plot_boxplot()`, `explore_all()` o `ExpNumViz()` que, a partir de una variable objetivo, producen diagramas multipanel de un determinado tipo, que relacionan la variable objetivo con el resto de variables.

Cabe destacar que ninguno de los paquetes anteriores incluye métodos gráficos que analicen, sistemáticamente, todas las variables de un conjunto de datos por pares, tampoco lo hace el paquete `brinton`, sino que hay que buscar funciones diseminadas en diferentes paquetes como, por ejemplo, `gpairs::gpairs()` o `GGally::ggpairs()` que producen diagramas generalizados de pares. Tampoco hemos encontrado métodos gráficos que incluyan matrices de paneles que crucen por pares diferentes grupos de variables, sino que hay que recurrir nuevamente a funciones específicas como `GGally::ggduo`. En este caso, el paquete `brinton` sí que incorpora la función `brinton::matrixplot()` que combina por pares, como veremos, diferentes variables según el tipo. También encontramos a faltar entre los paquetes consultados un mayor abanico de tipos de gráficas, como sí que hace el paquete `brinton`, con los que poder representar las relaciones entre variables. La mayoría de paquetes analizados se limitan a generar diagramas

de barras, de dispersión, de caja y de densidad.

6.3. EL ESPÉCIMEN DE GRÁFICAS BIVARIADAS

Para incorporar el análisis bivariado en las funciones `longplot()` y `plotup()` antes es necesario contar con un espécimen que incluya las gráficas que precisan de dos variables de entrada. Las gráficas bivariadas incluidas en el espécimen no solo han sido incorporadas a las funciones `longplot()` y `plotup()` sino que además pueden ser utilizadas por la nueva función `matrixplot()`. A continuación se describen los criterios generales que se han tenido en cuenta en el momento de elaborar este espécimen.

Criterios para ordenar el espécimen

El espécimen de gráficas bivariadas, a diferencia del espécimen de gráficas univariadas, no se incluye en la documentación del paquete instalable pero puede consultarse en la página web dedicada al paquete sciencegraph.github.io. El espécimen de gráficas bivariadas se estructura, antes que nada, según la tipología de las variables involucradas, de modo que las gráficas que representan dos variables de un mismo tipo se sitúan en la parte superior de la página web y las que combinan dos variables de diferente tipo, abajo. Las combinaciones de tipos consideradas hasta la fecha son las que figuran en el cuadro 6.1.

El espécimen incluye, para cada combinación de tipos de variables, una matriz de gráficas con tantas filas como tipos de gráficas han sido considerados hasta la fecha y tres columnas que, generalmente, muestran el mismo tipo de gráfica pero con diferente escala cromática (monocromática, escala de grises y escala de tonos de color), como se muestra a modo de ejemplo en la figura 6.1.

Hay casos especiales en los que las tres columnas del espécimen muestran una única escala cromática. Esto ocurre cuando la gráfica

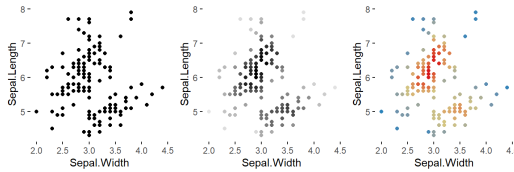


Figura 6.1: Vector de gráficas del espécimen para dos variables numéricas que se identifican como `scatter plot`, `bw scatter plot` y `color scatter plot`. Estas gráficas combinan las variables numéricas `Sepal.Length` y `Sepal.Width` del conjunto de datos `iris`. Fuente: Elaboración propia.

contiene una variable de tipo `factor` cuyos valores pueden ser ordenados según diferentes criterios. En estos casos, las tres columnas se utilizan para representar las siguientes tres opciones: gráfica con los valores ordenados según el orden de aparición, según el recuento de frecuencias o por orden alfabético (ver figura 6.2).

Si en vez de una única variable de tipo `factor`, la gráfica representa dos variables cuyos valores pueden ser ordenados según diferentes criterios, entonces el espécimen puede incluir dos filas con el mismo tipo de gráficas y escala cromática, pero con la segunda fila con las variables traspuestas tal como se aprecia en la figura 6.3

<code>Numeric</code>	<code>Numeric</code>
<code>Datetime</code>	<code>Datetime</code>
<code>Ordered</code>	<code>Ordered</code>
<code>Factor</code>	<code>Factor</code>
<code>Numeric</code>	<code>Datetime</code>
<code>Numeric</code>	<code>Ordered</code>
<code>Numeric</code>	<code>Factor</code>
<code>Factor</code>	<code>Ordered</code>

Cuadro 6.1: Combinaciones entre pares de tipos de variables consideradas hasta la fecha por el espécimen de gráficas bivariadas. Fuente: Elaboración propia.

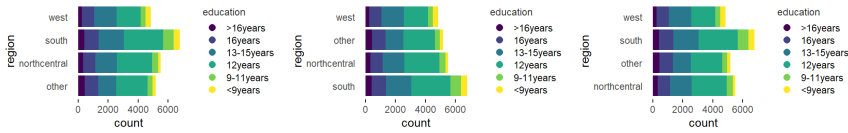


Figura 6.2: Vector de gráficas del espécimen para dos variables, una de tipo factor y otra factor ordenado, que se identifican como color stacked bar graph, color freq. reordered stacked bar graph y color alphab. reordered stacked bar graph. Estas gráficas combinan las variables region y education (factor y factor ordenado respectivamente) del conjunto de datos HI del paquete Ecdat. Fuente: Elaboración propia.

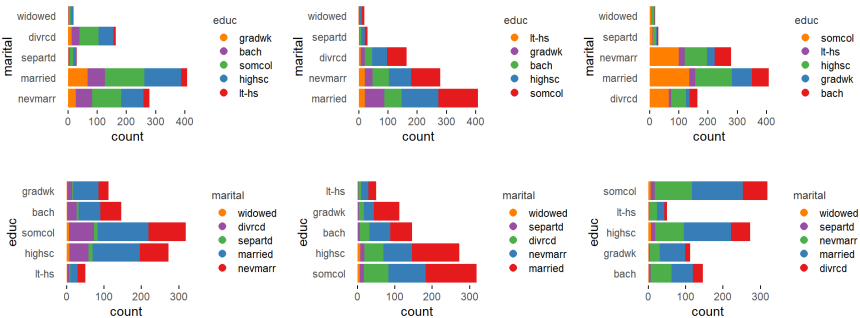





Figura 6.3: Doble vector de gráficas del espécimen para dos variables de tipo factor que se identifican como color stacked bar graph, color freq. reordered stacked bar graph y color alphab. reordered stacked bar graph el de arriba y transposed color stacked bar graph, transposed color freq. reordered stacked bar graph y transposed color alphab. reordered stacked bar graph el de abajo. Estas gráficas combinan las variables de tipo factor educ y marital del conjunto de datos jobs del paquete mediation. Fuente: Elaboración propia.

Criterios para el uso del tono de color

Como hemos comentado en la sección 6.3, los especímenes de gráficas suelen incluir tres columnas con un mismo tipo de gráfica pero con diferente escala cromática porque facilitan diferentes propiedades perceptivas básicas (ver sección 2.3). La escala monocromática no admite variaciones de tono, saturación ni luminosidad para codificar variables, por lo que cualquier variación de las componentes tiene que codificarse mediante otras VV como la posición en el plano, el tamaño, la forma, la orientación o el grano (ver figura 2.20). Por otro lado la escala de grises permite la percepción ordenada (por la cual las marcas de los diferentes valores permiten establecer un orden de menor a mayor) pero carece de propiedad asociativa, por la cual las marcas correspondientes a los diferentes valores estimulan por igual. Finalmente, la escala de tonos de color, de acuerdo con la teoría de Bertin, tan solo facilita la percepción asociativa y la selectiva, por la cual las marcas correspondientes a un determinado valor se pueden identificar de un vistazo.

Hemos utilizado cinco escalas de tono de color según los diferentes aspectos que codifican de las componentes. Por un lado, cuando el tono de color representa la densidad de observaciones como en la figura 6.1 o la frecuencia de observaciones como en la figura 6.4, utilizamos una escala divergente con tres tonos  (#2B83BA),  (#D4C582) y  (#D7191C) construida a partir de la escala ‘Spectral’ del paquete RColorBrewer.

Cuando en vez de representar cantidades o densidades, el color codifica el orden en la secuencia de observaciones como en la figura 6.5 o bien el orden entre los valores de una variable de tipo *ordered* como en la figura 6.2, entonces se ha utilizado la paleta *viridis* del paquete homónimo.

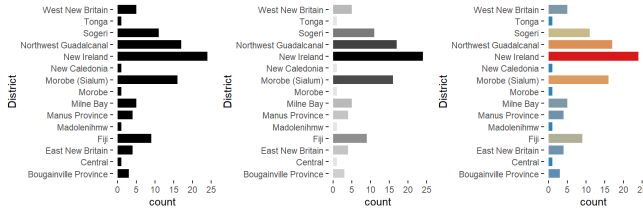


Figura 6.4: Vector de gráficas del espécimen para una variable de tipo carácter que se identifican como `bar graph`, `bw bar graph` y `color bar graph`. Estas gráficas representan la variable `District` del conjunto de datos `rockArt` del paquete `DAAG`. Fuente: Elaboración propia.

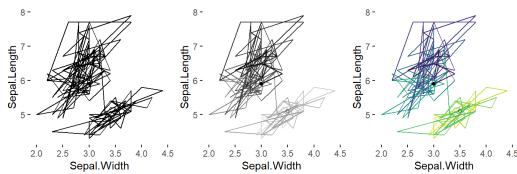


Figura 6.5: Vector de gráficas del espécimen para dos variables numéricas que se identifican como `path graph`, `bw path graph` y `color path graph`. Estas gráficas combinan las variables `Sepal.Width` y `Sepal.Length` del conjunto de datos `iris`. Fuente: Elaboración propia.

Cuando el color codifica el valor de una variable numérica calculada¹, como en la figura 6.6, que puede tomar valores positivos y negativos y de la que sabemos que el valor cero representa el valor neutro o valor sin magnitud, entonces utilizamos la paleta de color divergente `RdYlGn` del paquete `RColorBrewer` con tres tonos ■ (`#006837`), ■ (`#FFFFBF`) que representa el valor neutro, y ■ (`#A50026`). Si el color codifica el valor de una variable incluida en los datos, o un valor de una variable calculada de la que sabemos que el cero es una mera referencia y que, por consiguiente, no indica

¹Una variable numérica calculada es el resultado de hacer unas transformaciones de las variables en los datos seleccionadas. Las variables calculadas más frecuentes son los recuentos de observaciones que son necesariamente números naturales, otras variables calculadas, como por ejemplo el residuo de Pearson, pertenecen a los números reales.

ausencia de magnitud, entonces utilizamos la paleta divergente entre los tonos ■ (#018571) y ■ (#A6611A) como en el panel de la derecha de la figura 6.7.

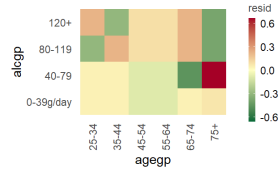


Figura 6.6: Gráfica del espécimen para dos variables de tipo factor ordenado que se ha etiquetado como `color residuals heatmap`. Esta gráfica combina las variables `alcgp` y `agegp` del conjunto de datos `esoph` y muestra también la variable calculada `resid` que corresponde al residuo de Pearson o residuo estandarizado de una frecuencia bivariada observada. El residuo en este caso puede ser tanto positivo como negativo siendo zero la ausencia de residuo. A mayor residuo absoluto, mayor contribución de la celda al estadístico de la prueba χ^2 . Fuente: Elaboración propia.

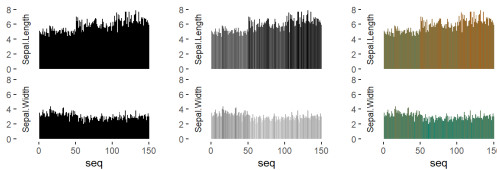


Figura 6.7: Vector de gráficas del espécimen para dos variables numéricas que se identifican como `stepped area graph`, `bw stepped area graph` y `color stepped area graph`. Estas gráficas combinan las variables `Sepal.Width` y `Sepal.Length` del conjunto de datos `iris`. Fuente: Elaboración propia.

Finalmente, cuando el tono de color se ha utilizado para codificar categorías que no guardan ninguna relación de orden como en la figura 6.3, hemos escogido la paleta cualitativa `Set1` del paquete `RColorbrewer`.

6.4. GRÁFICAS BIVARIADAS EN LAS FUNCIONES

En la función longplot

La función `longplot()`, como ya se describe en el capítulo 5, tiene una estructura muy simple: `longplot(data, vars, label = TRUE, dir = tempdir())`. El argumento `data` es el objeto de clase `data.frame` que contiene las variables a analizar. El argumento `vars` es un vector de clase `character` con los nombres las variables que se quiere analizar del conjunto de datos. El argumento `label` tiene por defecto el valor `TRUE` y es un objeto de clase `logical` que permite añadir etiquetas con los nombres asignados a cada una de las gráficas debajo de cada fila de la matriz de gráficas. El argumento `dir` permite especificar el directorio en el que se almacena el fichero `html` que la función genera.

Inicialmente el argumento `vars` admitía únicamente vectores de longitud 1 y ahora, con la incorporación del espécimen de gráficas bivariadas, ha pasado a admitir vectores de longitud 2. Hay que tener en cuenta, sin embargo, que a partir del nombre de las variables, la función deduce la clase de éstas y comprueba si la combinación de clases está entre las consideradas hasta la fecha en el espécimen (ver sección 6.3) y, en caso de que la combinación no esté incluida en el espécimen, la función devuelve un error.

Para comprobar el funcionamiento de la función `longplot()` con dos variables, se puede ejecutar el siguiente código que produce una página `html` (ver figura 6.8) con diferentes gráficas que combinan la fecha de descarga con el número de descargas diarias del paquete `lattice` durante la primera mitad de 2022.

```
downloads <- cranlogs::cran_downloads(from = "2022-01-01",
  to = "2022-07-01",
  packages = "lattice")
longplot(data = downloads, vars = c('date', 'count'))
```

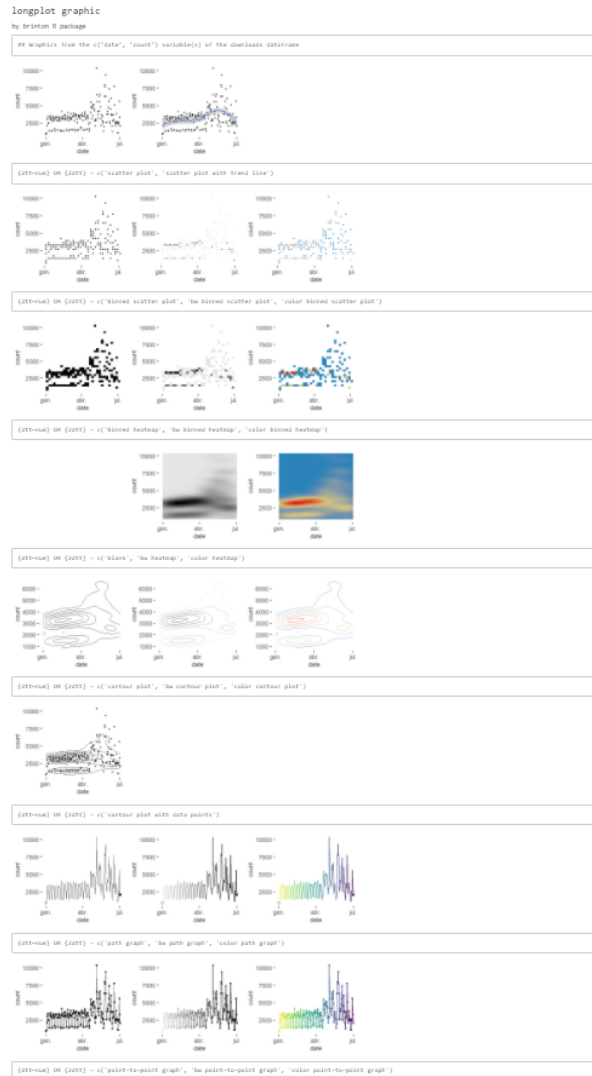


Figura 6.8: Gráfica longplot con gráficas que combinan la fecha de descarga con el número de descargas diarias del paquete `lattice` la primera mitad de 2022. Función: `longplot(data = downloads, vars = c('date', 'count'))`. Fuente: Elaboración propia.

