UNIVERSIDADE
DE VIGO

UNIVERSIDADE DA CORUÑA

NPRegfast: UN PAQUETE DE R PARA ANALIZAR INTERACCIONES FACTOR-POR-CURVA

Marta Sestelo

Máster en Técnicas Estadísticas

Universidad de Vigo

NPRegfast: UN PAQUETE DE R PARA ANALIZAR INTERACCIONES FACTOR-POR-CURVA

Marta Sestelo

D. Javier Roca Pardiñas, profesor del Departamento de Estadística e Investigación Operativa de la Universidad de Vigo, hace constar que el trabajo titulado

`NPRegfast`: un paquete de `R` para analizar interacciones factor-por-curva

ha sido realizado por Dña. Marta Sestelo Pérez, bajo su dirección para su presentación como Proyecto Fin de Máster del Máster en Técnicas Estadísticas.

Fdo.: Dña. Marta Sestelo Pérez

Fdo.: D. Javier Roca Pardiñas

Vigo, a 16 de Enero de 2012

Resumen

En este proyecto se presenta un nuevo paquete de R, `NPRegfast`, que permite estimar y hacer inferencia en modelos de regresión con interacciones factor-por-curva. En este tipo de interacciones, la relación entre la respuesta media y las covariables explicativas depende de los niveles de un factor. Esta librería permite comparar las curvas de regresión específicas de cada grupo e incluso comparar sus puntos críticos (p.ej. mínimos, máximos o puntos de inflexión) a través del estudio de sus derivadas. Asimismo, este programa permite no sólo obtener estimaciones basadas en modelos paramétricos clásicos (como es el caso del modelo alométrico, uno de los más utilizados en la gestión y evaluación pesquera) sino también mediante el uso de suavizadores no paramétricos tipo núcleo. La inferencia (intervalos y contrastes) está basada en técnicas de remuestreo bootstrap. Adicionalmente, se ha implementado la técnica de aceleración computacional binning de forma que `NPRegfast` resulta muy eficiente desde un punto de vista computacional. El software se ilustra utilizando datos biológicos, más concretamente, estimando una talla mínima de captura para la especie marina *Pollicipes pollicipes*.

Índice general

Anexo

Capítulo 1

Introducción

En muchas situaciones prácticas, la variable respuesta, Y, depende de una covariable continua, X. En este contexto, una posibilidad es considerar que la relación entre estas dos variables venga dada por un modelo de regresión no paramétrica del tipo

$$Y = m(X) + \varepsilon \tag{1.1}$$

donde m es una función suave o el efecto asociado con la covariable continua X y ε es el error, que se asume independiente de la covariable X. La principal ventaja de utilizar modelos como en (1.1) reside en la flexibilidad de m y en su fácil interpretación.

Una generalización del modelo "puro" en (1.1) es el modelo de regresión con interacciones factor-por-curva. En este tipo de modelos, la relación entre la respuesta media y las covariables explicativas puede variar dependiendo de los niveles de una variable categórica F. La posibilidad de incorporar este tipo de interacción en los modelos de regresión no paramétrica ya ha sido discutida por Hastie and Tibshirani (1990). Además, recientemente, Ruppert and Wand (1994) presentaron un algoritmo basado en splines penalizados (P-splines), que permite incorporar este tipo de interacciones en estos modelos.

Por lo tanto, para estudiar el posible efecto de F en la respuesta, se consideró el siguiente modelo de regresión no paramétrica que incluye dichas interacciones factor-por-curva

$$Y = f_0(X) + \begin{cases} f_1(X) + \varepsilon_1 & \text{si } F = 1 \\ \vdots & \\ f_M(X) + \varepsilon_M & \text{si } F = M \end{cases} \tag{1.2}$$

donde $\varepsilon_1, \ldots, \varepsilon_M$ son los errores de media cero para cada nivel del factor, f_0 es el efecto global de la X, y f_l es el efecto específico de la X asociado con el nivel l-ésimo del factor F. Por simplicidad de notación, se designa

$$m_l(X) = f_0(X) + f_l(X) \qquad \text{para} \quad l = 1, \ldots, M$$

Nótese que $m_l(X) = E(Y|X = x, F = l)$ es la curva de regresión de Y sobre X en el nivel l del factor F.

Cuando una interacción factor-por-curva es detectada en el modelo en (1.2), además de su estimación, podría ser interesante disponer de tests capaces de detectar qué efectos dependen del factor. Si estos tests resultan significativos y existe una verdadera interacción, interesaría también disponer de metodologías que permitan estudiar diversas características sobre las curvas de regresión y sus derivadas. Por ejemplo, una cuestión interesante sería comprobar si existen diferencias entre los distintos niveles del factor y/o comparar un punto crítico (como un máximo o un punto de inflexión) asociado a sus derivadas.

Con la librería que se presenta en este proyecto, `NPRegfast`, todas estas cuestiones pueden ser solventadas por el usuario final de una manera sencilla y computacionalmente eficiente. Para la estimación de los modelos se propone el uso de suavizadores no paramétricos tipo núcleo (Wand and Jones, 1995) mientras que la inferencia (intervalos y contrastes) está basada en técnicas de remuestreo bootstrap (Efron and Tibshirani, 1993; Efron, 1979). Adicionalmente, este programa permite no sólo obtener estimaciones suavizadas flexibles, sino también basadas en modelos paramétricos clásicos, como es el caso del modelo alométrico, uno de los más utilizados en el ámbito de la biología para estudiar la relación entre dos variables biométricas. Finalmente, se ha implementado la técnica de aceleración computacional binning (Fan and Marron, 1994) de forma que `NPRegfast` resulta muy eficiente desde un punto de vista computacional.

La estructura de este proyecto es la siguiente. En la Sección 1.1 se presenta el conjunto de datos que se analizará a modo de aplicación práctica. La metodología utilizada en la implementación del software se desarrolla en el Capítulo 2. Más concretamente, se presenta el algoritmo de estimación basado en los suavizadores tipo kernel, se proponen métodos bootstrap para la implementación de los distintos test diseñados para detectar diferencias significativas en las curvas atendiendo a sus derivadas, así como distintas cuestiones prácticas como la selección de ventana y la aceleración computacional basada en las técnicas binning. La implementación en un

entorno amigable para el usuario de la metodología desarrollada en este proyecto fue uno de los principales objetivos. Por ello, una descripción detallada del software implementado puede ser encontrada en el Capítulo 3. En el Capítulo 4, dicha metodología es utilizada para evaluar la relación talla-peso del percebe, *P. pollicipes*, y estimar así una talla de captura mínima. Finalmente, esta disertación termina con algunas conclusiones en el Capítulo 5.

1.1. Conjunto de datos

El percebe, *Pollicipes pollicipes* (Gmelin, 1789), es un cirrípedo pedunculado sésil, estrictamente litoral y esencialmente intermareal, que vive formando densos agregados en costas expuestas asociado a un elevado hidrodinamismo (Barnes, 1996). De las tres especies pertenecientes al género *Pollicipes* (Newman, 1987), *P. pollicipes* se extiende a lo largo de las costas atlánticas francesas, españolas, portuguesas, marroquíes y senegalesas. Existen además citas de esta especie en las costas mediterráneas españolas, francesas, marroquíes y argelinas (Barnes, 1996; Cruz, 2000; Darwin, 1851). Atendiendo a los fenómenos de explotación, la especie tropical pacífica, *Pollicipes elegans* Lesson, 1830, sufre una pequeña y localizada extracción en Costa Rica (Bernard, 1988) y en Perú (Pinilla, 1996; Ramírez et al., 2008), mientras que *Pollicipes polymerus* Sowerby, 1833 es explotado en las costas de Canadá (Bernard, 1988; Lauzier, 1999).

La especie atlántica, *P. pollicipes*, ha sido y es la más explotada de las tres especies, países como Francia, España, Portugal y Marruecos extraen este recurso de sus costas (Bernard, 1988; Cruz and Araujo, 1999; Girard, 1982; Goldberg, 1984). Su interés comercial se basa en el aprovechamiento de su pedúnculo muscular, parte comestible de esta especie que llega a alcanzar altos precios en el mercado (Goldberg, 1984). En Galicia (N.O. de España), máxima comunidad productora de percebes de España, la producción media anual declarada de *P. pollicipes* se sitúa aproximadamente en 400 Tm (dato oficial, Xunta de Galicia, http://www.pescadegalicia.com). Sin embargo, esta cantidad es muy inferior a la real, debido a que un gran volumen de capturas no es declarado. Como consecuencia de esta fuerte demanda por parte del mercado español, se hizo necesaria la importación de percebes (*P. pollicipes* y *P. polymerus*) desde Francia, Portugal, Marruecos y Canadá (Bernard, 1988; Girard, 1982; Molares, 1993).

Tanto en España como en Portugal, países donde la extracción de *P. pollicipes* es más notable, esta especie ha sufrido fenómenos de sobreexplotación en distinto grado

(Bernard, 1988; Cardoso and Yule, 1995; Cruz, 2000; Molares and Freire, 2003).

Los cirrípedos pedunculados, que incluyen a las especies del género *Pollicipes*, crecen en altura debido a un aumento en la longitud del pedúnculo y en anchura por acreción lamelar, que resulta de la adición de carbonato cálcico a las placas del capítulo (Anderson, 1994). Según Darwin (1854), factores ambientales como el alimento, la temperatura y la calidad del agua pueden influir en la forma y tamaño de individuos de la misma especie de cirrípedos.

A pesar de la importancia económica de *P. pollicipes* tanto en España como en otros países, nuestro conocimiento sobre la biología y ecología de esta especie es fragmentario, y varios aspectos deberían ser estudiados en profundidad, como por ejemplo el crecimiento en peso de este crustáceo. Atendiendo a esto, el principal objetivo de este estudio es estimar la ganancia en peso de los individuos a medida que aumenta su talla, estableciendo de esta manera la relación talla-peso para *P. pollicipes*. Con este propósito se utilizarán las funciones implementadas en el paquete `NPRegfast`.

Con el fin de estimar dicha ganancia en peso, se seleccionaron dos variables biométricas, longitud rostro-carenal (RC), variable que mejor representa el crecimiento de la especie (Cruz, 1993, 2000); y el peso individual (DW), que nos permite evaluar el aprovechamiento de este recurso. Para observar la relación entre estas dos variables, se han utilizado dos modelos de regresión, que son luego comparados, el modelo alométrico clásico y un modelo no paramétrico. Ambos modelos se pueden estimar por medio de la función principal `frfast` de la librería que se presenta en este proyecto.

En el caso del modelo no paramétrico, la relación talla-peso de *P. pollicipes* ha sido estimada mediante la utilización de modelos de regresión polinómicos locales basados en suavizadores tipo kernel. Estos modelos de regresión no paramétrica permiten ajustes más flexibles de los datos reales que las técnicas paramétricas de regresión usualmente empleadas. Asimismo, admiten el cálculo de la primera derivada de la curva de regresión permitiendo definir las distintas etapas de crecimiento de esta especie a medida que incrementa su tamaño. Además, el cálculo de esta derivada podría tener una aplicación directa en la gestión de esta especie, haciendo posible la estimación de una talla de captura.

Para el establecimiento de la talla de captura de cualquier especie sujeta a explotación deben considerarse diversos aspectos biológicos y ecológicos, como son la talla de los individuos en su primera reproducción, su tasa de crecimiento y su ciclo biológico. Adicionalmente, debe valorarse la ganancia en peso de cada ejemplar a lo largo del tiempo. En relación a esto, la Organización de las Naciones Unidas para la

Alimentación y la Agricultura (FAO) indica que "La finalidad básica de la evaluación de stocks es asesorar sobre la explotación óptima de recursos acuáticos vivos (...) y la evaluación de los stocks de peces se puede definir como la búsqueda del nivel de explotación que permita obtener, a largo plazo, el rendimiento máximo en peso de una pesquería" (Sparre and Venema, 1997). Según esta indicación, creemos que el estudio de las derivadas es extremadamente útil cuando se trata de establecer una talla de captura. En particular, este estudio propone que la talla mínima se corresponda con el punto (o talla) donde la primera derivada alcanza el máximo. A partir de este punto, la ganancia en peso de una talla a la siguiente disminuye.

La metodología expuesta en este proyecto, y el desarrollo de software asociado, se ha utilizado para resolver las cuestiones expuestas con anterioridad. Su desarrollo detallado, resultados y conclusiones se muestran en el Capítulo 4.

