UNIVERSIDAD NACIONAL AUTÓNOMA
DE MÉXICO

FACULTAD DE CIENCIAS

REDUCCIÓN DE DIMENSIÓN EN REGRESIÓN

T E S I S

QUE PARA OBTENER EL TÍTULO DE:

MATEMÁTICO

P R E S E N T A :

JUAN ANTONIO CORNEJO NIETO

TUTOR(A)
JORGE FRANCISCO DE LA VEGA GÓNGORA

2007

UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

Esta hoja certifica que he examinado esta copia del trabajo de tesis de

JUAN ANTONIO CORNEJO NIETO

y he encontrado que está completa y es satisfactoria en todos los sentidos y que todas y cada una de las revisiones necesarias para la presentación de su examen profesional han sido completadas.

Jorge de la Vega Góngora

Firma del Tutor de tesis

Hugo Villaseñor Hernández

Firma del Co-Tutor

Fecha

Facultad de Ciencias

REDUCCIÓN DE DIMENSIÓN EN REGRESIÓN

TESIS

QUE PARA OBTENER EL GRADO ACADÉMICO DE MATEMÁTICO PRESENTA

JUAN ANTONIO CORNEJO NIETO

COMO PARTE DE LOS REQUISITOS ACADÉMICOS DE LA FACULTAD DE CIENCIAS

Jorge de la Vega Góngora, Asesor

Diciembre, 2007

Hoja de Datos del Jurado

1.Datos del alumno Apellido paterno Apellido materno Nombre(s) Teléfono Universidad Nacional Autónoma de México Facultad de Ciencias Carrera Número de cuenta	1. Datos del alumno Cornejo Nieto Juan Antonio 52 68 85 34 Universidad Nacional Autónoma de México Facultad de Ciencias Matemáticas 083063798
2. Datos del tutor Grado Nombre(s) Apellido paterno Apellido materno	2. Datos del tutor M. en C. Jorge Francisco De la Vega Gongora
3. Datos del co-tutor Grado Nombre(s) Apellido paterno Apellido materno	3. Datos del co-tutor M. en C. Hugo Villaseñor Hernández
4. Datos del sinodal 1 Grado Nombre(s) Apellido paterno Apellido materno	4. Datos del sinodal 1 Mat. Margarita Elvira Chávez Cano
5. Datos del sinodal 2 Grado Nombre(s) Apellido paterno Apellido materno	5. Datos del sinodal 2 M. en C. Inocencio Rafael Madrid Ríos
6. Datos del sinodal 3 Grado Nombre(s) Apellido paterno Apellido materno	6. Datos del sinodal 3 M. en A. Alberto Manuel Padilla Terán
7.Datos del trabajo escrito. Título Subtitulo Número de páginas Año	7. Datos del trabajo escrito Reducción de Dimensión en Regresión 143 p 2007

Agradecimientos

A Jorge
Por ser una persona comprometida, trabajadora, que me ha mostrado y enseñado a aterrizar la teoría estadística y mostrado lo potente que puede ser; por ser un fraterno instructor, a quien gracias a su esfuerzo y motivación, fue un gran aliciente para elaborar y concluir este trabajo, además por ser un ejemplo a seguir (¡cuando yo tenga su edad, quiero ser como él!).

A Hugo
Por su apoyo incondicional, sencillez, y por demostrar que se puede pelear con la espada de la razón.

A Margarita Chávez y Rafael Madrid por dedicar su vida a modelar un mundo mejor compartiendo un poco de su conocimiento y por su disponibilidad para la revisión de este documento, y por sus insustituibles consejos.

A Willis (Alberto) por la motivación y por mostrarme con la práctica que la estadística es una grandiosa herramienta.

A Nicolay por su insistente manera de motivar, por su desinteresado apoyo y por influir genuinamente en la conclusión de este documento; Por compartir grandiosos momentos y por ser más que un maestro.

A Richard por enseñarme que las matemáticas es una forma de vida, por tu apoyo y motivación durante toda la carrera y por extenderme la mano en los momentos difíciles.

A Juan Pablo, Yuriko, Elías, Adolfo, Jorge A., Goyo y Daniel por ayudarme a crecer profesionalmente y por su motivación para con la vida.

A los "Físicos" (Pedro, Alex, Pablo, Malaco, Jose Antonio, y Ricardo) por su aportación en lo que soy.

A los “Matemáticos” (Rafael, Gil, Luis, Paty, Jesús, Galaviz y Enrique) por demostrarme que en la vida se puede llegar tan alto como se quiera y ajustar el rumbo cuando se requiera.

A los “Actuarios” (Luis Mario, Memo, Goyo, Rosario, Paulo y MariCarmen) por mostrarme el otro punto de vista.

A todos los profesores que tuvieron que ver con mi instrucción y que me obsequiaron otra forma de pensar, sobre todo a aquellos que llevan bien puesta la camiseta.

A la UNAM por ser una base y sustento en la educación del País, particularmente en la mía.

A quien no recuerdo ahora, pero que, como diría Frankenstein, ha contribuido en el monstruo que soy.

El Tiempo… ¿que es el tiempo?

Innumerables veces me topo con él, es como un recordatorio de lo que no se ha hecho, a veces es como un saco de plomo sobre mi espalda que presiona y no me permite respirar, a mi alrededor gran cantidad de cómplices de él, no se olvidan de lo que no he hecho mientras transcurre aquel… el tiempo. Ahora comprendo, …aquello que no se había hecho en su tiempo era porque no era el momento, faltaba haber vivido lo que viví, comprendo que no era el tiempo, sino el sentimiento de poder, me di la oportunidad de convencerme de que lo podía lograr; aunque no fue fácil, en el camino, infinitos obstáculos se interpusieron y nublaron la meta, lo que realmente se necesita es creer en uno mismo. ¿Que es el tiempo? para mi, el tiempo es como el dinero, lo inviertes o lo gastas, yo lo invertí y ahora gane… Confianza en mi, gracias a todos aquellos que me ayudaron a hacerlo posible.

Dedicatoria

A mi esposa Claudia, por crecer junto a mí y mostrarme que los obstáculos son tan grandes como uno los quiera ver, por sus consejos francos y templados que han rectificado y allanado mi camino; y por creer en mí.

A Sashell, Giovan y Demian, por ser más que un impulso para seguir adelante y por continuar enseñándome el significado de la vida.

A mis Padres, por darme mucho más de lo necesario, por las noches de desvelo y preocupación y, definitivamente, por darme la existencia.

A mis Hermanos, Güicho, Marcela, Alfredo y Miguel, por enseñarme a vivir y sobrevivir y por motivarme a seguir.

A mis Abuelos, Modesta, Juan, Silviano y Engracia, por compartir su cariño, dulzura y motivación a vivir.

A mis tíos, Manolo, Aída, Kiko, Nery y Celia C., por la gran influencia sobre mi pensar.

A mis primos, Carlos, Ángel, Lalo, Jambo, Manolo, Luis y Chela; y sobrinos, Jenny, Yosef, Pansai, Emilia, Julia, Sofi, Marco, Paco, Jair, Marco Antonio, Tanys y Sully; por compartir gratos momentos.

A mis otros amigos, Chucho, Javier, Alfredo, Olivia y Leo, por ser parte esencial en mi.

Índice general

Índice de figuras

Capítulo 1

Introducción

En varios libros de texto el análisis de Regresión se define como una metodología estadística para investigar y modelar relaciones entre variables, particularmente relaciones *lineales*. En esta tesis la regresión será considerada en un contexto más general, y se definirá como *el estudio de la distribución condicional* $F(y|x_1, \ldots, x_p)$ de una variable y, llamada *respuesta*, dadas las variables predictivas o *predictores* $x_1, \ldots, x_p$.

En situaciones prácticas y problemas reales es común encontrar situaciones en donde el número de predictores es muy grande, y por lo tanto no es posible aplicar la metodología de estudio sin usar la computadora. Es por esto que no se puede desvincular la teoría de regresión de su implementación computacional. Por otra parte, es muy importante tener herramientas que permitan simplificar un problema, típicamente la simplificación se hace para reducir el número de dimensiones involucradas en el problema. Esto vincula a la Regresión con el problema de reducción de dimensión, que es uno de los problemas principales que se estudian en la Estadística.

En esta tesis se plantean como objetivos los siguientes:

1. Aplicar instrumentos computacionales que permitan crear gráficas que faciliten el

estudio de un problema de regresión para obtener un modelo parsimonioso [1]. Se sugiere la utilización de dos sistemas computacionales, uno llamado *xlisp* (Tie90) [2] y el otro llamado `R` (IG96) [3]. La ventaja de estos paquetes es que permiten la creación de gráficas dinámicas y facilitan la ayuda de visualizaciones en la búsqueda de mejores modelos. En particular *xlisp*, a través de un conjunto de programas llamado `Arc`, facilita la implementación de muchas de las ideas asociadas con estimación de la dimensión de problemas de regresión.

2. Proveer un documento que sirva tanto de manual de uso de los programas mencionados, especializado en el contexto de regresión, así como de guía metodológica para plantear y resolver problemas específicos en la práctica diaria. Esta guía y manual son necesarios como primer paso para difundir las técnicas a estudiar en el medio académico a nivel licenciatura, ya que actualmente no hay documentos en español que sirvan para la enseñanza de la metodología presentada, en parte porque varias de las técnicas aún están en proceso de investigación en las universidades americanas. Algunas técnicas propuestas no son realmente nuevas, en realidad son combinaciones de métodos clásicos con el uso de la computadora que permite lle-

[1] El *Principio de Parsimonia* o *Navaja de Occam* indica *Entia non sunt multiplicanda sine necesitate*, los entes no deben multiplicarse sin necesidad, no expliques por lo más lo que puedas explicar por lo menos, o en términos matemáticos, el principio de parsimonia pretende conseguir la descripción (o explicación) de los datos con el modelo más simple posible.

[2] *xlisp* es un dialecto de Lisp, un lenguaje de programación que se basa en el procesamiento de listas. *xlisp* cuenta con extensiones para soportar programación orientada a objetos. A lo largo del tiempo se han elaborado muchas versiones de *xlisp* con diferentes metas.

[3] `R` es un programa que ofrece un ambiente para cálculo estadístico y gráficas. `R` provee una gran cantidad de procedimientos y funciones estadísticas, que incluyen modelos lineales generalizados, regresión no lineal, análisis de series de tiempo, pruebas clásicas paramétricas y no paramétricas, herramientas para el análisis multivariado de datos y otras facilidades para manejar bases de datos. También contiene una gran cantidad de funciones que dan un ambiente gráfico flexible para presentar datos de diversas maneras. Además existen paquetes adicionales disponibles para una variedad de aplicaciones y propósitos específicos.

var a cabo estimaciones no paramétricas y aplicar técnicas de suavizamiento para obtener un modelo más adecuado para los datos bajo consideración.

En el capítulo 2 se introduce el uso básico de *xlisp* y de `R` , poniendo particular énfasis en el primero de los programas, pues para el segundo hay una vasta bibliografía disponible, algunas de las referencias se incluirán en la bibliografía. En este capítulo se desarrollarán algunos ejemplos prácticos y cómo se pueden generar programas sencillos.

En el capítulo 3 se desarrollan aspectos del análisis de regresión clásica a través de su solución vía los programas introducidos en el capítulo anterior. Aquí se incluirán aspectos como estimación, predicción, diagnósticos y validación de un modelo. Como esta tesis no pretende ser un libro de texto en regresión, sino una guía metodológica, el análisis se hará a través de la aplicación a conjuntos de datos específicos. En la siguiente sección se presenta un ejemplo de aplicación de regresión en finanzas, que serve para mostrar algunas de las técnicas discutidas.

En el capítulo 4 se hace una revisión de los conceptos generales asociados con reducción de dimensión. Esta parte corresponde a una revisión de la literatura disponible y de cómo han sido implementados en `Arc` y `R`, así como de su aplicación concreta. Por último, en este capítulo se introducen las ideas generales de regresión gráfica y su aplicación a los mismos ejemplos introducidos en capítulos anteriores para terminar con el estudio de los casos.

Para finalizar la tesis, se discuten algunos problemas abiertos del enfoque de regresión presentado y las conclusiones generales de la tesis.

1.1. Aplicación a portafolios de inversión

Un ejemplo que se utiliza en esta sección para mostrar algunos desarrollos se basa en un problema de portafolios de inversión. El problema clásico de portafolio de mínima varianza es un problema clásico en Finanzas (Mar57). Aquí se aplicará un enfoque

estadístico en el que el problema de estimar las proporciones eficientes en que hay que invertir una cierta riqueza K entre n instrumentos de un portafolio de inversión se puede replantear como un problema de estimación de coeficientes en una regresión lineal. Este enfoque es parte de un estudio más amplio que utiliza métodos de regresión dinámica bayesiana desarrollado por Mark Britten-Jones (BJ98).

1.1.1. Modelo

El planteamiento es el siguiente. Sea $\mathbf{r}_t$ el vector de los rendimientos de n instrumentos financieros en el tiempo t. Por definición, los pesos de un portafolio de mínima varianza $\mathbf{w}_g$ minimizan la varianza de los rendimientos de un portafolio global, con la restricción de que tienen que sumar uno:

$$\mathbf{w}_g = \arg \min_{\{\mathbf{w}:\mathbf{1}'\mathbf{w}=1\}} \text{Var}\{\mathbf{r}_t'\mathbf{w}\}$$

Para incorporar la restricción a la función objetivo, se particionan los vectores de pesos y rendimientos separando la primera componente: $\mathbf{r}_t = (r_{t,1}, \mathbf{r}_{t,2})'$ y $\mathbf{w} = (w_1, \boldsymbol{\omega})'$. Entonces $w_1 = 1 - \mathbf{1}'\boldsymbol{\omega}$ y el rendimiento se puede escribir como

$$\mathbf{r}_t'\mathbf{w} = r_{t,1}w_1 + \mathbf{r}_{t,2}'\boldsymbol{\omega} = r_{t,1}(1 - \mathbf{1}'\boldsymbol{\omega}) + \mathbf{r}_{t,2}'\boldsymbol{\omega} = r_{t,1} + \mathbf{r}_t^{e'}\boldsymbol{\omega},$$

donde $\mathbf{r}_t^e = \mathbf{r}_{t,2} - \mathbf{1}r_{t,1}$. De este modo, el problema se puede escribir como

$$\min_{\boldsymbol{\omega}} \text{Var}\{r_{t,1} + \mathbf{r}_t^{e'}\boldsymbol{\omega}\}.$$

Esta función objetivo es equivalente al objetivo de una regresión por mínimos cuadrados ordinarios:

$$-r_{t,1} = -\alpha + \beta'\mathbf{r}_t^e + \varepsilon_t, \qquad \varepsilon_t \sim \mathcal{N}\left(0, \sigma^2\right)$$

para $t = 1, \ldots, T$, donde $\beta = \omega$. El término σ^2 es la varianza de los rendimientos del verdadero portafolio de mínima varianza.

En términos matriciales, considerando las observaciones disponibles en T periodos, el problema se puede escribir como:

$$-\mathbf{r}_1 = -\alpha\mathbf{1} + \mathbf{R}_e\boldsymbol{\omega} + \mathbf{e}, \qquad \mathbf{e} \sim \mathcal{N}\left(\mathbf{0}, \sigma^2\mathbf{I}\right)$$

En el modelo planteado, los predictores corresponden a los excesos de rendimiento de los otros instrumentos con respecto al primero, y la variable de respuesta es el rendimiento del primer instrumento.

Para el modelo de portafolio, la inclusión de una constante puede ser irrelevante, y se podría eliminar si se centran los datos, lo que es equivalente a multiplicar al vector respuesta y a la matriz de predictores por la matriz $\mathbf{M} = \mathbf{I} - n^{-1}\mathbf{1}\mathbf{1}'$. Sin embargo, al multiplicar por esta matriz se introduce correlación en los errores, ya que la covarianza de los errores sería $\sigma^2\mathbf{M}$. Como los coeficientes de la estimación no cambian si se estiman con datos centrados o con datos no centrados con constante, se prefiere estimar el modelo completo e ignorar la constante.

1.1.2. Datos

Los datos que se tienen disponibles son datos de tipos de cambio de 5 divisas con respecto al dólar americano. Las cinco divisas son: yen japonés (jpy), dólar canadiense (cad), franco suizo (chf), libra esterlina (gbp) y euro (eur). Los datos tienen periodicidad diaria de enero 3 de 1994 mayo 21 del 2004, para un total de 2710 observaciones diarias. La fuente de la información es Bloomberg [4].

Como no había euros disponibles antes de 1999, se utiliza al marco alemán como divisa sustituta en esas fechas. A partir de los precios se pueden obtener los rendimientos compuestos para diferentes intervalos de tiempo. Un rendimiento compuesto neto de k periodos se calcula como

$$R_t(k) = \frac{P_t - P_{t-k}}{P_{t-k}}.$$

[4]La dirección de Bloomberg en internet es http://www.bloomberg.com/

Cualquiera de las divisas puede ser tomada como la divisa de referencia $r_{t,1}$, siempre y cuando sea una divisa fuerte o dominante. Para los ejercicios que se consideran más adelante, la divisa de referencia es el Euro, y los excesos de rendimiento se calculan sobre esta divisa.

Las propiedades de la distribución de los rendimientos se pueden estudiar desde el punto de vista del tiempo o bien con respecto a su relación con rendimientos de otros instrumentos. Usualmente, se considera que los rendimientos individuales tienen una estructura dinámica en el tiempo, es decir, se considera la distribución de $\{R_{i1}, \ldots, R_{in}\}$ en términos condicionales; La distribución conjunta de los rendimientos se puede expresar como:

$$F(R_{i1}, \ldots, R_{in}) = F(R_{i1})F(R_{i2}|R_{i1}) \cdots F(R_{in}|R_{i(n-1)}, \ldots, R_{i1}).$$

Esta ecuación sugiere que para el estudio de los rendimientos, las distribuciones condicionales son más relevantes que las distribuciones marginales. Sin embargo, en un sentido práctico las distribuciones marginales pueden ser de interés como aproximación pues son más fáciles de estimar, y en algunos casos, los rendimientos tienen correlaciones seriales empíricas débiles y en estos casos, las distribuciones marginales están cerca de las condicionales. De este modo, es razonable considerar distribuciones marginales en la práctica.

Se han propuesto diferentes modelos para las distribuciones marginales en la literatura, incluyendo la distribución normal, la distribución lognormal, distribuciones estables y mezclas de normales. Una discusión más detallada se da en (Tsa02). Estas distribuciones se discutirán más adelante con el análisis de los datos.

Capítulo 2

Uso básico de *Arc*

Introducción

Este capítulo tiene como objetivo servir como un tutorial en el uso de *Arc* para hacer un análisis preliminar descriptivo de datos. Adicionalmente se introducen comandos básicos en la interacción con *Arc* vía ejemplos de aplicaciones en el contexto del análisis de datos.

En la primera sección, se describen características generales de *Arc*, sus elementos, su configuración básica, lectura, escritura y generación de datos. En esta parte se hace una intersección amplia con funciones que son propias de XLISP-STAT [1] pero que ayudan a interactuar mejor con el programa. En la siguiente sección se describen algunas características de *R*, con menor detalle y más enfocados al análisis de regresión. En la última sección se muestran ejemplos que muestran características esenciales de la aplicación de ambos programas.

[1] Es un sistema basado en el dialecto Xlisp, el cual permite el desarrollo de un ambiente estadístico completo programado en lenguaje lisp, que además está complementado con sistemas de programación orientada a objetos que permiten la creación de gráficos dinámicos para la representación de modelos estadísticos y modelos de regresión lineal y no lineal.

2.1. ¿Qué es *Arc*?

Arc es una herramienta de análisis estadístico para problemas de regresión, que fue escrito por Dennis Cook y Sanford Weisberg de la Universidad de Minnesota. Uno de los atributos más interesantes de *Arc* es que permite un análisis *dinámico* de datos de regresión tanto gráfica como numéricamente. Además, es de los pocos programas que incluyen dentro de su conjunto de métodos de análisis disponibles, aquellos que son más recientes y novedosos. Al estar basado en un lenguaje de programación orientado a objetos, permite implementar rápidamente ideas sin tener que escribir muchos programas y más bien trata de usar métodos ya disponibles o programados por otros usuarios, cumpliendo con el principio de la programación orientada a objetos.

Arc está escrito en *Lisp-Stat* (Tie90), que es un poderoso ambiente computacional para programación estadística [2] implementado usando el lenguaje *Xlisp-Plus*(Bet85; Bet88) el cual es un dialecto de *lisp*. Aunque esto puede ser un poco confuso, para utilizar *Arc* no es necesario conocer ninguno de los programas en los que se basa, sólo en el caso de requerir de hacer extensiones a los modelos implementados y para cálculos más detallados.

Arc puede obtenerse gratuitamente vía internet, en el siguiente portal:

`www.stat.umn.edu/arc/software.html`

Hay versiones disponibles para Unix o Linux, Macintosh y Windows. La utilización es esencialmente la misma en todos los sistemas aunque la descripción que se hará se basará en la versión para Windows.

Una vez que se obtiene el programa y éste es instalado (lo que es muy sencillo siguiendo las instrucciones), el primer contacto que se tiene con el programa es una pantalla como la que se muestra en la figura 2.1. En la ventana aparece información básica

[2] Otro proyecto con aplicaciones estadísticas orientadas al análisis multivariado de datos basado en Lisp-Stat es el programa *Vista*, escrito por Forrest Young. Este programa también es gratuito y puede obtenerse vía el web en `http://forrest.psych.unc.edu/research/vista-frames/index.html`

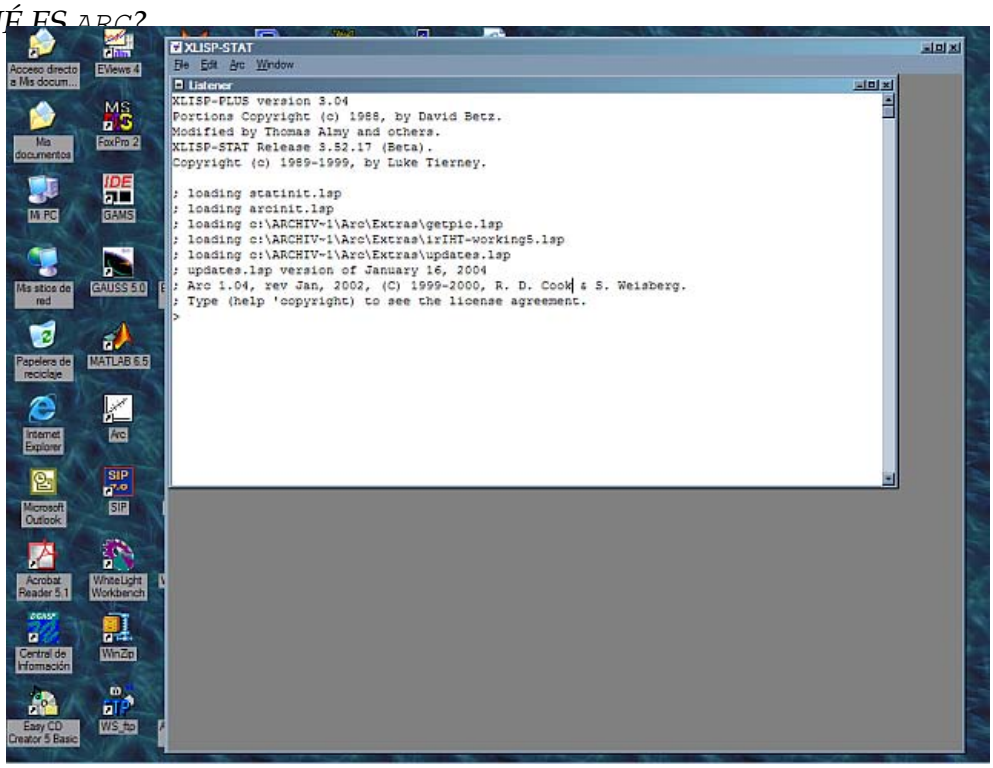


Figura 2.1: Pantalla inicial de *Arc*.

acerca de los autores de los programas y de las licencias. Esta ventana de texto permite la interacción entre el *listener* [3] y el usuario, y se pueden introducir comandos directamente (i.e. usando el programa como un intérprete) o bien usar los menús disponibles.

La ventana de texto contiene una barra con cuatro menús básicos: `File`, `Edit`, *Arc* y `Window`. Cada menú tiene una lista de varios comandos que realizan funciones básicas. Algunas de estas funciones serán descritas en el contexto de la aplicación del sistema a la resolución de problemas. Otras funciones son autodescriptivas, ya que son comunes en el entorno de los sistemas operativos gráficos en general. Por ejemplo, el menú `File` contiene los siguientes comandos:

[3] El *listener* es la ventana en la cual usted escribe los comandos. Cuando éste está listo para recibir un comando despliega ">". En ">" usted puede escribir una expresión. Puede utilizar el ratón o la tecla de retroceso para corregir cualquier equivocación mientras que escribe la expresión. Cuando la expresión es completa y oprime la tecla de intro(o enter) el *listener* pasa la expresión al *evaluator*. El *evaluator* evalúa la expresión y devuelve el resultado al *listener* para imprimir.

Opción	Descripción
Open	Da el acceso a los archivos de datos así como de los de programas
Dribble	Permite escribir a un archivo todos los comandos escritos en el listener.
Print...	Imprime el texto que aparece en el listener
Exit	Salida del programa
about XLISP-STAT	Da información sobre los autores y version de XLISP-STAT .

El menú `Edit` contiene algunos de los comandos mas comunes de cualquier aplicación de Microsoft Windows.

Opción	Atajo	Descripción
Undo	Ctrl-Z	Deshace la última operación realizada en sistema *Arc* .
Cut	Ctrl-X	Corta lo seleccionado.
Copy	Ctrl-C	Copia lo seleccionado.
Paste	Ctrl-V	Pega lo que permanece guardado en el portapapeles.
Clear	Del	Borra lo seleccionado.
Copy-Paste	Alt-V	Copia el texto seleccionado con el cursor al listener.

El menú *Arc* contiene las instrucciones elementales de *Arc* (estadística) como son:

Opción	Descripción
Load	Lee un archivo de datos o programa y lo carga en la memoria
Dribble	Permite escribir a un archivo todos los comandos escritos en el listener.
Calculate probability	Dado un cuantil, calcula la probabilidad en las colas de ese cuantil para algunas distribuciones
Calculate quantile	Dada una probabilidad y una distribución, calcula el cuantil asociado.
about *Arc*	Información general sobre *Arc*.
Help	Información muy general y primitiva sobre comandos en *lisp*.
Settings	Actualización de parámetros y constantes de la configuración original del sistema
Exit	Salida del programa

Por último, el menú `Window` también contiene las opciones elementales para la administración de ventanas que son comunes en Microsoft Windows.

Opción	Atajo	Descripción
Cascade	Shift+F5	Despliega todas las ventanas abiertas una detrás de otra.
Tile,	Shift+F4	Despliega todas las ventanas abiertas para verlas completas.
Arrange Icons,		Acomoda los iconos de las ventanas minimizadas
Close All		Cierra todas las ventanas abiertas.

2.1.1. Configuración básica

Cuando se instala *Arc* se crea un directorio con el nombre Arc que contiene todos los archivos necesarios para ejecutar el programa. En ese directorio hay dos subdirectorios que son relevantes para el usuario, como se describe abajo; un archivo llamado `config.lsp`, que contiene la información de configuración básica, como el directorio de inicio, el tamaño del font a usar y otras características básicas, y un archivo llamado `statinit.lsp` que determina acciones a ser tomadas cuando se inicia *Arc*. Los archivos con la extensión `lsp` pueden ser editados con cualquier editor de texto que utilice ASCII simple.

Los directorios que son relevantes para el usuario son:

i. el directorio "Data" y sus subdirectorios es en donde se encuentran los conjuntos de datos que el programa puede leer directamente sin especificar una ruta de acceso (por omisión). Si se quiere agregar un directorio personal con datos, se debe agregar la siguiente línea en el archivo `statinit.lsp`:

```
(add-data-directory "Aqui el nombre del directorio\\")
```

ii. El directorio "Extras" en el que se leen los programas que se desea que se carguen a la memoria al iniciar el programa. Esto es útil para agregar métodos complementarios a los ya escritos o para hacer pruebas sin modificar los programas disponibles.

Algunos otros parámetros sobre el comportamiento general de *Arc* pueden ser modificados vía **Arc** → **settings** de lo que se obtiene una lista de valores definidos internamente, por ejemplo la variable `*AUTO-SAVE*` sirve para determinar si se quiere guardar un archivo con los mensajes en la ventana de texto al salir en forma automática. Para modificar los valores de las variables, se selecciona la variable que se quiere modificar y se vuelve a seleccionar el menú **settings** → **Update selection**. Aquí aparece un diálogo explicando la variable seleccionada y sus posibles valores. Esto muestra cómo los menús disponibles son dinámicos y cambian dependiendo del contexto.

2.1.2. Manejo de datos

En varias partes de la tesis se pondrá énfasis en cómo obtener información sobre conjunto de datos. En esta sección se definirá cómo introducir los datos al programa y cómo se pueden hacer cálculos básicos, antes de comenzar con el análisis propiamente dicho.

El intérprete

La forma usual de trabajar en Arc es a través de archivos de datos, pero es posible introducir datos vía el listener.

La interacción con el sistema consiste en una conversación entre el usuario y el intérprete (en este caso es *lisp*). La conversación se inicia con el cursor

```
>
```

Si se introduce una expresión, el intérprete responde imprimiendo el resultado de la evaluación de esa expresión. Por ejemplo, introduciendo un número y tecleando Intro, el intérprete responde simplemente imprimiendo de vuelta el número en la siguiente línea y regresando un cursor:

```
>1
1
>
```

Operaciones con números se llevan a cabo combinando números y un símbolo, representando una operación, en expresiones compuestas como las siguientes:

```
>(+ 1 2 3)
6
>(* 2 3.7 8.2)
60.68
>(+ (* 2 3) 4)
10
```

La ventaja de usar esta notación es que se pueden hacer cálculos vectorizados. Por ejemplo, si se requiere restar la media a cada observación de un conjunto de datos se hace a través de la expresión `(- m (list a b c))`.

Una operación complicada como puede ser el cálculo de la cantidad $\frac{-b+\sqrt{b^2-4ac}}{2a}$ se puede introducir como:

```
>(/ (+ (- b) (sqrt (- (^ b 2) (* 4 a c)))) (* 2 a))
```

lo cual puede resultar tedioso. Una forma alternativa y más familiar es usar el siguiente comando, que es similar a como se haría en una calculadora:

```
>(eval (parcil "(-b+sqrt(b^2-4*a*c))/(2*a)"))
```

Creación de variables

lisp es un lenguaje que trabaja procesando listas. Una *lista* es una serie de elementos separados por espacios y contenidos entre paréntesis. Los elementos en una lista pueden ser otras listas o construcciones más complicadas. Usualmente el primer elemento de una lista es una construcción que le dice al programa qué hacer con el resto de la lista. Por ejemplo, la lista

```
>(def TC (list 11.43 11.51 11.38))
```

tiene tres elementos: `def`, `TC` y `(list 11.43 11.51 11.38)`. El primer elemento `def` es la instrucción que le dice al intérprete que se quiere definir algo. El siguiente elemento `TC` es el nombre a ser definido y el último elemento es la definición, que es en sí misma una lista cuyo primer elemento `list` es la instrucción que genera la lista y los números siguientes son los elementos de la lista.

