Trabajo Fin de Máster

Modelos de predicción para indicadores medioambientales multidimensionales

Jaime Jovanny Castillo Contreras

Máster en Técnicas Estadísticas

Curso 2022-2023

Propuesta de Trabajo Fin de Máster

Título en galego: Modelos de predicción para indicadores medioambientais multidimensionais
Título en español: Modelos de predicción para indicadores medioambientales multidimensionales
English title: Prediction models for multidimensional environmental indicators
Modalidad: Modalidad A
Autor/a: Jaime Jovanny Castillo Contreras, Universidad de Santiago de Compostela
Director/a: Wenceslao González Manteiga, Universidad de Santiago de Compostela
Breve resumen del trabajo: Con esta propuesta de trabajo de fin de máster se pretende abordar los siguientes objetivos: a) Una revisión general actualizada de los modelos de predicción en media de la inmisión del nivel de SO2 y NOx en el entorno de la central térmica de As Pontes. b) Una extensión a la modelización mencionada a través de la predicción con la regresión cuantil. c) Una predicción simultanea de más de un indicador ambiental a través de los conceptos generalizados de cuantil en más de una dimensión. d) Una ilustración comparativa de los distintos procedimientos con datos reales procedentes de la empresa o simulados.
Recomendaciones: se recomienda haber cursado cursos del máster en los tópicos “Modelos de Regresión”, “Modelos de Datos Funcionales”, “Estadística Matemática”, “Series de Tiempo” y el “Análisis Multivariante”.

Don Wenceslao González Manteiga, Catedrático de la Universidad de Santiago de Compostela, informa que el Trabajo Fin de Máster titulado

Modelos de predicción para indicadores medioambientales multidimensionales

fue realizado bajo su dirección por don Jaime Jovanny Castillo Contreras para el Máster en Técnicas Estadísticas. Estimando que el trabajo está terminado, da su conformidad para su presentación y defensa ante un tribunal.

En Santiago de Compostela, a xx de junio de 2022.

El director:
Don Wenceslao González Manteiga

GONZALEZ MANTEIGA WENCESLAO - 32618861P
Firmado digitalmente por GONZALEZ MANTEIGA WENCESLAO - 32618861P
Fecha: 2023.07.13 23:27:21 +02'00'

El autor:
Don Jaime Jovanny Castillo Contreras

Firmado por CASTILLO CONTRERAS, JAIME JOVANNY (FIRMA) el día 14/07/2023 con un certificado emitido por AC DNIE 006

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

1. Introducción a los problemas medioambientales **1**
1.1. Contexto histórico general . . . 1
1.2. Contexto histórico del problema medioambiental tratado en los artículos . . . 4
1.3. Planteamiento del problema de predicción de contaminantes . . . 5
1.3.1. Datos generados en las cercanías de las instalaciones de la central térmica de As Pontes . . . 6

2. Modelos de regresión en media **9**
2.1. Preámbulo matemático . . . 9
2.2. Técnicas de predicción puntual . . . 11
2.2.1. Modelización paramétrica . . . 11
2.2.2. Modelización no paramétrica . . . 12
2.2.3. Modelización semiparamétrica . . . 12
2.2.4. Modelización semiparamétrica bidimensional . . . 13
2.3. Técnicas de predicción por intervalos asintóticos . . . 15
2.3.1. Modelización paramétrica . . . 15
2.3.2. Modelización no paramétrica . . . 16

3. Diversas técnicas estadísticas desarrolladas para atacar el problema **19**
3.1. Mecanismo de matriz histórica . . . 19
3.2. Modelización que incluye variables exógenas . . . 20
3.3. Utilización de redes neuronales artificiales . . . 22
3.4. Aprovechamiento de técnicas del contexto de datos funcionales . . . 24
3.4.1. Construcción de regiones de confianza vía bootstrap . . . 26

4. Modelos de regresión cuantil **29**
4.1. Técnicas bootstrap en regresión cuantil . . . 32

5. Otros modelos existentes en la literatura **33**
5.1. Modelo de redes neuronales con parámetros espaciales . . . 33
5.2. Modelización físico-estadística . . . 34
5.3. Modelo geoestadístico . . . 36
5.4. Modelo funcional bidimensional . . . 38

6. Ilustración de algunos métodos de predicción con datos reales **41**
6.1. Pequeño análisis de datos y problema de imputación . . . 42
6.2. Elección de la muestra de entrenamiento y test . . . 45
6.3. Implementación de modelos y resultados . . . 45
6.3.1. Box-Jenkins . . . 45
6.3.2. No paramétrico . . . 47
6.3.3. Semiparamétrico . . . 48

6.3.4. Redes neuronales 48
6.3.5. Comparación final 50

Bibliografía **51**

Apéndice **53**

Capítulo 1

Introducción a los problemas medioambientales

1.1. Contexto histórico general

El desarrollo humano con sus consecuentes avances tecnológicos, nos ha llevado a utilizar para nuestro beneficio y aprovechamiento cada uno de los recursos naturales que tenemos a nuestra disposición:

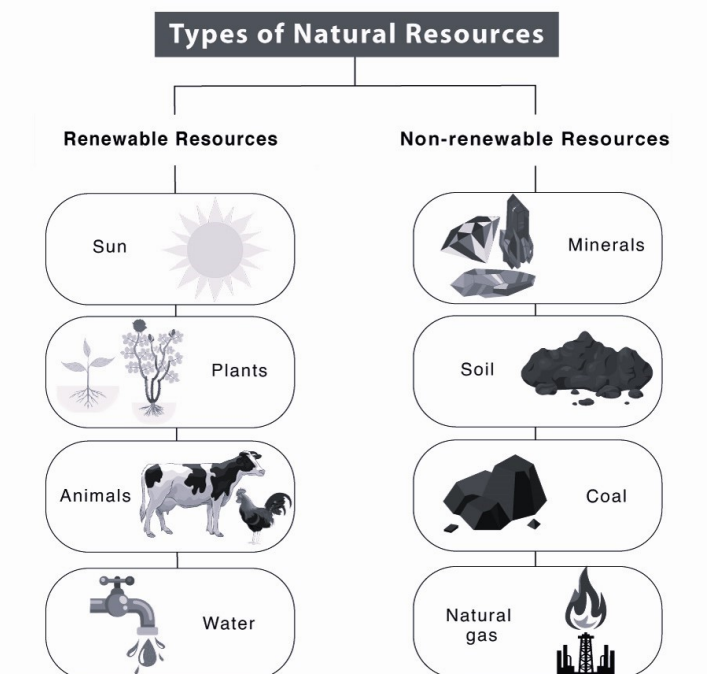


Figura 1.1: Clasificación de recursos naturales.

Como podemos apreciar en 1.1, los recursos naturales se pueden subdividir en dos grandes grupos:

- Renovables: aquellos que con el paso del tiempo se regeneran.
- No renovables: aquellos que existen en cantidades fijas sobre la tierra y que tardan cientos de miles de años en regenerarse.

Con la aparición de las primeras fábricas durante las revoluciones industriales el hombre consiguió implantar una economía de carácter urbano, industrial y mecanizado, dejando atrás siglos de arduo trabajo manual y la tracción animal para transporte de mercancías y pasajeros. Podemos destacar algunos de los primeros logros extraordinarios: la invención de la máquina de vapor de James Watt en 1769; en 1856 Bessemer diseña un proceso para producir acero a gran escala; para el año 1869 John Hyatt produce el primer plástico sintético de uso comercial; 1879 será cuando Von Siemens y Thomas Edison desarrollen de forma independiente los focos de luz de uso común, mientras que en 1880 Andrew Carnegie desarrolla el primer horno de fundición de acero de gran tamaño.

Sin embargo, de la mano de estos avances, aparecerán los primeros problemas medioambientales producidos por las dependencias industriales. Para el 1896 ya se había alzado la voz de Svante Arrhenius advirtiéndonos que el uso de combustibles fósiles *aceleraría el calentamiento* de la Tierra. Como podemos apreciar en la imagen 1.2, no se equivocaba.

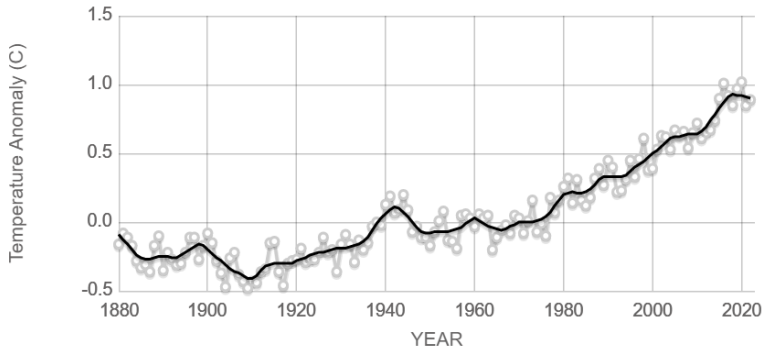


Figura 1.2: Cambio en la temperatura en la superficie terrestre comparada con la media a largo plazo entre 1951 y 1980 (en gris la media anual y en **negro** una **suavización lowess**).

La técnica de suavización *lowess* no es más que un ajuste polinómico local ponderado, empleado habitualmente para análisis exploratorios con el fin de intentar ver el tipo de relación que hay entre dos variables cuando ésta no es evidente a primera vista. [1]

Siguiendo cronológicamente, el primer gran desastre medioambiental tiene lugar en el año 1952. Para ese entonces llegará un fuerte frío durante el invierno a Londres, lo que llevará a la población a quemar mucho más carbón (de baja calidad, debido a la situación económica que atravesaba Inglaterra) con un alto contenido de azufre. *La Gran Niebla* cubriría entonces el cielo de Londres acabando con la vida de miles de personas (en su mayoría niños y personas con problemas respiratorios) por inhalación de humo y partículas de carbón, cerrando así uno de los capítulos más trágicos de la historia moderna.

A día de hoy, la situación no ha cambiado y más bien se ha agravado con respecto a lo que a temas de contaminación y sobre explotación de recursos naturales se refiere, añadiendo nuevos frentes con los que lidiar como se puede apreciar en 1.3.

[1] Para más información se recomienda consultar las referencias siguientes: desde un punto de vista más teórico Wasserman, L. (2005), *All of Nonparametric Statistics* o bien desde un punto de vista un poco más aplicado Bowman, A.W. y Azzalini, A. (1997), *Applied smoothing techniques for data analysis.*

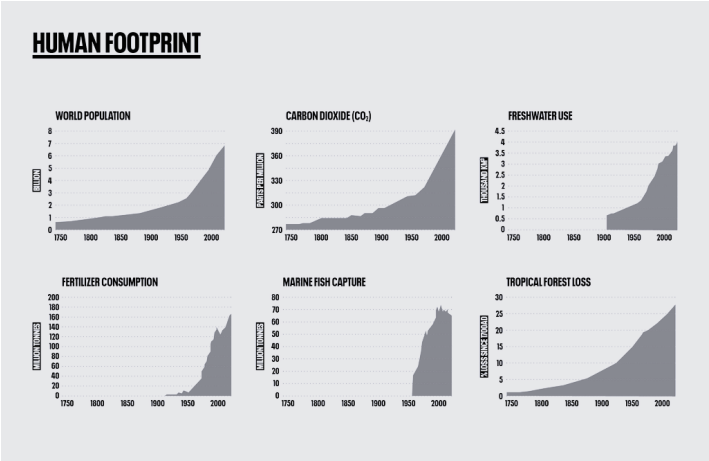


Figura 1.3: Algunos indicadores medioambientales

Un problema importante a tratar es el del exceso de generación de residuos (llegando a ser millones de toneladas), consecuencia de un consumo abusivo. En muchas ocasiones estos residuos y otros como pueden ser sustancias tóxicas procedentes de la industria y agricultura acaban en el mar, lo que genera un nuevo problema: la contaminación de los océanos. La agricultura no sostenible y la sobre explotación de madera está teniendo como consecuencia una agresiva deforestación, que a su vez tiene un efecto negativo en los ecosistemas, alterándolos y poniendo en peligro la biodiversidad (como podemos apreciar en 1.4).

(a) Contaminación del aire en Shangai.

(b) Limpiando crudo vertido en el mar por el buque *Exxon Valdez*.

(c) Contaminación por plásticos en el océano.

(d) Botellas y plásticos en una playa.

Figura 1.4: Algunos problemas medioambientales.

Con el fin de evitar incidentes como el de *La Gran Niebla* y el empeoramiento del ecosistema terrestre, es importante no solo poder controlar las emisiones producidas por la actividad de la producción de energía y sector industrial si no que también hay que aplicar medidas para la regulación del transporte, gestión de residuos, actividades agrarias y actividades domésticas. Es por ello que nacen organizaciones como por ejemplo Greenpeace, WWF, TNC, por mencionar algunas, con el fin de concienciar mediante campañas informativas, eventos y estudios a las personas sobre sus acciones y consecuencias al medioambiente fomentando medidas de conservación del planeta como el reciclaje o el uso de energías renovables a nivel personal e industrial. Asimismo, en los gobiernos de todo el mundo se crearon departamentos específicos con el fin de intentar proponer soluciones a estos problemas.

En cuanto al control de emisiones en el sector de producción de energía e industrial cabe destacar que los pioneros fueron los integrantes del parlamento del Reino Unido con el *Alkali Act 1863* que buscó limitar sus emisiones. El primer convenio internacional respecto a este asunto fue el de Ginebra 1979 sobre contaminación atmosférica transfronteriza (esto quiere decir que no es posible distinguir si la fuente es individual o colectiva) a gran distancia en el que se estableció un marco de cooperación intergubernamental para proteger la salud y el medioambiente contra la contaminación atmosférica que pueda afectar a varios países. A nivel europeo, la normativa vigente es la correspondiente a la directiva 2016/2284 relativa a la reducción de las emisiones nacionales de determinados contaminantes.