En la función plotup

La función `plotup()`, como se ha descrito también en el capítulo 5, tiene la estructura siguiente: `plotup(data, vars, diagram, output = "plots pane", dir = tempdir())`. Los argumentos `data`, `vars` y `dir` coinciden con los de la función `longplot`. El argumento `output` ofrece tres alternativas: `html` produce y muestra un fichero de este tipo que contiene la gráfica; `plots pane`, el valor por defecto, produce un objeto de clase `ggplot2` y lo muestra en el *plots pane* de RStudio; finalmente, `console` muestra en la consola el código necesario para reproducir la gráfica en cuestión. El argumento `diagram` permite explicitar nominalmente la gráfica que se pretende obtener de entre las que incluye el espécimen, siempre de acuerdo con el tipo de variables especificadas mediante el argumento `vars`.

Así pues, una vez observado el espécimen de gráficas bivariadas o el abanico de gráficas que produce la función `longplot()`, podemos obtener una gráfica específica (ver figura 6.9) de cualquiera de estas dos fuentes.

```
plotup(data = downloads, vars = c('date', 'count'), diagram = 'path graph')
```

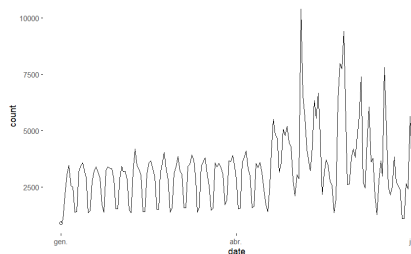


Figura 6.9: Gráfica de línea. Función: `plotup(data = downloads, vars = c('date', 'count'), diagram = 'path graph')`.

Fuente: Elaboración propia.

6.5. LA NUEVA FUNCIÓN MATRIXPLOT

Decíamos en la sección 5.4 que la librería `brinton` había sido creada para facilitar el análisis exploratorio de datos siguiendo el mantra de búsqueda de información visual *Overview first, zoom and filter, then details on demand* (Shneiderman, 1996). A la idea principal de acompañar al usuario en estas tres fases mediante tres funciones (`wideplot()`, `longplot()` y `plotup()`), hemos añadido la nueva función `matrixplot`.

Decíamos también que la función `wideplot()` está pensada para explorar la estructura de un conjunto de datos y que, a partir de ese punto podíamos solicitar la presentación de un catálogo mayor de gráficas para una variable en particular mediante la función `longplot()`, o podíamos también solicitar la presentación de una gráfica en particular mediante la función `plotup()` que permitía, adicionalmente, recuperar el código utilizado para su generación y así adaptarlo a nuestras necesidades.

La nueva función `matrixplot()` la situamos como un segundo paso justo después de la comprensión de la estructura del conjunto de datos, dado que permite ampliar el abanico de preguntas a las que las funciones `longplot()` y `plotup()` pueden dar respuesta. Las preguntas a las que la función `wideplot()` puede responder son sobre su estructura, por ejemplo, el número de variables que contiene un conjunto de datos y sus nombres, el número de registros, el tipo de variables, el orden del conjunto de datos o una muestra de los valores observados. También permite conocer características intravariante, como por ejemplo, la distribución de las variables, el orden en el que se suceden las observaciones, el rango de los valores observados, los valores únicos, los atípicos o los perdidos.

La función `matrixplot()`, en cambio, produce una variante de un tipo de gráfica conocida como *pairs plot* que está pensada para observar las interrelaciones entre pares de variables (o en otras palabras

análisis bivariado) y así elevar el nivel del análisis.

Antecedentes de la función `matrixplot`

Como hemos avanzado en el punto anterior, la función `matrixplot()` produce gráficas multipanel que combinan, por pares, las variables de un conjunto de datos y es un caso particular de las gráficas conocidas como diagramas de pares de variables (o *pairs plot* en inglés). Una solución clásica de diagrama de pares la podemos obtener fácilmente mediante la función `pairs()` de **R** que combina pares de variables numéricas mediante diagramas de dispersión, tal y como se puede apreciar en la figura 6.10.

Una mirada atenta a esta figura nos permite comprobar, sin embargo, que existen paneles que incluyen la variable `Species` del conjunto de datos `iris` la cual es categórica. Lo que sucede es que la variable categórica `Species`, que originalmente podía adquirir los valores `virginica`, `versicolor` y `setosa`, se representa recodificada como una variable numérica pudiendo ahora adquirir los valores 1, 2 y 3. Esta solución también se encuentra, por ejemplo, en el sistema GGobi (Cook et al., 2007) y permite obtener diagramas de dispersión para todas las combinaciones de pares de variables, independientemente de su tipo, aunque con el inconveniente de desconocer la correspondencia entre los niveles de la escala original y las etiquetas en la escala gráfica recodificada.

```
pairs(iris)
```

Otro antecedente de la función `matrixplot()` es el diagrama de pares generalizado (o *generalized pairs plot*) que, en vez de mostrar un único tipo de gráfica común en todos los paneles, presenta diferentes gráficas en función de los tipos de los pares de variables que se combinan en cada celda de la matriz (Emerson et al., 2013; Friendly, 2014).

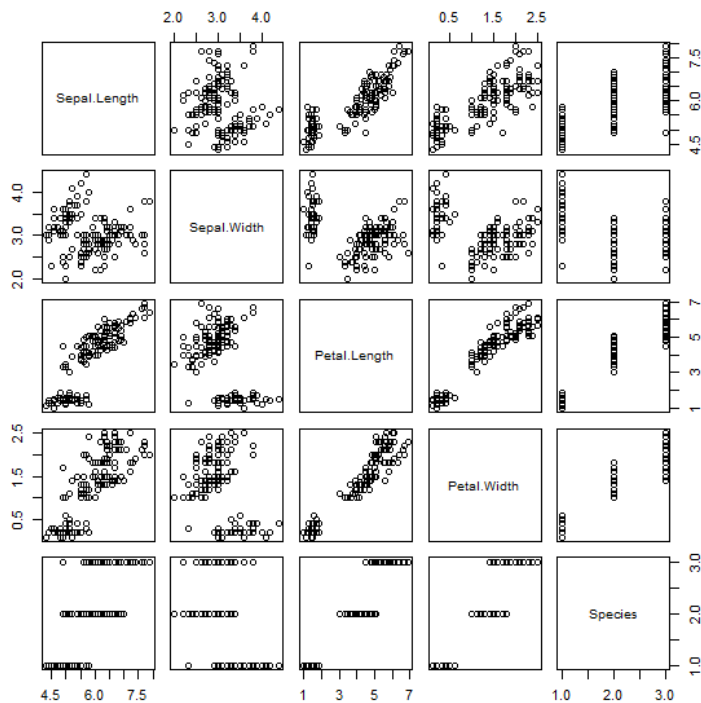


Figura 6.10: Matriz de diagramas de dispersión. Función: `pairs(iris)`. Fuente: Elaboración propia.

La gráfica generalizada de pares tiene dos principales implementaciones en **R**. La primera es la función `gpairs()` del paquete homónimo que se basa en el paquete `lattice`. La función `gpairs()` produce gráficas como la de la figura 6.11, en la que se observa que para cada combinación de pares de tipos de variables, el tipo de gráfica mostrado cambia. También se observa que la matriz triangular inferior y superior pueden mostrar diferentes tipos de gráficas a pesar de combinar unos mismos tipos de variables y cómo las etiquetas de las variables categóricas `smoker` y `day` no han sido recodificadas como numéricas.


```
gpairs::gpairs(reshape2::tips[, c('total_bill', 'tip', 'smoker', 'day')])
```

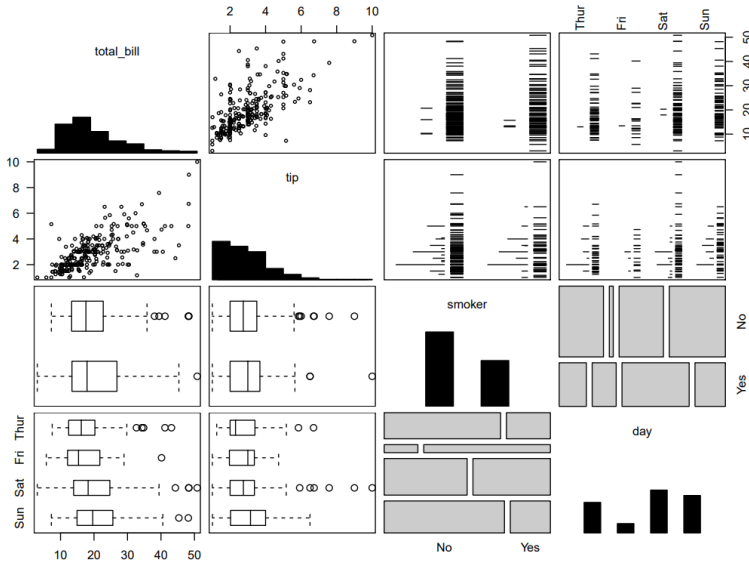


Figura 6.11: Diagrama generalizado de pares. Los paneles en la diagonal muestran histogramas si se trata de variables numéricas y diagramas de barras si son categóricas. Si se combinan 2 variables numéricas se muestran diagramas de dispersión, en el caso de dos variables categóricas se muestran gráficas de mosaico y si se combinan numéricas y categóricas, se muestran diagramas de caja o diagramas de tira. Función: `gpairs::gpairs(tips[, c('total_bill', 'tip', 'smoker', 'day')])`. Fuente: Elaboración propia.

La segunda implementación del diagrama de pares generalizado la tenemos en la función `ggpairs()` del paquete `GGally` sobre el paquete `ggplot2` que produce gráficas como la de la figura 6.12. Una ventaja de la función `ggpairs()` respecto a `gpairs()` es que la primera puede incorporar gráficas personalizadas por el usuario mientras que la segunda cuenta con un abanico de gráficas limitado. El paquete `GGally` incluye también la función `ggduo()` (Schloerke, 2017) que presenta una variante del diagrama de pares generalizado en el sentido que, en vez de combinar por pares todas las variables

de un conjunto de datos, combina por pares dos subconjuntos de variables de éste (ver figura 6.13). Una ventaja de `ggduo()` respecto a la gráfica generalizada de pares es que la matriz resultante no tiene porqué ser cuadrada ni tiene porqué repetir combinaciones entre pares de variables.

```
GGally::ggpairs(iris)
```

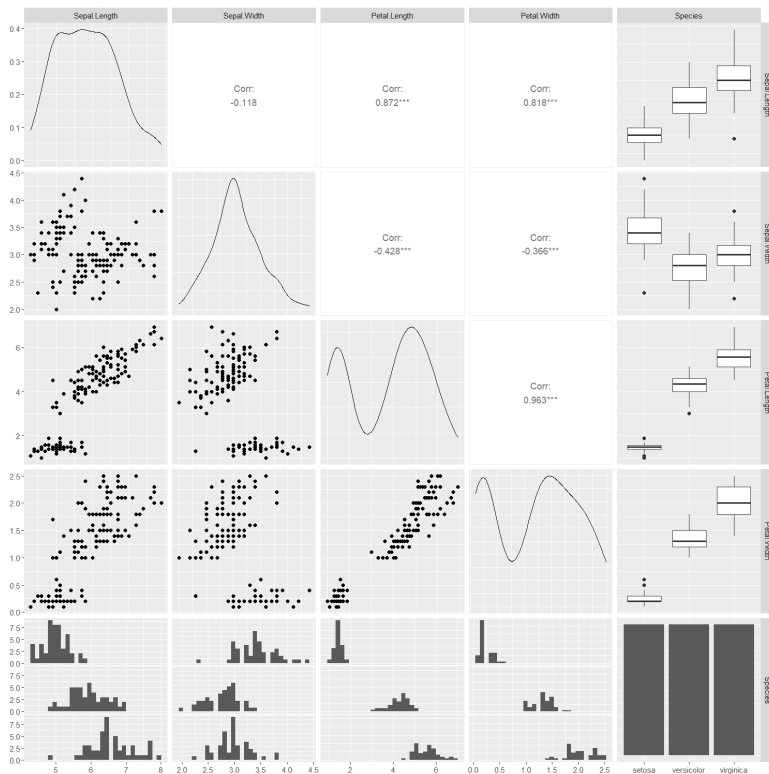


Figura 6.12: Diagrama generalizado de pares. Función: `GGally::ggpairs(iris)`. Fuente: Elaboración propia.

```
GGally::ggduo(iris, names(iris[5]), names(iris[1:4])) + theme_bw()
```

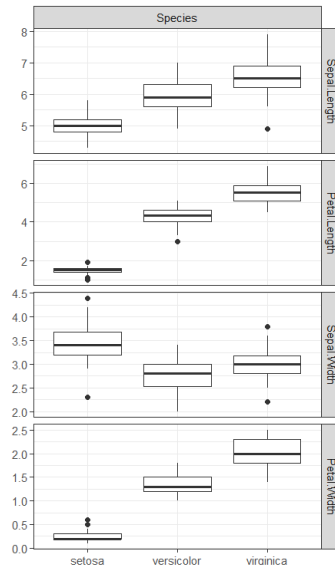


Figura 6.13: Diagrama de pares de variables de tipo cruzado. Función: `GGally::ggduo(iris, c("Species"), c("Sepal.Length", "Petal.Length", "Sepal.Width", "Petal.Width"))`. Fuente: Elaboración propia.

Problema que se pretende resolver

Los antecedentes de gráficas de pares muestran todos un problema no resuelto que está relacionado con el tamaño de los paneles en el caso de incluir variables categóricas. La función `pairs()` y también generalmente la función `gpairs()`, por ejemplo, recodifican las variables categóricas de modo que los diferentes valores de éstas pasan a tener un índice numérico. Por ejemplo, la variable `Seed` de clase `Ord.factor` del conjunto de datos `Loblolly` que incluye los valores `c('329', '327', '325', '307', '331', '311', '315', '321', '319', '301', '323', '309', '303', '305')`, al ser procesada por cualquiera de estas dos funciones, sufre una recodificación de los valores anteriores por los de una variable numérica, en este caso una secuencia del 1 al 14, como muestra la

figura 6.14. La relación entre los valores originales de la variable categórica y los valores recodificados en una variable numérica resulta del todo confusa.

