Trabajo Fin de Máster

Detección de derivas de sensores en series temporales

Sara Garcia Crespo

Máster en Técnicas Estadísticas

Curso 2020-2021

Propuesta de Trabajo Fin de Máster

Título en galego: Detección de derivas de sensores en series temporais
Título en español: Detección de derivas de sensores en series temporales
English title: Sensor drift detection in time series
Modalidad: Modalidad B
Autora: Sara Garcia Crespo, Universidad de Santiago de Compostela
Director: Manuel Febrero Bande, Universidad de Santiago de Compostela
Tutor: Andrés Gómez Tato, Centro de Supercomputación de Galicia (CESGA)
Breve resumen del trabajo: El objetivo del trabajo consiste en la identificación y evaluación de métodos y técnicas que permitan detectar y cuantificar derivas en sensores. Utilizando los datos proporcionados por el CESGA, realizamos un estudio práctico donde se evalúan los métodos analizados para verificar si reaccionan adecuadamente a las derivas. Esto resulta imprescindible para el correcto funcionamiento de la maquinaria en los sectores industriales ya que la mayoría de las veces no pueden ser recalibradas periódicamente.
Recomendaciones:

Don Manuel Febrero Bande, catedrático Área de Estadística e Investigación Operativa de la Universidad de Santiago de Compostelay don Andrés Gómez Tato, cargo 1 de Centro de Supercomputación de Galicia (CESGA), informan que el Trabajo Fin de Máster titulado

Detección de derivas de sensores en series temporales

fue realizado bajo su dirección por doña Sara Garcia Crespo para el Máster en Técnicas Estadísticas. Estimando que el trabajo está terminado, dan su conformidad para su presentación y defensa ante un tribunal.

En [lugar], a xx de [mes] de 20xx.

El director:

Don Manuel Febrero Bande

El tutor:

Don Andrés Gómez Tato

La autora:

Doña Sara Garcia Crespo

Agradecimientos

En primer lugar, agradecer a mi familia el apoyo continuo. Gracias a ellos he tenido la oportunidad de realizar el máster en estadística que me propuse y vivir esta nueva y tan valiosa etapa de mi vida que me ha aportado una gran experiencia profesional y personal.

También agradecer tanto al director como al tutor del trabajo el tiempo extra dedicado y los conocimientos que me han transmitido. Me hubiera encantado realizarlo de una manera más calmada para poder haber proporcionado mejores resultados, y de forma presencial, no con 980 kilómetros de por medio.

Índice general

Resumen

Resumen en español

En los sectores industriales los sensores utilizados para el control de los equipamientos pueden no ser del todo precisos y la información aportada en sus proyectos podría no ser válida. Los sensores suelen estar absortos en las maquinarias y pueden presentar desvíos sistemáticos a causa del paso del tiempo, envejecimiento o suciedad.

Con este trabajo se pretende detectar cualquier deriva o anomalía que pueda existir en los sensores. Debido a la complejidad existente en la recalibración periódica de los dispositivos, maquinarias y procesos en las industrias, la detección de derivas de sensores resulta una tarea imprescindible para el correcto funcionamiento de estos.

Para ello, se realiza, en primer lugar, un rastreo bibliográfico de los métodos existentes para poder realizar un análisis de desvíos. Se introducen diferentes modelos de regresión con sus técnicas de detección de atípicos, si las hubiera, y se plantea un posible procedimiento que nos ayude a identificar esos datos y que sirva para cualquier modelo. Finalmente, se realiza el estudio práctico con los datos aportados por el CESGA y se comprueba si los métodos aplicados son adecuados y si funcionan correctamente.

English abstract

In industrial sectors, the sensors used to control equipment may not be entirely accurate and the information provided in their projects may not be valid. The sensors are often absorbed in the machinery and can show systematic drifts due to the passage of time, aging or dirt.

With this work we are trying to detect any drift or anomaly that may exist in the sensors. Due to the complexity in the periodic recalibration of devices, machinery and processes in industries. The detection of sensor drifts is an essential task for the correct operation of these.

For this purpose, first of all, a literature review of the methods is carried out to perform drift analysis. Different regression models are introduced with their outlier detection techniques and a possible procedure is proposed to help us identify these data and that can be used for any model. Finally, a practical study is carried out with the data provided by CESGA and it is checked whether the applied methods are adequate and if they work correctly.

Prefacio

La Estadística tiene como uno de los principales objetivos construir modelos que se acerquen lo máximo posible a la realidad. Esto se consigue mediante un estudio estadístico cuyo proceso consiste, en primer lugar, en plantear el problema a analizar. En segundo lugar, recopilación de toda la información necesaria, así como la recogida de datos. Por último, se analizan esos datos y se interpretan, obteniendo las conclusiones de los resultados obtenidos.

En el ámbito industrial, ámbito al que pertenecen nuestros datos, es de vital importancia hallar aquellos datos que puedan ocasionar anomalías o desvíos en los sensores incorporados en la maquinaria industrial. Si no se detectan estas derivas a tiempo, provocará un mal funcionamiento en los equipamientos, los cuales acabarán no siendo útiles.

La metodología llevada a cabo en este trabajo fue la siguiente: se planteó el problema a abordar, estudiamos la situación y se plantearon diversas soluciones, llegando al objetivo principal de este trabajo que reside en probar varios métodos para detectar observaciones anómalas en un conjunto de datos concreto y poder aplicarlo, a posteriori, a casos reales como el nuestro.

El presente trabajo se divide en 5 capítulos. En los 3 primeros se plantean posibles métodos para un buen estudio de los datos. Son notoriamente distintos pero con el objetivo común de evaluar varios modelos que detecten observaciones atípicas.

En el capítulo 1 se estudian los **Modelos Lineales**, así como la validación de éstos, la estimación de sus parámetros y la detección de datos atípicos e influyentes para estos modelos. Se plantea esta alternativa en primer caso porque ofrece una gran simplicidad de interpretación y su puesta en funcionamiento tiene bajo costo computacional y muchas veces son utilizados previamente a la implementación de otros modelos ya que identifican mejor la aportación de cada variable predictora a la variable dependiente. Estos modelos parten con su propia técnica de diagnosis de observaciones atípicas caracterizada por el análisis de los residuos.

En el segundo capítulo se realiza un breve estudio de los **Modelos Aditivos** con sus respectivas funciones y ajustes. Esta otra alternativa se propuso por los resultados que ofrecen ya que son considerablemente aceptables y disponen de una mayor facilidad de interpretación que los siguientes que vamos a comentar y sin problemas de identificabilidad.

En el capítulo 3 se analizan las **Redes Neuronales** y su aplicación a situaciones como la nuestra mediante modelos que funcionan de manera análoga a unas redes neuronales centradas en aplicaciones biológicas. A pesar de la no identificabilidad de los parámetros, la débil interpretación de los modelos y el alto coste computacional, la red neuronal predice mejor que la regresión lineal.

Debido a que tanto con los Modelos Aditivos como Redes Neuronales no se han podido hallar técnicas concretas que encuentren esas anómalas observaciones, en el capítulo 4 se plantea un procedimiento bastante general cuya implementación podría ser de utilidad para detectar outliers en cualquiera de estos modelos. En este capítulo se lleva a cabo el estudio de simulación con los datos sintéticos que se realizará a traves del programa estadístico R, proyectando los resultados donde aplicaremos los métodos más eficientes a los datos del CESGA, acabando con una conclusión de las soluciones obtenidas.

Capítulo 1

Modelo lineal general

1.1. Introducción

Los modelos se emplean para comprender lo que sucede a nuestro alrededor, a partir de la observación de los acontecimientos, pudiendo realizar, incluso, predicciones sobre ellos. Los modelos de regresión sirven para representar la dependencia de una variable Y (variable respuesta, dependiente u output) con respecto a otra variable X (variable explicativa, independiente o input). Por tanto, los objetivos principales de los modelos de regresión serían los siguientes: conocer cómo la variable Y depende de X y, una vez construido el modelo de regresión, realizar predicciones del valor de Y conociendo el valor de X.

De forma general, el modelo de regresión trataría de expresar la media condicionada de la variable respuesta en función del valor que tome la variable explicativa y se detalla mediante la siguiente función:

$$f(x) = E(Y/X = x)$$

para cada posible valor x de X.

Los modelos se pueden clasificar como determinísticos o estocásticos. Con los determinísticos se podría predecir con total exactitud la variable respuesta siempre que las variables explicativas sean conocidas. Sin embargo, en la vida real una predicción tan exacta es prácticamente imposible y, por ello, se necesitan modelos estocásticos que incorporen un error impredecible ocasionado por la influencia de otras variables incontrolables, por errores de medida o por la aleatoriedad proveniente de la variable respuesta.

Con la modelización estadística se pretende estudiar la relación existente entre Y, y una serie de p variables explicativas $X_1, X_2, ..., X_p$ que se puede definir matemáticamente con la siguiente expresión:

$$Y = f(X_1, X_2, ..., X_p) + \epsilon$$

siendo f una función desconocida que trata de modelizar la mejor relación posible entre las variables, y ϵ es el error aleatorio, independiente de las variables explicativas y con media cero.

1.2. Modelo lineal simple

El modelo más utilizado es el del tipo lineal donde se modeliza la variable o variables objetivos a través de una combinación lineal de las variables explicativas. Los modelos lineales suelen ser simples y generalmente aportan una descripción adecuada sobre cómo los inputs afectan a los outputs.

Como hemos comentado anteriormente, el modelo de regresión definido de la forma más general sería

$$Y = f(X) + \epsilon$$

donde ϵ es el error que cumple $E(\epsilon/X = x) = 0$ para todo x.
Lo que ocurre en un modelo lineal simple es lo siguiente:

$$E(Y|X) = \alpha + \beta X.$$

Para construir el modelo adecuado hay que valorar ciertos aspectos que pueden influir, como el número de variables explicativas o variables respuesta, si son continuas o discretas, el tipo de distribución del error o la forma de la función de regresión, la cual puede ser lineal, polinómica, etc.

En esta sección, realizaremos una breve introducción al modelo de regresión más sencillo, es decir, el modelo de regresión lineal simple.

Este modelo debe cumplir ciertas hipótesis básicas, las cuales, en orden de importancia, serían:

• Existencia. Esta es la base principal de cualquier modelo, la existencia de $E(Y|X)$ y $Var(Y|X)$, necesarias para poder plantearse la construcción de modelos de regresión.

• Independencia. Suponiendo que, bajo el modelo de regresión, se obtiene una muestra de n observaciones, los errores $\epsilon_1, ..., \epsilon_n$ son independientes con recíproca correspondencia.

• Linealidad. La función de regresión debe ser lineal por lo que el modelo se definiría así:

$$Y = \beta_0 + \beta_1 X + \epsilon$$

donde β_0 y β_1 son parámetros desconocidos y ϵ es una variable aleatoria no observable. Esta hipótesis hace que nos situemos ante un modelo paramétrico.

• Homocedasticidad. Sea cual sea el valor de la variable explicativa, la varianza del error no varía, es decir:

$$Var(\epsilon/X = x) = \sigma^2 \quad \textit{para todo } x.$$

• Normalidad. El error tiene distribución normal

$$\epsilon \in N(0, \sigma^2).$$

En conclusión, para definir un modelo de regresión lineal simple, homocedástico, con errores normales e independientes, se necesitará una muestra aleatoria simple de n individuos, que formarán el conjunto de entrenamiento $\{(x_1, y_1), ..., (x_n, y_n)\}$, siendo $x_i = (x_{i1}, ..., x_{ip})$ para $i = 1, 2, ..., n$, y así, obtendremos el modelo

$$Y_i = \beta_0 + \beta_1 x_i + \epsilon_i$$

siendo $\epsilon_1, ..., \epsilon_n \in N(0, \sigma^2)$, independientes.