1.2. Contexto histórico del problema medioambiental tratado en los artículos

De entre todos los problemas medioambientales que existen, nosotros en este trabajo pondremos el foco en el control de emisiones buscando para ello predecir *contaminantes* en el *aire*. Actualmente sabemos que son dañinos tanto el ozono troposférico como el *polvo fino*. Con respecto al primero, está catalogado como un contaminante secundario, esto es, no es emitido directamente a la atmósfera si no que se forma a partir de reacciones fotoquímicas entre contaminantes primarios (concretamente cuando se juntan los óxidos de nitrógeno NO_x con los compuestos orgánicos volátiles $COVs$. Éstos últimos son hidrocarburos que están en estado gaseoso a temperatura ambiente normal o que bien son muy volátiles a dicha temperatura). Con lo que compete al segundo se emiten directamente a la atmósfera o también pueden formarse como partículas secundarias a partir de gases como el SO_2, los óxidos de nitrógeno NO_x y el amoníaco NH_3, siendo los dos primeros que hemos mencionado el objeto de nuestro estudio.

Nos remontamos al año 1976 cuando en la ciudad de A Coruña entra en funcionamiento, tras 4 años de previo acondicionamiento, la central térmica de As Pontes. Posee cuatro grupos de generación de 350 MW, preparada para trabajar con lignito local y capaz de cubrir por sí sola el 5 % de la demanda nacional de electricidad. Como dato curioso, su chimenea, *Endesa Termic*, en la fecha en la que se construyó era la más alta de Europa (con dimensiones 365 m de altura, 36 m en su base y 18 m en su cima).

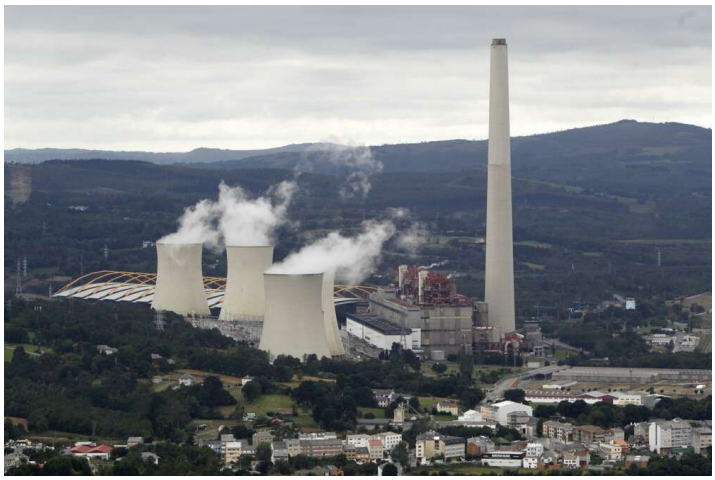


Figura 1.5: Central térmica de As Pontes

Como sabemos, no es oro todo lo que reluce y el alto contenido en azufre del lignito de la mina local empleado como materia prima la situó rápidamente entre las centrales más contaminantes de Europa. Esto llevará a la empresa a buscar fuentes alternativas.

El primer cambio sustancial tendrá lugar en el período que va de 1993 a 1996 utilizando mezclas (en proporción 50/50) de carbón local e importado (con menor contenido de azufre). Tras varios años en funcionamiento, en el año 2004 se extrajo en su totalidad el lignito de la mina local, por lo que hubo que llevar a cabo una nueva adaptación; en esta ocasión, en un período que abarca del año 2005 hasta 2008, con el uso al 100 % de carbón importado. Cerca de la central térmica, en este mismo año, 2008, entra en marcha un nuevo grupo de ciclo combinado alimentado con gas natural. Posee un generador consistente en dos turbinas de gas y una de vapor; como materia prima de uso en caso de emergencia emplea diésel. Así finaliza el período de actualizaciones en busca de fuentes alternativas.

Continuando cronológicamente, en el año 2018 liderará el ránking de centrales de carbón que más redujo sus emisiones, disminuyéndolas en un 70 %. En 2019 Endesa formalizará ante el Ministerio de Transición Ecológica la solicitud de cierre de los cuatro grupos de carbón. Entre los motivos de esta decisión estaban la fuerte caída del precio del gas así como el incremento del precio de los derechos de CO_2. Para mediados de 2021 Endesa apuntaba a su cierre total, sin embargo con el inicio de la guerra entre Rusia y Ucrania y la consecuente crisis del gas en 2022, el gobierno decidió mantener abierta parcialmente la central térmica (manteniendo dos generadores abiertos); se prevé que la situación siga así hasta la resolución del conflicto.

1.3. Planteamiento del problema de predicción de contaminantes

Como comentamos en apartados anteriores, en cualquier instalación generadora de energía mediante combustión, se necesita contar con un sistema de control de emisiones. Por esta razón, en la central térmica de As Pontes se integró dicho sistema al ya existente de control de calidad del aire.

Con este sistema, se cuantificarán los niveles de contaminantes presentes en dichas emisiones y con estos datos se buscará conocer, en particular, los *futuros niveles* de SO_2. Posteriormente, con la nueva estación de ciclo combinado instalada en el año 2008, se intentó *predecir las futuras emisiones de* NO_x. En la actualidad, se han aunado esfuerzos con el objetivo de obtener una *predicción simultánea de los niveles de* SO_2 *y* NO_x.

Para cumplir con los requisitos de calidad del aire estipulados por las organizaciones gubernamentales se necesitan obtener las predicciones de concentraciones a tiempo futuro. Para ello se emplearán

diversos modelos estadísticos que van desde las clásicas series de tiempo (metodología Box-Jenkins), pasando por modelos semiparamétricos, hasta otras metodologías más actuales como pueden ser los modelos de datos funcionales o las redes neuronales.

1.3.1. Datos generados en las cercanías de las instalaciones de la central térmica de As Pontes

Antes de empezar con la revisión de los modelos de predicción mencionados, vamos a comentar algunas características de los datos con los que se trabajaron. Notemos antes de nada que se consideró la *concentración media por hora* para obtener predicciones de los posibles valores futuros de SO_2 y NO_x.

- Los datos recibidos para ese entonces, habitualmente eran en tiempo real, bien minutales o pentaminutales.

- Las series de SO_2 medias por hora tienen un comportamiento singular, fuertemente influenciado por las condiciones climatológicas de la región y la topografía local.

- Toman valores cercanos a cero por largos períodos, como se aprecia en la figura 1.6

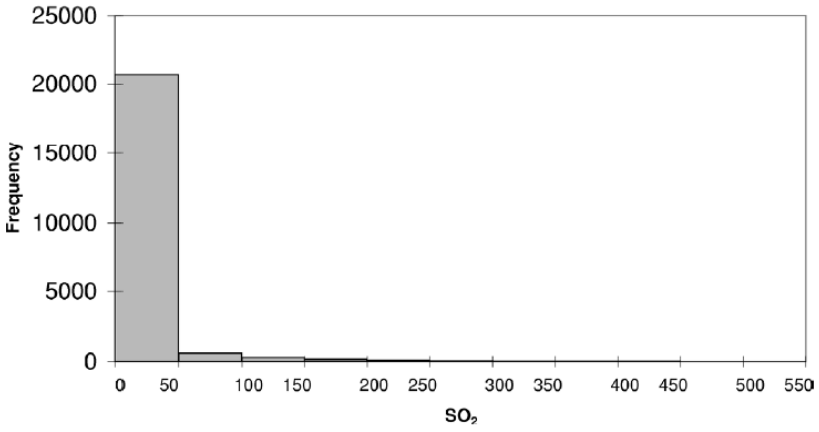


Figura 1.6: Histograma de valores diarios medidos cada 5 min de SO_2 en la central térmica de As Pontes entre los meses de abril y julio de 2022. Imagen tomada del artículo [3]

- Presentan episodios, ampliamente espaciados en el tiempo, donde se producen incrementos repentinos y abruptos, como se aprecia en 1.7

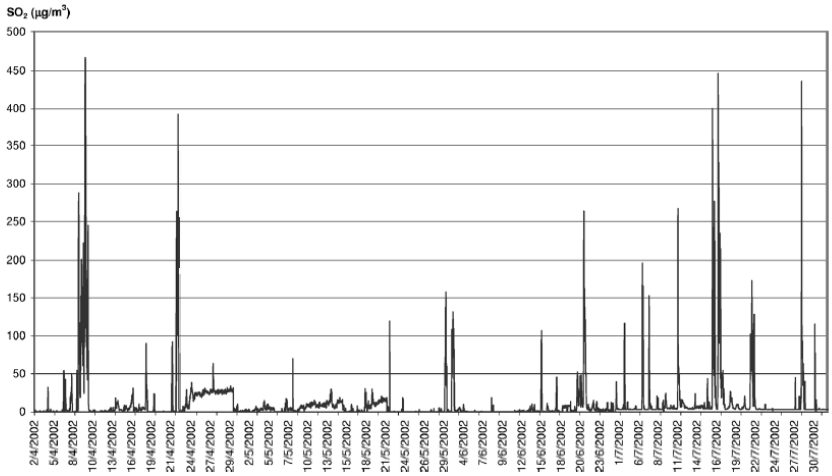


Figura 1.7: Valores diarios medidos cada 5 min de SO_2 medidos en la central térmica de As Pontes entre los meses de abril y julio de 2002. Imagen tomada del artículo [3].

- Las series de NO_x medias por hora tienen un comportamiento similar a las de SO_2, pero en una escala más pequeña, como podemos ver en 1.8.

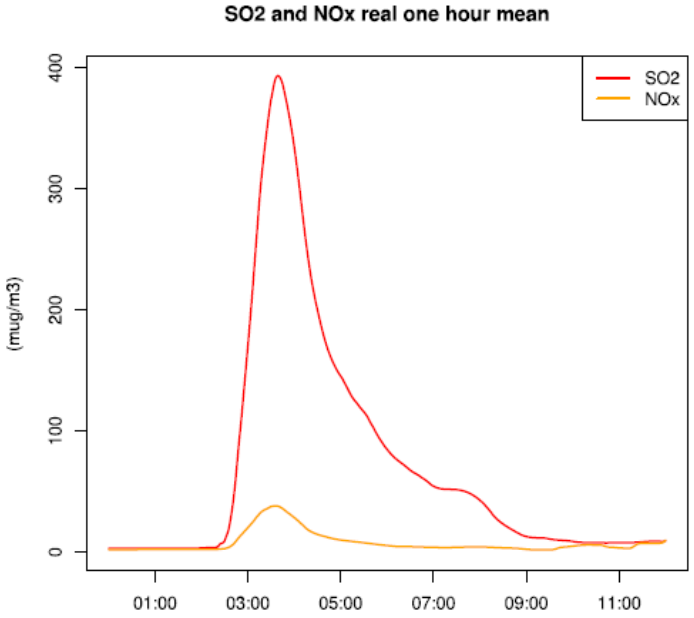


Figura 1.8: Comparativa de un ejemplo de las medias horarias de concentraciones de los niveles de SO_2 y NO_x medidos en la central térmica de As Pontes. Imagen sacada del artículo [6].

Capítulo 2

Modelos de regresión en media

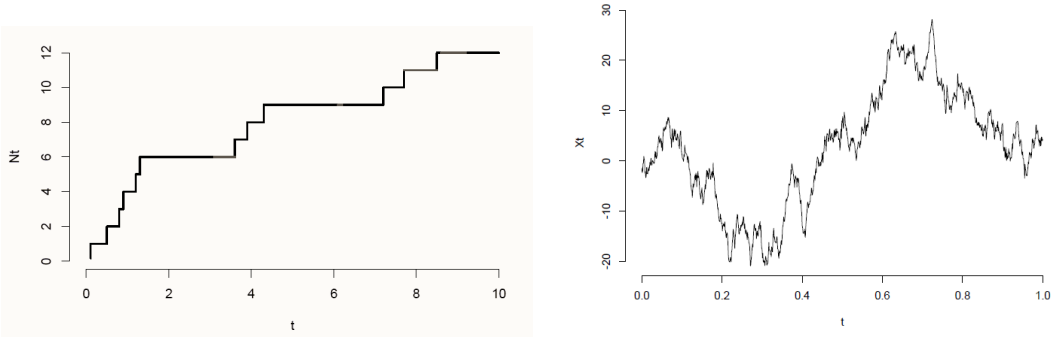
En los próximos capítulos, se comentarán los detalles más relevantes de los modelos para predicción principalmente de los niveles de SO_2 (si bien es cierto que mencionaremos alguno de predicción simultánea de SO_2 y NO_x) desarrollados a lo largo de estos años, desde los más antiguos hasta los más actuales, citando debidamente los artículos donde se explican cada uno de ellos con mayor profundidad.

2.1. Preámbulo matemático

El objeto matemático que nos permitirá modelar el comportamiento de los niveles de SO_2 y NO_x es el de *proceso estocástico*: esto no es más que un conjunto de variables aleatorias $\{X_t\}_{t\in C}$ (con C el conjunto de índices), definidas sobre un mismo espacio de probabilidad. En particular nosotros consideraremos procesos estocásticos donde $C = \mathbb{Z}$, esto es, el proceso estocástico será $\{\cdots, X_{-2}, X_{-1}, X_0, X_1, X_2, \cdots\}$, donde el subíndice t representa el instante de tiempo en que es observada.

