Revista de Matemática: Teoría y Aplicaciones 4(1): 77–85 (1997)

PARTICIÓN ÓPTIMA: EL ALGORITMO DE FISHER

José Luis Espinoza[1]

Resumen

El algoritmo de Fisher es un algoritmo que calcula exactamente una partición óptima en k clases de un conjunto Ω de n individuos a los que se les ha medido una variable real v. Tal clasificación que se obtiene, aunque no es necesariamente única, es óptima respecto a v y, en el sentido de minimizar la inercia intra-clase, está formada por clases contiguas. Además, se estudia un criterio para estimar el número óptimo de clases en que puede clasificarse el conjunto de datos respecto a v. Se presenta una implementación computacional del algoritmo, así como algunos resultados numéricos.

Palabras clave: partición, optimización, codificación.

1 Introducción

El algoritmo de Fisher está inspirado en la programación dinámica. En términos generales, la programación dinámica requiere optimizar un criterio numérico aditivo W que dependa de m pasos. Usualmente el proceso de optimización bajo la programación dinámica se desarrolla desde el final hasta el inicio: En primer lugar, se planifica el último (m-ésimo) paso. Al planificar el último paso es necesario hacer suposiciones acerca de cómo terminó el penúltimo ($m-1$ ésimo) paso y para cada uno de los diferentes casos que se obtienen, tomar la decisión óptima en el último paso. Igualmente se procede al momento de planificar el ante-penúltimo paso, con la ventaja de que por cada uno de los casos a que conduzca, ya se saben las decisiones en los dos últimos pasos que contribuyen a optimizar W. Se procede así sucesivamente hasta llegar al primer paso ([7]). Según Hartigan ([5]), lo medular del algoritmo de Fisher es la relación entre particiones óptimas en k clases y las particiones óptimas en $k-1$ clases.

Pese a que el algoritmo de Fisher tiene la limitación de producir una clasificación óptima sobre datos univariados, tiene gran importancia como método previo a otras técnicas de particionamiento para datos multivariados, tales como los métodos de segmentación [4] y los métodos de discriminación, tomando la variable cuantitativa como la

[1]Departamento de Matemáticas, Instituto Tecnológico de Costa Rica, Cartago, Costa Rica.

variable a explicar y las demás como variables explicativas. En aplicaciones que tienen que ver con la codificación y manejo de encuestas viene a ser una herramienta útil para la separación de variables cuantitativas tales como *salario* o *estatura* en rangos calculados de manera óptima.

Para empezar, partimos de un conjunto $\Omega = \{\omega_1, \omega_2, ..., \omega_n\}$ de n individuos u objetos de cualquier tipo a los que se les ha medido una variable cuantitativa $v : \Omega \to I\!R$. El objetivo es clasificar los individuos de Ω en k clases que, en el sentido de los valores que toma v, sean lo más separadas entre sí y homogéneas internamente. Por simplicidad, supongamos que los individuos están ordenados, según v, en forma creciente y que $v_1 < v_2 < ... < v_m$ son los m valores distintos que toma v en Ω ($m \leq n$). Es claro que si un subconjunto de individuos de Ω tienen el mismo valor de v, éstos deberán aparecer en una misma clase, por lo que con sólo uno de ellos que sea clasificado, será suficiente para la clasificación del resto. De esta manera, clasificar los individuos $\omega_1, \omega_2, ..., \omega_n$ de acuerdo a v equivale a clasificar los números reales $v_1, v_2, ..., v_m$ que pueden denotarse por los subíndices o identificadores: $\{1, 2, ..., m\}$.