1.3. Modelos lineales en general

Los modelos de regresión lineal simple tratan de explicar una variable respuesta Y en función de una única variable explicativa X. En cambio, existen situaciones más complejas donde hay más de una variable explicativa y más de una respuesta, pero aqui nos centraremos únicamente en el primer caso. Por tanto, la funcion f se expresaría como

$$f(X) = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + ... + \beta_p X_p$$

Nuestro objetivo es evaluar el grado de contribución de cada una de las variables explicativas en la explicación de Y, y predecir esa variable para algún conjunto de valores de $X_1, X_2, ..., X_p$, por eso, suponemos un modelo de regresión múltiple tal que

$$Y_i = \beta_0 + \beta_1 x_{i,1} + \beta_2 x_{i,2} + ... + \beta_{p-1} x_{i,p-1} + \epsilon_i, \quad i = 1, ...n$$

donde β_0 es el intercepto, $(\beta_1, ..., \beta_p)$ los parámetros que acompañan a las variables independientes, y ϵ_i errores normales independientes con media cero y varianza σ^2. Como se puede observar, se asume que la dependencia de $E(Y|X_1, ..., X_p)$ es lineal.

Además de la ampliación realizada anteriormente respecto la modelización simple clásica de la sección anterior, también se podría llevar a cabo una extensión mas general, permitiendo a la variable respuesta depender de múltiples variables predictoras como hemos dicho antes, pero con la libertad de que podrían ser transformaciones de los predictores originales, por ejemplo:

- $Y_i = \beta_0 + \beta_1 x_i + \beta_2 x_i^2 + \beta_3 x_i^3 + \epsilon_i$, donde la relación existente entre y y x es la del modelo cúbico.
- $Y_i = \beta_0 + \beta_1 x_i + \beta_2 z_i + \beta_3 \log(x_i z_i) + \epsilon_i$, modelo en el que y depende de las variables x y z y del log de su producto.

Cada uno de estos modelos se puede estimar como los lineales porque los parámetros β_j y los términos ϵ_i se presentan en el modelo de una forma lineal, a pesar de que las variables predictoras puedan aparecer en el modelo de manera no lineal.

A continuación, consideramos el modelo lineal general en notación matricial, tal que,

$$Y = X\beta + \epsilon$$

en el que X es una matriz no aleatoria, β es un vector de parámetros y $\epsilon \in N(0, \sigma^2 I)$ siendo σ^2 la varianza del error e I la matriz identidad.

Por tanto, bajo esa última expresión, se asume homocedasticidad, normalidad e independencia de los errores.

1.3.1. Estimación de los parámetros

Considerando una formulación más genérica del modelo lineal general como

$$f(X) = \beta_0 + \sum_{j=1}^{p} X_j \beta_j,$$

debemos estimar el vector de parámetros β_j que es desconocido y la varianza del error σ^2.

- **Estimación de los parámetros β_j.**

Los parámetros de estos modelos se pueden estimar encontrando los valores de β_j que hagan que los modelos se ajusten de la mejor forma posible a los datos observados.

Existen dos criterios para la estimación de los parámetros: Máxima verosimilitud (*maximum likelihood*) y mínimos cuadrados (*least squares*). Si se cumplen las hipótesis básicas anteriores, el método de mínimos cuadrados, que explicaremos a continuación, será equivalente al de máxima verosimilitud.

El método de estimación de mínimos cuadrados para el vector de parámetros β_j desconocido, consiste en escoger los coeficientes $\beta = (\beta_0, \beta_1, ..., \beta_p)^T$ que minimicen la suma de cuadrados residuales

$$RSS(\beta) = \Sigma_{i=1}^{N} (y_i - \beta_0 - \Sigma_{j=1}^{p} x_{ij} \beta_j)^2.$$

Este criterio será razonable estadísticamente si las observaciones de entrenamiento (x_i, y_i) son obtenidas de manera independiente y aleatoria de su población. Incluso si las x_i´s no se sacan de forma aleatoria, el criterio continuaría siendo válido si las y_i´s son independientes dados los inputs x_i.

También podemos escribir la fórmula anterior como

$$RSS(\beta) = (Y - X\beta)^T(Y - X\beta)$$

denotando por X la matrix de n x $(p+1)$ con un vector de inputs cada fila. Como nuestro objetivo es minimizar la función anterior, derivamos respecto de β e igualamos a cero y, así, obtenemos las ecuaciones normales de regresión

$$X^T X\beta = X^T Y$$

cuya solución es el estimador de β por mínimos cuadrados, tal que:

$$\hat{\beta} = (X^T X)^{-1} X^T Y.$$

Una vez obtenidos los estimadores de los parámetros, $\hat{\beta}$, podremos calcular las predicciones o valores ajustados para los individuos de la muestra, de la forma siguiente:

$$\hat{Y} = X\hat{\beta} = X(X^T X)^{-1} X^T Y$$

• **Estimación de la varianza del error.**

Para estimar la varianza del error, al no ser observables los errores, se emplean los residuos. Esos residuos se definen como la diferencia entre las observaciones y las predicciones, es decir,

$$\epsilon_i = Y_i - \hat{Y}_i = Y_i - x_i\hat{\beta} \quad i \in 1, ..., n.$$

Por tanto, el estimador de la varianza del error sería:

$$\hat{\sigma}^2 = \frac{1}{n-p-1}\sum_{i=1}^{N}\hat{\epsilon}_i^2 = \frac{1}{n-p-1}\sum_{i=1}^{N}(Y_i - \hat{Y}_i)^2 = \frac{1}{n-p-1}\sum_{i=1}^{N}(Y_i - x_i\hat{\beta})^2 = \frac{RSS}{n-p-1}$$

Cabe mencionar que se emplea como denominador $(n-p-1)$ para que el estimador de la varianza sea insesgado.

1.3.2. Validación de un modelo de regresión

Cuando nos encontramos ante un proceso de búsqueda del mejor modelo de regresión entre varios, el modelo seleccionado debe gozar de buenas propiedades como la sencillez, el buen ajuste y la eficiencia de las estimaciones y predicciones. También, se deben cuestionar y comprobar ciertas suposiciones básicas como las hipótesis de linealidad de la función de regresión, homocedasticidad, normalidad e independencia de los errores, ya que se pueden cometer graves errores en las conclusiones.

Empezaremos definiendo el coeficiente de determinación que se interpreta como una medida de ajuste de la regresión. Esta medida refleja la proximidad de las observaciones al modelo. Es decir, un coeficiente de determinación alto reflejará un modelo de regresión que ajusta bien los datos y que es muy útil para efectuar predicciones, sin embargo, por sí solo, no indica si el modelo es correcto.

El coeficiente de determinación se define como la proporción de varianza explicada y se calcula a través de la siguiente expresión:

$$R^2 = 1 - \frac{RSS}{TSS}$$

siendo $RSS = \sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2$ la suma residual de cuadrados, donde $\hat{Y}_i$ son las predicciones en base al modelo, y $TSS = \sum_{i=1}^{n}(Y_i - \bar{Y})^2$ la suma total de cuadrados con $\bar{Y}$ el valor medio de la variable Y .

Si existen modelos con distinto numero de variables explicativas que queramos comparar, se debe utilizar el coeficiente de determinación ajustado , definido como:

$$R^2 ajustado = 1 - \frac{RSS/(n-p-1)}{TSS/(n-1)}$$

siendo p el número total de parámetros y $n-p-1$ los grados de libertad de RSS.

• Diagnosis del modelo

Para comprobar si se cumplen las hipótesis de homocedasticidad, normalidad e independencia llevaremos a cabo un análisis de los residuos de la regresión, conocido como la diagnosis del modelo. En regresión simple, se representan las observaciones junto con el ajuste del modelo y se procede a la validación de éste. En cambio, en el caso múltiple, representamos los residuos estandarizados en el eje de ordenadas para cada una de las variables explicativas.

• Colinealidad

Los problemas de colinealidad surgen cuando el ajuste del modelo se encuentra con ciertos impedimentos a la hora de distinguir el efecto de cada una de las variables explicativas sobre la respuesta, hecho que se produce cuando las variables explicativas presentan correlación entre sí.

Sabiendo que R_j^2 es el coeficiente de determinación de la j-ésima variable explicativa, un buen indicador de este problema de colinealidad sería el factor de inflación de la varianza $1/(1-R_j^2)$, el cual se considera preocupante cuando toma un valor superior a 5.

• Métodos de selección de variables

La prioridad de estos métodos es buscar un modelo que, siendo lo más sencillo posible, sea el que mejor ajuste los datos, ya que cuando tratamos con muchas variables explicativas, se plantean múltiples posibilidades debido a las combinaciones, operaciones e interacciones entre éstas.

→ Métodos Forward y Backard.

Estos métodos respetan la estructura jerárquica del modelo. Los primeros, parten del modelo más sencillo posible y van añadiendo el término que resulte más significativo al introducirlo en el modelo, hasta que considere que no procede añadir más términos, es decir, cuando todos los elementos fuera del modelo son no significativos. Los Backward, en cambio, parten de un modelo que incorpora todos los efectos que puedan llegar a influir en la variable respuesta y va suprimiendo los términos menos significativos hasta que todos los coeficientes sean significativos. Además existen métodos que combinan los dos métodos anteriores.

La condición que hemos descrito anteriormente para suprimir o incluir algún término era la que se basa en la significación. Además, existen otros criterios que construyen una medida global de cada modelo. Cabría destacar el Criterio de Información de Akaike (AIC) y el Criterio de Información de Bayes (BIC) que se formulan de la siguiente manera:

$$AIC = -2\log(verosimilitud) + 2p$$

$$BIC = -2\log(verosimilitud) + p\log(n)$$

La idea de estos criterios es encontrar el modelo con una verosimilitud grande y pocos parámetros y, para ello, el valor del AIC o BIC debe ser pequeño. Debido a que la verosimilitud y el número de parámetros suelen estar contrapuestos, el objetivo es hallar aquel modelo que incluya los parámetros más útiles para aumentar la verosimilitud, y después el procedimiento de búsqueda podría ser backward, forward o alguno que mezcle ambos. Cabe mencionar que el R^2 ajustado, definido anteriormente, es otro de los criterios globales que existen.

1.3.3. Detección de datos atípicos e influyentes

• **Atípicos**

Un dato es considerado **atípico** cuando se sospeche que tal observación no sigue el modelo, es decir, se separa considerablemente del comportamiento esperado en base al modelo. En esta sección, comentaremos técnicas para hallar ese tipo de observaciones.

Sabemos que los residuos brutos, $\hat{\epsilon_i}$, son los errores estimados; sin embargo, para la detección de valores atípicos, se utilizan los residuos estandarizados, r_i, ya que éstos últimos son preferibles para la diagnosis de un posible incumplimiento del modelo y por tener más probabilidad de cumplir que tienen varianza común (uno), y se definen como:

$$r_i = \frac{\hat{\varepsilon}_i}{\hat{\sigma}\sqrt{1-h_{ii}}}.$$

Se considera que una observación es anómala, atípica o no corresponde con el modelo cuando su residuo estandarizado sea muy grande, en valor absoluto. Se clasifican como grandes aquellos valores superiores a 2 o inferiores a -2, que serían cuantiles de una normal estándar que contienen a más de un 95 % de los individuos.