Capítulo 2

Estimación e Inferencia

En este capítulo se presenta toda la metodología aplicada al modelo de interacción propuesto en (1.2), donde para estudiar el posible efecto de F en la respuesta, se consideraba el siguiente modelo

$$Y = f_0(X) + \begin{cases} f_1(X) + \varepsilon_1 & \text{si } F = 1 \\ \vdots & \\ f_M(X) + \varepsilon_M & \text{si } F = M \end{cases} \tag{2.1}$$

donde $\varepsilon_1, \ldots, \varepsilon_M$ son los errores de media cero para cada nivel del factor, f_0 es el efecto global de la X, y f_l es el efecto específico de la X asociado con el nivel l-ésimo del factor F.

En este proyecto, se propone la estimación de este tipo de modelos utilizando suavizadores locales lineales tipo kernel (Wand and Jones, 1995). Estos modelos de regresión no paramétrica permiten un ajuste más flexible de los datos que la técnicas de regresión paramétrica usualmente empleadas. Asimismo, permite el cálculo de las derivadas de las curvas de regresión.

La estructura de este capítulo es la siguiente. En la Sección 2.1 se detalla el algoritmo de estimación de estos modelos así como distintas cuestiones prácticas relacionadas con su aplicación, como la selección de ventana o la aceleración computacional. La inferencia, ya sea en base a los intervalos de confianza o por medio de contrastes, se desarrolla en la Sección 2.2

2.1. Estimación

En esta sección se presenta el estimador polinómico local del modelo (2.1) con una covariable continua X. Antes de estimar las funciones f_ls, deben imponerse algunas restricciones al modelo para evitar que diferentes combinaciones de $f_0, f_1, \ldots, f_M$ den lugar al mismo modelo. Como solución práctica para permitir la identificación, es necesaria la condición siguiente: la suma de los efectos específicos de todos los niveles debe ser cero. En concreto, para una muestra dada $\{X_i, F_i, Y_i\}_{i=1}^n$ siguiendo el modelo (2.1), esta condición viene dada por: $\sum_{l=1}^M f_l(X_i) = 0$ para $1 \leq i \leq n$. Nótese que esta condición no representa restricciones al modelo planteado ya que puede ser modificado para ajustarse a la condición de especificación dada.

La estimación de la r-ésima derivada de f_0 en un punto x viene dada por

$$\hat{f}_0^r(x) = r!\hat{\alpha}^r(x) \quad \text{para} \quad r = 0, 1, \ldots, R \tag{2.2}$$

donde $\left(\hat{\alpha}^0(x), \hat{\alpha}^1(x), \ldots, \hat{\alpha}^R(x)\right)$ es el minimizador

$$\sum_{i=1}^n \left\{ Y_i - \sum_{r=1}^{R+1} \alpha^r(x)(X_i - x)^{r-1} \right\}^2 \cdot K_h(X_i - x),$$

donde $K_h(\cdot) = K(\cdot/h)/h$, K_h denota la función kernel (una densidad simétrica) y $h > 0$ es el parámetro de suavización (o ventana). En este trabajo, la función kernel $K(\cdot)$ utilizada es el núcleo gaussiano $K(u) = 1/\sqrt{2\pi}\exp(-u^2/2)$.

Una vez obtenida la estimación de $\hat{f}_0$ para $l = 1, \ldots, M$ se computan los residuos $Y_i^l = Y_i - \hat{f}_0(X_i)$ y los pesos $W_i^l = I_{\{F_i = l\}}$, y se obtienen las estimaciones de la r-ésima derivada de f_l como

$$\hat{f}_l^r(x) = r!\hat{\alpha}_l^r(x) \quad \text{para} \quad r = 0, 1, \ldots, R \tag{2.3}$$

donde $\left(\hat{\alpha}_l^0(x), \hat{\alpha}_l^1(x), \ldots, \hat{\alpha}_l^R(x)\right)$ es el minimizador

$$\sum_{i=1}^n \left\{ Y_i^l - \sum_{r=1}^{R+1} \alpha_l^r(x)(X_i - x)^{r-1} \right\}^2 \cdot K_h(X_i - x) W_i^l$$

Nótese que las estimaciones obtenidas no tienen porqué cumplir la condición de identificación dada. Para que ésta se cumpla, es necesario el siguiente procedimiento. Para cada X_i se calcula la media de los efectos específicos de cada nivel, $S(X_i) = \sum_{l=1}^M \hat{f}_l(X_i)/M$, y se obtiene $\hat{f}_l(X_i) = \hat{f}_l(X_i) - S(X_i)$ y $\hat{f}_0(X_i) = \hat{f}_0(X_i) + S(X_i)$.

Finalmente, las curvas estimadas para cada nivel se corresponden con

$$\hat{m}_l^r(X) = \hat{f}_0^r(X) + \hat{f}_l^r(X) \quad \text{para} \quad l = 1, \ldots, M$$

Una vez obtenidas las curvas anteriores, puede resultar interesante hacer inferencia sobre sus puntos críticos, como mínimos, máximos o puntos de inflexión, a través del estudio de sus derivadas. A modo de ejemplo, para la aplicación práctica de este proyecto, resulta necesario determinar el punto que hace máxima la primera derivada de la curvas de regresión. Dicho punto crítico, x_{0_l}, podría definirse para cada nivel del factor como

$$x_{0l} = \arg\max m_l^1(X)$$

En la práctica, el verdadero $m_l^1(x)$ no es conocido y las decisiones deben tomarse en base a su estimación $\hat{m}_l^1(x)$. Un estimador natural of de x_{0l} puede ser definido como el maximizador de

$$\hat{m}_1^r(k_1), \ldots, \hat{m}_1^r(k_N)$$

siendo $k_1, \ldots, k_N$ un grid fino de N puntos equidistantes en el rango de los valores de X.

2.1.1. Selección de ventana

La implementación práctica del estimador local polinómico kernel requiere de la selección de un parámetro de suavización o ventana h. Se sabe que las estimaciones no paramétricas obtenidas dependen en gran medida de la h utilizada. Valores pequeños de h producen un efecto negativo en la varianza del estiamdor, aumentando la misma y reproduciendo prácticamente los datos. Por otro lado, valores altos del parámetro dan lugar a un sesgo elevado produciendo un sobresuavizado de los datos. Por ello, este parámetro controla el compromiso entre el sesgo y la varianza y la selección óptima del mismo sigue siendo un problema abierto. En la literatura se han sugerido diversas técnicas para llevar a cabo esta selección. Por ejemplo, los métodos "plug-in" (Ruppert et al., 1995), la validación cruzada (Stone, 1977) o las técnicas bootstrap (Marron, 1992). Para un visión detalla, se recomienda Wand and Jones (1995).

Como solución práctica, en `NPRegfast`, se ha utilizado el método de validación cruzada. Concretamente, las ventanas $h_0, h_1, \ldots, h_M$ indispensable para la estimación de las funciones $f_0, f_1, \ldots, f_M$ han sido seleccionadas automáticamente minimizando

el siguiente criterio de error de validación cruzada

$$CV_0 = \sum_{i=1}^{n} \left(Y_i - \hat{f}_0^{(-i)}(X_i) \right)^2 \quad \text{y} \quad CV_l = \sum_{i=1}^{n} W_i^l \left(Y_i^l - \hat{f}_l^{(-i)}(X_i) \right)^2 \tag{2.4}$$

donde $\hat{f}_0^{(-i)}$ y $\hat{f}_l^{(-i)}$ son las estimaciones locales polinómicas tipo kernel dejando fuera el i-ésimo elemento de la muestra.

2.1.2. Aceleración computacional

Para acelerar los procesos de estimación y los procesos de selección de ventana explicados en las secciones anteriores, se ha utilizado la técnica lineal binning. Una detallada explicación de esta técnica puede encontrarse en Fan and Marron (1994). En este apartado, se desarrolla una breve explicación de los procedimientos que se han utilizado para las versiones binning de los estimadores $\hat{f}_0^r(x)$ y $\hat{f}_l^r(x)$ dados en (2.2) y (2.3), respectivamente.

En el primer paso del algoritmo, se considera un grid de N puntos equidistantes $X_1^\bullet < \ldots < X_N^\bullet$ y se construye la muestra binning $\left\{X_j^\bullet, Y_j^\bullet\right\}_{j=1}^N$ con pesos $\left\{W_j^\bullet\right\}_{j=1}^N$ donde

$$Y_j^\bullet = \sum_{i=1}^{n} \left(1 - \left|X_i - X_j^\bullet\right|/\delta\right)_+ Y_i \quad \text{y} \quad W_j^\bullet = \sum_{i=1}^{n} \left(1 - \left|X_i - X_j^\bullet\right|/\delta\right)_+$$

con $X_+ = \max\{0, X\}$ y δ denotando la distancia entre dos nodos vecinos. Las aproximaciones binning $\hat{f}_0^r(x)$ del primer paso del algoritmo (estimación de f_0) detallado en la Sección 2.1 se obtienen minimizando

$$\sum_{i=1}^{N} \left\{ Y_i^\bullet - \sum_{r=1}^{R+1} \alpha^r \left(X_i^\bullet - X\right)^{r-1} \right\}^2 \cdot K\left(\frac{X_i^\bullet - X}{h}\right) W_i^\bullet$$

De la misma manera, las aproximaciones $\hat{f}_l^r(x)$ en el segundo paso del algoritmo (estimación de f_l) se obtienen minimizando

$$\sum_{i=1}^{N} \left\{ Y_i^{\bullet l} - \sum_{r=1}^{R+1} \alpha_l^r \left(X_i^\bullet - X\right)^{r-1} \right\}^2 \cdot K\left(\frac{X_i^\bullet - X}{h}\right) W_i^{\bullet l}$$

donde $Y_i^{\bullet l} = Y_i^\bullet - \hat{f}_0\left(X_i^\bullet\right)$ y $W_i^{\bullet l} = W_i^\bullet I_{\{F_i = l\}}$.

Como en el proceso de estimación con la técnica binning, los errores de CV en (2.4) pueden ser respectivamente aproximados por

$$CV_0 \approx \sum_{i=1}^{N} W_i^{\bullet} \left(\frac{Y_i^{\bullet(-i)}}{W_i^{\bullet}} - \hat{f}_0^{(-i)}(X_i^{\bullet}) \right)^2 \text{ y } CV_j \approx \sum_{i=1}^{N} W_i^{\bullet l} \left(\frac{Y_i^{\bullet l(-i)}}{W_i^{\bullet l}} - \hat{f}_l^{(-i)}(X_i^{\bullet}) \right)^2$$

donde $Y_i^{\bullet l(-i)} = Y_i^{\bullet} - \hat{f}_0^{(-i)}(X_i^{\bullet})$ y las estimaciones $\hat{f}_0^{(-i)}$ y $\hat{f}_l^{(-i)}$ $(l = 1, \ldots, M)$ se obtienen dejando fuera el i-ésimo punto del grid.

Estas aproximaciones reducen sustancialmente el tiempo de computación ya que para el cálculo de CV_l solo es necesario evaluar el kernel K en un máximo de N puntos diferentes para cada elección de ventana. Nótese que cuanto más fino es el grid de puntos seleccionados, mejor es la aproximación. La elección del número de puntos en el grid es un compromiso entre el error en la aproximación y la velocidad computacional. En la práctica, la elección se hará atendiendo al tamaño de muestra n y de la distribución de la covariable.

Un estudio detallado del compromiso entre el tiempo de computación y el error de las aproximaciones binning puede encontrarse en De Uña Álvarez and Roca Pardiñas (2009). La conclusión que puede tomarse de este estudio es que, a medida que el número de nodos aumenta, los errores de la aproximación disminuyen, pero el tiempo de espera se incrementa substancialmente. Por otro lado, la reducción del error es muy pequeña a partir de unos nodos en adelante.

Atendiendo a este resultado, se propone elegir un número de nodos N_0 lo suficientemente grande para que no haya diferencias significativas entre la estimación obtenida con este valor $(N = N_0)$ y las estimaciones obtenidas con más nodos $(N > N_0)$. Por ejemplo, N_0 puede ser seleccionado como el primer nodo N que verifica $\sum_{i=1}^{n} |\hat{m}_{F_i,N} - \hat{m}_{F_i,N-1}|/\hat{m}_{F_i,N-1} \leq \varepsilon$, donde $\hat{m}_{F_N}$ representa la estimación de $\hat{m}_{F_N}$ obtenida usando N un grid de N en cada dirección y ε es un valor pequeño, por ejemplo, $\varepsilon = 0{,}001$. Según esta indicación, en la aplicación a datos desarrollada en este proyecto, se ha utilizado un N=100.