2 Notación y terminología

- $P(m, k)$ denota una partición de Ω en k clases, donde $1 < k < m$.
- $p_1, p_2, ..., p_m$ denotan los pesos de los individuos, con $\sum_{i=1}^{m} p_i = 1$.
- $C_1, C_2, ..., C_k$ denotan las clases que forman $P(m, k)$.
- $n_1, n_2, ..., n_k$ el efectivo de cada clase, con $\sum_{j=1}^{k} n_j = m$.
- El criterio a minimizar es la *inercia intra-clase*,
  $W[P(m, k)] = \sum_{j=1}^{k} I(C_j)$. donde:

  $I(C_j) = \sum_{i \in C_j} p_i [v_i - G(C_j)]^2$ es la inercia de la clase C_j.
  $G(C_j) = \sum_{i \in C_j} \frac{p_i}{\mu_j} v_i$ es el centro de gravedad (promedio) de la clase C_j.
  $\mu_j = \sum_{i \in C_j} p_i$ es el peso de la clase C_j.
- P_l^i denota una partición óptima de $\{i, i+1, ...m\}$ en l clases.

El Algoritmo de Fisher calcula una partición P_k^1 de inercia intra-clase mínima. Por la manera en que se define $W[P(m, k)]$, éste es aditivo, lo que conlleva a poder conducir el proceso de optimización mediante la programación dinámica.

3 Algoritmo

Los pasos 1 y 2 del algoritmo de Fisher construyen recursivamente una sucesión P_l^i de particiones óptimas de $\{i, i+1, ..., m\}$ en l clases. En particular nos interesa P_k^1, partición óptima de $\{1, 2, ..., m\}$ en k clases.

Paso 1 : Para $i = 1, 2, ..., m$ se define $P_1^i = \{i, i+1, ..., m\}$ es, claramente, la particición óptima de $\{i, i+1, ..., m\}$ en una sola clase.

Paso 2 :
Para $l = 2, ..., k$ (etapa l) :

Para $i = k - l + 1, ..., m - l + 1$:

Se busca un $j_0 \in \{i, ..., m - l + 1\}$ tal que minimice:

$$I(\{i+1, ..., j\}) + W[P_{l-1}^{j+1}],$$

donde j varía en $\{i, ..., m - l + 1\}$
Se define:

$P_l^i = \{\{i, i+1, ..., j_0\}, P_{l-1}^{j_0+1}\}$, donde $P_{l-1}^{j_0+1}$ es la partición óptima de $\{j_0 + 1, ..., m\}$ en $l - 1$ clases, calculada en la etapa $l - 1$.

$W[P_l^i] = I(\{i, ..., j_0\}) + W[P_{l-1}^{j_0+1}]$.

Observación: En la etapa $l = k$ del paso 2, cuando $i = 1$, se construye la partición óptima de $\{1, ..., m\}$ en k clases buscada, $P_1^k = \{\{1, ..., j_0\}, P_{k-1}^{j_0+1}\}$ donde $P_{k-1}^{j_0+1}$ es la partición óptima de $\{j_0 + 1, ..., m\}$ en $k - 1$ clases, calculada en la etapa $k - 1$, con $j_0 \in \{1, 2, ..., m - k + 1\}$.

En [2] y [3] se demuestra que el algoritmo de Fisher construye una partición óptima de Ω en k clases. Para probarlo, se recurre a la forma particular de definir P_l^i en el algoritmo y a dos resultados: El primero afirma que en una partición óptima de un conjunto de números reales las clases son contiguas; es decir las clases no quedan traslapadas una con otra. El otro resultado afirma que si $P^* = \{P_1, P_2, ..., P_k\}$ es una partición óptima de Ω en k clases, entonces $\{P_2, ..., P_k\}$ es una partición óptima de $\Omega \setminus P_1$ en $k - 1$ clases.

3.1 Estimación del número óptimo de clases

El número de clases que mejor particiona los datos puede buscarse considerando un cambio significativo en el decrecimiento del criterio $W[P_k^1]$. Una forma de hacerlo es graficando $W[P_k^1]$ contra k [5] y elegir visualmente un punto que forma una especie de "codo". Sin embargo, esta forma visual no parece ser muy práctica. Una forma sencilla de hacerlo es minimizando el cociente:

$$\frac{\Delta W[P_k^1]}{\Delta W[P_{k-1}^1]} \text{ para } k = 2, ..., m - 1$$

donde $\Delta W[P_l^1] = W[P_l^1] - W[P_{l-1}^1]$. En la tabla 4 del ejemplo 1 se muestran algunos cálculos que ilustran la escogencia del k óptimo.