Este método se podría mejorar con unos residuos que reflejaran mejor la divergencia de la observación en cuestión respecto del modelo. Los cuales son denominados residuos estudentizados, cuya fórmula

$$t_i = \frac{Y_i - Y_{(i)}}{\sqrt{\hat{Var}(Y_i - Y_{(i)})}}$$

refleja la estandarización de $Y_i - Y_{(i)}$, siendo $Y_{(i)} = x_i\hat{\beta}_{(i)}$ y $\hat{\beta}_{(i)}$ el vector de parámetros estimados con todos los individuos menos el i-ésimo. Este residuo se basa en la idea de ajustar el modelo sin el individuo en cuestión, llamado(x_i, Y_i) y se construye como la diferencia de Y_i respecto del ajuste obtenido para x_i, teniendo en cuenta los demás individuos.

También, existe otra expresión de los residuos estudentizados que no necesita el ajuste de la regresión sin el individuo i-ésimo, conocida como

$$t_i = \frac{r_i}{\sqrt{\frac{n-p-r_i^2}{n-p-1}}}.$$

• **Influyentes**

Una observación será **influyente** cuando produzca grandes variaciones en el ajuste. Un dato atípico tiene mucha probabilidad de llegar a ser influyente, pero no siempre lo es.

Para poder considerar una observación como influyente habrá que tener en cuenta el apalancamiento o leverage y el distanciamiento de la respuesta observada respecto del modelo. La Distancia de Cook es una buena medida, la cual mide las diferencias entre los ajustes con todos los datos, $\hat{Y}_j$, y los ajustes con todos los datos menos el i-ésimo, $\hat{Y}_{j(i)}$, y su expresión matemática sería:

$$D_i = \frac{\sum_{j=1}^{n}(\hat{Y}_j - \hat{Y}_{j(i)})^2}{p\,\hat{\sigma}^2}$$

cuyo denominador se encarga de estandarizar la distancia.

Adicionalmente, se podría explicar la distancia de Cook en función de los residuos estandarizados, r_i, y el leverage, h_{ii}. Es este caso, la expresión quedaría de la siguiente manera

$$D_i = \frac{1}{p} r_i^2 \frac{h_{ii}}{1 - h_{ii}},$$

por esta razón, cuanto mayor sea el residuo estandarizado y el leverage, mayor será la distancia de Cook. Esta distancia se considerará preocupante si el valor resultante es mayor de $0,5$ o de 1.

Capítulo 2

Modelos Aditivos Generalizados

2.1. Introducción

Los modelos lineales tradicionales, como los vistos en el capítulo anterior, a pesar de que sean sencillamente atractivos, no son aconsejables ya que los efectos en la vida real suelen no ser lineales.

El problema que estos modelos pueden originar es que, en algunos casos, es muy restrictivo suponer que la variable respuesta se relaciona linealmente con las variables explicativas si la relación no es lineal. Debido a que muchas veces han surgido problemas a la hora de formular los modelos paramétricos porque no se ha encontrado ninguno que sea adecuado para los datos, surgieron los modelos no paramétricos.

Los modelos no paramétricos ofrecen más flexibilidad que la ofrecida por los paramétricos ya que permiten que la forma funcional de f pueda tomar cualquier función posible. Para la búsqueda de f se utilizan técnicas como las bases de funciones polinómicas, splines cúbicos o de suavizado, que logran la no-linealidad.

A pesar de ello, los modelos no paramétricos no son tan potentes como los paramétricos, y las interpretaciones sobre el grado de contribución de las variables explicativas sobre la variable respuesta no son tan precisas. Además, los ajustes que se realizan para f son tan complejos en algunas ocasiones que resulta difícil comprender la relación entre cada variable explicativa y la variable respuesta.

Así, para identificar efectos de regresión no lineales, surgieron los **Modelos Aditivos Generalizados**. Estos son una combinación de métodos paramétricos y no paramétricos. Son estadísticamente más flexibles que los que se centran únicamente en un ámbito. Por tanto, si nos ubicamos en el ámbito de la regresión, un modelo aditivo generalizado tiene la forma

$$E(Y|X_1, X_2, ..., X_p) = \alpha + f_1(X_1) + f_2(X_2) + ... + f_p(X_p).$$

De la misma manera que antes, Y es la variable respuesta y $X_1, X_2, ..., X_p$ representan las variables predictoras; las $f_j's$ son funciones no paramétricas de suavizado no especificadas.

Estas funciones, algunas de las cuales explicaremos brevemente en la siguiente sección, se estiman de una manera flexible, pudiendo revelar posibles no linealidades en el efecto de X_j. La función estimada sería $\hat{f}_j$ y su base principal es el diagrama de dispersión de suavizado. Habría que indicar que no todas las funciones tienen que ser no lineales, ya que podemos mezclar fácilmente lineales y otras formas paramétricas con términos no lineales. Finalmente, una vez ajustadas éstas, se proporciona un algoritmo para estimar simultáneamente todas las funciones.

2.2. Modelo aditivo generalizado

Trevor Hastie y Robert Tibshirani introdujeron el Modelo Aditivo Generalizado, el cual es una extensión del Modelo Lineal Generalizado (Hastie y Tibshirani, 1986). El Modelo Lineal Generalizado

asume que la influencia de las variables explicativas sobre la variable respuesta es de forma lineal, sin embargo, el **Modelo Aditivo Generalizado** va más allá. Cuando el supuesto de linealidad es inestable, se puede sustituir esa relación lineal por funciones no paramétricas:

$$y_i = \beta_0 + f_1(x_{1i}) + ... + f_p(x_{pi}) + \epsilon_i, \quad i = 1, ..., n.$$

Esta idea es la base en la que se sustenta el Modelo Aditivo. Podemos observar que los efectos de los predictores son aditivos para explicar la respuesta, es decir, cada predictor tiene efectos separados en la explicación de la respuesta.

Se puede representar las p funciones y analizar cómo influyen en la respuesta, puesto que se está representando cada variable por separado.

Los Modelos Aditivos tienen como objetivo modelizar la variable Y, o una característica de esta, a partir de la suma de distintas funciones suaves aplicadas a las distintas variables explicativas. Por lo que estos modelos pueden reemplazar modelos lineales en una amplia variedad de formas, por ejemplo, en una descomposición aditiva de series temporales, como

$$Y_t = S_t + T_t + \epsilon_t$$

donde S_t es la componente estacional, T_t es la tendencia y ϵ es el término de error.

2.3. Funciones de base

En los problemas de regresión, como hemos comentado anteriormente, $f(X) = E(Y|X)$ normalmente es no lineal y no aditiva en X, aún así, representar esa función mediante un modelo lineal es una aproximación muy conveniente ya que los modelos lineales son fáciles de interpretar e incluso puede resultar una aproximación necesaria si queremos evitar un sobreajuste de los datos.

A continuación, explicaremos brevemente algunos métodos que nos permitan trabajar más allá de la linealidad. La principal idea sería poder modificar el vector de inputs $\mathbf{X}$ con variables adicionales, las cuales son transformaciones de X, y así, usar modelos lineales con estas nuevas características de las variables. Vamos a mostrar distintas formas de construir las funciones y cómo suavizar dichas funciones obtenidas.

Para una mejor interpretación, comenzaremos teniendo en cuenta un modelo con una única variable explicativa, tal que

$$y_i = f(x_i) + \epsilon_i, \quad i = 1, ..., n$$

donde y_i es la variable respuesta, x_i, la variable explicativa, ϵ_i son errores independientes con media cero y varianza σ^2 y f es una función suave de las variables x_i.

El propósito principal es convertir el modelo anterior en uno lineal realizando una transformación en la función f. Para ello, esta función se podrá expresar como una combinación de funciones básicas $b_j(x)$ y un vector de parámetros β tal que

$$f(x) = \sum_{j=1}^{q} b_j(x_i)\beta_j.$$

Es decir, el modelo anterior quedaría representado linealmente de la siguiente forma

$$y_i = \sum_{j=1}^{q} b_j(x_i)\beta_j + \epsilon_i, \quad i = 1, 2, ..., n.$$

Especificando el modelo con funciones de suavizado en vez de con relaciones paramétricas se obtiene una interpretación bastante flexible de la dependencia de la respuesta en las covariables.

Lo bueno de este enfoque es que, una vez que las funciones de base b_j han sido determinadas, los modelos acaban siendo lineales en estas nuevas variables.

2.3.1. Base polinómica

Este método ajusta la funcion f mediante un polinomio de grado d a lo largo de todo el recorrido de X, es decir, se define un entorno $[a, b]$ para cada punto x, después estimamos la función de regresión en ese entorno y el ajuste local sería la función ajustada evaluada en x.

El modelo que se obtiene al hacer uso de una función polinómica de grado d es

$$y_i = \beta_0 + \beta_1 x_i + \beta_2 x_i^2 + ... + \beta_d x_i^d + \epsilon_i, \quad i = 1, 2, ..., n.$$

Las bases polinómicas suelen ser muy útiles en aquellas situaciones donde el interés de las propiedades de f se centra en las inmediaciones de un punto específico, sin embargo, cuando las cuestiones de interés se relacionan con f en todo su dominio, [0,1], las bases polinómicas presentan algunos problemas. En estas circunstancias, las bases de Splines funcionan bien , en gran medida porque se puede demostrar que las propiedades teóricas de aproximación son buenas.

2.3.2. Regresión con splines

La regresión polinómica vista anteriormente ajusta la funcion f a lo largo de todo el rango de X mediante un polinomio de grado considerable. Sin embargo, existe una alternativa que ofrece la posibilidad de ajustar f con polinomios distintos de menor grado en distintas regiones del rango de X.

Para demostrarlo, utilizaremos como ejemplo un polinomio cúbico definido a trozos para realizar el ajuste de un modelo de regresión cúbico, cuya forma, aplicando la expresión de la sección anterior, es la mostrada a continuación

$$y_i = \beta_0 + \beta_1 x_i + \beta_2 x_i^2 + \beta_3 x_i^3 + \epsilon_i$$

donde los coeficientes cambian en función de la región del rango de X, y esos puntos donde los coeficientes toman valores distintos se denominan nodos. Ese tipo de polinomios se conoce como polinomio cúbico estándar.

Ahora mostraremos un polinomio cúbico definido a trozos con un nodo en el punto x^*, donde se ajustan dos funciones polinómicas distintas y que se representaría como

$$y_i = \begin{cases} \beta_{01} + \beta_{11} x_i + \beta_{21} x_i^2 + \beta_{31} x_i^3 + \epsilon_i & si \;\; x_i < x^*, \\ \beta_{02} + \beta_{12} x_i + \beta_{22} x_i^2 + \beta_{32} x_i^3 + \epsilon_i & si \;\; x_i \geq x^*. \end{cases}$$

Esta técnica, denominada **regresión con splines**, representa un ajuste similar al del polinomio por partes, donde las regiones están separadas por puntos de ruptura o nodos y los coeficientes de dichos polinomios se podrán estimar mediante mínimos cuadrados. Como observamos en la Figura 2.1, el rango de X se divide en regiones diferentes y en cada una se ajusta una función polinómica distinta sobre las observaciones correspondientes. Aparte, la unión de los polinomios en estos puntos tiene que ser suave, por lo que se fuerza a que las derivadas sean continuas y, así, los polinomios serán también continuos en dichos nodos.