Una observación de un proceso estocástico se denotará por $\cdots, x_{-2}, x_{-1}, x_0, x_1, x_2, \cdots$ y se conoce como una *realización* o *trayectoria* del mismo.

Una *serie de tiempo* no es más que una realización o trayectoria parcial de un proceso estocástico: $x_1, x_2, \cdots, x_T$. En la figura 2.1 podemos apreciar dos ejemplos.



(a) Trayectoria parcial de un proceso de Poisson. (b) Trayectoria parcial de un movimiento browniano.

Figura 2.1: Ejemplo de trayectorias parciales

Una hipótesis interesante y útil que requerimos habitualmente para trabajar con los procesos estocásticos subyacentes generadores de una serie de tiempo es la de *estacionariedad*: un proceso es-

tocástico $\{X_t\}_t$ se dice estacionario si se cumple que

- $E(X_t) = \mu_t = \mu$ para todo t.
- $Var(X_t) = \sigma_t^2 = \sigma^2$ para todo t.
- Para todo t, k, $Cov(X_t, X_{t+k})$ depende únicamente de k.

Veamos un par de ejemplos.

1. Sean $\{X_t\}_t$ variables aleatorias incorreladas, con media nula y varianza finita σ_a^2 (también conocido como *ruido blanco*. Se puede apreciar un ejemplo en la primera imagen de 2.2). Es inmediato ver que:

   a) $E(X_t) = 0$ para todo t.

   b) $Var(X_t) = \sigma_a^2$, para todo t.

   c) $Cov(X_t, X_{t+k}) = E(X_t X_{t+k}) = \begin{cases} \sigma_a^2, & k = 0 \\ 0, & k \neq 0 \end{cases}$

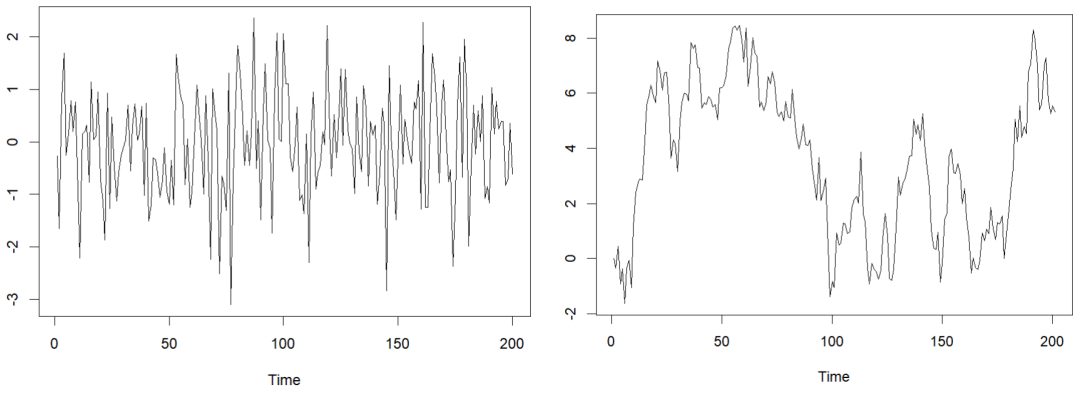
2. Consideremos ahora $X_t = ct + \sum_{j=1}^{t} a_j$ con $\{a_t\}_t$ ruido blanco y c un parámetro (este proceso estocástico es conocido como *random walk*; tenemos un ejemplo de éste en segunda figura de la imagen 2.2). Supongamos además que $t = 0$ y que $X_0 = 0$. Es fácil ver que:

   a) $E(X_t) = ct$ para $t = 1, 2, \cdots$.

   b) $Var(X_t) = \sigma_a^2 t$ para $t = 1, 2, \cdots$.

   c) $Cov(X_t, X_{t+k}) = E\big(\sum_{i=1}^{t} a_i \sum_{j=1}^{t+k} a_j\big) = \text{mín}\{t, t+k\}\sigma_a^2$, con $t, t+k = 1, 2, \cdots$.

El primer caso se trata de un proceso estacionario ya que cumple todas las condiciones de la definición dada, mientras que el segundo es evidentemente no estacionario, ya que posee una varianza *explosiva*, esto es, que al tomar t tendiendo a infinito se va haciendo cada vez más y más grande.



(a) Ejemplo de proceso estacionario: ruido blanco. (b) Ejemplo de proceso no estacionario: *random walk*.

Figura 2.2: Hipótesis de estacionariedad

El objetivo de esta memoria es revisar los diferentes modelos de predicción para los *futuros valores* de una serie de tiempo partiendo de un origen T, h instantes posteriores, empleando para ello diversas técnicas estadísticas.

Ahora bien, ¿cómo medimos qué tan buenas son nuestras predicciones? Para ello, vamos a mencionar algunas medidas de error habituales en predicción de series de tiempo. Supongamos pues que tenemos hecha ya una predicción con origen T y horizonte h que denotaremos por $\hat{x}_T(h)$; conociendo el valor que se pretendía predecir, x_{T+h}, se puede construir el error de predicción observado: $e_T(h) = x_{T+h} - \hat{x}_T(h)$. Así podemos definir

- MAE (mean absolute error)= $\frac{1}{H}\sum_{h=1}^{H} |e_T(h)|$.
- RMSE (root mean squared error)= $\sqrt{\frac{1}{H}\sum_{h=1}^{H} e_T^2(h)}$.
- MAPE (mean absolute porcentual error)= $\frac{1}{H}\sum_{h=1}^{H} |p_T(h)|$, siendo $p_T(h) = \frac{100 e_T(h)}{x_{T+h}}$.

Ahora pasaremos a fijar la notación empleada en los artículos. Sea (Z_l, Y_l), con $l = 0, \pm 1, \pm, \cdots$, series de tiempo estacionarias, siendo Z_l una serie r-dimensional e Y_l, la respuesta, una serie unidimensional. Queremos estimar $\varphi(z_l^0) = \varphi(F(\cdot|Z_l = z_l^0))$ donde $F(\cdot|Z_l = z_l^0)$ es la distribución condicional de Y_l dada $Z_l = z_l^0$, usando una muestra de tamaño n, $\{(Z_i, Y_i)\}_{i=1}^n$. En nuestro caso tomaremos como función φ la esperanza matemática.

En particular, cuando $Y_l = X_{l+k}, k \geq 1$ y $Z_l = (X_l, \cdots, X_{l-r+1})$, siendo X_l una serie estacionaria, estamos estimando la función de autoregresión de orden k,

$$\boxed{\varphi(x_1^0, \cdots, x_r^0) = E\big(X_{l+k}|(X_l, \cdots, X_{l-r+1}) = (x_1^0, \cdots, x_r^0)\big)} \tag{2.1}$$

usando la muestra $\{X_{t-m+1}, \cdots, X_t\}$ de tamaño m.

En los primeros años de desarrollo, la frecuencia de transimisión de datos era pentaminutal y la legislación de la época fijaba valores límite para la media bihoraria de SO_2, esto es,

$$X_t = \frac{1}{24}\sum_{i=0}^{23} SO_2(t-i),$$

donde $SO_2(t)$ representa la concentración de SO_2 en tiempo t (pentaminutal) medida en $\mu g/m^3$.

Para mantener el control de las concentraciones del mismo, la estación térmica tendría que actuar con al menos media hora de antelación (esto es, supongamos que recibimos un nuevo dato X_t, pues nosotros buscaremos predecir X_{t+6}) para poder actuar antes de que se produzcan niveles anormalmente altos. Es por ello que la velocidad y precisión de la predicción juegan un papel clave.

2.2. Técnicas de predicción puntual

2.2.1. Modelización paramétrica

Inicialmente para intentar resolver el problema 2.1 se trabajó empleando la metodología Box-Jenkins con modelos **ARIMA**.

Un *proceso estacionario* $\{X_t\}_t$ se dice ARMA(p, q) si admite la representación

$$X_t = c + a_t + \sum_{i=1}^{p} \phi_i X_{t-i} + \sum_{j=1}^{q} \theta_j a_{t-j},$$

siendo $\{a_t\}_t$ un proceso de ruido blanco (colección de variables aleatorias incorreladas, de media nula y varianza finita) y $c, \phi_1, \cdots, \phi_p, \theta_1, \cdots, \theta_q$ (con $\phi_p \neq 0, \theta_q \neq 0$) constantes. Otra forma de escribir los procesos ARMA(p, q) es la siguiente

$$\phi(B)X_t = c + \theta(B)a_t$$

siendo B el operador retardo definido por $BX_t = X_{t-1}$, $\phi(B) = 1 - \sum_{i=1}^{p} \phi_i B^i$ y $\theta(B) = 1 + \sum_{j=1}^{q} \theta_j B^j$.

Decimos que un proceso estocástico pertenece a la familia ARIMA(p, d, q) siempre que se pueda escribir en la forma

$$\phi(B)(1 - B)^d X_t = c + \theta(B)a_t$$

pero en esta ocasión el polinomio $\phi(B)$ no puede tener raíces de módulo 1. Otra forma equivalente de definir este modelo es que al diferenciar regularmente d veces se obtiene un proceso ARMA(p, q). Por ejemplo, supongamos que tenemos un proceso estocástico $\{Y_t\}_t$; si al trabajar con el proceso estocástico $Y_t - Y_{t-1}$ obtenemos un proceso ARMA(p, q), entonce estaríamos ante un proceso ARIMA$(p, 1, q)$.

Los datos con los que se trabajó presentaban una fuerte *falta de estacionariedad* durante los episodios de picos de concentraciones de SO_2. Esto provocaba que al ajustar dichos modelos, las innovaciones $\{a_t\}_t$, o mejor dicho, sus versiones *estimadas* a través de los residuos, fueran distintas de ruido blanco, lo que directamente invalidaba la utilización del modelo.

2.2.2. Modelización no paramétrica

Posteriormente, se intentó hacer una **modelización no paramétrica**. El contexto teórico de este modelo es el siguiente: supongamos que queremos estimar una variable aleatoria Y empleando como datos el conjunto de variables X. Podemos entonces escribir

$$Y = m(X) + \varepsilon,$$

donde $m(x) = E(Y|X = x)$ es una función a estimar. Además, debe cumplirse que $E(\varepsilon|X = x) = 0$. La estimación de estos modelos viene dada por el regresograma, vecinos más próximos y Nadaraya Watson, por mencionar algunos de los más importantes en la literatura. Dada una muestra de tamaño n, $\{(X_i, Y_i)\}_{i=1}^n$ este último tiene la forma:

$$\hat{m}(x) = \frac{\sum_{i=1}^{n} K\left(\frac{x - X_i}{h}\right) Y_i}{\sum_{j=1}^{n} K\left(\frac{x - X_j}{h}\right)} = \sum_{i=1}^{n} W_{i,h}(x) Y_i,$$

siendo $K(\cdot)$ una función tipo núcleo y h el parámetro de suavizado.

Hemos destacado la expresión explícita de este último estimador puesto que en el artículo [5] se obtuvo una generalización de la siguiente forma: supongamos que para estimar 2.1 usamos una muestra $\{(Z_i, Y_i)\}_{i=1}^n$, entonces se propondrá

$$\hat{\varphi}(z_l^0) = \sum_{i=1}^{n} W_{ni}\big(z_l^0, (Z_1, Y_1), \cdots, (Z_n, Y_n)\big) Y_i. \tag{2.2}$$

2.2.3. Modelización semiparamétrica

García-Jurado et al. (1995) en [5] propondrán una corrección al modelo no paramétrico, ideando un nuevo sistema de predicción consistente en la suma de una prediccción no paramétrica y una predicción paramétrica Box-Jenkins sobre la componente ARMA de las series, escrito formalmente,

$$Y_l = \varphi(Z_l) + e_l, \tag{2.3}$$

donde e_l tiene una estructura ARMA(p, q) independiente de Z_l; buscamos predecir Y_t después de observar las series Y_l hasta tiempo $t - k$ y Z_l hasta tiempo t. En particular, usando la muestra $(Z_{t-n+1-k}, Y_{t-n+1-k}), \cdots, (Z_{t-k}, Y_{t-k})$ de tamaño n, la predicción $\hat{Y}_t$ de Y_t viene dada por

$$\hat{\varphi}_n(Z_t) + \dot{e}_t,$$

donde $\hat{\varphi}_n$ es una estimación de 2.1 usando 2.2 ; $\dot{e}_t$ es la estimación Box-Jenkins k etapas por delante, construida con la componente ARMA de las series $\hat{e}_t = Y_t - \hat{\varphi}_n(Z_t)$.

Aplicado a la necesidad concreta de la empresa, se aplicó este modelo descrito a:

$$\boxed{X_{t+6} = E(X_{t+6}|X_t, X_{t-1}) + e_{t+6}}$$

El ***algoritmo***, para el caso en concreto que nos concierne sería:

1. Para cada t empleando una muestra con los datos relativos a las seis últimas horas, se estimó $E(X_{t+6}|X_t, X_{t-1})$ no paramétricamente mediante un estimador Nadaraya-Watson con un núcleo gaussiano y validación cruzada usando 2.2, .

2. Se calculan las series de tiempo residuales relativas a las últimas seis horas $\hat{e}_{t-64}, \cdots, \hat{e}_t$, donde $\hat{e}_i = X_i - \hat{E}(X_i|X_{i-6}, X_{i-7})$.

3. Para cada i y se ajustó el modelo ARIMA apropiado (empleando para ello algunos métodos de estimación como pueden ser mínimos cuadrados, mínimos cuadrados condicionados o máxima verosimilitud).