3.2 Consideraciones computacionales

La implementación del algoritmo de Fisher fue realizada con el lenguaje Modula-II ([6]), creando una biblioteca de subrutinas y estructuras llamada IOMATRIZ en la que se incluyen los procedimientos para lectura-escritura de matrices, ordenamiento de datos y los principales procedimientos empleados por el algoritmo[2].

El programa utiliza principalmente cuatro matrices: D, C, W e I , donde:

- D : Matriz de datos de entrada a clasificar, de tamaño $n \times p$, donde p es el número de variables que describen a los individuos, aunque interesa primordialmente manejar sólo la columna que corresponde a la variable cuantitativa v.
- C : Para almacenar la matriz D, una vez que ésta ha sido comprimida para clasificar sólo los datos que son diferentes en la variable numérica v (C también es de tamaño $n \times p$).
- $W = (W_{li})$: Matriz $k \times n$ donde $W_{li} = W[P_l^i]$.
- $I = (I_{ij})$: Matriz $m \times m$, donde $I_{ij} = I(\{i, ..., j\})$, $\forall i, j, 1 \leq i \leq j \leq m$.

Descripción del programa principal

1. Lectura de datos iniciales: Se pide el nombre del archivo en que se encuentran los datos a clasificar, son leídos para llenar la matriz D, se ordena D respecto a v.
2. Se comprimen los datos de D para obtener la matriz C en la que se incluyen sólo los datos de D que no tienen el mismo valor de v. De paso, se calcula el valor de m.
3. A partir de C, se llena la matriz I de inercias $I(\{i, ..., j\})$.
4. Se inicializa la primera fila de W usando los valores de la primera fila de I. Esto corresponde al *Paso 1* del algoritmo.
5. Se termina de calcular la matriz W usando la matriz I y la recursión del paso 2 del algoritmo. En cada cálculo se van guardando los índices j_0's. Usando la primera columna de W se estima el número óptimo de clases inherentes al conjunto de datos y se sugiere éste al usuario.
6. Se calcula la partición óptima de los datos de la matriz C en k clases haciendo un recorrido de los índices j_0's en los que ocrrían los óptimos.

[2]El programa, así como la forma de usarlo y la descripción del formato de datos de entrada y de salida pueden conseguirse con el autor en el Departamento de Matemática del Instituto Tecnológico de Costa Rica.

7. Se descomprimen los datos C. Esto consiste en asignar la clase obtenida para cada fila de C a todos los individuos de D que tienen el mismo valor v.

8. Se despliegan los resultados de la clasificación en pantalla y se graba en un archivo.

4 Ejemplos

Ejemplo 1: La tabla 1 contiene 16 datos tomados de [Hartigan74], pág. 131 y se refieren al tiempo logrado, en décimas de segundo, en carreras 100 metros en diferentes años de los juegos olímpicos. Todos los pesos se consideran iguales a $\frac{1}{16} = 0.0725$ y la variable respecto a la cual se va a clasificar es el *tiempo*.

Año	Tiempo	Peso	Año	Tiempo	Peso
1896	120.0	0.0725	1928	108.0	0.0725
1900	108.0	0.0725	1932	103.0	0.0725
1904	110.0	0.0725	1936	103.0	0.0725
1906	102.0	0.0725	1948	103.0	0.0725
1908	108.0	0.0725	1952	104.0	0.0725
1912	108.0	0.0725	1956	105.0	0.0725
1920	108.0	0.0725	1960	102.0	0.0725
1924	106.0	0.0725	1964	100.0	0.0725

Tabla 1.

