Trabajo Fin de Máster

Estudio de los errores en las predicciones de generación eléctrica a corto plazo de una planta fotovoltaica.

Julián Villanustre Otero

Máster en Técnicas Estadísticas

Curso 2022-2023

Propuesta de Trabajo Fin de Máster

Título en galego: Estudo dos erros nas predicións de xeración elétrica a curto prazo dunha planta fotovoltaica.
Título en español: Estudio de los errores en las predicciones de generación eléctrica a corto plazo de una planta fotovoltaica.
English title: Study of the errors in the predictions of electricity generation in the short term of a photovoltaic plant.
Modalidad: Modalidad B
Autor/a: Julián Villanustre Otero, Universidade de Santiago de Compostela
Director/a: Javier Roca Pardiñas, Universidade de Vigo
Tutor/a: Rogelio Peón Menendez, TSK; Luis Millan Monte, TSK
Breve resumen del trabajo: Disponemos de un sistema predictivo para estimar la producción de una planta fotovoltaica en los próximos 15 minutos a intervalos de 1 minuto. A posteriori conocemos el valor real de la variable predicha y, por tanto, los errores cometidos en dichas 15 predicciones. El objetivo es reproducir estos errores de forma coherente para introducirlos más tarde en un sistema de simulación que tiene la empresa. Para ello, se prueban multiples enfoques y finalmente se resuelve utilizando modelos GAM.

Don/doña Javier Roca Pardiñas, Catedrático de la Universidade de Vigo, don/doña Rogelio Peón Menendez, Director de Tecnología de TSK, y don/doña Luis Millan Monte, Ingeniero en el departamento de tecnología de TSK, informan que el Trabajo Fin de Máster titulado

Estudio de los errores en las predicciones de generación eléctrica a corto plazo de una planta fotovoltaica.

fue realizado bajo su dirección por don/doña Julián Villanustre Otero para el Máster en Técnicas Estadísticas. Estimando que el trabajo está terminado, dan su conformidad para su presentación y defensa ante un tribunal.

En Santiago de Compostela, a 28 de Enero de 2023.

ROCA PARDIÑAS JAVIER - 77592181W

Firmado digitalmente por ROCA PARDIÑAS JAVIER - 77592181W
Fecha: 2023.01.27 20:29:23 +01'00'

El/la director/a:
Don/doña Javier Roca Pardiñas

El/la autor/a:
Don/doña Julián Villanustre Otero

Declaración responsable. Para dar cumplimiento a la Ley 3/2022, de 24 de febrero, de convivencia universitaria, referente al plagio en el Trabajo Fin de Máster (Artículo 11, Disposición 2978 del BOE núm. 48 de 2022), **el/la autor/a declara** que el Trabajo Fin de Máster presentado es un documento original en el que se han tenido en cuenta las siguientes consideraciones relativas al uso de material de apoyo desarrollado por otros/as autores/as:

- Todas las fuentes usadas para la elaboración de este trabajo han sido citadas convenientemente (libros, artículos, apuntes de profesorado, páginas web, programas,...)
- Cualquier contenido copiado o traducido textualmente se ha puesto entre comillas, citando su procedencia.
- Se ha hecho constar explícitamente cuando un capítulo, sección, demostración,... sea una adaptación casi literal de alguna fuente existente.

Y, acepta que, si se demostrara lo contrario, se le apliquen las medidas disciplinarias que correspondan.

Índice general

Resumen IX

Introducción XI

1. Primeras ideas 1
- 1.1. Modelo de localización y escala . . . 1
- 1.2. Series de Tiempo . . . 4
  - 1.2.1. VARMA . . . 5
  - 1.2.2. ARCH . . . 6

2. Solución 7
- 2.1. Modelos GAM . . . 7
- 2.2. Selección de variables . . . 9
- 2.3. Ajuste del modelo . . . 16
- 2.4. Validación . . . 20

3. Conclusión 27

A. Algoritmo de selección de variables 29

Bibliografía 31

Resumen

Resumen en español

En las próximas páginas se presenta la memoria de las prácticas del alumno Julián Villanustre Otero. Se introduce el problema propuesto por la empresa TSK, estudiar un registro histórico de errores en la predicción de energía eléctrica generada, con el fin de poder reproducirlos e incorporarlos a una simulación. Luego se mencionan las distintas ideas que fueron surgiendo durante el desarrollo de las prácticas y se explica brevemente por qué se desecharon. Finalmente, se explican los modelos GAM, el método de selección de variables utilizado y se ajustan los modelos, terminando por presentar los resultados obtenidos.

English abstract

In the following pages, we tell the memories of the practices by Julián Villanustre Otero. First of all, we introduce the problem, to study a historic register of error with the objetive of reproduce them, for implement them in a simulation. Then, we mention the different approach that we considered through the project, and we explain because we rejected them. Finally, we show GAM models and the method for variables selection that we have used. Finally we submit the results.

Introducción

En las próximas páginas se presentará la memoria de las prácticas del alumno Julián Villanustre Otero con la empresa TSK.

Comenzaremos por presentar la empresa que oferta las prácticas. TSK es una compañía tecnológica y de servicios, que ofrece soluciones eficientes, sostenibles y digitales para el sector industrial y energético. Cuenta con más de 35 años de experiencia y más de 1000 empleados. Tiene proyectos en más de 50 paises y trabaja una gran diversidad de sectores, entre ellos podemos destacar siderurgia, cemento, papel, azúcar, fertilizantes, *gas to power*, puertos, tratamientos de aguas, medio ambiente, energía, almacenamiento y transporte de materias primas. Dentro de todos los sectores en los que trabajan, se encuentra el campo de las energías renovables, en el cual encuadramos el proyecto objeto de las prácticas.

En **TSK** desarrollan soluciones que ayudan a sus clientes en el camino hacia la eficiencia energética, la descarbonización y la digitalización de sus actividades. Sus principales objetivos son la reducción de costes, el incremento de la productividad, el ahorro energético, la disminución del impacto medioambiental y la mejora de las condiciones de trabajo de los trabajadores. Dentro del sector de las energías renovables, las soluciones aportadas por **TSK** integran multitud de tecnologías, entre las cuales están las plantas híbridas, que combinan la tecnología de las centrales solares fotovoltaicas con la tecnología convencional de las plantas de motores de gasoil.

El proyecto propuesto por la empresa, consiste en el estudio de los errores en las predicciones de generación eléctrica a corto plazo de una planta fotovoltaica. La motivación para predecir la electricidad

generada, es la creación de plantas híbridas, donde se combina una planta fotovoltaica con una planta de motores para mantener una producción estable, ya que la producción en plantas fotovoltaicas depende fuertemente de la radiación recibida, la cual no es constante. Por ello, la planta de motores de gasoil se utiliza para suplir la energía faltante en periodos nocturnos o de baja producción en la central fotovoltaica.

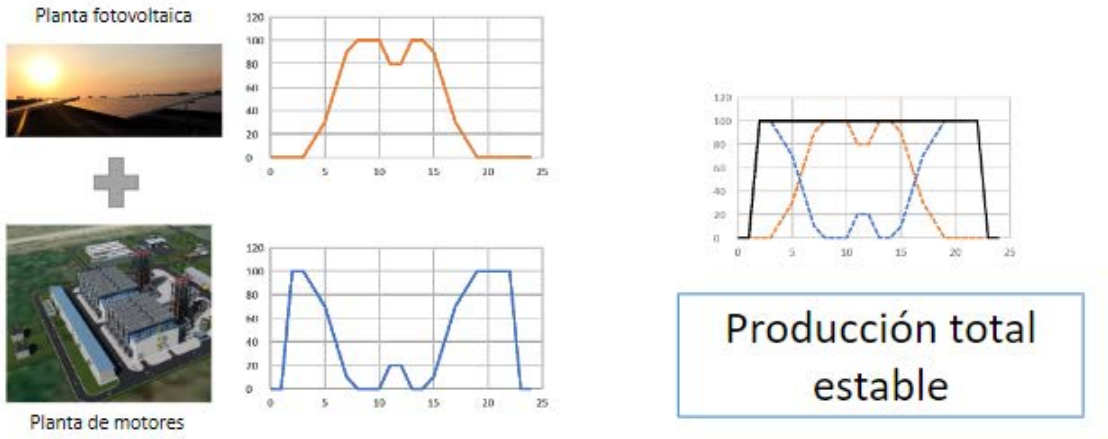


Figura 1: Gráficas de la producción de una central fotovoltaica y una central de motores.

La planta de motores tarda 5 minutos en arrancar, van ganando potencia paulatinamente, por eso es importante saber con algo de antelación cuánta energía se producirá en los próximos minutos en la planta fotovoltaica. Esto se ve gráficamente en la Figura 1. Para estimar la producción eléctrica se utilizan unas cámaras de nubes, que predicen la radiación que habrá en los próximos minutos; a su vez la radiación explica la energía que vamos a obtener cometiendo un determinado error, el cual nosotros intentaremos modelar para poder reproducirlo.

TSK ha registrado las predicciones que se hacen en cada instante t para los próximos 15 minutos y la radiación en cada instante t. También disponen da la producción real, por tanto tienen un registro histórico del error cometido, que son junto a la radiación los datos que nos envían. Denotaremos los errores por Y_i, donde i es el horizonte de la predicción de energía eléctrica para el que se está cometiendo el error. El objetivo del proyecto es modelar los errores con el fin de incorporarlos a una simulación. En dicha simulación nosotros solo disponemos de la radiación como información *a priori*. Los errores son las variables respuesta que queremos predecir.

Primero leemos los datos del fichero excel que nos proporcionó TSK y mostramos las primeras filas del *data.frame* para ir familiarizándonos con los datos.

```
##   t rad   y1   y2   y3   y4   y5   y6   y7   y8  y9 y10 y11 y12  y13  y14  y15
## 1 1 0.0 -0.2 -0.4 -0.6 -0.8 -1.2 -1.6 -1.6 -0.8 0.6 2.8 5.4 8.0  9.6 10.8 12.4
## 2 2 0.2 -0.4 -0.6 -0.8 -1.2 -1.6 -1.2 -0.8  0.4 2.4 5.0 7.0 8.6 10.0 11.0 12.0
## 3 3 0.4 -0.2 -0.4 -0.8 -1.2 -1.2 -0.6  0.0  1.8 4.2 6.2 7.6 9.0 10.2 10.6 11.4
## 4 4 0.6 -0.2 -0.6 -1.0 -1.2 -1.0 -0.2  1.0  3.2 5.2 6.4 7.8 8.8  9.6  9.8 10.6
## 5 5 0.8 -0.4 -0.8 -1.0 -1.2 -0.8  0.6  2.2  3.8 5.4 6.6 7.8 8.0  9.0  9.2  9.8
## 6 6 1.2 -0.4 -0.8 -1.0 -1.0  0.0  1.6  3.0  4.4 5.6 6.8 7.4 7.4  8.4  8.6  9.0
```

Por filas, tenemos el instante en el que nos encontramos y por columnas, los errores cometidos al predecir la energía eléctrica generada con distintos horizontes. Recordemos que Y_i son los errores

que se cometen al predecir en el instante t la producción eléctrica en el instante $t + i$. Cabe esperar que cuanto mayor sea el índice i mayores sean los errores cometidos. Presentamos un *summary* de los datos.

```
#Resumen de los datos. Medidas de localización.
summary(datos)
```

```
##        t              rad               y1                 y2
##  Min.   :   1   Min.   :   0.0   Min.   :-243.000   Min.   :-333.800
##  1st Qu.:1665   1st Qu.: 189.4   1st Qu.:  -3.200   1st Qu.:  -6.800
##  Median :3329   Median : 410.8   Median :   0.000   Median :  -0.200
##  Mean   :3329   Mean   : 454.9   Mean   :  -1.867   Mean   :  -3.768
##  3rd Qu.:4993   3rd Qu.: 734.6   3rd Qu.:   1.400   3rd Qu.:   2.600
##  Max.   :6657   Max.   :1239.4   Max.   : 328.800   Max.   : 417.000
##        y3                 y4                 y5                 y6
##  Min.   :-475.400   Min.   :-527.200   Min.   :-661.000   Min.   :-688.40
##  1st Qu.: -11.000   1st Qu.: -14.800   1st Qu.: -19.200   1st Qu.: -23.00
##  Median :  -0.400   Median :  -0.600   Median :  -0.600   Median :  -0.80
##  Mean   :  -5.583   Mean   :  -7.273   Mean   :  -9.267   Mean   : -11.02
##  3rd Qu.:   3.800   3rd Qu.:   4.800   3rd Qu.:   6.000   3rd Qu.:   6.80
##  Max.   : 446.400   Max.   : 560.000   Max.   : 729.000   Max.   : 703.80
##        y7                 y8                 y9                y10
##  Min.   :-743.40   Min.   :-786.6   Min.   :-830.80   Min.   :-834.20
##  1st Qu.: -27.00   1st Qu.: -30.2   1st Qu.: -34.60   1st Qu.: -38.00
##  Median :  -1.00   Median :  -1.2   Median :  -1.60   Median :  -1.80
##  Mean   : -12.66   Mean   : -14.7   Mean   : -17.07   Mean   : -18.31
##  3rd Qu.:   7.60   3rd Qu.:   8.8   3rd Qu.:   9.60   3rd Qu.:  10.20
##  Max.   : 690.40   Max.   : 736.6   Max.   : 769.80   Max.   : 817.40
##       y11                y12                y13                y14
##  Min.   :-819.00   Min.   :-806.60   Min.   :-813.00   Min.   :-827.20
##  1st Qu.: -40.00   1st Qu.: -43.20   1st Qu.: -45.20   1st Qu.: -46.80
##  Median :  -2.00   Median :  -2.20   Median :  -2.60   Median :  -2.80
##  Mean   : -18.13   Mean   : -18.02   Mean   : -17.72   Mean   : -17.47
##  3rd Qu.:  11.20   3rd Qu.:  12.40   3rd Qu.:  13.40   3rd Qu.:  15.20
##  Max.   : 848.20   Max.   : 864.00   Max.   : 903.40   Max.   : 929.00
##       y15
##  Min.   :-835.00
##  1st Qu.: -46.20
##  Median :  -3.00
##  Mean   : -17.53
##  3rd Qu.:  16.80
##  Max.   : 914.00
```

Vemos que la mediana es próxima a 0 para todos los errores. Además el rango intercuartílico es bastante pequeño y próximo a 0 si lo comparamos con los valores máximos y mínimos, que al estar tan alejados de la mayoría de los datos, influyen considerablemente en la media. También podemos notar que como suponíamos el error tiende a ser mayor cuanto mayor es el índice i. Ahora veremos gráficamente qué forma tienen los datos en la Figura 2.