4. A continuación, se obtuvo la predicción Box-Jenkins para $\hat{e}_{t+6}$. El predictor puntual es entonces

$$\hat{X}_{t+6} = \hat{E}(X_{t+6}|X_t, X_{t-1}) + \hat{e}_{t+6}.$$

2.2.4. Modelización semiparamétrica bidimensional

Mientras que hasta el momento solo hemos revisado modelos de predicción de SO_2, ahora veremos un ejemplo en el que se obtiene una predicción conjunta con NO_x.

En el artículo [6] de Wenceslao Manteiga et al. (2009), buscan desarrollar una generalización del modelo semiparamétrico propuesto por García Jurado et al. (1995) en [5] al caso bidimensional empleando la *estructura de correlación* entre las series que se pretenden predecir.

Empezaremos dando un par de definiciones.

Un proceso estocástico X_t se dirá *integrado de orden d*, y se denotará por $I(d)$, siempre que haciendo d diferencias regulares nos de como resultado un proceso estacionario.

Así por ejemplo, decir que $X_t \sim I(1)$ quiere decir que al definir $Y_t := X_t - X_{t-1}$, Y_t es estacionario; es claro que decir $X_t \sim I(0)$ significa que se trata de un proceso estacionario.

Sea $Z_t = (z_{1t}, \cdots, z_{nt})^T$ un vector formado por n series de tiempo $I(1)$. Z_t se dice *cointegrado* si existe un vector $\beta = (\beta_1, \cdots, \beta_n)^T$ tal que

$$\beta^T Z_t = \beta_1 z_{1t} + \cdots + \beta_n z_{nt} \sim I(0).$$

Cabe destacar que el vector β no es único; podríamos tomarlo normalizado, esto es, de la forma $\beta = (1, -\beta_2, \cdots, -\beta_n)$ y entonces la cointegración podría representarse de forma equivalente como

$$z_{1t} = \beta_2 z_{2t} + \cdots + \beta_n z_{nt} + u_t$$

con $u_t \sim I(0)$ llamado error de desequilibrio o residuo cointegral.

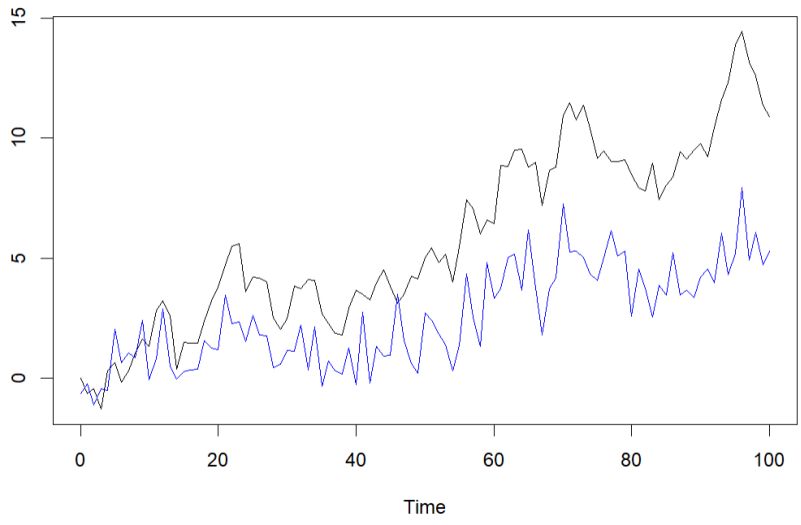


Figura 2.3: Ejemplo de series cointegradas: en azul tenemos la serie $z_{1t} = 0{,}45 z_{2t} + u_t$ con u_t valores aleatorios de una normal estándar y z_{2t} un proceso ARIMA(0,1,0)

Lo adecuado en el contexto en el que nos estamos moviendo (buscar predecir simultáneamente SO_2 y NO_x empleando sus correspondientes series) sería emplear un modelo $VAR(p)$ para el vector de n series Z_t

$$Z_t = \Phi D_t + \Pi_1 Z_{t-1} + \cdots + \Pi_p Z_{t-p} + \varepsilon_t, \quad t = 1, \cdots, T$$

donde D_t contiene términos deterministas y $\{\Pi_i\}_{i=1}^p$ matrices de coeficientes.

Sin embargo, si suponemos que Z_t es $I(1)$ y posiblemente cointegrado, entonces la representación VAR no es adecuada para realizar análisis dado que se desconocen las posibles relaciones de cointegración.

Por ello, el modelo VAR se cambia por un modelo $VECM(p)$, siendo $\Delta Z_t := Z_t - Z_{t-1}$.

$$\Delta Z_t = \Phi D_t + \Pi Z_{t-1} + \Gamma_1 Z_{t-1} + \cdots + \Gamma_{p-1} \Delta_{p-1} Z_{t-p+1} + \varepsilon_t,$$

donde

$$\Pi = \Pi_1 + \cdots + \Pi_p - I_n$$

y

$$\Gamma_k = -\sum_{j=k+1}^{p} \Pi_j, \; k = 1, \cdots, p-1.$$

A estas matrices se les llama respectivamente matriz *long-run impact* y matriz *short-run impact*. La primera de ellas nos da el *rango de cointegración* y nos da idea del número de relaciones de cointegración que existen (esto es desconocido por norma general por lo que Johansen propone un método secuencial basado en *likelihood ratio tests* para estimar este rango).

Presentados estos modelos y definiciones, fijemos el contexto en el que nos moveremos. Consideremos (Y_l, Z_l) con $l = 0, \pm 1, \pm 2, \pm 3, \cdots$ con Y_l series r-dimensionales integradas de orden 1 (probablemente cointegrada) y Z_l una serie q-dimensional.

Se considera el siguiente modelo

$$\boxed{Y_l = \varphi(Z_l) + e_l}$$

donde e_l tiene una estructura $VECM(p)$ independiente de Z_l.

La predicción de Y_t está dada por

$$\dot{Y_t} = \hat{\varphi}(Z_t) + \dot{e_t}$$

donde $\hat{\varphi}_n(Z_t)$ es la estimación no paramétrica de φ y $\dot{e_t}$ la predicción dada por el modelo VECM, k etapas por delante para las series residuales construidas como $\hat{e}_t = Y_t - \hat{\varphi}_n(Z_t)$.

Si bien es cierto que podríamos omitir la explicación del algoritmo adaptado al caso bidimensional, lo detallaremos un poco más con fines ilustrativos de apreciar las diferencias de trabajar con un modelo univariante vs bivariante.

En su día, se consideró X_t siendo la serie horaria bidimensional de niveles de SO_2 y NO_x en cada minuto t. Siguiendo el modelo descrito $Y_t = X_{t+k}$ y $Z_t = (X_t, X_t - X_{t-5})$. Buscamos predecir X_{t+30} y para ello emplearemos el siguiente ***algoritmo***

1. Para cada instante t, $\varphi(Z_t)$ es estimado con *modelos aditivos* (básicamente lo que se hace aquí es suponer que φ es aditiva con q componentes y su estimación se hace mediante técnicas *smooth backfitting*) y la información provista por la matriz histórica, independientemente para cada componente. La estimación de φ se hace 30 instantes por delante: $\dot{Y_t} = \dot{X_{t+30}} = \hat{\varphi}_{30}(Z_t) + \dot{e_{t+30}}$.
2. Las series residuales $\hat{e}_t$ se calcula haciendo $\hat{e}_t = Y_t - \hat{\varphi}_{30}(Z_t)$ y un test de adecuación al modelo considerado se efectúa para cada componente de las series correspondiente a las últimas 4 horas.
3. Si alguna de las series residuales no es ruido blanco se efectúa un test para ver si la serie residual vectorial es cointegrada. De ser el caso, se ajustará un modelo $VECM$ adecuado. Si las series no presentan cointegración, un modelo VAR se ajusta.
4. Obtenemos $\dot{e}_{t+30}$.
5. La estimación viene dada por $\dot{X}_{t+30} = \hat{\varphi}_{30}(Z_t) + \dot{e}_{t+30}$.

El artículo finaliza comentando que estas correciones de los errores empleando los modelos VAR(p) y VECM(p) combinados con el modelo semiparamétrico, cuando hay una estructura de dependencia entre las series, hacen una predicción bastante precisa.

2.3. Técnicas de predicción por intervalos asintóticos

Hasta aquí, nos hemos dedicado únicamente a revisar modelos de predicción puntual; esta sección está dedicada a ver brevemente un poco el trabajo hecho en [5] para la predicción por intervalos asintóticos.

2.3.1. Modelización paramétrica

Situémonos nuevamente en el contexto correspondiente a la sección 2.2.1. Empleando como estimación de $\dot{e_t}$ el ARMA estimado empleando el conjunto $\{\hat{e}_{t-(n+k)+1} = Y_{t-(n+k)+1} - \hat{\varphi}_n(Z_{t-(n+k)+1}), \cdots, \hat{e}_{t-k} = Y_{t-k} - \hat{\varphi}_n(Z_{t-k})\}$, un intervalo de predicción asintótico de nivel α para Y_t sería

$$\hat{\varphi}_n(Z_t) + \dot{e_t} \pm z_{\alpha/2}\left(\hat{\sigma}^2 \sum_{j=0}^{k-1} \hat{\pi}_j^2\right)^{1/2}$$

donde $z_{\alpha/2}$ representa el cuantil $1 - \alpha/2$ de la normal estándar, $\hat{\sigma}^2$ la estimación de la varianza asociada a la componente de ruido blanco de la series ARMA $\{e_l\}$ y $\hat{\pi}_j$ los coeficientes estimados de los polinomios π_j obtenidos de la relación

$$\pi(B) = \frac{\theta(B)}{\phi(B)(1-B)^d}$$

donde θ y ϕ son los coeficientes estimados de la parte ARMA $\{\hat{e}_t\}$.

Esto requiere que cuando decimos que vamos a ajustar un modelo ARMA para $\{e_l\}$, intrínsecamente estamos demandando como hipótesis que $\{a_t\}$ sea ruido blanco.

2.3.2. Modelización no paramétrica

¿Qué ocurre entonces si $\{a_t\}$ no es ruido blanco? Es en este momento, cuando pueden ser de utilidad *técnicas bootstrap*, éstas lo que buscan es intentar *aproximar* la *distribución* (desconocida) de un determinado *estadístico* empleando *remuestras* de la muestra original.

Bajo ciertas condiciones, se podrá asegurar que $F_\theta^* \xrightarrow{d} F_\theta$, esto es, que la *distribución bootstrap* del estadístico, converga en *distribución* a la *distribución original* que estamos buscando. Con esto hecho, podremos construir e_t^* teniendo garantía teórica de que estamos aproximando correctamente e_t[1].

La aplicación de estas técnicas a series de tiempo es, cuanto menos no inmediata, y es numerosa la bibliografía existente relativa a este tema.

Aprovechándose de ésta, García-Jurado, I. et al.(1995) propondrán un mecanismo de construcción del intervalo de predicción deseado inspirados por trabajos anteriores en esta área. Para ello, supondrán que en 2.3 $\{e_t\}$ tiene una estructura ARI(q,d), esto es

$$\phi(B)(1-B)^d e_l = a_l$$

con $\{a_l\}$ ruido blanco. Entonces, cuando efectuamos d diferencias regulares a este proceso se obtiene que las series $\tilde{e}_l = \nabla^d e_l$ siguen un modelo AR(q). Una vez aquí, es cuando en [5] se recoge de la literatura el mecanismo de Thombs y Schucany (1990) [18] propuesto para aproximar la distribución de e_t dados $e_{t-k}, e_{t-k-1}, \cdots, e_{t-(n+k)+1}$. El resultado en el que se basan básicamente dice lo siguiente:

Sea Y_j un proceso estacionario autorregresivo con de orden p, esto es, siendo $t = 0, \pm 1, \pm 2, \cdots$

$$Y_t = \delta + \sum_{i=1}^{p} \phi_i Y_{t-i} + a_t$$

con $\{a_j\}$ una sucesión de variables aleatorias independientes, de media nula y misma distribución F_a, con $E(a^2) = \sigma^2 < \infty$ y $c = (\delta, \phi_1, \cdots, \phi_p)$ constantes desconocidas. También se pedirá que $E(a^\alpha) < \infty$ para algún $\alpha > 2$. Denotamos por $c^* = (\delta^*, \phi_1^*, \cdots, \phi_p^*)$ los coeficientes obtenidos por bootstrap. Denotemos además por $(y_{t-n+1}, \cdots, y_t)$ una realzación de Y_j. Entonces a lo largo de casi todas las sucesiones muestrales, con $n \to \infty$ se tiene

- $c \to c^*$ en probabilidad condicional.
- $Y_{t+k}^* \xrightarrow{d} Y_{t+k}$.

En nuestro caso particular podemos obtener $\tilde{e^*}_{t+i-k}$ con $i = -n+1+d, \cdots, -q$ e $i = 1, \cdots, k$ y podemos producir las series bootstrap $\{\tilde{e^*}_{t-(n-d)+1-k}, \cdots, \tilde{e^*}_{t-q-k}, \tilde{e}_{t-q-k+1}, \cdots, \tilde{e}_{t-k}, \tilde{e^*}_{t-k+1}, \cdots, \tilde{e_t^*}\}$.