Para este conjunto de datos, siguiendo el criterio presentado en la sección 3.1, se ha estimado en $k = 4$ el número óptimo de clases. Este es el número más natural de grupos homogéneos y separados entre sí que pueden formarse de acuerdo con la variable *tiempo*. El conjunto de datos ha sido comprimido a sólo los 9 datos que son diferentes en la variable *tiempo*. Estos 9 datos se han clasificado en 4 clases, obteniendo la tabla 2:

Año	Tiempo	Peso	Clase	Año	Tiempo	Peso	Clase
1964	100.0	0.0725	1	1900	108.0	0.3625	3
1906	102.0	0.1450	1	1904	110.0	0.0725	3
1932	103.0	0.2175	1				
1952	104.0	0.0725	2	1896	120.0	0.0725	4
1956	105.0	0.0725	2				
1924	106.0	0.0725	2				

Tabla 2.

Al descomprimir la matriz anterior, se obtiene la tabla 3, que contiene los datos originales, ordenados y clasificados en 4 clases:

Año	Tiempo	Peso	Clase	Año	Tiempo	Peso	Clase
1964	100.0	0.0725	1	1900	108.0	0.3625	3
1906	102.0	0.1450	1	1908	108.0	0.0725	3
1960	102.0	0.0725	1	1912	108.0	0.0725	3
1932	103.0	0.2175	1	1920	108.0	0.0725	3
1936	103.0	0.0725	1	1928	108.0	0.0725	3
1948	103.0	0.0725	1	1904	110.0	0.0725	3
1952	104.0	0.0725	2	1896	120.0	0.0725	4
1956	105.0	0.0725	2				
1924	106.0	0.0725	2				

Tabla 3.

La matriz de inercias, $I = (I_{ij})$ entre segmentos consecutivos de individuos $I(\{i, i+1, ..., j\})$ es:

$$\begin{pmatrix} 0.00 & 0.19 & 0.49 & 0.70 & 1.12 & 1.80 & 7.37 & 9.16 & 24.05 \\ & 0.00 & 0.08 & 0.20 & 0.49 & 1.01 & 5.53 & 7.06 & 21.15 \\ & & 0.00 & 0.05 & 0.23 & 0.58 & 3.74 & 4.90 & 17.71 \\ & & & 0.00 & 0.03 & 0.14 & 1.36 & 1.99 & 12.65 \\ & & & & 0.00 & 0.03 & 0.68 & 1.15 & 11.02 \\ & & & & & 0.00 & 0.24 & 0.58 & 9.71 \\ & & & & & & 0.00 & 0.24 & 8.70 \\ & & & & & & & 0.00 & 3.62 \\ & & & & & & & & 0.00 \end{pmatrix}$$

La matriz $W = (W_{li})$, cuyas entradas W_{li} son la inercia total intra-clase de la partición óptima de $\{i, i+1, ..., 9\}$ en l clases, es:

$$\begin{pmatrix} 24.05 & 21.15 & 17.71 & 12.65 & 11.02 & 9.71 & 8.70 & 3.62 & 0.00 \\ 9.16 & 7.06 & 4.90 & 1.99 & 1.15 & 0.58 & 0.24 & 0.00 & \\ 1.70 & 1.07 & 0.81 & 0.38 & 0.27 & 0.24 & 0.00 & & \\ 0.88 & 0.47 & 0.33 & 0.14 & 0.03 & 0.00 & & & \\ 0.47 & 0.23 & 0.09 & 0.03 & 0.00 & & & & \\ 0.23 & 0.09 & 0.03 & 0.00 & & & & & \\ 0.09 & 0.03 & 0.00 & & & & & & \\ 0.03 & 0.00 & & & & & & & \\ 0.00 & & & & & & & & \end{pmatrix}$$

Para la escogencia del número óptimo de clases, puede observarse en la tabla 4 que el mínimo de $\frac{\Delta W[P_k^1]}{\Delta W[P_{k-1}^1]}$ se alcanza cuando $k = 4$, por lo que éste es el número de clases sugerido para particionar los datos.

k	W_k	$\Delta W[P_k^1]$	$\frac{\Delta W[P_k^1]}{\Delta W[P_{k-1}^1]}$
1	24.05	-24.05	—
2	9.16	-14.83	0.62
3	1.70	-7.46	0.50
4	0.88	-0.82	0.11
5	0.47	-0.41	0.50
6	0.23	-0.24	0.59
7	0.09	-0.14	0.58
8	0.04	-0.05	0.36
9	0.00	-0.04	0.8

Tabla 4. Estimación del número óptimo de clases : $k = 4$.