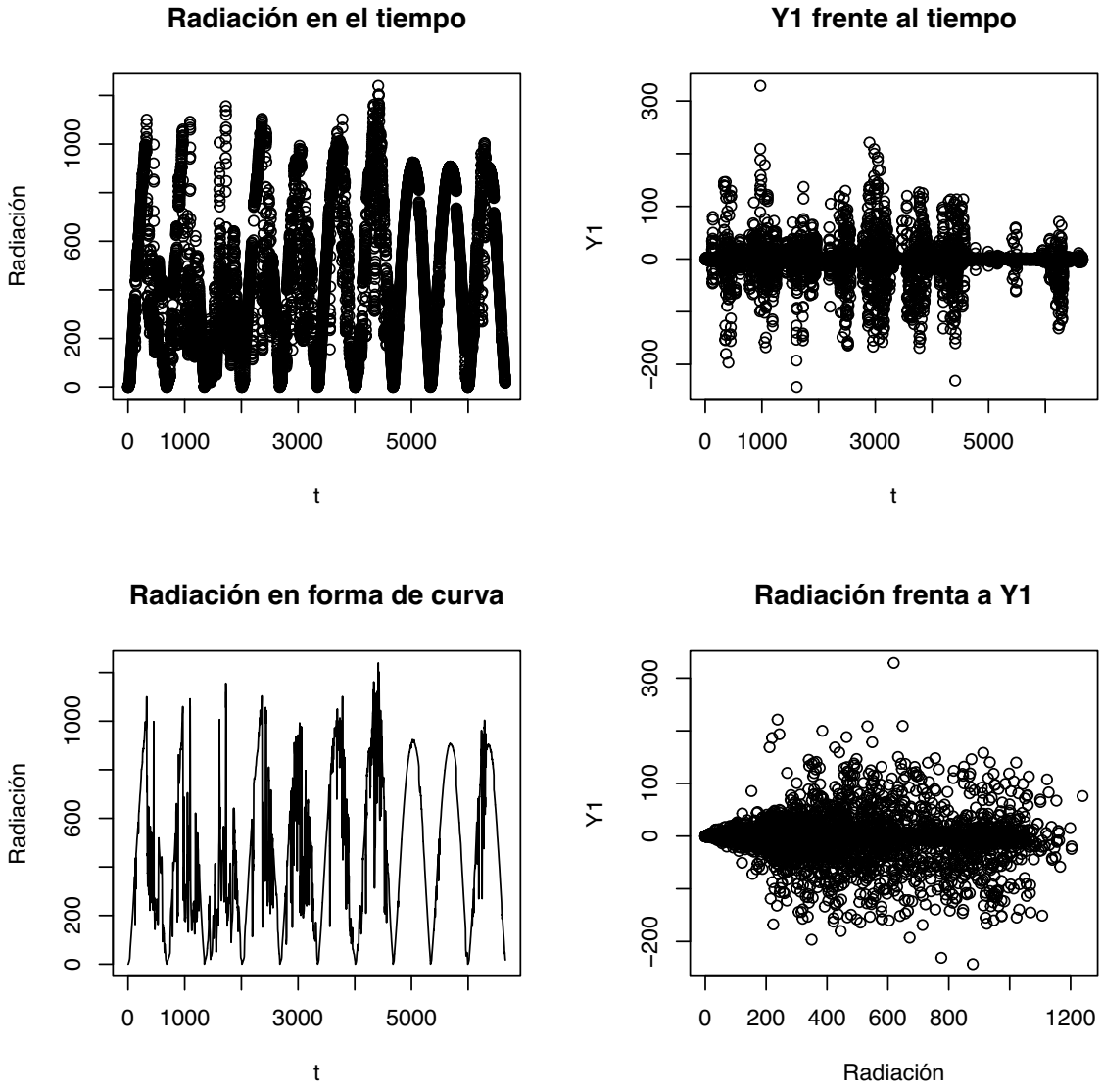


Figura 2: Gráficos para ilustrar la forma de los datos.

Observamos que la radiación tiene una forma periódica, lo cual es de esperar, ya que a lo largo del día el sol coge fuerza hasta el mediodía y luego la pierde paulatinamente hasta el atardecer y así sucesivamente. En el gráfico superior izquierdo no se llega a apreciar bien, pero en el inferior izquierdo, vemos que la radiación parece seguir una curva suave sumada a unas interferencias, que deducimos son efecto de la nubosidad. Respecto a los datos Y_1, se aprecia una relación temporal, podríamos pensar que está directamente relacionado con la radiación. Sin embargo en el último gráfico, solo apreciamos heterocedasticidad en la distribución de los errores Y_1 respecto de la radiación, pero en ningún caso vemos un patrón claro. Por ello en un primer momento nosotros no nos planteamos enfocar el problema como un problema de regresión, ya que a simple vista no hay variables explicativas. En la Figura 3 mostramos los gráficos de dispersión de varios errores frente a Y_1.

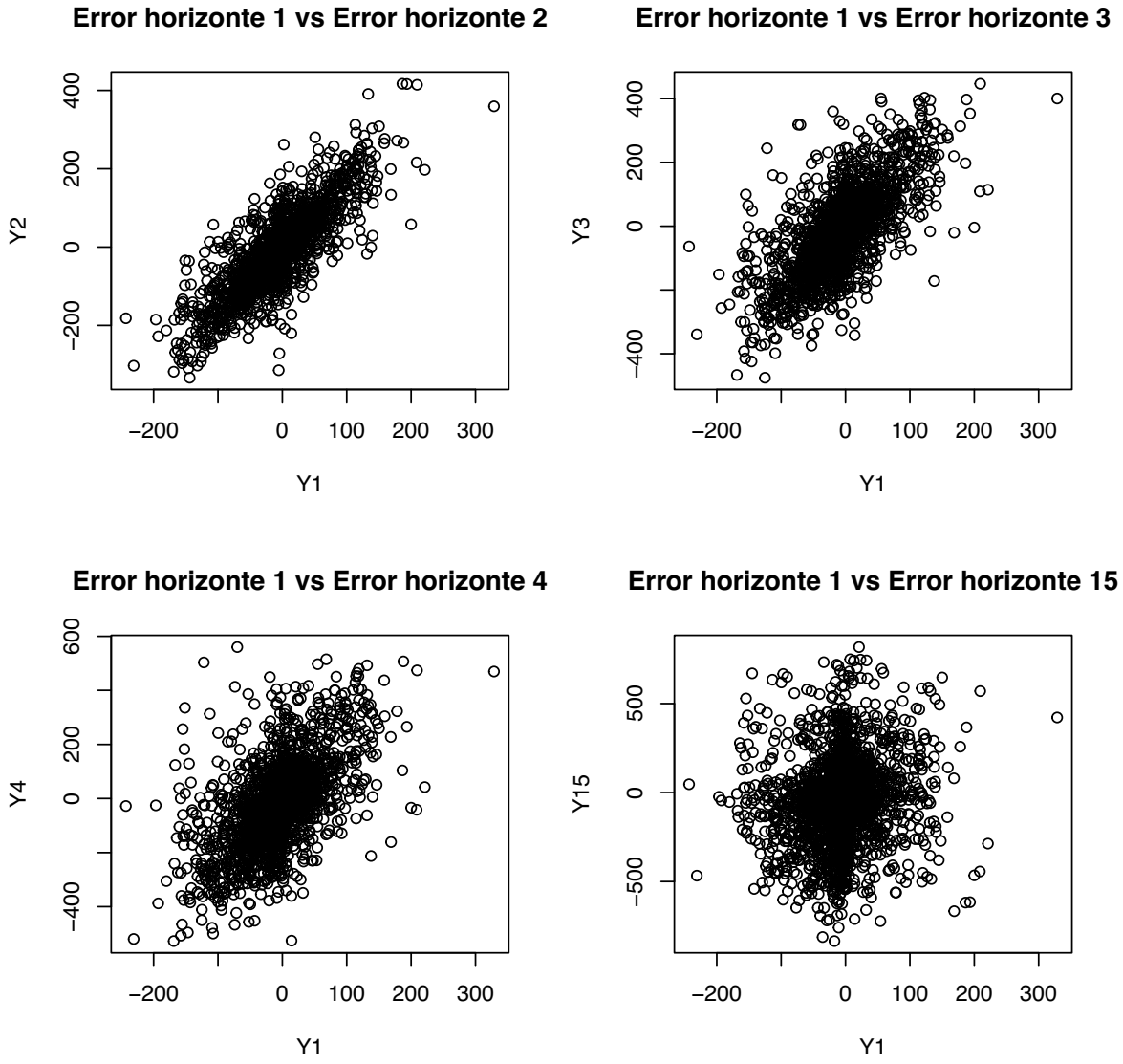


Figura 3: Distintos errores frente a Y_1.

Se aprecia claramente una relación lineal positiva entre el error Y_1 y el error Y_2. Esta relación desaparece según vamos tomando errores más alejados. Ahora calculamos la matriz de correlaciones para confirmar la conjetura de que *errores cercanos tienen alta correlación, mientras que errores más alejados tendrán coeficientes de correlación menores.*

```
##       y1   y2   y3   y4   y5   y6   y7   y8   y9  y10  y11  y12  y13  y14  y15
## y1  1.00 0.83 0.70 0.58 0.47 0.32 0.24 0.20 0.17 0.14 0.12 0.10 0.08 0.07 0.07
## y2  0.83 1.00 0.91 0.78 0.66 0.52 0.39 0.32 0.27 0.23 0.21 0.18 0.16 0.14 0.13
## y3  0.70 0.91 1.00 0.92 0.82 0.70 0.58 0.47 0.41 0.35 0.31 0.27 0.24 0.21 0.20
## y4  0.58 0.78 0.92 1.00 0.95 0.85 0.74 0.63 0.53 0.46 0.40 0.36 0.32 0.29 0.27
## y5  0.47 0.66 0.82 0.95 1.00 0.95 0.86 0.76 0.66 0.56 0.50 0.44 0.40 0.36 0.34
## y6  0.32 0.52 0.70 0.85 0.95 1.00 0.96 0.88 0.79 0.69 0.61 0.54 0.48 0.44 0.41
```

```
## y7  0.24 0.39 0.58 0.74 0.86 0.96 1.00 0.96 0.89 0.80 0.71 0.62 0.56 0.52 0.48
## y8  0.20 0.32 0.47 0.63 0.76 0.88 0.96 1.00 0.96 0.90 0.81 0.73 0.65 0.59 0.55
## y9  0.17 0.27 0.41 0.53 0.66 0.79 0.89 0.96 1.00 0.97 0.90 0.83 0.75 0.68 0.63
## y10 0.14 0.23 0.35 0.46 0.56 0.69 0.80 0.90 0.97 1.00 0.97 0.91 0.84 0.77 0.71
## y11 0.12 0.21 0.31 0.40 0.50 0.61 0.71 0.81 0.90 0.97 1.00 0.97 0.92 0.86 0.79
## y12 0.10 0.18 0.27 0.36 0.44 0.54 0.62 0.73 0.83 0.91 0.97 1.00 0.98 0.93 0.87
## y13 0.08 0.16 0.24 0.32 0.40 0.48 0.56 0.65 0.75 0.84 0.92 0.98 1.00 0.98 0.93
## y14 0.07 0.14 0.21 0.29 0.36 0.44 0.52 0.59 0.68 0.77 0.86 0.93 0.98 1.00 0.98
## y15 0.07 0.13 0.20 0.27 0.34 0.41 0.48 0.55 0.63 0.71 0.79 0.87 0.93 0.98 1.00
```

La matriz de correlación confirma lo que veíamos gráficamente, un error está fuertemente correlacionado con el siguiente. A medida que estos se alejan, pierden esa relación. Con este primer vistazo a los datos, pasamos a presentar algunos de los planteamientos que se le dieron al problema.

Capítulo 1

Primeras ideas

En este capítulo, haremos un breve recorrido por algunos de los enfoques que se le dieron al problema, pero finalmente fueron desechados. Esto nos servirá para comprender, con mayor profundidad, las particularidades de los datos con los que estamos trabajando. Mencionaremos aquellas ideas que consideramos más importantes y que tuvieron mayor desarrollo.

1.1. Modelo de localización y escala

El planteamiento inicial que se me propuso fue utilizar un modelo multivariante de localización y escala. A continuación presentamos el método.

Disponemos de $S = 6657$ series de $d = 15$ medidas de error $Y = (Y_1, \ldots, Y_d)$. Y tiene un vector de medias $\mu = (\mu_1, \ldots, \mu_d)$ y una matriz de covarianzas

$$\Sigma = \begin{pmatrix} \sigma_{11} & \cdots & \sigma_{1d} \\ \vdots & \ddots & \vdots \\ \sigma_{d1} & \cdots & \sigma_{dd} \end{pmatrix}. \tag{1.1}$$

Entonces Y se puede generar como $Y = \mu + \Sigma^{\frac{1}{2}} \cdot \epsilon$. Siendo la descomposición de Cholesky

$$\Sigma = \left(\Sigma^{\frac{1}{2}}\right)^t \cdot \Sigma^{\frac{1}{2}}$$

y $\epsilon = (\epsilon_1, \ldots, \epsilon_d)$ el vector de errores estandarizados con distribución libre que verifican,

$$\begin{aligned} &\mathbb{E}[\epsilon_i] = 0, \forall i \in 1, \ldots, 15 \\ &Var[\epsilon_i] = 1, \forall i \in 1, \ldots, 15 \\ &Cov[\epsilon_i, \epsilon_j] = 0, \forall i \neq j. \end{aligned}$$

La idea es obtener $\hat{\mu}, \hat{\Sigma}, \hat{\Sigma}^{\frac{1}{2}}$ y $\hat{F}_j(\epsilon_j)$ según el siguiente procedimiento:

1. Necesitaremos de una muestra de S series, de forma que para cada serie tengamos las correspon-

dientes d medidas de error. De este modo tendremos una matriz de $S \times d$ mediciones

$$\begin{pmatrix} Y_{11} & \cdots & Y_{1d} \\ \vdots & \ddots & \vdots \\ Y_{d1} & \cdots & Y_{dd} \end{pmatrix}.$$

2. Calcular el vector de medias de las medidas de error $\hat{\mu} = (\hat{\mu}_1, \ldots, \hat{\mu}_d)$ con $\hat{\mu}$ la media muestral

$$\hat{\mu}_j = \frac{\sum_{s=1}^{S} Y_{sj}}{S}.$$

3. A continuación, se obtiene la matriz ajustada de covarianzas

$$\Sigma = \begin{pmatrix} \sigma_{11} & \cdots & \sigma_{1d} \\ \vdots & \ddots & \vdots \\ \sigma_{d1} & \cdots & \sigma_{dd} \end{pmatrix}$$

siendo $\hat{\sigma_{jk}}$ la covarianza muestral

$$\hat{\sigma}_{jk} = \frac{\sum_{s=1}^{S} (Y_{sj} - \hat{\mu}_j)(Y_{sk} - \hat{\mu}_k)}{S}.$$

4. Se obtiene la descomposición de Cholesky de $\hat{\Sigma}$ que denotaremos por $\hat{\Sigma}^{\frac{1}{2}}$ (de forma que $(\hat{\Sigma}^{\frac{1}{2}})^t \cdot \hat{\Sigma}^{\frac{1}{2}} = \hat{\Sigma}$).

5. Se calculan los residuos estandarizados

$$\begin{pmatrix} \epsilon_{11} & \cdots & \epsilon_{1d} \\ \vdots & \ddots & \vdots \\ \epsilon_{S1} & \cdots & \epsilon_{Sd} \end{pmatrix}.$$

Con $\hat{\epsilon} = \Sigma^{-\frac{1}{2}}(X - \mu)$, es decir:

$$\begin{pmatrix} \epsilon_{s1} \\ \vdots \\ \epsilon_{sd} \end{pmatrix} = (\hat{\Sigma}^{-\frac{1}{2}})^t \begin{pmatrix} Y_{s1} - \hat{\mu}_1 \\ \vdots \\ Y_{sd} - \hat{\mu}_d \end{pmatrix} \quad \text{para } s = 1, \ldots, S.$$

6. Se rechazan todos los ϵ que no cumplan con un determinado criterio, por ejemplo, $|\epsilon| < t$, siendo t una constante, por ejemplo 6.

7. Se obtiene, para $j = 1, \ldots, d$ la función de distribución empírica de dichos errores

$$\hat{F}_j(\epsilon_j) = \frac{\sum_{s=1}^{S} I\left\{\hat{\epsilon}_{sj} \leq \epsilon_j\right\}}{S}$$

siendo $I\{A\}$ la función indicadora del suceso A.