Generalmente, el ajuste de f será mas flexible cuantos más nodos se usen, es decir, se producirá un ajuste demasiado suave si el rango de X es dividido en muchas regiones. Comentar que, si hay k nodos a lo largo del rango de X, se ajustan $k+1$ polinomios. Por tanto, un spline de grado d se define a través de un polinomio de ese mismo grado definido a trozos, cuyas primeras $d-1$ derivadas son continuas en cada uno de los nodos. En la práctica los más utilizados son lo de tercer grado, denominados **Splines Cúbicos**.

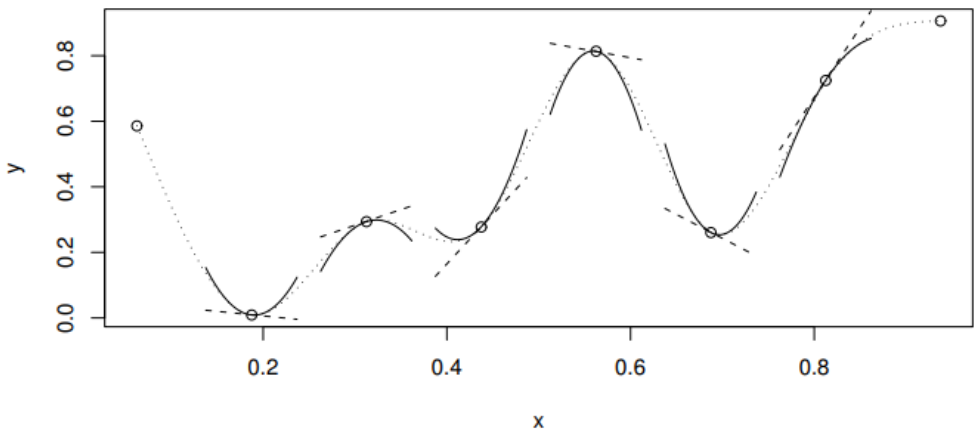


Figura 2.1: Este gráfico representa un spline cúbico que, como vemos, es una curva construida mediante diferentes polinomios cúbicos unidos de tal manera que la curva sea continua hasta la segunda derivada. Cada región tiene distintos coeficientes en función del polinomio, pero en los nodos se igualarán los valores. Los nodos del splines son los puntos de unión (∘). (Wood, 2006).

2.3.3. Splines de suavizado

En términos estadísticos, y_i es una variable que normalmente se mide con ruido, y generalmente es mas útil suavizar x_i, y_i, en vez de interpolarlas. En este apartado comentaremos un enfoque diferente al explicado anteriormente pero con el mismo objetivo de generar splines.

Cuando se nos plantea realizar un ajuste de una curva a unas determinadas observaciones, se desea hallar una función apropiada para minimizar

$$ECM = \sum_{i=1}^{n} (y_i - f(x_i))^2.$$

Podría surgir un problema de sobre-ajuste si no se imponen algunas restricciones sobre $f(x_i)$ ya que si $y_i = f(x_i) \ \ \forall i = 1, ..., n$, la expresión formulada anteriormente sería cero y se interpolaría todo el conjunto de entrenamiento. El ajuste obtenido en ese caso sería demasiado cambiante y lo que se pretende es encontrar una función f que minimice la fórmula anterior y que realice simultáneamente un ajuste suave adecuado. Con este método de splines evitaremos completamente el problema de la selección de nodos usando un conjunto máximo de nodos. Este nuevo ajuste es bastante complejo pero se controla mediante la regularización.

El Spline cúbico suavizado es el más utilizado y soluciona el problema que existía a la hora de buscar una función que minimice la suma residual de cuadrados penalizados entre todas las funciones $f(x)$ con dos derivadas continuas, tal que

$$RSS(f, \lambda) = \sum_{i=1}^{N} \{y_i - f(x_i)\}^2 + \lambda \int \{f''(t)\}^2 dt,$$

donde λ en un parámetro de suavizado fijo que establece una solución intermedia entre el primer término, que mide la proximidad del ajuste realizado a los datos, y el segundo término, que penaliza la variabilidad de la función, ya que $f''(t)$ mide cuánto cambia la pendiente de una función en t.

Existen dos tipos de funciones según el parámetro de suavizado λ que habría que comentar. El primero, cuando $\lambda \to 0$, f puede ser cualquier función que intercale los datos porque f será muy cambiante y el término de penalización deja de ser importante; y el segundo, cuando $\lambda \to \infty$, se obtiene una función f muy suavizada debido a que el término de penalización fuerza a que no haya una segunda derivada.

Para concluir, comentar que lo que se pretende es realizar un ajuste adecuado que se acerque al conjunto de entrenamiento y que sea suave, lo que se consigue con un valor intermedio de λ que alcance un equilibrio entre el ajuste de las observaciones y la suavidad de éste.

2.4. Ajuste Modelos Aditivos. Algoritmo Backfitting

Para ajustar los Modelos Aditivos y sus generalizaciones utilizaremos un algoritmo que describiremos a continuación. La base principal es el diagrama de dispersión de suavizado ya que ajusta, de una manera muy flexible, efectos no lineales. En este caso utilizaremos como nuestro diagrama de dispersión al Spline Cúbico de suavizado, uno de los más utilizados, como ya habíamos comentado.

En la sección 2.2 definimos el modelo aditivo, pero también podemos obtener una estimación global como la suma de las p funciones estimadas univariantes. Por consiguiente, el modelo aditivo quedaría como

$$y_i = \alpha + \sum_{j=1}^{p} f_j(x_{ij}) + \epsilon_i, \quad i = 1, ..., n$$

donde los errores ϵ_i, con $E(\epsilon_i) = 0$ y $\text{Var}(\epsilon_i) = \sigma^2$, son independientes de x_j.

Teniendo en cuenta la forma del Modelo Aditivo, dadas las observaciones x_i e y_i, un criterio como la suma de cuadrados penalizados de la sección anterior se podría utilizar para este problema. En este caso, obtendríamos

$$PRSS(\alpha, f_1, f_2, ..., f_p) = \sum_{i=1}^{N} \left(y_i - \alpha - \sum_{j=1}^{p} f_j(x_{ij}) \right)^2 + \sum_{j=1}^{p} \lambda_j \int f_j''(t_j)^2 dt_j,$$

donde λ_j son los parámetros de penalización.

El modelo que minimiza esta función es uno de spline cúbico aditivo, donde cada una de las funciones f_j es un spline cúbico en la componente X_j, con nodos en cada uno de los valores de x_{ij}. Las $f_j's$ son funciones sin forma restringida que solo tienen un supuesto de suavidad entendida en términos de derivabilidad de la función.

Además, para que las funciones sean únicas, se impone la condición de $\sum_{i=1}^{n} f_j(x_{ji}) = 0, \forall j$.

Como la constante α no se puede identificar ya que podemos eliminar o añadir cualquier constante a las funciones $f_j's$, se puede ajustar α de acuerdo a ello.

Es más, existe un procedimiento iterativo para encontrar la solución, el cual es el principal método de estimación para estos modelos. Ese método es el denominado **algoritmo backfitting**, de Buja, Hastie y Tibshirani (1989).

Este algoritmo se desarrolla de la siguiente manera: primero, para que α no varíe, se propone para α un valor tal que $\alpha = \frac{1}{N} \sum_1^N y_i$, $\hat{f}_j \equiv 0, \forall i, j$. En segundo lugar, para obtener una nueva estimación de $\hat{f}_j$, utilizamos $S_j \left[\{ y_i - \hat{\alpha} - \sum_{k \neq j} \hat{f}_j(x_{ij}) \}_1^N \right]$ siendo S_j un spline cúbico de suavizado. Este ajuste se hace para cada predictor, uno por uno, hasta que la funcion estimada, $\hat{f}_j$, se estabilice y se encuentre dentro de un umbral preestablecido.

Este algoritmo nos permite estimar las funciones mediante el suavizado de los residuos parciales, utilizando cualquier suavizador univariante de componentes. Este algoritmo es análogo al método *Gauss-Seidel* para el problema de mínimos cuadrados en los problemas de regresión lineal.

Capítulo 3

Redes Neuronales

3.1. Introducción

El campo de las Redes Neuronales se ha consolidado durante los últimos años dentro de las ciencias de la computación. Dentro de este campo, las que han causado mayor impacto han sido las Redes Neuronales Artificiales, RNA, debido a su aplicación práctica. Estas redes incorporan un conjunto de herramientas estadísticas centradas en la clasificación de patrones y la estimación de variables continuas que han sido de gran utilidad para resolver problemas relacionados con el mundo real. Una RNA es un modelo computacional y matemático cuya metodología y funcionamiento están inspirados en los sistemas biológicos. El cerebro del ser humano se estructura en unidades elementales, denominadas neuronas, que se conectan entre ellas mediante impulsos eléctricos y son las que nos otorgan la capacidad de procesar datos. Las neuronas biológicas emiten impulsos en función de distintos estímulos de entrada, los cuales pueden provenir de otras neuronas o de estímulos físicos que proceden de los órganos sensoriales.

Siguiendo ese procedimiento, surgieron las redes neuronales artificiales. Estas están formadas por un grupo interconectado de neuronas artificiales y procesos de información, cuyas conexiones disponen de un valor numérico llamado peso.

Este método, que se desarrolló tanto en el campo estadístico como en el de la inteligencia artificial, ha evolucionado abarcando muchas clases de modelos y métodos de aprendizaje. La idea principal es extraer combinaciones lineales de los inputs como características y luego modelar el objetivo como una función no lineal de esas características.

Una RNA está estructurada en varias capas. La primera capa o capa de entrada y la última o capa de salida siempre están presentes; sin embargo, las capas intermedias a veces no son necesarias, por eso también se suelen llamar capas ocultas. Las neuronas de la capa de entrada son las encargadas de recibir la información del exterior, las neuronas ocultas reciben información de otras neuronas y sus señales de entrada y salida permanecen dentro de la red y, por último, las neuronas de salida reciben la información procesada y la devuelven al exterior. El objetivo de este tipo de redes es descubrir alguna asociación entre la información aportada a las neuronas a través de la primera capa y el patrón de salida, proveniente del procesamiento de toda la información propagada de capa en capa.

En la siguiente sección explicaremos brevemente los distintos tipos de redes neuronales artificiales con conexiones hacia delante, es decir, redes donde el flujo desde las unidades de entrada hasta las de salida es estrictamente hacia delante, no hay conexiones hacia atrás independientemente de las capas que tenga. Estas redes utilizan algoritmos de entrenamiento supervisado como son el Perceptrón simple, el multicapa y la red Adaline.

3.2. El Perceptrón.

Una Red Neuronal es como un modelo de dos etapas de regresión o clasificación. Para regresión, caso en el que nos encontramos nosotros, habitualmente hay solo un output, una única variable de salida. Aunque también puede interpretar un modelo con varias variables respuestas.

Por tanto, cabría destacar un caso particular de las redes neuronales, los modelos lineales. Es el modelo más sencillo de redes neuronales, ya que consta de una sola capa de neuronas con una única salida. El diagrama del modelo lineal visto como una red neuronal sería el que se muestra en la *Figura 3.1*, que como se observa, es una red neuronal pero sin capas ocultas.

Este tipo de modelos también es conocido como Perceptrón simple, el cual fue introducido en 1958 por Frank Rosenblantt. Una de las características de este modelo que más interés despertó fue su capacidad de aprender a reconocer patrones.