Un ejemplo en donde se redefine una variable sin tener que definir de nuevo su tipo es el siguiente, en donde la variable `TC` se cambia por una lista de 4 números aleatorios uniformes:

```
>(setf TC (uniform-rand 4))
(0.925345 0.172929 0.951559 0.938518)
```

Como un ejemplo del uso directo del intérprete para análisis básico de datos, considérese los datos correspondientes a las primeras 15 observaciones de los logaritmos de los rendimientos diarios de acciones de Alcoa (aa) y de American Express (axp) de enero de 1990. Los datos están expresados como porcentajes.

A continuación se introducen los datos para crear las variables `aa` y `aex` y se obtienen estadísticas básicas como la media, la mediana, la desviación estándar y el rango intercuartil para la primera variable.

```
> (def aa (list -8.806 1.742 5.429 -3.993 -10.649
               -5.381 11.350 1.576 -12.177 2.370
                3.994 -1.137 5.780 -5.231 -3.492))
  AA
> (def aex (list -2.921 -3.827 -3.349 -18.147 5.218
                10.132 8.408 -5.163 4.980 -0.406
               -32.343 1.953 -2.235 4.690 -9.553))
  AEX
> (mean aa)
-1.24167
> (median aa)
-1.137
> (standard-deviation aa)
6.65819
> (interquartile-range aa)
8.488
```

También en este nivel se puede obtener una idea visual de los datos. Por ejemplo, los comandos `(histogram aa)` y `(plot-points aa aex)` genera las gráficas que se muestran en la Figura 2.2. Más adelante se verá cómo se pueden enriquecer estas gráficas

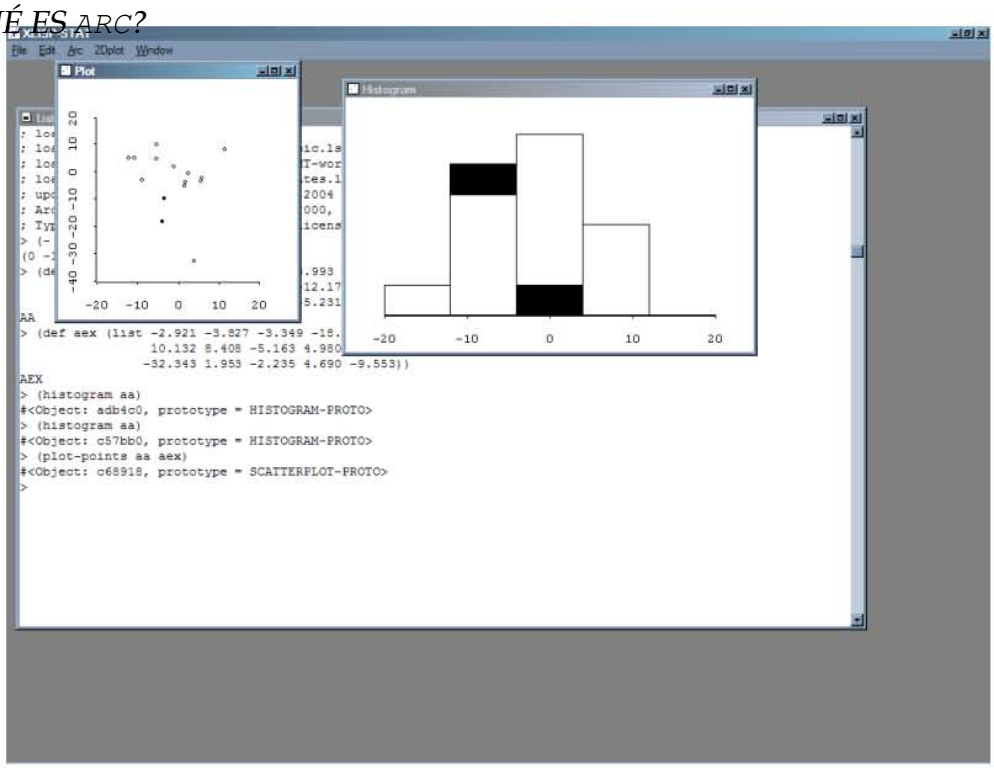


Figura 2.2: Ejemplo de gráficas ligadas.

para obtener mayor información de los datos. Estas gráficas pueden ser *ligadas* para identificar puntos marcados en una gráfica y visualizarlos en otra. Para esto, se selecciona la opción **Link View** que está en cada uno de los menús de gráficas que aparecen cuando se selecciona cada una de las gráficas.

En la tabla A.1 del apéndice se incluye un listado de algunas de las funciones estadísticas y matemáticas disponibles y en la tabla **??** se incluyen algunos comandos asociados con gráficas.

Guardando datos

Para guardar las variables introducidas directamente en el listener y no perder las variables creadas se utiliza el comando `make-dataset` con varios argumentos. Los argumentos de ciertos métodos se caracterizan por iniciar con dos puntos. El comando se

escribe con sus argumentos en diferentes renglones para facilitar su interpretación, pero esto no es necesario:

```
>(make-dataset
       :data (list aa aex)
       :data-names (list "Alcoa" "American Express")
       :name "Acciones")
```

El argumento `:data` define qué variables serán agregadas al conjunto de datos. El argumento `:data-names` define los nombres de las variables y el argumento `:name` define el nombre del conjunto de datos creado. Una vez que se teclea `Intro`, en la barra de menús aparecen dos nuevos elementos: uno que tiene el mismo nombre que se le dio al conjunto de datos, y otro con el nombre de `Graph&Fit`. Estos dos nuevos menús contienen varios elementos que permitirán hacer un análisis gráfico y numérico mucho más exhaustivo que el que se puede obtener introduciendo los comandos directamente. En los capítulos siguientes se desarrollarán muchas de las ideas asociadas a estos elementos.

En el menú con el nombre del conjunto de datos, hay un elemento `Save data set as...` que permite guardar el archivo con las variables. El archivo creado por *Arc* tiene extensión `.lsp` por omisión, pero el formato del archivo es de texto simple, que puede ser abierto con cualquier editor de texto como el Bloc de notas de Ms Windows. Al abrir el archivo con un editor de texto, se puede observar que la estructura del archivo es la siguiente:

```
dataset = Acciones
begin variables
Col 0 = Alcoa
Col 1 = American Express
end variables
begin data
(
 -8.806 -2.921
  1.742 -3.827
  5.429 -3.349
```

```
 -3.993 -18.147
-10.649   5.218
 -5.381  10.132
  11.35   8.408
  1.576  -5.163
-12.177   4.98
  2.37   -0.406
  3.994 -32.343
 -1.137   1.953
  5.78   -2.235
 -5.231   4.69
 -3.492 -9.553
)
```

Como se mencionó antes, introducir datos de este modo puede ser útil sólo para conjuntos pequeños de información. Para mayor cantidad de datos y variables, es necesario saber cómo introducir los datos vía archivos.

Leyendo datos

Un comentario importante antes de continuar: El listener no reconoce carácteres con acento. En lo que sigue se mostrarán los comentarios en donde algunas palabras no estarán acentuadas. Esta es la estructura general que tendrán los archivos de datos que se han guardado con *Arc*:

```
;las lineas con comentarios comienzan con punto y coma.
dataset = Nombre  ;Aqui se pone el nombre del conjunto de datos

begin description
   En esta seccion se describe la informacion numerica, numero de
   observaciones y las unidades en las que esta la informacion.
   ................................................
end description

missing = ?  ;Establece como se representan los casos perdidos
delete-missing = nil ; si se usa t, los casos con datos perdidos se eliminan
```

```
begin variables
  Col 0 = Nombre del campo o variable = Descripci\'{o}n breve de la variable
  Col 1 = Campo = Descripci\'{o}n del campo
  ..................................................
  Col k = Campo final = Descripcion del campo final
end variables

begin lisp
En esta secci\'{o}n se incluyen comandos en lisp que quiera ser procesada
end lisp

begin transformation
;en esta seccion se definen transformaciones de los datos originales, i.e.
Z = X1^2
W = (sin X1)
end transformation


begin data
 (
  d10 d11 d12 ... d1k
  d20 d21 d22 ... d2k
   ..................
  dn0 dn1 dn2 ... dnk )
  ; observacion: no es necesario utilizar "end data"
```

Aquí es importante hacer notar que la numeración en *Arc* comienza en 0.

Para leer un archivo se puede proceder por estos dos caminos:

1. Se puede crear un archivo con la estructura anterior utilizando un editor de texto y en *Arc* se utiliza la opción **File** → **Load** o se puede introducir directamente el comando `(load nombre-archivo)`. Cuando se carga un archivo de datos en *Arc* automáticamente se despliega en el listener el texto que está en la descripción de la información que contiene el archivo.

2. Si se tiene un archivo con los datos en columnas separados por espacios y con la primera línea del archivo indicando nombre de las variables, por ejemplo:

```
 "Y" "X1" "X2" "X3" "X4"
-1.1527809  0.974032697 -1.072108    0.28351    -0.740208
-1.0439365 -0.70178594   1.5885570   0.77945    -0.350112
-0.3468091 -0.68872108   2.88687300 -0.69858515 -0.069753
...
```

Se utiliza el comando `(load nombre-archivo)` directamente. En este caso se abrirá un diálogo para completar la información faltante. Si el archivo sólo tiene las columnas de variables sin los nombres, esta información también será solicitada mediante diálogos.

Usando el menú de datos

Una vez que se carga un archivo con datos, el menú de datos (data set menu), es el que permite manipular los datos y obtener estadísticas y gráficas descriptivas. Normalmente éste lleva el nombre del conjunto de datos; el segundo menú que aparece siempre se denota por `Graph&Fit` y contiene opciones las cuales permiten graficar y ajustar ciertos modelos; estas opciones serán discutidas en la sección 2.1.3.

En esta sección se describirán algunos de los elementos que conforman el menú de datos. Aquí se utilizarán los datos de los tipos de cambio que se encuentran en el archivo `tipo_cambio.lsp`

Description Despliega información acerca del conjunto de datos, si está disponible en el archivo de datos.

```
Arc 1.04, rev Jan, 2002, Fri Jun 4, 2004, 16:49:42.  Data set name: tipos_cambio
Datos de tipos de cambio diarios del dolar con respecto a 5 divisas
(dolares por la unidad monetaria respectiva)
Los datos est\'{a}n disponibles de enero 3 de 1994 al 21 de mayo de 2004
Los datos que se refieren a euros antes de 1999 son realmente marcos alemanes
Fuente: Bloomberg
Name Type    n    Info
```

```
anio Variate 2710 anio de la operacion
cad  Variate 2710 tipo de cambio d\'{o}lar canadiense / d\'{o}lar americano
chf  Variate 2710 tipo de cambio franco suizo / d\'{o}lar americano
dia  Variate 2710 dia del mes
eur  Variate 2710 tipo de cambio euro / d\'{o}lar americano
gbp  Variate 2710 tipo de cambio libra esterlina / d\'{o}lar americano
jpy  Variate 2710 tipo de cambio yen japon\'{e}s / d\'{o}lar americano
mes  Variate 2710 mes
```

Display summaries Despliega información estadística básica en la ventana de texto de las variables que se especifiquen. Por ejemplo, seleccionando las variables *cad* y *chf*, se obtiene información sumaria como número de observaciones, media, mediana, desviación estándar, valores extremos (mínimos y máximos) y correlaciones para los tipos de cambio:

```
Data set = tipos_cambio, Summary Statistics
Variable    N Average     Std Dev     Minimum      Median     Maximum
cad      2710  0.6959      0.039885    0.62027     0.69974     0.7866
chf      2710  0.70469     0.085135    0.54921     0.6918      0.89582

Data set = tipos_cambio, Sample Correlations
cad  1.0000  0.7112
chf  0.7112  1.0000
       cad     chf
```

Table data Permite obtener tablas de estadísticas de ciertas variables, condicionales a otras variables. Por ejemplo, se puede obtener una tabla con los tipos de cambio promedios anuales y las desviaciones estándares anuales del dólar canadiense y el franco suizo, además del número de observaciones. Al seleccionar esta opción aparece un ventana con tres rectángulos uno de las variables candidatas, el siguiente para colocar las variables de interés y el último para ubicar las variables condicionantes; en nuestro caso las variables de interés son *cad* y *chf* y la condicionada es *anio*. Además se seleccionan las opciones de `Cell count`, `Mean`, `SD` y `Display as list`, con lo que se obtiene:

```
Data set = tipos_cambio, Table of included cases
Col. 1 = anio
Col. 2 = Count
Col. 3 = cad[Mean]
Col. 4 = cad[SD]
Col. 5 = chf[Mean]
Col. 6 = chf[SD]
----------
1994 260  0.732195      0.0117022     0.733934      0.0378439
1995 260  0.728981      0.0123798     0.847912      0.0343155
1996 262  0.733445      0.00589425    0.810066      0.0276003
1997 261  0.722389      0.0113847      0.68983      0.0173521
1998 261  0.674755      0.022454      0.691206      0.0301023
1999 261  0.673237      0.00913278     0.66595      0.0252784
2000 260  0.673401      0.0127076     0.592601      0.0213637
2001 261  0.645926      0.0113954     0.593106      0.0194049
2002 261  0.637046      0.00948676    0.644913      0.0376489
2003 261  0.716117      0.038116      0.744451      0.0223574
2004 102  0.751014      0.0165945     0.788425      0.0164052
```

Se pueden crear tablas hasta con 7 variables condicionantes, y no hay restricción sobre el número de variables a condicionar. Hay varias alternativas en el diálogo que se obtiene.

Display data Lista los valores de todas las variables especificadas en el diálogo.

Display Case Names Muestra los nombres de cada caso en una ventana. Esta ventana puede ligarse a gráficas para seleccionar casos

Los siguientes elementos del menú de datos son para modificar los datos existentes.

Add a Variate Agrega nuevas variables al conjunto de datos. Por ejemplo, con la expresión `rgbp=1/gbp`, entonces una variable se agrega al conjunto de datos y su valor será el recíproco del tipo de cambio de la libra esterlina. También es posible agregar una expresión en lisp en el lado derecho de la ecuación.

Delete Variable Borra una variable existente en el conjunto de datos.

Rename Variable Cambia el nombre de una variable existente.

Transform Crea nuevas variables usando logaritmos y potencias.

Make Factors Crea factores de variables categóricas. Un factor aquí se entiende como una colección de variables indicadoras o "dummies".

Make Interactions Crea interacciones de factores y variables.

Set Case names Establece la variable que se utilizará para etiquetar cada caso observado.

Creando funciones

Una parte muy importante de XLISP-STAT es su capacidad para extender el conjunto de funciones que están definidas originalmente. *Arc* mismo es un conjunto de funciones que hacen uso extensivo de las funciones básicas de XLISP-STAT para hacer cálculos relativos a regresión.

Como ejemplo de la creación de funciones, en esta sección se definirá una función que permite obtener los rendimientos mensuales de un vector de precios, utilizando los datos del tipo de cambio del dólar canadiense `cad`. Más adelante se verá cómo se puede aplicar esta función a una matriz de datos en forma conveniente.

Para facilitar la exposición, se considerarán valores particulares al conjunto de datos que se tienen; para hacer la función más general se requiere escribir código adicional, por ejemplo, para determinar si se cuenta con información completa por cada año.

La siguiente función, llamada `rend-mensual` con argumento `divisa` realiza un cálculo simple de los rendimientos, a continuación se explican los detalles.

```
(defun rend-mensual (divisa)
  (let* ((rends ()))
      (dolist  (i (iseq 1994 2004))
          (dolist (j (if (/= i 2004) (iseq 1 12) (iseq 1 5)))
              (setf  rendi ( - (/ (select divisa (max
```

```
                                        (intersection
                                          (which (= anio i)) (which (= mes j)))))
                                (select divisa (min
                                        (intersection
                                          (which (= anio i)) (which (= mes j))))))  1))
          (setf rends (combine rends rendi))))
    (setf rends (rest rends))
    rends))
```

Para definir la función se utiliza `defun`. La función toma la variable `divisa` e itera sobre cada año definido en el vector `anio` para encontrar el valor final P_t e inicial P_{t-k} de cada mes (una excepción es el último año porque los datos están hasta mayo, por esta razón crea la lista de los meses con una secuencia del 1 al 12 y para el año 2004 la secuencia va de 1 a 5) y define el valor $R_t = \frac{P_t}{P_{t-k}} - 1$. Este valor es guardado en la variable `rendi` que se va agregando en la lista `rends`, por ultimo aplica la función `rest` que elimina la primera observación correspondiente a la creación de la variable; finalmente la variable `rends` es la que regresa la función.

```
> (rend-mensual cad)
(-0.0116568  -0.0141481 -0.0237063  0.00448528 -0.000867491  0.00216841
 -0.00310111  0.0164594  0.0167461 -0.00635813 -0.0152694   -0.0191169
 -0.00404772  0.0122739 -0.0021443  0.0314897  -0.0106577    0.00065597
  0.00285025  0.0186109  0.0018635 -0.00253221 -0.0083879    0.00073286
 ...
```

Para obtener los rendimientos de cada una de las divisas consideradas, se requiere aplicar esta función a cada una de ellas. Es posible hacer esto en forma automática como sigue. La instrucción

```
(def X (select (send tipos_cambio :data) (iseq 3 7)))
```

crea una variable `X` que es una lista con los datos correspondientes a las divisas en diferentes listas(en el conjunto de datos, las columnas 0, 1 y 2 contienen el día, mes y

año, respectivamente. aunque en la opción **Description** muestre las variables en orden alfabético). La instrucción

```
(def Y (mapcar #'rend-mensual X))
```

mapea la función `rend-mensual` a cada elemento de `X` y lo guarda en `Y` como listas los rendimientos de cada una de las divisas. Para identificar las fechas a las que corresponden los rendimientos, se requiere crear dos listas con los meses y los años correspondientes:

```
(def months (combine (repeat (iseq 1 12) 10) (iseq 1 5)))
(def years (repeat (iseq 1994 2004) (combine (repeat 12 10) 5)))
```

En la primera instrucción define la variable `months` a la que le asigna la lista de meses repitiendo 10 veces la secuencia del 1 al 12, posteriormente del 1 al 5. La segunda instrucción crea la lista `years` repitiendo 12 veces cada año de 1994 a 2003 y 5 veces 2004.

Finalmente, se puede generar un conjunto de datos que contenga los rendimientos, utilizando la función `append` concatenamos los rendimientos mensuales con el mes y año respectivos. La función `select` se utiliza para la identificación de los indices en este caso de los nombres de las variables de las columnas 3 a la 7 que son las correspondientes a las divisas, y las combina con las etiquetas `mes` y `año`:

```
(make-dataset
       :data (append  Y (list months) (list years))
       :data-names (combine (select (send tipos_cambio :names) (iseq 3 7)) "mes" "anio")
       :name "rendimientos")
```

Ahora se pueden guardar los datos en un archivo que se llamará `rend-tc.lsp`. En este conjunto de datos, que tiene un total de 125 observaciones mensuales, se pueden definir los excesos de rendimiento de cada variable con respecto al euro, por ejemplo `ejpy` se define como `jpy - eur`. Estas variables se crean con la opción `Add a variate...` Y guardamos las nuevas variables en el archivo junto con las variables rendimientos.[en] el mismo archivo.

2.1.3. Gráficas en *Arc*

El menú `Graph&Fit` que aparece cuando se lee un archivo con datos como se ha descrito en la sección anterior ofrece facilidades para crear y manipular objetos gráficos. Dentro de las gráficas que se pueden crear se incluyen: histogramas, gráficas de caja (boxplot), gráficas de dispersión de puntos (scatterplot), matrices de gráficas de dispersión y gráficas tridimensionales dinámicas.

Los elementos para crear gráficas en el menú de datos son los que a continuación se describen:

Plot of ... Permite crear histogramas y gráficas de dispersión de puntos en 2 ó 3 dimensiones. En el diálogo que aparece cuando se selecciona este elemento, aparece una ventana todas las variables de las qué un subconjunto pueden ser candidatos a graficarse, y una selección de ejes. Los ejes tienen nombres `H` (horizontal) para el eje de las abscisas, `V` (vertical) para el eje de las ordenadas y para el caso de los gráficos tridimensionales `O`(out-of-page) para el eje que está fuera del plano de la pantalla. Dependiendo de cuántos variables sean seleccionadas en los ejes es el tipo de gráfica que se obtiene. Para colocar las variables a graficarse en la ubicación del eje, primero se selecciona dando doble click sobre la variable la cual se ubicara ordenadamente sobre los ejes o dando un click sobre la variable y otro en el eje. Análogamente para las otras ventanas del dialogo.

Scatterplot Matrix of ... Crea una matriz de gráficas de dispersión.

Boxplot of ... Crea gráficas de caja.

Multipanel plot of ... Se puede crear una sucesión de gráficas manteniendo un eje fijo y variando el otro eje.

Probability plot of ... Crea una gráfica de probabilidad, también conocido como *qq-plot*.

Set Marks Permite definir una variable para marcar cada observación, usando colores y símbolos.

En cada gráfica hay varios menús y controles que permiten mejorar la información visual de los datos. Además, cada vez que se genera una gráfica, aparece un elemento en la barra de menús que permite realizar funciones adicionales sobre el objeto gráfico. A continuación se describirán las características especiales de cada una de estas gráficas. Para tal fin se utilizarán los rendimientos como datos de ejemplo.

Histogramas

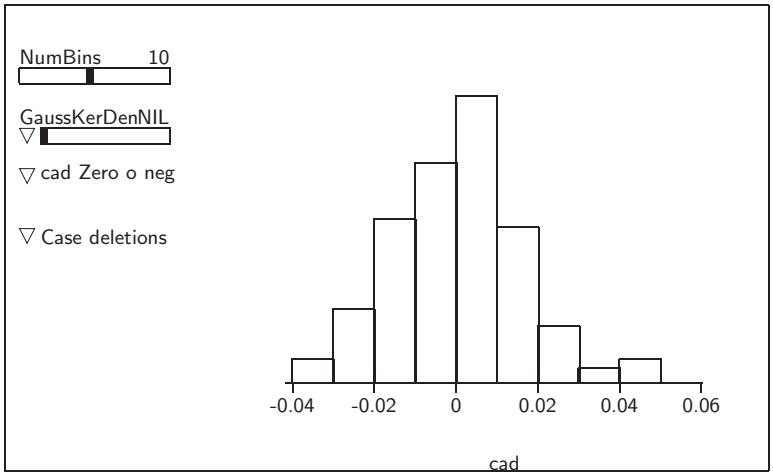


Figura 2.3: Histograma de los rendimientos canadienses

Un histograma se muestra como ejemplo en la figura 2.3, y tiene cuatro controles de tres tipos diferentes: el primero es una barra deslizante que permite seleccionar valores de ella, el último control es un menú que permite excluir observaciones de la gráfica y los dos restantes son combinación de menú y barra deslizante (en este ejemplo se encuentra inhabilitada la barra en el tercer control). Los triángulos representan los menús que permiten seleccionar más opciones autoexplicativas.

La primera barra sirve para seleccionar el número de subintervalos o `bins` que se usarán para construir el histograma.

La segunda barra sirve para agregar un suavizador para estimar la densidad de los datos. Este suavizador usa como `kernel` a la distribución normal. Conforme la barra se mueve a la derecha, el ancho de banda del estimador se incrementa, por lo que la curva obtenida es sobre-suavizada y sesgada. Cuando el ancho de banda es muy pequeño, la curva obtenida sub-suaviza los datos y se obtiene mucha variación (siempre hay una competencia entre sesgo y varianza). El valor óptimo del ancho de banda depende de la densidad verdadera de los datos y de cómo se defina el criterio de optimalidad para el ajuste de la curva. En *Arc* se usa el ancho de banda que sería óptimo para la estimación de una densidad normal: $h = 0.79\tilde{s}n^{1/5}$, donde $\tilde{s} = \text{mín}\{s, \frac{q_{0.75}-q_{0.25}}{1.34}\}$ y s es la desviación estándar muestral, y mapea este valor a 1 en la barra deslizante. Usualmente valores razonables a escoger en la barra están en el rango de 0.6 a 0.8 que corresponden a anchos de banda entre $0.6h$ y $0.8h$. Una explicación mucho más detallada del procedimiento de suavizamiento se encuentra en el libro de Simonoff (Sim96).

La tercera barra, que en el ejemplo presentado dice `Zero o Neg.` y en realidad está inhabilitada, es una barra que permite hacer transformaciones de la variable que se está graficando. La transformación es del tipo Box-Cox:

$$y^{(\lambda)} = \begin{cases} \frac{(y^\lambda - 1)}{\lambda} & \text{si } \lambda \neq 0 \\ \log y & \text{si } \lambda = 0 \end{cases}$$

La barra aquí aparece inhabilitada pues no se puede aplicar la transformación precedente a una variable con valores negativos o cero. Sin embargo, se puede utilizar sumando una constante par hacerlo positivo.

Gráficas en dos dimensiones

En la figura 2.4 se muestra una típica gráfica de dispersión. En la gráfica hay una paleta de colores y una de símbolos. que se usan para cambiar un color o un símbolo

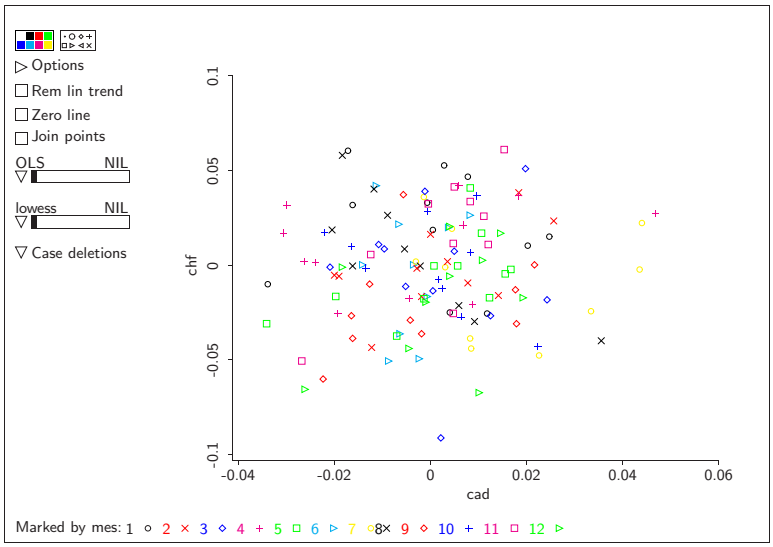


Figura 2.4: Gráfica de dispersión de puntos de *chf* vs. *cad*. Aquí se usan los meses como variable marcadora.

asignado a un punto en la gráfica.

Los otros controles que aparecen en la gráfica, de arriba a abajo, son:

1. **Options**: Este control da un dialogo que permite cambiar la apariencia de la gráfica cambiando escalas, etiquetas de los ejes, marcas, título, separar puntos sobrepuestos (jitter), etc. Además, con este menú se pueden agregar curvas de funciones arbitrarias.

2. Los tres controles siguientes son binarios: `Rem lin trend` elimina la tendencia lineal de los datos, graficando los residuales de la regresión de la ordenada en la abscisa. `Zero line` agrega una línea horizontal en el origen de las ordenadas, y `Join points` une con líneas los puntos en forma secuencial.

3. **OLS** es un control que permite agregar suavizadores paramétricos. Se puede cambiar la opción haciendo click en el triángulo. Algunas de las opciones que se incluyen son: mínimos cuadrados ordinarios (OLS), estimadores robustos de Huber (Huber M est), regresión logística, regresión de Poisson, o regresión Gamma. Al-

gunos detalles adicionales de estos métodos se verán en los ejemplos más adelante.

4. **lowess** (locally weighted scatterplot smoother) es un control que permite agregar suavizadores no paramétricos, como el estimador lowess que será descrito más adelante. También se puede agregar un "suavizador por rebanadas" que esencialmente divide el rango de la abscisa y calcula promedios de los puntos en cada rectángulo y después une los puntos.

   Con este control se permite obtener un estimador no paramétrico de la dispersión y agregar una banda a los datos, vía la opción `lowess ± SD`.

5. **Case Deletions** permite incluir o excluir puntos a las estimaciones.

Si hay una variable que marque los datos, se pueden hacer estimaciones por grupo de diferentes modos usando los menús 3 y 4. Un ejemplo se verá más adelante cuando se considere el análisis de covarianza.

Gráficas en tres dimensiones

Una gráfica tridimensional ha sido agregada como ejemplo. En esta gráfica, además de controles similares a los de las gráficas de dispersión, hay otros controles que permiten dirigir la forma en que se proyecta la gráfica sobre la pantalla para dar la impresión visual de tres dimensiones. Estos controles son `Rock`, `Pitch`, `Yaw` y `Roll`. Una descripción detallada de las opciones de esta gráfica y de la forma en que es construida se encuentra en (CW94).

Matrices de dispersión

En la figura 2.6 se muestra una matriz de gráficas de dispersión de los rendimientos de las divisas. Esta matriz es muy útil para encontrar transformaciones apropiadas

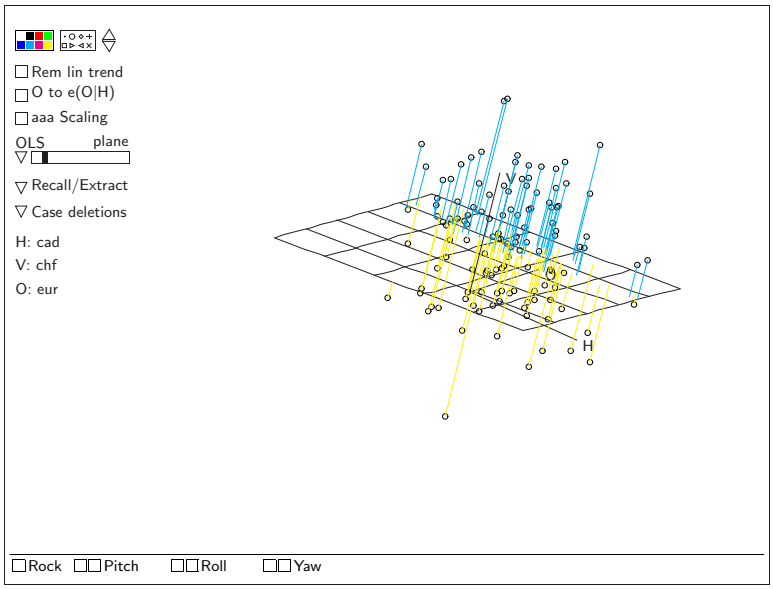


Figura 2.5: Gráfica tridimensional de tres divisas: *chf*, *cad*y *eur*. Un plano de mínimos cuadrados ha sido agregado y los residuales son mostrados explícitamente.

de las variables y además para verificar condiciones que permiten aplicar apropiadamente métodos gráficos de regresión. Adicionalmente, permiten encontrar posibles valores atípicos o extremos, los que pueden ser identificados en la gráfica cambiando el símbolo o color o etiquetando la observación. Los datos que se muestran en la diagonal son los valores mínimos y máximos de cada variable. Si hay una gráfica que sea relevante, se puede generar directamente si se presionan `Ctrl+Shift` y el botón izquierdo del ratón.