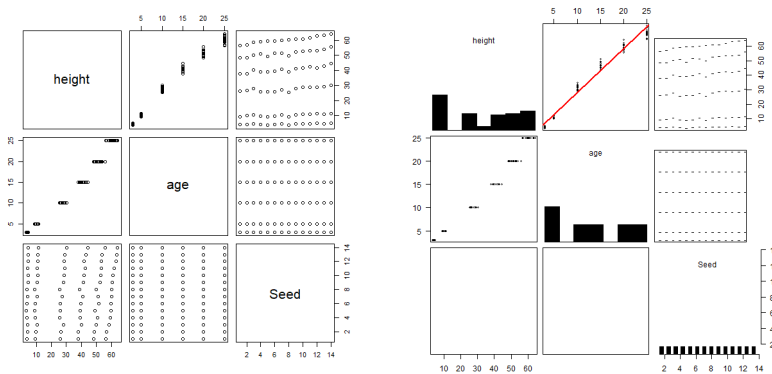


Figura 6.14: Comparación entre el diagrama de pares (izquierda) y el diagrama generalizado de pares (derecha). El primero producido con la función `pairs(Loblolly)` y el segundo con la función `gpairs::gpairs(Loblolly, upper.pairs = list(scatter = 'lm'))`. Fuente: Elaboración propia.

En el caso de la función `ggpairs()`, ésta no recodifica los valores de las variables categóricas sino que intenta acomodar todas las etiquetas de graduación de las escalas en el espacio limitado de los ejes, lo que suele provocar colisiones entre las etiquetas. Este problema se evidencia en la figura 6.15.

El problema subyacente es que tanto el sistema básico de gráficos de R que utiliza la función `pairs()` como el sistema de gráficos `grid` sobre el que basan los paquetes `lattice` y `ggplot2` y a su vez las funciones `gpairs()` y `ggpairs()` respectivamente, no establecen el tamaño del área gráfica en función de las etiquetas de graduación de las escalas a representar, sino que parten del espacio disponible en el lienzo virtual (*layout* en el sistema básico o *viewport* en el sistema `grid`) para hacer caber la gráfica en su conjunto. Esto incluye los diferentes paneles, los márgenes entre paneles, los márgenes exteriores, las leyendas, los títulos e incluso puede incluir los pies de figura. Esta

estrategia no suele ser problemática cuando los ejes espaciales soportan variables numéricas o de fechas, pero si la gráfica que necesitamos consta de ejes espaciales que soportan variables categóricas, entonces es fácil que las etiquetas de estas escalas de graduación colisionen unas con otras.

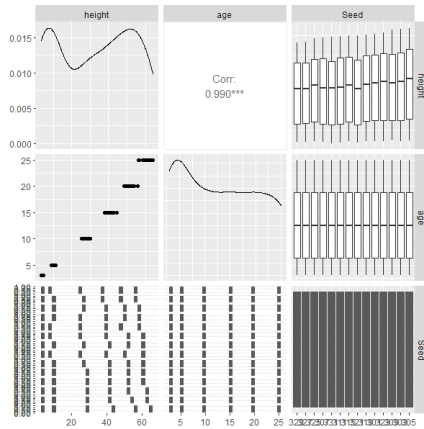


Figura 6.15: Diagrama generalizado de pares. Función: `GGally::ggpairs(Loblolly)`. Fuente: Elaboración propia.

Solución propuesta

Una posible solución al problema descrito en el apartado anterior, hubiera podido ser establecer las proporciones de cada panel en base al número de valores a mostrar por las etiquetas de graduación de las escalas en los ejes espaciales. Después, la suma de las dimensiones de cada panel se hubiera podido utilizar para calcular las dimensiones del lienzo, sin embargo la solución propuesta es diferente y se describe a continuación.

La nueva función `matrixplot()` parte de la idea de que cuanto más numerosas son las combinaciones a explorar entre variables, más sencillas han de ser las preguntas que se esperan responder al observar la gráfica resultante. Por este motivo, la salida de la función `matrixplot()` no combina todas las variables de un conjunto

de datos y genera diferentes tipos de gráficas diferentes sino que combina variables de uno o dos tipos diferentes y presenta, en todos los paneles, un único tipo de gráfica a escoger entre un abanico extenso. Para tener un mayor control sobre las dimensiones de las celdas de la matriz y que incluyan las etiquetas de los ejes espaciales, la función produce una página `html` en vez de un objeto de clase `ggplot2`. A modo de ejemplo, para producir paneles que incluyan todas las combinaciones entre las variables del conjunto de datos `Loblolly` de la figura 6.15, sería necesario ejecutar dos veces la función `matrixplot()`: la primera combinaría la variable ordenada `Seed` con las variables numéricas `height` y `age` (ver figura 6.16), mientras que la segunda combinaría las variables numéricas entre ellas (ver figura 6.17).

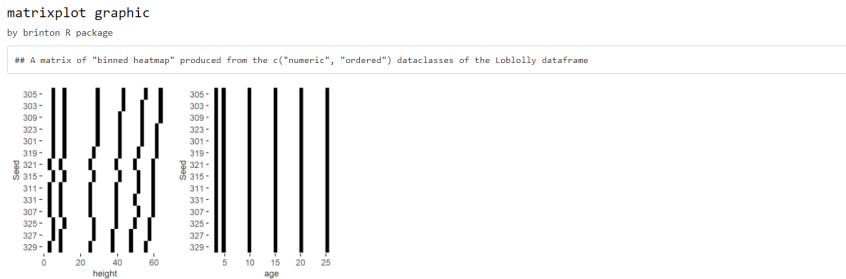


Figura 6.16: Diagrama de pares de variables de tipo cruzado. Función: `matrixplot(data = Loblolly, dataclass = c('numeric', 'ordered'), diagram = 'binned heatmap')`. Fuente: Elaboración propia.

La nueva función tiene la siguiente estructura: `matrixplot(data, dataclass = NULL, diagram = NULL, dir = tempdir())`. Los argumentos `data` y `dir` han sido descritos anteriormente en la sección 6.4, igual que el argumento `diagram` pero cabe destacar que este último, en esta función únicamente puede adoptar tipos de gráficas bivariadas, incluidas en el espécimen, para la combinación de variables especificada en el argumento `dataclass`. El argumento `dataclass` es un vector de clase `character` de longitud 1 o 2, con los nombres



Figura 6.17: Diagrama de pares de variables monotipo. Función: `matrixplot(data = Loblolly, dataclass = c('numeric', 'numeric'), diagram = 'scatter plot')`. Fuente: Elaboración propia.

de las clases de vectores que combinar, de entre los detallados en la sección 6.3.

TIPOS DE GRÁFICA ESTABLECIDA POR DEFECTO En el caso de que el argumento `diagram` no se haya explicitado en el momento de ejecutar la función, este argumento no toma un valor fijo por defecto sino que varía en función de los valores que adquiere el argumento `dataclass`. Para cada una de las combinaciones de variables que han sido consideradas hasta la versión 0.2.6 del paquete `brinton`, hemos elegido diferentes tipos de gráficas según los siguientes criterios:

- Para combinaciones de 2 variables numéricas o entre una variable numérica y otra de tipo factor (ya sea ordenado o no), la gráfica que produce por defecto es `color binned heatmap`. Hemos elegido este tipo de gráfica porque la velocidad de procesamiento es similar independientemente del número de observaciones.
- Para combinaciones que incluyen una o dos variables de tipo `datetime` (esta categoría incluye vectores de clase `Date`, `POSIXct` o `POSIXlt` o de modo más preciso, aquellos vectores para los cuales la función `lubridate::is.instant()` devuelve el valor `TRUE`), la gráfica por defecto es el `path diagram`. Se ha escogido este tipo de gráfica porque incorpora la secuencia en la

que han sido almacenados los registros en el conjunto de datos. Esta secuencia, es relativamente frecuente que guarde alguna relación con los valores de las variables de tipo `datetime`.

- Para combinaciones de variables de tipo factor, en cualquiera de sus clases `factor` u `ordered`, el tipo de gráfica elegida es `color heatmap` que representa el recuento bivariado de frecuencias de observaciones. Este tipo de gráfica tiene también la ventaja de no ver afectada la velocidad de procesamiento en función del número de registros.

DOS POSIBLES COMPOSICIONES DE PANELES En función de si los clases de variables, asociadas por pares, coinciden o son diferentes, se llega a dos escenarios diferentes.

Un primer escenario se produce si los pares de variables que se combinan en cada panel de la gráfica son de una misma clase. Este escenario produce lo que llamamos “diagrama de pares de variables monotipo” y puede darse en tres circunstancias diferentes: si el argumento `dataclass` es un vectores nulo, entonces el argumento toma por defecto el valor `dataclass = c('numeric', 'numeric')`; si el argumento no es nulo pero es un vector de longitud 1, por ejemplo `'ordered'`, los pares de variables serán entre variables de esta misma clase especificada, en este caso sería `dataclass = c('ordered', 'ordered')`; si el argumento es un vector de longitud 2 con una misma clase de variable repetida, por ejemplo `dataclass = c('factor', 'factor')`.

Cuando se da este primer escenario, una salida hubiera podido ser una matriz cuadrada clásica, en la que la diagonal mostrara los nombres de las variables como la de la figura 6.10. Sin embargo, la salida que se obtiene mediante la función `matrixplot()`, es la matriz que contiene únicamente las celdas bajo la diagonal principal, y que combina únicamente las variables del conjunto de datos de la clase especificada por el argumento `dataclass`. La salida de `matrixplot()`,

en vez de mostrar los nombres de las variables en la diagonal principal o en los ejes de la gráfica multipanel, muestra los nombres de las variables en los ejes espaciales de cada panel.

La composición de paneles que se obtiene en el primer escenario no incluye la matriz superior, dado que se trata de las mismas gráficas que las representadas pero con los ejes traspuestos. Tampoco incluye la diagonal dado que, para facilitar la interpretación de las gráficas, los nombres de las variables representadas en cada panel se encuentran en los ejes de coordenadas de cada panel. Otra razón para excluir la diagonal es que en los paneles de ésta confluyen una única variable y el análisis exploratorio univariado lo confiamos a las funciones `wideplot()` (descrita en la sección 5.4) y `longplot()` (descrita en la sección 5.4). La figura 6.18 muestra el resultado de aplicar la función `matrixplot()` sobre las variables numéricas del mismo conjunto de datos `iris` y mediante el tipo de diagrama `contour plot with data points`.

```
matrixplot(data = iris,  
           dataclass = c('numeric', 'numeric'),  
           diagram = 'contour plot with data points')
```

El segundo escenario se produce si los pares de variables que se combinan en cada panel de la gráfica son de diferente clase. Este escenario produce lo que llamamos “diagrama de pares de variables de tipo cruzado” y puede darse únicamente cuando el argumento `dataclass` es un vector de longitud 2 con dos clases de variables diferentes, por ejemplo `dataclass = c('numeric', 'factor')`. En este caso, la matriz de gráficas resultante es una matriz rectangular porque, a diferencia de lo que ocurre en el primer escenario, las variables de las filas y columnas no coinciden y todos los paneles de la matriz rectangular reúnen una combinación diferente de pares de variables. La figura 6.19 muestra el resultado de aplicar la función `matrixplot()` sobre las variables numéricas y de tipo factor del conjunto de datos `olive` del paquete `dslabs` que recoge datos de

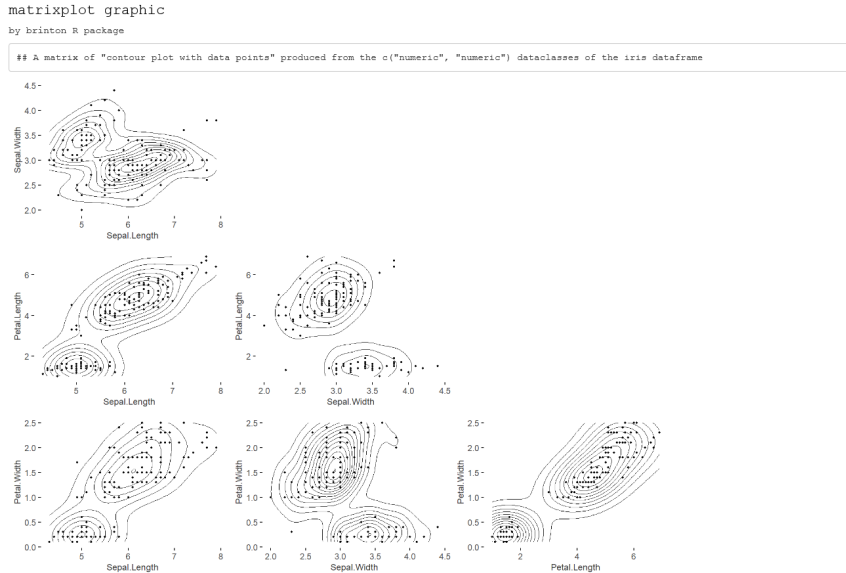


Figura 6.18: Diagrama de pares de variables monotipo. Función: `matrixplot(data = iris, dataclass = c('numeric', 'numeric'), diagram = 'contour plot with data points')`. Fuente: Elaboración propia.

composición de ácidos de diferentes muestras de aceite de oliva en tres regiones de Italia subdivididas en un total de 9 áreas.

```
library(dslabs)
data(olive)
matrixplot(data = olive,
           dataclass = c('numeric', 'factor'),
           diagram = 'color binned heatmap')
```

6.6. EJEMPLOS DE EXPLORACIÓN

En el capítulo 5 hemos mostrado algunas de las utilidades que las funciones `wideplot()`, `longplot()` y `plotup()` nos ofrecen para el análisis univariado y que permiten, por ejemplo, identificar variables índice, variables que ordenan un conjunto de datos o que pueden estar mal codificadas. También hemos mostrado algún ejemplo de

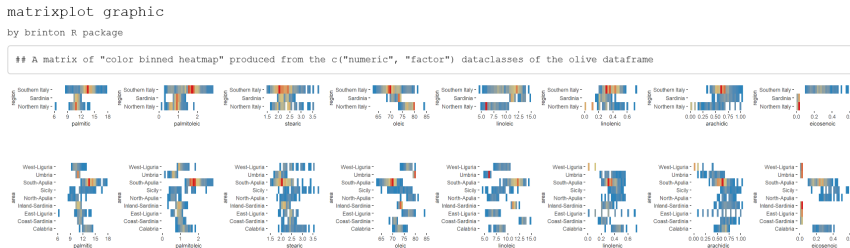


Figura 6.19: Diagrama de pares de variables de tipo cruzado. Función: `matrixplot(data = olive, dataclass = dataclass = c('numeric', 'factor'), diagram = 'color binned heatmap')`. Fuente: Elaboración propia.

cómo la presentación de gráficas diversas facilita una comprensión más poliédrica de los datos e incluso puede provocarnos un momento de serendipia. A continuación mostramos unos ejemplos de análisis gráfico exploratorio de datos que utiliza gráficas univariadas y bivariadas y que aprovecha las funciones anteriormente descritas más la nueva función `matrixplot()` del paquete `brinton`.

Exploración de discontinuidades

Nos disponemos a explorar uno de los 8 conjuntos de datos del paquete `asaur` que complementa el libro de Moore (2016). Utilizaremos concretamente el conjunto de datos `prostateSurvival` que consta de registros de diagnósticos de cáncer de próstata para el análisis de supervivencia y nos serviremos de las funciones del paquete `brinton` para explorar un aspecto de los resultados que puede tener consecuencias en la interpretación de éstos.