8. Se aproxima la distribución del estadístico de contraste usando Montecarlo:

Para $b = 1, \ldots, B$ (B grande, por ejemplo $B = 50000$)

- Se obtienen los valores simulados $(Y_1^b, \ldots, Y_d^b)$ de acuerdo a

$$\begin{pmatrix} Y_1^b \\ \vdots \\ Y_d^b \end{pmatrix} = \begin{pmatrix} \hat{\mu}_1 \\ \vdots \\ \hat{\mu}_d \end{pmatrix} + \hat{\Sigma}^{\frac{-1}{2}} \begin{pmatrix} \epsilon_1^b \\ \vdots \\ \epsilon_d^b \end{pmatrix},$$

donde $(\epsilon_1^b, \ldots \epsilon_d^b)$ han sido generados de forma independiente, de forma que cada ϵ_j^b se simula a partir de la función $\hat{F}_j$.

La idea es estandarizar los datos, consiguiendo unos errores **linealmente independientes** y calcular la distribución empírica de cada error $\hat{F}_j(\epsilon_j)$. Una vez calculada, para simular nuevos datos, bastaría con generar de forma independiente errores estandarizados ϵ_j a partir de la distribución empírica y aplicar la transformación $X = \mu + \Sigma^{\frac{1}{2}} \cdot \epsilon$. Este enfoque se ve reforzado por lo visto anteriormente en la introducción, ya que si observamos la Figura 3 vemos una clara relación lineal entre errores cercanos. Sobre el papel era un planteamiento sencillo, fácil de implementar en `R`. El problema aquí fue que estábamos suponiendo que tras extraer la relación lineal de los datos, obtendríamos independencia. Sin embargo, resultó que los datos presentaban una relación no lineal, como se puede observar en la Figura 1.1. Por lo cual al validar el modelo no obteníamos los resultados esperados.

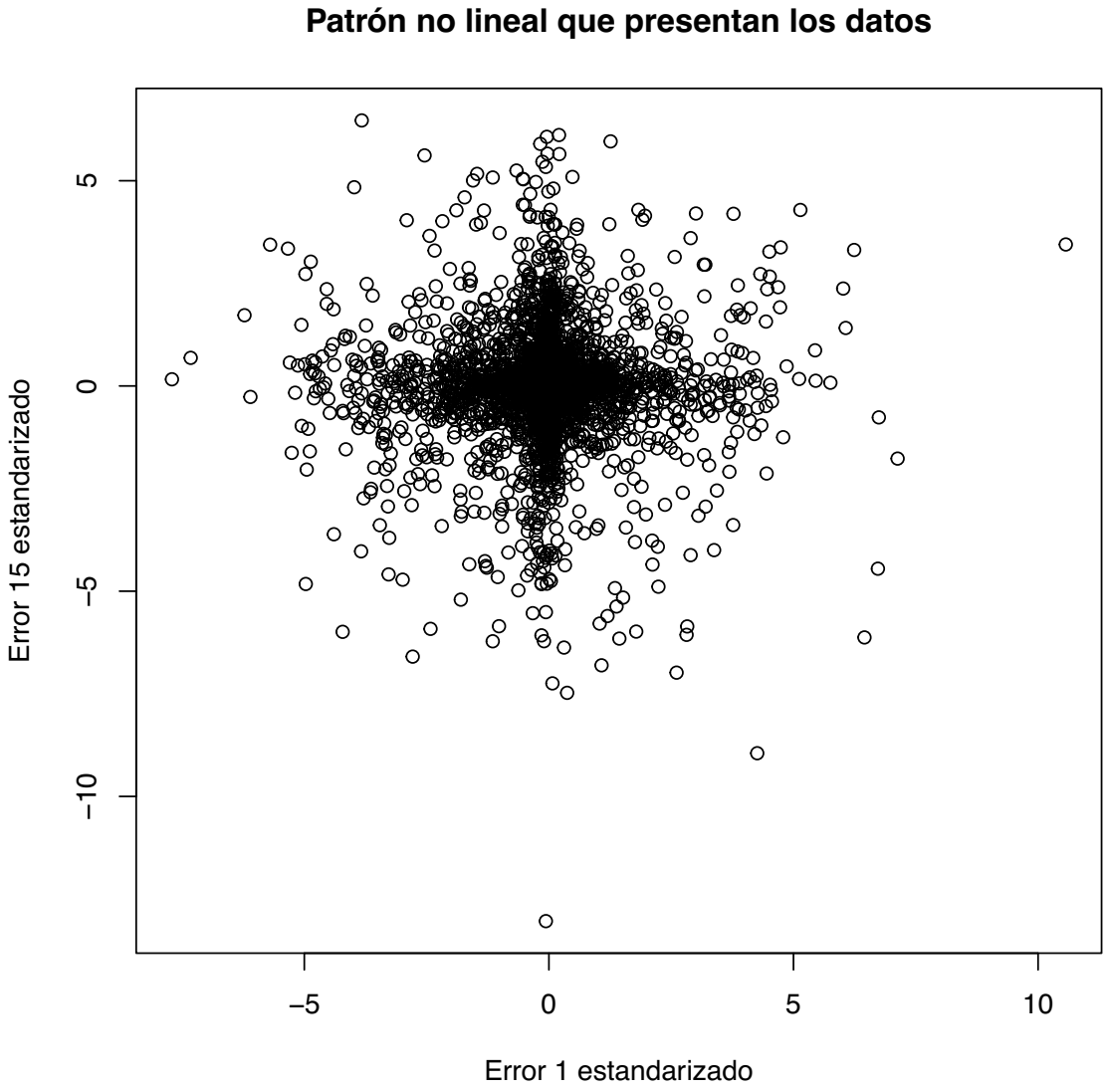


Figura 1.1: Relación no lineal entre los errores estandarizados ϵ_1 y ϵ_{15}

Cabe mencionar, que se puso sobre la mesa la opción de extraer la relación no lineal mediante el uso de cópulas. Este planteamiento fue rápidamente descartado debido a su complejidad y a la falta de garantías sobre la eficacia del método para nuestro problema. Además como veremos en la siguiente sección, aunque el método funcionase, los resultados obtenidos no tendrían ninguna utilidad, ya que al parecer hay relación temporal. Por lo tanto, simular errores sin tener en cuenta el orden de los mismos no daría resultados fieles a la realidad.

1.2. Series de Tiempo

Tras descartar el enfoque inicial del proyecto, comenzó la odisea de encontrar una nueva solución. Lo primero fue ver qué forma tenían los datos y comó se distribuían, en busca de alguna característica

que pudiésemos aprovechar. Al representar los datos por columnas (Figura 1.2) apreciamos una fuerte correlación temporal, lo cual significaba que el enfoque inicial habría producido un resultado muy malo aunque hubiese cumplido todas las hipótesis, ya que lo único que se tenía en cuenta era la distribución empírica, que no entiende de orden. Anteriormente se habían planteado los datos como S series de longitud $d = 15$, esta vez cambiábamos el enfoque, suponíamos que estábamos ante d series de tamaño $S = 6657$. Por lo tanto, nos pareció razonable afrontar el problema desde la teoría de las series de tiempo.

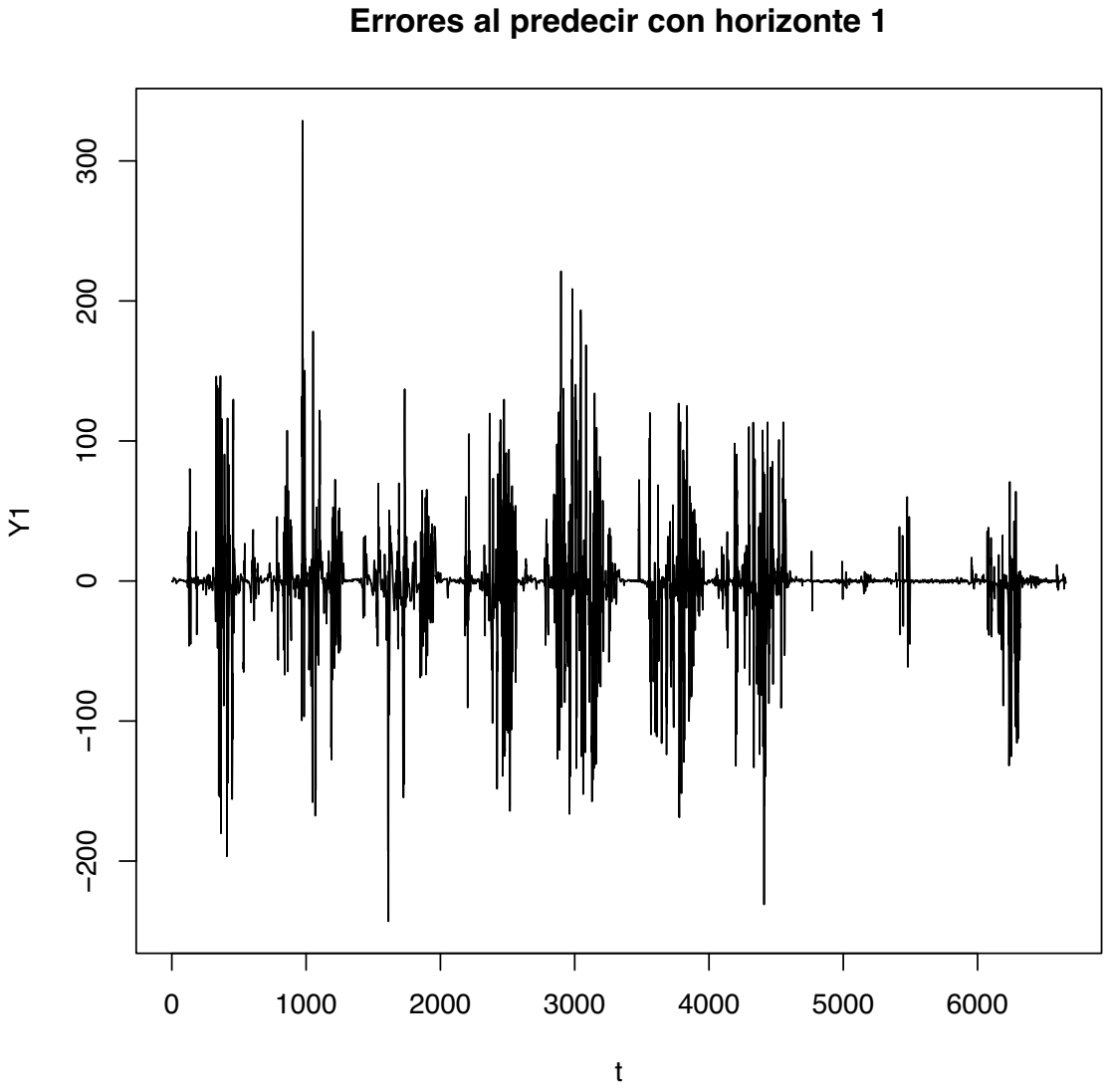


Figura 1.2: Al representar el error Y_1 en el tiempo, vemos una clara estructura temporal.

1.2.1. VARMA

La primera propuesta fue modelar los datos utilizando un modelo VARMA (Vector Auto-Regressive Moving Average), que resumiendo es un modelo que además de modelar la correlación temporal como

los modelos ARMA, también intenta captar la relación que existe entre las distintas series temporales. Si el lector quiere profundizar en el tema puede consultar Reinsel (2003) y Tsay (2013). Teníamos la facilidad de que es un método que ya está implementado en `R` en el paquete *MTS* (Lachmann *et al.* 2022). Sin embargo, al empezar a leer sobre el tema, vimos que toda la teoría se cimentaba sobre la hipótesis de que la serie temporal era estacionaria. Además el estudio de series temporales se basa en gran medida en contrastes de hipótesis, que sin estacionariedad, dejan de ser fiables. Cuando la condición de ser estacionaria se rompe por la existencia de una tendencia, existen soluciones como por ejemplo utilizar una transformación Box-Cox. En nuestro caso el problema era más complicado, en nuestros datos se observaba heterocedasticidad, además, esta no se limitaba a aumentar con el tiempo, sino que iba variando. Se puede comprobar en la Figura 1.2.

1.2.2. ARCH

Tras descartar el uso de un modelo VARMA, investigamos sobre qué métodos existían para modelar series temporales heterocedásticas. Lo cual, nos condujo a la regresión en series de tiempo y los modelos ARCH (Autoregressive Conditional Heteroskedasticity). La idea de los modelos ARCH es modelar la varianza de la serie temporal como una función de los residuos (h_t) respecto a una media, que podría ser 0 o ser la media estimada por un modelo de regresión.

Se dice que un proceso $\{y_t\}_{t\in I}$ sigue un modelo ARCH de orden p si:

$$\begin{aligned} &y_t|\psi_{t-1} \sim N(\mu_t, h_t) \\ &h_t = \phi_0 + \phi_1\epsilon_{t-1}^2 + \cdots + \phi_p\epsilon_{t-p}^2 + \nabla_t \\ &\epsilon_t = y_t - \mu_t. \end{aligned}$$

Donde ψ_{t-1} es la infomación disponible en el instante t sobre lo que ocurrió en los instantes anteriores, μ_t es la media del proceso en el instante t y ϕ_i con $i = 0, \ldots, p$ son constantes. Para más información sobre modelos ARCH el lector puede consultar Engle (1982), Bera y Higgins (1993) y Wei (2006). Este método cumplía con las hipótesis, sin embargo, tiene una limitación. Para obtener buenos resultados requiere que la varianza varíe de forma paulatina, sin brusquedades, justo al contrario que nuestros datos, como se puede observar en Figura 1.2. Con todo, mientras estudiábamos este método, contemplamos la posibilidad de centrarse en hacer regresión. Lo único que necesitábamos era encontrar variables explicativas razonablemente buenas.

Capítulo 2

Solución

Desde el inicio del proyecto, nos habíamos centrado en los errores, ya que entre ellos veíamos una fuerte relación. Luego notamos que a lo largo del tiempo también se relacionaban, pero estábamos pasando por alto una variable que era dato y que en un inicio no nos pareció relevante. La variable que estamos mencionando es la radiación, que en un primer análisis descriptivo pasa desapercibida, pero si miramos con detenimiento el gráfico de la radiación frente a los errores, observamos que las zonas con mayor error se corresponden con las zonas donde la radiación varía bruscamente. Por lo tanto parece razonable enfocar el problema como un problema de regresión donde la variable explicativa es la derivada de la radiación. Respecto al método de regresión utilizado, nos decantamos por los modelos GAM (Modelo aditivo generalizado), un modelo no paramétrico que permite mayor flexibilidad que los modelos lineales sin sacrificar interpretabilidad. A lo largo del capítulo explicaremos brevemente en que consisten los modelos GAM. También seleccionaremos por validación cruzada el mejor modelo y representaremos los resultados obtenidos.

2.1. Modelos GAM

Antes de ajustar el modelo presentaremos brevemente los modelos GAM (Modelos Aditivos Generalizados). Tanto para la explicación de los modelos aditivos como para su posterior ajuste en `R`, nos guiaremos por el libro de Fernández Casal *et al.*(2021) disponible en abierto.

Sea Y la variable respuesta y $X_1, \ldots, X_p$ las variables explicativas. Un modelo aditivo es de la forma:

$$Y = \mathcal{B}_0 + f_1(X_1) + f_2(X_2) + \cdots + f_p(X_p) + \epsilon$$

con f_i $i = 1, \ldots, p$, funciones suaves cualesquiera, $\mathcal{B}_0$ una constante que equivale al intercepto en la regresión lineal y ϵ errores independientes y normales. Si además consideramos funciones enlace, al estilo de la regresión lineal generalizada, estaremos hablando de modelos aditivos generalizados. Nótese que la regresión lineal generalizada es un caso particular de modelo GAM con $f_i(x) = \mathcal{B}_i x$, donde $\mathcal{B}_i$ son constantes. Para ajustar modelos GAM nosotros utilizaremos la función **gam()** del paquete *mgcv* de `R`, que admite efectos no paramétricos y lineales. En nuestro caso particular utilizaremos efectos no paramétricos, en concreto las funciones f_i que usaremos estarán formadas por *splines* de regresión. Los *splines* son curvas suaves definidas a trozos por polinomios de bajo grado, son muy utilizados para ajustar curvas, debido a que son fáciles de representar y no son costosos a nivel computacional. Para más información sobre *splines* se puede consultar en la literatura Ahlberg *et. al* (1967), Bartels *et. al* (1987) o Davis (1997).