Gráficas de caja

En las gráficas de caja como la que se muestra en la Figura 2.7 se puede observar las principales características de la densidad de los datos. Es particularmente útil en observaciones que provienen de diversos grupos y se tiene interés en comparar sus principales características como medias, medianas, rangos intercuartiles y dispersión. El control `Show Anova` muestra una tabla de ANOVA abreviada para probar la hipótesis de igualdad de medias por grupo. Estas gráficas usualmente no están ligadas a otras gráficas.

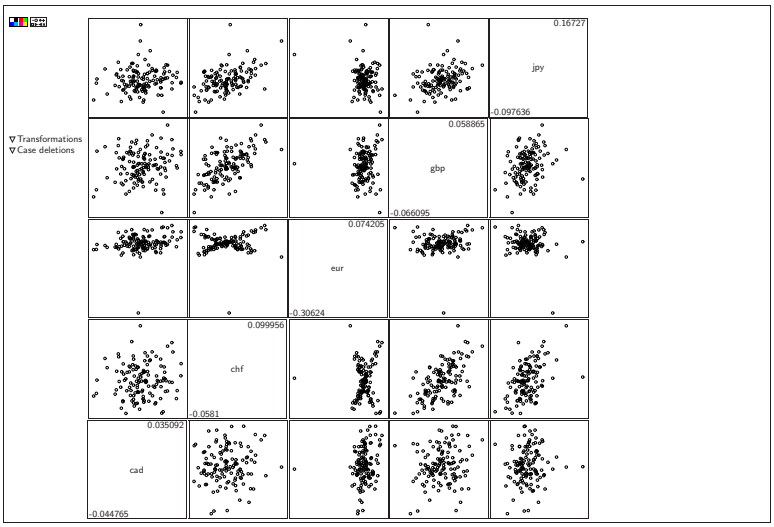


Figura 2.6: Matriz de gráficas de dispersión del rendimiento de las divisas.

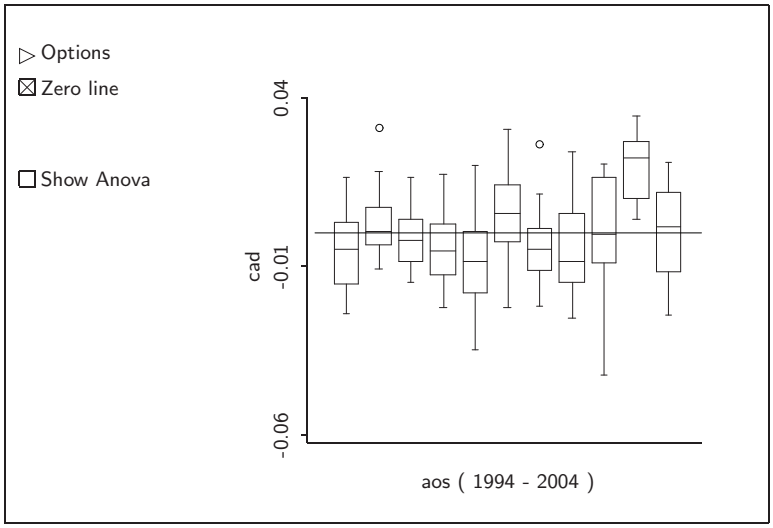


Figura 2.7: Gráfica de cajas del rendimiento *cad*, con datos anuales.

2.2. Ejemplo: Regresión no paramétrica

Los estimadores de regresión no paramétricos juegan un papel relevante para diagnosticar curvatura o varianza no constante en los modelos de regresión. Adicionalmente, permiten darse una idea del comportamiento de los datos sin la imposición de un modelo rígido.

En este ejemplo se introducirá un tipo particular de estimador no paramétrico de regresión que se conoce como loess o lowess, de *LOcally WEighted Smoothing Scatterplot*, introducido por Cleveland (Cle79) en 1979.

2.2.1. Promedios locales

La idea esencial en un ajuste no paramétrico de una curva $f(x)$ a un conjunto de datos (x_i, y_i) es hacer *promedios locales*. Si una curva de regresión $f(x) = E(y|x)$ es suficientemente suave, entonces observaciones con valores x cercanos a un punto focal x_0 serán informativos acerca de $f(x_0)$.

El procedimiento consiste en crear cajas o ventanas en torno a x_0 y mover estas ventanas continuamente sobre los datos, promediando las observaciones que caen dentro de la ventana. Se puede estimar $\hat{\{F\}}(x)$ en un gran número de puntos focales de x, usualmente equidistantes y cubriendo todo el rango de la variable independiente. Adicionalmente, la ventana puede ser de ancho fijo centrada en x_0 o se puede ajustar el ancho de la ventana para incluir un número constante m de observaciones, los vecinos más cercanos de x_0.

Este método tiene problemas de ajuste en los extremos en donde sólo hay vecinos de un sólo lado. Asimismo, se presenta sesgo en la presencia de valores extremos (outliers). Una forma de corregir estas deficiencias es introduciendo funciones ponderadoras como se describe a continuación.

2.2.2. Estimación de kernel

Una extensión de promedios locales es el *promedio local ponderado*, o estimación de `kernel`. La idea es darle mayor peso a las observaciones cercanas a x_0 y menos peso a las lejanas. Para esto, se consideran observaciones estandarizadas $z_i = \frac{x_i - x_0}{h}$ donde h es un factor de escala que se conoce como *ancho de banda* o *ventana*, y tiene una función

de calibración del ajuste. Una función de tipo kernel $K(z)$ usualmente es una función de densidad con media en x_0 que pondera la cercanía de las observaciones a x_0.

Con la introducción de los kérneles, es posible corregir el sesgo de estimación que persiste en los extremos del rango de la variable independiente. Una vez evaluados los pesos $w_i = K(z_i)$ se calcula el estimador de la regresión no paramétrica en x_0 como:

$$E(y|x_0) = \frac{\sum w_i y_i}{\sum w_i}.$$

2.2.3. Estimador loess o lowess

Loess o lowess es una implementación de suavizamiento no paramétrico que utiliza las ideas descritas anteriormente, para estimar en forma no paramétrica una curva de regresión.

El procedimiento consiste en que en cada ventana se realiza un ajuste polinomial en el punto focal x_0, usando ponderadores locales promedio $w_i = K(z_i)$. Esto es, se ajusta en cada ventana el modelo de mínimos cuadrados ponderados:

$$y_i = a + b_1(x_i - x_0) + b_2(x_i - x_0)^2 + \cdots + b_p(x_i - x_0)^p + e_i$$

usando como función objetivo de mínimos cuadrados a $\sum w_i e_i^2$ y una vez obtenidos los parámetros el valor promedio local ajustado (loess) es $\hat{E}(y|x_0) = a$. Adicionalmente, se puede obtener un estimador para la varianza tipo lowess del mismo modo que se obtiene el estimados para la media.

En la figura 2.8 se pueden observar los estimadores lowess para la media y la desviación estándar de la serie de tiempo del rendimiento del *euro*. Adicionalmente, se ha sobrepuesto en la gráfica la línea de mínimos cuadrados (OLS). Aquí se puede ver que una regresión lineal no sea la mejor forma de modelar la serie ya que muestra un comportamiento sinusoidal. Sin embargo, la dispersión de los datos parece ser constante a lo largo de la serie, lo que justifica el supuesto de varianza constante de los errores.

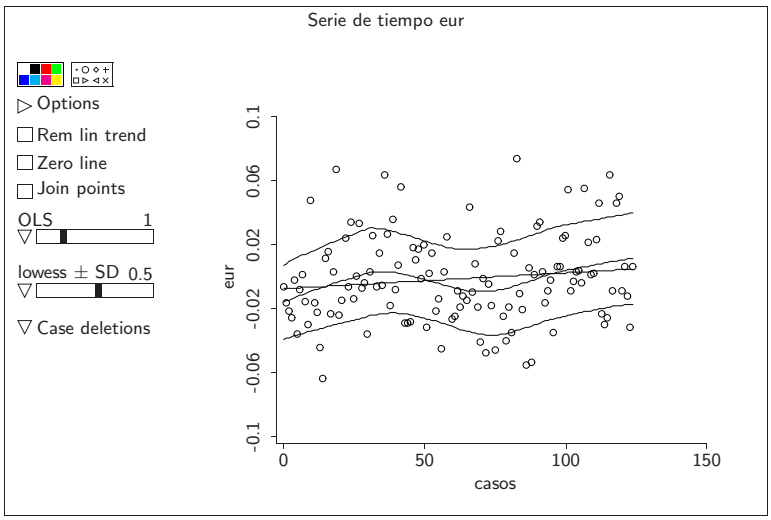


Figura 2.8: Gráfica de la serie de tiempo del rendimiento del *eur* con suavizador lowess

En la práctica, el ancho de banda h se escoge en valores entre 0.5 y 0.7 para establecer un balance entre suavizamiento y fidelidad a los datos.

2.3. Ejemplo: Regresión

En esta sección se estimará el modelo de regresión para portafolios que fue introducido antes.

Para ajustar el modelo, se seleccionan los elementos **Graph&Fit** → **Fit Linear LS...** y en el diálogo que se obtiene se ponen como predictores las variables `ecad`, `echf`, `egbp` y `ejpy` y como respuesta a la variable `eur` (de acuerdo al modelo, entonces los coeficientes del portafolio que serán obtenidos son negativos). El resultado que se obtiene es un modelo lineal llamado `L1` que genera un elemento en el menú principal, además obtiene la siguiente salida que es estándar en los problemas de regresión, con los valores estimados y la tabla de ANOVA:

```
Data set = rendimientos, Name of Fit = L1
```

```
Normal Regression
Kernel mean function = Identity
Response      = eur
Terms         = (ecad echf egbp ejpy)
Coefficient Estimates
Label      Estimate        Std. Error    t-value    p-value
Constant  -0.000230824     0.00120125     -0.192     0.8479
ecad      -0.579125        0.0570433     -10.152     0.0000
echf      -0.00688512      0.0545853      -0.126     0.8998
egbp      -0.270117        0.0683064      -3.954     0.0001
ejpy      -0.0796773       0.0383848      -2.076     0.0401

R Squared:               0.887287
Sigma hat:               0.013347
Number of cases:             125
Degrees of freedom:          120

Summary Analysis of Variance Table
Source         df       SS            MS           F    p-value
Regression      4  0.168284      0.0420711    236.16    0.0000
Residual      120  0.0213772     0.000178143
```

De acuerdo a este resultado, el modelo sugiere que se invierta 57.9 % en dólares canadienses, 0.7 % en francos suizos, 27.0 % en libras y 8.0 % en yenes. Estos porcentajes acumulan el 93.6 % del presupuesto, lo que sugiere que el resto, 6.4 % se invierta en euros. La ordenada al origen no es significativa, y de hecho, el coeficiente de francos suizos podría ser considerado como 0 en este modelo. El estimador de la desviación estándar del rendimiento del portafolio es $\hat{\boldsymbol{\sigma}} = 0.013347$ u 133 puntos base [4]. La R^2 del rendimiento del portafolio indica que el modelo lineal explica en 88.7 % la variabilidad de los rendimientos del euro [5].

Una gráfica de los residuales se muestra en la figura 2.9. Esta gráfica se puede obtener removiendo la tendencia lineal de la gráfica de los residuales del modelo $r_1 - \hat{r}_1$ versus

[4] Los puntos base es una medida típica de cambios en la tasa de interés o rendimiento; y un punto base corresponde a una diezmilésima de unidad.

[5] Si se ajusta un modelo sin constante, la R^2 pierde esta interpretación.

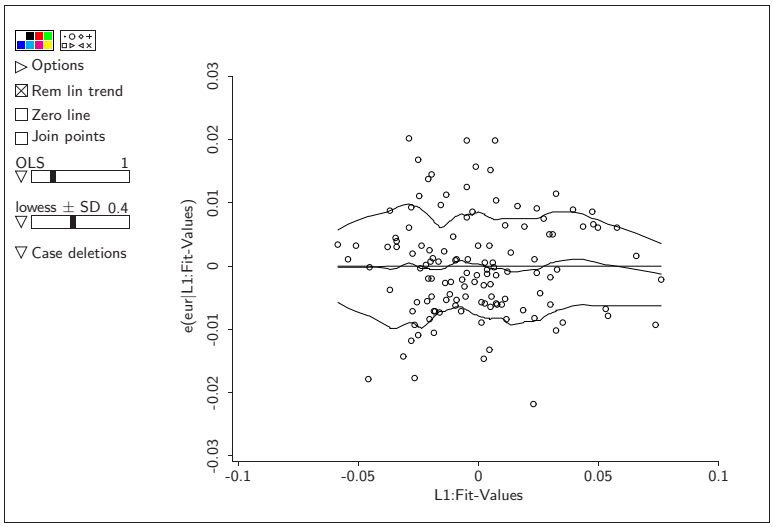


Figura 2.9: Gráfica de los residuales del modelo de portafolio ajustado.

los valores ajustados $\hat{y}$. Adicionalmente, se pueden agregar ajustes loess para la media y la desviación estándar. Para facilitar la visión, la observación 59 se eliminó del conjunto de datos ya que parece ser un valor atípico y al parecer también es influencial. En la gráfica puede apreciarse que el supuesto de varianza constante para los errores puede ser razonable, quizá excepto en los extremos. Sin embargo hay un número grande de observaciones que están lejos de la curva ajustada. Hay puntos que están tan lejos como 200 puntos base. Esto parece indicar de manera preliminar que el modelo de portafolios podría ser mejorado para obtener ponderaciones de inversión más robustas.

Para poder hacer un estudio más detallado de este modelo, como validación de los supuestos, verificación de valores influenciales y extremos (outliers) o bien para hacer predicciones y verificar hipótesis, se cuenta con más herramientas de análisis. Es posible, por ejemplo, obtener la matriz de covarianzas de los coeficientes estimados. Esta es útil para obtener intervalos de confianza para los parámetros y también para obtener predicciones o varianzas de combinaciones lineales de los coeficientes. Si $\hat{\boldsymbol{\beta}}$ es el estimador de mínimos cuadrados, entonces $\hat{\boldsymbol{\beta}} \sim \mathcal{N}_p\left(\boldsymbol{\beta}, \sigma^2(\mathbf{X}'\mathbf{X})^{-1}\right)$. La matriz de covarianzas estimada se obtiene reemplazando σ^2 por $\hat{\sigma}^2$. Esta matriz y la de correlaciones se obtiene vía **L1** $\rightarrow$

display variances:

```
Variance-covariance matrix of the coefficient estimates
Constant  1.2614E-6   1.2878E-6   1.6719E-7  -4.2762E-6   1.9439E-6
ecad      1.2878E-6   0.0030944   0.00014304 -0.0014535  -0.00059366
echf      1.6719E-7   0.00014304  0.0026835  -0.0020232  -0.00066842
egbp     -4.2762E-6  -0.0014535  -0.0020232   0.0041228  -0.00014409
ejpy      1.9439E-6  -0.00059366 -0.00066842 -0.00014409  0.0013106
           Constant      ecad        echf        egbp        ejpy
```

La raíz cuadrada de los elementos de la diagonal corresponden a los errores estándar de los coeficientes, que se reportan en la tabla de valores estimados.

En el próximo capítulo de esta tesis se hará un estudio más detallado de estos datos.

2.4. Ejemplo: Simulación

En este ejemplo se hará una simulación de un fenómeno estocástico. Se supondrá que se lanzan dardos a un blanco desde diferentes distancias. Se hacen 10,000 lanzamientos en total con 200 tiros por cada una de las 50 distancias de tiro. Cada lanzamiento se considera una variable Bernoulli que toma el valor de 1 si pega en el blanco. Para considerar el efecto de las diferentes distancias, se supondrá que la probabilidad de éxito es inversamente proporcional a un número aleatorio entre 1 y $d+1$. Definimos a X_d como la variable que cuenta el número de aciertos a distancia d. Entonces $X_d \sim$ **Bin** $(200, p_d)$. Aquí suponemos además que $X_1, X_2, \ldots, X_d$ son independientes.

Para este ejercicio se requiere simular variables aleatorias binomiales para cada lanzamiento. El siguiente script define una función sin argumentos llamada `simula` y genera 50 variables necesarias para cada distancia y crea un conjunto de datos llamado *dardos*. Una vez generado el conjunto de datos, éste es guardado en `dardos.lsp`

```
(defun simula ()
 (def dist (iseq 1 50))
```

```
(def freq (combine (binomial-rand 1 200 (/ 1 (+ 1 (random dist))))))
(make-dataset
       :data (list dist freq)
       :data-names (list "dist" "freq")
       :name "dardos"))
```

Para ejecutar la función `simula`, únicamente se debe escribir en el listener

```
>(simula)
```

Para el resto de este ejemplo, supondremos que los datos fueron observados en un experimento y que no se conoce la distribución de las probabilidades de acierto en cada distancia. De este modo, se supondrá que los aciertos forman una muestra aleatoria. El propósito del resto del ejemplo es mostrar qué cálculos y gráficas se pueden hacer para verificar normalidad de datos. Se considerarán las siguientes características:

1. Elaboración de un histograma.
2. Cálculo del rango intercuartil: si los datos son normalmente distribuidos, entonces el rango intercuartil $r_{iq} = q_{.75} - q_{.25}$ debe ser muy cercano a 1.35.
3. Regla empírica de probabilidad: a una distancia de una desviación estándar de la media, se esperaría concentrar cerca de 68 % de los datos; a dos desviaciones el cerca del 95 % y a tres desviaciones cerca del 99 % de las observaciones.
4. Gráficas de probabilidad de los datos.

Las transformaciones son muy importantes en estadística. Con frecuencia buscamos transformaciones que hagan que los datos se vean más normales. A continuación se considera cada uno de éstos casos.

1. ¿Cómo se construye un histograma? Los datos ya están dados en frecuencias, así que podemos graficar directamente las frecuencias versus las distancias: **Graph&Fit** → **Plot of...**

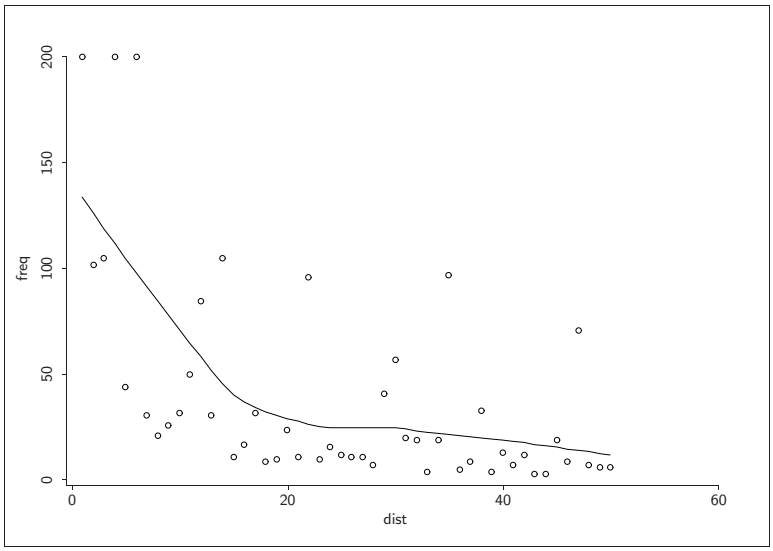


Figura 2.10: frecuencias por distancias.

Un suavizador lowess se agregó para dar una idea de la función de densidad. Podemos ver que la distribución es claramente asimétrica, es más probable dar en el blanco a distancias cortas, aunque hay distancias en las que ocasionalmente se tiene una buena racha de tiros acertados. Esto da indicios que una distribución normal no sería adecuada para modelar los datos.

Cuando los datos se dan en frecuencias, ¿Cómo se calcula la media y la desviación estándar?

En *Arc*, necesitamos agregar dos variables: una para las frecuencias relativas, definida como

```
relfreq=(/ freq (sum freq)) ;
```

y la segunda es una variable, digamos w que es el producto de las frecuencias relativas por las distancias:

`w=relfreq*dist` . Estas dos variables pueden agregarse vía el menú de datos **dardos** → **Add a variate...**. Finalmente, para obtener la media tenemos que sumar todos los valores de w, que se puede hacer introduciendo

```
>(sum w)
15.4592
```

la desviación estándar es similar: ahora necesitamos otra variable, digamos $w2$, que se define como

`w2=relfreq*(dist-15.4592)^2` . Sumamos los valores como antes y tomamos la raíz cuadrada.

```
> (sqrt (sum w2))
14.0124
```

2. Para obtener el primer cuartil q_1 y el tercer cuartil q_3 se requiere un poco más de trabajo. Necesitamos sumar las frecuencias relativas hasta que se obtenga un valor cercano a 0.25, luego la posición de esa probabilidad en la variable distancia corresponde a q_1 y de forma similar, sumamos las frecuencias relativas hasta que acumulemos 0.75 para encontrar la distancia que corresponda a q_3.

```
> (cumsum relfreq)
(0.101368 0.153066 0.206285 0.307653 0.329954 0.431323 0.447035 0.457679
 0.470857 0.487076 0.512418 0.555499 0.571211 0.62443 0.630005 0.638621 0.65484
 0.659402 0.66447 0.676635 0.68221 0.730867 0.735935 0.744045 0.750127 0.755702
 0.761277 0.764825 0.785606 0.814496 0.824633 0.834263 0.83629 0.84592 0.895084
 0.897618 0.902179 0.918905 0.920933 0.927522 0.931069 0.937152 0.938672
 0.940193 0.949823 0.954384 0.99037 0.993918 0.996959 1)
```

Las posiciones son 2 (0.206285) y 23 (0.744045) (recordar que se comienza a contar la posición desde 0) El rango intercuartil estandarizado es entonces $\frac{r_{iq}}{s} = \frac{23-2}{14.0124} = 1.4986$ el número no está muy alejado de 1.3.

3. Podemos también verificar la regla empírica

$$\begin{aligned} \bar{x} \pm s &= \text{(+ media (list (- s) s))} \approx (1.45, 29.47) \\ \bar{x} \pm 2s &= \text{(+ media (* 2 (list (- s) s)))} \approx (-12.57, 43.48) \\ \bar{x} \pm 3s &= \text{(+ media (* 3 (list (- s) s)))} \approx (-26.58, 57.50) \end{aligned}$$

Cerca de 1350 de las 1973=`(sum freq)` aciertos, o 68.42 % están en el primer intervalo, y cerca de 93.87 % (1852) están en el segundo intervalo, y el 100 % de las observaciones están en el tercer intervalo. La regla empírica exclusivamente diría que los datos se comportan similar a una distribución normal.

4. La última prueba es hacer una gráfica de probabilidades. Una complicación es que los datos están dados en frecuencias. Necesitamos primero manipular los datos para obtenerlos en el formato adecuado. Primero creamos una nueva variable llamada *dist2* con el comando:

```
> (def dist2 (repeat dist freq))
```

Esta variable descompone a las distancias en sus respectivas frecucencias. Ahora creamos un nuevo conjunto de datos con esta variable:

```
>(make-dataset :data (list dist2))
```

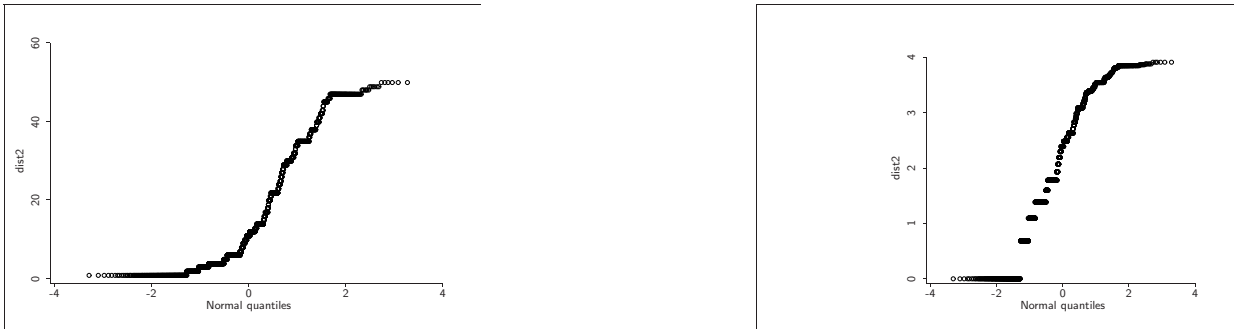
En *Arc*, seguimos la secuencia de comandos **Graph& Fit→ Probability plot of...** y seleccionamos la variable de interés, junto con la distribución normal para obtener la Figura 2.11a.

Esta gráfica muestra que los datos no se distribuyen normalmente. Una pregunta interesante podría ser ¿qué pasa si en lugar de los datos originales los transformamos? Si tomamos por ejemplo logaritmos, obtenemos la Figura 2.11b.

Aún con los datos transformados, se observa que los puntos no se alinean, sino que en los extremos de la gráfica hay curvatura , por lo que los datos transformados tampoco se aproximan a una distribución normal. Una opción disponible en una gráfica de probabilidades simular una cobertura (envelope) o banda en la que sería válido considerar una distribución normal.

Si los datos no están agrupados en frecuencias, las cosas son mucho más fáciles. Por ejemplo, para encontrar $q_{0.25}$ y $q_{0.75}$ sólo aplicamos las funciones:

```
> (quantile dist2 .25)
4
> (quantile dist2 .75)
25
> (mean dist2)
15.4592
> (standard-deviation dist2)
14.016
> (interquartile-range dist2)
21
```



a. Gráfica de probabilidad para las distancias.

b. Gráfica de probabilidad después de transformar los datos.

Figura 2.11: Gráficas de probabilidad para el ejemplo de la simulación

2.5. Otras aplicaciones

En este capítulo hemos mostrado algunas posibles aplicaciones de *Arc* para análisis de datos, y puede ser utilizado en una gran variedad de aplicaciones más, como regresión logística, regresión Poisson para el análisis de datos categóricos y modelos lineales generalizados, además de modelos de regresión no lineal. Una descripción detallada de aplicaciones en estas áreas puede encontrarse en (CW94; CW99).

Arc puede utilizarse como herramienta computacional en cursos como: Análisis de

regresión lineal, regresión no lineal, diseño de experimentos, control estadístico de la calidad, etc.

En lo que resta de esta tesis nos concentraremos a ejemplos de modelos de regresión lineal.

2.6. Introducción general a R.

2.6.1. Un poco de historia R

El lenguaje S se desarrolló en Laboratorios AT&T-Bell, principalmente por John M. Chambers, en 1976. Su uso se expandió rápidamente después de la publicación del libro de John Tukey "Exploratory Data Analysis" (EDA).

S-Plus es una versión comercial de S que inició en 1987. Su popularidad se incrementó dramáticamente después de 1990. Éste era caro y requería de licencia. S-Plus corre sobre varios ambientes: Windows, Unix y Linux, y tiene un GUI "Graphic User Interface", además soporta muchos formatos para importación y exportación de datos y gráficas.

R se comenzó a desarrollar por estadísticos a principios de los 90's, como una alternativa de código abierto a S-Plus en parte porque en esa época no había una versión de S-Plus para Linux. Los pioneros de R son Robert Gentleman y Ross Ihaka, de Nueva Zelanda. R se basa en el mismo lenguaje S utilizado en la versión 2000 de S-Plus con algunas excepciones menores.

Las ventajas que tiene R sobre otros paquetes es que está bien documentado, es fácil de obtener e instalar así como de actualizar sus bibliotecas de funciones. También correr en plataforma de Macintosh. Aunque solo cuenta con GUI ("Graphics User Interface") muy básica en Windows, que no está disponible en otras plataformas. Tiene menos capacidades para importar y exportar que en S-PlusṄo tiene capacidad de exportar a Powerpoint, en forma explícita.

2.6.2. ¿Porqué R?

Dos paquetes estadísticos importantes en el mercado están orientados a procedimientos: SASy SPSS. Sus principales desventajas es que carecen de buenas gráficas interacti-

vas, es difícil implementar nuevos métodos y a veces no se tiene idea de cómo se realizan ciertos cálculos.

En cambio S es paquete orientado a objetos, lo que permite análisis de datos y gráficas en forma interactiva, fácil de implementar nuevos métodos y distribuir a otros usuarios. Su código es abierto, i.e. el usuario sabe qué está haciendo el paquete. Y muchos de los investigadores de punta en estadística son programadores de R Brian Ripley, Luke Tierney, John Fox, Douglas Bates, Sanford Weisberg, etc.

S fue planeado para ser extensible, i.e. los usuarios escriben nuevas funciones, al igual que los desarrolladores. La documentación para agregar funciones es excelente. Las funciones creadas por los usuarios se invocan igual que las internas. Además los usuarios pueden crear sus propios tipos de datos y agregar atributos, p.ej. comentarios a cada pieza de datos de SÈn S puedes trabajar con elementos de datos que pueden ser complejos y asimétricos (por ejemplo árboles), la comunidad internacional de usuarios agregan nuevas capacidades todo el tiempo. Este software contiene un lenguaje de alto nivel, i.e. unos cuantos comandos hacen mucho trabajo, además permite crear las mejores gráficas científicas disponibles.

En R ya contiene varios programas, que han sido contribuciones de otros usuarios, que permiten realizar eficiente y correctamente varios análisis, como son:

- Análisis estadístico multivariado.
- Modelos lineales y modelos lineales generalizados.
- Simulación de procesos estocásticos, en particular análisis de series de tiempo y modelos econométricos.
- Cálculos de modelos bayesianos jerárquicos (en combinación con BUGS).

Lo que permite concentrarse más en la interpretación que en la implementación.

2.6.3. Configuración Inicial

Para trabajar en diferentes proyectos sin mezclar datos o variables, se recomienda:

1. Crea un directorio para un proyecto de trabajo, por ejemplo en:
   `D:/mis documentos/claseR/ .`
2. Crea un acceso directo al ejecutable `Rgui.exe` desde el escritorio. El ejecutable está en `rw1091\bin\`. Se puede copiar el Acceso directo de uno que ya esté creado.
3. Edita en el Acceso directo el directorio de inicio (click derecho → Propiedades → Iniciar en...) y escribir el nombre del directorio, entre comillas.
4. Hacer doble click sobre el acceso directo. Escribir `getwd()` para ver cuál es el directorio de trabajo. Si se quiere cambiar el directorio de trabajo durante una sesión, se usa, por ejemplo, `setwd(d:\mis documentos\otro\)`

Formas de ejecución

R puede ser ejecutado en tres formas: En su propia ventana (`Rgui.exe`) , en una ventana de DOS (`R.exe` o `Rterm.exe`), o en BATCH (`Rcmd.exe`). Estos archivos se encuentran en `rw1091\bin\`.

Al final de cada sesión se puede salvar todo lo que fue creado durante la sesión de trabajo (variables, comandos). Se crean dos archivos:

1. `.RData`, que contiene todas los objetos creados (excepto gráficas). Si este archivo existe cuando se inicia *R* automáticamente es cargado a la memoria.
2. `.Rhistory`, contiene todos los comandos introducidos en la sesión de trabajo. Es un archivo de texto que puede ser abierto con cualquier editor de texto.

En modo interactivo, *R*usualmente necesita un editor de texto, como Notepad, Winedit, WinEdt, WordPad, y otros editores de ASCII.