2.2. Inferencia

En esta sección, se proponen distintos contrastes basados en bootstrap que permiten verificar con significación estadística algunas de las características observadas en las estimaciones no paramétricas de las curvas anteriores: (a) contraste global de interacción (Subsección 2.2.1), (b) contraste de comparación para dos pares de curvas (Subsección 2.2.2), y (c) contrate para un modelo alométrico (Subsección 2.2.4).

Además, se desarrollará la metodología utilizada en la construcción de los intervalos de confianza bootstrap (Subsección 2.2.3).

2.2.1. Contraste global de interacción

Esta sección se centra en la implementación de un contraste de hipótesis que permita detectar interacciones factor-por-curva en el modelo de regresión en (2.1). El contraste se plantea unicamente entre las curvas de regresión iniciales, por lo que, en esta subsección, la notación de m_l^0 se convierte en unicamente m_l. La hipótesis nula planteada es la siguiente

$$H_0 : m_1(X) = \ldots = m_M(X)$$

y más concretamente, que el efecto de X no depende de los niveles del factor F. Así, el modelo bajo la hipótesis nula se correspondería con el siguiente

$$Y = f_0(X) + \varepsilon \tag{2.5}$$

Para contrastar dicha hipótesis se propone el uso de un estadístico basado en las estimaciones de las funciones m_l $(l = 1, \ldots, M)$

$$T = \sum_{l=1}^{M} \sum_{i=1}^{n} |\hat{f}_l(X_i)|$$

Nótese que el estadístico propuesto es una medida de la suma de los efectos específicos de todos los niveles $\hat{f}_l$, medida que fue forzada a ser cero (ver Subsección 2.1). Cabe destacar que, si se verifica la hipótesis nula, entonces el valor de T debe ser próximo a cero pero positivo. Consecuentemente, la regla del contraste basado en T consiste en rechazar la hipótesis nula si $T > T^{1-\alpha}$, siendo T^p el $p-$ percentil de T bajo la hipótesis nula. Sin embargo, es sabido que, en un contexto de regresión, la teoría asintótica utilizada para determinar los percentiles no es un tema cerrado, y las técnicas de remuestreo como el método bootstrap Efron (1979) (ver también Efron and Tibshirani, 1993; Härdle and Mammen, 1993; Kauermann and Opsomer, 2003) pueden ser aplicadas. Los pasos son los siguientes:

Paso 1. Se obtiene de la muestra $\{(X_i, F_i, Y_i)\}_{i=1}^n$ las estimaciones de $\hat{m}_{F_i}(X_i)$, para $i = 1, \ldots, n$ según el modelo (2.1) y, a su vez, $\hat{f}_l(X)$, y se computa el valor de T.

Paso 2. Se estima el modelo de regresión bajo la hipótesis nula H_0 en (2.5) y se

obtienen las estimaciones piloto $\hat{m}_{F_i}(X_i), i = 1, \ldots, n$.

Paso 3. Para $b = 1, \ldots, B$, se generan muestran bootstrap $\left\{X_i, F_i, Y_i^{\bullet b}\right\}_{i=1}^{n}$ con $Y_i^{\bullet b} = \hat{m}_{F_i}(X_i) + \varepsilon_i^{\bullet b}$, siendo

$$\hat{\epsilon}_i^{\bullet b} = \begin{cases} \hat{\epsilon}_i \cdot \frac{(1-\sqrt{5})}{2} & \text{with probability } p = \frac{5+\sqrt{5}}{10} \\ \hat{\epsilon}_i \cdot \frac{(1+\sqrt{5})}{2} & \text{with probability } p = \frac{5-\sqrt{5}}{10} \end{cases}$$

donde $\hat{\epsilon}_i = Y_i - \hat{m}_{F_i}(X_i)$ son los errores bajo H_0, y se calcula el estadístico bootstrap $T^{\bullet b}$. Nótese que $\hat{m}_{F_i}(X_i), i = 1, \ldots, n$ son las estimaciones obtenidas bajo la H_0.

La regla de decisión basada en T consiste en rechazar la hipótesis nula si $T > T^{1-\alpha}$, donde T^p es el p-percentil de los valores $T^{\bullet b}$ $(b = 1, \ldots, B)$, obtenidos en el **Paso 3**.

Adicionalmente, cuando se detecta una interacción factor-por-curva en un modelo, puede resultar interesante establecer en que puntos existen diferencias entre los efectos asociados a cada uno de los niveles del factor. En este trabajo, se propone para ello el uso de una medida de asociación. Tomando, por ejemplo, el primer nivel $F = 1$ como nivel de referencia y en base al modelo en (2.1), la medida propuesta puede ser considerada como la siguiente diferencia de curvas

$$dif_l(X) = f_l(X) - f_1(X) \quad l = 2, ...M \tag{2.6}$$

Estas curvas resultan de gran utilidad porque cada par de curvas puede ser reducido a una función de una sola dimensión que ofrece una fácil y atractiva interpretación. Además, las derivadas se obtienen directamente de las correspondientes derivadas de las funciones parciales de acuerdo con la fórmula

$$dif_l^r(X) = f_l^r(X) - f_1^r(X) \tag{2.7}$$

El procedimiento para estimar $dif_l^r(X)$ consiste unicamente en insertar en la expresión (2.7) las estimaciones resultantes del algoritmo tipo kernel de $f_l^r(X)$ y $f_1^r(X)$.

Para hacer inferencia sobre estas diferencias en cualquier punto X_i será necesario construir su correspondiente intervalo de confianza. La ausencia del cero en dicho intervalo indicará la existencia de diferencias significativas entre los dos niveles del factor en X_i. La construcción de estos intervalos se detalla en la Sección 2.2.3.

De manera análoga es posible comparar puntos críticos entre los distintos niveles del factor. El procedimiento para su estimación es equivalente pero insertando en la

expresión (2.7), por ejemplo, los máximos (rc_{0l}) para cada nivel obtenidos anteriormente.

2.2.2. Contraste de comparación para dos pares de curvas

Es importante resaltar que el intervalo de confianza calculado para $dif_l^r(X)$ representa un intervalo de confianza $100\,(1-\alpha)$ % para el verdadero valor de $dif_l^r(X)$ en cada uno de los valores de la variable X, pero que formalmente no permite hacer inferencia sobre estas curvas. Por ello, en esta sección se presenta un contraste para detectar significación estadística entre dos curvas, ya sean las estimaciones iniciales o sus derivadas.

Aunque el contraste global detecte la existencia de interacción, es decir, las curvas estimadas difieran entre niveles, puede que sus derivadas no lo hagan. Por este motivo, también podría resultar interesante aplicar este contraste para el caso de las derivadas de dos curvas.

En particular, para cada r=0, 1, 2, y tomando el primer nivel $F = 1$ como nivel de referencia, se considera la hipótesis nula

$$H_0^r(l) : dif_l^r(\cdot) = 0$$

Nótese que el uso de este contraste permite hacer inferencia sobre la forma funcional de la curva $dif_l^r(X)$ para cada nivel $l = 1, \ldots, M$ del factor F. Por ejemplo, (a) si $H_0^0(l)$ no es rechazada, esto significa que la curva m_l es la misma que la curva m_1 (curva de referencia) y si $H_0^0(l)$ es rechazada pero $H_0^1(l)$ no lo es, esto significa que dif_l^r es una constante.

Cabe destacar que si $H_0^r(l)$ es cierta entonces $dif_l^r\,(X) = dif_1^r\,(X)$, o equivalentemente $f_l^r(X) = f_1^r(X)$, y por lo tanto $f_l\,(X)$ sigue la forma $f_l\,(X) = f_1(X) + \sum_{j=1}^{r-1} \alpha_j X^j$ siendo α_j un parámetro desconocido. De acuerdo a esto, el modelo bajo la hipótesis nula viene dado por

$$Y = f_0\,(X) + f_1(X) + \textstyle\sum_{j=1}^{r-1} \alpha_j X^j + \varepsilon \tag{2.8}$$

Para contrastar $H_0^r(l)$ se propone un estadístico basado en las estimaciones directas no paramétricas de $\widehat{dif}_l^{\,r} = \hat{f}_l^r - \hat{f}_1^r$, del verdadero dif_l^r. El estadístico es el siguiente

$$S_r = \sum_{i=1}^{n} \left| \widehat{dif_l^r}\,(X_i) \right|$$

En base a cada estadístico S_r, con $r = 0, 1, 2$, la regla para contrastar la hipótesis nula con un nivel de confianza de α es rechazar la hipótesis nula si S_r es mayor que su α-percentil. Para determinar estos percentiles se aplicaron las técnicas bootstrap. El procedimiento bootstrap en este caso es el mismo que el presentado en la sección anterior. La única diferencia se encuentra en el Paso 2 del algoritmo, que ahora debe ser:

Paso 2. Se estima el modelo de regresión bajo la hipótesis nula H_0 en (2.8) y se obtienen las estimaciones piloto $\hat{m}_{F_i}(X_i), i = 1, \ldots, n$.

2.2.3. Intervalos de confianza

Una vez obtenidas las estimaciones de las curvas anteriores, para hacer inferencia sobre alguna característica relacionada con ellas, resulta imprescindible llevar a cabo la construcción de los intervalos de confianza. Estos intervalos se obtienen a partir de las estimaciones de m_l^r, y son útiles en diferentes contextos, como por ejemplo, para el valor de rc_{0l} o su diferencia entre dos niveles dados (Sección 2.1), para acotar los valores de $dif_l^r(X)$, o incluso para las estimaciones de $m_l^r(x)$ en un x. En cualquiera de estas situaciones, las construcción de los intervalos seguiría el mismo procedimiento.

En este trabajo, los intervalos de confianza han sido construidos utilizado el wild bootstrap (ver Härdle and Mammen, 1993; Härdle and Marron, 1991; Mammen, 1992). Este método de remuestreo es válido para modelos heterocedásticos donde la varianza de ε es una función de X. Los pasos para la construcción de estos intervalos de confianza para un valor R, obtenido del modelo (2.1) son los siguientes

Paso 1. Obtener las estimaciones de $\hat{R}$ de la muestra original.

Step 2. Para $b = 1, \ldots, B$ (p.ej. B=1000), se generan muestras bootstrap $\left\{(X_i, F_i, Y_i^{\bullet b})\right\}_{i=1}^n$ con $Y_i^{\bullet b} = \hat{m}_{F_i}(X_i) + \hat{\epsilon}_i^{\bullet b}$ siendo

$$\hat{\epsilon}_i^{\bullet b} = \begin{cases} \hat{\epsilon}_i \cdot \frac{(1-\sqrt{5})}{2} & \text{con probabilidad } p = \frac{5+\sqrt{5}}{10} \\ \hat{\epsilon}_i \cdot \frac{(1+\sqrt{5})}{2} & \text{con probabilidad } p = \frac{5-\sqrt{5}}{10} \end{cases}$$

donde $\hat{\epsilon}_i = Y_i - \hat{m}_{F_i}(X_i)$ son los errores del modelo no paramétrico, y se computa $\hat{R}^{\bullet b}$ de la misma manera que en el **Paso 1**.

Finalmente, el intervalo de confianza al $100(1-\alpha)\,\%$ para R viene dado por

$$I = \left(\widehat{R}^{\alpha/2}, \widehat{R}^{1-\alpha/2}\right)$$

donde $\hat{R}^p$ representa el percentil p de $\hat{R}^{\bullet 1}, \ldots, \hat{R}^{\bullet B}$.