Simplificando mucho lo que hace la función **gam()** es construir una base óptima del espacio de funciones, por defecto de tamaño $k = 9$, formada por *splines* penalizados. En dicha base representa las funciones suaves f_i. Luego, reformula el modelo como un modelo GLM, para posteriormente ajustarlo

adaptando el método *penalized re-weighted iterative least squares* (PIRLS). Para más detalles sobre este tipo de modelos ver la ayuda de `R` (Wood, S. 2006) o por ejemplo Hastie y Tibshirani (1990) o Wood (2017).

Para arrojar un poco de luz sobre el concepto de base *BSplines*, representamos una base de tamaño 9 en la Figura 2.1. La idea es que todas las funciones suaves con dominio $[0, 1]$ se pueden expresar como combinación lineal de las curvas que forman la base. Es decir, la base *BSplines* óptima, es una base fija, no depende de los valores de la función que tiene que representar, solo de su dominio y del tamaño que fijemos para la base.

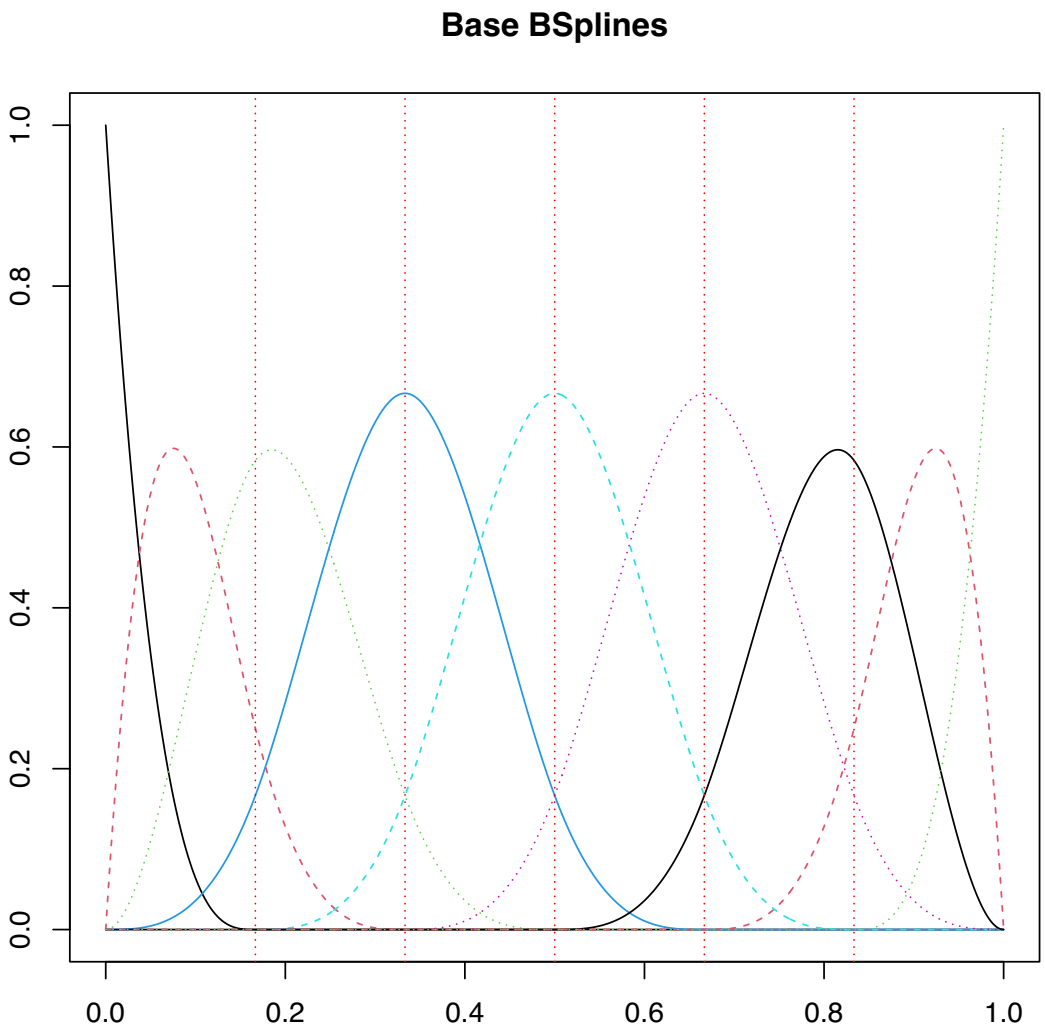


Figura 2.1: Base BSplines de tamaño $k = 9$ para funciones continuas en (0,1).

2.2. Selección de variables

Una vez presentados los modelos GAM, debemos hacernos cargo de una de las partes clave de la regresión, la selección de variables. Como ya sabemos un mayor número de variables explicativas no siempre conducen a un mejor modelo. Existe una gran variedad de metodos de selección de variables. En Fernández Casal *et al.* (2021) el autor utiliza el contraste ANOVA, donde se contrastan 2 modelos, uno que llamaremos modelo simple frente a otro más complejo, el cual suele ser el modelo simple sumándole una nueva variable explicativa. Este método tiene el inconveniente de que el contraste devuelve un nivel de significación utilizando el valor empírico de un estadístico y su distribución asintótica. Esta última depende de que el modelo cumpla las hipótesis de normalidad e independencia de los residuos. En la realidad esta condición es bastante idílica, como ya veremos nuestro modelo no cumple esas hipótesis, por lo que los resultados del contraste ANOVA no serían fiables. Como alternativa para la selección de variables, se puede escoger una medida de error y ajustar modelos, seleccionando aquel que cometa menos error. Nosotros lo que proponemos es utilizar validación cruzada para estimar la precisión de los modelos y mediante un algoritmo ajustar todas las posibles combinaciones que nosotros creamos convenientes. La validación cruzada es un método muy utilizado en aprendizaje estadístico y *machine learning* para evaluar la precisión de modelos tanto de clasificación como de regresión. Tal y como se ilustra en la Figura 2.2, consiste en dividir los datos de forma aleatoria en una muestra *train*, que se utilizará para ajustar el modelo y una muestra *test*, la cual se intentará predecir utilizando el modelo entrenado previamente. La idea es hacer este proceso varias veces, por ejemplo 10 veces y promediar la medida de error con la que estemos trabajando. En este caso particular utilizaremos

$$R^2 = 1 - \frac{\sum \text{Predicciones} - \text{Observaciones}}{\sum \text{Media de las observaciones} - \text{Observaciones}}$$

como medida de error. Nótese que el valor óptimo de precisión sería $R^2 = 1$. Además, cuanto más error tengamos en las predicciones más bajo será el valor de R^2, pudiendo llegar a ser negativo en caso de que las predicciones alcancen un error mayor al cometido por la media de las observaciones. Nosotros seleccionamos esta medida de error, porque nos parece que aporta infomación importante, ya que la alternativa más sencilla para resolver el problema es tomar la media de las observaciones. Con lo cual, mejorar el error cometido por la media es un requisito absolutamente imprescindible.

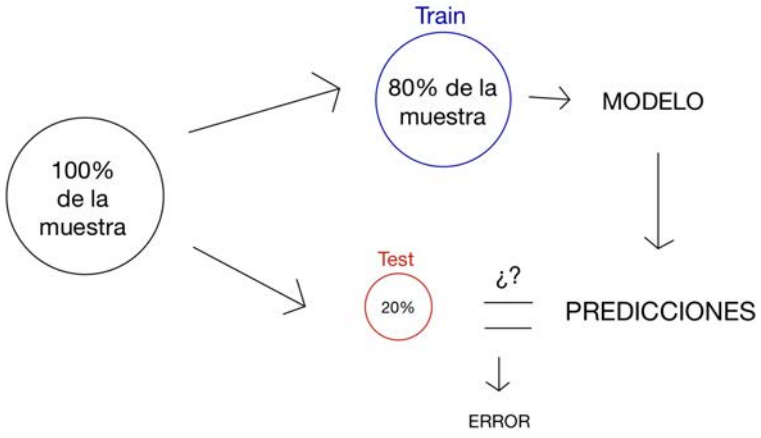


Figura 2.2: Esquema del proceso de validación cruzada en cada iteración (por ejemplo nosotros hacemos esto 10 veces para cada modelo).

Algo primordial en este enfoque, es encontrar algunas variables explicativas que ajusten razonablemente bien los errores Y_i. Ese será nuestro objetivo en las próximas páginas. Recordemos la forma de la radiación (Figura 2.3), este es el único dato de entrada del que disponemos, ya que para nosotros Y_i con $i = 1, \ldots, 15$ son variables respuesta.

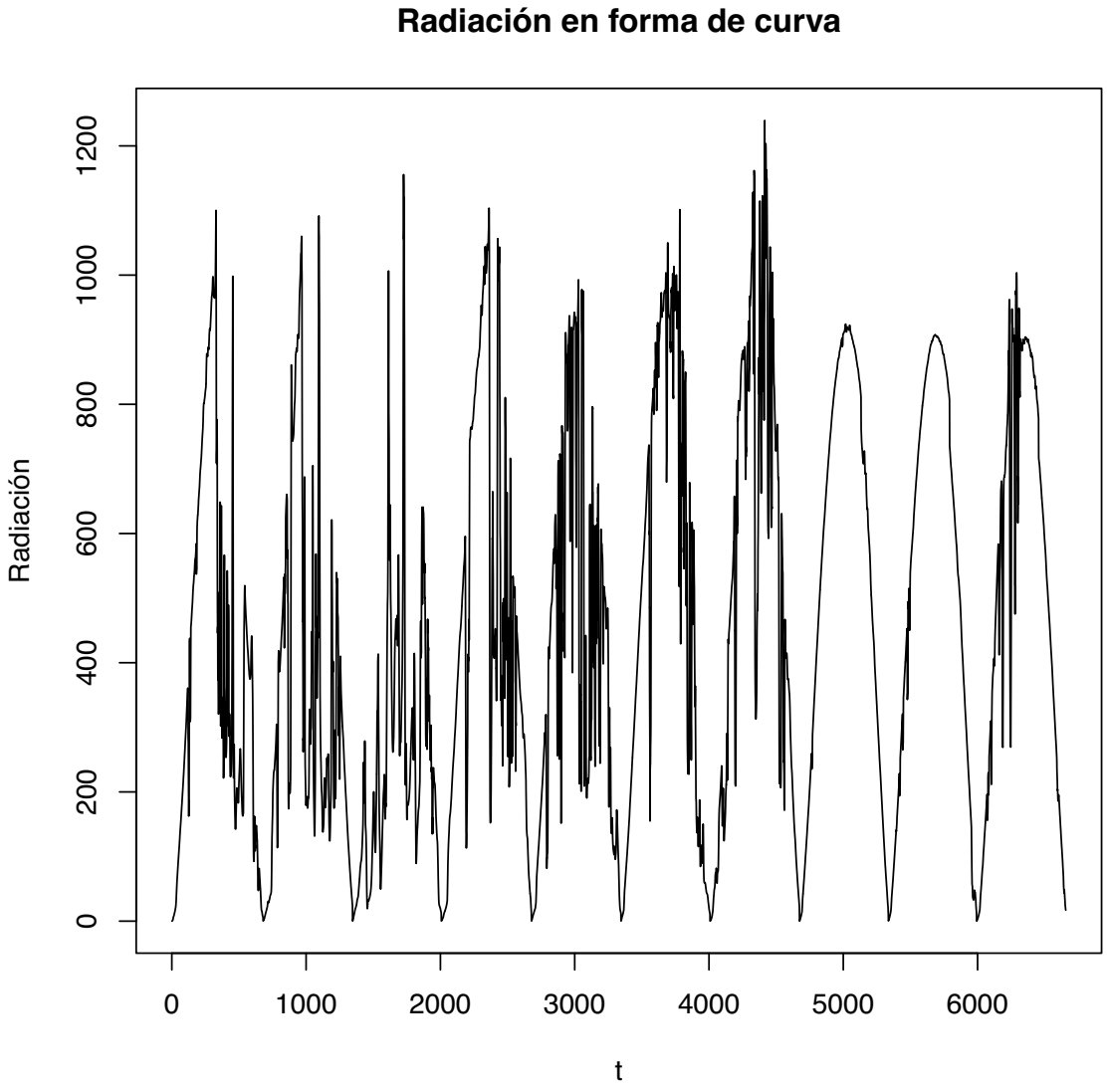


Figura 2.3: Curva de radición en el tiempo.

Los datos solo se miden por el día, en concreto la serie histórica mide datos de 10 días. Podríamos pensar en representar los datos separados por días. Lo cual equivale a hacer zoom sobre los gráficos anteriores.

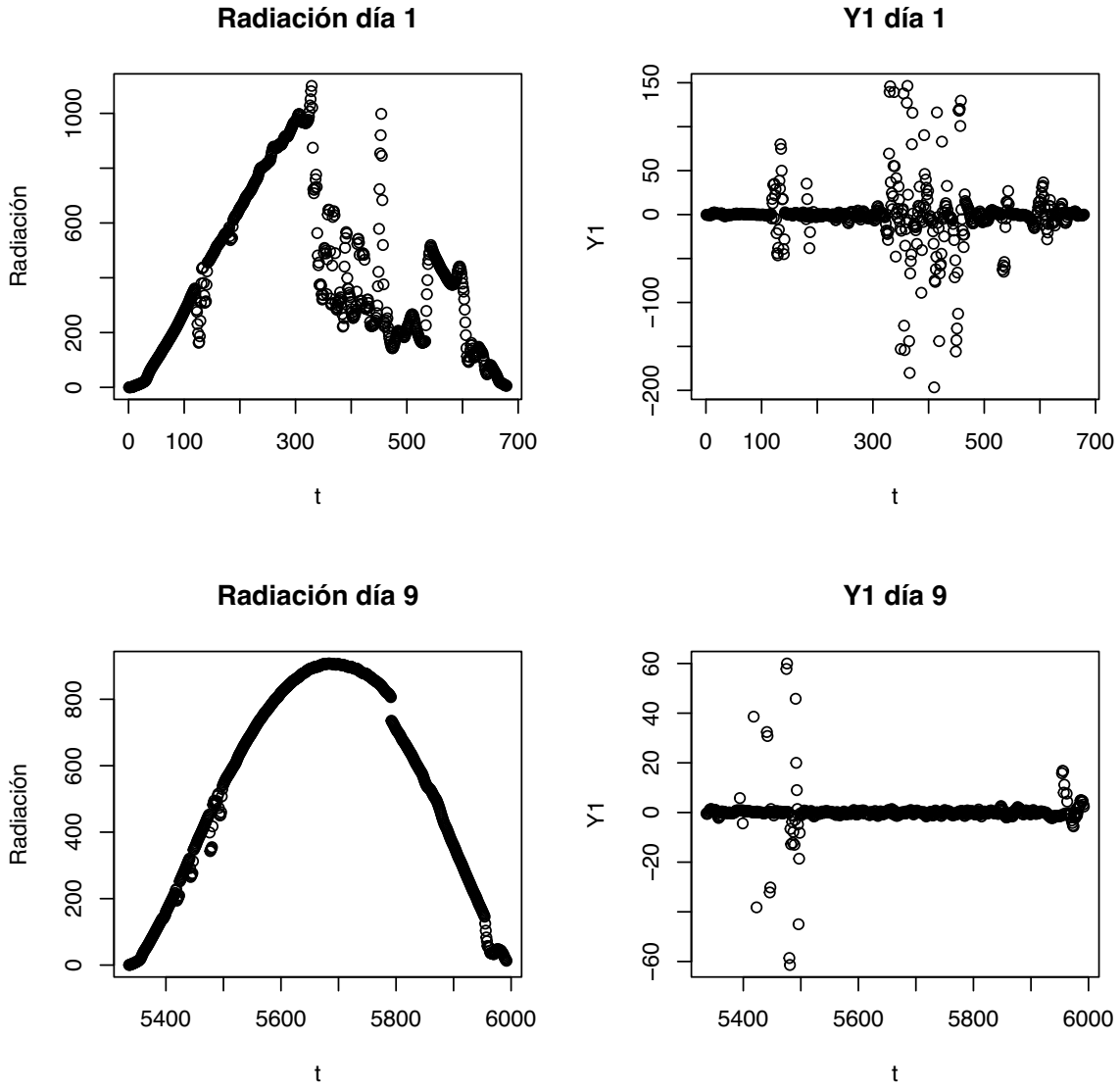


Figura 2.4: Curva de radiación y error Y_1 separados en días.