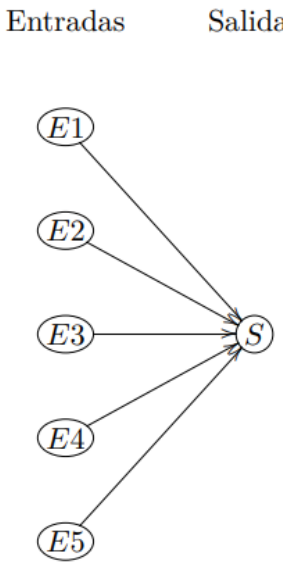


Figura 3.1: Diagrama de un modelo lineal.

De forma matemática, consideramos y_i la salida de la neurona i y se obtiene combinando la función de activación a la función de entrada, es decir,

$$y_i = f(in_j) + \epsilon_i = f(\sum_{j=1}^{n+1} w_{ij}x_j) + \epsilon_i$$

donde w_{ij} son los pesos de la red y los parámetros del modelo que determinan el efecto que la señal de las neuronas de entrada j tiene en la unidad de salida i, $f(x)$ es la función de activación y determina la salida de la red y la x_j el patrón de entrada.

También habría que comentar que $\sum_{j=1}^{n+1} w_{ij}x_j$ es la suma de entradas por sus pesos, definida como la función de entrada, in_j.

Despues de haber definido todas las variables a considerar, procederemos a la explicación de un proceso adaptativo para determinar los pesos. Se inicia con unos valores aleatorios y se van modificando iterativamente cuando la salida y no coincide con la salida deseada z.

Por tanto, para la modificación de los pesos seguiremos la regla conocida como regla de aprendizaje del Perceptrón simple y viene dada por la siguiente expresión:

$$w_j(k+1) = w_j(k) + \eta(k)[z(k) - y(k)]x_j(k)$$

donde $\eta(k)$ es un parámetro positivo conocido como tasa de aprendizaje, es el parámetro que controla el proceso de aprendizaje. Este funciona como una constante de proporcionalidad ya que cuanto más aumente su valor, el peso más se modificará, y viceversa.

La expresión anterior da a entender que la variación del peso w_j es proporcional al producto del error $z(k) - y(k)$ por la componente $j - ésima$ del patrón de entrada introducido en la iteración k, es decir, $x_j(k)$. Esta regla de aprendizaje se podría expresar de distinta forma si se utiliza una función de transferencia conocida como función signo, de forma que

$$\mathrm{w}_j(k+1) = \begin{cases} w_j(k) + 2\eta(k)x_j(k) & si \;\; y(k) = -1 \;\; y \;\; z(k) = 1, \\ w_j(k) & si \;\; y(k) = z(k), \\ w_j(k) - 2\eta(k)x_j(k) & si \;\; y(k) = 1 \;\; y \;\; z(k) = -1. \end{cases}$$

Esta nueva regla lo que haría sería modificar los pesos cuando fallase. Por ejemplo, considerando binarias las posibles entradas y salidas, en este caso $(-1, 1)$ cuando el patrón pertenece a la primera clase , $z(k) = 1$, y se le asigna a otra, $y(k) = -1$, refuerza el valor del peso añadiendo una cantidad proporcional al valor de la entrada. En cambio, si es al contrario, lo debilita.

3.2.1. Una generalización del Perceptron: El Adeline

El Adeline es otro modelo de redes neuronales, similar al Perceptrón pero ya no utiliza la función signo, sino que tiene como función de activación la función identidad ($f(x) = x$). Ahora la salida de la red es continua en lugar de binaria, debido a que es una función lineal de las entradas ponderadas con los pesos.

En esta situación seguiremos la regla Delta como regla de aprendizaje. Este nueva regla utiliza una función de coste o error que consiste en determinar los pesos con el objetivo de minimizar dicha función de error cuadrático. El error total se define como:

$$E = \frac{1}{2}\sum_{k=1}^{p}(z(k) - y(k))^2 = \frac{1}{2}\sum_{k=1}^{p}(z(k) - f(\sum_{j=1}^{n+1} w_j(k)x_j(k)))^2$$

Esta regla funciona realizando un ajuste en cada peso proporcial a la derivada del error respecto del peso, tal que

$$w_j(k+1) = w_j(k) - \eta\frac{\partial E}{\partial w_j(k)}$$

3.3. El Perceptrón Multicapa. Algoritmo Backpropagation

La función de salida para el Perceptrón multicapa con una única capa oculta, formada por L neuronas, se define con la siguiente expresión

$$y_i = f(\sum_{j=1}^{L} w_{ij}s_j) + \epsilon_i = f_1(\sum_{j=1}^{L} w_{ij}f_2(\sum_{r=1}^{N} t_{jr}x_r)) + \epsilon_i$$

donde t_{jr} es el peso que conecta la neurona oculta j con la neurona de entrada x_r, f_1 corresponde con la función de activación de las unidades de salida y f_2 de las unidades de la capa oculta.

Con estas circunstancias, la regla de aprendizaje supervisado que aplicaremos para calcular los pesos es la regla Delta, como en el caso del Adeline, pero ahora se pretende minimizar una función de error con alguna modificación, tal que

$$E = \frac{1}{2}\sum_{k=1}^{p}\sum_{i=1}^{M}(z_i(k) - y_i(k))^2.$$

Sabiendo esto, la forma de operar del algoritmo de retropropagación del error consiste en propagar por las capas existentes el patrón de entrada que se aplica a la primera capa hasta que llegue a la capa de salida y es en ese punto donde se compara con la salida deseada. Después se calcula el error para cad neurona de salida y se transmite hacia atrás por las capas intermedias. En este proceso es cuando se modifican los pesos según el error obtenido y los valores de las salidas ponderadas por sus pesos hasta llegar a la capa de entrada.

Capítulo 4

Estudio práctico

Para la puesta en marcha del estudio, se proponen modelos provenientes de la teoría explicada brevemente en los capítulos previos. Para este estudio, se utilizó una base de datos que nos proporcionó el *Centro de Supercomputación de Galicia (CESGA)* con 430800 observaciones, de la que no sabemos el significado de cada una de las variables por su complejidad. Este contratiempo no nos supone un gran inconveniente puesto que nuestro objetivo no se basa en el grado de aportación de cada variable explicativa a la variable respuesta, sino que nos centramos en la detección de datos atípicos, anómalos o también conocidos como *outliers*. Sin embargo, la descripción de cada una de ellas facilitaría la comprensión del estudio realizado.

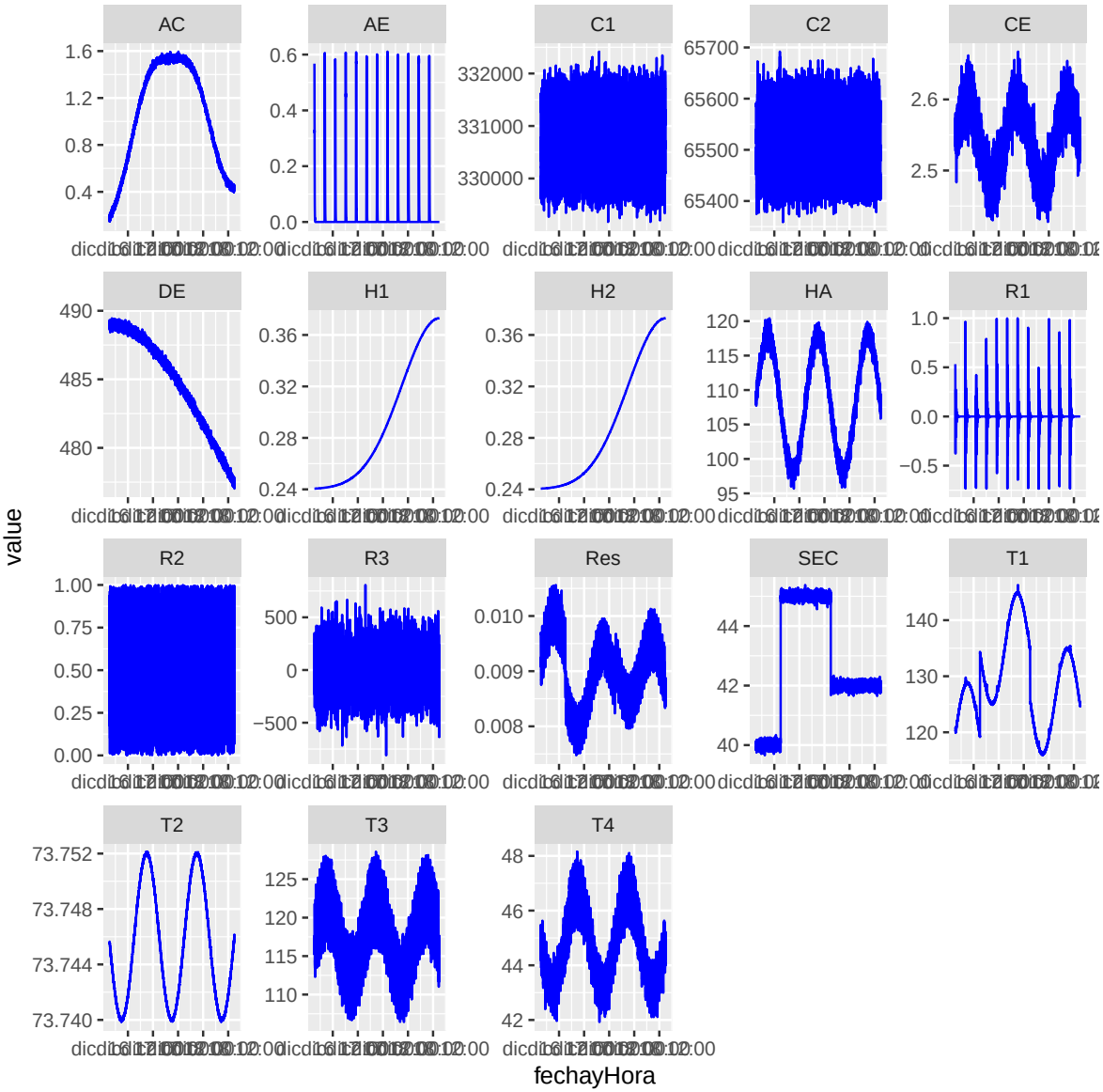
En un primer momento, el planteamiento estuvo enfocado en las series de tiempo, donde todas las series dependen, mucho o poco, de su pasado, donde podemos estimar un patrón aleatorio y estacionario. Cuando hablamos de predicción dinámica, series cointegradas o series multivariantes es porque aparte de su dependencia del pasado pensamos que puede haber relación entre las variables a lo largo del tiempo. Para eso, es interesante usar modelos de series de tiempo. Pero el modelo que nos plantearon desde el CESGA no cumplía ciertos requisitos de las series de tiempo ya que suponían que los datos eran independientes entre instantes de tiempo con errores aleatorios y toda la desviación sistemática se producía en la respuesta. Esto sería como generar variables aleatorias independientes, con algo ruido, y generar una respuesta que sea la combinación que quieras de esas variables más un desvío sistemático. Lo que realmente se podría interpretar como una estimación de un modelo de regresión donde, efectivamente, el paso del tiempo influirá a lo largo del proceso.