Los autores prueban que e_t^* se puede expresar en términos de $\tilde{e^*}_t, \cdots, \tilde{e^*}_{t-k+1}, e_{t-k}, \cdots, e_{t-d-k+1}$ (todos conocidos o bien estimables). La réplica bootstrap de este proceso un gran número de veces, permite obtener un intervalo de predicción de k etapas por delante de e_t:

$$\left(z_t^{*(\alpha/2)}, z_t^{*(1-\alpha/2)}\right)$$

siendo los extremos del intervalo el cuantil $\alpha/2$ y $1-\alpha/2$ respectivamente de la distribución bootstrap de e_t^*.

Por tanto, bajo el modelo inicialmente planteado, un intervalo de predicción asintótico semiparamétrico para Y_t usando bootstrap viene dado por

$$\left(\hat{\varphi}_n(Z_t) + \hat{z}_t^{*(\alpha/2)}, \hat{\varphi}_n(Z_t) + \hat{z}^{*(1-\alpha/2)}\right)$$

[1] Se puede ver más información con respecto a esta temática en *An Introduction to the Bootstrap*, B. Efron and R. J. Tibshirani (1994).

donde los cuantiles $\hat{z}^*$ son obtenidos de la componente ARMA $\{e_t\}$.

En ese mismo artículo [5] se comprobó experimentalmente que este intervalo de predicción resultó ser mejor que el intervalo de predicción inicialmente propuesto (aquel que trabajaba con la hipótesis de que $\{a_t\}_t$ era ruido blanco).

Capítulo 3

Diversas técnicas estadísticas desarrolladas para atacar el problema

3.1. Mecanismo de matriz histórica

Antes de continuar con la revisión que estamos haciendo, debemos incluir otra pequeña sección con el fin de destacar otra idea original para obtener las predicciones deseadas, si bien es cierto que no constituye un modelo como tal.

Como comentamos, los episodios de picos de SO_2 están ampliamente espaciados en el tiempo por lo que, probablemente, la muestra relativa a las seis últimas horas, contendrá, pocos de estos episodios; además de ello, la serie mientras no se producen dichos picos toma valores próximos a cero.

Por ello que en el artículo de Prada-Sánchez y Febrero-Bande (1997) [12] se revisa una idea ya empleada en [5], pensando en crear un sistema con dos objetivos:

- Usar la experiencia acumulada hasta ese momento.
- Obtener una muestra que sea representativa de los incidentes (ignorando la mayoría de observaciones similares cercanas a cero).

Así pues, procedieron de la siguiente forma:

1. En el artículo [12] para obtener la estimación de X_{t+6} se emplean un conjunto de 1000 tríadas $\{\big((x_1^1, x_2^1), x_8^1\big), \cdots, \big((x_1^{1000}, x_2^{1000}), x_8^{1000}\big)\}$, donde x_8^i es representativa de todo el rango de valores no cercanos a cero en las medias de 2 h y está acompañado de su par para predicción (x_1^i, x_2^i).

2. Usando las observaciones de los últimos 2 años se determinaron el rango de valores distintos de cero de las medias bihorarias.

3. Este rango fue divido en diez estratos conteniendo aproximadamente igual número de valores.

4. Posteriormente, al azar, se seleccionaron 100 valores x_8^i de cada estrato y se formaron las tríadas $\big((x_1^i, x_2^i), x_8^i\big)$ con los x_8^i y las medias bihorarias x_1^i, x_2^i observadas respectivamente en los 35 y 30 min previos.

Las 1000 triadas de esta muestra constituyen la semilla inicial $M_0^{2,6}$ de lo que se llamó en su día como la *matriz histórica* $M_t^{2,6}$.

Ahora bien, esta matriz es dinámica en el siguiente sentido actualizándose de la siguiente forma: cuando se recibe un dato nuevo x_t se determina cuál es el primer valor del estrato al que pertenece; la tríada en el estrato en $M_{t-1}^{2,6}$ se reemplaza por la tríada $\big((x_{t-7}, x_{t-6}), x_t\big)$ para construír $M_t^{2,6}$.

Las predicciones de la parte no paramétrica se vieron ampliamente mejoradas con este nuevo sistema de preprocesado de la muestra utilizada.

3.2. Modelización que incluye variables exógenas

Introduzcamos ahora otra aproximación al problema de predicción.

La información empleada por el modelo semiparamétrico 2.3 consistía en el pasado de la serie en sí misma (incluída en el término Z_l). Es entonces cuando en el artículo de Prada-Sánchez, Febrero Bande et al. (2000) [13] decidieron que podría ser interesante añadir información adicional al modelo como podría ser la meteorología u otras variables recogidas por las estaciones de control de emisiones. Entre ellas podemos destacar la temperatura, la velocidad del viento, la radiación solar y la humedad.

Recordemos que inicialmente para hacer predicciones consideramos los pares (Z_l, Y_l) siendo el primer elemento del par una serie r-dimensional y el segundo una serie unidimensional. Ahora consideraremos tríadas (V_l, Z_l, Y_l) con $l = 0, \pm 1, \pm 2, \cdots$ donde V_l es un vector q-dimensional de variables explicativas exógenas (esto es cuyo valor está determinado por factores externos al modelo en el que se incluyen). Con esto, se propondrá el siguiente modelo parcialmente lineal

$$\boxed{Y_l = V_l^T \beta + \varphi(Z_l) + \varepsilon_l} \tag{3.1}$$

con β un vector de coeficientes q-dimensional desconocido y ε_l un término de error de media cero.

Con esta aproximación se consiguió extender el horizonte de predicción hasta 1 h.

Se consideraron 4 formas distintas de aproximar la estimación del modelo 3.1.

- La primera es un caso particular en el que $\varphi(Z_l) = Z_l^T \gamma$, esto es:

  $$Y_l = V_l^T \beta + Z_l^T \gamma + \varepsilon_l,$$

  siendo γ un vector r-dimensional de coeficientes. Para muestras de tamaño k, el modelo se escribe matricialmente como

  $$Y = V\beta + Z\gamma + \varepsilon,$$

  con Y y ε vectores k-dimensionales y V y Z matrices $k \times q$ y $k \times r$ respectivamente y cuyas filas i-ésimas son V_i^T y Z_i^T. La estimación en dos pasos por mínimos cuadrados de β y γ (que denotaremos por $\hat{\beta}_0$ y $\hat{\gamma}_0$ respectivamente) vienen dadas por

  $$\begin{cases} \hat{\beta}_0 = \big(V^T(I - P_z)V\big)^{-1} V^T(I - P_z)Y \\ Z\hat{\gamma}_0 = P_z(Y - V\hat{\beta}_0) \end{cases}$$

  donde I es la matriz identidad y $P_z = Z(Z^TZ)^{-1}Z^T$. Así pues, una predicción natural de Y_n usando (V_n, Z_n) con la base de la muestra $\{(V_i, Z_i, Y_i)\}_{i=1}^k$ es

  $$p_k^0(V_n, Z_n) = V_n^T \hat{\beta}_0 + Z_n^T \hat{\gamma}_0.$$

- La segunda es modificando ligeramente p_k^0. Para ello se sustituirá P_z por una estimación suavizada no paramétrica Z_H dada por $(Z_H)_{ij} = w_j^{H,k}\big(Z_i, (Z_1, \cdots, Z_k)\big)$, y estimando β como

  $$\hat{\beta}_1 = \big(V^T(I - Z_H V)\big)^{-1} V^T(I - Z_H)Y,$$

  prediciendo Y_n como

  $$p_k^1(V_n, Z_n) = V_n^T \hat{\beta}_1 + \sum_{j=1}^{k} w_j^{H,k}\big(Z_n, (Z_1, \cdots, Z_k)\big)(Y_j - V_j^T \hat{\beta}_1).$$

- La tercera aproximación fue restar $E(Y_l|Z_l)$ en cada lado de 3.1, lo que nos proporciona

$$Y_l - E(Y_l|Z_l) = (V_l - E(V_l|Z_l))^T\beta + \varepsilon_l,$$

lo que constituye un modelo de regresión lineal relativo a Y_l y V_l después de ajustar por sus valores esperados dado Z_l. Escribiendo $\tilde{Y} = (I - Z_H)Y$ y $\tilde{V} = (I - Z_H)V$, llegamos a la estimación de β

$$\hat{\beta}_2 = (\tilde{V}^T\tilde{V})^{-1}\tilde{V}^T\tilde{Y}$$

y el predictor

$$p_k^2(V_n, Z_n) = V_n^T\hat{\beta}_2 + \sum_{j=1}^{k} w_j^{H,k}\big(Z_n, (Z_1, \cdots, Z_k)\big)(Y_j - V_j^T\hat{\beta}_2).$$

- La cuarta aproximación fue suponer que V_l es independiente de Z_l, lo que nos deja la posibilidad de hacer regresión de Y_l sobre V_l después de ajustar Y por su valor esperado dado Z_l. Esto nos lleva a la estimación de β

$$\hat{\beta}_3 = (V^TV)^{-1}V^T\tilde{Y},$$

y por tanto el predictor

$$p_k^3(V_n, Z_n) = V_n^T\hat{\beta}_3 + \sum_{j=1}^{k} w_j^{H,k}\big(Z_n, (Z_1, \cdots, Z_k)\big)Y_j.$$

Se buscó predecir en 1 h vista, esto es, $Y_l = X_{l+12}$, empleando para ello $Z_l = (X_l, X_{l-3})$. Se decidió tomar Z_l como muestras de X separadas por 15 min; esta decisión se tomó basados en la experiencia, puesto que se vio que el tomar componentes adyacentes no era demasiado útil.

En todos los estimadores tenemos un término $\omega_j^{H,k}$, esto no es más que pesos generados vía kernels y H una matriz simétrica $r \times r$. Más concretamente, para el caso que nos compete éste, tiene la forma

$$\omega_j^{H,k}\big((X_n, X_{n-3}), (M_n)\big) = \frac{K\big(H^{-1/2}(X_n - X_j, X_{n-3} - X_{j-3})\big)}{\sum_{i\in I} K\big(H^{-1/2}(X_n - X_i, X_{n-3} - X_{i-3})\big)}$$

donde I es el número de filas de la matriz histórica M_n (el índice j denota una fila), K el núcleo gaussiano bidimensional $K(u) = (2\pi)^{-1}e^{-(1/2)u^Tu}$ con $u \in \mathbb{R}^2$ y $H = h^2S$, siendo S la matriz muestral de varianzas y covarianzas construidas con $\{(X_i, X_{i-3})\}_{i\in I}$ y h un parámetro de suavizado. Éste último se calculó empleando el conocido método *leave-one-out-cross-validation* (LOOCV)[1] :

$$\hat{h} = \min_h CV(h) = \min_h \sum_{i\in I}\big(Y_i - p_k^{j,(i)}(W_i)\big)^2$$

donde $p_k^j = \hat{\varphi}_k$, siendo p_k^1, p_k^2, p_k^3; $W_i = Z_i = (X_i, X_{i-3})$ para estimar $\hat{\varphi}_k$ y $W_i = (V_i, Z_i)$ para estimar p_k^1, p_k^2, p_k^3. El superíndice (i) en $p_k^{j,(i)}$ indica que la i-ésima fila de la matriz histórica es ignorada para construir el predictor correspondiente.

Es bien sabido LOOCV es un método computacionalmente exhaustivo, por ello, los autores pensaron que podría ser interesante calcular h dividiendo la muestra en dos trozos con conjuntos de índices de la muestra de entramiento I_1 e I_2 el índice de las muestra test. Así, se tomará h de forma que

$$\min_h TV(h) = \min_h \sum_{i\in I_2}\big(Y_i - p_{k/2}^{j,(I_2)}(V_i, Z_i)\big)^2$$

con $j = 1, 2, 3$ y $p_{k/2}^{j,(I_2)}$ construidos empleando la muestra dada por la matriz histórica.

Como comentarios finales destacamos lo siguiente:

[1] Las referencias sobre este método son numerosas. Por mencionar alguna (en concreto su séptimo capítulo): T. Hastie, R. Tibshirani, and J. Friedman, *The elements of statistical learning* (2001).

- p^1, p^2, p^3 deben ser corregidos añadiendo la predicción Box-Jenkins de la correspondiente parte residual $Y_n - p_k^j(V_n, Z_n)$ con $j = 1, 2, 3$.
- Suponiendo que el modelo parcialmente lineal es apropiado para modelizar el comportamiento de las emisiones de SO_2, p_k^1 y p_k^3 tuvieron el mejor desempeño durante los picos pero predecían que éstos se producirían más tarde de lo que en realidad sucedían.
- Las variables exógenas estaban muy poco correlacionadas con Z y por tanto eran poco útiles a la hora de intentar explicar Y.
- Otras variables que se suponían que mejorarían la predicción, en la época solo podrían ser medidas esporádicamente con los recursos disponibles.

3.3. Utilización de redes neuronales artificiales

Para el momento en que se desarrolló este modelo ya había sido efectuado un cambio en la directiva del concilio europeo que limitaba los valores de SO_2 y NO_x. Lo sustancialmente novedoso fue que el control se debería hacer sobre las medias horarias de los valores de SO_2, esto es, se trabajaría con las series

$$X_t = \frac{1}{12} \sum_{i=0}^{11} SO_2(t-i).$$

Además de esto los límites de valores máximos de SO_2 eran más estrictos que en la anterior normativa.

Al trabajar con estas nuevas medias, las series eran menos suaves por lo que la predicción se tornaba mucho más complicada de llevar a cabo.

Inicialmente se propuso usar los modelos a los que hemos hecho alusión en párrafos anteriores empleando las medias horarias de SO_2; sin embargo, las predicciones obtenidas poseían excesiva variabilidad (como podemos apreciar en la figura 3.3) por lo que se buscó idear un nuevo método de predicción.