Una vez instalado el paquete `asaur` y cargado en memoria el conjunto de datos `data(prostateSurvival)`, vemos en la documentación que se trata de un conjunto de datos de clase `data.frame` con 14.294 registros y 5 variables. La variable `grade` adquiere dos posibles valores: `mode` (moderadamente diferenciada; se refiere a como difieren los tejidos extraídos mediante biopsia respecto a tejidos

sanos, que se sitúan entre 4 y 7 respecto a la escala de Gleason) y *poor* (apenas diferenciada; entre 8 y 10 en la escala de Gleason). La variable `stage` adquiere tres posibles valores: `T1ab` (Estado T1, diagnosticado clínicamente), `T1c` (Estado T1, diagnosticado mediante test PSA), and `T2` (Estado T2 en el que el cáncer se puede palpar u observar en un estudio de imagen pero solamente afecta la próstata). La variable `ageGroup` divide la muestra según estratos de edad en 4 valores: 66–69, 70–74, 75–79 y 80+. La variable `survTime` representa el tiempo en meses, desde la diagnosis hasta la muerte o el final del seguimiento. Finalmente, la variable `status` puede adquirir tres valores: 0 (censurado), 1 (muerte por cáncer de próstata) o 2 (muerte por otras causas).

Una vez observados los parámetros básicos del conjunto de datos, utilizamos la función `wideplot()` para explorar gráficamente su estructura y observamos en la figura 6.20 que la variable `survTime` indica está codificada como numérica a pesar de tratarse de un intervalo de tiempo. Observamos también en el panel marcado en rojo, que muestra el recuento de valores únicos, cómo entre los valores 20 y 25 de `survTime` hay un salto evidente en los valores. En el mapa de calor marcado en naranja, podemos comprobar que el salto de los valores se observa a lo largo de todo el conjunto de datos. Por otro lado, podemos observar en el panel enmarcado en azul, que a pesar de que la variable `status` está descrita como una variable categórica, en el conjunto de datos se encuentra codificada como una variable de números enteros (`int`). En el análisis de supervivencia es habitual tratar la variable que informa del estado como una variable numérica pero, para los intereses de la exploración que estamos llevando a cabo, decidimos recodificarla como `factor`.

```
wideplot(data = prostateSurvival)
prostateSurvival$status <- as.factor(prostateSurvival$status)
```

Una vez recodificada la variable `status` podemos volver a generar

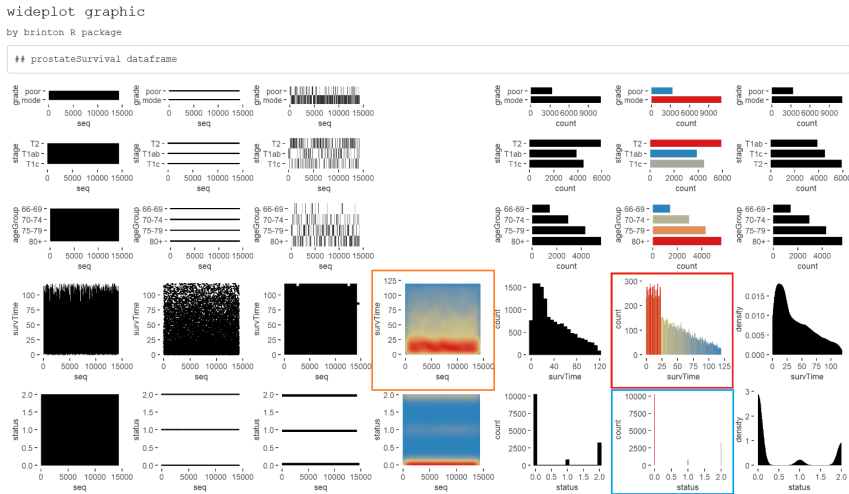


Figura 6.20: Gráfica wideplot. Función `wideplot(data = prostateSurvival)`. Fuente: Elaboración propia.

la gráfica *wideplot* y comprobar cómo la recodificación incide en la salida de esta función (ver figura 6.21).

```
wideplot(data = prostateSurvival)
```

Ahora tenemos una variable de números enteros y cuatro variables de tipo factor. Una posible exploración puede ser la relación entre el salto en la distribución de la variable `survTime` que es de tipo `numeric` y el resto de variables de clase `factor`. Para explorar las posibles relaciones entre pares de variables utilizamos la función `matrixplot` que produce la figura 6.22. La salida de esta función nos permite observar que el salto de valor entre los 24 y 25 meses de supervivencia afecta los diferentes valores de las variables `grade`, `stage` y `ageGroup` pero tan solo uno de los niveles de la variable `status`, concretamente el nivel 0, se ve afectado. Para los niveles 1 y 2, las tiras horizontales correspondientes, no muestran cambios de tono de color en ese intervalo.

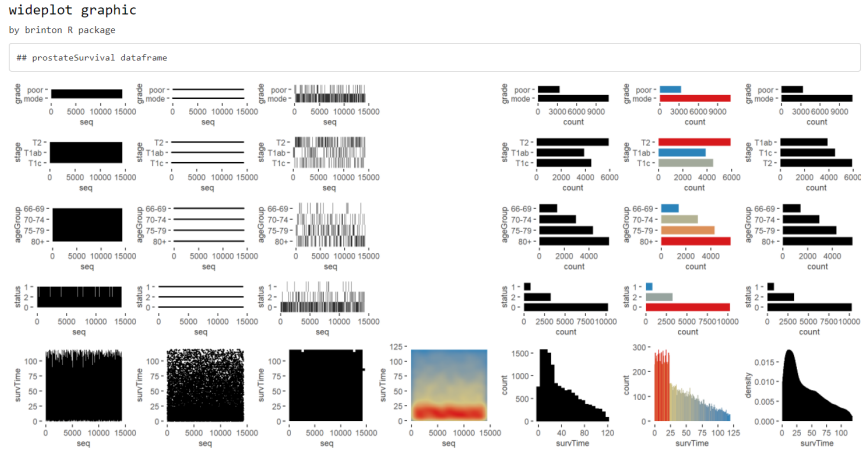


Figura 6.21: Gráfica wideplot. Función: `wideplot(prostateSurvival)` una vez recodificada la variable `status` a tipo factor. Fuente: Elaboración propia.

```
matrixplot(data = prostateSurvival, dataclass = c('numeric', 'factor'))
```

Si queremos explorar con más detalle la relación entre las variables `survTime` y `status`, nos apoyamos en la función `longplot` que mostrará el catálogo entero de gráficas disponibles para estas dos variables. El resultado es una página `html` muy extensa de la que reproducimos la sección que nos ha parecido más interesante, porque muestra las distribuciones marginales para cada nivel de la variable de tipo `status` (ver figura 6.23). Del conjunto de gráficas mostradas, escogemos la etiquetada como `color density plot` porque es la que parece contrastar mejor las diferencia entre las distribuciones de los diferentes niveles y, además, identifica con diferentes tonos de color cada uno de los niveles.

```
longplot(data = prostateSurvival, var = c('survTime', 'status'))
```

Una vez elegida la gráfica `color density plot`, pasamos este nombre como argumento de la función `matrixplot()`, y ésta substituirá el diagrama `color binned heatmap` que produce por defecto



Figura 6.22: Diagrama de pares de variables de tipo cruzado. Función: `matrixplot(data = prostateSurvival, dataclass = c('numeric', 'factor'), diagram = 'color binned heatmap')`. Fuente: Elaboración propia.

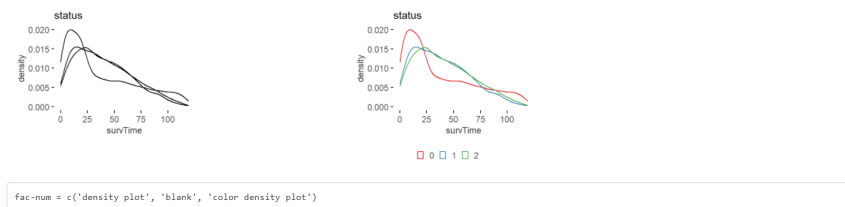


Figura 6.23: Sección de la gráfica longplot. Función: `longplot(prostateSurvival, var = c('survTime', 'status'))`. Fuente: Elaboración propia.

la función cuando la combinación de pares de variables es de tipo `numeric~factor` (ver figura 6.22), por una matriz de diagramas de densidad para cada par de variables. La nueva matriz facilita la exploración de la relación de la discontinuidad de la variable `survTime` con las diferentes variables de tipo factor (ver figura 6.24) y nos permite comprobar que la variable `status` es la única que tiene un nivel que concentra los valores discontinuos.

```
matrixplot(data = prostateSurvival,
           dataclass = c('numeric', 'factor'),
           diagram = 'color density plot')
```

matrixplot graphic

by brinton R package

A matrix of "color density plot" produced from the c("numeric", "factor") dataclasses of the prostateSurvival dataframe

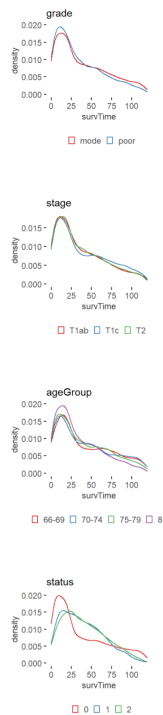


Figura 6.24: Diagrama de pares de variables de tipo cruzado. Función `matrixplot(data = prostateSurvival, dataclass = c('factor', 'numeric'), diagram = 'color density plot')`. Fuente: Elaboración propia.

Una vez identificada la variable `status` y su nivel 0 como el grupo que concentra los valores con un salto en la distribución de la variable `survTime`, filtramos el conjunto de datos por este nivel y nos ayudamos de la función `plotup()` para observar su distribución (ver figura 6.25).


```
prSurvSt0 <- prostateSurvival[prostateSurvival$status == 0,]
plotup(data = prSurvSt0, var = 'survTime', diagram = 'color bar graph')
```

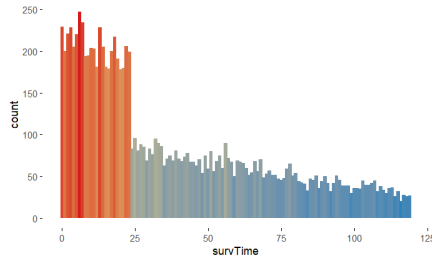


Figura 6.25: Gráfica de barras. Función: `plotup(data = prSurvSt0, var = 'survTime', diagram = 'color bar graph')`. Fuente: Elaboración propia.

Luego nos puede interesar comparar la distribución de dos grupos diferentes según si el valor de `status` es igual o diferente a cero. Para esto creamos una nueva variable `new_status` que adquiere dos posibles valores: `status 0` o `status 1` or `2`. Luego recuperamos la función que utiliza `brinton` para hacer la gráfica anterior y añadimos una línea al final del código para convertirla en multipanel, según los valores de la nueva variable `new_status` (ver figura 6.26).

```
prostateSurvival$new_status <- NA
prostateSurvival[prostateSurvival$status == 0, "new_status"]
  <- "status 0"
prostateSurvival[prostateSurvival$status != 0, "new_status"]
  <- "status 1 or 2"
```

```
plotup(data = prostateSurvival,
       var = c("survTime"),
       diagram = 'color bar graph',
       output = "console")
```

```
## binwidth <- (max(prostateSurvival['survTime'], na.rm=TRUE) -
## min(prostateSurvival['survTime'], na.rm=TRUE))/100
## ggplot(prostateSurvival, aes(x=survTime, fill=..count..)) +
##   geom_bar(stat='count', width=binwidth, position = 'identity') +
##   scale_fill_gradientn(colours = colorRampPalette(
##     rev(RColorBrewer::brewer.pal(4, 'Spectral'))(3)) +
```

```
## theme_minimal() +
## theme(panel.grid = element_line(colour = NA),
##       axis.ticks = element_line(color = 'black'),
##       legend.position='none')
```

```
binwidth <- 1
ggplot(data = prostateSurvival, aes(x=survTime, fill=..count..)) +
  geom_bar(stat='count', width=binwidth, position = 'identity') +
  scale_fill_gradientn(colours = colorRampPalette(
    rev(RColorBrewer::brewer.pal(4, 'Spectral'))(3)) +
  theme_minimal() +
  theme(panel.grid = element_line(colour = NA),
        axis.ticks = element_line(color = 'black'),
        legend.position='none') +
  facet_wrap(new_status~., ncol = 1)
```

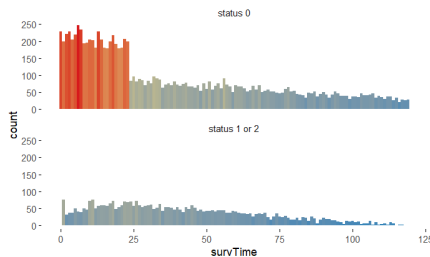


Figura 6.26: Gráfica multipanel de barras producida a partir del código recuperado por la función `plotup`. Fuente: Elaboración propia.

La figura 6.26 nos permite comparar las distribuciones de la supervivencia según los dos grupos con `status` igual o diferente a 0. La observación atenta de esta figura nos permite formular algunas hipótesis que puedan explicar porqué existe un salto, entre los 24 y 25 meses, en el recuento de pacientes que han sobrevivido hasta el final del periodo de estudio o hasta el final de su seguimiento. Una posibilidad es que los últimos dos años del estudio, entre 2000 y 2002, se incorporaran numerosos pacientes y que, una vez acabado el estudio, éstos sobrevivieran. Otra posibilidad es que los participantes del estudio, pasados los dos primeros años después del primer

diagnóstico, por algún motivo fueran tratados por centros que no informaran la base de datos fuente del estudio. Una tercera posibilidad es que hubiera alguna condición en los criterios de participación del estudio de Lu-Yao et al. (2009) que favoreciera el salto en el recuento. Finalmente, dado que el conjunto de datos graficado es un conjunto de datos simulado a partir de los resultados del conjunto original, podría haber también algún error en la simulación del conjunto de datos.

Depuración de valores atípicos

En este segundo ejemplo vamos a explorar el divulgado conjunto de datos `diamonds` del paquete `ggplot2` y a filtrar los valores atípicos para una mejor visualización de las relaciones entre pares de variables. Una vez instalado el paquete y cargado en memoria el conjunto de datos `data(diamonds)`, vemos en la documentación que se trata de un conjunto de datos de clase `tibble` con 53.940 registros y 10 variables. Entre las variables encontramos las dimensiones `x`, `y`, `z` que son valores reales positivos. Luego tenemos la proporción `depth = 2z/(x+y)` que es función de las anteriores dimensiones, `table` que es la proporción entre la tabla y el diámetro (ver figura 6.27). Otras variables numéricas son `carat` (medida de masa) y `price` (medida de coste, número natural). Las demás variables `cut` (calidad del tallado), `color` (calidad del color) y `clarity` (calidad de la claridad) son cualitativas de tipo factor ordenado.

Una vez observados los parámetros básicos del conjunto de datos, para poder introducirlo como argumento en la familia de funciones del paquete `brinton`, transformamos el objeto de clase `tibble` en un objeto `data.frame` que llamamos `diamonds.df`. Una vez transformado, exploramos el nuevo objeto `data.frame` primero mediante la función `wideplot` que produce la figura 6.28. Observamos en los paneles enmarcados en rojo que el tipo de gráfica `binned heatmap`

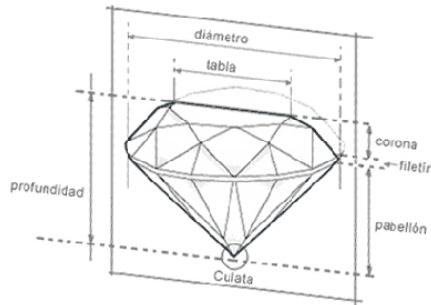


Figura 6.27: Esquema de las principales partes de un diamante.
Fuente: zircone.com.

permite identificar con mucha facilidad los valores atípicos en variables numéricas porque, a diferencia de las gráficas de línea y de punto, es un tipo de gráfica no sensible al número de registros del conjunto de datos y además, a diferencia de los histogramas y las gráficas de barras, en cada celda del tamiz se informa únicamente si existen o no observaciones, independientemente del recuento de éstas. Por otro lado, observamos que las variables categóricas no presentan a priori valores atípicos ni tampoco valores perdidos que se identificarían por pertenecer a una nueva categoría NA entre los valores de estas variables.

```
diamonds.df <- as.data.frame(diamonds)
wideplot(data = diamonds.df, label = TRUE)
```

Si decidimos explorar con más detalle los posibles valores atípicos en las variables numéricas de conjunto de datos `diamonds`, un siguiente paso puede ser representar las relaciones entre pares de variables mediante este mismo tipo de gráfica. Para esto, en vez de utilizar la función `wideplot()`, utilizamos la función `matrixplot()` para variables numéricas y explicitando el tipo de gráfica a producir. El resultado se observa en la figura 6.29.

