Elements of Statistical Learning
Ejercicios Capítulo 9

Ejercicio 9.1.

$$Sy = S\hat{y} + Sr$$
$$\text{Que } S\hat{y} = \hat{y} \text{ implica que } Sy = \hat{y} + Sr$$
$$\hat{y} = X(X'X)^{-1}X'y$$
$$S = X(X'X)^{-1}X'$$
$$S\hat{y} = X(X'X)^{-1}X'X(X'X)^{-1}X'$$
$$S\hat{y} = X(X'X)^{-1}X'y$$
$$S\hat{y} = \hat{y}$$

Local linear regression:

$$\hat{f}(x_0) = b(x_0)'(B'W(x_0)B)^{-1}B'W(x_0)y$$
$$\text{Donde: } B = X$$
$$\text{Entonces: } \hat{f}(x_0) = b(x_0)'(X'W(x_0)X)^{-1}X'W(x_0)y$$

Existe una función $\hat{f}(.)$ estimada para cada uno de los valores de x y utilizada únicamente para evaluar el fit en ese mismo punto. La estimación para el resto de puntos, basándose únicamente en el fit hecho en el punto x_0, se obtiene calculando:

$$\hat{f} = X'(X'WX)^{-1}X'Wy$$

Donde W es la matriz de pesos correspondiente al punto en el cual se realizó la estimación

Se debe entonces demostrar que $S\hat{y} = \hat{y}$

$$\hat{y} = X(X'WX)^{-1}X'Wy$$
$$S = X(X'X)^{-1}X'$$
$$S\hat{y} = X(X'X)^{-1}X'X(X'WX)^{-1}X'Wy$$
$$S\hat{y} = X(X'WX)^{-1}X'Wy$$
$$S\hat{y} = \hat{y}$$

Ejercicio 9.2 (a).

El modelo adtivo tiene N observaciones y p funciones. EL algoritmo Gauss-Seidel para resolver ecuaciones lineales resuelve sucesivamente para la variable z_j en la j-ésima ecuación, tomando al resto de variables con sus valores estimados hasta el momento. Entonces, considerando el sistema de ecuaciones presentado, la resolución por *backfitting* para la ecuación j sería: $f_j = S_j[y_i - \sum_{k \neq j} \hat{f}_k(x_{ik})]$

$$
\begin{gathered}
f_1 = S_1[y - f_2 - f_3 - ... - f_p] \\
f_2 = S_2[y - f_1 - f_3 - ... - f_p] \\
. \\
. \\
. \\
f_p = S_p[y - f_1 - f_2 - ... - f_{p-1}]
\end{gathered}
$$

Dejando al lado derecho todos los términos on y:

$$
\begin{gathered}
f_1 + S_1 f_2 + S_1 f_3 + ... + S_1 f_p = S_1 y \\
S_2 f_1 + f_2 + S_2 f_3 + ... + S_2 f_p = S_2 y \\
. \\
. \\
. \\
S_p f_1 + S_p f_2 + S_p f_3 + ... + f_p = S_p y
\end{gathered}
$$

Tomando en cuenta que f_j es un vector Nx1 y S_j es una matriz NxN, las ecuaciones de arriba se pueden expresar como un sistema de ecuaciones $Az = b$ con las siguientes matrices particionadas:

$$
A = \begin{bmatrix} I & S_1 & S_1 & \dots & S_1 \\ S_2 & I & S_2 & \dots & S_2 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ S_p & S_p & S_p & \dots & I \end{bmatrix}
$$

$$
z = \begin{bmatrix} f_1 \\ f_2 \\ \vdots \\ f_p \end{bmatrix}
$$

$$
b = \begin{bmatrix} S_1 y \\ S_2 y \\ \vdots \\ S_p y \end{bmatrix}
$$

Utilizando el algoritmo Gauss-Seidel, la ecuación j del sistema arriba planteado se resolvería:

$$S_j f_1 + S_j f_2 + ... + f_j + ... + S_j f_p = S_j y$$

Factorizando:

$$f_j = S_j[y - f_1 - f_2 - ... - f_p]$$

Que es justamente el algoritmo *backfitting*

Ejercicio 9.2 (b).