Apariencia inicial

La apariencia inicial de *R* (colores, ventanas, fonts, etc.) puede ser cambiada. El archivo que contiene toda la información del ambiente es `rw1091\etc\Rconsole`. Éste permite crear configuraciones especiales en cada directorio de trabajo. Para esto:

Se requiere crear un archivo de nombre `.Renviron` en el directorio de trabajo. En este archivo se definen las variables de ambiente. Se debe agregar la línea:

```
R_USER="d:\pon aqui tu directorio de trabajo"
```

Para saber cuál es el valor de la variable `R_USER` usa `Sys.getenv("R_USER")`.

Archivos de configuración importantes

Hay tres archivos de configuración que pueden ser cambiados por el usuario: hay versiones de los tres en el directorio `R_HOME\etc`:

Rconsole Atributos de aspecto.

Rprofile Código de *R* de opciones generales, como editor a ser usado, y código que quiere ser ejecutado al inicio (y al final)de una sesión. Aquí se puede definir funciones `.First` y `.Last`.

Rdevga Configuración de elementos gráficos iniciales (fonts,tamaños, etc). Usualmente no se requieren cambios.

2.6.4. Reglas generales

- Comandos pueden introducirse en más de una línea. el `prompt` de continuación es: "+"

- Comandos múltiples pueden ser introducidos en una misma línea separados por ";"
- Comentarios son con "#".
- Espacios y tabuladores son ignorados excepto cuando están entre comillas.
- Mayúsculas y minúsculas son diferentes.
- Se pueden usar las teclas ↑ o ↓ para navegar entre los comandos que han sido tecleados previamente.
- Se puede obtener ayuda de una función, por ejemplo `sin`, usando la ayuda HTML o bien tecleando `?sin`.

2.6.5. Ejemplos

- Los comandos más elementales consisten de expresiones o asignaciones.

```
> 2+3
> sqrt(3/4)/(1/3-2*pi^2)
> x <- rnorm(100,sd=4) ; y <- runif(100,-2,5)
> plot(x,y)
> z <- cbind(x,y)
> t(z) %*% z
> crossprod(z)
> h <- chull(x,y)
> polygon(x[h],y[h],dens=15,angle=30)
> objects() #o bien ls()
> cosangle <- function(vec,mat){
+   vec1<-vec/sqrt(crossprod(vec))
+   qq<-svd(crossprod(mat))
+   crossprod(solve(qq$v%*%diag(sqrt(qq$d))) %*%
+ (t(mat)%*%vec1)) }
```

- Objetos más complejos tienen clases que determinan cómo serán impresos o graficados.

2.6.6. Descripción de objetos en *R*.

En R todo es objeto, hasta las funciones y los operadores. Se trabaja con objetos aplicando funciones sobre ellos. Los *atributos* de un objeto son a su vez objetos subordinados que contienen información sobre el objeto principal. Todos los objetos tienen por lo menos 2 atributos: `mode` y `length`. Se puede accesar a éstos con las funciones del mismo nombre.

mode es lo que distingue a objetos de datos (`numeric`, `complex`, `character`, `logical`, etc.) de los objetos de lenguaje (`function`, `call`, `expression`, `name`, `missing`, etc.).

Se puede crear un objeto `resultado` con atributos `se` y `p` usando

```
resultado <- structure(x,se=se, p=p)
```

Los objetos también tienen una `clase`, que es un atributo más. Esta `clase` es fundamental para programación orientada a objetos. La clase de los objetos con los que los usuarios trabajan con mayor frecuencia son: *vectores, funciones* y *listas.*

Usualmente, no se requiere saber si los valores numéricos son `enteros`, `reales` o `complejos`, y si son de precisión simple o doble. En R todos valores numéricos tienen precisión doble.

Vectores y matrices

Vectores

- Los *vectores* pueden ser numéricos, lógicos o de caracteres, pero no se pueden mezclar.

```
> x <- 1:5
> colores <- c("rojo","amarillo","negro","azul","blanco")
> names(x) <- colores
```

```
> x
  rojo   amarillo    negro     azul   blanco
     1        2         3        4        5
```

- Para accesar a elementos de un vector los referimos por el nombre del vector y la posición de la entrada de interés entre corchetes: `x[2]` o `x[1:3]`, o por el nombre de la entrada, si se tiene: `x["rojo"]`. Y para eliminar alguna entrada del vector se antepone el signo "-´´ a la posición de la entrada que se desea quitar: `x[-1]`. con lo que se reduce la dimensión del vector.
  - Valores lógicos son `TRUE(T)` y `FALSE(F)` son variables con esos valores.
  - Complejos: `sqrt(-2)` vs. `sqrt(-2+0i)`
- Otras formas de crear vectores:

```
z <- vector("tipo",long)
z <- numeric(long)
z <- character(long)
z <- logical(long)
z <- numeric(0) es mejor que z <- NULL
```

Matrices

- Una *matriz* es un vector con atributo `dim`.

```
> x <- 1:20
> x
 [1]  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20
> dim(x) <- c(2,10)
> x
      [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10]
[1,]    1    3    5    7    9   11   13   15   17    19
[2,]    2    4    6    8   10   12   14   16   18    20
> dim(x) <- NULL
> x
 [1]  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20
```

- Para crear una matriz, se puede usar `matrix(x,2,5)`. La matriz también se puede llenar por renglones usando `matrix(x,2,5,byrow = T)`.
- Se pueden accesar a los elementos de la matriz usando `x[m,n]`, donde `[m,n]` refiere la posición de la entrada `[renglon,columna]` y, los renglones y columnas completas se utiliza `x[m,]` y `x[,n]`, respectivamente.
- El número de renglones y columnas pueden obtenerse con las funciones `nrow` y `ncol`.
- Las funciones `cbind` y `rbind` se usan para concatenar vectores o matrices por columna o por renglón.
- Un *Arreglo* es una extensión de una matriz, tomando el atributo `dim` de longitud 3 o más.

```
x <- 1:30
dim(x) <- c(5,2,3)
dimnames(x) <-list(color=colores,tipo=c("gordo","flaco"),var=c("H","M","N"))
attributes(x)
```

- Los elementos se pueden accesar como en el caso de vectores.
- La convención en *R* para llenar un arreglo es la misma que en Fortran: el primer índice es el más rápido y el último más lento.
- Vectores, matrices, arreglos y factores son considerados el mismo objeto pero con diferentes atributos.

Listas

- Una *lista* es un objeto que sirve para contener otros objetos que pueden ser de diferentes tipos. Es similar a las estructuras que se definen en algunos lenguajes de programación como *Pascal*.

- Esta puede ser la estructura de datos más interesante para almacenar datos:

```
Bono <-
list(tipo="Cete",plazo=28,vence="01/04/04",tasa=7.58,nodos=c(9.94,9.96,9.93))
```

- A los elementos de la lista se puede accesar por número de posición o por nombre:

```
Bono[[3]] #tercer objeto en la lista
Bono[3] #lista con un componente
Bono$vence
Bono[3:4] #se seleccionan varios elementos como vector
Bono <- c(Bono,precio=9.99865)
Bono$precio <- NULL  #o Bono[["precio"]] <- NULL Elimina el componente precio
Bono$precio <-list(NULL) #Hace precio NULL en el arreglo.
```

- Los nombres de los campos pueden ser abreviados al número mínimo de caracteres que los hacen únicos.
- La función `unlist` convierte una lista en un vector (dominan los caracteres).

Factores

- Los *factores* son tipos especiales de vectores que se utilizan para guardar variables categóricas.
- Los factores son importantes para las funciones estadísticas. Se tratan de una forma diferente que a un vector de etiquetas.

```
operacion <- factor(c("rep","dir","rep","rep","rep","dir","desc","desc"))
```

- Los niveles son ordenados alfabéticamente. Algunas funciones le dan un valor especial al primer nivel, por lo que a veces se requiere dar explícitamente los niveles.
- Cuando los datos categóricos son ordinales se usa:

```
ingreso <- ordered(c("M","L","M","H","M","L","L","H"),levels=c("L","M","H"))
```

- Se puede discretizar una variable usando `cut`:

```
z <- rnorm(100)
u <- cut(z,breaks = c(-4,-3,-2,-1,,0,1,2,3,4))
u <- ordered(u, levels(u))
```

- Los factores permiten clasificar o agrupar datos, y son muy útiles para indexar.

Data Frames

- Los *data frames* son listas hechas de vectores de la misma longitud, pero pueden ser de diferente tipo.
- Un data frame tiene atributos que las listas no tienen. La función `data.frame` genera un data frame. Se puede usar la función `cbind` o `rbind` como si los datos fueran una matriz, pero regresa un data frame.

```
z <- data.frame(  inst = factor(c("Cete","Cete","M0","M1","M1")),
                precio = c(9.91,9.93,9.988,9.87,9.67),
                ven =rep("01/10/04",5))
cbind(z,z)
attributes(z)
```

- Los data frames pueden ser indexados como matrices o como listas.

```
z[2] #es un data frame
z[[2]] #es un vector, que es el segundo elemento de la lista
z[z$inst == "Cete",]
```

- Los elementos de un data frame se pueden accesar en la forma usual o usando `attach` (al final `detach`).

- Matrices y listas pueden ser anexadas a un data frame:

```
y <- matrix(rnorm(10),nrow=5)
z <- data.frame(z,y)
```

Indexación

Para vectores, los vectores índices pueden ser de 5 tipos:

1. Un vector lógico: `y[y<0]`
2. Un vector de enteros positivos o un factor: `z[z$inst == "M0",]`
3. Un vector de enteros negativos: sirve para *excluir* términos: `y[-c(2,5,8)]`
4. Un vector de cadenas de caracteres: esto sólo aplica cuando el objeto tiene nombres, de otra forma, devuelve `NA`.
5. La posición de índice es vacía. `y[]`. Se comporta como si se reemplazara el índice por `1:length(y)`.

- Un arreglo puede ser indexado por una matriz: si el arreglo tiene k índices, la matriz índice de ser de dimensiones $m \times k$ y cada renglón de la matriz es usado como un conjunto de índices especificando un elemento del arreglo.
- Índices ceros no caen en ningún caso anterior: un índice 0 en un vector ya creado para nada que al asignarle un valor no lo acepta:

```
a <- 1:4; a[0]; a[0] <- 10; a
```

2.6.7. Introducción a gráficas

- *R* permite hacer gráficas de muy diversos tipos, con alta calidad, listas para imprimir en artículos y libros.

- Una panorámica muy general de las gráficas puede verse con `demo(graphics)`.

Tipos de gráficas

1. Histogramas,densidades.
2. Gráficas de dispersión de puntos.
3. Matrices de dispersión.
4. Líneas de funciones escalonadas.
5. Boxplots o gráficas de caja
6. Funciones en general
7. Gráficas combinadas.

Función `plot()`

- `plot` es la función más básica. Se puede aplicar a muchos objetos y dar un tipo de gráfica que sería el más apropiado para ese objeto.
- Hay muchos argumentos que pueden ser cambiados por el usuario: `main, xlim, ylim, sub, xlab, cex, type, pch, lty, col`...
- Podemos manipular parámetros más generales usando la función `par`. Por ejemplo podemos determinar cuantas gráficas por hoja queremos.

Paquetes gráficos importantes

- El paquete `trellis` tiene muchas funciones necesarias para gráficas condicionales complejas.
- El paquete `car` También incluye gráficas preconstruidas que son interesantes.

- El paquete `iplots` permite hacer gráficas dinámicas y tridimensionales utilizando *Java* (necesita tener java instalado en la computadora para que funcione.

Guardando gráficas

- En Windows, podemos usar los menús y guardar con distintos formatos. También podemos especificar donde queremos guardar el gráfico:

```
pdf(file = "f1.pdf", width = 8, height = 10)
plot(rnorm(10))
dev.off()
```

- O bien, podemos copiar una figura a un archivo: `plot(runif(50)) dev.copy2eps()`

2.6.8. Definición de funciones.

- Ya hemos visto varios ejemplos de funciones. La estructura básica es

```
my.funcion <- function(x,y,etc) {(definicion de pasos)}
> z <- my.funcion(3,5)
```

- Las funciones pueden devolver números, vectores, matrices, listas, mensajes o gráficas.

Propiedades de las funciones.

- Una función se puede definir dentro de otra función. Si se define la función f_2 dentro de la función f_1, entonces:
  - Si se le llama a f_2 dentro de f_1, se usará en forma anidada, y

  - La función f_2 no será visible fuera de f_1

- Una función puede ser terminada usando `return`, `stop` o bien mandando un mensaje con `warning` pero continuando la función.
- En muchos casos se obtiene una descripción de la función llamando su nombre (sin paréntesis).
- Un argumento especial de las funciones son tres puntos: . . .. Usualmente se usa para pasar argumentos de una función a otra, y también para tener un número variables de argumentos

Argumentos de funciones.

- Las funciones pueden tener sus argumentos *especificados* o *no especificados* (. . .) cuando la función

```
my.funcion <- function(x,y=1,...){  ********      }
```

- Los argumentos *formales* son los que se usan en la definición de la función, y los *reales* son los que se usan en una llamada a la función.

```
my.funcion(4,color=T)
```

- Hay una serie de reglas que definen cómo se aparejan los argumentos formales y los reales.
- Para conocer los argumentos de una función usamos `args`.

Reglas de argumentos

1. Los argumentos reales especificados de la forma `nombre=valor` donde el nombre es *exactamente* el nombre de un argumento formal, son apareados primero. Si el argumento formal aparece después de . . ., ésta es la única forma en que será apareado.
2. Argumentos especificados de la forma `nombre=valor` para los que hay una correspondencia parcial con un argumento formal, son apareados.
3. Si hay argumentos reales sin nombre, son apareados a los parámetros formales uno por uno en la sucesión.
4. Todos los restantes argumentos reales que no están apareados, formarán parte del argumento formal . . ., si hay uno, y si no ocurre un error.
5. Tener argumentos formales no apareados no es un error.

Control de ejecución: condicionales

- se puede usar: `if (cond)  else (cond)`
- Las condiciones pueden incluir &,|, (componente por componente)o las siguientes que funcionan sobre escalares lógicos: && (evalúa el lado derecho sólo si el izquierdo es verdadero) o || (sólo si el lado izquierdo es falso)
- Otra posibilidad es la función `ifelse` sobre vectores.
- Para sustituir if's anidados, se usa la función `switch`.

Control de ejecución: ciclos

- Las funciones disponibles son `for, while, repeat`.
- `while (condicion.logica) instrucción`. Esta función termina cuando la condición es falsa.
- `for (variable.loop in valores) instrucción`

- `repeat (condición) instrucción`.
- Para salir de los ciclos en cualquier punto, usamos `break`.
- Para saltar a la siguiente iteración, se usa `next`.

Seguimiento de ejecución.

- `traceback()` imprime la pila de llamada de funciones que condujo a un error.
- `debug(fun)` y `undebug(fun)` prende y apaga una bandera para rastrear una función
- `browser()` interrumpe la ejecución de una función en un punto específico para rastrear los pasos siguientes.

Capítulo 3

Aspectos de Regresión clásica

Introducción.

El objetivo de éste capítulo es continuar con el ejemplo introducido en la sección 2.3 para mostrar más aspectos necesarios en el estudio analítico de regresión y extender detalles del modelo. Además se introducirá un conjunto nuevo de datos para resaltar algunos aspectos de la regresión clásica, estos datos se refieren a la demanda trimestral de ciertos productos. Utilizando estos ejemplos se mostrará una extensión del análisis en cuatro direcciones: (1) inferencia, validación y diagnósticos del modelo; (2) extensión a un modelo con la inclusión de factores para analizar resultados en términos anuales; (3) análisis de covarianza y de los submodelos asociados y (4) realización de predicciones utilizando el modelo.

3.1. Validación e inferencia

Antes y después de ajustar un modelo de regresión es necesario verificar si los supuestos de tal modelo son válidos para que el proceso de inferencia tenga validez y utilidad. En esta sección se revisarán los pasos a seguir y se harán los ajustes necesarios, si éstos son

posibles, para poder usar el modelo en el proceso de inferencia.

3.1.1. Validación del modelo

Para hacer un diagnóstico y evaluación del modelo de regresión, se requiere lo siguiente:

1. Conocer previamente los datos y revisar si es posible transformarlos para hacer factibles los supuestos.
2. Verificar si la varianza de los errores es constante.
3. Verificar si no hay falta de ajuste en el modelo vía una prueba de curvatura.
4. Identificar valores extremos (outliers) y/o puntos influenciales, para determinar la sensibilidad del modelo. Esto se hace a través de las distancias de Cook.
5. Verificar la normalidad de los residuales con la finalidad de poder hacer inferencia sobre los parámetros y sobre predicciones.

No hay un orden particular definido para los procedimientos anteriores. De hecho, muchas veces hay que seguir un proceso iterativo para converger a un modelo apropiado para los datos. Aquí se propone un posible camino para el análisis, que creemos, es eficiente.

Primero: ver los datos

Lo primero que se sugiere hacer es una gráfica de matriz de dispersión o scatterplot para darnos una idea de los datos y de sus interrelaciones, qué tan alejados están de una distribución normal y tratar de identificar casos extraños o especiales. En la gráfica 3.1 se muestra una matriz de dispersión de los excesos de rendimiento de varias divisas respecto al euro, en ésta se puede observar que hay un punto correspondiente al caso 59,

que se aleja considerablemente de los datos. Esta observación se aleja en dirección de la media de la distribución de las observaciones, por lo que, en términos de los excesos, la observación no es necesariamente un *outlier* pero puede ser un punto influencial. Sin embargo, en términos de las distribuciones bivariadas de los rendimientos simples, o de las distribuciones marginales, podría aún ser un *outlier*.

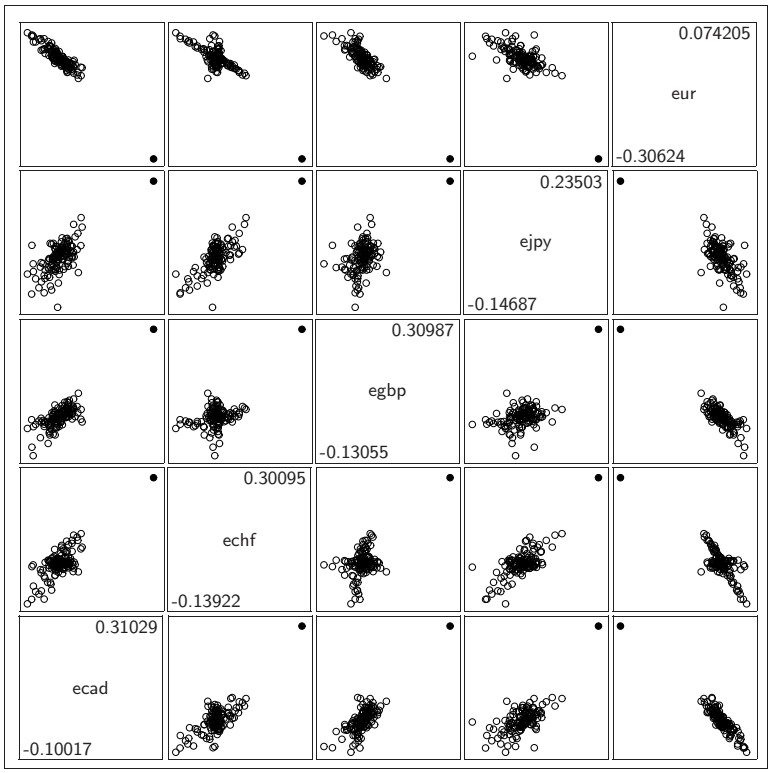


Figura 3.1: Gráfica de los excesos de rendimiento respecto al euro.

La gráfica con este punto no es muy informativa del comportamiento general de los datos, por lo que se volvió a hacer la gráfica después de eliminar dicho punto.

En la figura 3.2 se muestran los datos sin la observación 59. En esta gráfica se puede apreciar mejor el comportamiento general de los excesos de rendimiento respecto al euro. En la mayoría de las gráficas se puede observar que las distribuciones bivariadas siguen un patrón elíptico, con la excepción de las gráficas del franco suizo(*echf*) contra la libra esterlina(*egbp*) y contra el euro (*eur*). Esto se puede explicar considerando que hasta 1998

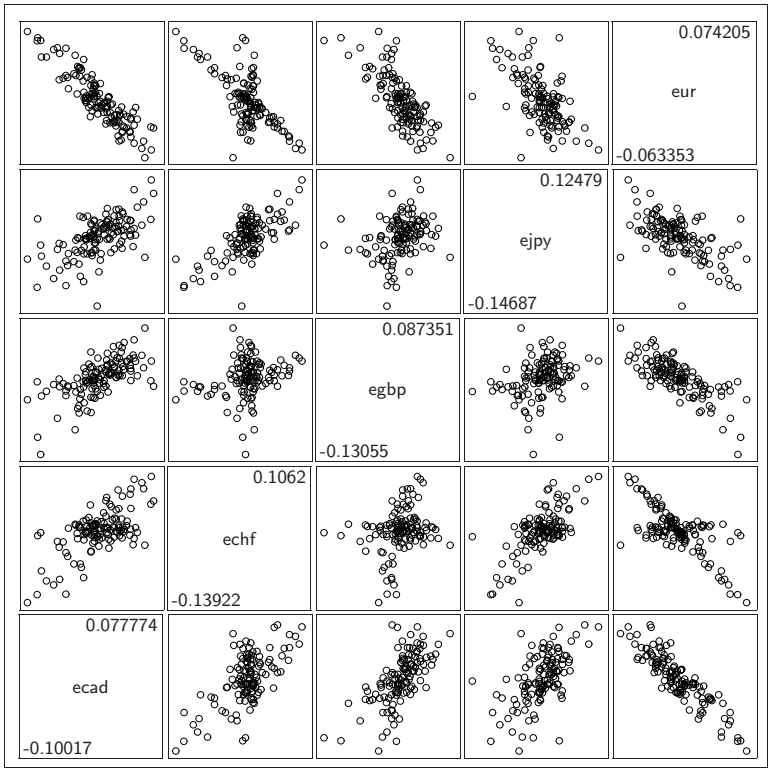


Figura 3.2: Gráfica de los excesos de rendimiento respecto al euro después de eliminar la observación 59.

el euro estaba dominado principalmente por el marco, y la correlación del franco suizo con el marco era positiva, mientras que es negativa con respecto al euro.

Una forma alternativa de ver los datos es en forma de serie de tiempo. La figura 3.3 muestra las series de tiempo de cuatro divisas.

Segundo: análisis empírico de los datos.

Antes de continuar con el modelo de regresión, se hará un análisis de las propiedades empíricas de los rendimientos. El análisis sigue un desarrollo similar al de (Tsa02). Primero se obtienen estimaciones de los primeros cuatro momentos de la distribución de los rendimientos y de los excesos mensuales. A la salida típica de estadísticas sumarias de `Arc` se agregó el cálculo de la asimetría y la kurtosis muestral (tercer y cuarto momento respec-

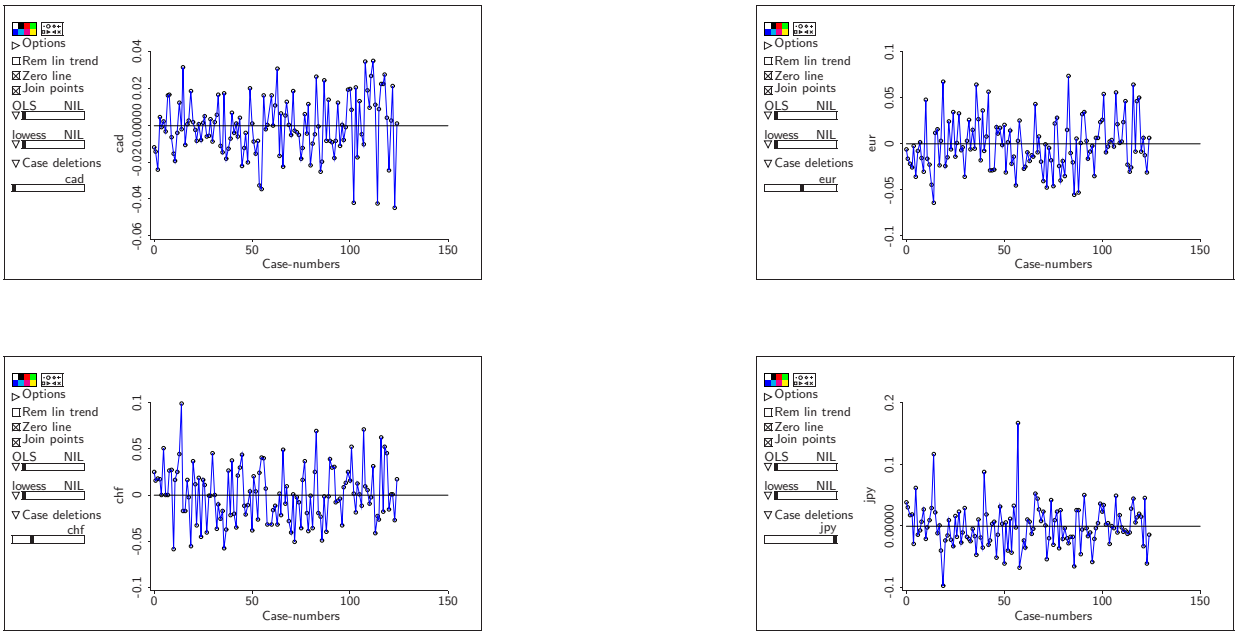


Figura 3.3: Series de tiempo de los rendimientos mensuales de las divisas

tivamente). Los resultados obtenidos se muestran a continuación:

Data set = rend_mensual, Summary Statistics

Variable	N	Average	Std Dev	Minimum	Median	Maximum	Skew	Kurt
cad	125	-0.00058271	0.015952	-0.044765	-0.00086749	0.035092	-0.124749	3.09833
chf	125	0.0013086	0.030219	-0.0581	7.1301E-5	0.099956	0.362267	2.95995
eur	125	-0.0032843	0.039109	-0.30624	-0.0041203	0.074205	-3.47881	29.87420
gbp	125	0.0010777	0.021108	-0.066094	0.00074697	0.058865	-0.127154	3.08716
jpy	125	0.00021789	0.035749	-0.097636	-0.0032066	0.16727	0.968973	6.64895
ecad	125	0.0027016	0.039221	-0.066022	0.0017863	0.3022	3.40677	28.15480
echf	125	0.0045929	0.048568	-0.12275	0.0016479	0.31155	1.91254	15.56460
egbp	125	0.004362	0.042419	-0.13055	0.0040494	0.30987	2.54887	23.58890
ejpy	125	0.0035022	0.057738	-0.16514	-0.0019818	0.38291	2.34431	17.54610

Excepto para los francos suizos, la kurtosis está arriba de 3, indicando que las distribuciones de los rendimientos y de los excesos tienden a tener colas pesadas: hay muchos valores extremos con respecto a la distribución normal. Los excesos con respecto al euro tienden a ser más sesgados y a tener valores extremos más grandes que los rendimientos

de cada divisa. Esto se debe a que los euros mismos tienen valores extremos. Esto sugiere que el supuesto de normalidad para los rendimientos y los excesos puede no ser adecuado. Sin embargo, las cosas cambian significativamente cuando se elimina el caso 59, como se ve en la siguiente tabla:

```
Data set = rend_mensual, Summary Statistics
Deleted cases are
(59)
```

Variable	N	Average	Std Dev	Minimum	Median	Maximum	Skew	Kurt
cad	124	-0.00055489	0.016014	-0.044765	-0.00077855	0.035092	-0.129448	3.07655
chf	124	0.0012763	0.030339	-0.0581	3.5651E-5	0.099956	0.364055	2.93846
eur	124	-0.00084112	0.028103	-0.063353	-0.0039278	0.074205	0.471624	2.95225
gbp	124	0.0010571	0.021192	-0.066094	0.00034219	0.058865	-0.123764	3.06251
jpy	124	-0.00039868	0.03522	-0.097636	-0.0032403	0.16727	0.990861	7.0016
ecad	124	0.00028623	0.028559	-0.066022	0.0015388	0.061484	-0.227057	2.50461
echf	124	0.0021174	0.040073	-0.12275	0.0015624	0.16331	-0.035852	5.83586
egbp	124	0.0018983	0.03239	-0.13055	0.0038363	0.087351	-0.821577	5.32603
ejpy	124	0.00044244	0.046701	-0.16514	-0.0020154	0.18109	0.303442	5.96276

Los estimadores del sesgo y la kurtosis muestran un comportamiento mucho más parecido a la distribución normal quizá con excepción del yen japonés, después de quitar ese caso. Esto sugiere que el supuesto de normalidad puede ser razonable, y en este sentido se supondrá normalidad de las observaciones. El caso 59 será eliminado del análisis ya que es completamente atípico. Una posible explicación de ese caso, es el cambio de la unidad monetaria del marco alemán al euro dado en diciembre de 1998, en donde sufrió un cambio en la tendencia del tipo de cambio respecto al dólar americano.

Las gráficas de las distribuciones marginales se muestran en la figura 3.4. Como se dijo anteriormente, es común suponer que los rendimientos de instrumentos financieros siguen distribuciones lognormales. En las gráficas mostradas, se puede ver en particular que la distribución de los excesos de rendimientos en el caso de las libras tiende a mostrar colas más largas del lado negativo de los rendimientos, que se confirma con una gráfica de probabilidades. Sin embargo, en términos generales, se puede ver que la distribución normal a los excesos de rendimientos se aproxima razonablemente bien. Esto se confirmó

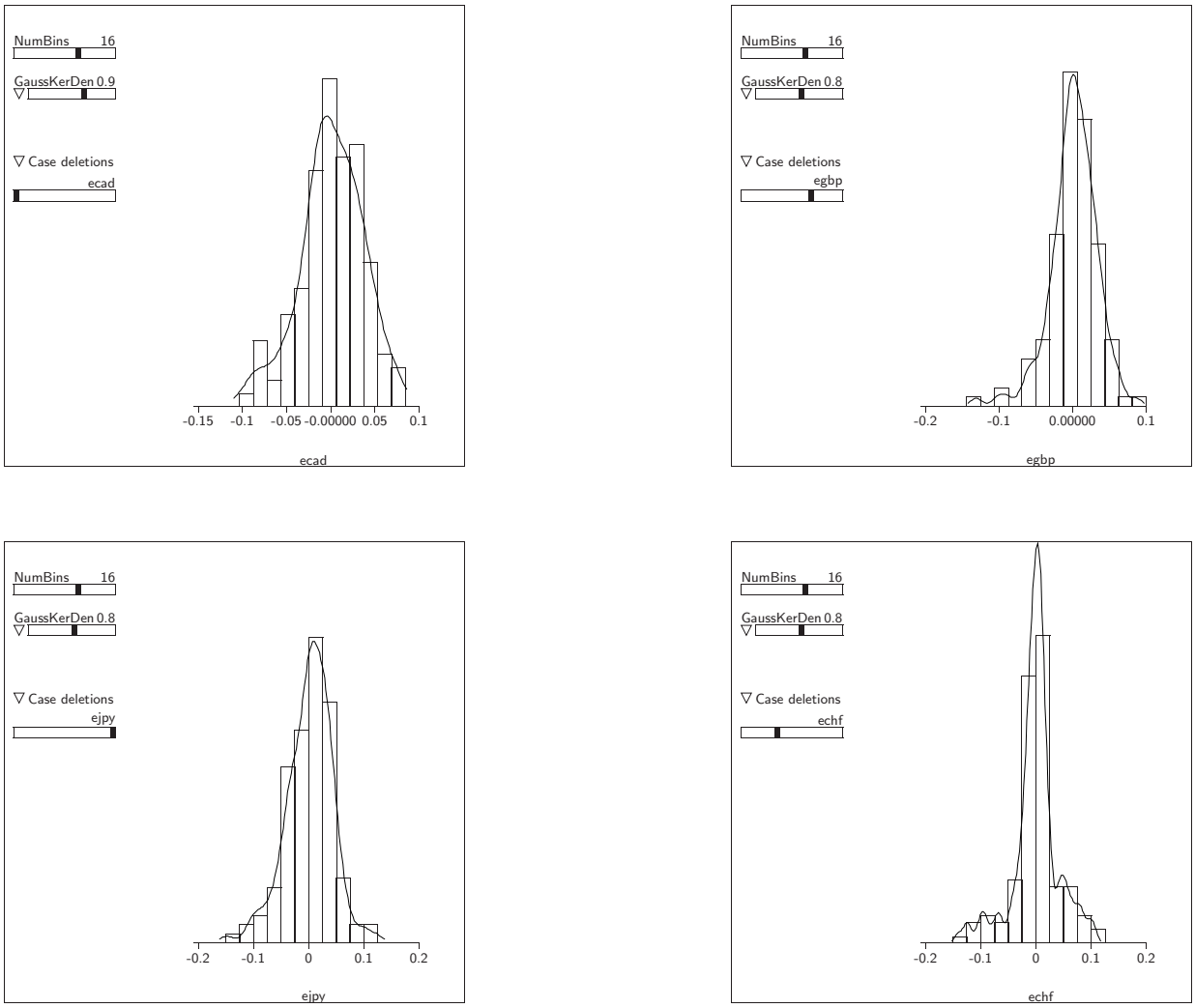


Figura 3.4: Distribuciones marginales de los excesos de rendimiento

con el análisis empírico hecho previamente.