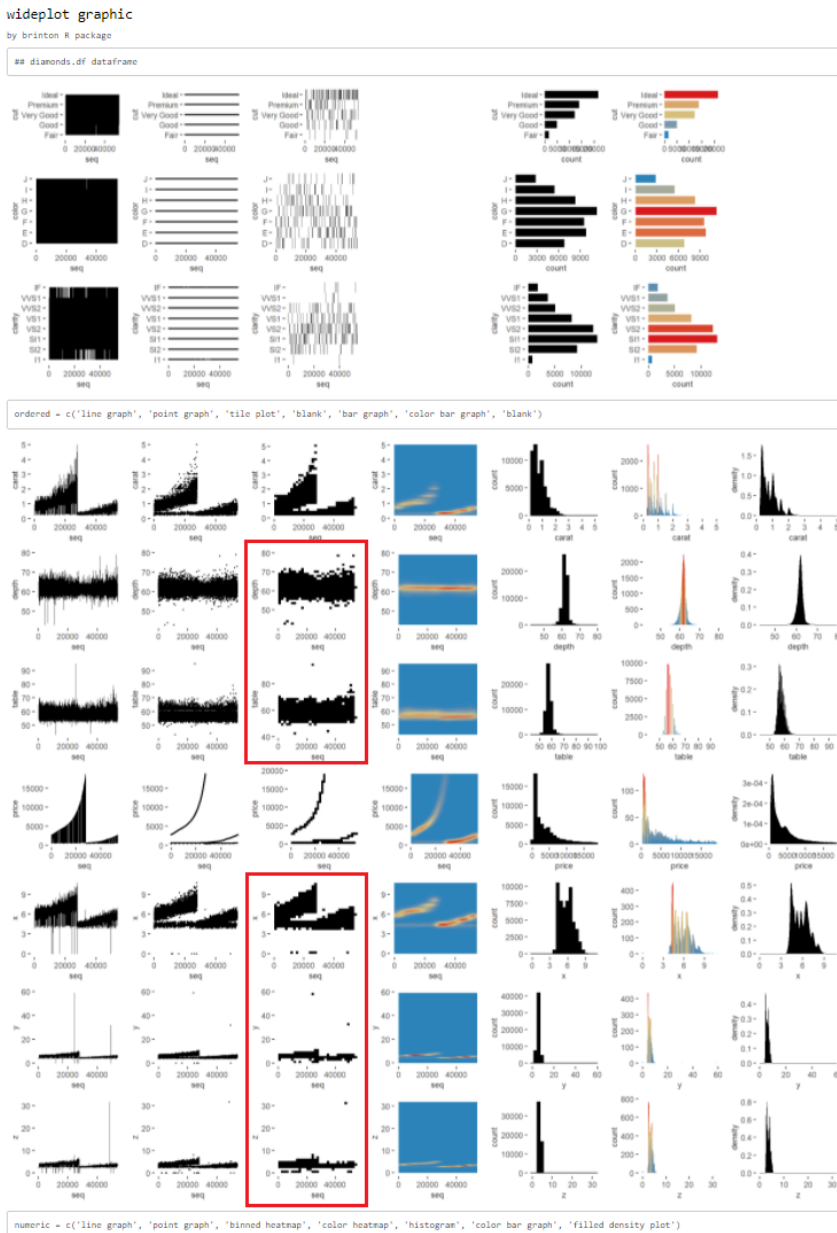


Figura 6.28: Gráfica wideplot. Función `wideplot(data = diamonds.df, dataclass = 'numeric', label = TRUE)`. Fuente: Elaboración propia.

```
matrixplot(data = diamonds.df,
           dataclass = "numeric",
           diagram = "binned heatmap")
```

matrixplot graphic

by brinton R package

```
## A matrix of "binned heatmap" produced from the c("numeric", "numeric") dataclasses of the diamonds.df dataframe
```

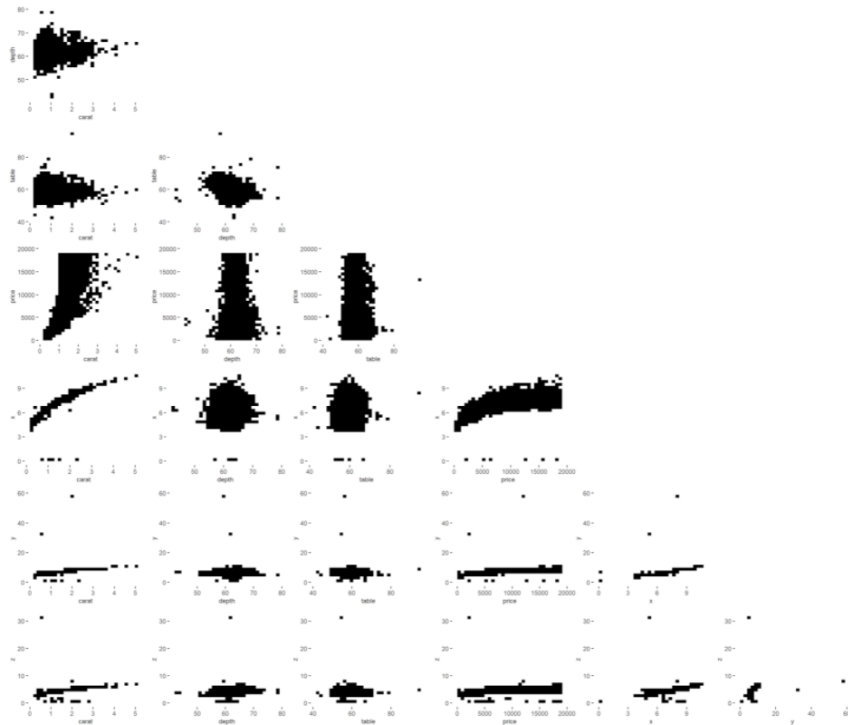


Figura 6.29: Diagrama de pares de variables monotipo. Función: `matrixplot(diamonds.df, dataclass = 'numeric', diagram = 'binned heatmap')`. Fuente: Elaboración propia.

A partir de la observación de la figura 6.29 podemos depurar el conjunto de datos excluyendo los registros que quedan por debajo o por encima de ciertos umbrales para alguna de las variables numéricas. Los registros que descartamos son aquellos que tienen informada una *x* menor de 3, una *y* mayor de 20, una *z* menor de 2 o mayor de 10, una *depth* menor de 50 o mayor de 75 y, finalmente, una *table* menor de 45 o mayor de 70.

Una vez excluidos los valores atípicos, podemos volver a llamar la función `matrixplot()` para comprobar el resultado, pero esta vez, en vez de volver a utilizar el tipo de gráfica `binned heatmap` que es especialmente útil para detectar valores atípicos, podemos utilizar el diagrama `color binned heatmap` que añade información ya que representa también el recuento de observaciones en cada celda del tamiz. En la figura 6.30 que presenta el resultado de la función `matrixplot()` una vez depurado el conjunto de datos, podemos observar cómo las relaciones entre las variables se aprecian con mucha mayor claridad que en la figura 6.29.

```
diamonds.df <- diamonds.df[diamonds.df$x >= 3, ]
diamonds.df <- diamonds.df[diamonds.df$y <= 20,]
diamonds.df <- diamonds.df[diamonds.df$z <= 10,]
diamonds.df <- diamonds.df[diamonds.df$z >= 2, ]
diamonds.df <- diamonds.df[diamonds.df$depth >= 50,]
diamonds.df <- diamonds.df[diamonds.df$depth <= 75,]
diamonds.df <- diamonds.df[diamonds.df$table >= 40,]
diamonds.df <- diamonds.df[diamonds.df$table <= 75,]
```

```
matrixplot(data = diamonds.df,
           dataclass = "numeric",
           diagram = "color binned heatmap")
```

De la observación de la figura 6.30 podemos, por ejemplo, dirigir nuestra atención hacia la relación aparentemente lineal entre las variables `x` e `y`. Para explorar con más detalle la relación entre estas dos variables podemos utilizar la función `longplot()` que presentará esta relación mediante todas las gráficas implementadas en el paquete `brinton`. Entre las gráficas reproducidas por esta función, ponemos el foco en la fila reproducida en la figura 6.31 que incluye las gráficas `contour plot`, `bw contour plot` y `color contour plot`. Estas tres gráficas evidencian la relación lineal entre estas dos variables pero además permiten identificar unos pocos valores alrededor de los cuales se concentran las observaciones.

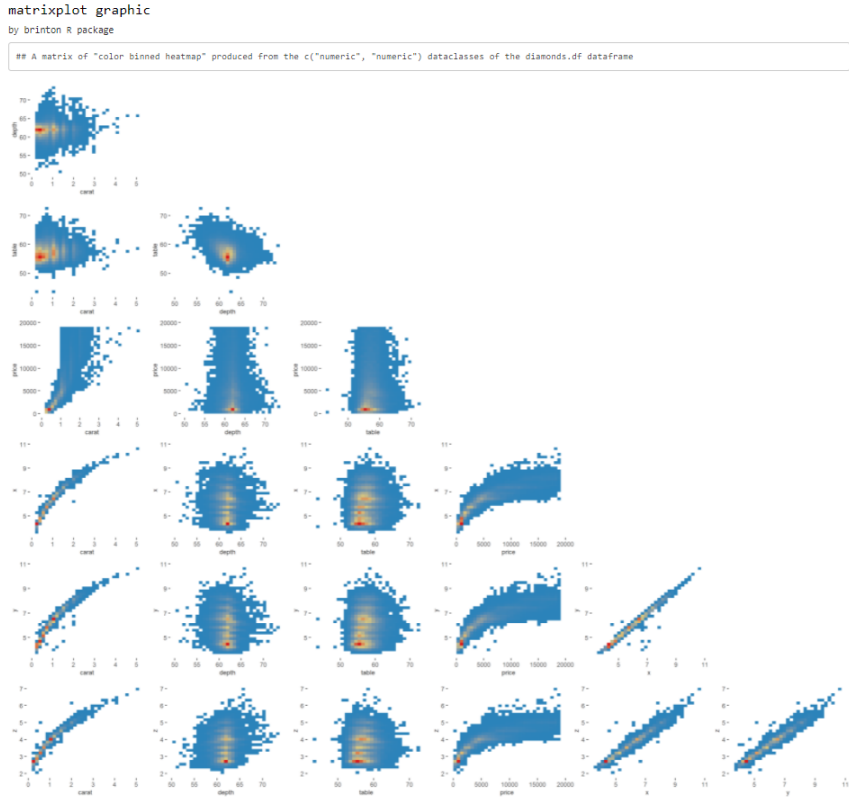
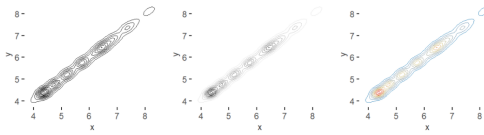


Figura 6.30: Diagrama de pares de variables monotipo. Función `matrixplot(data = diamonds.df, dataclass = 'numeric', diagram = 'color binned heatmap')`. Fuente: Elaboración propia.

```
longplot(data = diamonds.df, vars = c('x', 'y'))
```

De entre estas tres gráficas de la figura 6.31 escogemos, por ejemplo, la identificada como `color contour plot` que permite identificar los 5 puntos alrededor de los cuales se concentran las observaciones y utilizamos la función `plotup()` para reproducir esta gráfica en particular (ver figura 6.32) y también para recuperar el código necesario para reproducirla.



```
2num = c('contour plot', 'bw contour plot', 'color contour plot')
```

Figura 6.31: Sección de la gráfica longplot. Función: `longplot(data = diamonds.df, var = c('x', 'y'))`. Fuente: Elaboración propia.

```
plotup(data = diamonds.df,
       vars = c('x', 'y'),
       diagram = 'color contour plot')
```

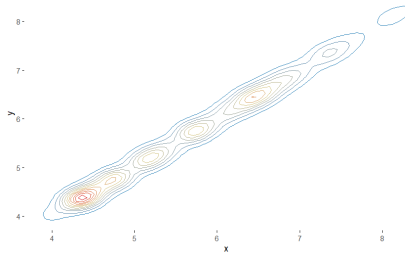


Figura 6.32: Gráfica de curvas de nivel producida con la función `plotup(data = diamonds.df, vars = c('x', 'y'), diagram = 'color contour plot')`. Fuente: Elaboración propia.

```
plotup(data = diamonds.df,
       vars = c('x', 'y'),
       diagram = 'color contour plot',
       output = 'console')
```

```
## ggplot(diamonds.df, aes(x=x, y=y)) +
##   stat_density_2d(aes(color=..level..), size=0.2) +
##   scale_color_gradientn(colours = colorRampPalette(
##     rev(RColorBrewer::brewer.pal(4, 'Spectral'))(3)) +
##   theme_minimal() +
##   theme(panel.grid = element_line(colour = NA),
##         axis.ticks = element_line(color = 'black'),
##         legend.position='none')
```

Finalmente, a partir del código que nos ha devuelto la función `plotup()`, podemos editar la función `ggplot()` para reproducir una gráfica similar a la anterior, conocida como gráfica de curvas de nivel sombreada (o `filled contour plot`) pero no incluida en el espécimen de gráficas bivariadas del paquete `brinton`. Esta figura permite identificar con bastante precisión los 5 valores de `x` e `y` que forman los núcleos principales y que son aproximadamente 4.4, 4.7, 5.2, 5.8 y 6.5mm.

```
newplot <- ggplot(diamonds.df, aes(x=x, y=y)) +
  stat_density_2d(aes(fill=..level..), size=0.2, geom = "polygon") +
  scale_fill_gradientn(colours = colorRampPalette(
    rev(RColorBrewer::brewer.pal(4, 'Spectral'))(3)) +
  theme_minimal() +
  theme(panel.grid = element_line(colour = NA),
        axis.ticks = element_line(color = 'black'),
        legend.position='none',
        panel.background = element_rect(fill = "#2B83BA", color = NA))

newplot
```

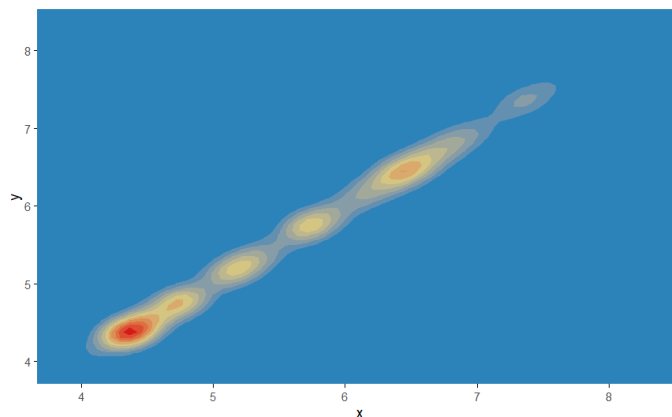


Figura 6.33: Gráfica de curvas de nivel sombreadas producida a partir del código de la función `ggplot()` devuelto por la función `plotup()`. Fuente: Elaboración propia.