Si S_1 y S_2 son *smoothing operators* simétricos con valores propios entre 0 y 1. Con cualquier valor inicial, el algoritmo *backfitting* converge y da una fórmula para las iteraciones finales. Cada iteración del algoritmo *backfitting* tiene la siguiente forma:

$$f_1(t) = S_1 y - S_1 f_2(t-1)$$
$$f_2(t) = S_2 y - S_2 f_1(t-1)$$

Reemplazando sucesivamente los términos rezagados en las ecuaciones se obtiene:

$$f_1(t) = S_1 y - S_1(S_2 y - S_2 f_1(t-2))$$
$$f_1(t) = S_1 y - S_1 S_2 y + S_1 S_2 f_1(t-2)$$
$$f_1(t) = S_1 y - S_1 S_2 y + S_1 S_2(S_1 y - S_1 f_2(t-3))$$
$$f_1(t) = S_1 y - S_1 S_2 y + S_1 S_2 S_1 y - S_1 S_2 S_1 f_2(t-3)$$
$$f_1(t) = S_1 y - S_1 S_2 y + S_1 S_2 S_1 y - S_1 S_2 S_1 S_2 y + S_1 S_2 S_1 S_2 f_1(t-4)$$

Se itera k veces hasta que $t-k=0$. Sin pérdida de generalidad se asumirá que t es par. Entonces:

$$f_1(t) = (S_1 - S_1 S_2 + S_1 S_2 S_1 - (S_1 S_2)^2 + ... - (S_1 S_2)^{k/2})y + (S_1 S_2)^{k/2} f_1(0)$$

El término que pre multiplica a y se puede simplificar un poco más.

$$(S_1 - S_1 S_2 + S_1 S_2 S_1 - (S_1 S_2)^2 + ... - (S_1 S_2)^{k/2})$$
$$(I + S_1 S_2 + (S_1 S_2)^2 + ... + (S_1 S_2)^{k/2-1})S_1 - (I + S_1 S_2 + (S_1 S_2)^2 + ... + (S_1 S_2)^{k/2-1})S_1 S_2$$
$$(I + S_1 S_2 + (S_1 S_2)^2 + ... + (S_1 S_2)^{k/2-1})(S_1 - S_1 S_2)$$

La convergencia se estima en el infinito. Debemos demostrar que mientras más grande es k, la importancia del valor inicial decrece hasta hacerse nulo en el infinito.

$$\lim_{k\to\infty}(I + S_1 S_2 + (S_1 S_2)^2 + ... + (S_1 S_2)^{k/2-1}) = (I - S_1 S_2)^{-1}$$

Cuando k tiene al infinito, la parte de la izquierda se convierte en una sucesión geométrica infinita que, en el equivalente numérico, converge a $\frac{1}{1-a}$ donde a es la razón geométrica. En el caso matricial, la razón es $(S_1 S_2)$, por lo que todo convergería a $(I - S_1 S_2)^{-1}$.

Por la naturaleza *shrinking* de las matrices S_j y la presencia de valores propios positivos y menores que 1, $S_1 S_2$ elevado a una potencia infinita, tenderá a la matriz nula. Por eso:

$$\lim_{k\to\infty}(S_1 S_2)^{k/2} f_1(0) = 0$$
$$\lim_{k\to\infty} f_1(t) = (I - S_1 S_2)^{-1}(S_1 - S_1 S_2)y$$

Por simetría,

$$f_2(t) = (I - S_2 S_1)^{-1}(S_2 - S_2 S_1)y$$

Ejercicio 9.3.

Procedimiento *backfitting* con proyecciones ortogonales. D es la matriz de regresión compuesta por la familia de spline functions.

El procedimiento *backfitting* y la resolución del sistema de ecuaciones planteado, se caracterizan por la ecuación:

$$f_j = S_j[y - \sum_{k \neq j} f_k]$$
$$S_j = N_j(N_j'N_j)^{-1}N_j'$$
$$f_j = N_j\theta_j$$
$$N_j\theta_j + N_j(N_j'N_j)^{-1}N_j'(\sum_{k \neq j} f_k) = N_j(N_j'N_j)^{-1}N_j'y$$

Premultiplicando todo por N_j'

$$N_j'N_j\theta_j + N_j'(\sum_{k \neq j} f_k) = N_j'y$$
$$N_j'N_j\theta_j + N_j'(\sum_{k \neq j} N_k\theta_k) = N_j'y$$
$$N_j'(\sum_{k} N_k\theta_k) = N_j'y$$

La sumatoria $\sum_{k} N_k\theta_k$ puede expersarse como producto de las siguientes dos matrices particionadas:

$$\begin{bmatrix} N_1 & N_2 & N_3 & \dots & N_p \end{bmatrix} \text{ y } \theta = \begin{bmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \\ \vdots \\ \theta \end{bmatrix}$$

Nos indican que la matriz D es la matriz de regresión, es decir, que contiene todas las matrices N. En ese caso:

$$D = \begin{bmatrix} N_1 & N_2 & N_3 & \dots & N_p \end{bmatrix}$$

$$N_j'D\theta = N_j'y \tag{1}$$

La expresión (1) solo resuelve la ecuación j del sistema. Podemos expresar todo el sistema en su conjunto usando matrices particionadas:

$$\begin{bmatrix} N_1' \\ N_2' \\ N_3' \\ \vdots \\ N_p' \end{bmatrix} D = \begin{bmatrix} N_1' \\ N_2' \\ N_3' \\ \vdots \\ N_p' \end{bmatrix} y$$

Por la definición de D, se concluye que:

$$D'D\theta = D'y$$

Ejercicio 9.4.