2.2.4. Contraste para un modelo alométrico

En las secciones anteriores se han descrito distintas cuestiones relacionas con los modelos de regresión no paramétrica. Sin embargo, existen otros modelos de tipo paramétrico que, según el contexto, pueden ser adecuados para modelar un conjunto de datos. Uno de los más utilizados en el ámbito de la biología, con el que se suele estudiar la relación entre dos variables biométricas, es el modelo alométrico clásico, $Y = aX^b$, propuesto por Huxley (1924), que normalmente se convierte a su expresión logarítmica

$$logY = log\, a + b\, logX = a^* + b^* logX + \varepsilon \tag{2.9}$$

siendo a una constante y b el exponente de la forma aritmética de la ecuación y la pendiente de la recta de regresión en su forma logarítmica. Esta conversión, simple tanto conceptual como matemáticamente, facilita la estimación de sus parámetros por regresión lineal. Una vez obtenidas $\hat{a}^*$ y $\hat{b}^*$ ajustando el modelo en (2.9), se vuelve a la escala original de los parámetros, $\hat{a} = exp(\hat{a}^*)$ y $\hat{b} = \hat{b}^*$, y se obtiene el modelo estimado $\widehat{Y} = \hat{a}X^{\hat{b}}$. Además, la estimación de la derivada de Y vendrá dada por $\widehat{Y}' = \hat{a}\hat{b}X^{\hat{b}-1}$.

A pesar del hecho de que estos modelos paramétricos son frecuentemente utilizados, existe un problema asociado a su uso. En ciertas situaciones, la asunción de una determinada curva en los efectos de una covariable resulta muy restrictiva y no es soportada por los datos. En este contexto, las técnicas de regresión no paramétrica se encargan de modelar la dependencia entre Y y X sin especificar de antemano la función que une la covariable a la respuesta.

Con el fin de facilitar la elección de un modelo adecuado a los datos, minimizando así la pérdida de información, se ha desarrollado un test basado en bootstrap que contrasta si los datos siguen un modelo alométrico clásico. En este caso, por simplicidad, no se han tenido en cuenta la interacción y se ha asumido el siguiente modelo general

$$Y = m(X) + \varepsilon \tag{2.10}$$

donde ε es el error que se asume de media cero y $m(X)$ es una función suave y desconocida.

El objetivo es contratar la hipótesis nula de un modelo alométrico $H_0 : m(X) = aX^b$ $versus$ la hipótesis alternativa H_1 siendo m una función no paramétrica desco-

nocida, o de manera análoga, $H_1 : m(X) = aX^b + g(X)$.

Para contrastar H_0 se propone el uso del siguiente estadístico

$$Q = \sum_{i=1}^{n} |\hat{g}(X_i)| \tag{2.11}$$

La regla para contrastar H_0 con un nivel de confianza de α es que se rechaza la hipótesis nula si Q es mayor que su α-percentil. Para aproximar los valores de la distribución del estadístico bajo la hipótesis nula se utilizaron las técnicas bootstrap, en este caso el wild bootstrap (Härdle and Mammen, 1993; Härdle and Marron, 1991; Mammen, 1992). Los pasos del procedimiento son los siguientes:

Paso 1. Obtener de la muestra $\{(X_i, Y_i)\}_{i=1}^n$ las estimaciones del modelo en (2.10) y computar el valor de Q.

Paso 2. Para $b = 1, \ldots, B$ (e.g. B=1000), generar muestras bootstrap $\left\{(X_i, Y_i^{\bullet b})\right\}_{i=1}^n$ con $Y_i^{\bullet b} = \hat{a}X_i^{\hat{b}} + \hat{\epsilon}_i^{\bullet b}$ siendo

$$\hat{\epsilon}_i^{\bullet b} = \begin{cases} \hat{\epsilon}_i \cdot \frac{(1-\sqrt{5})}{2} & \text{with probability } p = \frac{5+\sqrt{5}}{10} \\ \hat{\epsilon}_i \cdot \frac{(1+\sqrt{5})}{2} & \text{with probability } p = \frac{5-\sqrt{5}}{10} \end{cases}$$

donde $\hat{\epsilon}_i = Y_i - \hat{a}X_i^{\hat{b}}$ son los errores del modelo alométrico, y computar $Q^{\bullet b}$ de la misma manera que en el **Paso 1**.

Finalmente, el regla del contraste basada en Q consiste en rechazar la hipótesis nula si $Q > Q^{1-\alpha}$, donde T^p es el $p-$percentil empírico de los valores de $Q^{\bullet b}(b = 1, ..., B)$ obtenidos anteriormente.

Capítulo 3

Desarrollo de software

En el capítulo anterior se ha expuesto la metodología utilizada para llevar a cabo la estimación de los modelos de regresión con interacciones factor-por-curva y los distintos contrastes desarrollados. El presente capítulo centra su atención en como se puede llevar a cabo un análisis práctico en el ambiente de computación `R` (R Development Core Team, 2009).

Ya que los procesos de estimación de las metodologías desarrolladas implican un elevado gasto computacional, se ha utilizado Fortran (FORmula TRANslation, Fortran 95 Language Guide, 1995) como lenguaje de programación. Sin embargo, para facilitar su uso en la práctica, estas metodología se han implementado en un paquete de `R` amigable para el usuario final, `NPRegfast`. Este software aporta salidas numéricas y gráficas de los modelos de regresión revisados en el Capítulo 2.

El diseño del paquete `NPRegfast` se ha realizado siguiendo las pautas de otras funciones o paquetes de regresión en `R`. De acuerdo a esto, la función principal de esta librería es la función `frfast()`, que ajusta, por defecto, un modelo de regresión no paramétrica mediante suavizadores tipo núcleo. Los resúmenes numéricos y gráficos del objeto ajustado se pueden obtener utilizando las funciones `print.frfast()`, `summary.frfast()` y `plot.frfast()`. Los máximos de las estimaciones y de sus derivadas, así como las diferencias entre ellos por niveles del factor, se obtienen con la función `maxp()` y `maxp.diff()`, respectivamente. La salida gráfica con las diferencias entre niveles del factor, tanto para las estimaciones como para sus derivadas, se obtienen con la función `plot.diff()`. Por último, la función `contrast()` realiza un contraste donde la hipótesis nula se corresponde con que los datos siguen un modelo alométrico.

A continuación, se presenta en detalle la librería `NPRegfast`. El uso de este paquete

se ilustra con la base de datos `barnacle`, con medidas de longitud rostro-carenal (RC) y peso seco (DW) de 6686 percebes recogidos en dos localidades del litoral atlántico gallego.

3.1. Función `frfast()`

La función principal del paquete es `frfast()` que crea un objeto de la clase `frfast`. Esta función ajusta, por defecto, un modelo de regresión no paramétrica utilizando para ello suavizadores locales lineales tipo kernel. Los argumentos principales de esta función se presentan en la Tabla 3.1. La llamada a la función es la siguiente

```
frfast(x, y, f = NULL, model=1, h=NULL, w =NULL, p=2,kbin=100,
nc=NULL, ncmax=5, ikernel=1, iopt=1, nboot=500, c2=NULL,
rankl=NULL, ranku=NULL)
```

Nótese que por medio del argumento `f`, el usuario puede decidir ajustar un modelo de regresión entre la variable respuesta `y` y la covariable `x` con interacción o sin ella, mientras que con el argumento `model` se indica el tipo de ajuste, no paramétrico (`model`=1) o alométrico clásico (`model`=2).

La sintaxis concreta con la base de datos `barnacle` se muestra a continuación. En el ejemplo se ajusta, de manera no paramétrica, un modelo entre la talla de los percebes (RC) y su peso seco (DW), utilizando la localidad como factor (F).

```
R> library(NPRegfast)
R> data(barnacle)
R> summary(barnacle)
       F               RC              PS
 Min.   :1.000   Min.   : 1.9   Min.   :0.000
 1st Qu.:1.000   1st Qu.: 8.3   1st Qu.:0.130
 Median :2.000   Median :12.6   Median :0.410
 Mean   :1.515   Mean   :12.3   Mean   :0.558
 3rd Qu.:2.000   3rd Qu.:16.1   3rd Qu.:0.870
 Max.   :2.000   Max.   :25.1   Max.   :3.140

R> fit<-frfast(x=barnacle$RC,y=barnacle$PS,f=barnacle$F)
```

Un resumen numérico de los resultados del ajuste se obtiene llamando a las funciones `print.frfast()` o `summary.frfast()`.

Argumento	Descripción
`x`	Vector con los valores de la variable explicativa. Valores perdidos no están permitidos.
`y`	Vector con los valores de la variable respuesta. Valores perdidos no están permitidos.
`f`	Vector con los valores del factor a tener en cuenta en el modelo.
`model`	"`np`" ajusta un modelo de regresión no paramétrico mediante suavizadores locales lineales tipo núcleo. "`alo`" ajusta un modelo alométrico.
`h`	Parámetro de suavización o ventana. Valores grandes de ventana se corresponden con estimaciones suaves, valores pequeños se corresponden con estimaciones rugosas. Por defecto, el valor de la ventana se selecciona por validación cruzada.
`w`	Vector con los pesos asociados a cada dato.
`p`	Grado del polinomio.
`kbin`	Número de nodos binning.
`ikernel`	Argumento que determina el tipo de núcleo. `ikernel=1` se corresponde con el núcleo Gaussiano.
`nboot`	Número de réplicas bootstrap.
`rankl`	Número o vector con el que se especifica el mínimo valor del intervalo donde se busca el valor de la `x` que maximiza la estimación, primera y segunda derivada (para cada nivel). Por defecto se utiliza el mínimo valor de `x`.
`ranku`	Número o vector con el que se especifica el máximo valor del intervalo donde se busca el valor de la `x` que maximiza la estimación, primera y segunda derivada (para cada nivel). Por defecto se utiliza el máximo valor de `x`.

Tabla 3.1: Resumen de los argumentos de la función `frfast`.

```
R> fit
Call:
frfast(x=barnacle$RC,y=barnacle$PS,f=barnacle$F)

*******************************************
Nonparametric Model
*******************************************
Bandwidth: 74.3 75.6
Degree of polinomium: 2
Number of bootstrap repeats: 500
Number of binning nodes 100


The number of data is:  6686
The factor's levels are:  1 2
The number of data for the level 1 is: 3242
The number of data for the level 2 is: 3444

Summaries for the variable y (for each level):
Level 1 :
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
 0.0000  0.1300  0.4100  0.5697  0.9000  2.8700

Level 2 :
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
 0.0000  0.1200  0.4100  0.5469  0.8500  3.1400
```

Como se puede observar, `summary.frfast()` devuelve la llamada a la función, el tipo de ajuste realizado (alométrico o no paramétrico), la ventana utilizada, el grado del polinonio, el número de replicas bootstrap y el número de nodos binning empleados en la estimación. En el ejemplo, se observan los valores utilizados por defecto. Se muestra también un pequeño resumen numérico para cada uno de los niveles del factor.

3.2. Función `plot.frfast()`

La función `plot.frfast()` dibuja la estimación, primera y segunda derivada para cada uno de los niveles del factor a partir de un objeto de la clase `frfast`. Dichas estimaciones se muestran con los intervalos de confianza al 95 % obtenidos mediante bootstrap. Los principales argumento de esta función se muestran en la Tabla 3.2.