Clasificación supervisada

Para este último ejemplo vamos a utilizar el conjunto de datos `olive` (Forina et al., 1983) incluido en el paquete `dslabs` ya mencionado en el apartado 6.5, que recoge datos de composición de ácidos (palmítico, palmitoleico, esteárico, oleico, linolénico, araquídico y eicosénico) de un total de 572 muestras de aceite de oliva de tres regiones de Italia (Sardeña, norte y sur de Italia) subdivididas en un total de 9 áreas. De la región de Sardeña se tienen muestras de la costa y del interior de la isla, del norte de Italia se tienen muestras de Umbria y del este y oeste de Liguria, y del sur de Italia se tienen muestras de Calabria, Sicilia y del norte y del sur de Apulia. Este ejemplo de exploración intenta clasificar las muestras de los diferentes aceites a partir de la concentración de ácidos en éstas y mediante las funciones del paquete `brinton`.

Después de cargar el paquete `dslabs` y el conjunto de datos `olive` lo primero que hacemos es consultar su estructura mediante la función `str()` y observamos que hay dos variables categóricas, `region` y `area`, que se encuentran codificadas como factores y que corresponden a dos divisiones geográficas de las muestras. Observamos también que los 8 ácidos que permiten clasificar las muestras, se corresponden con las 8 variables de tipo numérico.

```
library(dslabs)
data(olive)

str(olive)
```

```
## 'data.frame':   572 obs. of  10 variables:
## $ region      : Factor w/ 3 levels 'Northern Italy',...: 3 3 3 3 3 3 3 ...
## $ area       : Factor w/ 9 levels 'Calabria','Coast-Sardinia',...: 5 5 ...
## $ palmitic   : num  10.75 10.88 9.11 9.66 10.51 ...
## $ palmitoleic: num  0.75 0.73 0.54 0.57 0.67 0.49 0.66 0.61 0.6 0.55 ...
## $ stearic    : num  2.26 2.24 2.46 2.4 2.59 2.68 2.64 2.35 2.39 2.13 ...
## $ oleic      : num  78.2 77.1 81.1 79.5 77.7 ...
## $ linoleic   : num  6.72 7.81 5.49 6.19 6.72 6.78 6.18 7.34 7.09 6.33 ...
## $ linolenic  : num  0.36 0.31 0.31 0.5 0.5 0.51 0.49 0.39 0.46 0.26 ...
```

```
## $ arachidic : num 0.6 0.61 0.63 0.78 0.8 0.7 0.56 0.64 0.83 0.52 ...
## $ eicosenoic : num 0.29 0.29 0.29 0.35 0.46 0.44 0.29 0.35 0.33 0.3 ...
```

La función `wideplot(olive)` (ver figura 6.34) nos permite también observar su estructura y la distribución de las variables. Observamos, por ejemplo, que las variables de tipo factor no tienen valores perdidos dado que, si existieran, en las dos primeras filas de la gráfica *wideplot*, que corresponden a las dos variables categóricas, los ejes verticales mostrarían una nueva categoría llamada NA. Observamos también en los últimos paneles de las dos primeras filas, que predominan las muestras del sur de Italia (especialmente del sur de Apulia). Vemos también en los segundos paneles de las filas 4, 5, 8 y 9 que el grado de precisión en la medición de los ácidos no es uniforme (especialmente para los ácidos linolénico y araquídico de las filas 8 y 9). Finalmente, también nos llaman la atención los paneles 1, 2, y 3 de la última fila, en los que podemos contar en el eje de las abscisas, aproximadamente, unas 200 muestras que carecen de ácido eicosenoico o en las que la presencia de éste es despreciable.

```
wideplot(data = olive)
```

Para empezar a clasificar las muestras según su origen a partir de la composición de ácidos, tenemos que explorar las relaciones entre las variables que representan las regiones y los diferentes ácidos en las muestras. Para observar estas relaciones, podemos cruzar pares de variables de tipo `factor` (que corresponden a las regiones) con las variables de tipo `numeric` (que corresponden a los diferentes ácidos) mediante histogramas estratificados y esto se puede hacer fácilmente con la función `matrixplot()` que produce la figura 6.35. En la última columna de la primera fila de esta figura observamos que únicamente las muestras de aceite de una región, identificada con el color verde, contienen una concentración apreciable de ácidos eicosenoicos. Si

wideplot graphic

by brinton R package

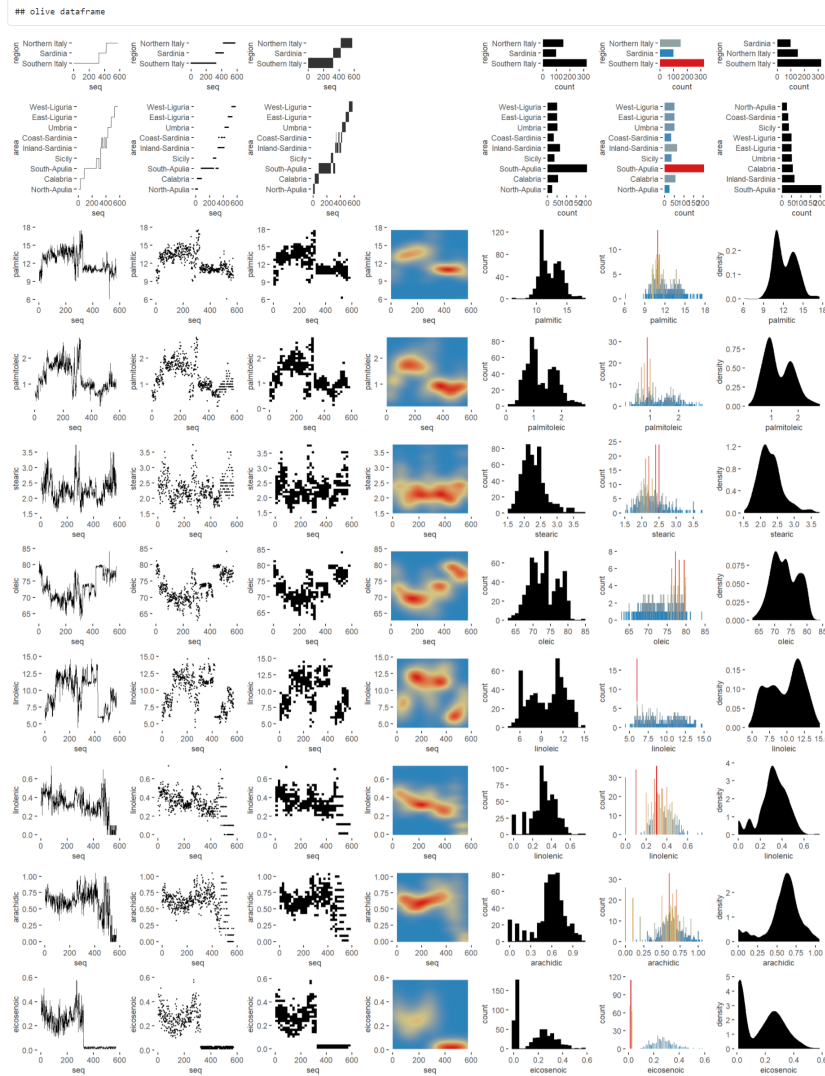


Figura 6.34: Gráfica wideplot. Función:wideplot(olive). Fuente: Elaboración propia.

luego observamos la leyenda con los colores, deducimos que estas muestras corresponden a los aceites del sur de Italia.

```
matrixplot(data = olive,
           dataclass = c('numeric', 'factor'), 'color stacked histogram')
```

matrixplot graphic

by brinton R package

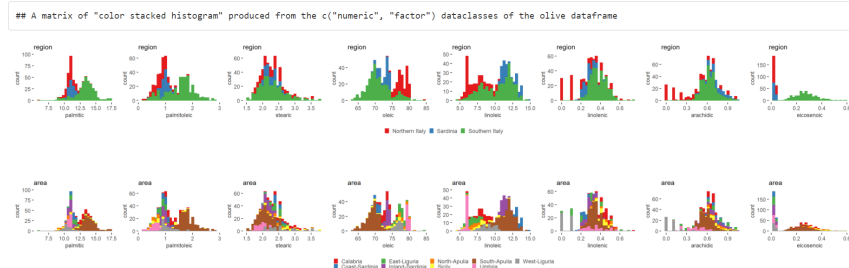


Figura 6.35: Diagrama de pares de variables de tipo cruzado. Función: `matrixplot(olive, dataclass = c('numeric', 'factor'), 'color stacked histogram')`. Fuente: Elaboración propia.

Si queremos observar con mayor detalle la gráfica que permite identificar las muestras del sur de Italia, podemos utilizar la función `plotup()` para producir la figura 6.36). En esta gráfica se observa también que las muestras del norte de Italia y de Sardinia contienen todas una concentración de este ácido próxima a cero.

Ahora nos queda encontrar el modo de clasificar las muestras de las dos regiones que quedan por identificar. Para esto, filtramos el conjunto de datos y excluimos los aceites del sur de Italia que ya sabemos clasificar inequívocamente y guardamos el resultado como `olive.filtered.A`. Podemos utilizar nuevamente la función `matrixplot()` con el propósito de relacionar los diferentes ácidos con las regiones de Sardinia y del norte de Italia (ver figura 6.37). En la quinta columna de la fila superior de la gráfica multipanel, podemos observar cómo la composición de ácidos linoleicos permite separar con bastante confianza los aceites de ambas regiones y, si observamos

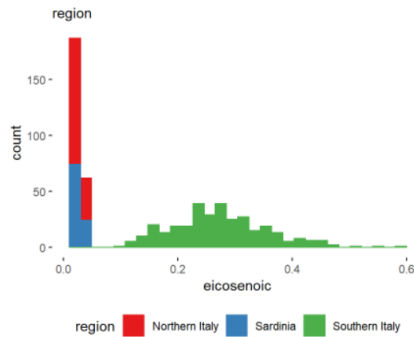


Figura 6.36: Histograma estratificado. Función: `plotup(data = olive, vars = c('eicosenoic', 'region'), diagram = 'color stacked histogram')`. Fuente: Elaboración propia.

la leyenda, vemos que los aceites del norte de Italia contienen una menor concentración de ácido linoleico. Nuevamente, si queremos observar con mayor detalle esta gráfica, la función `plotup()` nos permite reproducirla (ver figura 6.38).

```
olive.filtered.A <- olive[olive$region != 'Southern Italy',]

matrixplot(data = olive.filtered.A,
            dataclass = c('numeric', 'factor'),
            diagram = 'color stacked histogram')
```

matrixplot graphic
by brinton R package



Figura 6.37: Diagrama de pares de variables de tipo cruzado. Función: `matrixplot(olive.filtered.A, dataclass = c('numeric', 'factor'), 'color stacked histogram')`. Fuente: Elaboración propia.

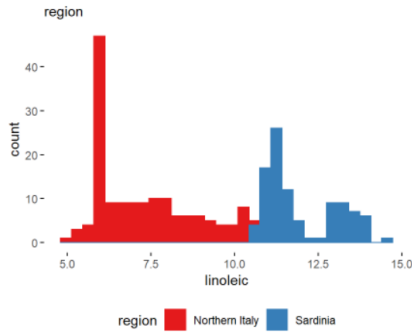


Figura 6.38: Histograma estratificado. Función: `plotup(data = olive.filtered.A, vars = c('linoleic', 'region'), diagram = 'color stacked histogram')`. Fuente: Elaboración propia.

A pesar de que el contenido de ácido linoleico es un buen indicador de la región de origen, la concentración de este ácido en las muestras de ambas regiones comparten un espacio común. Para aumentar la confianza de la clasificación supervisada podemos explorar también si las combinaciones entre pares de ácidos, (en este caso de pares de variables numéricas) nos permite diferenciar mejor su origen entre estas dos regiones. A tal efecto utilizamos la función `matrixplot()` para relacionar los diferentes ácidos mediante diagramas de dispersión y obtenemos la figura 6.39.

```
matrixplot(data = olive.filtered.A,
           dataclass = c('numeric', 'numeric'),
           diagram = 'scatter plot')
```

Entre los paneles de la figura 6.39 nos llama la atención el panel de la cuarta fila y cuarta columna que combina los ácidos oleico y linoleico porque las marcas se encuentran sensiblemente alineadas. También nos llama la atención el panel de la fila seis y columna cinco que combina los ácidos linoleico y araquídico porque muestra dos zonas claramente separadas en las que se concentran los puntos. Para explorar si estas combinaciones de ácidos nos permiten identificar

matrixplot graphic

by brinton R package

A matrix of "scatter plot" produced from the c("numeric", "numeric") dataclasses of the olive.filtered.A dataframe

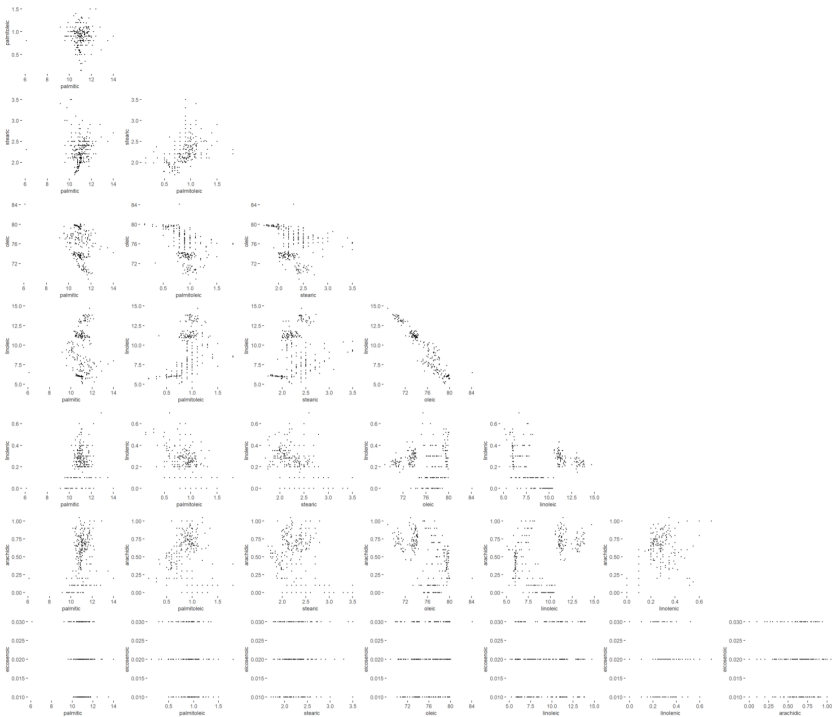


Figura 6.39: Diagrama de pares de variables monotipo. Función: `matrixplot(olive.filtered.A, dataclass = c('numeric', 'numeric'), 'scatter plot')`. Fuente: Elaboración propia.

los aceites de una y otra región, recuperamos la función que genera la gráfica de ambos paneles mediante la función `plotup()` y las modificamos para incluir la variable región. La gráfica de la figura 6.40 evidencia que es posible separar las muestras de las dos regiones mediante una línea que combina los ácidos oleico y linoleico aunque sigue existiendo una frontera difusa entre las muestras de ambas regiones. También se aprecia fácilmente que las muestras de Sardeña se concentran alrededor de dos núcleos que probablemente corresponden a las muestras de la costa y del interior de Sardeña.

```
plotup(data = olive.filtered.A,
       vars = c("linoleic", "oleic"),
       diagram = "scatter plot", output = "console")
```

```
ggplot(olive.filtered.A, aes(x=linoleic, y=oleic, color=region)) +
  geom_point() +
  scale_color_brewer(type = 'qual', palette = 'Set1') +
  theme_minimal() +
  theme(panel.grid = element_line(colour = NA),
        axis.ticks = element_line(color = 'black'))
```

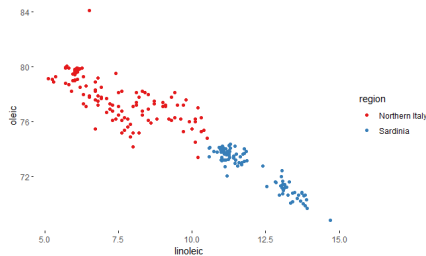


Figura 6.40: Diagrama de dispersión por grupos producido a partir del código de la función `ggplot()` devuelto por la función `plotup()`. Fuente: Elaboración propia.