Segundo: transformar

Se pueden buscar transformaciones a nivel marginal o en forma conjunta. Sin embargo, la complejidad puede aumentar dramáticamente cuando se tienen muchos predictores. En un sentido práctico, las transformaciones se pueden buscar en pares. Cuando los datos bivariados no muestran un patrón elíptico, es usual buscar algún tipo de trans-

formación de los predictores de tal forma que la distribución conjunta de las variables sea aproximadamente normal. En este caso esto no sería necesario, ya que los datos muestran un comportamiento aproximado elíptico. Está claro que la precisión y calidad de las inferencias que se hagan serán sensibles a qué tanto se alejen los datos de los supuestos.

Usualmente las transformaciones se realizan sobre variables positivas. En el caso de observaciones negativas a veces es necesario agregar alguna constante para poder realizar la transformación.

Tercero: ajustar modelo y analizar residuales

Una vez que se han aplicado transformaciones a los predictores para obtener distribuciones conjuntas aproximadas a la distribución normal, el siguiente paso es estudiar los residuales de un modelo. El modelo estimado en la sección 2.3 reajustado sin la observación 59 será el modelo de trabajo. El ajuste de este modelo es:

```
Data set = rend_mensual, Name of Fit = L2 Deleted cases are (59)
Normal Regression Kernel mean function = Identity
Response      = eur
Terms         = (ecad echf egbp ejpy)
Coefficient Estimates
Label      Estimate        Std. Error    t-value    p-value
Constant   -0.000146585    0.00119527    -0.123     0.9026
ecad       -0.549475       0.0597823     -9.191     0.0000
echf       -0.0104990      0.0543075     -0.193     0.8470
egbp       -0.253881       0.0686851     -3.696     0.0003
ejpy       -0.0747928      0.0382825     -1.954     0.0531

R Squared:               0.784374
Sigma hat:               0.0132671
Number of cases:             125
Number of cases used:        124
Degrees of freedom:          119

Summary Analysis of Variance Table
Source         df       SS            MS           F    p-value
Regression      4  0.0761944     0.0190486    108.22    0.0000
```

```
Residual      119  0.020946      0.000176017
```

El portafolio estimado es $(cad, chf, gbp, jpy, eur)$ = (54.95 %, 1.05 %, 25.39 %, 7.48 %, 11.13 %). La gráfica de probabilidad de los residuales no se ajusta bien a una linea, muestra una tendencia a una distribución con colas pesadas, como se observa en los excesos de rendimiento. En la figura 3.5 se muestra la gráfica de probabilidad con su envoltura simulada y la observación 59 resaltada.

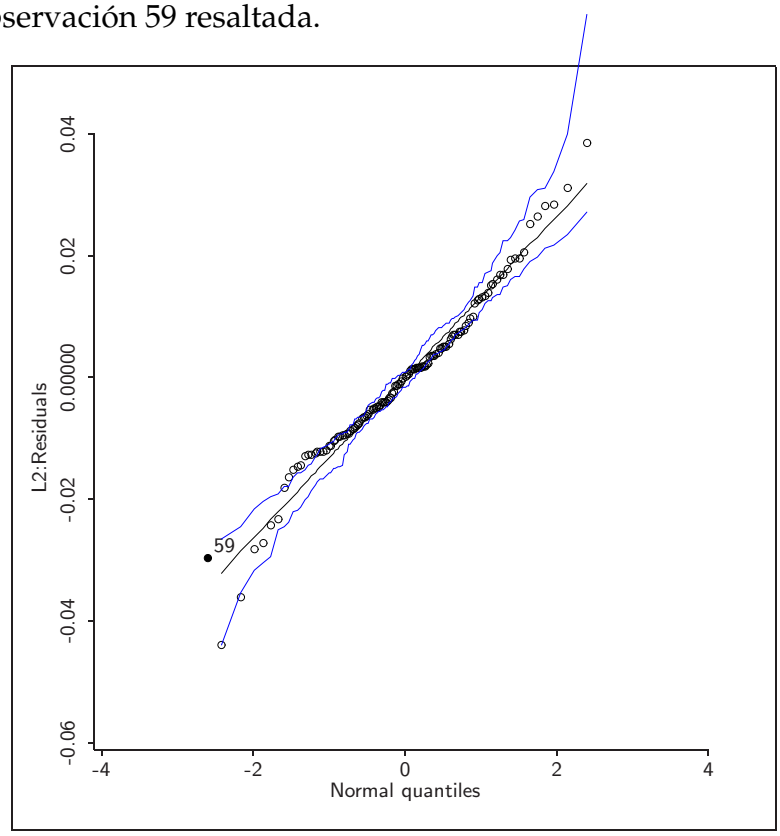


Figura 3.5: Residuales del modelo

Una gráfica de los residuales estandarizados puede mostrar si hay más observaciones que puedan ser valores extremos y otras características, como curvatura y varianza no constante. Para hacer una gráfica de los residuales estandarizados, $r = \frac{e}{\hat{\sigma}}$, hay que agregar una variable. En el menú **rend-mensual** → **Add a variate...** hay que introducir la expresión: `d = (/ (send L1 :Residuals) 0.0132671)`. Después se hace la gráfica de los *Case-numbers* en el eje horizontal y d en el vertical. En las opciones de la gráfica,

se pueden agregar las líneas $y = -2$ y $y = 2$ que usualmente se grafican para mostrar la banda que contiene aproximadamente 95 % de las observaciones. Por ultimo, se pueden mostrar el punto eliminado usando la opción `highlight deleted cases` del menú `Case deletions`. La gráfica se muestra en la figura 3.6.

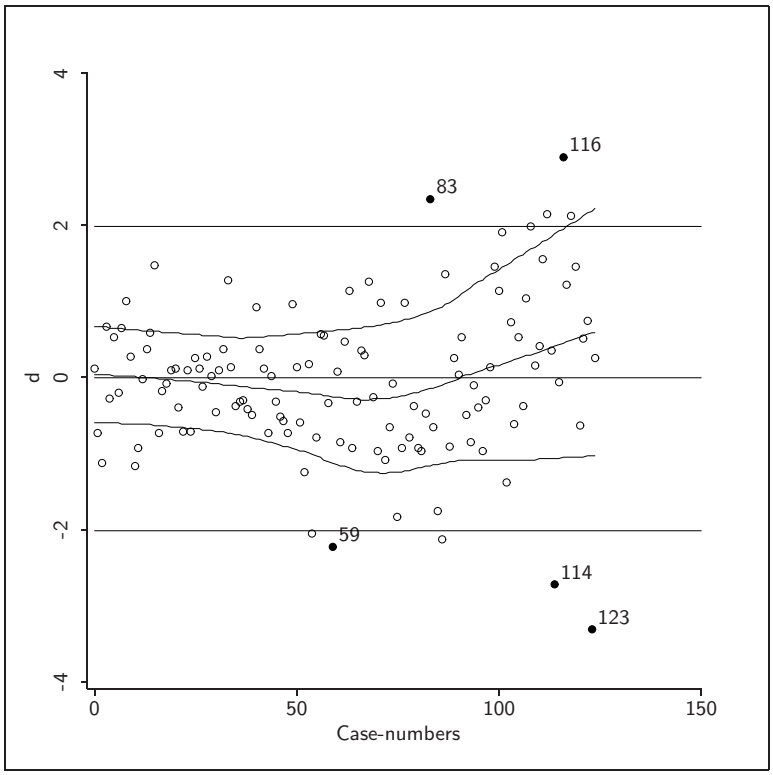


Figura 3.6: Residuales estandarizados del modelo.

La gráfica de los residuales parece indicar que (a) hay curvatura, (b) la varianza no parece constante, (c) hay otros puntos que parecen ser *outliers*, en particular la observación 123 y (d) la observación 59 no parece ser un *outlier*.

Para verificar si la curvatura es significativa, se aplica la prueba de Tukey de no aditividad. El valor de la estadística de esta prueba y su p-value puede obtenerse para cualquier combinación lineal de los predictores. Se obtiene vía el menú **L1** $\rightarrow$ **Residual Plots...** Esto da una gráfica multipanel con los valores. En todos los casos, los p-values parecen indicar que no hay curvatura significativa. Entonces podemos suponer que no

hay falta de ajuste en los datos.

La prueba de varianza no constante se obtiene modelando a la varianza como una función de los predictores (usualmente la exponencial de una combinación lineal arbitraria) y aplicando una prueba de score (score test) para obtener la estadística de prueba. En `Arc`, esta prueba se obtiene de **L1** $\rightarrow$ **Non constant variance plot** en donde se grafíca la combinación lineal considerada contra la raíz cuadrada del valor absoluto de los residuales. Considerando diferentes combinaciones lineales, se obtiene que el p-value=0.06 que es el más pequeño es en la dirección de la media. Como el p-value está muy cerca del nivel de referencia, se podría considerar como solución (a) transformar la variable de respuesta, (b) usar mínimos cuadrados ponderados o (c) usar un modelo lineal generalizado. Ninguno de estos enfoques parecen apropiados para estos datos.

Para verificar puntos influenciales, se calculan las distancias de Cook D_i. Se tiene que $D_{59} = 0.49605$ y es la más alta de todas. De acuerdo a Cook y Weisberg (CW99), los puntos con distancias mayores a 0.5 son candidatos a puntos influenciales significativos. Aunque esta distancia no es una prueba formal, confirma que es muy probable que esta observación podría tener un efecto significativo en el ajuste del modelo de regresión, y por lo tanto, en los pesos obtenidos por divisa. Este punto es un ejemplo de un punto influencial que no es *outlier*. La decisión de dejar esta observación fuera del modelo es razonable, esto lo podemos observar en la magnitud en el cambio de los coeficientes dentro del portafolio de inversión, como se muestra en la siguiente tabla.

Modelo	$ecad$	$echf$	$egbp$	$ejpy$	eur
Con observación influencial	57.91 %	0.69 %	27.01 %	7.97 %	6.42 %
Sin observación influencial	54.95 %	1.05 %	25.39 %	7.48 %	11.13 %
Variación	2.96 %	0.36 %	1.62 %	0.49 %	-4.71 %

El resto de las distancias de Cook son mucho menores a 0.5 y se muestran en la figura 3.7. Se concluye que los posibles *outliers* presentes en los datos no son influenciales.

Como conclusión del análisis de los residuales, se puede decir que el modelo es un ajuste muy razonable y que el modelo, aunque presenta desviaciones de los supuestos,

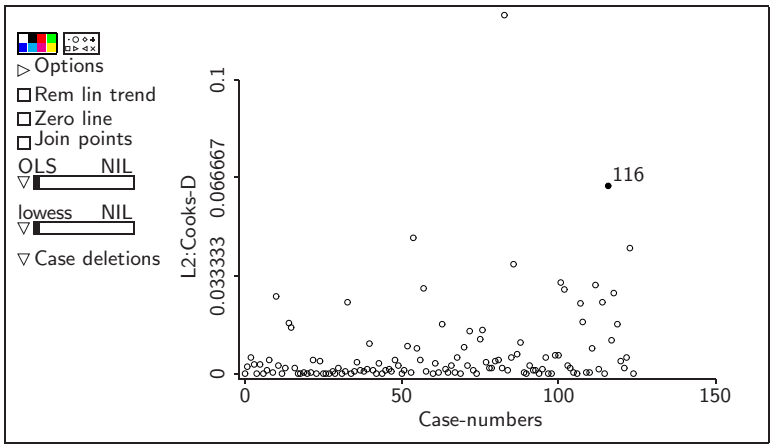


Figura 3.7: Distancias de Cook obtenidas del modelo El punto más alto es la observación 59.

estas desviaciones no son significativas. Esto valida el modelo considerado.

3.1.2. Inferencia

En esta sección se obtendrán intervalos de confianza para los coeficientes del modelo. En el contexto del problema que estamos considerando, esto es equivalente a determinar los rangos de inversión en cada una de las divisas en factibles sin cambiar el modelo.

La distribución conjunta de $\hat{\boldsymbol{\beta}}$ es

$$\hat{\boldsymbol{\beta}} \sim \mathcal{N}_5 \left(\boldsymbol{\beta}, \sigma^2 (\mathbf{R}_e' \mathbf{R}_e)^{-1} \right)$$

La matriz de covarianzas de los coeficientes es

$$\hat{\sigma}^2 (\mathbf{R}_e' \mathbf{R}_e)^{-1} = \begin{pmatrix} 1.429 & & & & \\ 2.107 & 3,573.90 & & & \\ -1.665 & 189.96 & 0.003 & & \\ -3.658 & -1,704.80 & -0.002 & 0.005 & \\ 1.566 & -670.07 & -0.001 & -0.000 & 0.001 \end{pmatrix} \times 10^{-6}$$

Con esta matriz podemos encontrar los intervalos de confianza para diferentes pesos, así como para diferentes combinaciones lineales de los coeficientes en general.

3.2. Introducción de factores

Los predictores categóricos incrementan el acervo de modelos que se pueden estudiar utilizando la metodología de regresión. Las variables categóricas tienen dos o mas niveles, por ejemplo, la variable *sexo* tiene dos niveles: femenino y masculino, la variable *turno* tiene 3 niveles: diurno, mixto y nocturno. En ocasiones es conveniente categorizar una variable continua agrupando rangos de valores como niveles; por ejemplo, la edad de una persona puede utilizarse para clasificar a una persona en cuatro posibles niveles: infante, joven, maduro y viejo.

Los predictores categóricos sirven para distinguir poblaciones sobre variables cualitativas. Para indicar los niveles de un predictor categórico o variable categórica se tiene que utilizar una variable numérica llamada *indicadora* o *dummy* y al conjunto de éstas que representan los niveles de una variable categórica se le llama *factor*.

Supongamos que C es una variable categórica de dos niveles, asignamos una variable indicadora v_1 con dos niveles $\{1, 0\}$. Para describir un predictor categórico con dos niveles sólo se requiere de una variable indicadora. Para describir un predictor con tres niveles se requieren de dos variables indicadoras v_2 y v_3 como se muestra en la siguiente tabla

Nivel	v_2	v_3
1	0	0
2	1	0
3	0	1

Por convención las variables v_i indica el nivel i. Análogamente para un predictor categórico C con l niveles, requiere $l - 1$ variables indicadoras $v_2, v_3, \ldots, v_l$. En cualquier observación, a lo más una de las variables indicadoras es 1 y en el resto 0. Si para una

observación, todas las variables que forman el *factor* son cero entonces C es igual a su primer nivel. Si $v_j = 1$ entonces C es iguala su j-ésimo nivel.

En `Arc` es sencillo convertir un predictor categórico en *factores*, sólo se utiliza la opción `Make factors` del submenú de datos creado al cargar un archivo de datos. El primer nivel siempre se tomará como la *base* para medir el efecto de las otras variables. `Arc` asigna al factor el mismo nombre de la variable con el prefijo $\{F\}$, i.e. El factor $\{F\}C$ representa la colección de variables indicadoras que describen al predictor categórico C y las variables indicadoras se representan en `Arc` como $\{F\}C[i]$ para el i-ésimo nivel. Esta no es la única forma de definir un *factor*, en las otras formas, la manera de representar los niveles cambia la interpretación de los coeficientes en el modelo de regresión.

Por ejemplo, si Ds es la variable categórica de días hábiles de la semana (donde lunes es el día 1, martes el 2, ... y viernes el 5). Creamos el factor $\{F\}Ds$ obteniendo los niveles de la siguiente forma

Día	Nivel	v_2	v_3	v_4	v_5
lunes	1	0	0	0	0
martes	2	1	0	0	0
miércoles	3	0	1	0	0
jueves	4	0	0	1	0
viernes	5	0	0	0	1

En las siguiente sección usaremos este factor en conjunción con otros predictores.

3.3. Análisis de covarianza y submodelos

El análisis de covarianza (ANCOVA) es el estudio de problemas de regresión que incluyen predictores (o términos) tanto de tipo continuo como categóricos. Para la explicación del tema se tomarán datos diarios de la demanda de un producto en donde verificaremos la influencia temporal, tanto del día de la semana como del mes, sobre la demanda (ingreso en miles de pesos) de cierto producto. Para esto consideraremos infor-

mación diaria de los años 2003 y 2004. Es conveniente aclarar que la demanda negativa se refiere a devolución del producto.

Aunque el análisis más apropiado de este tipo de datos sería con modelos de series de tiempo, es también útil considerar un modelo de regresión lineal simple para identificar las tendencias y significancia de las relaciones temporales de las variables.

Este ejemplo tiene dos predictores categóricos: el día de la semana *Ds* con cinco niveles o categorías y el mes del año (*M*) con doce niveles, y la respuesta continua es la demanda del producto (*dem*). Utilizamos el factor creado anteriormente de la variable *Ds*; esto crea las variables indicadoras $\{F\}Ds[2]$, $\{F\}Ds[3]$, $\{F\}Ds[4]$ y $\{F\}Ds[5]$. Nos referiremos a todas las variables indicadoras que conforman a Ds, como el factor $\{F\}Ds$.

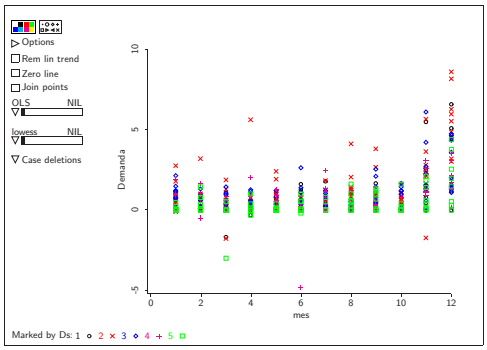
Consideremos primero la regresión de *dem* respecto a Ds y M o en nuestra notación, queremos entender

$$dem|(Ds, M).$$

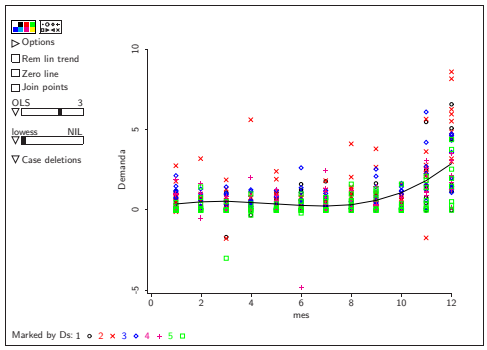
Una gráfica de estos datos se muestra en la Figura 3.8(a). Nótese que el factor *Ds* se agrega como una variable para marcar los niveles, que al seleccionar una marca, en la gráfica únicamente se muestran los puntos correspondientes a ese nivel. Hay cuatro casos a considerar:

1. un modelo de regresión lineal simple para todos los rangos,
2. líneas paralelas (misma forma, diferentes ordenadas),
3. líneas con una ordenada al origen común (formas diferentes, misma ordenada al origen), y
4. líneas diferentes para cada rango (una línea por categoría).

El modelo más general es el caso 4 y los casos 1,2 y 3 son submodelos de éste. El caso 1 es un submodelo de los casos 2 y 3; y los casos 2 y 3 no están relacionados.



a. Datos con variable para marcar



b. Modelo 1

Figura 3.8: Demanda temporal

3.3.1. Modelo de regresión lineal simple: modelo 1

El modelo de regresión lineal más simple para este conjunto de datos es

$$dem = \beta_0 + \beta_1 M,$$

que esencialmente ignora la variable categórica. En `Arc` se puede ajustar el modelo con respuesta *dem* y predictor *M* con un polinomio de grado 3 que representa una tendencia como función del tiempo (así lo elegimos ya que hay mucha similitud con un ajuste *lowess* que es una estimación no paramétrica discutida en el capítulo anterior), por lo que el modelo de regresión lineal queda de la siguiente forma

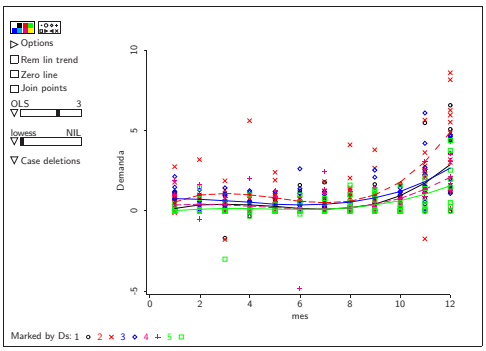
$$dem = \beta_0 + \beta_1 M + \beta_2 M^2 + \beta_3 M^3,$$

La gráfica se muestra en la Figura 3.9(a)

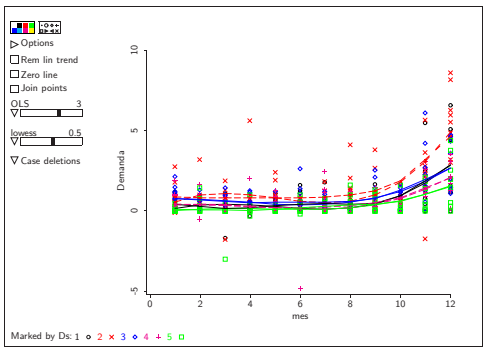
Las variables adicionales M^2 y M^3 son transformaciones potencia de la variable M. Estas se crean con la opción se utiliza la opción `Transform...` del submenu creado al cargar un archivo de datos. E incluimos estas variables en el modelo, obteniendo

```
Data set = egrntodiasem, Name of Fit = L1
Normal Regression
Kernel mean function = Identity
```

```
Response      = dem
Terms         = (M M^2 M^3)
Coefficient Estimates
Label      Estimate        Std. Error    t-value    p-value
Constant   0.00408587      0.265466        0.015     0.9877
  M        0.477491        0.169636        2.815     0.0051
  M^2     -0.123242        0.0296875      -4.151     0.0000
  M^3      0.00862501      0.00150318      5.738     0.0000

R Squared:                0.344922
Sigma hat:                1.05757
Number of cases:             508
Degrees of freedom:          504

Summary Analysis of Variance Table
Source         df       SS            MS           F     p-value
Regression      3    296.81        98.9366      88.46     0.0000
Residual      504    563.703       1.11846
 Lack of fit    8    32.9029       4.11287       3.84     0.0002
 Pure Error   496      530.8       1.07016
```



a. Ajuste polinomial.



b. comparado con ajuste *lowess*

Figura 3.9: Ajuste polinomial de grado 3

3.3.2. Líneas paralelas: modelo 2

La ecuación para este modelo es de la forma:

$$dem = \beta_0 + \beta_1 M + \beta_2 M^2 + \beta_3 M^3 + \sum_{i=1}^{4} \beta_{4i} Ds[i]$$

o bien se puede escribir en forma corta como:

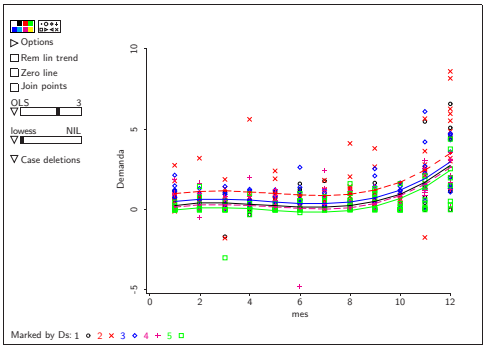
$$dem = \beta_0 + \beta_1 M + \beta_2 M^2 + \beta_3 M^3 + \beta_4 \{F\} Ds.$$

Ahora incluimos como predictor a la variable factor $\{F\}Ds$ y para verlo en una gráfica seleccionamos la opción "`Fit by marks-parallel`" en el menú `OLS` de la gráfica. La salida de este modelo se da a continuación y la Figura 3.10(a) muestra la gráfica.

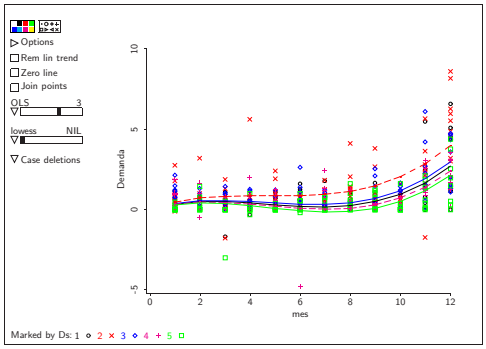
```
Data set = egrntodiasem, Name of Fit = L2
Normal Regression
Kernel mean function = Identity
Response      = dem
Terms         = (M M^2 M^3 {F}Ds)
Coefficient Estimates
Label          Estimate        Std. Error    t-value    p-value
Constant      -0.0566302       0.266886       -0.212     0.8320
  M            0.435692        0.160446        2.716     0.0068
  M^2         -0.115821        0.0280798      -4.125     0.0000
  M^3          0.00825220      0.00142177      5.804     0.0000
{F}Ds[2]       0.740602        0.138916        5.331     0.0000
{F}Ds[3]       0.234056        0.138922        1.685     0.0927
{F}Ds[4]      -0.109401        0.141094       -0.775     0.4385
{F}Ds[5]      -0.288441        0.139681       -2.065     0.0394

R Squared:               0.419749
Sigma hat:               0.999313
Number of cases:              508
Degrees of freedom:           500

Summary Analysis of Variance Table
Source         df       SS            MS           F    p-value
Regression      7   361.199       51.5999      51.67    0.0000
Residual      500   499.313      0.998626
```

```
Lack of fit    52    107.027       2.05821       2.35     0.0000
Pure Error    448    392.286      0.875638
```

Hay cinco líneas: la linea para el nivel 1 es $-0.0566 + 0.4357M - 0.1158M^2 + 0.0082M^3$ que para $M = 1$ tenemos $dem = 0.2715$, la línea para el rango 2 en $M = 1$ se incrementa en 0.7406 quedando en 1.0121, análogamente para los otros rangos en $M = 1$ son $0.5055, 0.1621$ y -0.0169, respectivamente. Es decir, la interpretación de los coeficientes β_{4i} de las variables indicadoras $FDs[i]$ significan un desplazamiento del polinomio ajustado para el nivel 1 en dirección y magnitud del coeficiente, como se muestra en la figura 3.10(a).



a. Líneas paralelas.



b. Ordenada a un origen común

Figura 3.10: Ajustes polinomiales paralelos y con igual intercepto

3.3.3. Líneas con ordenada al origen común: modelo 3

La ecuación de este modelo tiene la forma:

$$dem = \beta_0 + \beta_1 M + \beta_2 M^2 + \beta_3 M^3 + \sum_{i=1}^{3} \beta_{4i} M^i \{F\} Ds.$$

Ahora se incluye como predictor la interacción de *{F}Ds* y las variables potencia de *M* y en la gráfica se debe seleccionar "`Fit by marks-equal intercept`" en el menú

OLS de la gráfica. Para generar la interacción, ir al menú de los datos y seleccionar el submenú "`Make interactions...`", obteniendo los términos $M*\{F\}Ds$, $M^2*\{F\}Ds$ y $M^3*\{F\}Ds$. La salida se muestra abajo y la Figura 3.10(b) muestra la gráfica.

```
Data set = egrntodiasem, Name of Fit = L3
Normal Regression
Kernel mean function = Identity
Response      = dem
Terms         = (M M^2 M^3 M*{F}Ds M^2*{F}Ds M^3*{F}Ds)
Coefficient Estimates
Label                Estimate        Std. Error    t-value    p-value
Constant             0.0499685       0.242806        0.206     0.8370
  M                  0.396861        0.192222        2.065     0.0395
  M^2               -0.114274        0.0393653      -2.903     0.0039
  M^3                0.00839609      0.00218428      3.844     0.0001
M.{F}Ds[2]           0.478727        0.184500        2.595     0.0097
M.{F}Ds[3]           0.0886788       0.184829        0.480     0.6316
M.{F}Ds[4]          -0.0785880       0.188917       -0.416     0.6776
M.{F}Ds[5]          -0.283413        0.187576       -1.511     0.1315
M^2.{F}Ds[2]        -0.107791        0.0459583      -2.345     0.0194
M^2.{F}Ds[3]        -0.00254718      0.0459004      -0.055     0.9558
M^2.{F}Ds[4]         0.0263042       0.0468805       0.561     0.5750
M^2.{F}Ds[5]         0.0778106       0.0465475       1.672     0.0952
M^3.{F}Ds[2]         0.00682042      0.00272555      2.502     0.0127
M^3.{F}Ds[3]        -0.000461618     0.00271558     -0.170     0.8651
M^3.{F}Ds[4]        -0.00206091      0.00277739     -0.742     0.4584
M^3.{F}Ds[5]        -0.00527263      0.00275969     -1.911     0.0566

R Squared:                0.46601
Sigma hat:                0.966413
Number of cases:             508
Degrees of freedom:          492

Summary Analysis of Variance Table
Source         df       SS            MS           F    p-value
Regression     15   401.007       26.7338      28.62    0.0000
Residual      492   459.505      0.933954
 Lack of fit   44   67.2193       1.52771       1.74    0.0030
 Pure Error   448   392.286      0.875638
```

Es importante notar que únicamente para el nivel 2 el ajuste es bueno, ya que en los

términos donde aparece la variable indicadora $\{F\}Ds[2]$ son significativos.