Los resultados del siguiente código se pueden observar en la Figura 3.1.

```
R> plot(fit,der=c(0,1),xlab="RC (mm)",ylab="DW (g)")
```

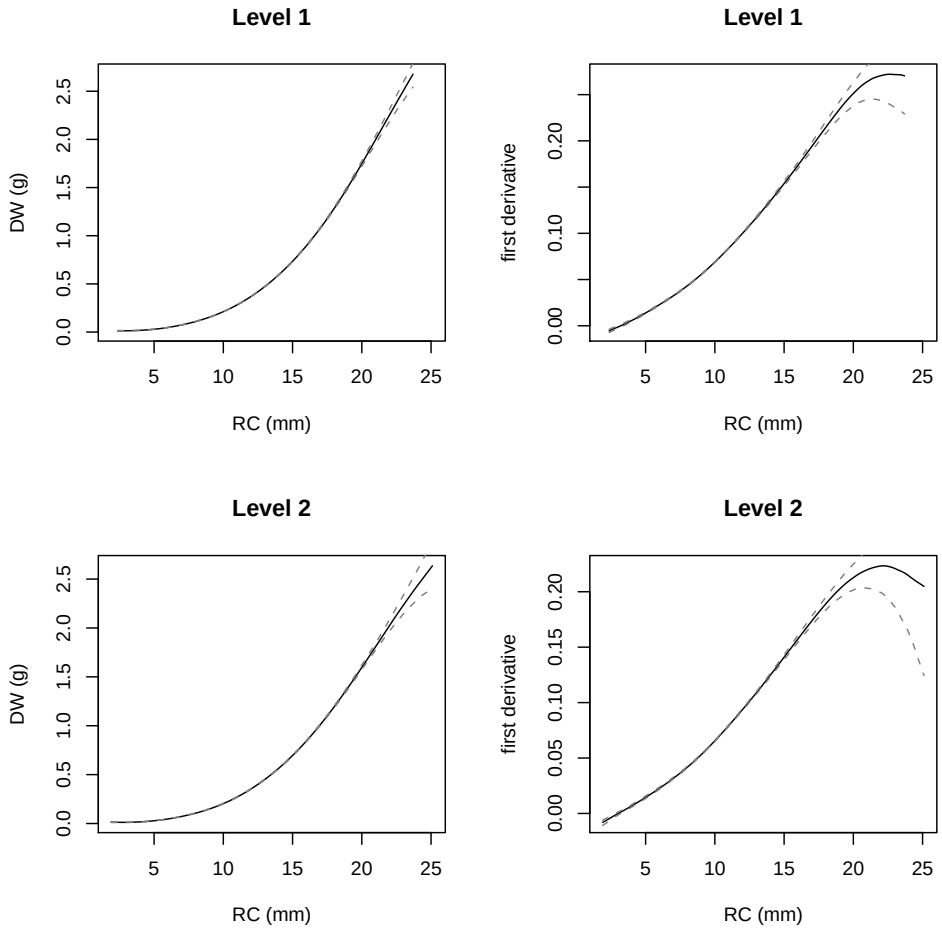

Figura 3.1: Estimación y primera derivada (líneas continuas) con intervalos de confianza al 95 % (líneas discontinuas) de la relación entre la talla y el peso de los percebes. Level 1: localidad 1. Level 2: localidad 2.

Argumento	Descripción
`model`	Objeto de la clase `frfast`.
`fac`	Número o vector que determina el nivel del factor que se pretende dibujar.
`der`	Número o vector que determina el proceso de inferencia. Por defecto es `NULL`, obteniéndose un gráfico con la estimación, primera y segunda derivada. Si el término es `0`, el gráfico muestra la estimación inicial. Si es `1` o `2`, se dibuja la primera o segunda derivada, respectivamente.
`xlab`	Título para el eje x.
`ylab`	Título para el eje y.
`col`	Especificación para el color del gráfico.
`ICcol`	Especificación para el color de los intervalos de confianza.
`main`	Título del gráfico.
`type`	Tipo de gráfico que se debe utilizar. Los tipos posibles pueden ser, `p` para puntos, `l` para líneas, etc. Ver detalles en ?`par`.
`ICtype`	Tipo de gráfico que se puede utilizar para los intervalos de confianza. Ver detalles en ?`par`.
`...`	Otras opciones.

Tabla 3.2: Resumen de los argumentos de la función `plot.frfast`.

3.3. Función `maxp()`

Con la función `maxp()` es posible estimar el valor de la covariable `x` que hace máxima la estimación inicial, primera y segunda derivada, para cada uno de los niveles del factor. Para hacer inferencia sobre dichos puntos, se calculan sus intervalos de confianza al 95 % mediante bootstrap. Los argumento de la función se presentan en la Tabla 3.3.

```
R> maxp(fit,der=1)
         Max point 95% IC_lower 95% IC_upper
Level 1  22.75445     21.28036           NA
Level 2  22.12743     20.78048           NA
```

En el código superior se muestra un ejemplo de uso de esta función. Si en la salida se muestra un `NA`, éste indica que no ha sido posible calcular el correspondiente valor. Si el `NA` se corresponde con la estimación significa que el máximo de la curva

se alcanza en el último valor de la variable x. En el caso de que el NA se encuentre en algún extremo del intervalo, debe deducirse que los máximos correspondientes a las réplicas bootstrap son mayores que el máximo de la covariable y por ello, se omite dicho calor.

Argumento	Descripción
model	Objeto de la clase frfast obtenido al ajustar un modelo paramétrico o no paramétrico a los datos.
der	Número que determina el proceso de inferencia. Por defecto es NULL, con lo que se muestra el máximo tanto para la estimación inicial como para ambas derivadas. Si el término es 0, el gráfico muestra la estimación inicial. Si es 1 o 2, se muestra el máximo para la primera o segunda derivada, respectivamente.

Tabla 3.3: Argumentos de la función maxp.

3.4. Función plot.diff()

Con la función plot.diff() el usuario puede visualizar las diferencias entre las curvas estimas para dos niveles dados del factor, tanto para la estimación inicial como para la primera o segunda derivada. Su salida muestra tres gráficos, los dos primeros se corresponden con las estimaciones para cada uno de los factores especificados en los argumentos, y el tercero muestra la curva de las diferencias de las estimaciones anteriores para cada valor de x. Los intervalos de confianza para esa diferencia se obtienen mediante bootstrap. Nótese que la ausencia del cero en el intervalo sugiere la existencia de diferencias significativas entre los dos niveles del factor en ese valor de la x. Los argumentos principales de plot.diff() se muestran en la Tabla 3.4.

Los resultados del siguiente código se muestran en la Figura 3.2.

```
R> plot.diff(fit,1,2,der=1,xlab="RC (mm)",ylab="DW (mm)")
```

Argumento	Descripción
`model`	Objeto de la clase `frfast`.
`factor1`	Primer nivel del factor para el que se estimarán las diferencias entre las curvas.
`factor2`	Segundo nivel del factor para el que se estimarán las diferencias entre las curvas.
`der`	Número o vector que determina el proceso de inferencia. Por defecto es `NULL`, obteniéndose un gráfico con la estimación, primera y segunda derivada. Si el término es `0`, el gráfico muestra la estimación inicial. Si es `1` o `2`, se dibuja la primera o segunda derivada, respectivamente.
`ylab`	Título para el eje y.
`xlab`	Título para el eje x.
`col`	Especificación para el color del gráfico.
`ICcol`	Especificación para el color de los intervalos de confianza.
`main`	Título del gráfico.
`type`	Tipo de gráfico que se debe utilizar. Los tipos posibles pueden ser, `p` para puntos, `l` para líneas, etc. Ver detalles en ?`par`.
`ICtype`	Tipo de gráfico que se puede utilizar para los intervalos de confianza. Ver detalles en ?`par`.
`...`	Otras opciones.

Tabla 3.4: Argumentos de la función `plot.diff`.

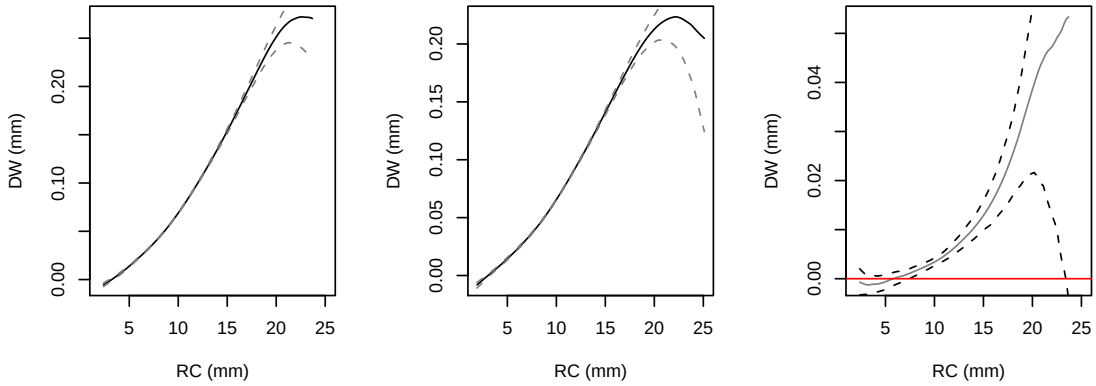

Figura 3.2: Panel izquierda: primera derivada (línea continua) con intervalos de confianza (IC) al 95 % (línea discontinua) de la relación entre la talla y el peso de los percebes para la localidad 1. Panel central: primera derivada (línea continua) con IC al 95 % (línea discontinua) de la relación entre la talla y el peso de los percebes para la localidad 2. Panel derecho: diferencias (línea continua) con IC al 95 % (línea discontinua) entre las dos curvas anteriores. Linea roja: $y = 0$.

3.5. Función `maxp.diff()`

Una vez estimados los puntos de la variable `x` donde se alcanza el máximo de las curvas para cada nivel del factor, la función `maxp.diff()` permite al usuario estimar la diferencia entre dichos máximos para cada dos niveles. Con el fin de poder hacer inferencia, se han construido los IC para esa diferencia entre máximos, también aplicando las técnicas bootstrap. Si el intervalo de confianza contiene al cero, el valor de la `x` donde se alcanza el máximo es el mismo para los dos niveles comprobados. Los argumentos de esta función se presentan en la Tabla 3.5.

El siguiente código muestra las diferencias (con sus IC) entre los máximos de la primera derivada en las dos localidades de ejemplo.

```
> maxp.diff(fit,der=2)

           Factor2 Factor1 Max points Diff. 95% IC_lower Diff.
First_der       2       1           -0.627             -2.475
           95% IC_upper Diff.
First_der               3.41
```

Cabe destacar que las diferencias se obtienen de un factor en relación a otro y siempre como `factor2-factor1`.

Argumento	Descripción
`model`	Objeto de la clase `frfast`.
`factor1`	Primer nivel del factor para el que se estimarán las diferencias entre los puntos máximos.
`factor2`	Segundo nivel del factor para el que se estimarán las diferencias entre los puntos máximos.
`der`	Número que determina el proceso de inferencia. Por defecto es `NULL`, obteniéndose un gráfico con la estimación, primera y segunda derivada. Si el término es `0`, se muestra la diferencia para las estimaciones iniciales. Si es `1` o `2`, se referirá a la primera y segunda derivada, respectivamente.

Tabla 3.5: Resumen de los argumentos de la función `maxp.diff`.

3.6. Función `contrast()`

La función `contrast()` realiza un test basado en bootstrap que contrasta si los datos siguen un modelo paramétrico clásico. El usuario puede utilizar esta función para contrastar la hipótesis nula de un modelo alométrico, $H_0 = m(X) = aX^b$ *vs.* la hipótesis alternativa, H_1, siendo m una función no paramétrica desconocida. La técnica de remuestreo utilizada para implementar este contraste ha sido el wild bootstrap. Los argumentos de la función se muestran en la Tabla 3.6.

Argumento	Descripción
`x`	Vector con los valores de la variable explicativa.
`y`	Vector con los valores de la variable respuesta.
`f`	Vector con los valores del factor a tener en cuenta en el modelo. Por defecto es `NULL`.

Tabla 3.6: Argumentos de la función `contrast`.

En el siguiente código se muestra un ejemplo de aplicación donde se contrasta si alguno de los dos niveles del factor (localidad) sigue un modelo alométrico.

```
R> contrast(x=barnacle$RC,y=barnacle$PS,f=barnacle$F)
         Statistic p-value
Level 1     0.010   0.010
Level 2     0.016   0.016
```

Capítulo 4

Aplicación a datos reales

La metodología implementada en `NPRegfast` ha sido utilizada para estimar la relación talla-peso en *Pollicipes pollicipes* y determinar a su vez una talla de captura mínima para esta especie.

Para ello, se recogieron muestras de percebes en la zona intermareal de cinco localidades representativas del litoral atlántico gallego, que corresponden con tramos de costa donde esta especie es explotada. El estudio se desarrolló a lo largo de dos años, de enero de 2006 a diciembre de 2007, procurando mantener una periodicidad de muestreo mensual.

De cada uno de los individuos se midieron las siguientes variables biométricas: longitud rostro-carenal (RC, máxima distancia a lo largo del capítulo entre el rostro y la carena, y variable que mejor representa el crecimiento de la especie (Cruz, 1993, 2000)), y peso seco (DW), obtenido secando los individuos en una estufa de aire forzado durante 24 horas a 100 ^{o}C (Montero-Torreiro and Martínez, 2003). Todas las medidas fueron tomadas utilizando un calibre digital de 0.1 mm, y una balanza de precisión de 0.01 g. El número total de individuos medidos fue de 16562.