Es aquí donde Fernández de Castro et al. (2003) en [4] consideraron que sería interesante aprovecharse de dos cualidades de las *redes neuronales*: su flexibilidad y capacidad de adaptación; por ello utilizaron una red neuronal *backpropagation* que posee la siguiente estructura que podemos apreciar en la siguiente imagen 3.1

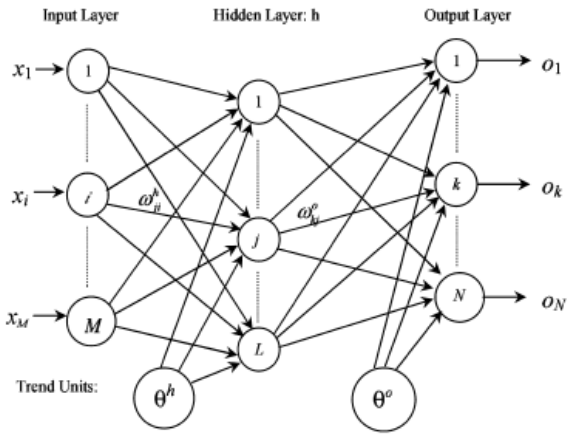


Figura 3.1: Red neuronal *backpropagation*. Imagen sacada del artículo [4].

Este modelo tiene una *capa de entrada* con M nodos, una *capa oculta* con L nodos y una *capa de salida* con N nodos.

Nuesto objetivo es que dados $X = (x_1, x_2, \cdots, x_M)^T$ un vector de datos e $Y = (y_1, y_2, \cdots, y_N)^T$ un vector respuesta definir unas conexiones entre los nodos de la red neuronal (lo que se conoce como *topología* de la red neuronal) de manera que el vector de outputs $O = (o_1, o_2, \cdots, o_N)$ reproduzca lo mejor posible Y.

Denotaremos por h los elementos de la capa oculta y por o los elementos de la cada de salida. Como se indica en el gráfico, definimos por ω_{ji}^h como el peso asociado al arco que va de la capa i inicial al nodo j de la capa oculta. f_k^o como la función de activación del nodo k en la capa de salida; θ_j^h es la tendencia del nodo j de la capa h; θ_k^o es la tendencia del nodo k de la capa de salida; o_j^h es la salida del nodo j de la capa h y o_k como la salida del nodo k de la capa de salida.

Así, la salida en cada nodo de la capa de salida puede ser escrita para cada $k = 1, \cdots, N$ como

$$o_k = f_k^o\left(\theta_k^o + \sum_{j=1}^{L} \omega_{kj}^o f_j^h\left(\theta_j^h + \sum_{i=1}^{M} \omega_{ji}^h x_i\right)\right).$$

Así pues el entrenamiento de la red neuronal consistirá en buscar los *pesos* asociados a las conexiones entre nodos de manera que se obtengan las mejores predicciones posibles minimizando algún criterio de error.

La red neuronal que se diseñó nos permitía obtener predicciones con antelación de media hora de la media horaria de los valores de SO_2, x_{t+6}. La respuesta en nuestro caso tiene dimensión 1, ya que estamos buscando una predicción puntual de x_{t+6}. El input de la red neuronal consiste en el vector bidimensional $\mathbf{X} = (x_{t-3}, x_t)^t$ que contiene para cada t los niveles de SO_2 15 min antes y el actual. Así pues la forma de nuestra red neuronal sería la que se aprecia en la imagen 3.2

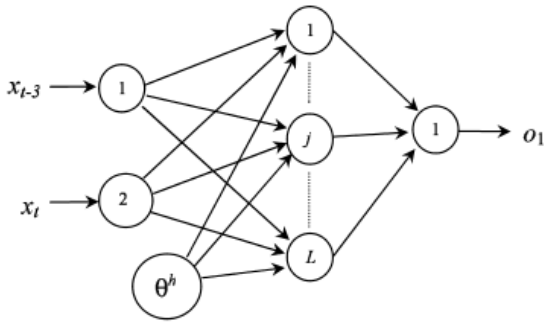


Figura 3.2: Topología de la red neuronal diseñada para la predicción de los niveles de SO_2. Figura sacada del artículo [4].

La función logística se empleó como función de activación en los nodos de la capa oculta y la identidad en la capa de salida. Así pues, el predictor dado por la red neuronal se puede escribir

$$\boxed{\hat{x}_{t+6} = o_1 = \sum_{j=1}^{L} \omega_{1j}^o f_j^h(\theta_j^h + \omega_{j1}^h x_{t-3} + \omega_{j2}^h x_t)}$$

con $f_j^h(z) = \frac{1}{1+e^{-z}}$ para cada j, ω_{j1}^h es el peso asociado al arco que va al nodo j de la capa h desde el nodo 1 de la capa anterior, análogamente ω_{j2}, mientras que ω_{1j}^o es el peso asociado al arco que va al nodo 1 de la capa de salida desde el nodo j de la capa anterior. Los pesos $\{\omega_{1j}, \omega_{2j}, \omega_{1j}^o, j = 1, \cdots, L\}$ y las tendencias $\{\theta_j^h, j = 1, \cdots, L\}$ se determinan durante el proceso de entrenamiento. Para éste, emplearemos la matriz histórica pero esta vez con vectores de la forma (x_{t-3}, x_t, x_{t+6}) procedentes de datos del 1999.

Para finalizar veamos en 3.3 un ejemplo de predicción.

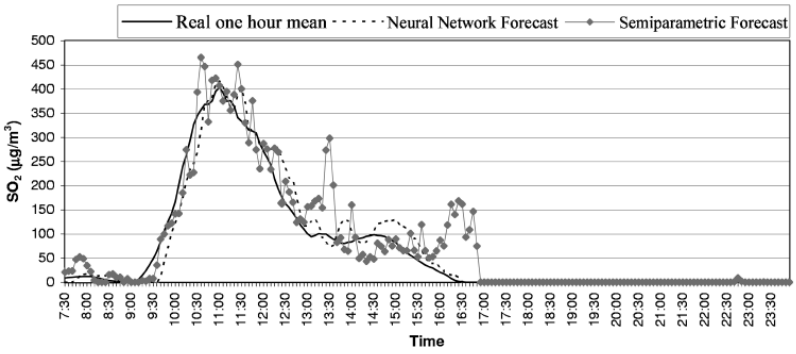


Figura 3.3: Red neuronal vs modelo semiparamétrico en la predicción del día 20-05-2000. Figura sacada del artículo [4].

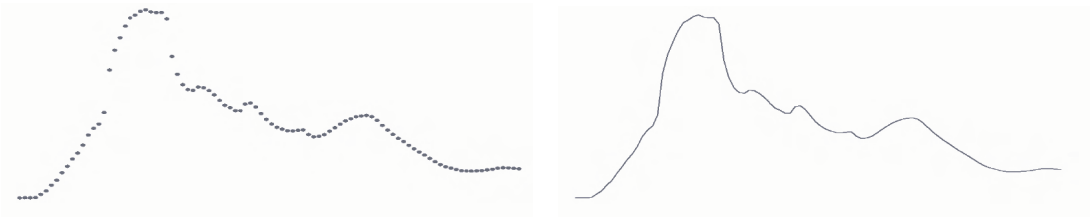
Cabe destacar como nota final que la red neuronal aprende de los patrones que infiere de los datos; por lo tanto la muestra de entrenamiento que se tome deben incluir algún ejemplo de lo que buscamos predecir. Por lo tanto, es necesario que en nuestro caso particular nos aseguremos que le ofrecemos datos que posean episodios de picos de niveles de SO_2 con el fin de que la red neuronal pueda *aprender* de ellos.

3.4. Aprovechamiento de técnicas del contexto de datos funcionales

Hasta ahora, hemos considerado modelos que toman *vectores* como datos para predecir, como se suele hacer habitualmente en estadística. Ahora bien, ¿qué pasa si tomamos como datos *curvas*? Es aquí donde nos introducimos en las técnicas propias de *datos funcionales*.

En el artículo [3] de Fernández de Castro et al. (2005) emplearán herramientas de esta área. Se empieza por considerar un proceso estocástico en tiempo continuo $x(t)$. Intentaremos hacer predicciones sobre futuros valores $\{x(u), u \geq T\}$ empleando la información contenida en el infinito número de variables del pasado $\{x(u), u \leq T\}$ considerando, a diferencia de los modelos anteriores, *porciones* de este proceso estocástico como *curvas*.

En nuestro caso particular queremos predecir los niveles de SO_2 en la próxima media hora. Nuestras muestras las tomaremos de esta longitud, esto es, de media hora. Cada una de ellas consistirá en seis datos (mediciones) consecutivos que trataremos como observaciones de un proceso estocástico en tiempo continuo que modeliza los niveles de SO_2. Gráficamente la idea se puede apreciar en 3.4



(a) Puntos muestrales. (b) Reconstrucción de las curvas muestrales.

Figura 3.4: Muestra funcional considerada. Imagen sacada de [7].

No impondremos restricciones de suavidad para estas curvas puesto que lo que buscamos es precisamente predecir estos rápidos incrementos o descensos.

Nos restringiremos a analizar la dependencia en un único retardo puesto que nuestras muestras están siendo tomadas cada 5 min (cada curva está siendo muestreada en 6 puntos distintos) lo que parece suficiente como para recoger la evolución del proceso. Así pues consideraremos variables aleatorias en $H = L^2([0,6])$ (recordemos que L^2 es un espacio de Hilbert[2]) de la forma siguiente

$$X_n(u) = x(6n+u), u \in [0,6], n \in \mathbb{N}^*.$$

El modelo a considerar es el siguiente:

$$\boxed{X_n = \rho(X_{n-1}) + \varepsilon_n}$$

siendo ε_n ruido blanco fuerte hilbertiano, esto es, una sucesión de variables i.i.d con valores en H que satisfacen con $n \in \mathbb{Z}$ que $E\varepsilon_n = 0$ y que $0 < E||\varepsilon_n||_H^2 = \sigma^2 < \infty$ y $\rho : H \to H$ una función a ser estimada.

Como parece lógico, debemos modificar el mecanismo de matriz histórica para adaptarlo a este caso. Para ello, se consideraron (empleando datos del 2001) vectores de la forma (X_n, X_{n+1}) donde cada X_n está compuesto por seis medidas de los niveles de SO_2 (para más detalles de cómo se construyó puede verse el artículo [3]).

En este paper se proponen dos métodos de estimación.

- El primero (llamado "lineal") de ellos es suponer que ρ es un operador lineal acotado en H. Con hipótesis no demasiado exigentes se probó que este modelo autorregresivo tiene una única solución estacionaria.

  La estimación del operador ρ se apoya en la relación $D = \rho \cdot C$, donde D es el operador dado por $D(x) = E(\langle X_0, x\rangle X_1)$ y $C(x) = E(\langle X_0, x\rangle X_0)$. Lo habitual es que C no sea invertible, es por ello que se buscará una proyección en un espacio de dimensión finita k_n como estimación de ρ. Este k_n está relacionado con la tasa de decrecimiento de los autovalores de C.

  El ***algoritmo*** de estimación tiene los siguientes pasos:

  1. Calcular, empleando componentes principales, los estimadores empíricos de los autovalores y autovectores del operador de covarianzas C asociado a X_n.
  2. Proyectar $D = \rho \cdot C$ en el subespacio formado por los primeros k_n autovectores asociados a los k_n autovalores más grandes de C.
  3. Obtener un estimador ρ_n de ρ usando la relación proyectada siempre que sea posible (por cuestiones de invertibilidad como comentamos anteriormente).

- El segundo de ellos emplea una extensión del clásico Nadaraya-Watson al contexto de datos funcionales. En él, ρ puede ser estimado con el estimador tipo núcleo funcional usando los n vectores funcionales (X_i, X_{i+1}) de la matriz histórica modificada $\{(X_{i_1}, X_{i_1+1}), (X_{i_2}, X_{i_2+1}), \cdots, (X_{i_N}, X_{i_N+1})\}$

  $$\hat{\rho}_{h_N}(x) = \frac{\sum_{j=1}^{N} X_{i_j+1} \cdot K(||X_{i_j} - x||/h_N)}{\sum_{j=1}^{N} K(||X_{i_j} - x||/h_N)}$$

  donde K denota una función tipo núcleo, N es el tamaño muestral y h_N (calculado vía validación cruzada) es la ventana y x pertenece a H.

Dado que estamos trabajando con objetos que son funciones, debemos adaptar las medidas de error a este contexto. Denotemos por $\hat{X}$ la predicción para la variable aleatoria X. Se emplearán como

[2]Para este tipo de espacios matemáticos, necesitamos saber un poco de Teoría de la Medida y Análisis Funcional. Como referencias, respectivamente citamos: Halmos, P. R. (1950), *Measure Theory* y Kreyszig, E. (1989), *Introductory Functional Analysis with Applications.*

medidas de error los errores empíricos en L^p para $p = 1, 2$ en una muestra de n medias horas (o seis datos de 5 min) siguientes:

$$||\hat{X} - X||_{L^p} = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{1}{6}\sum_{j=1}^{6}|\hat{X}_i^j - X_i^j|^p\right)^{1/p}.$$

También se usa el error en L^∞, calculado como

$$||\hat{X} - X||_{L^\infty} = \frac{1}{n}\sum_{i=1}^{n}\sup_{j=1,\cdots,6}|\hat{X}_i^j - X_i^j|.$$

3.4.1. Construcción de regiones de confianza vía bootstrap

Sería interesante obtener regiones de confianza para nuestras predicciones. Para ello, se emplearán técnicas bootstrap adaptadas a este contexto funcional.