En la Figura 2.4 vemos que lo que produce los errores no es la cantidad de radiación, sino que es su variación, es decir su derivada. Si denotamos la radiación en el instante t como R_t, podemos aproximar la derivada en el instante t como

$$dR_t = \frac{R_{t+1} - R_t}{(t+1) - t} = R_{t+1} - R_t.$$

Cuando se produce un cambio de día, la radiación en el instante t y la radiación en el instante $t+1$ no son datos contiguos. Por tanto, el valor de la derivada en esos puntos se supondrá igual a su valor en el instante anterior. Se calcula la derivada de la radición en `R` teniendo esto en cuenta y se representan los resultados.

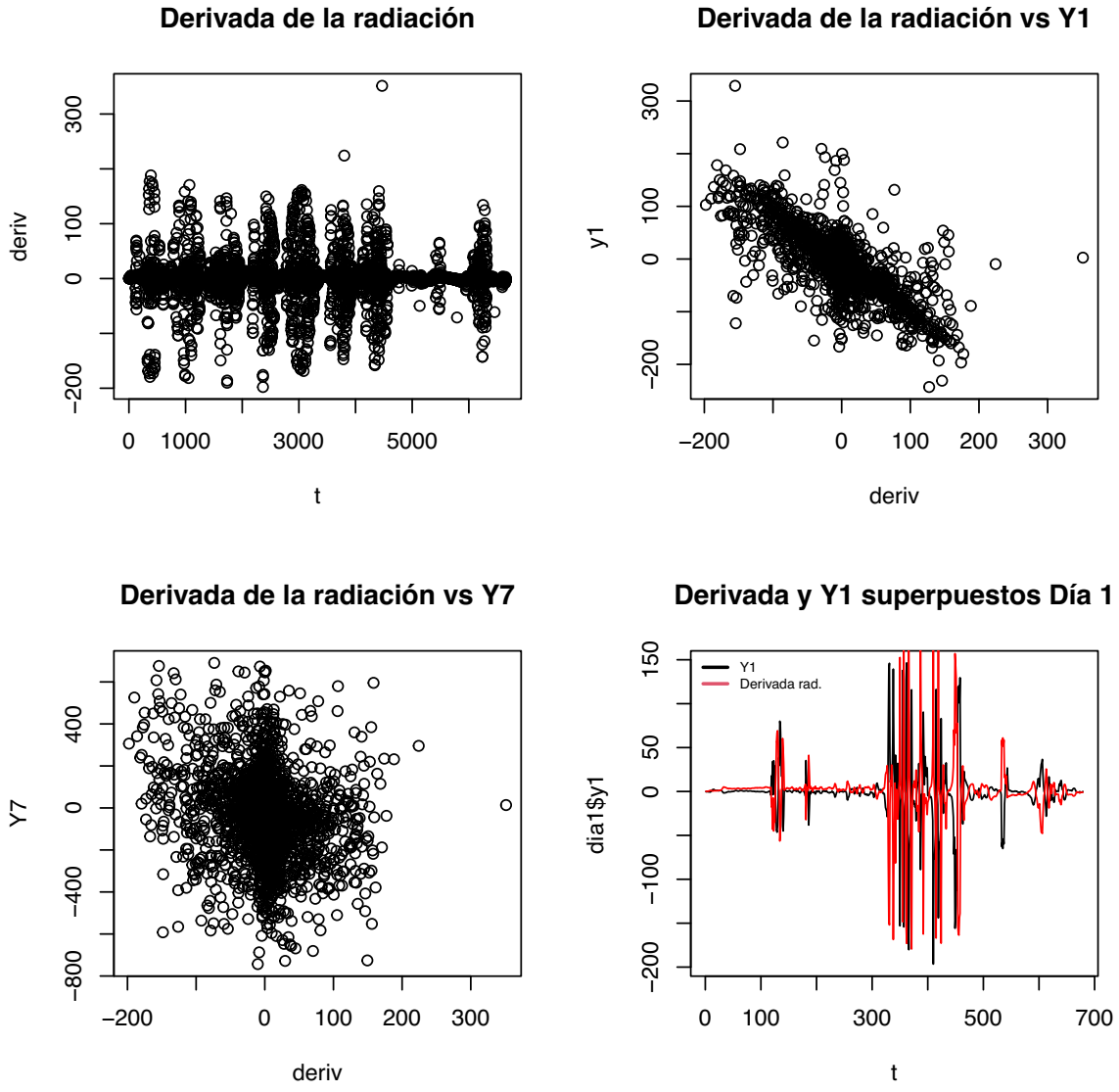


Figura 2.5: Gráficos para ver la relación entre la derivada de la radiación y algunos errores Y_i.

Los resultados reflejados en Figura 2.5 son de lo más interesantes para nuestro proyecto. En el diagrama de dispersión de la deriavada de la radiación frente a los errores Y_1, podemos ver claramente una relación lineal inversa. Lo cual se confirma al superponer la derivada con Y_1, ya que parece ser un reflejo de Y_1. Por otra parte según tomamos errores más avanzados en el futuro, observamos que se pierde la relación lineal y empezamos a tener un gráfico de dispersión sin patrones claros.

Vemos que la derivada de la radiación explica razonablemente bien los errores Y_1, ahora intentaremos razonar el motivo. Recordemos que Y_1 es el error que se comete al estimar la energía eléctrica producida en el instante $t+1$ con los datos disponibles en el instante t. La derivada nos da información sobre la suavidad de la curva de radiación cuando pasa del instante t al instante $t+1$, siendo este último el horizonte de predicción de Y_1. Esto nos sugiere la siguiente conjetura, *la derivada de la radiación en el instante* $t+(i-1)$ *explica razonablemente bien la variable* Y_i. A continuación definimos una función en `R` que desplaza los valores de la derivada, para emparejar la derivada en el instante $t+(i-1)$ con

la variable Y_i en el instante t.

```
## Vector original:  1 2 3 4 5 6
## Vector avanzado 1 puesto:  2 3 4 5 6 0
## Vector avanzado 3 puestos:  4 5 6 0 0 0
```

Una vez definida la función **avanzar()**, representamos algunos errores Y_i frente a la derivada avanzada $i-1$ puestos. Veamos en la Figura 2.6 si gráficamente se refuerza nuestra conjetura.

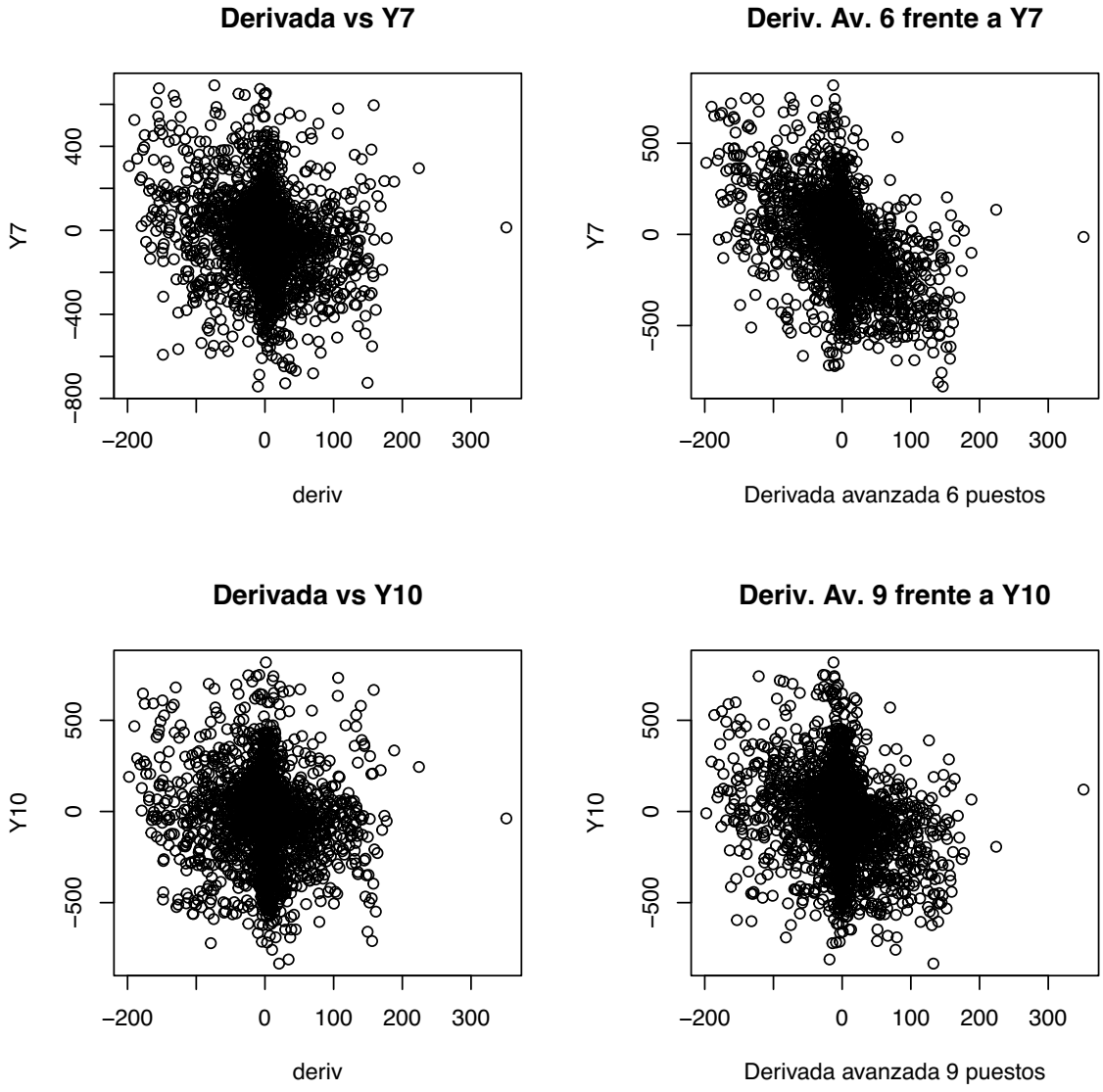


Figura 2.6: Gráficos de dispersión de errores Y_i y la derivada de la radiación avanzada $i-1$ puestos respectivamente.

Lamentablemente, los resultados no son tan buenos como esperábamos. Intentemos ver analíticamente lo que ocurre. Para ello, ajustaremos un modelo GAM utilizadando como única variable

explicativa la derivada avanzada distintos puestos, para ver comó de bien ajustamos los datos en cada caso. Los resultados se muestran en el Cuadro 2.1.

```
#Bucle para explorar que variables explican más variabilidad y ver si refuerza
#nuestra teoría
library(mgcv)
R_2=matrix(NA,16,15)
for (i in 1:15){
  for (k in 0:15){
  gammodelo<-gam(as.formula(paste(paste0("y",i),
                                  "~s(avanzar(deriv,",k,"))")),data=DATA)
  R_2[k+1,i]<-summary(gammodelo)$dev.expl
}}
```

Deviance explained																
Modelo para	$derivav_0$	$derivav_1$	$derivav_2$	$derivav_3$	$derivav_4$	$derivav_5$	$derivav_6$	$derivav_7$	$derivav_8$	$derivav_9$	$derivav_{10}$	$derivav_{11}$	$derivav_{12}$	$derivav_{13}$	$derivav_{14}$	$derivav_{15}$
Y_1	0.5905	0.186	0.0648	0.0204	0.0275	0.1209	0.0399	0.0226	0.0208	0.013	0.0122	0.017	0.015	0.0086	0.0147	0.0131
Y_2	0.5096	0.4955	0.1547	0.0442	0.0169	0.0754	0.1003	0.0365	0.0264	0.0196	0.0153	0.0144	0.016	0.012	0.0126	0.0160
Y_3	0.4049	0.5266	0.4052	0.125	0.0294	0.0273	0.066	0.0718	0.036	0.0255	0.0197	0.0159	0.0156	0.0153	0.015	0.0146
Y_4	0.3088	0.4503	0.4702	0.3252	0.0858	0.0182	0.0297	0.0541	0.0598	0.0294	0.0209	0.0165	0.0143	0.0149	0.0173	0.0161
Y_5	0.2027	0.3607	0.4278	0.3934	0.2363	0.0462	0.0176	0.0282	0.0485	0.0487	0.0254	0.0186	0.0158	0.014	0.0164	0.0170
Y_6	0.106	0.2613	0.3771	0.4032	0.3339	0.1704	0.0364	0.0146	0.0255	0.0416	0.0452	0.0257	0.0204	0.0176	0.0174	0.0177
Y_7	0.0695	0.1573	0.2943	0.3768	0.3606	0.2612	0.1368	0.0301	0.0154	0.026	0.0436	0.0471	0.0289	0.0219	0.0209	0.0185
Y_8	0.0506	0.1064	0.184	0.2961	0.3385	0.29	0.217	0.1191	0.0284	0.0176	0.0297	0.0466	0.0519	0.0319	0.0268	0.0233
Y_9	0.04	0.0809	0.1318	0.1968	0.2795	0.2861	0.2511	0.1948	0.1043	0.0283	0.0217	0.0337	0.0505	0.0526	0.0339	0.0262
Y_{10}	0.0361	0.0644	0.1031	0.1447	0.1926	0.2412	0.255	0.2342	0.1784	0.095	0.0262	0.021	0.035	0.0516	0.0548	0.0348
Y_{11}	0.0314	0.0555	0.0833	0.1155	0.1447	0.1671	0.2177	0.2369	0.2118	0.1578	0.0818	0.0209	0.0197	0.0352	0.0521	0.0527
Y_{12}	0.0252	0.044	0.0671	0.0893	0.1127	0.125	0.1537	0.2066	0.2236	0.1982	0.146	0.0732	0.0165	0.0192	0.0361	0.0503
Y_{13}	0.0205	0.036	0.0553	0.0745	0.0913	0.1027	0.1213	0.1527	0.2025	0.2127	0.1842	0.1317	0.0608	0.0136	0.0194	0.0352
Y_{14}	0.0177	0.03	0.0469	0.0629	0.0781	0.0851	0.1017	0.1219	0.1503	0.1912	0.1952	0.1642	0.1126	0.0516	0.0124	0.0184
Y_{15}	0.0158	0.0264	0.0404	0.0553	0.0696	0.0756	0.0879	0.1061	0.1231	0.1433	0.1762	0.173	0.1404	0.0961	0.0436	0.0118

Cuadro 2.1: Tabla resumen de la varibilidad explicada por los modelos con una sola variable explicativa $derivav_j$ para cada error Y_i.