La relación que define el crecimiento en peso de una especie con respecto a su talla es una de las más frecuentemente utilizadas en pesquerías y es un importante elemento en dinámica de poblaciones (Oniye et al., 2006). Por ello, esta relación talla-peso ha sido estudiada en varias especies marina utilizando diferentes modelo paramétricos que resultan fáciles de aplicar y estimar (i.e. Nieto-Navarro et al., 2010; Ramón et al., 2010; Pinheiro and Fiscarelli, 2009; Ismen et al., 2007; Neves et al., 2009; Froese, 2006). Uno de los modelos de este tipo, más ampliamente utilizados, es el modelo alométrico, $DW = aRC^b$, propuesto por Huxley (1924), que normalmente se convierte a su expresión logarítmica. Esta conversión, simple tanto conceptual como

matemáticamente, facilita la estimación de sus parámetros por regresión lineal.

A pesar del hecho de que estos modelos paramétricos son frecuentemente utilizados, existe un problema asociado a su uso. En ciertas situaciones, la asunción de una determinada curva en los efectos de una covariable resulta muy restrictiva y no es soportada por los datos. En este contexto, las técnicas de regresión no paramétrica se encargan de modelar la dependencia entre DW y RC sin especificar de antemano la función que une la covariable a la respuesta. Por ello, para estimar la ganancia en peso de los individuos a medida que incrementan su talla, en este estudio se propone el uso de un modelo no paramétrico general del tipo

$$DW = m(RC) + \varepsilon \tag{4.1}$$

donde m es una función suave y ε es el error, que se asume con media cero y varianza en función de la covariable RC. Nótese que en este tipo de modelo no es necesario establecer una forma paramétrica de m, y que además, un caso específico de (4.1) es el modelo alométrico anidado obtenido usando $m(RC) = aRC^b$.

En la Figura 4.1 se representan las curvas de regresión estimadas de los dos modelos anteriores junto con sus derivadas. Las líneas grises y negras se refieren al modelo alométrico y no paramétrico, respectivamente. Como se puede observar, las curvas de ambos modelos son funciones monótonas crecientes, donde los valores de DW aumentan con el aumento de los valores RC. Sin embargo, el modelo no paramétrico detecta variaciones en la parte final de la figura que el modelo alométrico no es capaz de detectar.

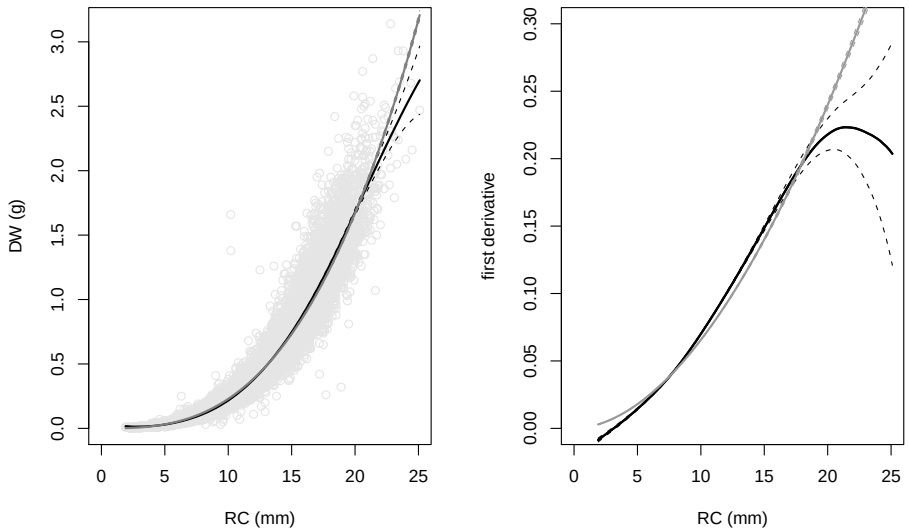

Figura 4.1: Curvas de regresión y primeras derivadas (líneas continuas) con IC bootstrap al 95 %(líneas discontinuas) para el peso seco y la longitud rostro-carenal. Líneas grises: modelo alométrico. Líneas negras: modelo no paramétrico.

Para comprobar cual de los modelos propuestos explica con mayor detalle la información aportada por los datos, se aplicó uno de los test implementado que contrasta la hipótesis nula $H_0 : DW = aRC^b$. El resultado obtenido es que, para un nivel de confianza del 5 %, la hipótesis nula es rechazada (p-valor $< 0{,}01$). Con base en estos resultados, el uso del modelo no paramétrico podría ser una buena alternativa frente al modelo clásico. La Figura 4.2, que muestra el estudio global basado en todos los datos, refleja la curva de regresión no paramétrica de la ganancia en peso con el aumento de RC, con base en el modelo propuesto.

Es importante subrayar el hecho de que el aumento en peso por unidad de RC (dado por la primera derivada de m) registra un máximo en una determinada talla, denominada rc_0, a partir de la cual, la ganancia en peso disminuye (o por lo menos se mantiene constante). Consecuentemente, este estudio propone que la talla mínima de captura nunca debería ser menor que este rc_0. En el estudio global, este rc_0 se corresponde con un RC de 21.5 mm (línea vertical continua de la Figura 4.2).

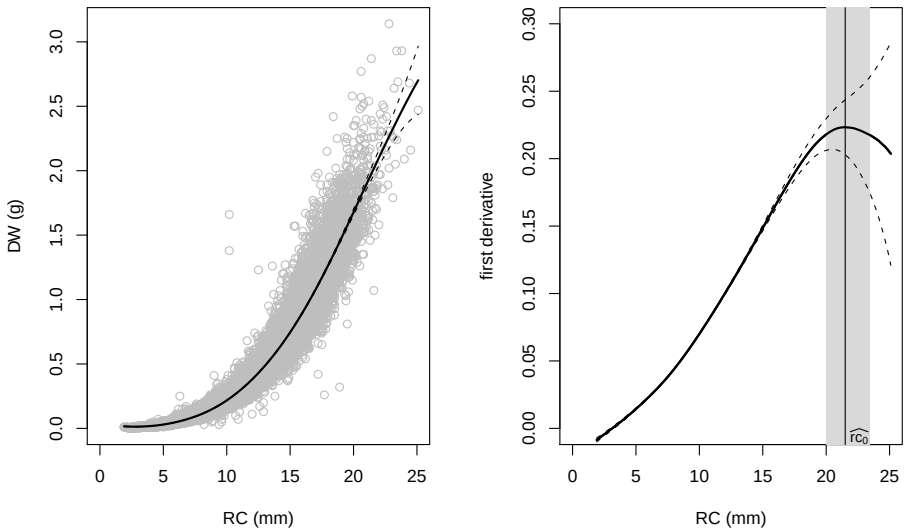

Figura 4.2: Curva de regresión y primera derivada (líneas continuas) con intervalos de confianza bootstrap al 95 % (líneas discontinuas) para el peso seco y la longitud rostro-carenal (estudio global). Línea vertical continua: rc_0 estimado. Área sombreada: intervalo de confianza construido para $\widehat{rc_0}$.

Para comprobar si la relación talla-peso permanece constante a lo largo del tiempo y no es alterada por una posible variabilidad anual en el crecimiento de esta especie, el estudio fue repetido incluyendo la interacción con el factor año (primera y segunda fila de la Figura 4.3, año 2006 y 2007, respectivamente). En esta situación, resulta importante determinar si realmente existe un efecto en la respuesta correspondiente al factor o si las curvas anteriores son prácticamente la misma. Para ello, se ha aplicado los contrastes desarrollados en el Capítulo 2. El p-valor obtenido es menor que 0.01 en ambos casos, por lo que se rechaza la hipótesis nula de igualdad de curvas.

Aunque las curvas anteriores muestren diferencias entre años, en este estudio resulta de gran interés comprobar si la talla buscada (rc_0) es la misma para ambos periodos. En este caso se ha utilizado la diferencia entre puntos críticos. El estadístico obtenido es de 0.0812 (-3.2264, 3.1562). Estos valores sugieren que, aunque exista un efecto del factor en la respuesta, y las curvas y sus derivadas sean distintas entre el año 2006 y 2007, la talla en la que se alcanza el máximo rendimiento en peso es significativamente igual.

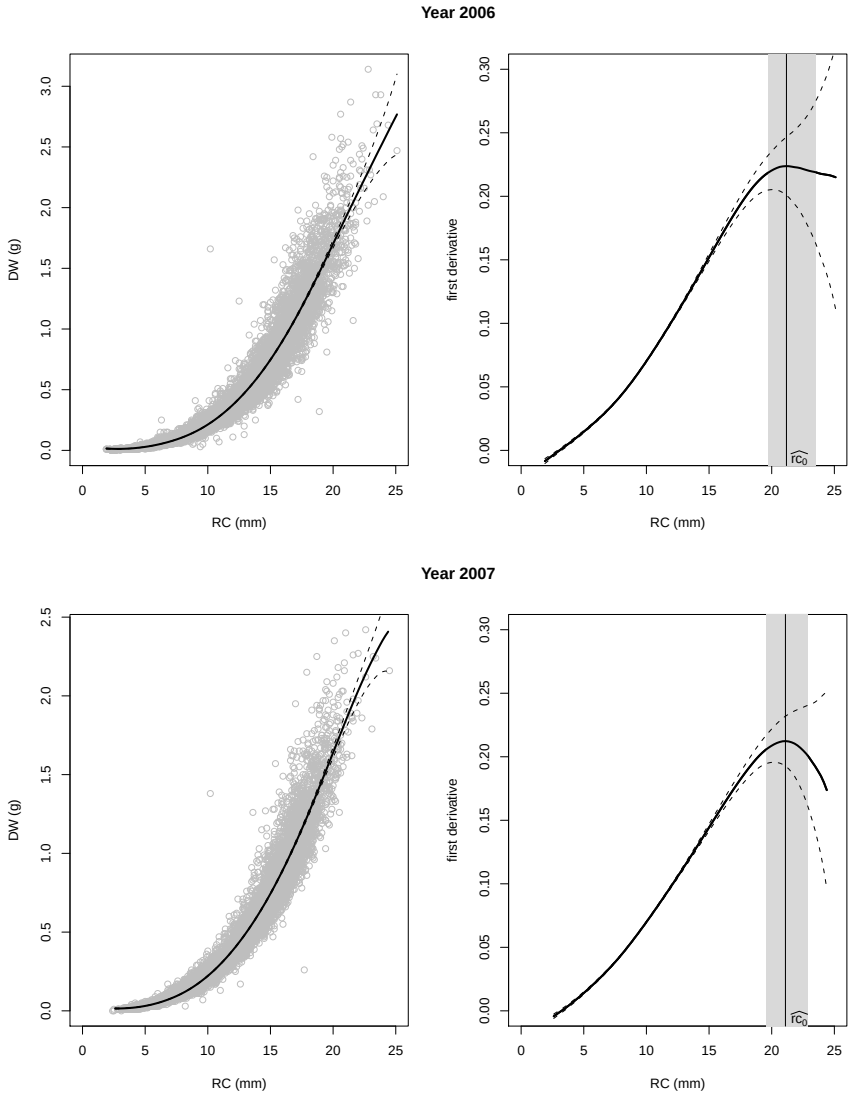

Figura 4.3: Curva de regresión y primera derivada (líneas continuas) con intervalos de confianza bootstrap al 95 % (líneas discontinuas) para el peso seco y la longitud rostro-carenal. Primera fila: año 2006; segunda fila: año 2007. Línea vertical continua: rc_0 estimado. Área sombreada: intervalo de confianza construido para $\widehat{rc_0}$. Año 2006: 21.18 (19.75,23.56). Año 2007: 21.10 (19.60,22.89).

Este estudio describe una nueva aproximación para estimar la relación talla-peso en esta especie basándose en el uso de un modelo no paramétrico. Los resultados obtenidos indican que modelar los datos de manera no paramétrica supone ser capaz de detectar efectos de los valores finales de la distribución, mientras que el uso de modelos más rígidos, como el modelo clásico alométrico, podrían distorsionar la relación estudiada. En el ejemplo utilizado en esta aplicación, la elección arbitraria del modelo alométrico habría causado la pérdida de gran parte de la información. Según esto se sugiere que la relación talla-peso puede ser explicada fiablemente por un modelo no paramétrico.

Con base en el modelo anterior, se ha propuesto también un método para estimar una talla de captura para este crustáceo y se ha constatado, por medio de los diversos contrastes aplicados, que aunque el crecimiento de la especie pueda diferir entre años, la talla donde se alcanza su máximo rendimiento en peso es constante con el paso del tiempo.