En primer lugar, realizamos un arreglo de los datos para su óptima utilización y, una vez listos iniciaremos el estudio. Procedemos, antes que nada, a una breve evalución visual mediante unos gráficos que relacionan cada variable con el tiempo. En nuestro caso, el tiempo viene determinado por la variable "fechayHora", por lo que habría que mencionar que esta variable indica la fecha y la hora a la que se ha tomado cada observación, con una diferencia entre cada una de un segundo. Se divide la base de datos en dos submuestras, donde la primera mitad corresponderá con los datos de entrenamiento que utilizaremos para la estimación de los modelos y, la segunda, con los datos de test para una evaluación y comprobación al realizar futuras predicciones. Como lo más justo sería considerar un conjunto sin datos atípicos para todos los modelos que se prueben, utlizaremos el de entrenamiento.

```
datos <- read.table("Sara.csv", header = TRUE, sep = ",", dec = ".", na.strings = "NA")

# Arreglamos variable indicadora de la fecha y la hora.
datos$X <- lubridate::ymd_hms(datos$X)
names(datos)[1] <- "fechayHora"

train<- datos[1:215400,]
```

```
# Sacamos una muestra

N<-nrow(train)
set.seed(123)
n<-5000
s<-sample(N,n)
muestra<-train[s,]

# Ordenamos la muestra

ii=order(muestra[,1])
muestra<- muestra[ii,]
```

```
# Mostramos todas las variables en funcion de la variable "fechayHora"

ldf <- pivot_longer(muestra,-fechayHora)
ggplot(ldf,aes(fechayHora,value)) + geom_path(color = "blue") + facet_wrap(~name,scales = "free")
```



A través de este gráfico se observa que, efectivamente, el paso del tiempo influye en cada una de las variables del modelo de una manera diferente. Siendo "Res" la variable respuesta que queremos analizar.

4.1. Modelo lineal

Comenzamos con el análisis del Modelo Lineal, donde se crea un modelo con todas las variables explicativas, excepto "fechayHora". Como criterio para la selección de variables, ya que algunas han resultado no ser significativas, aplicamos el criterio AIC con el método Backward mediante la función step de R y obtenemos el siguiente resumen del modelo:

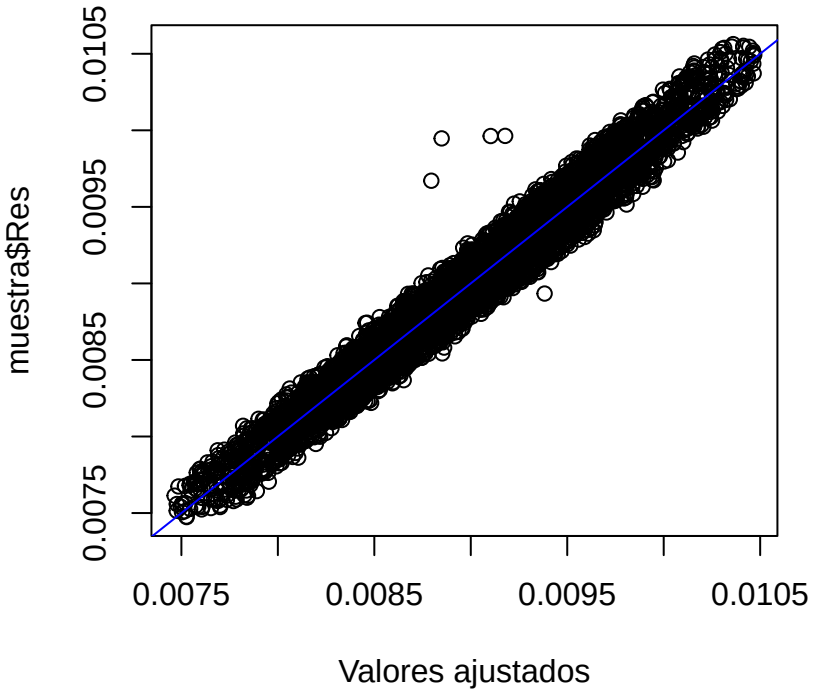
```
# Modelo Lineal

modl <- lm(data = muestra[,-1],Res~.)
```

```
summary(modl)

# Metodo de seleccion de variables

modl=step(modl)
```

```
summary(modl)

##
## Call:
## lm(formula = Res ~ H1 + T1 + T2 + T3 + T4 + C2 + AE + AC + DE,
##     data = muestra)
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -4.489e-04 -1.027e-04 -2.300e-07  1.002e-04  1.098e-03
##
## Coefficients:
##               Estimate Std. Error  t value Pr(>|t|)
## (Intercept)  1.480e+01  2.252e-01   65.733  < 2e-16 ***
## H1          -1.822e-03  4.379e-04   -4.160 3.24e-05 ***
## T1          -6.181e-05  5.201e-07 -118.839  < 2e-16 ***
## T2          -2.005e-01  3.052e-03  -65.703  < 2e-16 ***
## T3           6.469e-05  1.226e-06   52.779  < 2e-16 ***
## T4           1.204e-04  6.533e-06   18.425  < 2e-16 ***
## C2          -1.095e-07  2.497e-08   -4.387 1.17e-05 ***
## AE           3.736e-04  2.109e-05   17.717  < 2e-16 ***
## AC           4.143e-04  7.250e-06   57.141  < 2e-16 ***
## DE          -1.235e-05  5.663e-06   -2.182   0.0292 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.0001279 on 4990 degrees of freedom
## Multiple R-squared:  0.961,Adjusted R-squared:  0.9609
## F-statistic: 1.366e+04 on 9 and 4990 DF,  p-value: < 2.2e-16
```

El R^2 ajustado tiene un valor de 0.9609, por lo que se podría considerar que el modelo de regresión ajusta adecuadamente los datos ya que, como vimos, el coeficiente de determinación es una buena medida de ajuste de la regresión, a pesar de que, únicamente con este dato, no podamos confirmar si el modelo es el correcto.

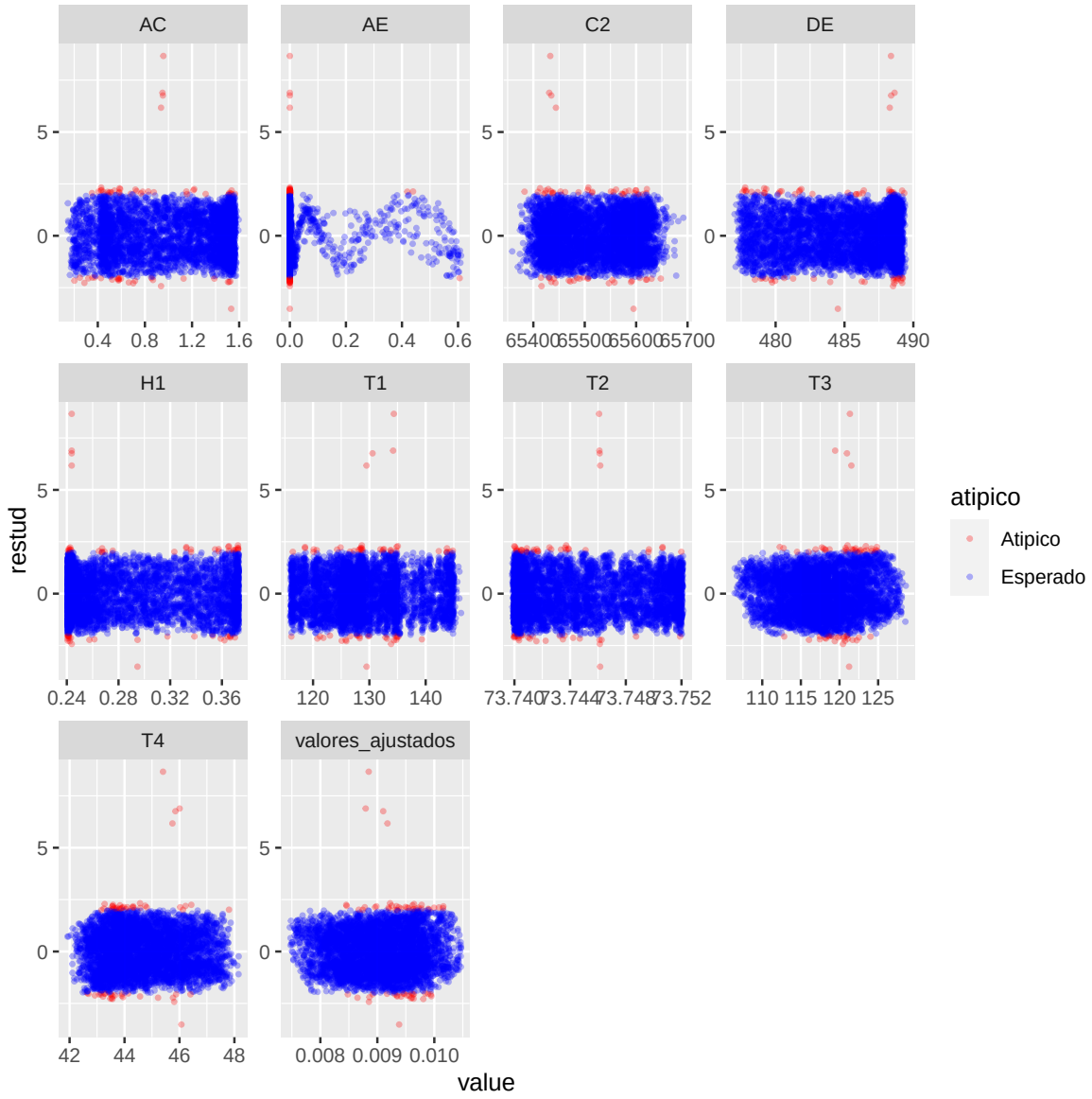
```
plot(muestra$Res~fitted(modl),xlab="Valores ajustados")
abline(lm(muestra$Res~fitted(modl)), col="blue")
```



En el gráfico anterior se muestra la relación entre la variable respuesta "Res" frente a los valores ajustados una vez estimado el modelo lineal. La linea azul representa la recta de ese mismo ajuste. Este diagrama de dispersión parece una buena predicción de nuestro modelo a pesar de que ciertos puntos no siguen la trayectoria del resto de la nube de puntos. En esta gráfica se expone el resultado real contra el resultado que nosotros hemos predicho, por lo que se podría pensar que nuestro modelo pierde eficacia a la hora de predecir algunos valores que toma nuestra variable respuesta entre 0,0088 y 0,0095.

4.1.1. Detección de atípicos

A continuación procedemos a la detección de atípicos. En el Capítulo 1 explicamos una posible forma para detectarlos a través de los residuos estudentizados.



Se puede observar que la mayor parte de residuos de las variables se encuentra en el intervalo $[-2, 2]$ y muy pocos son los que se salen de ese patrón. Aquellos que no se encuentran en ese rango son los que consideramos atípicos, concretamente 63.

Este es un método bastante encillo para la detección de estas observaciones concretas. A mayores, explicaremos el procedimiento de otro criterio un poco mas enrevesado que utilizaremos con el modelo lineal y con los siguientes.

El procedimiento utilizado consiste en estimar el valor medio de la respuesta al introducir un conjunto de nuevos individuos, de los que se conocen los valores de las variables explicativas. Pero el foco no está en esa predicción en concreto, sino que nuestro interés es predecir cuál es el intervalo que puede abarcar el valor de la variable respuesta para unos valores de las varibles explicativas. En regresión, ese intervalo recibe el nombre de intervalo de predicción, el cual va asociado a la dispersión de las observaciones.