Varios de los autores más importantes en el contexto de datos funcionales como pueden ser Fraiman y Muñiz (2001) o Ramsay y Silverman (1997) descubrieron que emplear técnicas finito-dimensionales a este tipo de datos no es adecuado.

En particular, tenemos que adaptar a este contexto las medidas de *centralización*. Para ello proponen retomar la noción de *profundidad* univariante. Recordemos que si F es la función de distsribución de una cierta variable aleatoria y x un dato, definimos D como

$$D(x) = 1 - |0{,}5 - F(x)|.$$

La profundidad mide la cercanía a la mediana *med*. Es inmediato ver que en la mediana se alcanza el valor máximo.

Consideremos ahora un conjunto de datos funcionales $X(t)$, con $t \in [a, b]$ y una muestra $X_1(t), \cdots, X_p(t)$ que sigue la misma distribución que $X(t)$. Para cada punto t, calcularemos la distribución empírica

$$F_{p,t}(x) = \frac{1}{p}\sum_{j=1}^{p}\mathbb{I}_{X_j(t)\leq x}.$$

Llamemos D_p la profundidad empírica univariante. Entonces

$$D_{p,t}(x) = 1 - |0{,}5 - F_{p,t}(x)|.$$

Fraiman y Muniz consideraron analizar el *índice integrado*

$$I_i = \int_a^b D_{p,t}(X_i(t))dt.$$

Éste mide globalmente la cercanía a la mediana empírica (podemos pensar ésta como la curva para la que el índice es máximo).

Con esta medida, podemos entonces ordenar las curvas según su cercanía a la curva mediana (y tendríamos entonces una idea de cómo están distribuidas las curvas con respecto a su *centro*).

Comentado esto, podemos proceder a ver el primer método. Éste supone que las series de tiempo constituyen un proceso de Markov. Para obtener p estimaciones bootstrap una etapa por delante $Y^*_{m+1,1}, \cdots, Y^*_{m+1,p}$ en el punto Y_m, los pasos serían los siguientes:

1. Con la matriz histórica construir bloques muestrales de longitud dos

$$B_j = \{X_j, X_{j+1}\},\ j = 1, \cdots, N$$

2. Calcular las probabilidades siguientes

$$\hat{p}_j = \frac{K(||X_j - Y_m||/h)}{\sum_{i=1}^{N} K(||X_i - Y_m||/h)}$$

con h la ventana óptima (bien local o global).

3. Aleatoriamente elegir p bloques $\{Y^*_{m,i}, Y^*_{m+1,i}\}$ con probabilidad $\hat{p}_j$ de elegir $\{Y^*_{m,i}, Y^*_{m+1,i}\} = \{X_j, X_{j+1}\}$ y extraer el segundo elemento $Y^*_{m+1,i}$, obteniendo así una sucesión con las posibles réplicas $Y^*_{m+1,1}, \cdots, Y^*_{m+1,p}$.

4. Ordenar las réplicas bootstrap en orden descendente respecto del índice I_i, obteniendo así los estadísticos de orden

$$Y^*_{m+1,1:1}, \cdots, Y^*_{m+1,p:p}$$

El segundo método, supone que el modelo tiene exactamente la forma que comentamos en el inicio de la sección. Así, se propondrá el siguiente algoritmo bootstrap (que podemos pensar como una generalización del bootstrap residual)

1. Calcular los residuos hacia adelante $i = 2, \cdots, n+1$ y su versión corregida

$$\hat{a}_i = X_i - \hat{\rho} X_{i-1},\ \hat{a}'_i = \hat{a}_i - \frac{1}{n}\sum_{i=2}^{n+1} \hat{a}_i.$$

2. Calcular las componentes principales de $\hat{a}'_i$

$$\hat{a}'_i = c^i_1 V_1 + \cdots + c^i_{kn} V_{k_n}.$$

3. Para cada coordenada c_l calcular su distribución empírica $F^{c_l}_n$ con $l = 1, \cdots, k_n$.

4. Generar c^*_l con funciones de distribución $F^{c_l}_n$, $l = 1, \cdots, k_n$, y construir los residuos bootstrap

$$\hat{a}^*_i = c^*_1 V_1 + \cdots + c^*_{kn} V_{k_n}.$$

5. Generar las réplicas bootstrap

$$Y^*_{m+1,i} = \hat{\rho} Y_m + \hat{a}^*_i.$$

6. Ordenar las réplicas bootstrap en orden decreciente según el índice I_i.

En ambos casos cabe destacar que al final de los procedimientos ordenamos según el índice integrado. Esto se hace para obtener una región con las curvas límite que encierran nuestras predicciones.

El estudio concluye con la comparación con los modelos anteriores (tanto de redes neuronales como del modelo semiparamétrico). Se vio que el modelo de datos funcionales, en general, era menos efectivo a la hora de llevar a cabo las predicciones.

Se deja propuesto que sería interesante trabajar con datos minutales (lo que permitiría interpolar la curva de niveles de SO_2 en más puntos y así tener una información más detallada del comportamiento de las curvas). Asimismo, el integrar variables meteorológicas (como ya se pensó en el modelo semiparamétrico), si bien no se pueden medir con exactitud se pueden aproximar numéricamente modelos de ecuaciones en derivadas parciales para su predicción, que, mezclados con los modelos estadísticos propuestos, serían muy potentes para realizar las previsiones deseadas.

No podemos acabar el capítulo sin recalcar que quedan varios artículos presentes en la literatura sin mencionar, fruto de las prolíficas colaboraciones entre la universidad y ENDESA. Por ejemplo, otro enfoque que no hemos abordado es intentar predecir probabilidades de ocurrencia de los episodios de picos de concentración de niveles de SO_2, como se puede ver en los artículos de Roca Pardiñas et al. (2004) [16] y Roca Pardiñas et al. (2005) [15] empleando técnicas de modelos aditivos generalizados.

Capítulo 4

Modelos de regresión cuantil

Hasta ahora, nosotros nos hemos preocupado de la revisión de aquellos modelos de predicción para niveles de SO_2. Ahora bien, recordemos que en el año 2008, se implantó un nuevo sistema de ciclo combinado que tiene como elemento traza contaminante el NO_x. En esta dirección se mueve el trabajo [2] de Mercedes Amboage et al. (2016).

Uno de los motivos principales para elegir el modelo de regresión cuantil es que es robusto, mientras que la regresión en media se puede ver afectada por datos atípicos presentes en la muestra con la que se trabaje (como podemos ver en el caso del 20 de octubre de 2011 en la figura 4.1)

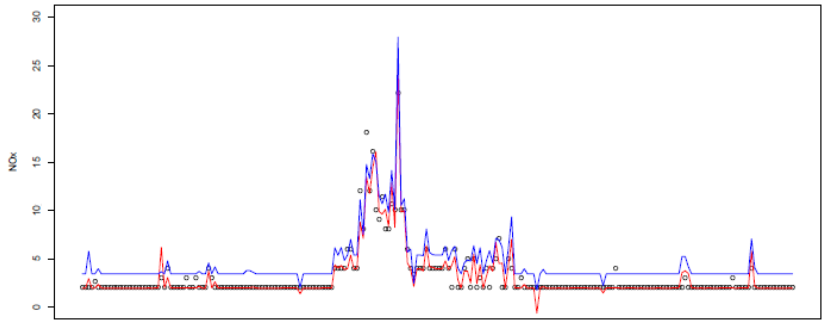


Figura 4.1: Regresión en media vs regresión en mediana. Imagen sacada de [7].

El primer cambio sustancial es que la concentración de los niveles de NO_x son medidos en esta ocasión minutalmente. Además, la estación meteorológica local recogerá (también de forma minutal) tres variables que incluiremos en nuestro modelo: temperatura, velocidad y dirección del viento (esta última se considerará escalar puesto que lo que se hizo en su día fue medir el valor absoluto del ángulo de desviación respecto al norte). Lo que se buscará es predecir la concentración de NO_x en tiempo $(t+30)$ basados en la información disponible en tiempo t (donde t y $t+30$ están medidos en minutos). Se considerará el siguiente modelo de regresión:

$$\boxed{Y_t = \beta_0 + \beta_1 X_t + \beta_2 X_{t,grad} + \beta_3 Z_{1t} + \beta_4 Z_{2t} + \beta_5 Z_{3t} + \varepsilon_t} \tag{4.1}$$

donde X_t es la concentración de NO_x en tiempo t, $X_{t,grad} = X_t - X_{t-5}$ representa el gradiente de la concentración de NO_x en el intervalo de los últimos 5 minutos y ε_t representa el error. Denotemos por $P_t = (1, X_t, X_{t,grad}, Z_{1t}, Z_{2t}, Z_{3t})'$ el vector de predictores y $\beta = (\beta_0, \beta_1, \beta_2, \beta_3, \beta_4, \beta_5)'$ el vector de coeficientes a estimar.

En esta ocasión se construirán *intervalos de predicción* para los valores deseados.

Antes de continuar debemos puntualizar un detalle teórico. Recordemos que un intervalo de predicción para un valor Y_{t_0} es un intervalo que se espera que contenga al valor verdadero Y_{t_0} con una alta probabilidad $(1-\alpha)$. Llamamos cobertura *incondicional* a

$$P\big(Y_{t_0} \in (L_{t_0}, U_{t_0})\big) = 1-\alpha$$

habiendo obtenido los extremos inferior y superior del intervalo como funciones de la muestra de entrenamiento y los valores de los predictores P_{t_0} en tiempo t_0. Esta probabilidad está definida para *todas* las posibles muestras de entrenamiento y nuevas observaciones. Sin embargo, en la práctica habitualmente lo que tenemos es un valor determinado para P_{t_0}, por lo que podemos definir la cobertura *condicional*

$$P\big(Y_{t_0} \in (L_{t_0}, U_{t_0}) | P_{t_0} = p_{t_0}\big).$$

Es inmediato darse cuenta que si la cobertura condicional tiene nivel $(1-\alpha)$ entonces la cobertura incondicional también. Por este motivo, garantizar la cobertura condicional es más fuerte (ya que ésta implica la otra), por lo que tendremos que explotar al máximo la información proveniente de P_{t_0}.

Volvamos al modelo dado por la ecuación 4.2. Si en ella considerásemos que $E(\varepsilon_t) = 0$ estaríamos ante el caso habitual de estimación mediante la minimización de los residuos al cuadrado:

$$\hat{\beta}_{LS} = arg\,\min_{\beta} \sum_t \big(Y_t - \beta' P_t\big)^2.$$

Bajo este contexto el intervalo de confianza para Y_t es

$$\left(\hat{Y}_{t_0,LS} - t_{n-6,\alpha/2}\hat{\sigma}\sqrt{1+P'_{t_0}(X'X)^{-1}P_{t_0}}, \hat{Y}_{t_0,LS} + t_{n-6,\alpha/2}\hat{\sigma}\sqrt{1+P'_{t_0}(X'X)^{-1}P_{t_0}}\right)$$

donde $t_{n-6,\alpha/2}$ es el cuantil $(1-\alpha/2)$ de la distribución t-Student con $n-6$ grados de libertad, X es la matriz de diseño de la muestra de entrenamiento y

$$\hat{\sigma}^2 = \frac{1}{n-6}\sum_t \big(Y_t - \hat{\beta}'_{LS}P_t\big)^2$$

es la varianza del error estimada con ésta de entrenamiento. Este método demanda fuertemente de dos hipótesis para trabajar con él: homocedasticidad y normalidad de los errores.

Con el fin de ser más laxos con respecto a estas hipótesis, los autores consideraron $P(\varepsilon_t \leq 0) = \tau$ con $\tau \in (0,1)$. Básicamente esto impone que la proporción de errores no positivos se espera que sea τ; pensado de otra forma es equivalente a que la proporción de observaciones que estén por debajo de la función de regresión sea igual a τ. Planteado de esta forma, la función de regresión es el *cuantil condicional* τ de la variable respuesta. Esto nos permitirá *focalizar* nuestra predicción en bien valores altos o bajos de la variable respuesta según hagamos la elección de τ.

Con el fin de detallar el problema de regresión cuantil necesitamos considerar $\tau \in (0,1)$ y ρ_τ la función de pérdida cuantil definida por

$$\rho_\tau(z) = \begin{cases} (1-\tau)\cdot|z|, & z \leq 0 \\ \tau \cdot z, & z > 0. \end{cases}$$

La estimación del modelo 4.1 con la regresión cuantil τ viene dada por

$$\hat{\beta}(\tau) = arg\,\min_{\beta} \sum_t \rho_\tau(Y_t - \beta' P_t). \tag{4.2}$$

Antes de seguir, parémonos a observar un momento la forma que tiene la función ρ_τ. Para ello la representaremos para distintos τ.