En observar el diagrama de dispersión que combina los ácidos linoleico y araquídico de la figura 6.41, vemos que las dos zonas de concentración de puntos corresponden efectivamente a los aceites de las dos regiones que buscamos identificar, en conclusión, la combinación de ácidos linoleico y araquídico, permite clasificación con mayor precisión los aceites de estas regiones.

```
plotup(data = olive.filtered.A,
       vars = c("linoleic", "arachidic"),
       diagram = "scatter plot", output = "console")
```

```
ggplot(olive.filtered.A, aes(x=linoleic, y=arachidic, color=region)) +
  geom_point() +
  scale_color_brewer(type = 'qual', palette = 'Set1') +
  theme_minimal() +
  theme(panel.grid = element_line(colour = NA),
        axis.ticks = element_line(color = 'black'))
```

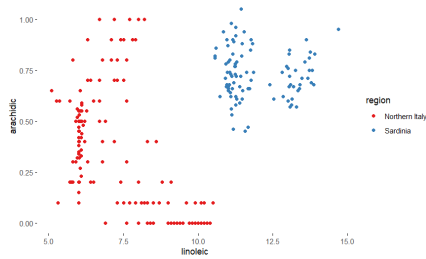


Figura 6.41: Diagrama de dispersión por grupos producido a partir del código de la función `ggplot()` devuelto por la función `plotup()`. Fuente: Elaboración propia.

Una vez hallado el modo de clasificar los aceites de las tres regiones, podríamos pasar a buscar la manera de clasificar los aceites de cada una de las áreas comprendidas en estas regiones, pero para este propósito sería necesario combinar gráficas de tres o más variables, como por ejemplo, matrices de diagramas de dispersión con puntos coloreados según la variable `area`, o diagramas de dispersión dinámicos tridimensionales que combinen a la vez los valores de tres ácidos. Dada la limitación actual de 2 variables de entrada del paquete `brinton`, preferimos dejar en este punto el ejemplo y trabajar en la ampliación del paquete y más concretamente en un próximo espécimen de gráficas trivariadas que puedan ser implementadas en las funciones `longplot()`, `matrixplot()` y `plotup()`.

CONCLUSIONES Y FUTURA LÍNEA DE INVESTIGACIÓN

“A visualization that clearly conveys the meaning of the data not only helps explain the researcher’s ideas but also cast a glow of credibility and capability.” — Peter R. Keller & Mary M. Keller

Estamos inmersos en una explosión de datos que hace necesario ampliar y mejorar los métodos que permiten extraer información de éstos. Uno de los primeros procesos para convertir datos en información se conoce como EDA (*Exploratory Data Analysis*) que consiste en observar las características de un conjunto de datos, sin poner el acento en el modelado de los datos o el contraste de hipótesis preconcebidas. Si esta exploración se sirve de gráficas que representan los datos, entonces se conoce como GEDA (*Graphical Exploratory Data Analysis*).

Observar los datos mediante gráficas, sin hipótesis preconcebidas, y que estas gráficas nos fueren a descubrir aspectos de los datos que puedan sugerir nuevas hipótesis, nos conduce a lo que se conoce como el problema gráfico: de entre el abanico de gráficas posibles... ¿cuál elegir? Aquí entran en servicio los recomendadores de gráficas estadísticas y los sistemas autoGEDA (*Automated Graphical Exploratory Data Analysis*).

Recomendación de gráficas estadísticas

La recomendación de gráficas estadísticas se puede hacer siguiendo diferentes estrategias. Por un lado, a partir de las características de los datos, como por ejemplo, el número de variables a relacionar, las características de las variables por separado, las características de

las relaciones entre éstas, la forma como se estructuran los datos y su procedencia o utilidad para la que se han recogido. Por otro lado tenemos las características de los usuarios receptores, esto es, las características de la percepción humana, la tarea a realizar por el usuario, el recuerdo de selecciones previas y las convenciones sociales. También se pueden recomendar unas u otras gráficas en función de las características del canal de comunicación, por ejemplo, debido a limitaciones en la transmisión de datos, de procesamiento o de la pantalla en la que se proyecta la gráfica y, finalmente, a partir de las características, más o menos concretas, del tipo de gráfica deseada.

Entre las estrategias que se pueden seguir para recomendar gráficas estadísticas, tienen especial relevancia el número de variables a relacionar y las características de las variables por separado. Entre las características que se pueden describir de cada una de las variables y que tienen incidencia en la selección de una u otra gráfica estadística, encontramos aspectos como, por ejemplo, la escala de medición de las variables, la consideración de éstas como predictoras o de respuesta, el número de observaciones o el número de valores diferentes observados.

ESTRATEGIA TEORIZADA Dada una selección limitada de variables de un conjunto de datos, cuanto más detallada es la caracterización de estas variables, menor es el número de gráficas estadísticas que pueden resultar de interés para el usuario. A partir de esta premisa, este trabajo ha detallado una caracterización multidimensional de las variables por separado que resulta útil para escoger qué gráficas mostrar a un usuario a partir de las características de las variables seleccionadas por éste. La caracterización propuesta considera la escala de medición gráfica, el método de agregación de los datos, la ciclicidad del espacio muestral, la conveniencia de mostrar explícitamente la escala de la variable y la longitud de ésta. A partir de la caracterización propuesta de las variables por separado y de las gráficas estadísticas a las que cada combinación de variables se puede

asociar, se establece un marco con el que se pueden clasificar las gráficas estadísticas y con el que hallar nuevos y diferentes métodos gráficos.

Con el paso del tiempo y la mayor experiencia, consideramos otra caracterización diferente a la propuesta. Esta otra solución, en vez de tratar los valores secuenciales como transformables en valores tamizados o dispersos, propone tratar esta propiedad como una nueva dimensión aplicada no a una variable en particular, sino al conjunto de los datos. De este modo, la agregación de los datos se compone de valores dispersos o tamizados que pueden a su vez caracterizarse como consecutivos o de orden arbitrario.

Este cambio añadiría claridad a la caracterización dado que la secuencia en la que se presentan los datos ordena en conjunto todas las variables y no una variable en particular. También porque unas veces, la secuencia se encuentra definida como una variable diferenciada que puede considerarse como esta nueva dimensión, mientras que otras veces, la secuencia se encuentra implícita en el orden en el que se presentan los datos. Así pues, el carácter consecutivo o de orden arbitrario equivale a considerar una variable “comodín” que determina el orden de la secuencia.

Otra consideración es que la clasificación presentada se ciñe a las gráficas estadísticas entendidas como diagramas por lo que no se ha tenido en cuenta dimensiones que podrían ser útiles para graficar mapas estadísticos, como por ejemplo la georeferenciabilidad de una variable categórica (Millán-Martínez y Valero-Mora, 2017) que permite vincular variables como “código postal” o “sección censal” respecto a diferentes polígonos en un espacio métrico. Tampoco se ha tenido en cuenta la consideración de las variables como “predictora” o “respuesta” que se ha mostrado útil para clasificar técnicas de visualización científica (Keller y Keller, 1993), o la consideración como dimensión o medida que se ha demostrado útil para la selección de diagramas (Mackinlay et al., 2007), porque estas dos caracterizaciones

se encuentran relacionadas con la consideración de las variables como dispersas o tamizadas.

Otra distinción clásica de las variables, que aquí no hemos tenido en cuenta, es su clasificación como continuas o discretas. El motivo por el que no hemos incluido esta distinción es que las variables en los datos, tienen siempre un determinado grado de precisión o una determinada carencia entre observaciones por lo que no existen, en los datos, variables puramente continuas. En nuestro caso, las variables tamizadas hubieran podido también diferenciarse entre continuas o discretas, pero esta continuidad, no referida al origen de los datos, sino a la representación de éstos. Un ejemplo sencillo pero claro de las ventajas de esta diferenciación puede ser el de una muestra de una variable numérica no acotada que, caracterizada como dispersa se podría representar mediante un diagrama uniaxial de punto pero que, caracterizada como tamizada discreta, podría dar lugar a un histograma y que, caracterizada como tamizada continua, podría dar lugar a un diagrama de densidad o de violín.

La caracterización de las variables propuesta, a pesar de las posibles mejoras a las que se pueda someter, puede ser el germen de una gramática de las gráficas que, en vez de basarse en modelos de representación, se base en las propiedades de las variables. Esto se traduciría, por ejemplo, en definir una variable como ambigua para eliminar un determinado eje de coordenadas o una determinada leyenda, definir una variable como cíclica para convertir un eje de coordenadas ortogonal en uno circular, o definir una variable como de tipo tamizado para convertir, por ejemplo, un diagrama uniaxial de punto en un histograma o un diagrama de dispersión en un mapa de calor. La gran ventaja de esta nueva gramática de las gráficas que puede derivar de la caracterización propuesta, es que pondría el acento en el conocimiento de los datos, en vez de ponerlo en el conocimiento de la gráfica buscada que, en muchos casos se desconoce. De esta gramática de las gráficas se podrían beneficiar especialmente

los programas para edición de gráficas estadísticas en el campo de la infografía, la inteligencia de negocios o la explotación gráfica de datos abiertos en portales web para usuarios no son necesariamente expertos en estadística.

No hay que perder de vista, sin embargo, que los conjuntos de datos se encuentran almacenados, generalmente, en sistemas informáticos que ya tienen caracterizadas las variables bajo un criterio que no persigue la mejor visualización de éstos sino minimizar el espacio de almacenamiento. Dada esta caracterización preestablecida, esperar que un usuario vuelva a caracterizar las variables nuevamente para obtener una gráfica, es esperar demasiado. Volver a caracterizar las variables pone una barrera entre los datos y el usuario, más si tenemos en cuenta que los usuarios no están necesariamente familiarizados con los datos.

Superar la barrera que supone tener que caracterizar los datos tiene tres posibles soluciones. La primera solución pasa por aprovechar la caracterización preestablecida de los datos para, en base a ésta, sugerir las gráficas estadísticas. La segunda solución pasa por hacer suposiciones en base a los datos, de modo que la caracterización de las variables sea transparente para el usuario, que en caso de ser errónea, podría modificar. La tercera solución pasaría por almacenar los datos primando su posible explotación gráfica en vez de primar el espacio necesario en un disco duro o cualquier otro soporte. Esta última solución, que es la más improbable, supondría añadir información en los conjuntos de datos sobre la existencia de un orden y una distancia entre los valores observados. Si hay orden entre los valores interesaría conocer también el rango, la ciclicidad y la cardinalidad del espacio muestral. También resultaría interesante identificar la existencia de variables índice o las variables utilizadas para agregar el resto de registros y añadir, si fuera necesario, variables que fijarían la secuencia de las observaciones o variables con el recuento de registros agregados.

ESTRATEGIA IMPLEMENTADA De entre las posibles soluciones para evitar que el usuario se vea obligado a caracterizar las variables, nosotros hemos seguido la primera, en nuestro caso, aprovechar la caracterización preestablecida en ámbito específico del entorno de programación estadística **R**. En este trabajo hemos presentado el paquete `brinton` que incluye las funciones `wideplot()`, `longplot()`, `matrixplot()` y `plotup()` que presentan de manera automática gráficas estadísticas, asisten al usuario en la exploración de los conjuntos de datos mediante gráficas univariadas y bivariadas, a la vez que facilitan la elección, edición y representación de una determinada gráfica por parte del usuario.

Cada función del paquete `brinton` añade una alternativa novedosa en el ámbito de la exploración gráfica automatizada de datos y las funciones, en conjunto, facilitan y aceleran el proceso de generación de información a partir de un conjunto de datos. En un futuro cercano, la utilidad del paquete `brinton` será reforzada mediante la incorporación de nuevas especies en los especímenes de gráficas univariadas y bivariadas así como la incorporación de un nuevo espécimen de gráficas trivariadas y nuevas funciones que complementen las existentes.

Futuras líneas de investigación

Dado el abanico de gráficas que el paquete `brinton` proporciona y la facilidad con la que los usuarios pueden elegir entre una u otra gráfica, una futura línea de investigación es relacionar las diferentes gráficas con la utilidad que tienen para los usuarios.

Cada gráfica estadística puede dar respuesta, con mayor o menor certeza, a diferentes preguntas o propósitos de los usuarios. Conocer el propósito de éstos en el momento de elegir una determinada gráfica y conocer su grado de satisfacción con la gráfica escogida, permitiría añadir mayor precisión en la recomendación de gráficas estadísticas dado que el abanico de gráficas a mostrar, se reduciría a aquellas que

son compatibles con las características de las variables seleccionadas y que ofrecen mejor expectativa de satisfacción para la utilidad que el usuario espera.

BIBLIOGRAFÍA

- Auguie, B. (2017). *gridExtra: Miscellaneous Functions for “Grid” Graphics*. R package version 2.3.
- Auguie, B. (2019). *egg: Extensions for 'ggplot2': Custom Geom, Custom Themes, Plot Alignment, Labelled Panels, Symmetric Scales, and Fixed Panel Size*. R package version 0.4.5.
- Bachi, R. (1968). *Graphical rational patterns: A new approach to graphical presentation of statistics*. Transaction Publishers.
- Becker, R. A., Cleveland, W. S., y Shyu, M.-J. (1996). The visual design and control of trellis display. *Journal of computational and Graphical Statistics*, 5(2):123–155.
- Belgrano, J. C., López, A., y Urcelay, J. M. (1953). *Tratado de nomografía*. Instituto técnico de la construcción y del cemento, Madrid.
- Benger, W. y Hege, H.-C. (2006). *Strategies for Direct Visualization of Second-Rank Tensor Fields*, pages 191–214. Springer Berlin Heidelberg, Berlin, Heidelberg.
- Benson, W. H. y Kitous, B. (1977). Interactive analysis and display of tabular data.[descripton of chart program for report design]. Technical report, California Univ., Berkeley (USA). Lawrence Berkeley Lab.
- Bertin, J. (1967). *Sémiologie graphique*. Mouton, Paris.
- Bertin, J. (1977). *La graphique et le traitement graphique de l'information*. Nouvelle Bibliothèque Scientifique. Flammarion.

- Borner, K. (2015). *Atlas of knowledge: Anyone can map*. MIT Press.
- Brinton, W. (1939). *Graphic Presentation*. Brinton associates.
- Cairo, A. (2011). *El arte funcional: Infografía y visualización de información*. Alamut, Madrid.
- Card, S. (2007). Information visualization. In Sears, A. y Jacko, J. A., editors, *The human-computer interaction handbook: fundamentals, evolving technologies and emerging applications*, chapter 26, pages 509–542. CRC press.
- Card, S., MacKinlay, J., y Shneiderman, B. (1999). *Readings in Information Visualization: Using Vision to Think*. The Morgan Kaufmann series in interactive technologies. Morgan Kaufmann Publ.
- Carr, D. (1994). Using gray in plots. *Statistical Computing & Graphics Newsletter*, 5(2):11–14.
- Casas, P. (2020). *funModeling: Exploratory Data Analysis and Data Preparation Tool-Box*. R package version 1.9.4.
- Casner, S. M. (1991). Task-analytic approach to the automated design of graphic presentations. *ACM Transactions on Graphics (TOG)*, 10(2):111–151.
- Chang, W. y Borges Ribeiro, B. (2018). *shinydashboard: Create Dashboards with 'Shiny'*. R package version 0.7.1.
- Cleveland, W. (1985). *The Elements of Graphing Data*. Hobart Press, Summit, New Jersey.
- Cleveland, W. S. y McGill, R. (1984). Graphical perception: Theory, experimentation, and application to the development of graphical methods. *Journal of the American statistical association*, 79(387):531–554.