Los intervalos de confianza representan el espacio donde podemos encontrar un parámetro estadístico preciso, como puede ser la media, desviación estándar,etc., con una determinada probabilidad. En cambio, nosotros lo que buscamos es representar el espacio donde se distribuyen los datos con una

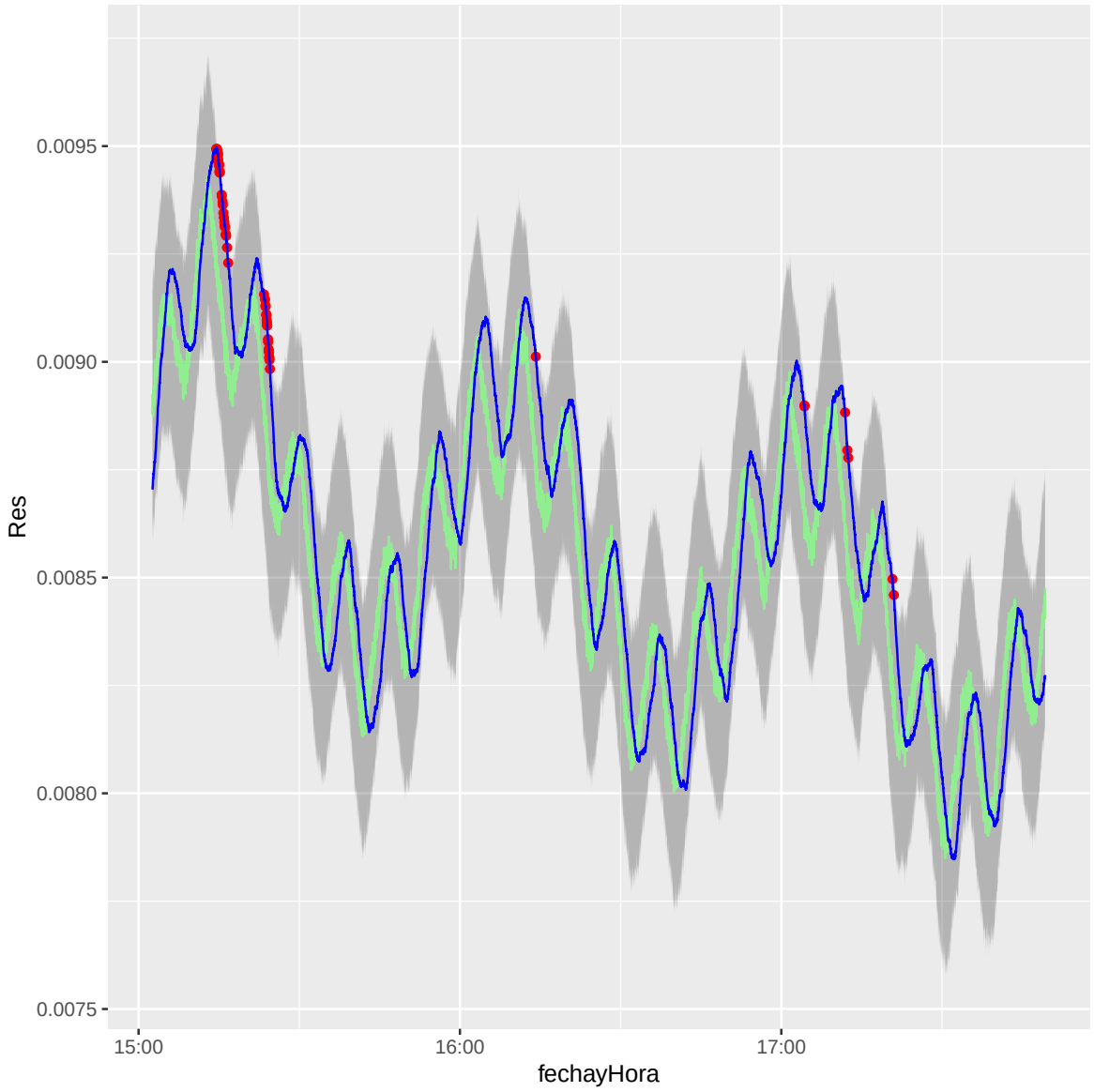
probabilidad asociada. Por tanto, ese es nuestro objetivo, la distribución de los datos en su espacio, es decir, poder representar la probabilidad de encontrar una observación dentro de un intervalo y, así, podremos detectar la existencia de datos atípicos. Es este caso, consideraremos como atípica una observación cuando se encuentre fuera del intervalo antes mencionado.

Este método consistirá en que, una vez estimado el modelo en cuestión con la primera mitad de los datos, los cuales consideramos que están bien, empecemos a analizar si las siguientes observaciones son o no atípicas. Esta detección la llevaremos a cabo a traves de las predicciones que hagamos del modelo a lo largo del tiempo, la cual es la manera razonable de evaluar los nuevos datos. Dibujaremos mediante un gráfico esas predicciones con su correspondiente intervalo. Además, a ese gráfico le añadiremos un conjunto de datos que serán la continuación de aquellos considerados de entrenamiento. Una vez hayamos analizado ese trozo con ese cantidad de observaciones, marcaremos los outliers que detectemos y los extraeremos del conjunto de datos analizado. Así, incorporaremos ese conjunto sin outliers como nueva estimacion para seguir iterando y continuaremos esa técnica hasta que lleguemos al final.

```
# Prediccion
# Sacamos una submuestra con los 10000 primeros datos de la base de datos de test
ini10k<- datos[215401:225400,]
pre <- as.data.frame(predict(modl,ini10k,interval = "prediction", level=0.975 ))
names(pre) <- c("fit_modl","low_modl","upp_modl")

# Datos con las prediciones de modl
ini10k <- cbind(ini10k,pre)
ini10k$error <- ini10k$Res - ini10k$fit_modl
head(ini10k)
```

```
# Intervalo de prediccion variable

lo <- ini10k$low_modl
up <- ini10k$upp_modl

# Deteccion de Outliers

ini10k$aviso <- ifelse(ini10k$Res > up | ini10k$Res < lo,"Outlier","Normal")
```

```
grafdf <- ini10k[,c("fechayHora","Res","fit_modl","low_modl","upp_modl")]

ggplot(ini10k,aes(fechayHora,Res)) +
  geom_ribbon(inherit.aes = F,aes(x = fechayHora,ymin = lo,ymax = up),alpha = 0.3)+
  geom_point(data = subset(ini10k, aviso == "Outlier"),color = "red")+
  geom_path(aes(x = fechayHora, y = fit_modl), color = "lightgreen") +
  geom_path(color = "blue")
```




```
# Sacamos los outliers de la muestra con las nuevas observaciones

nor=subset(ini10k, aviso=="Normal")
out<- subset(ini10k, aviso=="Outlier" )
nrow(out)

## [1] 59

out_lineal<- out$fechayHora
```

El gráfico se obtiene poniendo en práctica todo lo comentado. El color verde se designa a la prediccion realizada con el modelo lineal junto con su intervalo, el cual aparece sombreado en gris. La linea azul indica el valor real que toma la variable respuesta en cada segundo evaluado y los puntos rojos son aquellas observaciones que sobresalen del intervalo, ya que hemos designado como valor atípico

aquellos que sobrepasen el rango del intervalo. Una vez ya detectados, los extraemos de la submuestra e incorporamos ese conjunto sin outliers a la nueva estimación para realizar un análisis equivalente a los siguientes nuevos valores.

Aplicando este criterio observamos que se han detectado más outliers que con el criterio anteriormente empleado sobre los residuos estudentizados. Este hecho puede ser debido a la existencia de relaciones entre las variables retardadas, ya que la predicción (linea verde) está algo más desplazada hacia la izquierda de lo que debería. Usando el cuantil 95 % obtenemos 332 datos atípicos, que son muchos más que los obtenidos con el método de los residuos, 63. Esto se debe a la falta de variables retardas, para poder identificar las observaciones anómalas mediante este modelo y poder compararlas con el siguiente, subimos el cuantil a 97,5 %, y obtenemos 59 outliers que se pueden apreciar en el gráfico que hemos analizado anteriormente y las compararemos más adelante con las del próximo modelo.

4.2. Modelo Aditivo

Para la detección de outliers del Modelo Aditivo aplicaremos una metodologia análoga a la de los modelos lineales.

Inicialmente, para la estimación del modelo aditivo se aplica el ajuste mediante splines, cuya función en R es la que se representa mediante $s()$.

```
# Modelo aditivo


aditivo <- gam(Res ~ s(H1) + s(H2) + s(T1) + s(T2) + s(T3) + s(T4) + s(SEC)
+ s(C1) + s(C2) + s(AE) + s(AC) + s(HA) + s(DE) + s(CE) + s(R1) + s(R2)
+ s(R3), data = muestra)

adit <- gam(Res ~ s(T1) + s(T2) + s(T3) + s(T4) + s(C1) + s(AE) + s(AC)
            + s(DE) + s(R1), data = muestra)
summary(adit)

##
## Family: gaussian
## Link function: identity
##
## Formula:
## Res ~ s(T1) + s(T2) + s(T3) + s(T4) + s(C1) + s(AE) + s(AC) +
##     s(DE) + s(R1)
##
## Parametric coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 9.018e-03  1.725e-06    5228   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
##         edf Ref.df       F p-value
## s(T1) 2.065  2.566 1512.64  <2e-16 ***
## s(T2) 4.085  5.036  472.74  <2e-16 ***
## s(T3) 6.256  7.437  454.08  <2e-16 ***
## s(T4) 1.000  1.000  398.28  <2e-16 ***
```

```
## s(C1) 1.000  1.001    3.66  0.0558 .
## s(AE) 8.429  8.902   35.26  <2e-16 ***
## s(AC) 5.399  6.625   77.19  <2e-16 ***
## s(DE) 7.134  8.131   59.93  <2e-16 ***
## s(R1) 8.531  8.932   10.96  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) =  0.964   Deviance explained = 96.5%
## GCV = 1.5009e-08  Scale est. = 1.4874e-08  n = 5000
```

Empezamos ajustando un modelo con todas las variables explicativas, algunas de las cuales resultan no significativas. A pesar de que no haya un método automático de selección de variables como en el caso de modelos lineales, realizamos esta extracción manual, quedándonos con aquellas que resultan significativas.

Como ya hemos comentado, el objetivo de nuestro trabajo no es la diagnosis de los modelos como tal, sino la detección de outliers en los distintos modelos. Por ello, al ver el resumen de nuestro modelo ponemos el foco en el valor de la estimación de la varianza residual, nombrado como scale estimated.

En estos modelos la complejidad de la predicción es un poco mayor ya que en los modelos lineales la funcion predict() nos proporcionaba directamente el valor de la predicción y la banda superior e inferior del intervalo. En cambio, en este caso, se muestra el valor de la predicción media, no el de una nueva observación. Debido a esto, obtenemos el valor de la desviación estándar de predicción de una nueva observación de la siguiente forma:

```
ini10<-datos[215401:225400,]
var.residual<-1.4874e-08
fits = predict(adit, newdata=ini10, type='response', se=T)
desv.est<-sqrt((fits$se.fit)^2+ var.residual)
predicts = data.frame(ini10, fits) %>%
  mutate(lower = fit - 2.24*desv.est,
         upper = fit + 2.24*desv.est)
```

```
# Datos con las prediciones del modelo aditivo

predicts$error <- predicts$Res - predicts$fit
head(predicts)

# Intervalo de prediccIOn variable

lo <- predicts$lower
up <- predicts$upper

# Deteccion de Outliers

predicts$aviso <- ifelse(predicts$Res > up | predicts$Res < lo,"Outlier","Normal")

grafdf <- predicts[,c("fechayHora","Res","fit","lower","upper")]
```

```
ggplot(predicts,aes(fechayHora,Res)) +
  geom_ribbon(inherit.aes = F,aes(x = fechayHora,ymin = lo,ymax = up),alpha = 0.3)+
  geom_point(data = subset(predicts, aviso == "Outlier"),color = "red")+
  geom_path(aes(x = fechayHora, y = fit), color = "lightgreen") +
  geom_path(color = "blue")
```

De la misma manera que en el caso lineal, en color rojo aparecen los outliers encontrados aplicando nuestra metodología. Se puede apreciar que los valores no coinciden con las observaciones atípicas del modelo lineal a pesar de partir de la misma base de datos. Además, habría que destacar la diferencia que hay entre los valores reales y el ajuste, que también es debido a la posible influencia de las variables retardadas. En este caso obtenemos muchas menos observaciones atípicas, unicamente 3, con el cuantil 97,5 %, sin embargo, con el cuantil 95 % obteníamos 119.