Partiendo de la convergencia hallada en el ejercicio 9.2, se sabe que el algoritmo *backfitting* converge a $(I - S_2S_1)^{-1}(S_2 - S_2S_1)y$. Si ambos S_j son iguales, la convergencia es a $(I - S^2)^{-1}(S - S^2)y$. El error sería:

$$y - f_1 - f_2$$

Como f_1 y f_2 son iguales, el error quedaría:

$$\begin{gathered}
y - 2f \\
y - 2(I - S^2)^{-1}(S - S^2)y \\
y - 2((I - S)(I + S))^{-1}(I - S)Sy \\
y - 2(I + S)^{-1}(I - S)^{-1}(I - S)Sy \\
y - 2(I + S)^{-1}Sy \\
(I + S)^{-1}((I + S)y - 2Sy) \\
(I + S)^{-1}(y - Sy) \\
(I + S)^{-1}(I - S)y
\end{gathered}$$

Residual sum of squares:

$$\begin{gathered}
RSS = e'e \\
RSS = [(I + S)^{-1}(I - S)y]'[(I + S)^{-1}(I - S)y] \\
RSS = y'(I - S)(I + S)^{-1}(I + S)^{-1}(I - S)y
\end{gathered}$$

Ejercicio 9.5 (a).

Los grados de libertad de un *fit* son: $\sum_i cov(y_i, \hat{y}_i)/\sigma^2$

$$df = \sum_i cov(y_i, \hat{y}_i)/\sigma^2 = \sum_i cov(\hat{y}_i + e_i, \hat{y}_i)/\sigma^2 = \sum_i [var(\hat{y}_i) + cov(e_i, \hat{y}_i)]/\sigma^2 = \sum_i var(\hat{y}_i)/\sigma^2$$

La varianza de $\hat{y}_i$ en el caso general de m *terminal nodes*, donde cada grupo G_i tiene $|G_i|$ observaciones que lo componen, es:

$$df = \sum_i var(\hat{y}_i)/\sigma^2$$
$$df = \sum_{i=1}^{m} \sum_{j \in G_i} var(\hat{y}_i)/\sigma^2$$

La predicción para cada observación es la media del grupo al que pertenece

$$\hat{y}_i = \sum_{i \in G_k} y_i/|G_k| \text{ si la observación } i \in G_k$$
$$var(\hat{y}_i) = var(\sum_{i \in G_k} \frac{y_i}{|G_k|})$$
$$var(\hat{y}_i) = \frac{1}{|G_k|^2} var(\sum_{i \in G_k} y_i)$$
$$var(\hat{y}_i) = \frac{1}{|G_k|^2} \sum_{i \in G_k} var(y_i)$$
$$var(y_i) = \sigma^2$$
$$var(\hat{y}_i) = \frac{|G_k|\sigma^2}{|G_k|^2}$$
$$var(\hat{y}_i) = \frac{\sigma^2}{|G_k|}$$
$$\text{Reemplazando: } df = \sum_{i=1}^{m} \sum_{j \in G_i} \frac{\sigma^2}{|G_k|\sigma^2}$$
$$df = \sum_{i=1}^{m} \sum_{j \in G_i} \frac{1}{|G_k|} = \sum_{i=1}^{m} \frac{|G_k|}{|G_k|}$$
$$df = \sum_{i=1}^{m} 1 = m$$

Ejercicio 9.5 (e).

Consideremos un caso con n observaciones y k nodos terminales. Si los árboles de regresión fueran un operador lineal, $Sy = \hat{y}$. La matriz S debería tener la siguiente forma:

$$S = \begin{bmatrix} \frac{1}{|G_1|} & 0 & 0 & 0 & \dots & 0 \\ 0 & \frac{1}{|G_2|} & \frac{1}{|G_2|} & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \\ 0 & 0 & 0 & \frac{1}{|G_k|} & \dots & \frac{1}{|G_k|} \end{bmatrix}$$

Como cada observación de la base de datos pertenece siempre a un solo grupo, en la diagonal de la matriz S siempre estará $\frac{1}{|G_i|}$, donde G_i es el grupo al que pertenece la observación que se va a estimar. Esto sucede porque dentro de la media del grupo que pertenece a determinado nodo, siempre estará incluida la observación que se va a estimar. Esto solo sucede en caso se esté estimando dentro del training set.

Como en la diagonal siempre aparecerá el elemento $\frac{1}{|G_i|}$ y cada observación pertenece como máximo a un grupo, la suma de estos elementos será m. Esto porque cada $\frac{1}{|G_k|}$ aparece $|G_k|$ veces.