3.3.4. Modelo general: modelo 4

Ahora la ecuación para este modelo incluye cuatro términos:

$$dem = \beta_0 + \beta_1 M + \beta_2 M^2 + \beta_3 M^3 + \beta_4 \{F\}Ds + \sum_{i=1}^{3} \beta_{5i} M^i \{F\}Ds.$$

En la gráfica, seleccionen la opción "`Fit by marks-general`". La salida se muestra abajo y la Figura 3.11 muestra la gráfica con los ajustes correspondientes.

```
Data set = egrntodiasem, Name of Fit = L4
Normal Regression
Kernel mean function = Identity
Response      = dem
Terms         = (M M^2 M^3 {F}Ds M*{F}Ds M^2*{F}Ds M^3*{F}Ds)
Coefficient Estimates
Label               Estimate        Std. Error    t-value    p-value
Constant           -0.317013        0.553782       -0.572     0.5673
  M                 0.611266        0.348646        1.753     0.0802
  M^2              -0.148225        0.0606049      -2.446     0.0148
  M^3               0.00997355      0.00305926      3.260     0.0012
{F}Ds[2]            0.0781491       0.784122        0.100     0.9207
{F}Ds[3]            1.04933         0.783997        1.338     0.1814
{F}Ds[4]            0.575078        0.774045        0.743     0.4579
{F}Ds[5]            0.154889        0.757667        0.204     0.8381
M.{F}Ds[2]          0.433491        0.494505        0.877     0.3811
M.{F}Ds[3]         -0.520729        0.492083       -1.058     0.2905
M.{F}Ds[4]         -0.417408        0.494983       -0.843     0.3995
M.{F}Ds[5]         -0.369257        0.488038       -0.757     0.4496
M^2.{F}Ds[2]       -0.100683        0.0860519      -1.170     0.2426
M^2.{F}Ds[3]        0.0935676       0.0854043       1.096     0.2738
M^2.{F}Ds[4]        0.0801710       0.0866452       0.925     0.3553
M^2.{F}Ds[5]        0.0910582       0.0856986       1.063     0.2885
M^3.{F}Ds[2]        0.00649170      0.00434480      1.494     0.1358
M^3.{F}Ds[3]       -0.00491297      0.00430411     -1.141     0.2542
M^3.{F}Ds[4]       -0.00456992      0.00438889     -1.041     0.2983
```

```
M^3.{F}Ds[5]        -0.00587774      0.00434971     -1.351      0.1772

R Squared:                0.468773
Sigma hat:                0.967852
Number of cases:               508
Degrees of freedom:            488

Summary Analysis of Variance Table
Source          df       SS             MS            F     p-value
Regression      19    403.385        21.2308       22.66     0.0000
Residual       488    457.128       0.936737
 Lack of fit    40    64.8419        1.62105       1.85     0.0017
 Pure Error    448    392.286       0.875638
```

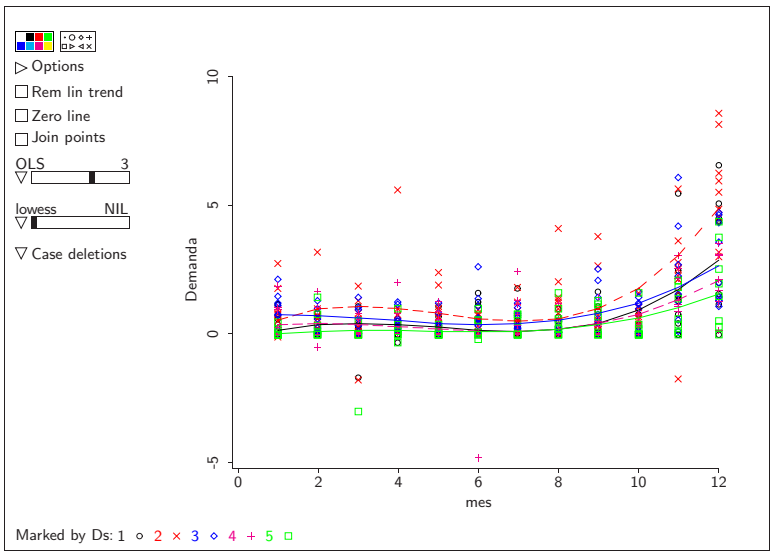


Figura 3.11: Caso general: polinomios diferentes de grado 3.

La salida de regresión muestra que solo las potencias de M pueden ser significativas, M^2 y M^3 al 5 % y además M a un 10 % de significancia. Los factores y las respectivas interacciones de los factores con el componente M no son significativas. Esto indica que no hay un efecto por día de la semana que afecte el comportamiento de la tendencia. Un modelo más simple que este modelo sería suficiente para explicar la variación en la respuesta. En la siguiente sección se discutirá cómo se pueden comparar diferentes

modelos para hacer una selección apropiada.

3.4. Comparando submodelos de un modelo base

El enfoque general para comparar submodelos de un modelo base es usar la prueba F que resulta de dividir las sumas de cuadrados de los errores o residuales de dos modelos: suponemos que la hipótesis nula es un submodelo del de la hipótesis alternativa en el sentido de que cada término que aparece en el modelo de la hipótesis nula también aparece en el de la alternativa. En este caso se dice que los modelos deben estar *anidados*. Por ejemplo, para comparar los modelos 2 y 4, se establecen las hipótesis del siguiente modo:

$$NH: \quad dem = \beta_0 + \sum_{i=1}^{3} \beta_i M^i + \beta_4 \{F\} Ds$$
$$AH: \quad dem = \beta_0 + \sum_{i=1}^{3} \beta_i M^i + \beta_4 \{F\} Ds + \sum_{i=1}^{3} \beta_{5i} M^i \{F\} Ds$$

Esto es equivalente a la hipótesis de que
$NH : \beta_{5i} = 0$, para $i \in \{1, 2, 3\}$. Calculamos la estadística de prueba F como (recordar que la notación es $RSS = SS_{res}$ la suma de cuadrados de los residuales y gl los grados de libertad):

$$F = \frac{\frac{RSS_{NH} - RSS_{AH}}{gl_{NH} - gl_{AH}}}{\hat{\sigma}^2_{AH}}$$

Hay que recordar que $\frac{RSS_{AH}}{gl_{AH}} = \hat{\sigma}^2_{AH}$. Esta estadística tiene distribución F con $(gl_{NH} - gl_{AH})$ grados de libertad en el numerador y gl_{AH} grados de libertad en el denominador. Se pueden obtener estos números de las salidas dadas de los modelos 2 y 4:

$RSS_{NH} = 499.313$, $RSS_{AH} = 457.128$, $gl_{NH} = 500$, $gl_{AH} = 488$ y $\hat{\sigma}^2_{AH} = 0.9367$. De este modo

$$F = \frac{(499.313 - 457.128)/(500 - 488)}{0.9367} = 3.7528$$

con 12 y 488 gl. Esto da un p-value=0.00001848, lo que dice que el modelo general (modelo 4) explica mejor que el modelo 2 que tiene líneas paralelas.

Se puede obtener un resumen de las cantidades necesarias para hacer las pruebas de F de un modelo base y los submodelos correspondientes ejecutando el submenú "`Examine Submodels...`" del menú del modelo en consideración en la hipótesis alternativa, en este caso el modelo 4 .

```
Data set = egrntodiasem, Name of Fit = L4
Normal Regression
Kernel mean function = Identity
Response      = dem
Terms         = (M M^2 M^3 {F}Ds M*{F}Ds M^2*{F}Ds M^3*{F}Ds)
Sequential Analysis of Variance
All fits include an intercept.
Base model = (Ones M M^2 M^3 {F}Ds)
                          Total                    Change
Predictor            df   RSS          |    df  RSS           MS
Base Model          500   499.313      |
M*{F}Ds             496   482.639      |     4  16.6739       4.16848
M^2*{F}Ds           492   467.563      |     4  15.0761       3.76903
M^3*{F}Ds           488   457.128      |     4  10.4352       2.60880
Residual                               |   488  457.128       0.936737
```

3.5. Predicción

En esta sección consideraremos un ejemplo que utiliza un modelo de regresión lineal para hacer predicción sobre la variable de respuesta. Para tal fin utilizaremos un ejemplo diferente. Los ejemplos que hemos utilizado no son apropiados para hacer una predicción

de una nueva observación. En el ejemplo de portafolios, la variable de respuesta es una variable artificial, y lo único que es importante en ese modelo es el vector de coeficientes estimados. En el ejemplo de la demanda, una predicción en realidad sería un pronóstico, y hay mejores modelos para hacer pronósticos que sólo un modelo de regresión, en donde lo que se haría es extrapolar el modelo, con los peligros que esto representa.

Para un ejemplo sobre predicción, consideraremos un ejemplo de datos utilizados por los sociólogos. Duncan (Dun61) realizó un estudio observacional para establecer la relación entre el prestigio de una ocupación con la educación, el ingreso y el porcentaje de mujeres en cada ocupación. Fox (Fox97) sugiere actualizar los datos de Duncan con datos del censo canadiense de 1970, que son los datos que serán utilizados en el ejemplo que se desarrollará aquí.

Los datos tienen la siguiente descripción:

```
Arc 1.06, rev July 2004, Sun Apr 3, 2005, 18:08:03.  Data set name: prestigio
1971 Canadian Occupational Prestige Data
Source: Census of Canada, 1971, Volume 3, Part 6, pp. 19-1--19-21
C. Moore, Departments of Sociology, York University and
University of Victoria.
Name           Type    n    Info
codigo         Variate 102  Canadian Census occupational code
educacion      Variate 102  Average education of incumbents, years
ingreso        Variate 102  Average income of incumbents, dollars
mujeres        Variate 102  Percent of incumbents who are women
prestigio      Variate 102  Pineo-Porter prestige score for occupation
tipo_ocupacion Text    98   Type of occupation: prof =  professional and technical
                                                wc   =  white collar
                                                bc   =  blue collar
                                                ?    =  missing (not classified)
titulo         Text    102  Occupational title
```

El ajuste de un modelo en las variables que se indicaron en el estudio original tiene como ecuación ajustada:

$$\hat{prestigio} = -6.794 + 4.187\textit{educacion} + 0.001314\textit{ingreso} - 0.008905\textit{percen.mujeres}$$

La salida de *Arc* del ajuste se muestra a continuación. Todos los coeficientes parecen ser significativos excepto la variable que representa el porcentaje de mujeres en esa ocupación. Para interpretar los coeficientes del modelo estimado, se requiere entender las unidades en los que están medidos. Los indices de prestigio están medidos en una escala arbitraria, y tiene un rango de valores entre 14.8 y 87.2 para 102 ocupaciones diferentes. la dispersión del índice entre ocupaciones es de cerca de 24.4 puntos.

La educación está medida en años, y por lo tanto, el impacto de ésta en la educación es considerable, un poco más de cuatro puntos en promedio por cada año de educación manteniendo el ingreso y la composición de género constantes.Del mismo modo, el efecto del ingreso también es sustancial, es de un punto por cada 1000 dólares canadienses en promedio.

```
Data set = prestigio, Name of Fit = L1
4 cases are missing at least one value.
Normal Regression
Kernel mean function = Identity
Response      = prestigio
Terms         = (educacion ingreso mujeres)
Coefficient Estimates
Label       Estimate        Std. Error    t-value    p-value
Constant   -6.79433         3.23909        -2.098     0.0385
educacion   4.18664         0.388701       10.771     0.0000
ingreso     0.00131356      0.000277781     4.729     0.0000
mujeres    -0.00890516      0.0304071      -0.293     0.7702

R Squared:               0.798177
Sigma hat:                7.84647
Number of cases:             102
Degrees of freedom:           98

Summary Analysis of Variance Table
Source         df       SS            MS           F    p-value
Regression      3   23861.9       7953.95     129.19    0.0000
Residual       98   6033.57        61.567
 Lack of fit   97   5987.49       61.7267       1.34    0.6103
 Pure Error     1     46.08         46.08
```

Para predecir el índice de prestigio de una ocupación con un ingreso promedio de 12,000 cad, una educación promedio de 13 años y con una composición que incluye a un 50 % de mujeres, hacemos lo siguiente. Una vez ajustado el modelo, se genera un menú L1. Se selecciona el submenú "`Prediction...`" del menú L1, y en el diálogo que aparece se selecciona la opción "Prediction" y se introducen los valores de los predictores necesarios para hacer la predicción. Se obtiene entonces la siguiente salida:

```
 Data set = prestigio, Name of Fit = L1
4 cases are missing at least one value.
Normal Regression
Kernel mean function = Identity
Response      = prestigio
Terms         = (educacion ingreso mujeres)
Term values   = (13 12000 50)
Prediction = 62.9494, se(pred) = 8.01997, weight = 1
Leverage = 0.0447, Max(h_i) = 0.3422
Estimated population mean value  = 62.9494, se = 1.65916
```

El valor estimado de la predicción es el mismo que el valor de la superficie estimada en el vector de predictors $\mathbf{x}_0 = (1, 13, 12000, 50)$ (incluyendo el vector de constantes en la matriz de diseño). Entonces el índice de prestigio estimado es

$$\textit{prestigio}|(\textit{educacion} = 13, \textit{ingreso} = 12,000, \textit{mujeres} = 50) = 62.9494$$

La información proporcionada por `Arc` también nos da el error estándar para estimar el valor promedio de la respuesta y el error estándar para estimar una nueva observación. Éste último es el error estándar de predicción, que debe ser un mayor al de estimar un valor promedio ya que el error agrega incertidumbre. Con estos valores, se puede calcular un intervalo de confianza para la predicción. Por ejemplo, un *intervalo de predicción* de 95 % de confianza tiene la forma

$$\text{estimado} \pm t_{n-k,.975} \times se_{pred}$$

donde $t_{n-k,.975}$ es un cuantíl o percentíl de una distribución t con $n-k$ grados de libertad y donde k es el número de términos en el modelo. Este número se puede calcular ya sea usando un menú en Arc, o tecleando:

```
>(t-quant .975 98)
1.98447
```

Así que el intervalo es $62.9494 \pm 1.98447 \times 8.01997$ o $(47.034, 78.8648)$, dando un intervalo de predicción muy amplio, indicando la gran variabilidad que tienen los datos.

Capítulo 4

Reducción de dimensión en regresión.

4.1. Introducción

En un contexto muy amplio, el análisis de regresión se entiende como el estudio de la distribución condicional de una *variable de respuesta* y dados los valores de los *predictores* $x_1, x_2, \ldots, x_p$.

$$F(y|x_1, x_2, \ldots, x_p) \equiv F(y|\mathbf{x}) \tag{4.1}$$

La conveniencia de definir a la regresión como el estudio de distribuciones condicionales, estriba en que permite incluir varios modelos y técnicas de modelado en un objetivo común, sin poner restricciones paramétricas que limiten la utilidad del modelo. Por ejemplo, en la expresión 4.1 no se hace referencia necesariamente al modelo de regresión lineal clásico. Si y es una variable categórica, el modelo especificado en 4.1 incluye regresión logística o regresión Poisson.

El estudio de la distribución condicional *per se* puede ser extremadamente complicado, en particular cuando la función de distribución es desconocida. Adicionalmente, en muchos problemas no se necesita conocer toda la distribución, sino ciertas caracterís-

ticas de ésta, usualmente la función de regresión (o función media) $E(y|\mathbf{x})$ o la función varianza $V(y|\mathbf{x})$, o bien la mediana $med(y|\mathbf{x}) = \underset{y}{^{arg}}\{F(y|\mathbf{x}) = 0.5\}$ o algún otro percentil $T_\alpha(\mathbf{x}) = P(Y > \alpha|\mathbf{x}) = 1 - F(\alpha|\mathbf{x})$, entre otras características numéricas.

De esta forma, en el análisis de regresión se consideran varias situaciones: desde el uso de técnicas de suavizamiento para estimar las funciones $E(y|\mathbf{x})$ y $V(y|\mathbf{x})$ en forma no paramétrica hasta los modelos paramétricos que se estudian en los modelos lineales generalizados, que incluyen a los modelos lineales clásicos, regresión logística, regresión Poisson, modelo de análisis de varianza, entre otros casos. En adelante supondremos que las funciones $E(y|\mathbf{x})$ y $V(y|\mathbf{x})$ existen y son finitas.

La representación gráfica siempre ha jugado un papel importante en el estudio de la regresión, particularmente cuando el número de predictores es pequeño. Por ejemplo, en el caso de una regresión simple con un solo predictor, una gráfica de dispersión de la variable respuesta y contra el predictor x puede proporcionar información útil sobre la curvatura en la función media, posible heteroscedasticidad [1]de los errores y modelos admisibles. Porque es fácil notar el patrón principal en la gráfica mientras simultáneamente se reconoce desviaciones brutas del modelo, también es posible identificar visualmente valores extremos, casos influenciales sin la necesidad de preespecificar un modelo paramétrico.

Las mismas facilidades están presentes al mostrar una regresión con dos predictores en una gráfica tridimensional ($3D$) rotada. Asimismo, todavía es posible el despliegue gráfico de datos para regresiones con tres predictores, en donde la respuesta y se sustituye con una versión discreta $\widetilde{y}$, construida particionando su rango, y asignando los predictores a los ejes y marcando los puntos con símbolos o colores correspondientes a las categorías de $\widetilde{y}$.

Generalmente no es posible construir una visualización comprensiva para una regre-

[1]Una *heteroscedasticidad* es la existencia de una varianza no constante en las perturbaciones aleatorias de un modelo econométrico.

sión con cuatro o más predictores. Una alternativa consiste en construir una matriz de gráficas de dispersión donde, es posible observar relaciones bivariadas que podrían proporcionar información útil sobre la distribución de $y|\mathbf{x}$ pero que también podrían llevar a falsas interpretaciones. Un procedimiento común en regresión multiple es primero ajustar un modelo adecuado y posteriormente observar los datos en gráficas de baja dimensión, como una gráfica de valores ajustados contra uno de los predictores, o gráficas de los residuales del modelo, diseñadas para proporcionar información sobre aspectos específicos del ajuste.

Como vemos, un problema estrechamente ligado al análisis de regresión es el problema de reducción de dimensión. Si el número de predictores p es pequeño, digamos 1 ó 2 es incluso factible graficar los datos para obtener información visual que permita entender mejor la relación entre las variables. Sin embargo, la mayoría de los problemas de la vida real involucran grandes cantidades de variables que pueden contribuir a explicar el comportamiento de la variable de interés.

Ejemplo

Un problema típico en la industria del crédito consiste en predecir el comportamiento de pago de sus potenciales clientes y basar la decisión de otorgar el crédito con base en esa predicción. De acuerdo en sus registros históricos, un banco, por ejemplo, puede clasificar a sus clientes en "buenos" y "malos" pagadores. La información que este banco tiene disponible de cada uno de sus clientes incluye su ingreso, años en el mismo empleo, monto de insolvencia en otros créditos, cantidad de tarjetas de crédito, y muchas otras variables obtenidas de la información proporcionada inicialmente por el cliente y a través de reportes de los burós de crédito. Es común tener 200 o más predictores potenciales. La variable de respuesta en este problema es una variable categórica con dos niveles. Para estudiar este problema usualmente se propone un modelo paramétrico de regresión logística. Sea $y = 1$ si el cliente es buen pagador y 0 en otro caso. Entonces $y|\mathbf{x}$ puede modelarse como una variable aleatoria Bernoulli con probabilidad de éxito dada

por

$$P(y = 1|\mathbf{x}) = \frac{1}{1 + e^{-\beta'\mathbf{x}}}$$

y el problema se reduce a estimar β. Obviamente, si el número de predictores es muy grande, la posibilidad de que el modelo tenga poder predictivo se irá reduciendo, además que se requerirán grandes cantidades de información para que los estimadores reduzcan su varianza. Sin un mecanismo de reducción, el problema sería demasiado complejo y difícil de estudiar o en el mejor de los casos, de implementar en la práctica. El analista de crédito se enfrenta a la pregunta de como combinar la información en todos los predictores en un modelo simple sin perdida significativa de información.

✠

Ejemplo

Una de las actividades prioritarias de los bancos centrales es la de proveer a la economía de un país de efectivo, procurando que el sistema de pagos funcione de manera eficiente; por ende tiene la responsabilidad de surtir a la economía de papel moneda, por lo que éste debe fabricar el efectivo o mandarlo a producir. Una pregunta interesante es conocer las cantidades de billetes que se requerirán para los futuros años, pero antes debemos saber cuál es la duración promedio de los billetes y monedas. Con el fin de estimar la duración promedio se debe tomar en cuenta el contexto macroeconómico esperado para los próximos años de acuerdo a su política monetaria planeada, tales variables proyectadas podrían ser: la cantidad de efectivo (en monto), la inflación, el producto interno bruto, índice de precios, tipo de cambio, etc. Un modelo parsimonioso que disminuya el número de variables es importante para la implementación y seguimiento del modelo de predicción.

✠

La idea básica de este enfoque gráfico es reducir la dimensión del vector de predictores sin o con poca pérdida de información en la regresión y sin requerir de modelo alguno, teniendo como objetivo obtener una solución gráfica que muestre toda la infor-

mación de la respuesta que está disponible en los predictores.

Esta metodología tiene potencial para tomar un papel fundamental en el escenario exploratorio y de diagnóstico del análisis de regresión.

Resumiendo, podemos considerar una metodología que tenga como meta encontrar una *gráfica resumen suficiente*, como se define a continuación:

Definición 4.1 *Una* GRÁFICA RESUMEN SUFICIENTE [2] *es aquella que contiene toda la información de la regresión en la muestra sobre la distribución condicional de la variable respuesta dados los predictores.*

Por ejemplo, con un solo predictor la gráfica de dispersión o "scatterplot" de la variable de respuesta contra ese predictor es siempre una *gráfica resumen suficiente* porque toda la información de la muestra acerca de la regresión está contenida en esta gráfica. El papel que la *gráfica resumen suficiente* juega en regresión es similar al de un parámetro que se debe estimar en un problema estadístico típico.

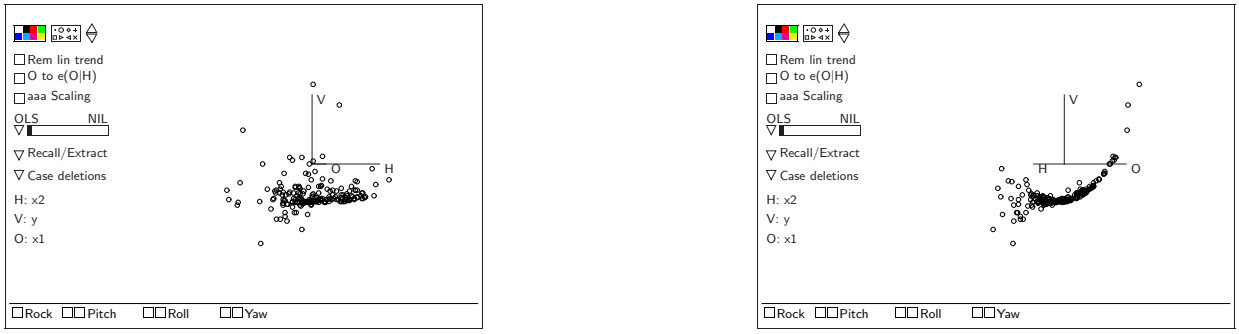
Las gráficas en tres ($3D$) o más variables son más complejas que las $2D$ y por ende, puede ser más difícil comprender completamente la información, sin embargo es posible obtener descripciones útiles principalmente sobre la función media y la función varianza.

Supóngase que se tiene una gráfica $3D$ con 2 predictores y la variable respuesta en el eje vertical; se pueden obtener vistas $2D$ rotando la gráfica $3D$ sobre su eje vertical hasta encontrar algún patrón en los datos, esta gráfica $2D$ puede sugerir características sobre la función media y la función varianza de la regresión, esto se muestra en la Figura 4.1(b). Estos argumentos caracterizan las vistas $2D$, pero pueden no tener nada que ver con la gráfica completa $3D$.

[2] Una *gráfica resumen suficiente* es la análoga a una *estadística suficiente*, la cual se define como aquella que utiliza toda la información contenida en la muestra. Una estadística T es *suficiente* para un parámetro θ si la distribución conjunta de la muestra dado T, se encuentra libre de θ, i.e. dado T entonces la muestra no tiene nada nuevo que aportar respecto a θ.

Ejemplo

Para simular una relación de dos predictores con 200 datos se generaron x_1 y x_2, $x_i \sim N(0,1)$ con $i \in \{1,2\}$ y $x_1 \perp\!\!\!\perp x_2$, i.e. independientes normalmente distribuidos. Y se creó la variable respuesta como $y_j = x_{1_j}^3 + (x_{1_j} - x_{2_j})^2 + \varepsilon$, donde ε tiene distribución $N(0,1)$ normal estándar independiente de x_1 y x_2, y j es un índice temporal.



a. No se observa algún patrón definido.

b. Existe un patrón

Figura 4.1: Vistas diferentes de una misma relación

✠

Uno de los objetivos generales del análisis estadístico es reducir los complejos resúmenes por otros más simples pero sin pérdida de información importante. Aplicando esta idea a las gráficas, podríamos sustituir la gráfica $3D$ por una $2D$ sí toda la información de la regresión estuviera contenida en la gráfica $2D$. Obsérvese que cuando la gráfica $3D$ es rotada alrededor de su eje vertical un ángulo θ, se obtiene una gráfica $2D$ de la variable respuesta contra una combinación lineal de los dos predictores $h(\theta) = b(cos\theta)x_1 + c(sen\theta)x_2$. Llamaremos a la gráfica de y contra $h(\theta)$ una vista $2D$ de la gráfica $3D$. Si la variable y depende de los predictores solamente a través de la combinación lineal $h(\theta)$, entonces la vista $2D$ de y contra $h(\theta)$ será una *gráfica resumen suficiente*. Es preferible, en general, realizar el análisis sobre una gráfica $2D$ en lugar que de una gráfica $3D$ más compleja. Surgen dos preguntas: ¿cuándo existe una gráfica $2D$ resumen suficiente?, y si así es ¿cómo

se puede estimar?

Cada vista $2D$ de la gráfica $3D$ de y contra $h(\theta)$ tiene una función media $E(y|h(\theta))$ y una función varianza $Var(y|h(\theta))$. Al girar la gráfica $3D$ y detenerla en la vista $2D$ que tiene visualmente la variación más pequeña sobre la función media, o equivalentemente donde la variación media $E(Var(y|h(\theta)))$ se reduce al mínimo. Este es un método visual para estimar una gráfica resumen. La Figura 4.1 muestra este ángulo. Es necesario un método que nos ayude a decidir si la gráfica resumen estimada pierde información sobre la regresión. Antes de describirlo definiremos nuevos conceptos.

4.2. Dimensión Estructural.

Definición 4.2 *Supóngase una regresión con la variable respuesta y y con p predictores $\mathbf{x} = (x_1, x_2, \ldots, x_p)'$. La* DIMENSIÓN ESTRUCTURAL *de la regresión es el número más pequeño de combinaciones lineales de* $\mathbf{x}$ *necesarias para caracterizar la regresión sin pérdida de información. La dimensión estructural es siempre un número entero entre* 0 *y* p*, y se dice que la regresión tiene estructura* $0D, 1D, ..., pD$*.*

4.2.1. Estructura Cero-Dimensional ó $0D$

Si $y|\mathbf{x}$ no depende de $\mathbf{x}$, decimos que la estructura es $0D$ porque ninguna combinación lineal de $\mathbf{x}$ proporciona alguna información sobre y, ya que $y \perp\!\!\!\perp \mathbf{x} \Rightarrow y \perp\!\!\!\perp f(x)$ para cualquier f *xlisp* (CB90, p.155). Con estructura $0D$, un histograma de y es una *gráfica resumen suficiente* puesto que ninguna gráfica que involucre a $\mathbf{x}$ contiene más información sobre y. Dada una estructura $0D$, la función media $E(y|\mathbf{x})$ y la función varianza $Var(y|\mathbf{x})$ son constantes para cualquier vista $2D$, y por lo tanto, si hay dependencia evidente en alguna vista $2D$ no es sostenible una estructura $0D$.

4.2.2. Estructura Unidimensional ó $1D$

Una regresión tiene estructura $1D$ si y depende de $\mathbf{x}$ a través de una sola combinación lineal de los predictores $\beta'\mathbf{x}$. Entonces no hay más información en los valores por separado de los predictores que la contenida en la combinación lineal $\beta'\mathbf{x}$; i.e. y es independiente de $\mathbf{x}$ dado $\beta'\mathbf{x}$ ($y \perp\!\!\!\perp \mathbf{x}|\beta'\mathbf{x}$). Si la regresión tiene estructura $1D$ entonces una gráfica $2D$ de y contra $\beta'\mathbf{x}$ es una *gráfica resumen suficiente*.

Muchos modelos usados comúnmente en regresión tienen estructura $1D$. Un ejemplo es

$$E(y|\mathbf{x}) = M(\beta'\mathbf{x}) \qquad y \qquad Var(y|\mathbf{x}) = \sigma^2 \tag{4.2}$$

donde M es el núcleo o "kernel"de la función media, que determina su forma. La forma de M debe ser aparente de una gráfica resumen, así que M puede estimarse de tal gráfica. Otro ejemplo con una estructura $1D$ es

$$E(y|\mathbf{x}) = \eta_0 \qquad y \qquad Var(y|\mathbf{x}) = V(\beta'\mathbf{x}) \tag{4.3}$$

La función $V(\beta'\mathbf{x})$ es llamada el núcleo de la función varianza, ésta es no negativa y puede tener un valor diferente para cada valor de $\beta'\mathbf{x}$. En este modelo la función media es constante para toda $\mathbf{x}$, pero la función varianza cambia con $\mathbf{x}$.

El modelo más general con estructura $1D$ es

$$E(y|\mathbf{x}) = M(\beta'\mathbf{x}) \qquad \text{y} \qquad Var(y|\mathbf{x}) = V(\beta'\mathbf{x}) \tag{4.4}$$

Éste modelo incluye a los previos como casos especiales. Para este modelo, la media y la varianza dependen de la misma combinación lineal $\beta'\mathbf{x}$.

Si se supone que los datos tienen estructura $1D$ y que la vista $2D$ es un buen resumen, entonces el núcleo de la función media M y el de la función varianza V pueden ser vi-

sualizados y estimados a partir de tal gráfica. En síntesis, con una estructura $1D$ la gráfica $2D$ de y contra $\beta'\mathbf{x}$ es una *gráfica resumen suficiente*. Si se conociera $\beta'\mathbf{x}$, se podría sustituir $\mathbf{x}$ por $h^* = \beta'\mathbf{x}$ sin pérdida de información en la regresión. Esto es una idea relevante porque permite utilizar una metodología aplicada en la regresión simple para comprender una regresión múltiple. El problema en este caso se reduce a estimar β, y para esto mostraremos algunos métodos más adelante.

4.2.3. Estructura Bidimensional ó $2D$

Una regresión tiene estructura $2D$ si dos combinaciones lineales $\beta_1'\mathbf{x}$ y $\beta_2'\mathbf{x}$ son necesarias para caracterizar la regresión, de modo que y sea independiente de $\mathbf{x}$ dado $B'\mathbf{x}$ donde B es la matriz con columnas β_1 y β_2. Con estructura $2D$, cualquier gráfica $2D$ de la variable respuesta contra una combinación lineal de predictores pierde información, por lo que se requiere una gráfica $3D$ para resumir la información de la regresión. Un modelo con una estructura $2D$ es

$$E(y|\mathbf{x}) = M(\beta_1'\mathbf{x}) \qquad y \qquad Var(y|\mathbf{x}) = V(\beta_2'\mathbf{x}) \tag{4.5}$$

con la condición de que β_1 y β_2 no sean exactamente colineales. Ambas combinaciones son necesarias para conocer completamente la distribución de $y|\mathbf{x}$.