4.3. Modelo Redes Neuronales

Empezamos estimando un modelo de Redes Neuronales. Como no existe una regla definida que nos indique el número de capas ocultas que debe tener una Red ni el número de neuronas, ajustamos una Red Neuronal con una capa oculta.

Para un mejor funcionamiento, usamos las variables seleccionadas en el modelo lineal como punto de inicio para este método. Realizamos el procedimiento con la función neuralnet(), la cual estima el modelo usando el algoritmo backpropagation comentado previamente en la teoria correspondiente a ese tema. Los argumentos a utilizar son "hidden" que acepta un vector con el número de neuronas para cada capa oculta y el argumento "linear.output" se usa para especificar mediante la indicación TRUE si queremos hacer regresión o FALSE para clasificacación.

```
# Ajustamos modelo con las variables definitivas del modelo lineal.


f <- Res~ H1+T1+T2+T3+T4+C2+AE+AC+DE
  set.seed(1480)
  nn1 <- neuralnet(f,data=muestra,
                   hidden=1,
                   linear.output=T)
```

Podemos ver la estructura y las componentes propias de las redes neuronales en el gráfico. El flujo que obtienen las neuronas de la capa de entrada proviene de la información aportada por nuestas variables explicativas. Los pesos son las cantidades que conectan las neuronas de cada capa, obtenido mediante el ajuste que consiste en modificar esa cantidad hasta que el valor obtenido en la variable respuesta coincida con el deseado.

```
yPredInt <- nnetPredInt(nn1, x, y, newData)
head(yPredInt)

##   yPredValue lowerBound upperBound
## 1  0.5022544   0.500985  0.5035238
## 2  0.5022544   0.500985  0.5035238
## 3  0.5022544   0.500985  0.5035238
## 4  0.5022544   0.500985  0.5035238
## 5  0.5022544   0.500985  0.5035238
## 6  0.5022544   0.500985  0.5035238
```

La red neuronal que estimamos está prediciendo la media y por eso los valores de predicción a simple vista no tienen diferencia alguna.

Como las variables que se usan no discriminan bien y la red tiene pocos nodos, la predicción que nos da es la media. Por ello, aumentamos el número de nodos a 50 y, al predecir, obtenemos pequeñas diferencias entre los valores de predicción. Ahora utilizamos la funcion nnet(), la cual ajusta la red neuronal con una única capa oculta, y posiblemente con conexiones de salto de capa, ya que con la anterior función seguían sin ser notorias las posibles diferencias.

```
f <- Res~ H1+T1+T2+T3+T4+C2+AE+AC+DE
set.seed(1480)
nnG2<-nnet(f, data=muestra,
           size=50, linout=TRUE, decay=1e-4,
           maxit=5000)
```

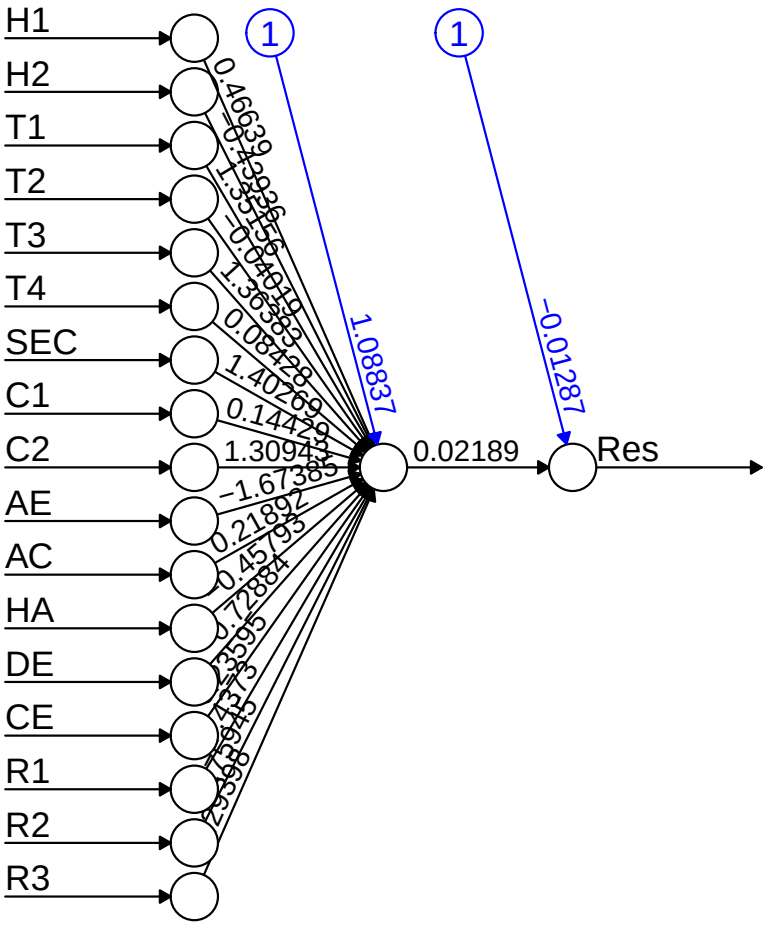


Figura 4.1: Red Neuronal con una capa oculta de un única neurona.

```
test$fit<-predict(nnG2,newdata = test)
head(test$fit)
```

```
##                [,1]
## 215401 0.008905308
## 215402 0.008860374
## 215403 0.008877838
## 215404 0.008940303
## 215405 0.008841911
## 215406 0.008830999
```

Como hemos dicho, aquí si que se observan valores distintos para las predicciones. En este caso, la libraria nnet no proporciona desviaciones típicas de los valores de predicción para poder calcular las bandas superior e inferior del intervalo, y si se quisieran, habría que elaborar una estrategia Bootstrap.

Capítulo 5

Conclusión

Este proceso de detección de outliers sirve para todos los modelos que nos planteemos, pero no porque un modelo de un tipo tenga los mismos outliers que otro distinto, sino porque es un procedimiento bastante general. Si nos fijamos y evaluamos los outliers detalladamente, nos daremos cuenta de que podemos encontrar outliers para un modelo lineal pero que no coincidan con los outliers del modelo aditivo, por ejemplo, debido a que la relación entre la respuesta y las variables regresoras no tiene porqué ser lineal. Esto se puede observar en la tabla final, donde, efectivamente, los outliers detectados con el modelo lineal no son los mismos que los del modelo aditivo. De hecho, este último parece ser más flexible ya que la cantidad de datos atípicos es considerablemente menor.

Por tanto, aunque nuestro punto de partida sea la misma base de datos, nuestro procedimiento detectará como atípicos unos determinados valores u otros en función del tipo de modelo que estimemos, lo que dependerá de muchos y diversos factores que habría que tener en cuenta en función de los datos a trabajar y de nuestros objetivos y requisitos.

Se escogieron tres tipos de métodos para comprobar los resultados y observar posibles diferencias. Aunque ese era nuestro objetivo, con los modelos de redes neuronales no se ha podido finalizar como se deseaba. A pesar de ello y de la falta de variables retardadas, se han podido detectar los valores atípicos y las desviaciones deseadas. En el caso de los modelos lineales se ha podido indagar más al tener su propio método de detección de atípicos, el basado en los residuos estudentizados, y poder compararlo con nuestro enfoque. Mencionar que el procedimiento que hemos planteado parece haberse adaptado tanto al modelo lineal como al aditivo de forma correcta y, posiblemente, también se ajuste al modelo de redes neuronales adecuacamente.

Modelo Lineal	Modelo aditivo
2019-12-18 15:14:30, 2019-12-18 15:14:34, 2019-12-18 15:14:35,	2019-12-18 15:52:24
2019-12-18 15:14:38, 2019-12-18 15:14:41, 2019-12-18 15:14:44,	2019-12-18 15:52:30
2019-12-18 15:14:49, 2019-12-18 15:14:50 , 2019-12-18 15:14:52,	2019-12-18 15:52:58
2019-12-18 15:14:59, 2019-12-18 15:15:00, 2019-12-18 15:15:01,	
2019-12-18 15:15:02, 2019-12-18 15:15:07, 2019-12-18 15:15:08,	
2019-12-18 15:15:09, 2019-12-18 15:15:30, 2019-12-18 15:15:32,	
2019-12-18 15:15:41, 2019-12-18 15:15:43, 2019-12-18 15:15:50,	
2019-12-18 15:15:56, 2019-12-18 15:16:01, 2019-12-18 15:16:05,	
2019-12-18 15:16:07, 2019-12-18 15:16:10, 2019-12-18 15:16:16,	
2019-12-18 15:16:18, 2019-12-18 15:16:21, 2019-12-18 15:16:31,	
2019-12-18 15:16:44, 2019-12-18 15:23:25, 2019-12-18 15:23:37,	
2019-12-18 15:23:42, 2019-12-18 15:23:43, 2019-12-18 15:23:51,	
2019-12-18 15:23:52, 2019-12-18 15:23:55, 2019-12-18 15:23:58,	
2019-12-18 15:23:59, 2019-12-18 15:24:00, 2019-12-18 15:24:01,	
2019-12-18 15:24:10, 2019-12-18 15:24:12, 2019-12-18 15:24:17,	
2019-12-18 15:24:20, 2019-12-18 15:24:21, 2019-12-18 15:24:24,	
2019-12-18 15:24:25, 2019-12-18 15:24:26, 2019-12-18 15:24:32,	
2019-12-18 16:14:10, 2019-12-18 17:04:19, 2019-12-18 17:04:20,	
2019-12-18 17:11:55, 2019-12-18 17:12:20, 2019-12-18 17:12:30,	
2019-12-18 17:20:46, 2019-12-18 17:21:01	

Bibliografía

[1] Buja A, Hastie T.J, Tibshirani R. (1989). Linear smoothers and additive models. The Annals of Statistics, Vol. 17..

[2] Friedman J, Hastie T, Tibshirani R (2008) The Elements of Statistical Learning. Data Mining, Inference and Prediction. Second Edition.

[3] James G, Witten D, Hastie T, Tibshirani R. (2014) An Introduction to Statistical Learning: With Application in R. Springer Publishing Company.

[4] Kröse B, Van Der Smagt P. (1996) An introduction to Neural Networks. Eighth Edition.

[5] Marcus W.Beck (2018) NeuralNetTools: Visualization and Analysis Tools for Neural Networks. Journal of Statistical Software. Volume 85, Issue 11. https://www.jstatsoft.org/article/view/v085i11

[6] Saavedra P, Ramil L.(2019-2020) Apuntes de la asignatura de Modelos de Regresión. Universidad de Santiago de Compostela.

[7] Wood, Simon N. (2006) Generalized Additive Models: An Introduction with R. Chapman and Hall/CRC.