- Comtois, D. (2019). *summarytools: Tools to Quickly and Neatly Summarize Data*. R package version 0.9.3.
- Cook, D., Swayne, D. F., y Buja, A. (2007). *Interactive and dynamic graphics for data analysis: with R and GGobi*. Springer Science & Business Media.
- Costa, J. (1998). *La esquemática: Visualizar la información*. Paidós Estética. Paidós, Barcelona.
- Cui, B. (2019). *DataExplorer: Automate Data Exploration and Treatment*. R package version 0.8.0.
- Dang, T. N. y Wilkinson, L. (2014). Scagexplorer: Exploring scatterplots by their scagnostics. In *2014 IEEE Pacific visualization symposium*, pages 73–80. IEEE.
- Dayanand Ubrangala, R. K., Prasad Kondapalli, R., y Putatunda, S. (2019). *SmartEDA: Summarize and Explore the Data*. R package version 0.3.2.
- Emerson, J. W., Green, W. A., Schloerke, B., Crowley, J., Cook, D., Hofmann, H., y Wickham, H. (2013). The generalized pairs plot. *Journal of Computational and Graphical Statistics*, 22(1):79–91.
- Escribano, J. J. (2003). La nomografía: una ciencia olvidada. *Números. Revista de Didáctica de las Matemáticas*, 54:41–49.
- Forina, M., Armanino, C., Lanteri, S., y Tiscornia, E. (1983). Classification of olive oils from their fatty acid composition. In *Food research and data analysis: proceedings from the IUFoST Symposium, September 20-23, 1982, Oslo, Norway/edited by H. Martens and H. Russwurm, Jr*. London: Applied Science Publishers, 1983.
- Friendly, M. (2009). Milestones in the history of thematic cartography, statistical graphics, and data visualization.

- Friendly, M. (2014). Comment on “the generalized pairs plot”. *Journal of Computational and Graphical Statistics*, 23(1):290–291.
- Friendly, M. (2018). Lecture 2: Standard graphics in r. OpenCourseWare.
- Friendly, M. y Denis, D. J. (2006). Milestones in the history of thematic cartography, statistical graphics, and data visualization.
- Friendly, M. y Meyer, D. (2015). *Discrete Data Analysis with R: Visualization and Modeling Techniques for Categorical and Count Data*. Chapman & Hall/CRC Texts in Statistical Science. CRC Press.
- Gassen, J. (2020). *ExPanDaR: Explore Your Data Interactively*. R package version 0.5.3.
- Gnanamgari, S. (1981). *Information presentation through default displays*. University of Pennsylvania. Ph.D. dissertation.
- Gotz, D. y Wen, Z. (2009). Behavior-driven visualization recommendation. In *Proceedings of the 14th international conference on Intelligent user interfaces*, pages 315–324.
- Hartigan, J. A. (1975). Printer graphics for clustering. *Journal of Statistical Computation and Simulation*, 4(3):187–213.
- Heinzen, E., Sinnwell, J., Atkinson, E., Gunderson, T., y Dougherty, G. (2021). *arsenal: An Arsenal of 'R' Functions for Large-Scale Statistical Summaries*. R package version 3.6.3.
- Hu, K., Bakker, M. A., Li, S., Kraska, T., y Hidalgo, C. (2019). Vizml: A machine learning approach to visualization recommendation. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, pages 1–12.

- Iannone, R., Allaire, J., y Borges, B. (2018). *flexdashboard: R Markdown Format for Flexible Dashboards*. R package version 0.5.1.1.
- Jansen, Y. (2014). *Physical and tangible information visualization*. PhD thesis, Université Paris Sud-Paris XI.
- Kamps, T. (1999). *Diagram Design: A Constructive Theory*. Springer Berlin Heidelberg.
- Keim, D., Kohlhammer, J., Ellis, G., y Mansmann, F. (2010). Mastering the information age solving problems with visual analytics.
- Keller, P. y Keller, M. (1993). *Visual Cues; Practical Data Visualization*. IEEE Computer Society Press.
- Krasser, R. (2021). *explore: Simplifies Exploratory Data Analysis*. R package version 0.7.1.
- Lee, D. J.-L., Dev, H., Hu, H., Elmeleegy, H., y Parameswaran, A. (2019). Avoiding drill-down fallacies with vispilot: Assisted exploration of data subsets. In *Proceedings of the 24th International Conference on Intelligent User Interfaces*, pages 186–196.
- Levkowitz, H. (1997). *Color theory and modeling for computer graphics, visualization, and multimedia applications*. Springer.
- Levkowitz, H. y Herman, G. T. (1987). Towards an optimal color scale. *Computer Graphics*, 87:92–98.
- Lin, H., Moritz, D., y Heer, J. (2020). *Dziban: Balancing Agency and Automation in Visualization Design via Anchored Recommendations*, page 1–12. Association for Computing Machinery, New York, NY, USA.
- Lu-Yao, G. L., Albertsen, P. C., Moore, D. F., Shih, W., Lin, Y., DiPaola, R. S., Barry, M. J., Zietman, A., O’Leary, M., Walker-Corkery, E., et al. (2009). Outcomes of localized prostate cancer following conservative management. *Jama*, 302(11):1202–1209.

- MacEachren, A. (2004). *How Maps Work: Representation, Visualization, and Design*. Guilford Publications.
- Mackinlay, J. (1986). Automating the design of graphical presentations of relational information. *ACM Transactions on Graphics*, 5(2):110–141.
- Mackinlay, J. y Genesereth, M. R. (1985). Expressiveness and language choice. *Data & Knowledge Engineering*, 1(1):17–29.
- Mackinlay, J., Hanrahan, P., y Stolte, C. (2007). Show me: Automatic presentation for visual analysis. *IEEE Transactions on Visualization and Computer Graphics*, 13(6):1137–1144.
- Mann, M. E., Bradley, R. S., y Hughes, M. K. (1999). Northern hemisphere temperatures during the past millennium: Inferences, uncertainties, and limitations. *Geophysical research letters*, 26(6):759–762.
- Millán-Martínez, P. y Oller, R. (2020). A graphical eda tool with ggplot2: brinton. *R Journal*, 12(2).
- Millán-Martínez, P. y Valero-Mora, P. (2017). A strategy for automating the presentation of statistical graphics for users without data visualization expertise - a position paper. In *Proceedings of the 12th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP 2017)*, pages 294–298.
- Millán-Martínez, P. y Valero-Mora, P. (2018). Automating statistical diagrammatic representations with data characterization. *Information Visualization*, 17(4):316–334.
- Moore, D. (2016). *Applied Survival Analysis Using R*. Use R! Springer International Publishing.

- Mosteller, F. y Tukey, J. (1977). *Data analysis and regression: a second course in statistics*. Addison-Wesley series in behavioral science. Addison-Wesley Pub. Co.
- Mutlu, B., Veas, E., Trattner, C., y Sabol, V. (2015). Vizrec: A two-stage recommender system for personalized visualizations. In *Proceedings of the 20th International Conference on Intelligent User Interfaces Companion*, IUI Companion '15, pages 49–52, New York, NY, USA. ACM.
- Nuñez, J. R., Anderton, C. R., y Renslow, R. S. (2018). Optimizing colormaps with consideration for color vision deficiency to enable accurate interpretation of scientific data. *PLOS ONE*, 13(7):1–14.
- Oliver, R. L. (1977). Effect of expectation and disconfirmation on postexposure product evaluations: An alternative interpretation. *Journal of applied psychology*, 62(4):480.
- Paley, B. W. (2006). Mapping scientific paradigms.
- Palsky, G. (2017). La sémiologie graphique de Jacques Bertin a cinquante ans. *visioncarto*.
- Pedersen, T. L. (2020). *patchwork: The Composer of Plots*. R package version 1.1.1.
- Perin, C., Dragicevic, P., y Fekete, J.-D. (2014). Revisiting bertin matrices: New interactions for crafting tabular visualizations. *IEEE transactions on visualization and computer graphics*, 20(12):2082–2091.
- Perin, C., Fekete, J.-D., y Dragicevic, P. (2018). Jacques bertin’s legacy in information visualization and the reorderable matrix. *Cartography and Geographic Information Science*, 46.

- Petersen, A. H. y Ekstrøm, C. T. (2019). datamaid: Your assistant for documenting supervised data quality screening in r. *Journal of Statistical Software*, 90(6):1–38.
- Pyle, D. (1999). *Data preparation for data mining*. Morgan Kaufmann Publishers, San Francisco.
- Ramón y Cajal, S. (1923). The cajal legacy.
- Rao, R. y Card, S. K. (1994). The table lens: Merging graphical and symbolic representations in an interactive focus + context visualization for tabular information. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '94, pages 318–322, New York, NY, USA. ACM.
- Riche, N., Hurter, C., Diakopoulos, N., y Carpendale, S. (2018). *Data-Driven Storytelling*. AK Peters Visualization Series. CRC Press.
- Roth, R. E. (2017). *Visual Variables*, pages 1–11. John Wiley and Sons, Ltd.
- Roth, S. F., Kolojejchick, J., Mattis, J., y Goldstein, J. (1994). Interactive graphic design using automatic presentation knowledge. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 112–117. ACM.
- Roth, S. F. y Mattis, J. (1990). Data characterization for intelligent graphics presentation. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 193–200. ACM.
- Rushworth, A. (2019). *inspectdf: Inspection, Comparison and Visualisation of Data Frames*. R package version 0.0.4.
- Ryu, C. (2021). *dlookr: Tools for Data Diagnosis, Exploration, Transformation*. R package version 0.5.4.

- Sarawagi, S., Agrawal, R., y Megiddo, N. (1998). Discovery-driven exploration of olap data cubes. In *International Conference on Extending Database Technology*, pages 168–182. Springer.
- Sarkar, D. (2008). *Lattice: Multivariate Data Visualization with R*. Springer, New York. ISBN 978-0-387-75968-5.
- Schloerke, B. (2017). *Generalized Plot Matrices, Automatic Cognos-tics, and Efficient Data Exploration*. PhD thesis, Purdue University.
- Schulz, H.-J., Nocke, T., Heitzler, M., y Schumann, H. (2016). A systematic view on data descriptors for the visual analysis of tabular data. *Information Visualization*.
- Seibelt, P. (2017). *xray: X Ray Vision on your Datasets*. R package version 0.2.
- Seo, J. y Shneiderman, B. (2005). A rank-by-feature framework for interactive exploration of multidimensional data. *Information visualization*, 4(2):96–113.
- Shneiderman, B. (1996). The eyes have it: a task by data type taxonomy for information visualizations. In *Visual Languages, 1996. Proceedings., IEEE Symposium on*, pages 336–343.
- Staniak, M. y Biecek, P. (2019). The Landscape of R Packages for Automated Exploratory Data Analysis. *The R Journal*, 11(2):347–369.
- Stevens, S. S. (1946). On the theory of scales of measurement. *Science*, 103:677–680.
- Tennekes, M., de Jonge, E., Daas, P. J., et al. (2013). Visualizing and inspecting large datasets with tableplots. *Journal of Data Science*, 11(1):43–58.

- Therneau, T. M. (2015). *A Package for Survival Analysis in S*. version 2.38.
- Theus, M. (2016). *Trellis Displays*, pages 1–10. John Wiley and Sons, Ltd.
- Theus, M. y Urbanek, S. (2008). *Interactive Graphics for Data Analysis: Principles and Examples (Computer Science and Data Analysis)*. Chapman & Hall/CRC.
- Tierney, N. (2017). visdat: Visualising whole data frames. *JOSS*, 2(16):355.
- Treinish, L. A. y Rothfus, L. P. (1997). Three-dimensional visualization for support of operational forecasting at the 1996 centennial olympic games. In *Proceedings of the 13th IIPS Conference*, pages 2–7.
- Tufte, E. R. (1983). *The Visual Display of Quantitative Information*. Graphics Press, Cheshire.
- Tukey, J. (1977). *Exploratory Data Analysis*. Addison-Wesley series in behavioral science. Addison-Wesley Publishing Company.
- Ullman, J. (1980). *Principles of database systems*. Computer software engineering series. Computer Science Press.
- Unwin, A., Theus, M., y Hofmann, H. (2006). *Graphics of Large Datasets: Visualizing a Million (Statistics and Computing)*. Springer-Verlag, Berlin, Heidelberg.
- Valero-Mora, P., Ledesma, R. D., y Friendly, M. (2012). The history of vista: The visual statistics system. *Wiley Interdisciplinary Reviews: Computational Statistics*, 4(3):295–306.

- Vartak, M., Rahman, S., Madden, S., Parameswaran, A., y Polyzotis, N. (2015). Seedb: Efficient data-driven visualization recommendations to support visual analytics. In *Proceedings of the VLDB Endowment International Conference on Very Large Data Bases*, volume 8, page 2182. NIH Public Access.
- von Engelhardt, J. (2002). *The Language of Graphics: A Framework for the Analysis of Syntax and Meaning in Maps, Charts and Diagrams*. PhD thesis, University of Amsterdam, Institute for Logic, Language and Computation.
- Ware, C. (2004). Figure credits. In Ware, C., editor, *Information Visualization*, Interactive Technologies, pages xv–xvi. Academic Press, San Diego, second edition.
- Wickham, H. (2009). *ggplot2: Elegant Graphics for Data Analysis*. Springer-Verlag New York.
- Wickham, H. (2014). Tidy data. *Journal of statistical software*, 59(1):1–23.
- Wickham, H. (2016). *ggplot2: Elegant Graphics for Data Analysis*. Springer-Verlag New York.
- Wilke, C. O. (2020). *cowplot: Streamlined Plot Theme and Plot Annotations for 'ggplot2'*. R package version 1.1.1.
- Wilkinson, L. (1999). *The Grammar of Graphics*. Springer-Verlag New York, Inc., New York, NY, USA.
- Wilkinson, L. (2005). *The Grammar of Graphics (Statistics and Computing)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2nd edition.
- Wilkinson, L. y Anand, A. (2018). *scagnostics: Compute scagnostics - scatterplot diagnostics*. R package version 0.2-4.1.

- Wilkinson, L., Anand, A., y Grossman, R. (2005). Graph-theoretic scagnostics. In *IEEE Symposium on Information Visualization, 2005. INFOVIS 2005.*, pages 157–164.
- Wills, G. y Wilkinson, L. (2010). Autovis: automatic visualization. *Information Visualization*, 9(1):47–69.
- Wongsuphasawat, K., Moritz, D., Anand, A., Mackinlay, J., Howe, B., y Heer, J. (2016). Towards a general-purpose query language for visualization recommendation. In *ACM SIGMOD Human-in-the-Loop Data Analysis (HILDA)*.
- Xie, Y., Allaire, J., y Grolemond, G. (2018). *R Markdown: The Definitive Guide*. Chapman and Hall/CRC, Boca Raton, Florida. ISBN 9781138359338.
- Zipf, G. K. (1935). *The Psychobiology of Language*. Routledge, New York, NY, USA.