Ejemplo 2: La tabla de datos que se expone a continuación es una tabla codificada de las respuestas de un grupo de 35 estudiantes del Instituto Tecnológico de Costa Rica, en Julio de 1995, a una secuencia de preguntas, algunas de ellas intentando medir cierto nivel de vida de su familia y clasificarlos de acuerdo con el salario conjunto de sus padres. Se consideran todos los pesos iguales a 1/35.

i	ing	t	ep	em	tv	ca	at	pv	co	i	ing	t	ep	em	tv	ca	at	pv	co
1	350	1	6	5	4	0	3	4	0	19	250	0	6	5	2	1	1	4	4
2	481	1	4	6	1	0	2	8	0	20	100	0	3	3	1	0	1	1	0
3	150	0	3	4	2	0	0	4	0	21	400	1	5	5	4	1	3	1	3
4	100	1	6	6	4	0	0	1	0	22	900	0	6	6	5	1	3	1	4
5	48	0	1	2	1	0	0	2	0	23	100	1	1	1	3	0	0	1	0
6	35	0	1	1	3	0	0	3	0	24	110	0	2	2	2	0	1	4	4
7	300	0	6	3	2	1	2	1	3	25	180	1	2	6	1	0	1	1	3
8	60	0	3	3	1	0	1	2	4	26	120	1	5	5	2	1	1	1	0
9	50	1	2	1	3	0	1	2	4	27	250	0	4	2	2	0	2	1	2
10	150	0	6	4	1	0	0	7	3	28	160	1	6	6	5	0	0	1	0
11	130	0	6	3	1	0	0	1	3	29	67	0	5	5	2	0	1	4	0
12	740	1	6	6	2	1	3	1	4	30	30	0	3	2	2	0	0	7	0
13	140	1	4	3	1	0	1	3	0	31	90	1	5	3	2	0	1	1	4
14	40	1	4	2	1	0	1	2	0	32	100	0	4	3	2	0	1	1	4
15	110	1	3	6	2	0	1	1	0	33	50	0	5	4	1	0	0	1	0
16	200	1	6	5	3	0	2	1	2	34	105	0	3	1	1	0	0	7	0
17	67	1	3	5	3	0	0	8	0	35	180	1	5	5	4	0	0	7	0
18	740	0	6	6	2	0	1	1	4										

Tabla 5

Significado de la codificación:

i : Indice del individuo, su identificador.

ing : Ingreso conjunto en salarios de los padres de familia (en miles de colones).

t : Si ambos trabajan (1 = Sí, 0 = No).

ep, em : Escolaridad del padre y de la madre, respectivamente (1 = Primaria incompleta, 2 = Primaria completa, 3 = Secundaria incompleta, 4 = Secundaria completa, 5 = Universitaria incompleta, 6 = Universitaria completa).

tv : Número de televisores que poseen en la casa.

ca : Si están suscritos a algún servicio de televisión por cable (1 = Sí , 0 = No).

at : Número de automóviles que posee la familia.

pv : Provincia en que habitan (1 = San José, 2 = Alajuela, 3 = Cartago, 4 = Heredia, 5 = Limón, 6 = Guanacaste, 7 = Puntarenas, 8 = extranjero).

co : Si tienen computadora en la casa y qué tipo (0 = no tiene, 1 = XT, 2 = AT-286, 3 = AT-386, 4 = AT-486 o más avanzada).

Para este conjunto de datos, se ha estimado que $k = 3$ es el número óptimo de clases.

Los datos quedan ordenados y clasificados en 3 clases que podrían interpretarse como 3 grandes categorías de ingresos: BAJO, MEDIO y ALTO.