En el Cuadro 2.1, está el porcentaje de variabilidad explicada para cada modelo GAM, considerando como variable explicativa la derivada original y la derivada avanzada hasta 15 puestos, en orden. Vemos que la derivada avanzada $i-1$ instantes no es la mejor explicativa para Y_i, sin embargo podemos notar que si avanzamos la derivada más de $i-1$ puestos, el porcentaje de variabilidad explicada cae. Por mayor comodidad de aquí en adelante denotaremos a la derivada de la radiación avanzada k puestos como $derivav_k$. En este punto comprendemos que los errores estan relacionados con la variabilidad de la radición en los instantes previos al horizonte de predicción de cada error. Es decir, para el error Y_5 influye el comportamiento de la radiación entre el instante t y el instante $t+i$. Esta información está contenida en la variables $derivav_k$ con $k = 0, \ldots, 4$. Con esto ya tenemos las variables explicativas cantidatas para cada error Y_i. En general para cada Y_i, serán cantidatas a variables explicativas las variables $derivav_j$ con $j \in 1, \ldots, (i-1)$.
La idea ahora es ajustar un modelo GAM a cada error Y_i. Para el error Y_1 por ejemplo, solo disponemos de una variable explicativa, $derivav_0$, que es la derivada sin avanzar ningun puesto. Sin embargo para

Y_3 tenemos como variables explicativas, $derivav_0$, $derivav_1$ y $derivav_2$, con lo cual tenemos 3 modelos con 1 variable explicativa, 3 modelos con 2 variables explicativas y 1 modelo con 3 variables explicativa, lo cual hace un total de 7 modelos posibles. Para el error Y_{15}, el número de posibles modelos con 4 variables explicativas es $\binom{15}{4} = 1365$, número de posibles combinaciones con 15 elementos en grupos de 4. El número de posibles modelos con 3 variables explicativas es $\binom{15}{3} = 455$, para 2 variables explicativas tenemos $\binom{15}{2} = 105$ modelos y finalmente hay 15 combinaciones con un solo elemento. Por tanto para el error Y_{15} si ajustamos todos los modelos que tengan a lo sumo 4 variables explicativas, tenemos que ajustar $1365+455+105+15 = 1940$. En general el número de modelos con m explicativas que se pueden ajustar cuando tenemos n catidatas es, n sobre m, lo cual se dispára rapidamenta, por ello, solo comprabaremos modelos con un máximo de 4 variables explicativas. El código utilizado para la validación cruzada se puede consultar en el Apéndice A.
Tras horas de ejecución el algoritmo devuelve los siguientes modelos como los mejores que ha encontrado, tomando como medida de error R^2. En la Cuadro 2.2 resumimos los resultados obtenidos.

Variable respuesta	V. Explicativa 1	V. Explicativa 2	V. Explicativa 3	V. Explicativa 4	R^2
Y_1	$derivav_0$	–	–	–	0.5941
Y_2	$derivav_0$	$derivav_1$	–	–	0.5247
Y_3	$derivav_0$	$derivav_1$	$derivav_2$	–	0.5683
Y_4	$derivav_0$	$derivav_1$	$derivav_2$	$derivav_3$	0.5863
Y_5	$derivav_0$	$derivav_1$	$derivav_3$	$derivav_4$	0.5117
Y_6	$derivav_0$	$derivav_2$	$derivav_4$	$derivav_5$	0.5231
Y_7	$derivav_1$	$derivav_3$	$derivav_4$	$derivav_5$	0.5047
Y_8	$derivav_1$	$derivav_4$	$derivav_6$	–	0.4669
Y_9	$derivav_1$	$derivav_4$	$derivav_6$	–	0.4421
Y_{10}	$derivav_2$	$derivav_4$	$derivav_7$	$derivav_8$	0.4459
Y_{11}	$derivav_3$	$derivav_6$	$derivav_8$	$derivav_9$	0.4165
Y_{12}	$derivav_4$	$derivav_7$	$derivav_9$	$derivav_{11}$	0.4145
Y_{13}	$derivav_5$	$derivav_8$	$derivav_{10}$	$derivav_{12}$	0.3674
Y_{14}	$derivav_6$	$derivav_8$	$derivav_{11}$	$derivav_{13}$	0.3574
Y_{15}	$derivav_7$	$derivav_{10}$	$derivav_{12}$	$derivav_{14}$	0.3224

Cuadro 2.2: Resumen de los mejores modelos encontrados y del R^2 estimado por validación cruzada.

2.3. Ajuste del modelo

Ahora ajustaremos los modelos obtenidos en la sección anterior para cada variable Y_i. Para ello utilizaremos las función **gam()** del paquete **mgcv**.

```
#library(mgcv)
modelo=list(NULL)
modelo[[1]]=gam(y1~s(derivav0,k=30),data=DATA)
modelo[[2]]=gam(y2~s(derivav0,k=30)+s(derivav1,k=30),data=DATA)
modelo[[3]]=gam(y3~s(derivav0,k=30)+s(derivav1,k=30)+
                  s(derivav2,k=30),data=DATA)
modelo[[4]]=gam(y4~s(derivav0,k=30)+s(derivav1,k=30)+
                  s(derivav2,k=30)+s(derivav3,k=30),data=DATA)
modelo[[5]]=gam(y5~s(derivav0,k=30)+s(derivav1,k=30)+
                  s(derivav3,k=30)+s(derivav4,k=30),data=DATA)
modelo[[6]]=gam(y6~s(derivav0,k=30)+s(derivav2,k=30)+
                  s(derivav4,k=30)+s(derivav5,k=30),data=DATA)
modelo[[7]]=gam(y7~s(derivav1,k=30)+s(derivav3,k=30)+
                  s(derivav4,k=30)+s(derivav5,k=30),data=DATA)
modelo[[8]]=gam(y8~s(derivav1,k=30)+s(derivav4,k=30)+
                  s(derivav6,k=30),data=DATA)
modelo[[9]]=gam(y9~s(derivav1,k=30)+s(derivav4,k=30)+
                  s(derivav6,k=30),data=DATA)
modelo[[10]]=gam(y10~s(derivav2,k=30)+s(derivav4,k=30)+
                   s(derivav7,k=30)+s(derivav8,k=30),data=DATA)
modelo[[11]]=gam(y11~s(derivav3,k=30)+s(derivav6,k=30)+
                   s(derivav8,k=30)+s(derivav9,k=30),data=DATA)
modelo[[12]]=gam(y12~s(derivav4,k=30)+s(derivav7,k=30)+
                   s(derivav9,k=30)+s(derivav11,k=30),data=DATA)
modelo[[13]]=gam(y13~s(derivav5,k=30)+s(derivav8,k=30)+
                   s(derivav10,k=30)+s(derivav12,k=30),data=DATA)
modelo[[14]]=gam(y14~s(derivav6,k=30)+s(derivav8,k=30)+
                   s(derivav11,k=30)+s(derivav13,k=30),data=DATA)
modelo[[15]]=gam(y15~s(derivav7,k=30)+s(derivav10,k=30)+
                   s(derivav12,k=30)+s(derivav14,k=30),data=DATA)
```

En el Cuadro 2.3 se presenta como medida de bondad de ajuste la cantidad de variabilidad explicada, o R^2. Esta medida de bondad de ajuste es la que calcula por defecto da función **gam()**.

Modelo	Deviance Explained o R^2
Modelo Y_1	0.6014
Modelo Y_2	0.6264
Modelo Y_3	0.6493
Modelo Y_4	0.6555
Modelo Y_5	0.6029
Modelo Y_6	0.5748
Modelo Y_7	0.5514
Modelo Y_8	0.523
Modelo Y_9	0.4904
Modelo Y_{10}	0.5076
Modelo Y_{11}	0.4903
Modelo Y_{12}	0.4654
Modelo Y_{13}	0.4344
Modelo Y_{14}	0.3981
Modelo Y_{15}	0.3801

Cuadro 2.3: Tabla resumen de la bondad de ajuste de los modelos entrenados con la totalidad de los datos.

Vemos que los primeros modelos son buenos. Un valor de R^2 entorno a 0,6 se puede considerar bueno. Los últimos modelos son mucho más difíciles de ajustar, aun así obtenemos valores de bondad de ajuste entorno a 0,4 lo cual es aceptable. A continuación representaremos gráficamente los resultados obtenidos, con el fin de que el lector tenga un apoyo visual sobre la bondad de ajuste de los modelos. Para no representar todos los gráficos ya que son similares, representaremos los resultados de los modelo para Y_1, Y_5, Y_{10} e Y_{15}. Primero mostraremos los valores ajustados por el modelo frente a los valores reales (Figura 2.7), lo ideal en este tipo de gráficos es que los puntos se ajusten a una recta de pendiente 1 e intercepto 0.

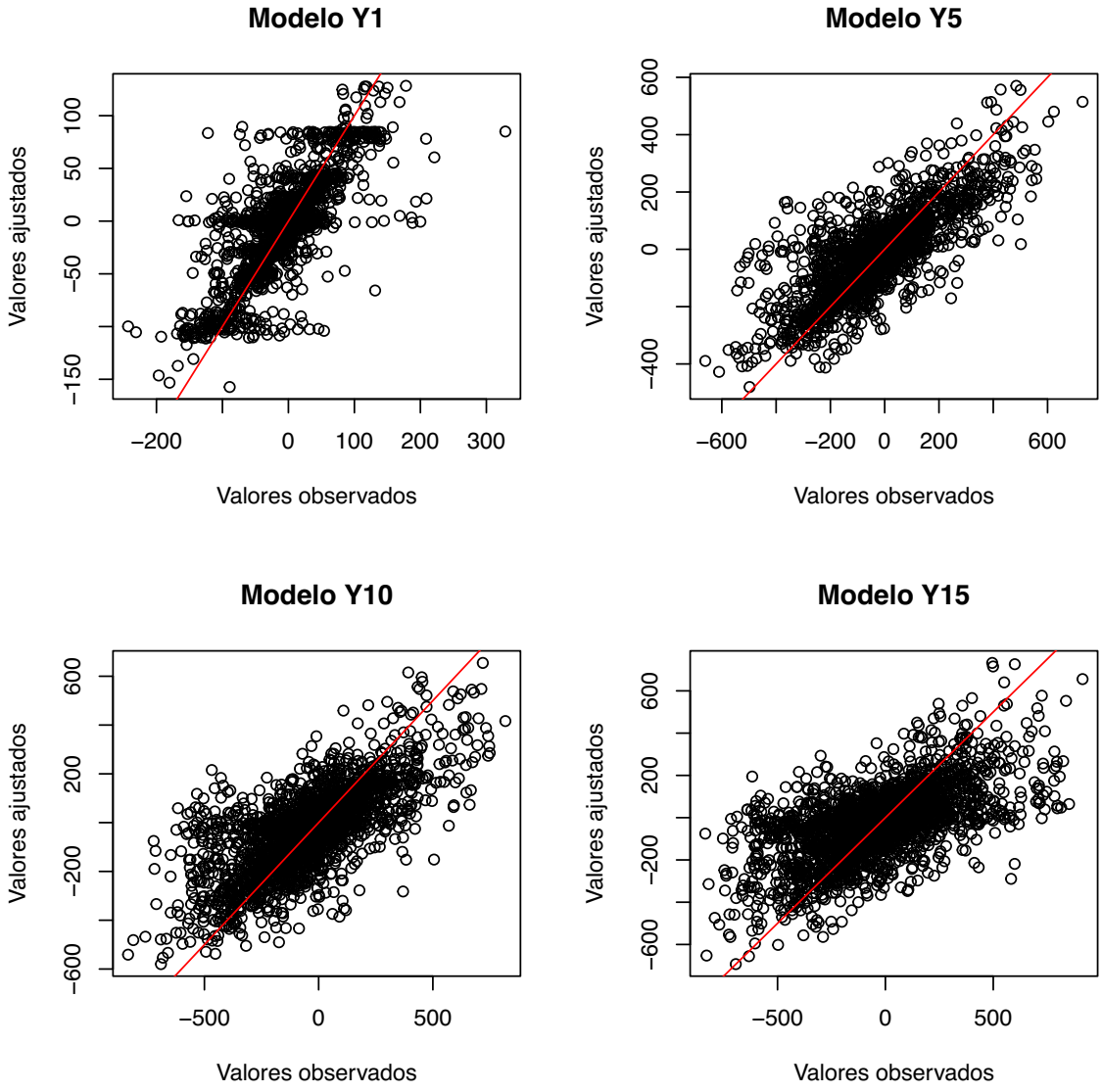


Figura 2.7: Gráficos de los valores observados frente a los valores ajustados.

Tal y como nos indicaban las medidas analítica, los modelos para Y_1 e Y_5, se ajustan mejor a la recta que los modelos Y_{10} e Y_{15}. Con todo debemos insistir en que respecto a otros modelos que se probaron con anterioridad, estos resultados son muy buenos. Para que se vea con mayor claridad como ajustan los datos estos modelos, vamos a representar los datos observados superpuestos con los datos ajustados respecto al tiempo, en la Figura 2.8. Así, a su vez, veremos que lo que creíamos que era una relación temporal entre los errores, se puede explicar mediante la derivada de la radiación.

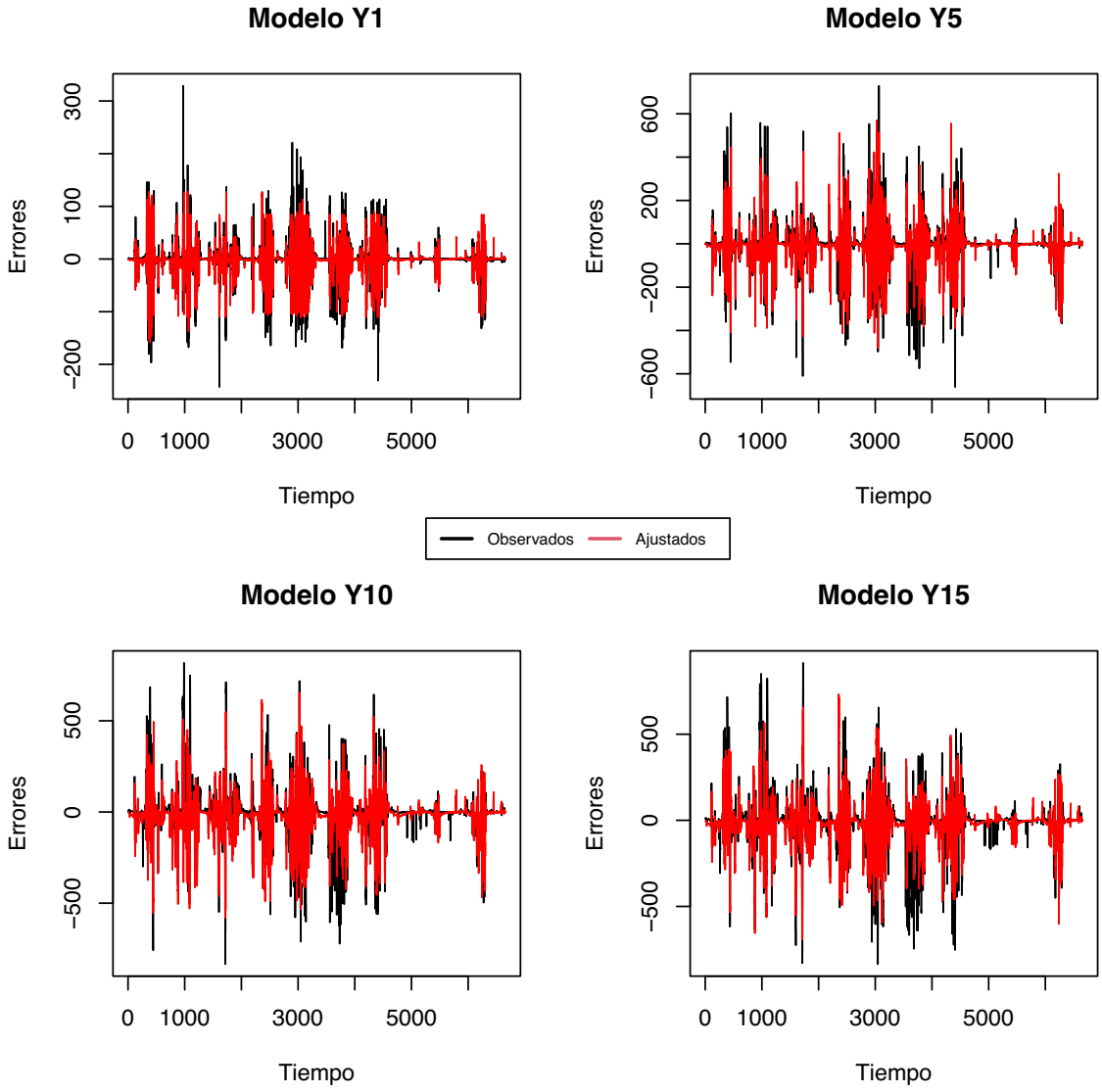


Figura 2.8: Gráficos de los valores observados superpuestos con los valores ajustados a lo largo del tiempo.

