¿Casualidad o detalle feo?

Uno.

Cuando empezamos a trabajar con el concepto de *profundidad* en datos funcionales, utilizábamos un código programado por Sara López-Pintado. Los cálculos no eran rápidos, en parte por el gran número de comparaciones de pares de funciones que se requiere hacer. En el código esto se hace de forma sencilla , pero no muy eficaz, con bucles anidados. Andrés y yo estuvimos un tiempo pensando cómo aumentar la velocidad. En el archivo «IdeasProfundidad», con fecha 23 de enero del 2006, escribí lo siguiente:

> Procedimiento 3
> La versión generalizada de la profundidad, como las dos anteriores propuestas, no aprecia picos estrechos y pronunciados. Esto puede añadir robustez frente a datos anómalos, pero no aprecia datos atípicos que pueden ser característicos, por lo que se pierde información. Una definición de profundidad que tiene en cuenta más directamente la forma de la curva es ir haciendo cortes verticales en las curvas de cada grupo. Para cada corte se mide lo cercana que está cada curva (punto de corte) a la media. Ahora hay dos variantes, considerar esas distancias tal cual (Procedimiento 3.1), como números reales, o sólo los órdenes de las curvas respecto a él (Procedimiento 3.2), es decir, como números naturales. Después de hacer estos cortes, una curva tiene atribuido un vector de valores: cada uno de esos valores es el orden de profundidad de esa curva en esa sección. En general una curva es más profunda cuanto menores son estos valores del vector, y esto se puede medir por el cuadrado de la norma euclídea. Hay varias ventajas con esta definición:
>
> i. Tiene en cuenta la altura de los picos
> ii. Se pueden aprovechar los cálculos anteriores cuando añadimos más cortes verticales. Esto permite seleccionar variables cada 2n y añadir después más si es necesario
> iii. Muy rápido de calcularse
>
> (Lo que se hace en López-Pintado y Romo (2005) hace algo así, pero sólo teniendo en cuenta en cada corte el orden de la curva y los puntos máximo y mínimo. En cada corte se asigna 1 a la curva si está entre ellos y 0 si no; del vector de cada curva, de ceros y unos, la asignación final es la suma de los unos, en el caso de la profundidad generalizada, o un 1 si todos los elementos de ese vector han sido unos y un 0 si no. Finalmente, por esta asignación, la profundidad es proporcional a la suma de esas cantidades.)

Dos.

Ninguna idea de ese archivo pareció llamar la atención de mis directores de tesis. Andrés y yo seguimos trabajando en una metodología para hacer los cálculos, donde cada comparación de pares de curvas tuviese que hacerse sólo una vez. Hicimos código que utilizaba hipermatrices, pero lo abandonamos cuando supimos que ya había algo hecho (ver el siguiente punto).

Tres.

En la lectura de tesis de Aurora, doctoranda de Juan, nos enteramos Andrés y yo de que ella había encontrado una forma de hacer los cálculos de forma eficaz. A mí –y creo que a Andrés también– me sorprendió que Juan no nos hubiese informado de que el problema estaba razonablemente resuelto, o, al menos, de que era material de la tesis de Aurora.

Cuatro.

Al redactar el primer artículo de mi tesis, incluí la frase «Nevertheless, it is possible to do these comparisons only once by implementing López-Pintado and Romo (2006)'s method conveniently; such an implementation allows o using their methods with bigger samples sizes.». Sara, coautora que revisó el artículo, cambió la frase por «However, we have conveniently implemented the notion of depth in López-Pintado and Romo (2006) so it is computationally feasible and applicable to high o sample sizes.». Creo que esto es falso y ha sido Aurora quien lo implementó correctamente.

Cinco.

Como recordaba el resultado de Aurora, me parecía que quizá debíamos citarla. En un correo

electrónico incluí hace poco el siguiente texto:

3) El último párrafo de la sección 4 me parece que ha mejorado mucho. Fue en el seminario de la tesis de Aurora donde me enteré de que en una buena implementación bastaba con hacer las comparaciones una vez, si hay que meter alguna referencia nueva lo indicáis, por favor.

Ni Sara ni Juan hicieron ningún comentario.

Seis.

Hace poco he consultado en la tesis de Aurora, el apartado «3.2.1. Un cálculo eficiente de la profundidad por bandas generalizada», donde se incluye una proposición que describe una forma de hacer más eficaz el cálculo escribiendo:

$$
\begin{aligned}
GBD(y_i) &= \frac{1}{\binom{n}{2}} \sum_{1\le i_1<i_2\le n} \frac{1}{d}\sum_{k=1}^{d} I_{\{min(y_{i_1,k},y_{i_2,k})\le y_{i,k}\le max(y_{i_1,k},y_{i_2,k})\}} = \\
&= \frac{1}{\binom{n}{2}}\frac{1}{d}\sum_{k=1}^{d}\sum_{1\le i_1<i_2\le n} I_{\{min(y_{i_1,k},y_{i_2,k})\le y_{i,k}\}} I_{\{max(y_{i_1,k},y_{i_2,k})\ge y_{i,k}\}} = \\
&= \frac{1}{\binom{n}{2}}\frac{1}{d}\sum_{k=1}^{d}\sum_{1\le i_1<i_2\le n} \left(I_{\{min(y_{i_1,k},y_{i_2,k})< y_{i,k}\}} + I_{\{min(y_{i_1,k},y_{i_2,k})= y_{i,k}\}}\right) \times \\
&\times \left(I_{\{max(y_{i_1,k},y_{i_2,k})> y_{i,k}\}} + I_{\{max(y_{i_1,k},y_{i_2,k})= y_{i,k}\}}\right) = \\
&= \frac{1}{d\times\binom{n}{2}}\sum_{k=1}^{d}[(l_k-1)\cdot(n-(l_k+\eta_k-1)) + (l_k-1)\cdot\eta_k + \\
&+ \eta_k\cdot(n-(l_k+\eta_k-1)) + \binom{\eta_k}{2}] = \\
&= \frac{1}{d\times\binom{n}{2}}\sum_{k=1}^{d}\left((n-l_k+1)(l_k-1+\eta_k) - \eta_k^2 + \binom{\eta_k}{2}\right).
\end{aligned}
$$

■

Título: Métodos de clustering en datos de expresión génica
Autor(es): Torrente Orihuela, Aurora
Director(es): Romo, Juan
Brazma, Alvis
Editor: Universidad Carlos III de Madrid. Departamento de Estadística
http://e-archivo.uc3m.es/dspace/bitstream/10016/2387/1/Tesis%20Aurora%20Torrente.pdf

Final.

Los cálculos anteriores recuerda mucho mi idea.

6 de junio del 2009
David Casado de Lucas