Clase No. 1 : INGRESO "BAJO":

i	ing	t	ep	em	tv	ca	at	pv	co	i	ing	t	ep	em	tv	ca	at	pv	co
30	30	0	3	2	2	0	0	7	0	32	100	0	4	3	2	0	1	1	4
6	35	0	1	1	3	0	0	3	0	34	105	0	3	1	1	0	0	7	0
14	40	1	4	2	1	0	1	2	0	15	110	1	3	6	2	0	1	1	0
5	48	0	1	2	1	0	0	2	0	24	110	0	2	2	2	0	1	4	4
9	50	1	2	1	3	0	1	2	4	26	120	1	5	5	2	1	1	1	0
33	50	0	5	4	1	0	0	1	0	11	130	0	6	3	1	0	0	1	3
8	60	0	3	3	1	0	1	2	4	13	140	1	4	3	1	0	1	3	0
17	67	1	3	5	3	0	0	8	0	3	150	0	3	4	2	0	0	4	0
29	67	0	5	5	2	0	1	4	0	10	150	0	6	4	1	0	0	7	3
31	90	1	5	3	2	0	1	1	4	28	160	1	6	6	5	0	0	1	0
4	100	1	6	6	4	0	0	1	0	25	180	1	2	6	1	0	1	1	3
20	100	0	3	3	1	0	1	1	0	35	180	1	5	5	4	0	0	7	0
23	100	1	1	1	3	0	0	1	0	16	200	1	6	5	3	0	2	1	2

Tabla 6a

Clase No. 2 :INGRESO "MEDIO":

i	ing	t	ep	em	tv	ca	at	pv	co
19	250	0	6	5	2	1	1	4	4
27	250	0	4	2	2	0	2	1	2
7	300	0	6	3	2	1	2	1	3
1	350	1	6	5	4	0	3	4	0
21	400	1	5	5	4	1	3	1	3
2	481	1	4	6	1	0	2	8	0

Tabla 6b

Clase No. 3 : INGRESO "ALTO"

i	ing	t	ep	em	tv	ca	at	pv	co
12	740	1	6	6	2	1	3	1	4
18	740	0	6	6	2	0	1	1	4
22	900	0	6	6	5	1	3	1	4

Tabla 6c

5 Conclusiones

El algoritmo de Fisher calcula una partición óptima de un conjunto de datos respecto a una variable numérica que se les haya medido, a diferencia de otros algoritmos, como los de *nubes dinámicas*, que lo que dan es una partición aproximada. Pese a la gran utilidad que tiene en muchas aplicaciones, tiene la limitación de usar sólo una variable para clasificar por lo que deja de ser útil cuando se pretende obtener una partición óptima respecto a varias variables.

La utilidad principal del algoritmo reside en la posibilidad de escoger rangos óptimos para variables numéricas comúnmente usadas en diversos cuestionarios (salario, estatura, etc...).

La implementación del método ha sido importante, debido a que es una herramienta de codificación de uso común en análisis de datos y se le agregó al programa una rutina para estimar el número óptimo de clases, inherentes al conjunto de datos, siguiendo el método del "codo". La salida del programa en forma de un archivo con los datos particionados y, por lo tanto, clasificados (como en las tablas 6a, 6b y 6c), hace posible la discriminación de éstos, con la variable recodificada por el método de Fisher como variable a explicar.

Referencias

[1] Chandon, J. L.; Pinson, S. (1981) *Analyse Typologique. Théories et Applications.* Masson, Paris.

[2] Diday, E.; Lemaire, J.; Pouget, J.; Testu, F. (1982) *Eléments d'Analyse de Données.* Dunod, Paris.

[3] Espinoza, J. L.; Mora, W.; Trejos,J. (1988) *Clasificación Automática.* Memorias de Seminario de Graduación, Universidad de Costa Rica, San José.

[4] Espinoza, J. L.; Trejos, J. (1989) "Clasificación por particiones", *Ciencia y Tecnología*, Vol. XIII, Nos. 1-2: 129–154.

[5] Hartigan, J. (1974) *Clustering Algorithms.* John Wiley & Sons, New York.

[6] King, K. (1988) *TopSpeed Modula-2. Language Tutorial*. Jensen & Partners International, U.S.A.

[7] Ventsel, E. (1983) *Investigación de Operaciones. Problemas, Principios, Metodología*. Editorial Mir, Moscú.