Otro ejemplo con estructura $2D$ es

$$E(y|\mathbf{x}) = M(\beta_1'\mathbf{x}, \beta_2'\mathbf{x}) \qquad y \qquad Var(y|\mathbf{x}) = \sigma^2 \tag{4.6}$$

La función media depende de las dos combinaciones lineales, mientras que la función varianza es constante. Un ejemplo simple de un modelo de este tipo con dos predictores tiene como núcleo de la función media

$$M(\beta_1'\mathbf{x}, \beta_2'\mathbf{x}) = \frac{(\beta_1'\mathbf{x})^2 - (\beta_2'\mathbf{x})^2}{4} = \frac{[(x_1 + x_2)^2 - (x_1 - x_2)^2]}{4} = x_1 x_2$$

donde $\beta_1'\mathbf{x} = x_1 + x_2$ y $\beta_2'\mathbf{x} = x_1 - x_2$. Un producto de dos predictores se puede obtener a partir de dos combinaciones lineales de ellos, así que esta función media da una estructura 2D.

4.3. Subespacios de reducción de dimensión.

Como ya hemos comentado, una posible forma de atacar el problema de dimensión es tratar de encontrar un número menor de nuevos predictores o términos, que sean funciones simples de los predictores originales, de tal forma que toda la información relevante en la regresión se conserve en el conjunto de nuevos términos. Si se restringe a que los nuevos términos sean funciones lineales de los originales, entonces el problema se traduce del siguiente modo. Nuestro objetivo será encontrar una matriz B de dimensiones $p \times q$ con $q \ll p$ tal que $F(y|\mathbf{x}) = F(y|B'\mathbf{x})$, $\forall y$, $\mathbf{x} \in \mathcal{X}$ donde $\mathcal{X}$ es el espacio muestral de los predictores. La condición anterior es equivalente a decir que y es independiente de $\mathbf{x}$ dado $B'\mathbf{x}$. Geométricamente, esto corresponde a proyectar los puntos a un espacio de menor dimensión que el original.

A las combinaciones lineales $\beta_1'\mathbf{x}, \ldots \beta_d'\mathbf{x}$ se les llama *predictores suficientes* o *términos suficientes*.

Una matriz así siempre existe trivialmente ya que $B = I_{p\times p}$ satisface la condición, aunque no resuelve el problema ya que no hay ninguna reducción. Para poder avanzar en la solución del problema hay que introducir algunos conceptos.

Definición 4.3 *Sea $S(B)$ el espacio generado por las columnas de B. A los espacios generados por cualquier matriz cuyas columnas formen una base para $S(B)$, se les llamará* SUBESPACIO DE REDUCCIÓN DE DIMENSIÓN (SRD) *para la regresión de y en $\mathbf{x}$.*

El interés principal será obtener información acerca del subespacio SRD con la menor dimensión posible, con el fin de caracterizar la distribución condicional $F(y|\mathbf{x})$. De esta

forma, la gráfica de $(q+1)$ dimensiones de y versus $B'\mathbf{x}$ es una *gráfica resumen suficiente*, y si es posible encontrar un mínimo SRD, entonces podríamos obtener una *gráfica resumen suficiente minimal*, que llevaría al análisis mas simple posible de un problema.

Un subespacio candidato para ser el mínimo SRD es la intersección de todos los SRD de la regresión de y sobre $\mathbf{x}$,

$$S_{y|\mathbf{x}} = \bigcap_{\{\beta\}} S_{SRD}(\beta), \text{ donde } \beta \text{ es una base de } B$$

$S_{y|x}$ es siempre un subespacio lineal, sin embargo pueden presentarse dos problemas:

i. $S_{y|\mathbf{x}}$ no necesariamente es un SRD.
   Por ejemplo, si $\mathbf{x} = (x_1, x_2)'$ es uniformemente distribuido en el circulo unitario, y $y|\mathbf{x} = x_1^2 + \varepsilon$, entonces $S_1 = S_{SRD}((1,0))$ y $S_2 = S_{SRD}((0,1))$ son ambos SRD, pero la intersección, $S_{y|\mathbf{x}} = \{0\}$ no es un SRD.

ii. Un mínimo SRD no necesariamente es el único.
   En el ejemplo anterior, S_1 y S_2 son ambos mínimos SRD.

La siguiente definición introduce una restricción para evitar la no unicidad.

Definición 4.4 *Si existe un subespacio S tal que*

i. S es un SRD y

ii. Si $S \subset S_{SRD}(\beta)$ para cada S que es un SRD,

entonces S es llamado el SUBESPACIO DE REDUCCIÓN DE DIMENSIÓN CENTRAL *(o* SUBESPACIO CENTRAL*).*

De esta forma, es claro que cuando existe el *Subespacio Central*, es único y corresponde a la intersección de los SRD, o sea $S_{y|\mathbf{x}}$. Por último, cabe hacer la observación de que un único mínimo SRD no necesariamente es un *Subespacio Central*.

Ejemplo

Si $p = 3$ y $(x_1, x_2, x_3)'$ es uniforme en la esfera unitaria $x_1^2 + x_2^2 + x_3^2 = 1$ y $y|\mathbf{x} = x_1^2 + \varepsilon$ se sigue que $S_1 = S((1,0,0))$ es el único mínimo SRD. $S_2 = S((0,1,0),(0,0,1))$ es un SRD pero $S_1 \nsubseteq S_2$. En este problema el *Subespacio Central* no existe.

✠

Si el *Subespacio Central* existe, entonces es el único mínimo SRD. Para garantizar la existencia del *Subespacio Central* en un problema particular, se requiere imponer restricciones tanto en la distribución de $y|\mathbf{x}$ como en la distribución marginal de $\mathbf{x}$. Estas restricciones se discutirán en la sección 4.3.1.

El *Subespacio Central* establece un contexto y es un hiperparámetro para atacar el problema de reducción de dimensión. A continuación se establece formalmente la definición de dimensión.

Definición 4.5 *La* DIMENSIÓN ESTRUCTURAL DE LA REGRESIÓN *es la dimensión del Subespacio Central* $d = dim(S_{y|\mathbf{x}})$.

Ejemplo

Si $d = 0$, entonces $y \perp\!\!\!\perp \mathbf{x}$ (y y $\mathbf{x}$ son independientes), así que $\mathbf{x}$ no provee información sobre y, así que una gráfica que describa la distribución de y, como un histograma, una estimación de la densidad o una gráfica de caja sería una gráfica resumen suficiente minimal.

✠

Si $d = 1$, entonces una sola combinación lineal $\beta'\mathbf{x}$ es suficiente y la gráfica resumen suficiente minimal será $\{y, \beta'\mathbf{x}\}$.

Ejemplo

Modelos generales de regresión $1D$ son los siguientes:

i. $E(y|\mathbf{x}) = M(\beta'\mathbf{x})$ con $Var(y|\mathbf{x}) = \sigma^2$.

ii. $E(y|\mathbf{x}) = y_0$ y $Var(y|\mathbf{x}) = V(\beta'\mathbf{x})$.

✠

La dimensión estructural d de una regresión es un índice de su complejidad. Las regresiones con $d = 0$ son triviales. Las regresiones con $d = 1$ pueden ser complicados pero generalmente son menos complicados que regresiones con estructura $2D$, etcétera. La dimension estructural y los *predictores suficientes* $B'\mathbf{x}$ de una regresión son generalmente desconocidos.

Una *gráfica resumen suficiente minimal* para un modelo $2D$ podría ser una gráfica tridimensional dinámica que permita apreciar varias proyecciones del problema. Estas gráficas se pueden elaborar en `Arc`.

A partir del *Subespacio Central* surgen nuevos problemas y preguntas: ¿cómo puede estimarse $S_{y|\mathbf{x}}$ y su dimensión d ? ¿Qué condiciones son necesarias para garantizar la estimación? En las próximas secciones se darán respuesta a estas preguntas.

4.3.1. Garantizando la existencia del Subespacio Central

Cook (Coo98, pp.108–112) demuestra que se puede garantizar la existencia del *Subespacio Central* bajo las siguientes condiciones:

i. Si x tiene una densidad con un soporte convexo, o

ii. Si el problema se puede caracterizar con la función de regresión $E(y|\mathbf{x})$, esto es $\{y \perp\!\!\!\perp \mathbf{x}|E(y|\mathbf{x})\}$, y

iii. $E(y|\mathbf{x})$ se puede expresar como una serie convergente de potencias en x.

De acuerdo con Cook, la suposición de un *Subespacio Central* debe ser razonable en la mayoría de los problemas de regresión, así que en lo que sigue, *se supondrá que el subespacio central existe para todos lo problemas de regresión a menos de que explícitamente se diga lo contrario*.

4.4. Estimación del subespacio central SRD.

En esta sección revisaremos los supuestos requeridos por algunos de los métodos para estimar direcciones en el subespacio central. Posteriormente revisaremos las principales ideas asociadas a tales métodos y mostraremos su implementación mediante algunos ejemplos.

4.4.1. Supuestos necesarios para la estimación de $S_{y|\mathbf{x}}$

Para poder estimar el subespacio central se requiere imponer restricciones en la distribución de los predictores, para garantizar que ciertas relaciones en dimensiones superiores no se distorsionan cuando son proyectadas a dimensiones inferiores. Chiaramonte y Cook (CC97) hacen notar la importancia de que las restricciones sean sobre la distribución de $\mathbf{x}$ y no en la distribución que es objeto de estudio, $y|\mathbf{x}$, como es común en la práctica, y que éste punto es clave en el estudio de reducción de dimensión, ya que la distribución de $\mathbf{x}$ se conoce al menos parcialmente, y en el caso de diseño de experimentos, puede ser incluso controlada.

La primera condición es fundamental para todos los métodos de estimación del subespacio central, y la segunda condición es necesaria para algunos métodos.

Primera condición: Predictores Relacionados Linealmente (PRL)

La condición fundamental para estimar el subespacio central es la de predictores linealmente relacionados (PRL) en el siguiente sentido. Si la regresión tiene estructura kD, caracterizada por la matriz B, entonces se requiere que la regresión de cada predictor sobre los predictores suficientes sea lineal, esto es:

$$E(x_j|B'\mathbf{x}) = a_j + b'_j(B'\mathbf{x}), \quad j = 1, ..., p$$

Esta condición no puede verificarse directamente porque B es desconocida. En la práctica, para resolver este punto, se puede imponer la restricción adicional de que la condición se cumpla para toda B; a veces puede parecer una condición muy restrictiva, pero Eaton (Eat86) demostró que esta condición es equivalente a pedir que la distribución conjunta de los predictores tenga simetría elíptica, y la distribución normal cumple tal condición. Adicionalmente, Hall y Li (HL92) por una parte, y Diaconis y Freedman (DF84) por otra, resaltan que esta condición no es muy severa, ya que las proyecciones en baja dimensión de datos multidimensionales son aproximadamente normales. Empero, mientras mas lejos de satisfacer la condición más probable es que se causen problemas.

Se han trabajado métodos para lograr que los predictores se aproximen suficiente a la condición requerida, mediante transformaciones de las variables originales. Por ejemplo, Cook y Nachtsheim (CN94) sugieren usar unos pesos, conocidos como pesos de Voronoi, para lograr que la distribución conjunta de los datos transformados se aproxime a una distribución elíptica. Otra forma consiste en buscar transformaciones simultáneas de los datos originales, particularmente de la familia de Box-Cox, para aproximar los datos a una normal multivariada. En `Arc`, tal estrategia está implementada. Tal serie de transformaciones no afecta los resultados ya que la distribución de $y|\mathbf{x}$ es la misma que la de $y|T(\mathbf{x})$ cuando T son transformaciones de tales familias.

Segunda condición: Varianzas Constantes (VC)

Algunos métodos que se discutan posteriormente requieren además el supuesto de varianza constante (VC) en la distribución de $\mathbf{x}$:

$$Cov(x_i x_j|B'\mathbf{x}) = \sigma_{ij}, \quad i,j \in \{1,...,p\}$$

De nuevo, si los datos se alejan mucho de la condición, entonces pueden surgir problemas en el proceso de estimación de direcciones en el subespacio central. Sin embargo, las transformaciones mencionadas anteriormente pueden ayudar a que la condición

se cumpla aproximadamente, en particular, utilizando transformaciones de Box-Cox. En términos generales, esta condición es mucho menos importante que la condición de predictores linealmente relacionados.

4.4.2. Métodos de estimación en general

En esta subsección se supondrá que se satisfacen las condiciones *PRL* y *VC* discutidas en la subsección anterior.

Para estimar el subespacio central hay tanto métodos numéricos como gráficos. Cada método tiene ventajas y desventajas dependiendo de la complejidad de la regresión particular. En la sección siguiente, se describirán las características generales de los métodos numéricos y la teoría detrás de los métodos gráficos, junto con algunos ejemplos.

En términos generales, la metodología se puede resumir como sigue: cada método se concentra en la estimación de p combinaciones lineales del predictor x:

$$\mathbf{x} = (x_1, ..., x_p)' \quad \Leftrightarrow \quad \mathbf{W} = (\widehat{\beta}_1' x, ..., \widehat{\beta}_p' x)' = \widehat{B} x$$

y los vectores estimados $\widehat{\beta}_1, ..., \widehat{\beta}_p$ se ordenan de acuerdo a la "verosimilitud"de que pertenezcan a $S_{y|x}$. Si $dim(S_{y|x}) = d$ entonces las primeras d combinaciones lineales son suficientes.

Hay esencialmente cuatro métodos numéricos[3] para estimar las direcciones en el subespacio central:

1. Mínimos Cuadrados Ordinarios (OLS),
2. Regresión Inversa Particionada (SIR),
3. Estimación Particionada de la Varianza Promedio (SAVE), y

[3]para mantener consistencia con las referencias bibliográficas, se mantendrá la abreviatura en inglés.

4. Direcciones Hessianas Principales (pHd).

El primer método es bien conocido, pero hay que darle un contexto en este enfoque. Los tres métodos restantes se basan en el siguiente procedimiento general:

Supóngase que es posible obtener un estimador consistente $\widehat{M}$ de una matriz poblacional M de $p \times l$ que tiene la propiedad de que $S(M) \subseteq S_{y|x}$. La matriz M es llamada la matriz Kernel y depende del método usado. De esta forma, la matriz $\widehat{M}$ se puede utilizar para inferir al menos una porción de $S_{y|x}$, el vector W se calcula como $\widehat{M}x = W$. Más delante se verá cómo se puede obtener una matriz kernel para cada método.

4.5. Métodos numéricos

Los métodos numéricos listados en la sección anterior han sido implementados tanto en `Arc` como en `R`. En esta sección mostraremos cómo se aplican los métodos a ejemplos simulados. El paquete `dr`(Wei01) que incluye funciones para todos los métodos debe ser cargado en la sesión de `R`. En `Arc` los métodos están disponibles en **Graph & Fit** → **Inverse Regression...**, en donde se puede escoger el método que se desee aplicar.

4.5.1. El método de mínimos cuadrados (OLS)

Para facilitar el contexto, supóngase que se tiene un modelo $1D$ que depende de una sola combinación lineal en la función media y la función varianza:

$$E(y|x) = M(\beta' x) \quad \text{y} \quad Var(y|x) = V(\beta' x)$$

donde M y V son funciones arbitrarias, y $E(y|x)$ y $Var(y|x)$ dependen de x solo a través de β. La pregunta relevante es: ¿se puede estimar β sin hacer ningún supuesto sobre M y V? El siguiente resultado, conocido como el teorema de Li-Duan (LD89), da respuesta afirmativa y pone el método de mínimos cuadrados en contexto:

Teorema 4.1 *(Li-Duan) Sea $S_{y|x}^{(\alpha)}$ un SRD y supóngase que*

i. La condición LR *se satisface, esto es $E(x|\alpha' x) = a + b(\alpha' x)$, y*

ii. $Var(x)$ es definida positiva.

Entonces $\beta_{OLS} \in S_{y|x}^{(\alpha)}$.

La prueba de este teorema se puede encontrar en (LD89). Entonces, el método de mínimos cuadrados proporciona un estimador de $S_{y|x}$ cuando el modelo es $1D$. Nótese que OLS sólo se usa como una forma de estimación y en realidad no se hace ningún supuesto sobre la forma del modelo.

Cuando el modelo tiene dimensión mayor a 1, el vector estimado por mínimos cuadrados sigue siendo un vector que pertenece al subespacio central, aunque no hay forma de saber cuál de todos. En el modelo propuesto, la gráfica de y vs $\hat{y} = \widehat{\beta}'_{OLS} x$ es una gráfica resumen suficiente, y de la gráfica se puede inferir las formas de M y V.

Ejemplo

Consideremos un problema de regresión simulado con $p = 5$ predictores y $n = 500$ casos; cada predictor tiene distribución normal estándar independiente de los otros y la respuesta tiene media dada por $E(y|\mathbf{x}) = sen(x_1 - x_2)$ y función varianza dada por $\text{Var}(y|\mathbf{x}) = a(x_1 - x_2)^2$ para una cierta constante a.

En este ejemplo $1D$, $\beta = (1, -1, 0, 0, 0)$ y los predictores, al tener distribución normal, satisfacen las condiciones requeridas por el teorema de Li-Duan. Consideramos dos casos para ponderar el efecto de la función varianza: $a = 0.2$ y $a = 2$. En la Tabla 4.1 se muestra el resultado de la estimación para $a = 0.2$. Claramente, el estimador de mínimos cuadrados recupera la dirección correcta en este caso un múltiplo, y como en este caso la función varianza es relativamente pequeña, en la gráfica 4.2 (a) de $\hat{y}$ versus y vemos que es posible

Cuadro 4.1: Resultado del ajuste para el caso $a = 0.2$, $n = 500$ y $p = 5$ para la función media en el ejemplo

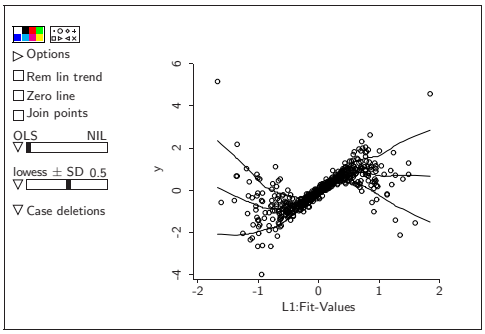
```
Data set = li_duan, Name of Fit = L1
Normal Regression
Kernel mean function = Identity
Response      = y
Terms         = (X1 X2 X3 X4 X5)
Coefficient Estimates
Label      Estimate        Std. Error    t-value    p-value
Constant   0.0321076       0.0320791       1.001     0.3174
X1          0.405941       0.0318661      12.739     0.0000
X2         -0.438996       0.0325823     -13.473     0.0000
X3         -0.0230788      0.0327163      -0.705     0.4809
X4          0.0419242      0.0321937       1.302     0.1934
X5         -0.0286794      0.0314346      -0.912     0.3620

R Squared:               0.396809
Sigma hat:               0.715395
Number of cases:             500
Degrees of freedom:          494

Summary Analysis of Variance Table
Source         df       SS            MS           F    p-value
Regression      5   166.321       33.2641      65.00    0.0000
Residual      494   252.824       0.51179
```

recuperar la función $M = sen$, y se recupera el efecto de la función varianza que se puede percibir que aumenta en los extremos en forma simétrica si se agrega lowess a la gráfica.

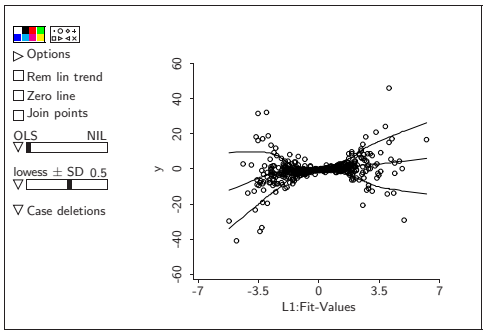
En el caso $a = 2$, el efecto de la varianza es mayor, como se puede ver en la gráfica 4.2 (b), y en la Tabla 4.2 se puede ver que mínimos cuadrados aún encuentra aproximadamente bien la dirección correcta, aunque ya no es exactamente un múltiplo de ésta, por el efecto de la función varianza que ahora es más importante. En este caso, con algún método de diagnóstico se puede intentar volver constante la varianza, y después reajustar para encontrar un mejor estimador de mínimos cuadrados que se aproxime a la dirección real.

Cuadro 4.2: Resultado del ajuste para el caso $a = 2$, $n = 500$ y $p = 5$ para la función media en el ejemplo

```
Data set = li_duan, Name of Fit = L1
Normal Regression
Kernel mean function = Identity
Response      = y
Terms         = (X1 X2 X3 X4 X5)
Coefficient Estimates
Label      Estimate        Std. Error    t-value    p-value
Constant   0.0899895       0.285488        0.315     0.7527
X1         1.51707         0.273306        5.551     0.0000
X2        -1.05091         0.285581       -3.680     0.0003
X3         0.112435        0.282559        0.398     0.6909
X4         0.00987396      0.274280        0.036     0.9713
X5         0.0485254       0.296378        0.164     0.8700

R Squared:               0.0791396
Sigma hat:                6.35208
Number of cases:             500
Degrees of freedom:          494

Summary Analysis of Variance Table
Source         df       SS            MS           F    p-value
Regression      5   1713.01       342.601       8.49    0.0000
Residual      494   19932.4       40.3489
```



a. Con ponderador de 0.2



b. Con ponderador de 2

Figura 4.2: Comparación de los datos ajustados contra la respuesta con diferentes ponderadores

✠

4.5.2. Regresión Inversa Particionada (SIR)

Regresión inversa particionada *SIR (sliced inverse regression)* es un método que sirve para estimar predictores suficientes cuando $d > 1$. La idea detrás de SIR es que es posible aprender de la regresión inversa $x|y$, puede ser informativa de $y|x$, y facilita la visualización, ya que $x|y$ es un conjunto de p gráficos bidimensionales de x_i vs y.

Considérese la trayectoria de la regresión inversa $E(x|Y = y)$ cuando y varía. En general, esto genera una curva en $\mathbb{R}^p$. El centro de la curva se ubica en $E(E(x|y)) = E(x)$. El siguiente teorema (dado por Li) muestra que bajo condiciones adecuadas, la curva reside en un espacio *afín* de dimensión k que esta relacionado al subespacio central.

Teorema 4.2 *(Li) Si se satisface PRL, entonces la curva centrada $E(x|y) - E(x)$ está contenido en el subespacio generado por $\Sigma_x\beta_k$ con $k = 1, ..., p$, donde $\Sigma_x = COV(x)$.*

El subespacio generado por $\Sigma_x\beta_k$, a su vez está contenido en $S_{y|x}$.

Las pruebas de estos resultados se pueden encontrar en Cook (1998) (Corolario 10.1 y la proposición 11.1). De los resultados anteriores, Li estableció un algoritmo para calcular

la matriz kernel. El algoritmo es el siguiente: Sea $(y_1, x_1), ..., (y_n, x_n)$ el conjunto de datos originales.

1. Ordenar los datos según los valores de y.

2. Se particiona el rango de y en h intervalos, de forma que hay aproximadamente el mismo número de observaciones en cada segmento. Sea n_h el número de elementos en el segmento h. En la práctica el número de intervalos se toma entre 8 y 15, y claramente depende de p, n y d.

3. En cada segmento de la partición, se calcula la media muestral de x,

$$\overline{x}_h = \sum_{i \in h} \frac{x_{(i)}}{n_h}$$

   Los valores de y solo fueron utilizados para generar la partición, y no se vuelven a utilizar.

4. Calcula la matriz de covarianzas muestral para las medias de x, ponderadas por el número de observaciones en el intervalo h_i:

$$\widehat{\sum} = \sum_{i=1}^{h} \frac{n_h}{n} (\overline{x}_i - \overline{x})(\overline{x}_i - \overline{x})'$$

5. Calcula la covarianza muestral de las x,

$$\widehat{\sum}_x = \sum_{i=1}^{h} \frac{1}{n} (\overline{x}_i - \overline{x})(\overline{x}_i - \overline{x})'$$

6. Obtener los eigenvectores de $\widehat{\sum}$ con respecto a $\widehat{\sum}_x$; esto es, los vectores $\widehat{\beta}_i$ tales que

$$\widehat{\sum} \widehat{\beta}_i = \widehat{\lambda}_i \widehat{\sum}_x \widehat{\beta}_i \quad \text{, con} \quad \widehat{\lambda}_1 \leq \widehat{\lambda}_2 \leq \cdots \widehat{\lambda}_p$$

7. El vector W se obtiene con $\widehat{\beta}_1' x,\ \widehat{\beta}_2' x, \ldots,\ \widehat{\beta}_p' x$.

8. Graficar y versus $\widehat{\beta}_1' x, \widehat{\beta}_2' x, ..., \widehat{\beta}_p' x$. De aquí se obtiene información relevante sobre la estructura de la regresión.

Ejemplo

En el siguiente ejemplo, consideramos $p = 5$ predictores independientes con distribución normal estándar, y el tamaño de muestra $n = 100$. La variable de respuesta es generada con la siguiente función racional:

$$y|\mathbf{x} = \frac{x_1}{0.5 + (x_2 + 1.5)^2} + \sigma\varepsilon.$$

Con la finalidad de resaltar los puntos importantes del método, la respuesta fue generada sin error, tomando $\sigma = 0$. El número de segmentos considerado es de $h = 8$. En este caso, la dimensión del espacio central es 2. La salida que se muestra en la Tabla 4.3 muestra que hay dos valores propios grandes, y la dimensión encontrada es la correcta. Las direcciones verdaderas son los vectores canónicos e_1 y e_2, mientras que las direcciones encontradas por SIR son los vectores

$$(0.993, 0.067, 0.016, -0.080, 0.057) \text{ y } (-0.136, -0.956, -0.069, -0.197, -0.155)$$

. Una posible forma de medir la cercanía entre los subespacios estimado y real es a través de correlaciones canónicas, pero no lo haremos en este trabajo.

La correspondiente salida que se obtiene de *R* se muestra en la tabla 4.4. Ambas salidas son esencialmente idénticas.

De acuerdo con lo que se ha dicho hasta aquí, una gráfica resumen apropiada sería la que se obtiene considerando la respuesta versus las primeras dos direcciones encontradas por SIR. La Figura 4.3 muestra tal gráfica, con una superficie cuadrática en esas dos direcciones sobreimpuesta como aproximación.

✠

Una limitación importante de SIR, en donde el procedimiento puede perder predictores suficientes, es en el caso en que hay relaciones de simetría, ya que el procedimiento

Cuadro 4.3: Resultado del ajuste para SIR usando *Arc*.

```
Inverse Regression SIR
Name of Dataset = sir1
Name of Fit = I2.SIR
Response = y
Predictors = (X1 X2 X3 X4 X5)

Number of slices = 8
Slices sizes are:  (13 13 13 13 12 12 12 12)
Std. coef. use predictors scaled to have SD equal to one.
                 Lin Comb 1        Lin Comb 2        Lin Comb 3
Predictors        Raw    Std.       Raw    Std.       Raw    Std.
X1              0.993   0.993    -0.136  -0.128     0.190   0.204
X2              0.067   0.072    -0.956  -0.966    -0.051  -0.059
X3              0.016   0.016    -0.069  -0.066    -0.356  -0.385
X4             -0.080  -0.074    -0.197  -0.171    -0.000  -0.000
X5              0.057   0.052    -0.155  -0.134     0.914   0.898

Eigenvalues          0.786            0.395            0.083
R^2(OLS| SIR)        0.990            0.990            0.997

Approximate Chi-squared test statistics based on partial
sums of eigenvalues times 100

Number of    Test
Components   Statistic     df p-value
   1         131.45        35   0.000
   2         52.866        24   0.001
   3          13.36        15   0.574
   4          5.097         8   0.747
```

Cuadro 4.4: Resultado del ajuste para SIR usando *R*.

```
> library(dr)
> msir <- dr(y~X, nslices=8)
> summary(msir)

Call:
dr(formula = y ~ X, nslices = 8)

Method:
sir with 8 slices, n = 100, using weights.

Slice Sizes:
13 13 13 13 12 12 12 12

Eigenvectors:
       Dir1    Dir2       Dir3    Dir4
X1  0.99276 -0.1359 -0.1904437  0.2874
X2  0.06731 -0.9560  0.0510721  0.1343
X3  0.01555 -0.0693  0.3555694 -0.6104
X4 -0.07996 -0.1970  0.0002627  0.6453
X5  0.05704 -0.1549 -0.9136154 -0.3322

              Dir1   Dir2    Dir3    Dir4
Eigenvalues 0.7858 0.3951 0.08263 0.04296
R^2(OLS|dr) 0.9902 0.9905 0.99710 0.99746

Asymp. Chi-square tests for dimension:
               Stat df   p-value
0D vs >= 1D 131.451 35 4.351e-13
1D vs >= 2D  52.866 24 6.025e-04
2D vs >= 3D  13.360 15 5.745e-01
3D vs >= 4D   5.097  8 7.472e-01
```

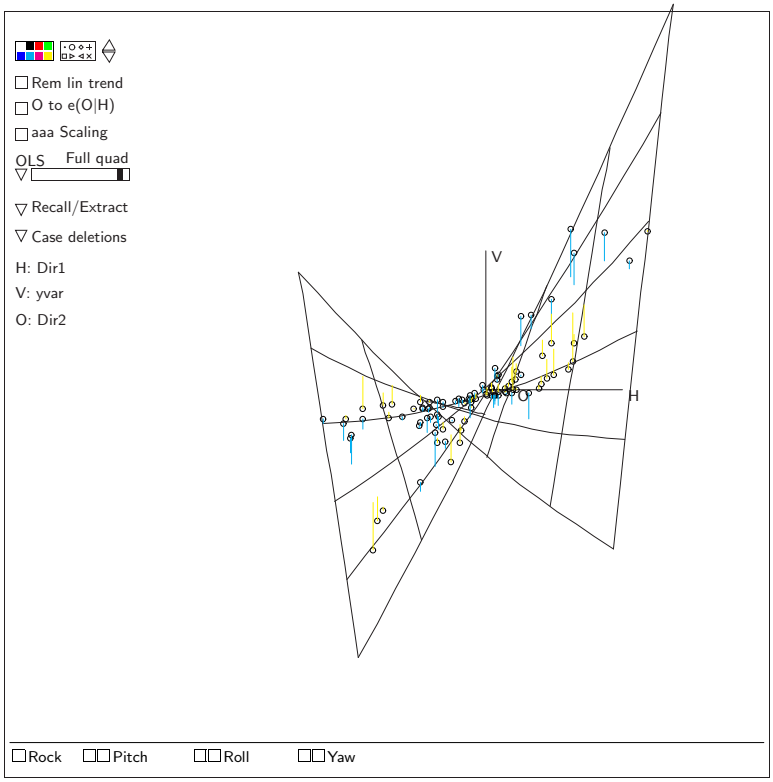


Figura 4.3: Respuesta vs. las primeras dos direcciones estimadas por SIR

de estimación obtiene promedios de los predictores y cuando éstos son cercanos a 0, se pierde la habilidad para detectar tales casos. A veces es conveniente, para resolver estas posibilidades, usar varios métodos combinados.