En la Figura 2.8 vemos que el modelo capta muy bien las zonas en las que el error es practicamente nulo y las zonas en las que este tiene mayor variabilidad, en lineas generales consigue solapar bien los datos originales. Podemos concluir que estamos satisfechos con los modelos que hemos encontrado para ajustar los datos. Además como la selección de variables se ha hecho por validación cruzada sabemos que la bondad de ajuste se mantiene estable cuando se trata de predecir nuevos datos, al menos para aquellas zonas que pertenecen a la muestra total.
Como consideración a mayores en este apartado, comentar que otra opción interesante para la selección de variables sería el método *k-fold*, que es un caso particular de la validación cruzada. En *k-fold* en vez de sortear los datos cada vez que ajustamos un modelo, lo que se hace es dividir los datos en k carpetas de forma aleatoria. Luego se utilizan $k-1$ carpetas para entrenar el modelo y 1 carpeta para calcular el error cometido al predecir nuevos datos. Este método, presenta la ventaja de que nos asegura que

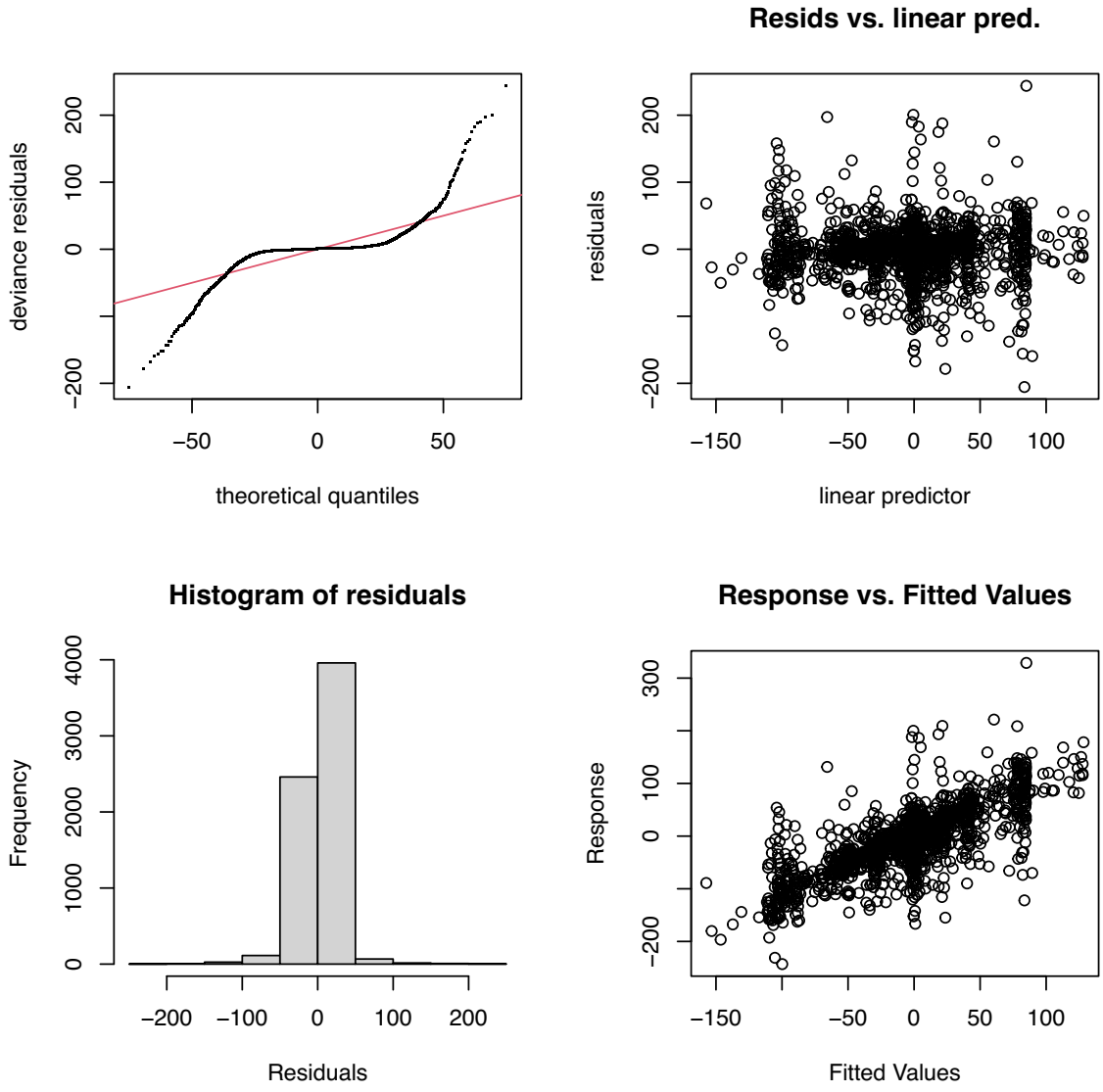
todos los datos pasan por la muestra de entrenamiento y por la muestra de test, además por norma general es un método que tiene menos sesgo a la hora de estimar la bondad de ajuste. *K-fold* se utiliza cuando el tamaño de la muestra que tenemos disponible es pequeño. En nuestro caso para cada error Y_i tenemos 6657 observaciones por lo tanto el método de validación cruzada utilizado debería producir resultados aceptables. Para más información consultar Refaeilzadeh *et al.* (2008).

2.4. Validación

Una vez que hemos ajustado los modelo, debemos validar las hipótesis de los mismos. Recordemos que los modelos GAM eran de la forma

$$Y = \mathcal{B}_0 + f_1(X_1) + f_2(X_2) + \cdots + f_p(X_p) + \epsilon.$$

Donde f_i, $i = 1, \ldots, p$ son funciones suaves cualesquiera, $\mathcal{B}_0$ es el intercepto y ϵ son los errores que se suponen normales e independientes entre sí. Las hipótesis de normalidad e independecia en estos modelos, son importates para los resultados asintóticos, por ejemplo cuando hacemos un contraste de hipótesis se supone que el estadístico de contraste tiene una determinada distribución asintótica, lo cual en caso de que los errores no sean normales e independientes, puede no ser cierto. Sin embargo de cara a la predicción, la violación de estas hipótesis es algo asumible. Cuando trabajamos con datos reales, es realmente complicado que los errores sean totalmente normales e independiente. Por otra parte, si alguna de las variables explicativas no aporta información relevante sobre la respuesta puede aparecer un problema de concurvidad, el análogo a la colinealidad en regresión lineal. El problema de la concurvidad es bastante grave, ya que puede hacer que las predicciones de nuestro modelo no sean consistentes. La concurvidad toma valores entre 0 y 1, valores cercanos a 1 pueden representar un problema.
Para validar las hipótesis de los modelos utilizaremos las funciones **gam.check()** y **concurvity()**. A mayores representaremos como es el efecto de cada una de las explicativas sobre los modelos, lo cual es una de las ventajas de los modelos GAM, que pese a ser más flexibles que los modelos lineales mantienen su interpretabilidad. Al igual que anteriormente, en este documento no veremos la validación de todos los modelos, ya que los resultado son muy similares entre ellos, nos centraremos en exponer los casos extremos, los modelos para Y_1 e Y_{15}.

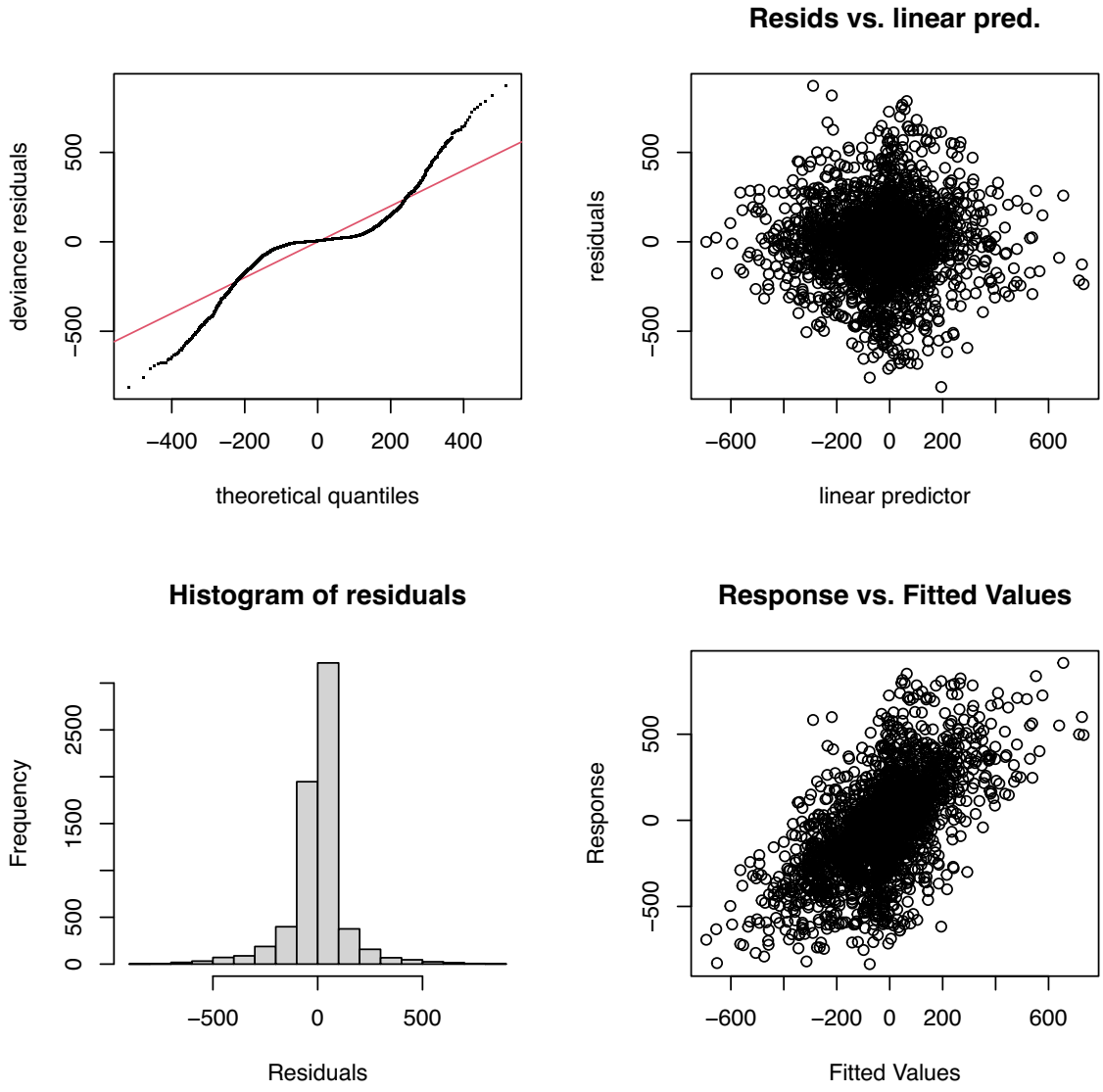



```
##
## Method: GCV   Optimizer: magic
## Smoothing parameter selection converged after 6 iterations.
## The RMS GCV score gradient at convergence was 0.0001473784 .
## The Hessian was positive definite.
## Model rank =  30 / 30
##
## Basis dimension (k) checking results. Low p-value (k-index<1) may
## indicate that k is too low, especially if edf is close to k'.
##
##               k'  edf k-index p-value
## s(derivav0) 29.0 27.3    0.97    0.02 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##                 para  s(derivav0)
## worst    3.60914e-14 3.609140e-14
## observed 3.60914e-14 2.908329e-18
## estimate 3.60914e-14 2.028051e-17
```

Figura 2.9: Gráficos para la validación de modelo para Y_1 generados por la función **gam.check()** y valores de la concurvidad.

En los gráficos de la izquierda nos gustaría observar normalidad. En el gráfico superior derecho, nos interesa que no se observen patrones. Finalmente, en gráfico restante, debemos ver que los puntos se ajusten a una recta. En este caso vemos que no hay normalidad en los residuos, en gran medida se debe a la alta densidad que se observa entorno a 0. Respecto al gráfico superior derecho al menos no se aprecia un efecto heterocedástico respecto al predictor. Finalmente los valores predichos frente a los ajustados si que se ajustan razonablemente bien a una recta. Como comentábamos antes, no se cumplen todas las hipótesis sobre los residuos del modelo. Sin embargo cuando trabajamos con datos reales, debemos ser un poco flexibles, siendo siempre conscientes de las limitaciones que conlleva la violación de ciertas hipótesis. En este caso, nos obliga a descartar el test ANOVA como posible método de selección de variables, lo cual nos condujo a la alternativa de la validación cruzada. En lo que respecta a la concurvidad en este caso solo tenemos una variable explicativa, por lo cual no tendríamos ningún problema. Ahora mostraremos los gráficos de validación y los niveles de concurvidad del modelo para Y_{15}.




```
##
## Method: GCV   Optimizer: magic
## Smoothing parameter selection converged after 11 iterations.
## The RMS GCV score gradient at convergence was 0.01210503 .
## The Hessian was positive definite.
## Model rank =  117 / 117
##
## Basis dimension (k) checking results. Low p-value (k-index<1) may
## indicate that k is too low, especially if edf is close to k'.
##
##                 k'  edf k-index p-value
## s(derivav7)  29.0 25.2    0.97   0.035 *
## s(derivav10) 29.0 23.0    1.01   0.745
## s(derivav12) 29.0 23.9    0.99   0.230
```

```
## s(derivav14) 29.0 25.5    0.98   0.155
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##                  para s(derivav7) s(derivav10) s(derivav12) s(derivav14)
## worst    1.326141e-13   0.4836014    0.6159231    0.6500880    0.5570637
## observed 1.326141e-13   0.2181953    0.2868611    0.4454547    0.1740980
## estimate 1.326141e-13   0.2055885    0.2552063    0.3532159    0.1679574
```

Figura 2.10: Gráficos para la validación de modelo para Y_{15} generados por la función **gam.check()** y valores de la concurvidad.

Observado la Figura 2.10 podemos sacar las mismas conclusiones que las mencionadas anteriormente para la Figura 2.9. En general todos los modelos presentan los mismos problemas que el modelo para Y_1, la única diferencias es que cuanto mayor es el índice i menos se ajusta el cuarto gráfico a una recta, siendo el modelo para Y_{15} el caso más extremo. Respecto a la concurvidad los valores son asumibles, por lo tanto no debemos preocuparos. Finalmente mencionar que para ajustar los modelos estamos utilizando una base de *BSplines* de tamaño 30, esto no afecta mucho a los resultas mientras que si lo hace en el tiempo de computación, pero un tamaño de base mayor implica una mayor flexibilidad en las funciones que forman el modelo. Finalmente, veamos el efecto de las variables explicativas sobre la variable respuesta.