Capítulo 5

Conclusión

El paquete `NPRegfast` presentado en este documento es el resultado de la implementación en `R` de la metodología desarrollada para la estimación de modelos de regresión con interacciones factor-por-curva.

Dicha metodología centra su atención en el planteamiento y ejecución de nuevos contrastes que permitan estudiar diversas características relacionadas con las curvas de regresión y sus derivadas.

El primer test desarrollado se centra en contrastar de manera global la presencia o ausencia de interacción, o dicho de otro modo, determina si el efecto de la covariable depende del factor. En el caso de existir interacción, la siguiente cuestión que se plantea es comprobar si existen diferencias entre dos curvas dadas, pudiendo contrastar la igualdad de las estimaciones iniciales o bien sus derivadas. Por último, se ha planteado un contraste más específico, indicado para facilitar la elección de un modelo (alométrico o no paramétrico) adecuado a un conjunto de datos.

Bibliografía

Anderson, D.T., 1994. Barnacles: structure, function, development and evolution. Chapman & Hall, London.

Barnes, M., 1996. Pedunculate cirripedes of the genus *Pollicipes*. Oceanography and Marine Biology: An Annual Review 34, 303–394.

Bernard, F.R., 1988. Potential fishery for the gooseneck barnacle *Pollicipes polymerus* (Sowerby, 1833) in British Columbia. Fisheries Research 6, 287–298.

Cardoso, A.C., Yule, A.B., 1995. Aspects of the reproductive biology of *Pollicipes pollicipes* (Cirripedia; Lepadomorpha) from the southwest coast of Portugal. Netherlands Journal of Aquatic Ecology 29, 391–396.

Cruz, T., 1993. Growth of *Pollicipes pollicipes* (Gmelin, 1790) (Cirripedia, Lepadomorpha) on the SW coast of Portugal. Crustaceana 65, 151–158.

Cruz, T., 2000. Biologia e ecologia do percebe, *Pollicipes pollicipes* (Gmelin, 1790), no litoral sudoeste português. Ph.D. thesis. Universidad de Évora.

Cruz, T., Araujo, J., 1999. Reproductive patterns of *Pollicipes pollicipes* (Cirripedia: Scalpellomorpha) on the southwestern coast of Portugal. Journal of Crustacean Biology 18, 260–267.

Darwin, C., 1851. A monograph on the subclass *Cirripedia*, with figures of all the species. The *Lepadidae*; or, pedunculated cirripedes. London: The Ray Society.

Darwin, C., 1854. A monograph on the subclass *Cirripedia*, with figures of all the species. The *Balanidae*; or, sessile cirripedes; the *Verrucidae*, etc. . London: The Ray Society.

De Uña Álvarez, J., Roca Pardiñas, J., 2009. Additive models in censored regression. Computational Statistics & Data Analysis 53, 3490–3501.

Efron, B., 1979. Bootstrap methods: another look at the jackknife. Annals of Statistics 7, 1–26.

Efron, E., Tibshirani, R.J., 1993. An introduction to the Bootstrap. Chapman and Hall, London.

Fan, J., Marron, J., 1994. Fast implementation of nonparametric curve estimators. Journal of Computational and Graphical Statistics 3, 35–56.

Froese, R., 2006. Cube law, condition factor and weight-length relationships: history, meta-analysis and recommendations. Journal of Applied Ichthyology 22, 241–253.

Gehrke, W., 1995. Fortran 95 Language Guide. Springer, London.

Girard, S., 1982. Etude du stock de pouces-pieds de Belle-île et de son exploitation. Master's thesis. Mémoire de fin d'études. ENSAR.

Goldberg, H., 1984. Posibilidades de cultivo de percebe, *Pollicipes cornucopia* Leach, en sistemas flotantes. Informes Técnicos del Instituto Español de Oceanografía 11, 1–13.

Härdle, W., Mammen, E., 1993. Comparing nonparametric versus parametric regression fits. The Annals of Statistics 21, pp. 1926–1947.

Härdle, W., Marron, J.S., 1991. Bootstrap simultaneous error bars for nonparametric regression. The Annals of Statistics 19, pp. 778–796.

Hastie, T., Tibshirani, R., 1990. Generalized Additive Models. London: Chapman and Hall.

Huxley, J.S., 1924. Constant differential growth-ratios and their significance. Nature 114, 895–896.

Ismen, A., Yigin, C., Ismen, P., 2007. Age, growth, reproductive biology and feed of the common guitarfish (*Rhinobatos rhinobatos* Linnaeus, 1758) in İskenderun Bay, the eastern Mediterranean Sea. Fisheries Research 84, 263–269.

Kauermann, G., Opsomer, J., 2003. Local Likelihood Estimation in Generalized Additive Models. Scandinavian Journal of Statistics 30, 317–337.

Lauzier, R.B., 1999. A review of the biology and fisheries of the goose barnacle (*Pollicipes polymerus* Sowerby, 1833). Fisheries and Oceans Canada, Canadian Stock Assessment Secretariat Research Document 99/111, 30.

Mammen, E., 1992. When Does Bootstrap Work?: Asymptotic Results & Simulations. Springer-Verlag, New York.

Marron, J.S., 1992. Bootstrap bandwidth selection, in: LePage, R., Billard, L. (Eds.), Exploring the Limits of Bootstrap. Wiley-Interscience, pp. 249–262.

Molares, J., 1993. Estudio del ciclo biológico del percebe (*Pollicipes cornucopia* Leach) de las costas de Galicia. Alimentaria 248, 9–69.

Molares, J., Freire, J., 2003. Development and perspectives for community-based management of the goose barnacle (*Pollicipes pollicipes*) fisheries in Galicia (NW Spain). Fisheries Research 65, 485–492.

Montero-Torreiro, M.F., Martínez, P.G., 2003. Seasonal changes in the biochemical composition of body components of the sea urchin, *Paracentrotus lividus*, in Lorbe (Galicia, north-western Spain). Journal of the Marine Biological Association of the United Kingdom 83, 575–581.

Neves, A., Cabral, H., Sequeira, V., Figueiredo, I., Moura, T., Gordo, L.S., 2009. Distribution patterns and reproduction of the cuttlefish, *Sepia officinalis* in the Sado estuary (Portugal). Journal of the Marine Biological Association of the UK 89, 579–584.

Newman, W.A., 1987. Evolution of cirripedes and their major groups, in: Southward, A.J. (Ed.), Barnacle biology (Crustacean Issues 5). A. A. Balkema, Rotterdam, pp. 3–44.

Nieto-Navarro, J.T., Zetina-Rejon, M., Arreguin-Sanchez, F., Arcos-Huitron, N.E., Pena-Messina, E., 2010. Length-Weight Relationship of Demersal Fish from the Eastern Coast of the Mouth of the Gulf of California. Journal of Fisheries and Aquatic Science 5, 494–502.

Oniye, S., Adebote, D., Usman, S., Makpo, J., 2006. Some aspects of the biology of *Protopterus annectens* (Owen) in Jachi dam near Katsina, Katsina state, Nigeria. Journal of Fisheries and Aquatic Science 1, 136–141.

Pinheiro, M.A.A., Fiscarelli, A.G., 2009. Length-weight relationship and condition factor of the mangrove crab *Ucides cordatus* (Linnaeus, 1763) (Crustacea, Brachyura, Ucididae). Brazilian Archives of Biology and Technology 52, 397–406.

Pinilla, F., 1996. Variación temporal de la densidad y biomasa de la población de percebes *Pollicipes elegans* de Lobitos, Piura, Perú. Ph.D. thesis. Tesis Ingeniero Pesquero. UNALM, Lima-Perú.

R Development Core Team, 2009. R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing. Vienna, Austria. ISBN 3-900051-07-0.

Ramírez, P., la Cruz, J.D., Castañeda, J., Galán, J., 2008. Prospección de los bancos naturales de pulpo *Octopus mimus* y evaluación de percebes *Pollicipes elegans* en las islas Lobos de Afuera, Lambayeque (11-16 Julio 2008). Informe Técnico Instituto del Mar del Perú , 20.

Ramón, M., Lleonart, J., Massutí, E., 2010. Royal cucumber (*Stichopus regalis*) in the northwestern Mediterranean: Distribution pattern and fishery. Fisheries Research 105, 21–27.

Ruppert, D., Sheather, S.J., Wand, M.P., 1995. An effective bandwidth selector for local least squares regression. Journal of the American Statistical Association 90, pp. 1257–1270.

Ruppert, D., Wand, M.P., 1994. Multivariate locally weighted least squares regression. The Annals of Statistics 22, 1346–1370.

Sparre, P., Venema, S., 1997. Introduction to tropical fish stock assessment. Part 1. Manual. FAO Fisheries Technical Paper Rev. 2, 420 pp.

Stone, C.J., 1977. Consistent nonparametric regression. The Annals of Statistics 5, pp. 595–620.

Wand, M.P., Jones, M.C., 1995. Kernel Smoothing. Chapman & Hall: London.

Anexo
Package 'NPRegfast'

Type Package

Title Nonparametric estimation for analyzing interactions factor-by-curve

Version 1.0

Date 2011-11-13

Author Marta Sestelo, Nora M. Villanueva and Javier Roca-Pardiñas

Maintainer Marta Sestelo <sestelo@uvigo.es>

Description This package allows the user to obtain nonparametric estimates using local linear kernel smoothers and compare them between factor's levels. Also a feature of the package is its ability to draw inference about critical points, such as maxima or change points linked to the derivative curves. The inference (confidence intervals and tests) is based on bootstrap. This package allows not only to obtain smooth estimates also based on classical parametric models, as allometric model, one of the most used models in biology frameworks usually used to study the relationship between two biometrical variables. Additionally, we have implemented binning type acceleration techniques.

License GPL

LazyLoad yes

R topics documented:

`NPRegfast-package` *Nonparametric estimation by using local linear kernel smoothers*

Description

This package provides a method for obtain nonparametric estimates using local linear kernel smoothers.

Particular features of the package are facilities for fast smoothness estimation, and the calculation of their first and second derivative. User can define the smoothers parameters. Confidences intervals calculation is provided by bootstrap methods. Binning techniques were applied to speed up computation in the estimation and testing processes.

Details

Package:	NPRegfast
Type:	Package
Version:	1.0
Date:	2011-11-13

`NPRegfast` provides functions for nonparametric regression models `frfast`, `plot.frfast`. The term `frfast` is taken to include any nonparametric regression estimated by local lineal kernel smoothers. A number of other functions such `summary.frfast` are also provided, for extracting information from a fitted `frfastObject`.

For a listing of all routines in the NPRegfast package type: `library(help="NPRegfast")`. For an overview of the NPRegfast package see

NPRegfast-package.

Author(s)

Marta Sestelo, Nora M. Villanueva and Javier Roca-Pardiñas.

Maintainer: Marta Sestelo <sestelo@uvigo.es>

References

Efron, B. (1979). Bootstrap methods: another look at the jackknife. Annals of Statistics, 7:1-26.

Efron, E. and Tibshirani, R. J. (1993). An introduction to the Bootstrap. Chapman and Hall, London.

Wand, M. P. and Jones, M. C. (1995). Kernel Smoothing. Chapman & Hall, London.

Examples

```
## See examples for frfast
```

`frfast` *Fitting nonparametric models*

Description

`frfast` is used to fit nonparametric models by using local linear kernel smoothers.

Usage

```
frfast(x, y, f = NULL, model = 1, h = NULL, w = NULL, p = 2,
       kbin = 100, nc = NULL, ncmax = 5, ikernel = 1, iopt = 1,
       nboot = 500, c2 = NULL, rankl = NULL, ranku = NULL)
```

Arguments

`x`	vector of `x` data. Missing values are not allowed.
`y`	vector of `y` data. Missing values are not accepted.
`f`	vector of `factor` data to take into account in the model.
`model`	the nonparametric regression fitting by local linear kernel smoothers (`model = 1`). `model = 2` is used to fit an allometric model.

`h` the kernel bandwidth smoothing parameter. Large values of bandwidth make smoother estimates, smaller values of bandwidth make less smooth estimates. The default is a bandwidth compute by cross validation.

`w` weights on the data.

`p` degree of a polynomial.

`kbin` number of binning nodes over which the function is to be estimated.

`ikernel` numeric which determines the smoothing kernel. By default `ikernel = 1`, this is, the Gaussian density function.