Ejemplo

Considérese ahora un ejemplo similar al previo, pero en donde la respuesta fue generada usando la función $y = (x_1 + x_2)^2$. La Tabla 4.5 muestra el resultado de la estimación en donde SIR no pudo recuperar la dimensión correcta, ya que las pruebas indican dimensión 0, y las direcciones encontradas no corresponden a $\mathbf{e}_1 + \mathbf{e}_2$.

✠

Cuadro 4.5: Resultado del ajuste para SIR usando *Arc*.

```
Inverse Regression SIR
Name of Dataset = sir2
Name of Fit = I1.SIR
Response = y
Predictors = (X1 X2 X3 X4 X5)

Number of slices = 8
Slices sizes are:  (13 13 13 13 12 12 12 12)
Std. coef. use predictors scaled to have SD equal to one.
                Lin Comb 1       Lin Comb 2       Lin Comb 3
Predictors       Raw    Std.      Raw    Std.      Raw    Std.
X1            -0.887  -0.882    0.405   0.409    0.318   0.321
X2             0.266   0.297    0.533   0.604    0.464   0.526
X3             0.298   0.295   -0.008  -0.008    0.420   0.421
X4            -0.100  -0.090   -0.730  -0.672    0.444   0.408
X5             0.211   0.198    0.134   0.127   -0.556  -0.526

Eigenvalues         0.180           0.115           0.081
R^2(OLS| SIR)       0.257           0.735           0.914

Approximate Chi-squared test statistics based on partial
sums of eigenvalues times 100

Number of    Test
Components   Statistic     df p-value
   1         41.455        35   0.210
   2         23.414        24   0.496
   3         11.952        15   0.683
   4         3.8474         8   0.871
```

4.5.3. Estimación Particionada de la Varianza Promedio (SAVE)

Como un alternativa al problema de dependencia simétrica en *SIR*, Cook y Weisberg (1991) propusieron un nuevo método para encontrar una matriz kernel M. Este método supone que se satisfacen ambas condiciones PRL y CV.

Cuando ambas condiciones se cumplen, puede probarse que

$$S\{E(I - Var(x|y))^2\} \subseteq S_{y|x}$$

Aquí se considera a $E(I - Var(x|y))^2$ como un operador lineal.
Algoritmo de *SAVE*:

1. Construir la media muestral $\widehat{E}(x)$ y $\widehat{\sum}_x$, y calcular los predictores muestrales estandarizados
$$\widehat{Z}_i = \widehat{\sum}_x^{-1/2}(x_i - \widehat{E}(x)) \quad \text{para} \quad i = 1, \ldots, n.$$

2. Dividir el rango de y en H subintervalos y construir la matriz de covarianzas $\widehat{V}_s$ de las $\widehat{z}$ usando los datos en cada subintervalo $s = 1, ..., H$. $\widehat{V}_s$ estima a $Var(Z|\widehat{y})$ en el subintervalo s.

3. Construir
$$M = \sum_{s=1}^{H} \frac{n_s}{n}(I - \widehat{V}_s)^2$$

4. El j-ésimo predictor *SAVE* muestral se construye como
$$S_i = b_j{}'\widehat{z}_i \quad j = 1, \ldots, p; \quad i = 1, \ldots, n.$$
donde b_j es el j-ésimo eigenvector de M que corresponde a S_n eigenvalor muestral ordenado $\widehat{\lambda}_1 \geq ... \geq \widehat{\lambda}_p$.

5. Finalmente, los eigenvectores b_j en la escala z se pueden transformar a la escala original como $h_j = \widehat{\sum}_x^{-1/2} b_j$.

Sin embargo, para SAVE no hay una estadística de prueba asintótica como la hay para SIR. Como alternativa, se requiere calcular una prueba basado en métodos de permutación, como se describe en (Yin00). Para poder obtener esta prueba, es necesario incluir el archivo `permtest.lsp` en el directorio `Extras` para que se complete el menú asociado con el modelo ajustado por SAVE en *Arc*. Al cargar este archivo, aparece la opción adicional **Permutation tests** y al seleccionar este menú aparece un recuadro pidiendo el número de permutaciones a realizar. Usualmente un valor cercano a 50 es suficiente.

Ejemplo

Retomamos el mismo ejemplo en el que SIR no funcionó con el conjunto de datos que se llama `sir2`, donde la respuesta fue generada con la función $y = (x_1 + x_2)^2$. Para este caso, la salida de *Arc* se muestra en la tabla 4.6. Aquí podemos ver que la primera dirección estimada es $(-0.628, -0.765, 0.133, 0.014, -0.044)$ que es relativamente cerca de la dirección real $(1, 1, 0, 0, 0)$. Al final también se incluyen los resultados de la prueba de permutación en la que se prueba que 1D es significativa.

La correspondiente salida en *R* es mostrada en la Tabla 4.7. Para poder tener acceso a las pruebas de permutación se requiere usar la función `dr.permutation.test` El número de permutaciones que realiza la función se especifica con el argumento `npermute`, que tiene 50 como valor por omisión.

✠

4.5.4. Direcciones hessianas principales (pHd)

Cuando los predictores son normalmente distribuidos, es posible asociar la matriz Hessiana $H(x) = \frac{\partial E(y|x)}{\partial x \partial x'}$ de la función promedio al subespacio central. Como $E(y|x) = E(y|B'x)$ para $B \in \mathcal{S}_{y|x}$, se sigue que

$$H(x) = BH(B'x)B'$$

y los eigenvectores de $H(x)$ viven en $\mathcal{S}_{y|x}$.

Cuadro 4.6: Resultado del ajuste para SAVE usando *Arc*.

```
Inverse Regression SAVE
Name of Dataset = sir2
Name of Fit = I3.SAVE
Response = y
Predictors = (X1 X2 X3 X4 X5)

Number of slices = 8
Slices sizes are:  (13 13 13 13 12 12 12 12)
Std. coef. use predictors scaled to have SD equal to one.
                Lin Comb 1       Lin Comb 2       Lin Comb 3
Predictors       Raw    Std.      Raw    Std.      Raw    Std.
X1            -0.628  -0.585    0.541   0.562    0.415   0.424
X2            -0.765  -0.801   -0.133  -0.156   -0.231  -0.265
X3             0.133   0.123    0.300   0.311    0.604   0.614
X4             0.014   0.012    0.400   0.379   -0.261  -0.243
X5            -0.044  -0.038   -0.663  -0.647    0.585   0.561

Eigenvalues         2.025           0.667           0.447
R^2(OLS|SAVE)       0.783           0.870           0.871
Inverse Regression SAVE

Name of Dataset = sir2
Name of Fit = I3.SAVE
Response = y
Predictors = (X1 X2 X3 X4 X5)

Number of slices = 8
Slices sizes are:  (13 13 13 13 12 12 12 12)

Permutation pvalues for
sums of eigenvalues times 100
Number of permutations: 50

Number of    Test             Permutation
Components   Statistic          p-value
   1         388.39             0.000
   2         185.84             0.380
   3         119.14             0.130
```

Cuadro 4.7: Resultado del ajuste para SAVE usando *R*.

```
> msave <- dr(y~X,method="save",nslices=8)
> summary(msave)

Call:
dr(formula = y ~ X, method = "save", nslices = 8)

Method:
save with 8 slices, n = 100, using weights.

Slice Sizes:
13 13 13 13 12 12 12 12

Eigenvectors:
       Dir1    Dir2    Dir3    Dir4
X1  0.62810 -0.5409 -0.4154  0.3346
X2  0.76527  0.1334  0.2306 -0.4105
X3 -0.13310 -0.3002 -0.6038 -0.7312
X4 -0.01450 -0.4001  0.2606  0.1311
X5  0.04385  0.6629 -0.5847  0.4095

              Dir1   Dir2   Dir3   Dir4
Eigenvalues 2.0255 0.6671 0.4469 0.4291
R^2(OLS|dr) 0.7835 0.8696 0.8705 0.8823

> dr.permutation.test(msave)

Permutation tests
Number of permutations:
[1] 50

Test results:
              Stat p-value
0D vs >= 1D 388.39       0
1D vs >= 2D 185.84       0.380
2D vs >= 3D 119.14       0.130
```

Como H típicamente varía con x a menos que la superficie sea de dimensión menor que 2, se sustituye H por $\mathrm{E}(H)$, y bajo el supuesto de normalidad, $\mathrm{E}(H) = M$, donde $M = \mathrm{E}((y - \mathrm{E}(y)) \times xx')$.

Si $\mathcal{S}_{yxx}$ denota $\mathcal{S}(M)$, entonces $\mathcal{S}_{yxx} \subset \mathcal{S}_{y|x}$.

Un estimador consistente de M se obtiene de

$$\hat{M} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \bar{y})\hat{x}_i\hat{x}_i'.$$

Esta versión de pHd basada en la respuesta se denota pHdy.

Como la matriz Hessiana no cambia agregando funciones lineales de x, entonces la misma ecuación de arriba puede usarse reemplazando y por los residuales de la regresión lineal de y en x, r. Esta versión se denota como pHdr.

Estadística de prueba

El procedimiento de inferencia para $d = \dim(\mathcal{S}_{y|x})$ depende de la versión de pHd que se use. En el caso de pHdy, la inferencia es sobre d directamente, pero en el caso de pHdr la dimensión es sobre $d - 1$.

La estadística de prueba es de la forma

$$\hat{\Lambda}_m(\xi) = \frac{n}{2\hat{\mathrm{Var}}(\xi)}\sum_{j=m+1}^{p}\hat{\lambda}_j^2$$

donde ξ es el vector de respuestas y para pHdy, o el vector de residuales para pHdr.

De nuevo la distribución de $\hat{\Lambda}_m(\xi)$ depende de la distribución de los predictores.

a. Cuando los predictores son normales, $\hat{\Lambda}_m(\xi) \overset{a}{\sim} \chi^2_{\frac{(p-k)(p-k+1)}{2}}$

b. Cuando los predictores satisfacen PLR, la distribución asintótica de $\hat{\Lambda}_m(\xi)$ es la misma distribución de $C = \frac{1}{2\mathrm{Var}(\xi)}K$, donde K es una combinación de $(p-k)(p-k+1)/2$ variables ji-cuadradas, cada una con un g.l.

Cuadro 4.8: Resultado del ajuste para pHd usando *Arc*.

```
Inverse Regression pHd(OLS residuals)
Name of Dataset = sir1
Name of Fit = I4.pHd
Response = OLS residuals based on y
Predictors = (X1 X2 X3 X4 X5)
Std. coef. use predictors scaled to have SD equal to one.
                 Lin Comb 1       Lin Comb 2       Lin Comb 3
Predictors       Raw    Std.      Raw    Std.      Raw    Std.
X1             0.849   0.835    0.596   0.575   -0.222  -0.220
X2            -0.505  -0.533    0.752   0.778   -0.183  -0.195
X3             0.008   0.008    0.100   0.097    0.946   0.946
X4            -0.019  -0.017    0.171   0.151    0.149   0.135
X5             0.153   0.138   -0.200  -0.177    0.017   0.015

Eigenvalues         0.490          -0.278           0.136
R^2(OLS| pHd)       0.701           0.970           0.979

Tests from sums of squared eigenvalues times 192.863
Number of   Eigenvalue                   ChiSq
Components  Partial Sum         DF     p-value
      1       66.53          |  15      0.000
      2       20.21          |  10      0.027
      3       5.319          |   6      0.504
      4       1.750          |   3      0.626
```

Ejemplo

Como ejemplo consideraremos el mismo conjunto de datos que fue utilizado para el primer ejemplo de SIR, además, sólo consideramos el caso de pHd. Recordando que la inferencia se realiza considerando los residuales de la regresión de la respuesta en los predictores, vemos que se estima la dimensión correcta. Por otra parte, las direcciones encontradas por pHd son relativamente cercanas a las direcciones correctas, aunque esto no es tan claro como en el caso de SIR.

✠

4.6. Métodos gráficos

El objetivo de la regresión gráfica es construir gráficas resumen suficientes en regresiones con más de dos predictores sin suponer un modelo para la distribución de $y|\mathbf{x}$. Esto puede ayudar en un análisis de regresión a encontrar estructura en los datos y a seleccionar un primer modelo para estudio posterior. También puede ser de utilidad para verificar un modelo que ya ha sido propuesto.

Supóngase que el vector de predictores está dado por $\mathbf{x} = (x_1, \ldots, x_p)'$ y que se seleccionan dos predictores para separarlos del resto, por ejemplo, por simplicidad supongamos que se seleccionan los dos primeros, x_1 y x_2; y denotemos al resto como $\mathbf{x}_3$.

La pregunta que se formula en regresión gráfica es la siguiente: ¿Es posible reemplazar x_1 y x_2 por una combinación lineal de ellos, $g = b_1x_1 + b_2x_2$ sin pérdida de información? Una gráfica que puede ayudar a decidir si los predictores se pueden sustituir de la forma indicada arriba es una *gráfica de variable agregada*. Estas gráficas sirven para medir el efecto o contribución de un término en la función media, ajustada o corregida por la presencia de otros términos. Proveen una forma de visualizar la adición de una variable al modelo de regresión lineal, es como una forma de "ver" la estadística t de un coeficiente de la regresión.

Para construir una gráfica de variable agregada, suponemos que se tiene un modelo de regresión lineal que relaciona y con $x_1, x_2, \ldots, x_{p-1}$ y se quiere agregar un predictor x_p. Lo que se requiere entonces es obtener los residuales $\hat{e}(y|x_1, \ldots, x_{p-1})$. Estos residuales representan la porción de y que no es explicada por los predictores originales. Posteriormente, se obtienen los residuales $\hat{e}(x_p|x_1, \ldots, x_{p-1})$, que corresponden a la parte de x_p que no es explicada por los predictores originales. La gráfica de variable agregada es la gráfica de estos residuales, y si esta gráfica muestra una pendiente, quiere decir que aún hay algo en x_p que ayuda a explicar la respuesta.

El primer paso en regresión gráfica consiste en asegurarse que la condición PRL se

cumple al menos aproximadamente. El siguiente paso consiste en encontrar nuevos predictores que son combinaciones lineales de los originales, utilizando los métodos para reducción de dimensión que se han mostrado anteriormente. Para elegir las combinaciones lineales con más información disponible, se construyen p predictores nuevos que son combinaciones lineales *no correlacionadas* de los predictores originales de la siguiente manera:

- La primera combinación lineal que se puede utilizar es la que se obtiene por el método de mínimos cuadrados ordinarios, $g_1 = \mathbf{b}'_{OLS}x$ Si la regresión tiene estructura D1 y los predictores satisfacen la condición PRL, entonces toda la información de la regresión estará contenida en g_1.
- La segunda combinación lineal se toma como $g_2 = \mathbf{b}'_2 x$ condicionada a que no esté correlacionada con g_1.
- La tercera combinación lineal no deberá estar correlacionada a g_1 y g_2, y así sucesivamente.

Las combinaciones lineales $g_2, g_3, \ldots g_p$ se pueden basar en los métodos que se han discutido previamente, por ejemplo en SIR o en SAVE.

Ejemplo

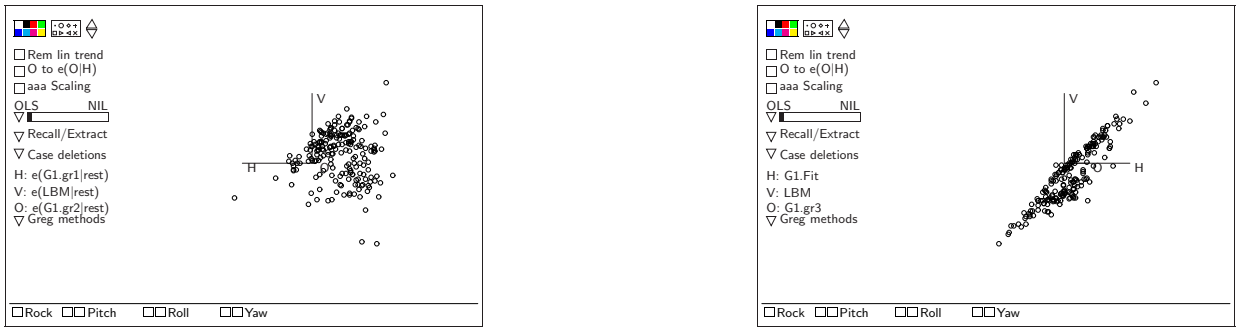
Para mostrar el proceso de regresión gráfica, consideraremos un conjunto de datos reales obtenidos del Instituto Australiano del deporte (AIS). Se puede ver que estos datos no violan severamente el supuesto PRL.

Para obtener los predictores gráficos, se puede seleccionar el submenú **Graphical regression** en el menú **Graph& Fit**. De la salida se obtiene el siguiente resultado:

```
Data set = AIS, Name of Fit = G1
Greg Regression
```

```
Response       = LBM
Terms          = (Ht Wt RCC)
Uncorrelated predictors defined by pHd
The active predictors are the following linear combinations of the
original predictors:
             Ht      Wt     RCC
G1.Fit    0.040   0.117   0.992
G1.gr1    0.008  -0.024   1.000
G1.gr2   -0.311   0.169   0.935
```

Lo que sigue es obtener las gráficas 3D de variables agregadas. Eligiendo las últimas dos varibles $gr2$ y $gr3$ juzgamos que la gráfica muestra estructura 1D, como puede verse en la dirección mostrada en la Figura 4.4. Entonces seleccionamos **Dimension 1** en el menú que aparece con **Greg Methods**. Este menú procederá a combinar los dos predictores gráficos en el predictor gr_3. La siguiente gráfica es una gráfica 3D de *LBM* versus (Fit, gr_3), mostrada en la Figura 4.4.



a. Estructura 1D

b. Estructura 2D

Figura 4.4: Gráficas de variable agregada para los datos AIS

Juzgamos que esta gráfica tiene estructura 2D, ya que podemos ver que hay dos tendencias lineales con diferentes pendientes y por lo tanto se requieren dos combinaciones

lineales diferentes para caracterizar la función media de *LBM* versus *Fit* y gr_3. La conclusión obtenida es que la regresión completa tiene al menos estructura 2D. Suponiendo que esta es la estructura 2D correcta, La gráfica resumen suficiente final está dada por la gráfica tridimensional de *LBM* sobre (Fit, gr_3), donde gr_3 es una combinación lineal de los otros dos predictores gráficos. En particular, la vista 2D mostrada en la Figura 4.4 puede ser una buena gráfica resumen. La gráfica sugiere que hay dos diferentes regresiones involucradas.

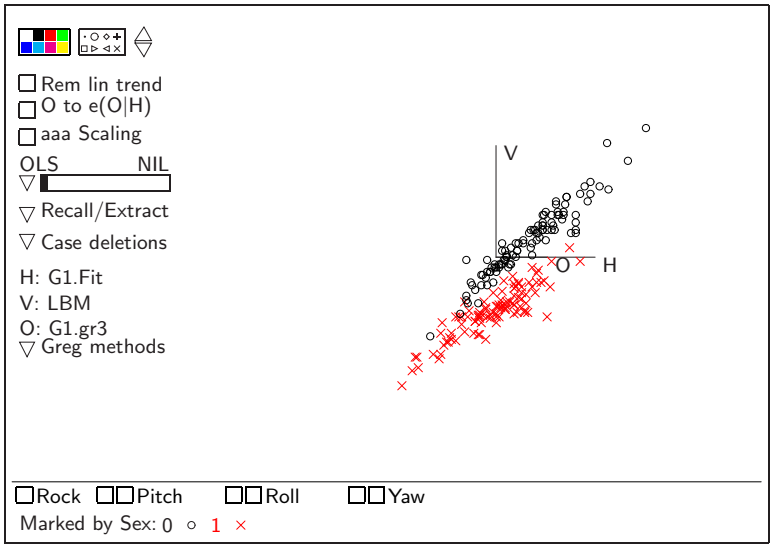


Figura 4.5: Gráfica resumen para los datos del AIS, marcados por sexo.

La Figura 4.5 muestra que los hombres y mujeres requieren diferentes funciones de regresión.

✠

Capítulo 5

Conclusiones y problemas abiertos.

El tema de reducción de dimensión es un tema importante que ha tenido un desarrollo muy fuerte en los últimos años. En regresión, en particular, se ha avanzado considerablemente gracias a la disponibilidad de herramientas computacionales que permiten analizar un problema en forma interactiva, y que ha facilitado la implementación de metodología tanto gráfica como numérica.

En esta tesis se han mostrado algunos métodos que han sido desarrollados para reducción de dimensión en el contexto de problemas de regresión y se ha mostrado cómo se implementan computacionalmente tales técnicas para su uso en la práctica. También se ha mostrado como se puede hacer análisis de regresión en forma interactiva usando el programa especializado en regresión `Arc` y cómo se pueden extender algunos de los métodos cuantitativos en `R`.

En los últimos años ha habido un avance importante en el contexto de métodos para la reducción de dimensión usando el marco conceptual de los espacios centrales. Por ejemplo, Cook y Li (CL02) han desarrollado nueva metodología para ser aplicada en el caso particular en el que lo único que es relevante al analista es el estudio de la función media exclusivamente, mostrando que el método de direcciones principales hessianas estima direcciones en un subespacio en particular, y desarrollando un método que ellos

llaman transformaciones hessianas iterativas (IHT por sus siglas en inglés). Por medio de un método iterativo, se pueden estimar vectores en el subespacio de interés. Si β es un vector en tal subespacio, y H es cualquier matriz cuyas columnas generan el subespacio central, entonces los vectores $H\beta, H^2\beta, \ldots, H^s\beta$ todos son vectores que pertenecen al subespacio de interés, para cualquier entero positivo s.

Bura y Cook (BC02) proponen un nuevo método de estimación para la dimensión de un problema de regresión basado en estimación de curvas paramétricas a las regresiones inversas de cada predictor en la respuesta. A este método se le llama PIR (parametric Inverse regression).

Otra área importante es la estimación de funciones media cuando hay predictores categóricos. En este contexto, se han hecho avances utilizando el método pHd del que se han creado varias versiones. Una versión llamada pHd cuadrático (CL95) es útil en el contexto de modelos de diseños experimentales. Yin y Cook (YC02) aplican la idea de subespacios centrales en el contexto de análisis discriminante y clasificación.

Un tema que aún queda por desarrollar es la aplicación de las ideas de subespacio central y dimensión en el contexto de las series de tiempo.

Con respecto a las pruebas de dimensión, ha habido progresos recientes en relación a las distribuciones de las estadísticas de prueba. Por ejemplo, la prueba asintótica para SAVE (YW04) estuvo por ser implementada computacionalmente hasta 2005, donde Shao, Cook y Weisberg (YSW07) presentaron una prueba factible computacionalmente para probar la dimensión del subespacio central en un problema de regresión basado en SAVE. Bentler y Xie (BX00) propusieron correcciones a las pruebas originales de Cook y Li que son más fáciles de calcular y son más robustas.

No cabe duda que este campo de investigación propone muchas nuevas ideas para explorar y desarrollar en los próximos años, y que paulatinamente irá tomando más importancia, debido a lo significativo que se ha vuelto el análisis de bases de datos masivas para encontrar patrones de comportamiento y relaciones entre variables y donde defini-

tivamente es importante la reducción de la dimensión para poder convertir los datos en información.

Apéndice A

Resumen de funciones comúnmente usadas

A.1. Tabla de funciones

A.1.1. Listas

Concepto	Sintaxis	Descripción
list	(list elementos)	función con la cual se identifica una lista de elementos.
def	(def nombre (list datos))	se utiliza para asignarle nombre a una lista de datos.
iseq	(iseq n m)	genera una lista de números enteros consecutivos desde n hasta m.
rseq	(rseq a b k)	genera una lista de k números reales igualmente especiados entre a y b.
repeat	(repeat lista patrón)	genera secuencias de una lista con un patrón particular.
select	(select lista elemento)	selecciona un elemento o un grupo de un lista o vector.
remove	(remove elemento lista)	deselecciona un elemento o un grupo de un lista de datos.
which	(which lista)	obtiene los indices de los elementos donde la lista no sea NIL.
append	(append x y z)	combina varias listas cortas x y z en una larga.
setf	(setf cambs sustits)	reemplaza de una lista los elementos cambs por los sustits.

A.1.2. Aritméticas

Concepto	Sintaxis	Descripción
;		caracter correspondiente a los comentarios.
+	(+ argumentos)	retorna la suma de los argumentos.
-	(- primer subsecuentes)	resta del primer número todos los subsecuentes.
*	(* argumentos)	obtiene el producto de sus argumentos.
/	(/ primer subsecuentes)	divide el primer número por cada uno de los subsecuentes.
^ó **	(^base potencia)	calcula la base elevada a una potencia.
abs	(abs número)	obtiene el valor absoluto de un número.
pi		variable referente a la constante predefinida π.
log	(log argumentos)	calcula el logaritmo natural de los argumentos, principalmente utilizado para realizar trasformaciones.
exp	(exp potencia)	calcula el número e elevado a una potencia.
sqrt	(sqrt número)	calcula la raiz cuadrada de un número.
max	(max argumentos)	regresa el número más grande de sus argumentos.
min	(min argumentos)	regresa el menor número de sus argumentos.
random	(random número)	genera un número pseudo-aleatorio distribuido uniformente entre cero y un número.
ceiling	(ceiling numreal)	devuelve el número entero más pequeño mayor a numreal.
floor	(floor numreal)	devuelve el número entero más grande inferior a numreal.

A.1.3. Estadísticas

Concepto	Sintaxis	Descripción
mean	(mean lista)	promedio de una lista de números.
median	(median lista)	mediana de una lista de números.
standart-deviation	(standart-deviation lista)	desviación estándar de una lista de datos.
interquartile-range	(interquartile-range lista)	rango o amplitud del intervalo entre los cuartiles aplicado a una lista de datos.
histogram	(histogram lista)	obtiene un histógrama de un conjunto de datos.
boxplot	(boxplot muestras)	obtiene un(os) diagrama(s) de caja de un(as) muestra(s) de datos.
plot-point	(plot-point x y)	genera una gráfica de dispersión o scatterplot de las variables x contra y.
plot-line	(plot-line x y)	genera una gráfica de dispersión con los puntos conectados por líneas de las variables x contra y.
plot-function	(plot-function función rango)	grafica una función de una variable en un rango especificado.

A.1.4. Lógicas

Concepto	Sintaxis	Descripción
T		referente al valor lógico verdadero.
NIL		referente al valor lógico falso o resultado nulo de alguna operación.
<	(<números)	regresa T sí los números están en orden estrictamente decreciente. NIL en otro caso.
<=	(<= números)	regresa T sí los números están en orden no decreciente. NIL en otro caso.
=	(= números)	regresa T sí los números son todos iguales. NIL en otro caso.
/=	(/= datos)	obtiene T sí dos datos adyascentes son diferentes. NIL en otro caso.
>=	(>= números)	regresa T sí los números están en orden no creciente. NIL en otro caso.
>	(>números)	regresa T sí los números están en orden estrictamente creciente. NIL en otro caso.

Referencias

E. Bura and R. Dennis Cook. Estimating the structural dimension of regressions via parametric inverse regression. *Journal of the American Statistical Association*, 21:867–889, 2002.

David Betz. An xlisp tutorial. *BYTE*, page 221, 1985.

David Betz. Xlisp: An experimental object-oriented programming language. *Reference Manual for XLISP version 2.0*, 1988.

Mark Britten-Jones. Statistical portfolio optimization part 1: The global minimum variance portfolio. working paper, London Businness School, February 1998.

Peter M. Bentler and Jun Xie. Corrections to test statistics in principal Hessian directions. *Statistics & Probability Letters*, 47:381–389, 2000.

George Cassela and Roger L. Berger. *Statistical Inference*. Duxbury Press, 1990.

F. Chiaramonte and R. D. Cook. On foundations of regression graphics. working paper, University of Minnesota, School of Statistics, 1997.

Ching Shui Cheng and Ker Chau Li. A study of the method of principal hessian direction for analysis of data from designed experiments. *Statistica Sinica*, 5:617–639, 1995.

R. D. Cook and Bing Li. Dimension reduction for conditional mean. *Annals of Statistics*, 30:455–474, 2002.

William Cleveland. Robust locally weighted regression and smoothing scatterplots. *Journal of the American Statistical Association*, 74:829–836, 1979.

R. D. Cook and C. J. Nachtsheim. Re-weighting to achieve elliptically contoured covariates in regression. *Journal of the American Statistical Association*, 89:592–600, 1994.

R. Dennis Cook. *Regression Graphics. Ideas for studying regression trough graphics*. Wiley Series in Probability and Mathematical Statistics. John Wiley & Sons, New York, 1998.

R. Dennis Cook and Sanford Weisberg. *An Introduction to Regression Graphics*. John Wiley, 1994.

R. Dennis Cook and Sanford Weisberg. *Applied regression including computation and graphics*. Wiley & Sons, 1999.

P. Diaconis and D. Freedman. Asymptotics of graphical projections pursuit. *The Annals of Statistics*, 12:793–815, 1984.

O.D. Duncan. *A socioeconomic index for all occupations: Occupations and Social Status*. Reiss, Jr., 1961.

Morris L. Eaton. A characterization of spherical distributions. *Journal of Multivariate Analysis*, 20:272–276, 1986.

John Fox. *Applied Regression Analysis, Linear Models and Related Methods*. SAGE Publications, California, 1997.

P. Hall and K. C. Li. On almost linearity of low dimensional projections from high dimensional data. *The Annals of Statistics*, 21:867–889, 1992.

Ross Ihaka and Robert Gentleman. R: A language for data analysis and graphics. *Journal of Computational and Graphical Statistics*, 5(3):299–314, 1996.

K. C. Li and N. Duan. Regression analysis under link violation. *Annals of Statistics*, 17:1009–1052, 1989.

Harry Markowitz. A simplex method for the portfolio selection problem. Technical Report 27, Cowles Foundation, Yale University, 1957. available at http://ideas.repec.org/p/cwl/cwldpp/27.html.

Jeffrey S. Simonoff. *Smoothing Methods in Statistics*. Springer Series in Statistics. Springer, 1996.

Luke Tierney. *Lisp-Stat. An Object-Oriented Environment for Statistical Computing and Dynamic Graphics*. John Wiley & Sons, New York, 1990.

Ruey S. Tsay. *Analysis of Financial Time Series*. Wiley, 2002.

Sanford Weisberg. Inverse regression in R. 2001.

Xiangrong Yin and R. D. Cook. Dimention reduction for the conditional kth moment in regression. *Journal of the Royal Statistical Society*, 64:159–175, 2002.

Xiangrong Yin. *Dimension Reduction Using Inverse Third Moments and Central k-Th Moment Subspaces*. PhD thesis, School of Statistics, University of Minnesota, 2000.

R. Dennis Cook Yongwu Shao and Sanford Weisberg. Marginal tests with sliced average variance estimation. *Biometrika*, 94:285–296, 2007.

Xiangron Yin and Sanford Weisberg. Test of dimension with sliced average variance estimation. 2004.