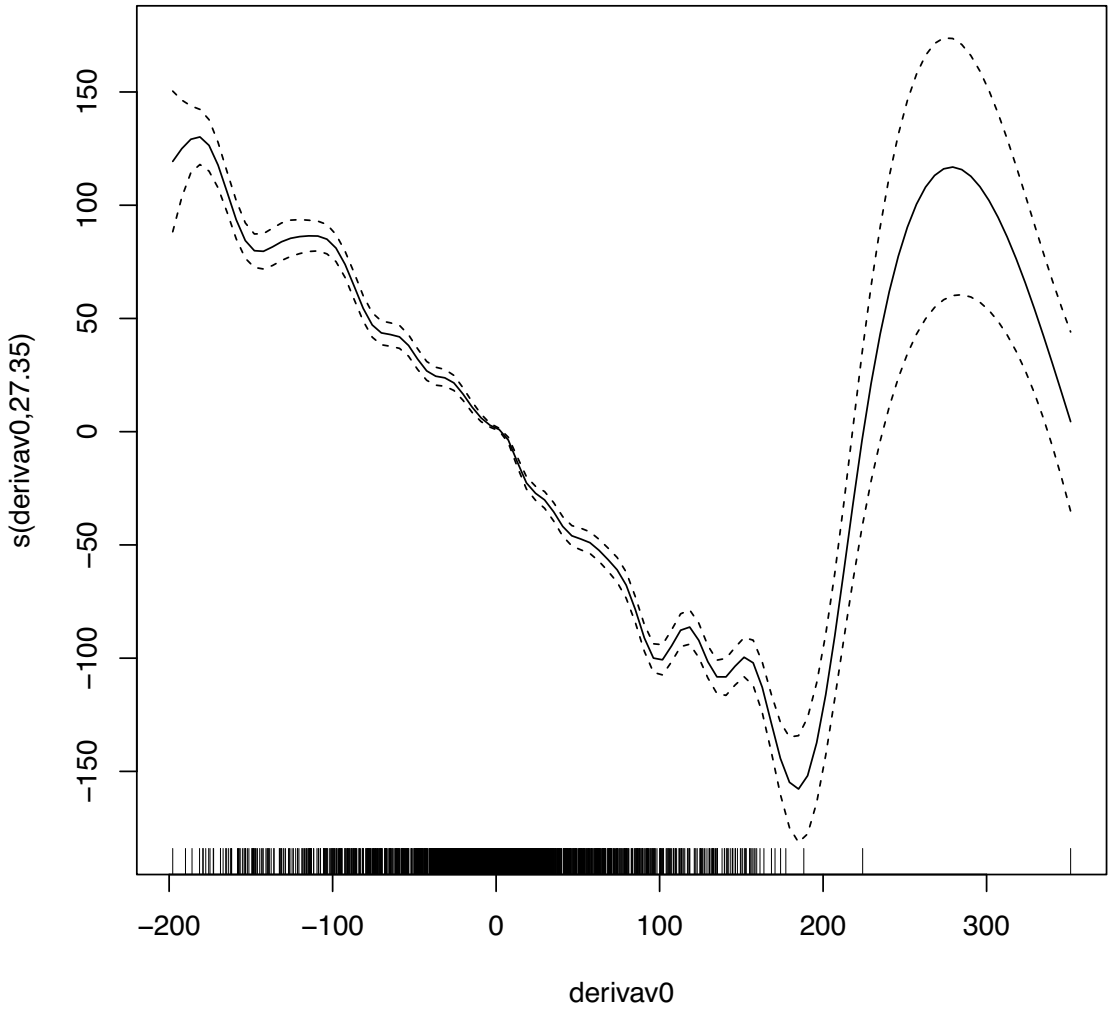


Figura 2.11: Efecto de la variable explicativa $derivav_0$ en el modelo para Y_1.

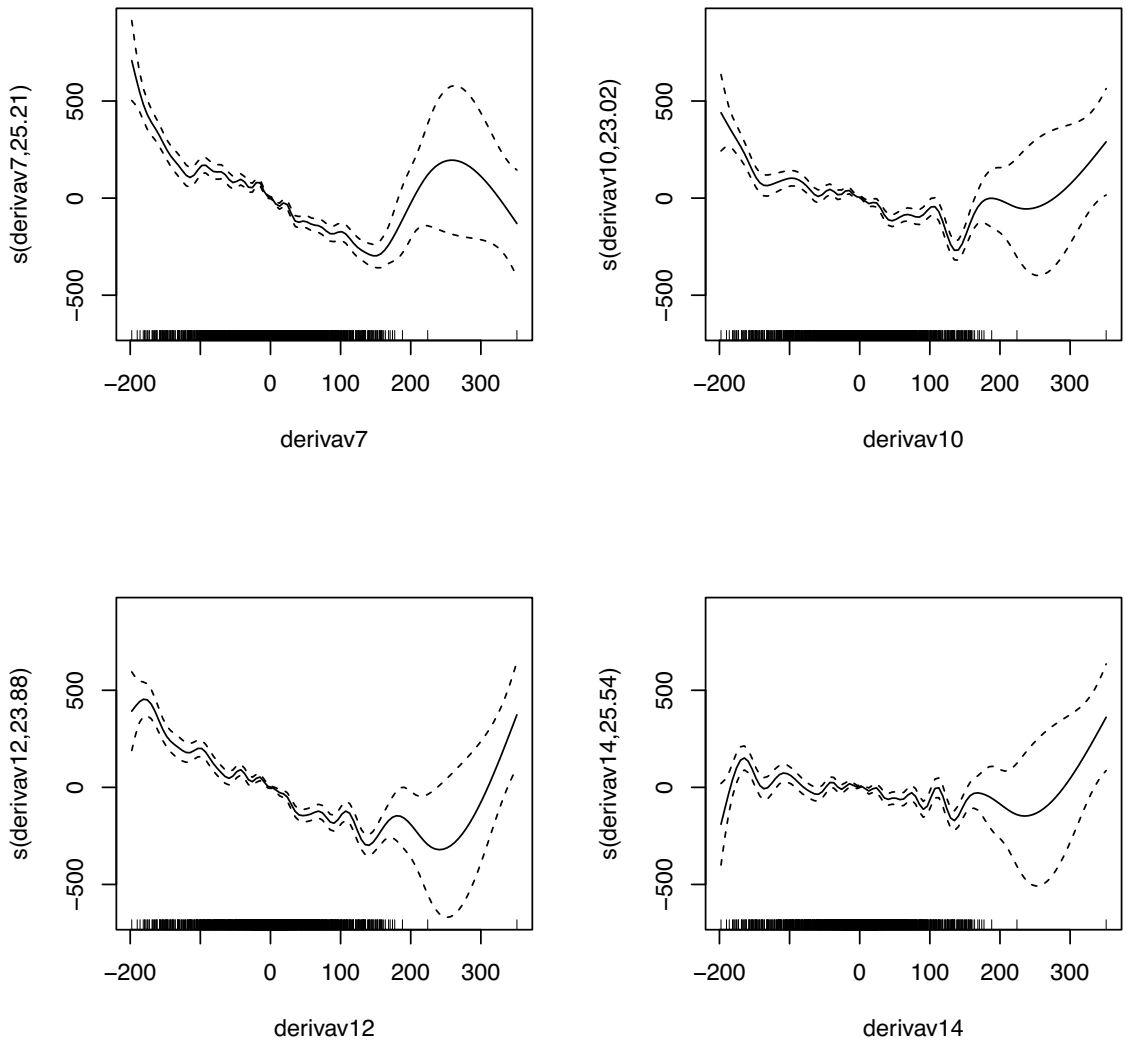


Figura 2.12: Efectos de las variables explicativas en el modelo para Y_{15}.

Sobre el efecto de las variables explicativas debemos notar que la derivada tiene un dato atípico próximo a 350, al no tener más datos cercanos a ese valor existe la posibilidad de que el modelo este sobreajustando nos datos, perdinde efectividad para errores donde las variables explicativas se sitúen entre 200 y 350. Para evitar un posible problema en esa zona lo ideal sería tomar más mediciones para intentar muestrear datos que tenga valores de las variables explicativas entre 200 y 350. En regresión lineal sería obligatorio extraer los datos atípicos del conjunto de datos, pero en los modelos GAM vemos que la influencia de los datos atípicos no se extiende al efecto de las variables explicativas en las zonas de más densidad. Por ello nosotros mantendremos el modelo sin variación.
Finalmente comentar que nos podríamos plantear utilizar la regresión lineal, nosotros nos decantamos por los modelos GAM porque los resultados son sensiblemente mejores llegando a explicar un 10 % más de variabilidad de los datos sin perder interpretabilidad.

Capítulo 3

Conclusión

Tras 5 meses de práctica con **TSK** diría que mi experiencia ha sido satisfactoria. Al principio el proyecto parecía muy sencillo, pero debido a las características de los datos resultó una odisea planater una solución. Lo cual supuso muchas horas de trabajo que finalmente no se ven reflejadas en el resultado final. Esta, ha sido mi primera experiencia enfrentandome a un problema para el cual no tenía predefinidos que métodos que, podía o no, utilizar. Me explico, en el ámbito académico cada problema va asociado a una matería, por lo que sabemos que la solución se alcanza mediante los conocimientos dado en esa asignatura en concreto. En este caso es diferente. Me enfrenté a un problema con una cantidad casi ilimitada de herramientas, lo cual creo que forma parte de la dificultad de afrontar problemas reales.
Ciertamente considero que la solución a la que he llegado dista un poco de la solución óptima. A nivel personal me molesta que el modelo no cumpla las hipótesis al pie de la letra, como los típicos modelos de juguete que se utilizan para enseñar los distintos métodos a nivel académico. Con todo, pienso que los modelos GAM obtenidos mejoran notablemente los planteamientos de:

- Introducir los errores tal cual al simulador.
- Utilizar la media de las observaciones como predicción de los errores.

Además pese a ser modelos más complejos que los lineales, también producen mejores resultados. Finalmente comentar que con más tiempo me hubise gustado programar un algoritmo de validación cruzada que implementase el método *k-fold*, ya que creo que esto podría mejorar la información que tenemos sobre la bondad de ajuste de los modelos y consecuentemente mejorar los resultados obtenidos. Respecto a mi desempeño, tras la experiencia adquirida estos meses, hubiese enfocado el proyecto de otra manera. Una vez descartado el enfoque inicial considero que "me lance" a buscar nuevos métodos y alternativas, antes de detenerme a estudiar en profundidad los datos que tenía a mi disposición. Otro fallo que noté, fue la búsqueda obsesiva por un método perfecto, llegando a plantearme soluciones demasiado rebuscadas. En definitiva, una experiencia enriquecedora de la cual creo que he aprendido cosa positivas para el futuro.

Apéndice A

Algoritmo de selección de variables

A continuación se muestra el código de R utilizado para escoger las variables explicativas utilizando validación cruzada.

```
#Número de observaciones del conjuto de datos
nobs=nrow(DATA)
#Selecciono una semilla para poder reproducir los resultado
set.seed(0509)
#Inicializo variables que utilizaré en el bucle
indcomb=NULL
best=NULL
cond2=rep(0,15)
comb=NULL
bestcomb=NULL
#
for (m in 1:4){
  for (d in 1:15){
    if (d==1 & m==1){combinaciones<- combn(0,m,simplify = FALSE)}
    if (d==2 & m<=2){combinaciones<- combn(0:1,m,simplify = FALSE)}
    if (d==3 & m<=3){combinaciones<- combn(0:2,m,simplify = FALSE)}
    if (d>3){combinaciones<- combn(0:(d-1),m,simplify = FALSE)}
 cond=-100 #Inicializo la variable cond con un valor suficientemente bajo para
 #que el primer valor de R2 que obtenga sea mejor.

    for (i in 1:length(combinaciones)){#Este bucle recorre todas las
      #combinaciones de m elementos para cada variable respuesta Y_d

      formula=as.formula(paste(paste0("y",d),"~",paste0("+s(derivav",
                combinaciones[[i]],",k=30)",collapse = "")))
      R2<-NULL
      #El siguiente bucle es el de la validación cruzada, entrenamos un modelo
      #utilizando un 80% de los datos seleccionados alazar. Luego con el 20%
      #restante hacemos predicción y calculamos el error cometido. Hacemos este
      #procedimiento 10 veces y obtenemos una media del error cometido en la
      #predicción.
      for (j in 1:10){
          ind<-sample(nobs,0.8*nobs)
          train<-DATA[ind,c((d+2),19:33)]
```

```
          test<-DATA[-ind,c((d+2),19:33)]
          modelo=gam(formula,data=train)
          pred=NULL
          nw<-data.frame(test)
          pred=predict(modelo,newdata=nw)
          R2[j]= 1 - sum((test[,1] - pred)^2)/sum((test[,1] - mean(test[,1]))^2)
      }
      #Condición que guarda el índice de la combinación de variables que
      #obtiene el mejor resultado.
      if (mean(R2)>cond){indcomb=i
      cond=mean(R2)}
    }
 #Mejor R2 para cada uno de los error Y_d considerando un modelo
 #con m variables explicativas.
 best[d]=cond
 #Combinación de m variables explicativas que obtuvo el mejor resultado.
 comb[[d]]=combinaciones[[indcomb]]
  }
 #Bucle que selecciona el mejor resultado de entre los mejores
  #resultados para cada valor de m
  for (d in 1:15){
  if (best[d]>cond2[d]){bestcomb[[d]]=comb[[d]]
  cond2[d]=best[d]
  }}}
dput(bestcomb, "bestcomb.txt")
dput(cond2, "bestR2.txt")
```

Al terminar el algoritmo guarda en documentos **.txt** la mejor combinación de variables explicativas que ha encontrado y el promedio de los R^2 que se obtuvieron mediante validación cruzada para cada uno de los 15 modelos. El tiempo de ejecución es superior a las 5 horas utilizando un ordenador con procesador *11th Gen Intel(R) Core(TM) i5-1135G7 2.40GHz*

Bibliografía

[1] Ahlberg J. H., Nielson E. N., Walsh J. L. (1967). The theory of Splines and Their Applications. New York: Academic.

[2] Bartels, R. H., Beatty, J. C., Barsky, B. A. (1987). An Introduction to Splines for Use in computer Graphics and Geometric Modeling. Morgan Kaufmann Publishers.

[3] Costas-Bouzas J., Fernández-Casal R., Oviedo-de la Fuente M. (2021). Aprendizaje Estadístico. https://rubenfcasal.github.io/aprendizaje_estadistico

[4] Davis P. (1997). B-splines and Geometric desing. SIAM News. vol. 29. no. 5.

[5] Hastie T., Tibshirani R. (1990). Generalized Additive Models. Chapman & Hall.

[6] Lachmann J., Tsay R. S., Wood D. (2022). MTS: All-Purpose Toolkit for Analyzing Multivariate Time Series (MTS) and Estimating Multivariate Volatility Models. https://cran.r-project.org/web/packages/MTS/index.html

[7] Refaeilzadeh P., Tang L., Lui H. (2008). K-fold Cross-Validation. Arizona State University.

[8] Reinsel, G. C. (2003). Elements of multivariante time series analysis. Springer Science & Business Media.

[9] Tsay, R. S. (2013). Multivariate time series analysis: with R and financial applications. John Wiley & Sons.

[10] Wei W. (2006). Time Series Analysis Univariate and Multivariate Methods. 2nd edition. Pearson College Div.

[11] Wood, S. (2006). mgcv: Mixed GAM Computation Vehicle with Automatic Smoothness Estimation. http://www.cran.r-project.org/package=NPCirc

[12] Wood S. N. (2017). Generalized Additive Models: An Introduction with R, Second Edition. Chapman & Hall/CRC.