`nboot` number of bootstrap repeats.

`rankle` number or vector specifying the minimum value for an interval at which to search the `x` value which maximizes the estimate, first or second derivative (for each level). The default is the minimum data value.

`ranku` number or vector specifying the maximum value for an interval at which to search the `x` value which maximizes the estimate, first or second derivative (for each level). The default is the maximum data value.

Value

An object is returned with the following elements:

`x` vector of values of the grid points at which model is to be estimate.

`p` matrix of values of the grid points at which to compute the estimate, their first and second derivative.

`pl` lower values of 95 % confidence interval for the estimate, their first and second derivative.

`pu` upper values of 95 % confidence interval for the estimate, their first and second derivative.

`diff` differences between the estimation values of a couple of levels (i. e. level 2 - level 1). The same procedure for their first and second derivative.

`diffl` lower values of 95 % confidence interval for the differences between the estimation values of a couple of levels. It is performed for their first and second derivative.

`diffu` upper values of 95 % confidence interval for the differences between the estimation values of a couple of levels. It is performed for their first and second derivative.

`boot` number of bootstrap repeats.

`n`	total number of data
`dp`	degree of a polynomial.
`h`	the kernel bandwidth smoothing parameter.
`grid`	the number of equally espaced points at which to estimate the curves.
`mod`	factor's level for each data.
`xdata`	original x values
`data`	original y values
`w`	weights on the data.
`fact`	factor's level.
`nf`	number of levels.
`kbin`	number of binning nodes over which the function is to be estimated.
`ikernel`	character which determines the smoothing kernel. By default `ikernel = 1`, this is, the Gaussian density function.
`max`	value of covariate `x` which maximizes the estimate, first or second derivative.
`maxl`	lower value of 95 % confidence interval for the value `max`.
`maxi`	upper value of 95 % confidence interval for the value `max`.
`maxboot`	values of the covariate `x` which maximizes the estimate, first or second derivative for each bootstrap repeats.
`diffmax`	differences between the estimation of `max` for a couple of levels (i. e. level 2 - level 1). The same procedure for their first and second derivative.
`diffmaxl`	lower value of 95 % confidence interval for the value `diffmax`.
`diffmaxu`	upper value of 95 % confidence interval for the value `diffmax`.
`repboot`	matrix of values of the grid points at which to compute the estimate, their first and second derivative for each bootstrap repeat.
`ranku`	minimum value for an interval at which to search the `x` value which maximizes the estimate, first or second derivative (for each level). The default is the minimum data value.
`rankl`	maximum value for an interval at which to search the `x` value which maximizes the estimate, first or second derivative (for each level). The default is the maximum data value.

Author(s)

Marta Sestelo, Nora M. Villanueva and Javier Roca-Pardiñas.

Examples

```
library(NPRegfast)
data(barnacle)


################################################
# Nonparametric regression without interactions
################################################
fit<-frfast(x=barnacle$RC,y=barnacle$PS)
fit

summary(fit)

# Change the number of binning nodes and bootstrap replicates
fit<-frfast(x=barnacle$RC,y=barnacle$PS,kbin=200,nboot=1000)


############################################
# Nonparametric regression with interactions
############################################
fit2<-frfast(x=barnacle$RC,y=barnacle$PS,f=barnacle$F)
fit2

summary(fit2)
```

`summary.frfast` *Summarizing fits of frfast class*

Description

Takes a fitted `frfast` object produced by `frfast()` and produces various useful summaries from it.

Usage

```
summary.frfast(model)
```

Arguments

`model`	a fitted `frfast` object as produced by `frfast()`.

Details

`print.frfast` tries to be smart about `summary.frfast`.

Value

`summary.frfast` computes and returns a list of summary information for a fitted `frfast` object.

`model`	type of estimate.
`h`	the kernel bandwidth smoothing parameter.
`dp`	degree of a polynomial.
`nboot`	number of bootstrap repeats.
`grid`	number of binning nodes over which the function is to be estimated.
`n`	total number of data.
`fmod`	factor's levels.

Author(s)

Marta Sestelo, Nora M. Villanueva and Javier Roca-Pardiñas.

Examples

```
library(NPRegfast)
data(barnacle)

################################################
# Nonparametric regression without interactions
################################################
fit<-frfast(x=barnacle$RC,y=barnacle$PS)
summary(fit)

############################################
# Nonparametric regression with interactions
############################################
fit2<-frfast(x=barnacle$RC,y=barnacle$PS,f=barnacle$F)
summary(fit2)
```

`plot.frfast` *Visualization of frfast objects*

Description

Useful for drawing the estimation, first and second derivative (for each factor).

Usage

```
plot.frfast(model, fac = NULL, der = NULL, xlab = "x",
 ylab = "y", col = "black", ICcol = "grey", main = "title",
 type = "l", ICtype = "l", ...)
```

Arguments

`model`	`frfast` object.
`fac`	number or vector which determines the level to take into account in the plot. By default is `NULL`.
`der`	number or vector which determines any inference process. By default `der` is `NULL`. If this term is `0`, the plot show the initial estimate. If it is `1` or `2`, it is designed for the first or second derivative, respectively.
`xlab`	a title for the x axis.
`ylab`	a title for the y axis.
`col`	a specification for the default plotting color.
`ICcol`	a specification for the default confidence intervals plotting color.
`main`	an overall title for the plot.
`type`	what type of plot should be drawn. Possible types are, `p` for points, `l` for lines, `o` for overplotted, etc. See details in ?`par`.
`ICtype`	what type of plot should be drawn for confidence intervals. Possible types are, `p` for points, `l` for lines, `o` for overplotted. See details in ?`par`.
`...`	other options.

Value

simply produce a plot.

Author(s)

Marta Sestelo, Nora M. Villanueva and Javier Roca-Pardiñas.

Examples

```
library(NPRegfast)
data(barnacle)

##############################################
# Nonparametric regression without interactions
##############################################
fit<-frfast(x=barnacle$RC,y=barnacle$PS)
plot.frfast(fit,der=c(0,1))
plot.frfast(fit,der=c(0,1),col="red",ICcol="blue")


##############################################
# Nonparametric regression with interactions
##############################################
fit2<-frfast(x=barnacle$RC,y=barnacle$PS,f=barnacle$F)
plot.frfast(fit2)
plot.frfast(fit2,der=1,fac=2)
plot.frfast(fit2,der=2,col="red",ICcol="green")
```

`maxp` *Maximum points for the estimate, first and second derivative, with their 95 % confidence intervals*

Description

Value of covariate `x` which maximizes the estimate, first and second derivative, for each level of the factor.

Usage

```
maxp(model, der = NULL)
```

Arguments

`model`	parametric or nonparametric regression out obtained by `frfast` function.
`der`	number which determines any inference process. By default `der` is `NULL`. If this term is `0`, the calculate of the maximum point is for the estimate. If it is `1` or `2`, it is designed for the first or second derivative, respectively.

Value

An object is returned with the following elements:

`Estimation`	outputs for the estimation where it is included maximum points, and their 95 % confidence intervals (for each level).
`Firs_der`	outputs for first derivative with maximum points and their 95 % confidence intervals (for each level).
`Second_der`	outputs for second derivative. It means, maximum points and 95 % confidence intervals (for each level).

Author(s)

Marta Sestelo, Nora M. Villanueva and Javier Roca-Pardiñas.

Examples

```
library(NPRegfast)
data(barnacle)

################################################
# Nonparametric regression without interactions
################################################
fit<-frfast(x=barnacle$RC,y=barnacle$PS)
maxp(fit)
maxp(fit,der=0)
maxp(fit,der=1)
maxp(fit,der=2)


################################################
# Nonparametric regression with interactions
################################################
fit2<-frfast(x=barnacle$RC,y=barnacle$PS,f=barnacle$F)
maxp(fit2)
maxp(fit2,der=0)
maxp(fit2,der=1)
maxp(fit2,der=2)
```

`plot.diff` *Visualization of the differences between the estimation of curves for two factor's levels*

Description

Useful for drawing the differences between the estimation of curves (initial estimate, first or second derivative) for two factor's levels. Missing values of factor's levels is not allowed.

Usage

```
plot.diff(model, factor1, factor2, der = NULL, xlab = "x",
ylab = "y", col = "black", ICcol = "grey", type = "l",
ICtype = "l", ...)
```

Arguments

`model`	allometric or nonparametric regression model obtained by `frfast` function.
`factor1`	first factor's level at which to perform the differences between curves. Missing values are not allowed.
`factor2`	second factor's level at which to perform the differences between curves. Missing values are not allowed.
`der`	number or vector which determines any inference process. By default `der` is `NULL`. If this term is `0`, the calculate of the maximum point is for the estimate. If it is `1` or `2`, it is designed for the first or second derivative, respectively.
`xlab`	a title for the x axis.
`ylab`	a title for the y axis.
`col`	a specification for the default plotting color.
`ICcol`	a specification for the default confidence intervals plotting color.
`main`	an overall title for the plot.
`type`	what type of plot should be drawn. Possible types are, `p` for points, `l` for lines, `o` for overplotted, etc. See details in ?`par`.
`ICtype`	what type of plot should be drawn for confidence intervals. Possible types are, `p` for points, `l` for lines, `o` for overplotted. See details in ?`par`.
`...`	other options.

Details

simply produce a plot.

Author(s)

Marta Sestelo, Nora M. Villanueva and Javier Roca-Pardiñas.

Examples

```
library(NPRegfast)
data(barnacle)

fit2<-diff(x=barnacle$RC,y=barnacle$PS,f=barnacle$F)

plot.diff(fit2)
plot.diff(fit2,factor1=2,factor2=1,der=1)
plot.diff(fit2,factor1=2,factor=1,der=2,col="red",ICcol="green")
```

`maxp.diff` *Differences between the estimation of maximum points for two factor's levels*

Description

Differences between the estimation of `max` for two factor's levels. `max`, a returned element of class `frfast`, is the value of covariate `x` which maximizes the estimate, first or second derivative.

Usage

```
maxp.diff(model, factor2 = NULL, factor1 = NULL, der = NULL)
```

Arguments

`model`	parametric or nonparametric regression model obtained by `frfast` function.
`factor1`	first factor's level at which to perform the differences between maximum points.
`factor2`	second factor's level at which to perform the differences between maximum points.
`der`	number which determines any inference process. By default `der` is `NULL`. If this term is `0`, the calculate of the differences for maximum point is for the estimate. If it is `1` or `2`, it is designed for the first or second derivative, respectively.

Details

Differences are calculated by subtracting a factor relative to another (`factor2 - factor1`). By default `factor2` and `factor1` are `NULL`, so the differences calculated are for all possible combinations between two factors.

Value

An object is returned with the following element:

`maxp.diff` a table with a couple of factor's level where it is used to calculate the differences between maximum points, and their 95 % interval confidence (for the estimation, first and second derivative).

Author(s)

Marta Sestelo, Nora M. Villanueva and Javier Roca-Pardiñas.

Examples

```
library(NPRegfast)
data(barnacle)

fit2<-frfast(x=barnacle$RC,y=barnacle$PS,f=barnacle$F)
maxp.diff(fit2,factor1=2,factor1=1)
maxp.diff(fit2,factor1=2,factor1=1,der=0)
maxp.diff(fit2,factor1=2,factor1=1,der=1)
maxp.diff(fit2,factor1=2,factor1=1,der=2)
```

`contrast` *Bootstrap based test for testing a parametric allometric model*

Description

`contrast` is used to test the null hypothesis of an allometric model $H_0 = m(X) = aX^b$ *vs.* general hypothesis H_1 being m an unknown nonparametric function. To implement this test we have used the wild bootstrap.

Usage

```
contrast(x, y, f = NULL)
```

Arguments

`x`	vector of `x` data. Missing values are not allowed.
`y`	vector of `y` data. Missing values are not accepted.
`f`	vector of `factor` data to take into account in the model.

Value

An object is returned with the following elements:

`value`	the p-value of the test (for each factor).
`statistic`	the value of the statistic obtained by using likelihood ratio test.

Author(s)

Marta Sestelo, Nora M. Villanueva and Javier Roca-Pardiñas.

Examples

```
library(NPRegfast)
data(barnacle)

contrast(x=barnacle$RC,y=contrast$PS)
contrast(x=barnacle$RC,y=barnacle$PS,f=barnacle$F)
```