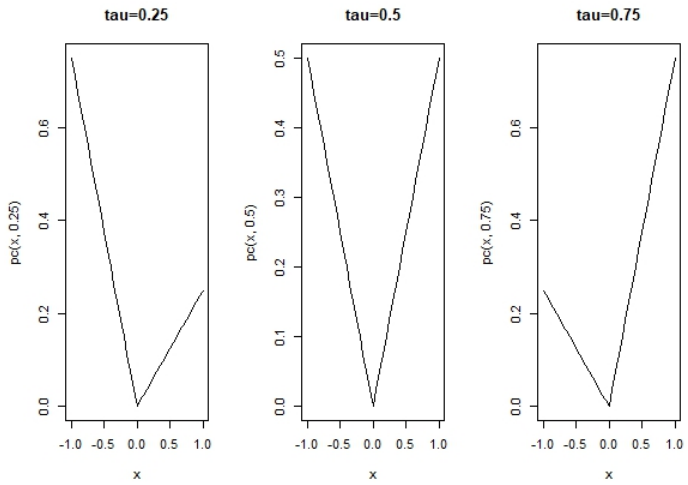


Figura 4.2: Representación de la función de pérdida cuantílica en el intervalo $[-1, 1]$ para $\tau = \{0{,}25, 0{,}5, 0{,}75\}$

En el caso $\tau = 0{,}25$ podemos observar que tiende a penalizar las observaciones inferiores a la función de regresión. Nuestras predicciones dadas por 4.2 en este sentido están *obligadas* a pasar por éstas puesto que cometer errores en ellas hará que la función objetivo del problema de minimización se vea incrementado. Por el contrario $\tau = 0{,}75$ penaliza aquellas observaciones superiores a la función de regresión y por tanto ésta tenderá a pasar por valores superiores a la función de regresión (por el mismo motivo que en el caso $\tau = 0{,}25$).

En esta ocasión no se empleó la matriz histórica para la estimación del modelo si no que se usaron observaciones que cubren un período de 10 días (como datos de entrenamiento) y los 10 días siguientes a éstos para evaluar las predicciones obtenidas.

Pasamos ahora a comentar brevemente algunas de las técnicas de estimación que se propusieron en este artículo.

- La primera consiste en tomar como intervalo de predicción para Y_{t_0} de nivel $1 - \alpha$

$$(\hat{Y}_{t_0,\tau=\alpha/2}, \hat{Y}_{t_0,\tau=1-\alpha/2}),$$

  siendo los extremos las estimaciones de los cuantiles $1 - \alpha/2$ y $\alpha/2$ de Y_{t_0} condicionados a los valores de los predictores P_{t_0}. Cabe destacar que este intervalo no requiere homocedasticidad y es adaptable a cualquier distribución de error. Presenta un problema y es que la cobertura empírica es más pequeña que la cobertura nominal.

- La segunda es prácticamente igual a la inicialmente comentada solo que esta vez hace una pequeña corrección al intervalo:

$$(\hat{Y}_{t_0,\tau=\alpha/2-\delta}, \hat{Y}_{t_0,\tau=1-\alpha/2+\delta}),$$

  siendo $\delta = 0{,}5(z_{1-\alpha/2}/n)$, siendo $z_{1-\alpha/2}$ el cuantil $1 - \alpha/2$ de la distribución normal estándar y n el tamaño de la muestra de entrenamiento.

4.1. Técnicas bootstrap en regresión cuantil

Con la idea de proponer un intervalo de predicción que mantenga un compromiso entre las dos definiciones de cobertura que hemos visto, Mercedes Amboage et al. (2016) combinarán técnicas de regresión cuantil con un método de remuestreo.

Pasamos a detallar el algoritmo empleado.

1. Calcular las réplicas bootstrap tanto de la muestra de entramiento como de la nueva observación, esto es
$$\begin{cases} Y_t^\star = \hat{\beta}(\tau = 0{,}5)' P_t + \varepsilon_t^\star \quad t \in \{1, \cdots, n\} \\ Y_{t_0}^\star = \hat{\beta}(\tau = 0{,}5)' P_{t_0} + \varepsilon_{t_0}^\star. \end{cases}$$
En esta expresión hay varios elementos que detallaremos cómo se obtienen.
   - $\hat{\beta}(\tau = 0{,}5)$ es una estimación de los coeficientes en la regresión en mediana obtenida con la muestra de entrenamiento.
   - $\varepsilon_t^\star = w_t |r_t|$ donde $|\cdot|$ denota el valor absoluto y $r_t = Y_t - \hat{\beta}(\tau = 0{,}5)' P_t$ (residuos en la muestra de entrenamiento original).
   - w_t son pesos que siguen la distribución discreta con valores 1 y -1 con probabilidades 0.5 respectivamente.
   - El error bootstrap para una nueva observación viene dado por $\varepsilon_{t_0}^\star = w_{t_0} |r_{t_0}|$, donde w_{t_0} sigue la misma distribución de w_t y el residuo r_{t_0} sigue la distribución
$$\hat{F}(r | P_{t_0}) = \sum_{t=1}^{n} \mathbb{I}(r_t \leq r) W_{t, P_{t_0}}$$
siendo $\mathbb{I}$ la función indicadora y
$$W_{t, P_{t_0}} = \frac{K\big((\hat{\beta}(\tau = 0{,}5)' P_t - \hat{\beta}(\tau = 0{,}5)' P_{t_0})/h\big)}{\sum_{s=1}^{n} K\big((\hat{\beta}(\tau = 0{,}5)' P_s - \hat{\beta}(\tau = 0{,}5)' P_{t_0})/h\big)}$$
siendo $h = cn^{-1/5}$ con c una constante que depende de cantidades desconocidas (en el artículo tras diversas discusiones se tomó igual a 1).

2. Basado en la muestra bootstrap recién construida se puede obtener una réplica bootsrap de los coeficientes que será denotada por $\hat{\beta}^\star(\tau = 0{,}5)$. Así podemos calcular los errores de predicción bootstrap
$$D^\star = Y_{t_0}^\star - \hat{\beta}^\star(\tau = 0{,}5)' P_{t_0}.$$

3. Repetir los pasos 1 y 2 para obtener $\{D_i^\star\}_{i=1}^B$. La distribución empírica de esta muestra es una aproximación Monte Carlo de la función de distribución de $G^\star$ (distribución bootstrap de los errores de predicción), de la que calcularemos sus sus cuantiles $G^{\star -1}(\alpha/2)$ y $G^{\star -1}(1 - \alpha/2)$.

Con todo esto, estamos en condiciones de construir el intervalo de predicción deseado
$$\big(\hat{Y}_{t_0, \tau = 0{,}5} + G^{\star -1}(\alpha/2), \hat{Y}_{t_0, \tau = 0{,}5} + G^{\star -1}(1 - \alpha/2)\big),$$

Una vez evaluado el modelo en la muestra test, en general, el tercer método es el que ofrece mejores resultados. Para más detalles ver [2].

Capítulo 5

Otros modelos existentes en la literatura

En este apartado mencionaremos algunos problemas planteamientos con objetivos similares a los del capítulo segundo de esta memoria, pero tratados desde puntos de vista diferentes empleando otras herramientas conceptuales y metodológicas.

5.1. Modelo de redes neuronales con parámetros espaciales

Los modelos propuestos en el artículo de Atakan Kurt et al. (2010) [1] buscan predecir SO_2, CO y material particulado en los próximos 3 días (desde el día en que se busca llevar a cabo la aproximación) en Besiktas.

Para ello, emplearán datos (tomados desde el 2005, entre los que se incluyen temperatura durante el día, temperatura durante la noche, humedad, velocidad del viento, dirección del viento, presión, día de la semana, fecha y niveles de los 3 contaminantes mencionados inicialmente) procedentes de 10 estaciones de control de calidad del aire repartidas a lo largo de diferentes distritos en Estambul.

En particular, el modelo que comentaremos se trata de una red neuronal con la particularidad de que incluye *parámetros espaciales* vía modelización geográfica. Los autores justifican que, al menos de forma heurística, el rendimiento debiera ser mejor puesto que el proceso de difusión de los contaminantes a predecir dependen, además de las condiciones meteorológicas, de:

- Localización geográfica a través de la que se mueven.
- Localización geográfica de los puntos de recogida de datos.
- Proximidad entre los diversos puntos de recogida de datos (a menor distancia entre los mismos se espera que haya un comportamiento de los contaminantes similar).

En el artículo, se explican y emplean para predicción 3 modelos con estas características. Específicamente, detallaremos brevemente el *basado en distancias*, que se llama así porque tiene en cuenta las distancias entre los distritos en los que se registran los datos. Los autores basan su estudio en la idea de que los niveles de los contaminantes de aire por distrito son inversamente proporcionales a la distancia entre dos distritos.

Los que se emplearon fueron Fatih, Uskudar, Sariyer, Kartal y Yenibosna. Cabe destacar que estos sitios no se eligieron al azar: se buscó que las tríadas de distritos formen entre sí figuras lo más parecidas posibles a triángulos (en concreto tendrían que ser equiláteros y el distrito central debe estar situado lo más al centro del triángulo posible) que encierren la zona donde que se quiere obtener las predicciones (en nuestro caso Besiktas). Los vértices de los posibles triángulos se obtienen vía experimentación bajo esta premisa.

Conocidas las distancias entre el distrito a predecir y los vértices se puede obtener una media ponderada de los contaminantes. Para el caso de buscar predecir SO_2 y empleando los distritos de Fatih, Uskudar y Sariyer

$$newSO_2 = SO_2Fatih * 0{,}32 + SO_2Uskudar * 0{,}45 + SO_2Sariyer * 0{,}23,$$

donde cada coeficiente representa las distancias normalizadas entre Besiktas y el distrito en concreto de Estambul. Esta nueva variable será empleada como variable explicativa (además de las mencionadas inicialmente).

Este estudio diseñó también una red neuronal que no incluía parámetros espaciales y se comprobó que fue ampliamente superada por aquellas que sí incluían esta información, en particular, por el modelo basado en distancias.

Se concluye diciendo que la red neuronal puede ser adaptada para la predicción de otros contaminantes siempre y cuando se elijan de manera adecuada las ciudades próximas entre sí.

5.2. Modelización físico-estadística

Hemos titulado este apartado así puesto que el artículo de Youngdeok Hwang et al. (2018) [8] propone una estimación de los niveles de contaminantes mediante un interesante punto de vista que mezcla herramientas de la Teoría de fluídos como de la Programación matemática. Lo ingenioso de este artículo es reformular el modelo físico que describe el proceso de emisión de contaminantes como un problema de regresión.

La contaminación del aire se produce por emisiones transportadas por procesos físicos dirigidos por el viento. La calidad de la predicción del modelo físico (que describe estos procesos) depende casi en su totalidad de la precisión de la información que se recibe. Así, matemáticamente hablando, estamos ante un *problema inverso* puesto que queremos estimar los parámetros de un modelo teórico del que solo sabemos sus *output* y los datos observados.

Pasamos ahora a explicar brevemente el modelo físico que describe el proceso de emisión de contaminantes. Dentro de la Física, para el caso que nos compete se puede emplear para el modelaje deseado lo que se conoce como procesos de *dispersión*. Éste está formado por *advección* (que es la transferencia de contaminantes de un lugar a otro) y *difusión* (que caracteriza el movimiento de contaminantes de una región de alta concentración a una de baja debido a la turbulencia atmosférica; se puede ver un ejemplo en 5.1).

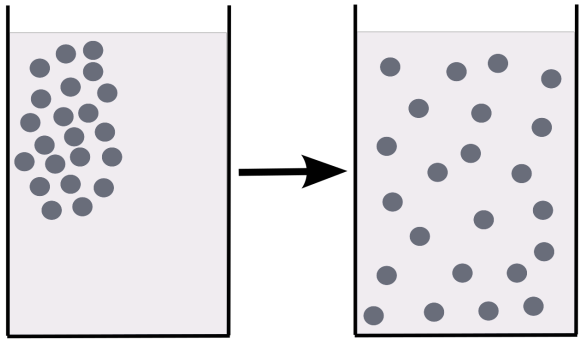


Figura 5.1: Proceso de difusión

Denotemos $\phi(s,t)$ la concentración del contaminante en el lugar s y tiempo t. El proceso de dispersión se puede expresar como

$$\frac{\partial \phi(s,t)}{\partial t} = -\nabla \cdot (u(s,t)\phi(s,t)) + \nabla \cdot \{K(s,t;u) \cdot \nabla \phi(s,t)\} + Q(s,t), \tag{5.1}$$

lo que implica que el cambio temporal de la concentración de contaminante en un lugar está determinado por la advección (primer término de la suma) y la difusión (segundo término). $u(s,t)$ es la velocidad del viento (que podemos obtener mediante modelos de predicción del tiempo), K es una matriz de coeficientes de difusión (que define la tasa de mezcla del contaminante) y el término $Q(s,t)$ representa la tasa de contaminante añadido en el lugar s en tiempo t.

El siguiente paso es establecer una *rejilla* en la que dividir el espacio para poder aplicar técnicas numéricas (como podría ser el método de diferencias finitas). Una vez hecho esto y empleando el modelo 5.1, denotemos por $X_{t,ij}$ como la dispersión calculada usando dicho modelo en la localización de la monitorización i por s_i, en tiempo t, que sale de la fuente j. La concentración de contaminación observada en s_i en tiempo t y el ruido asociado a la medición se denotan por $y_{t,i}$ y $\varepsilon_{t,i}$ respectivamente para $i = 1, \cdots, n$ y $t = 1, \cdots, T$.

Denotando la intensidad emisión de la fuente j por β_j y asumiendo que hay p fuentes, podemos proponer el siguiente modelo

$$y_{t,i} = \sum_{j=1}^{p} X_{t,ij}\beta_{j,h(t)} + \varepsilon_{t,i}, \tag{5.2}$$

donde $h(t) = (t \mod 24) + 1$.

Denotemos por β la matriz $p \times 24$ de intensidad de emisión. La gran mayoría de localizaciones tendrán un efecto negligente en la contribución de emisión de contaminantes. Esto nos lleva a ver que varios $\beta(j)$ debieran ser 0 debido a lo que acabamos de comentar, por lo que β seguramente sea una matriz tipo *sparse*. Por tanto, podría ser interesante emplear una penalización tipo lasso. El autor propone también que las filas de β probablemente sean dependientes y que su rango sea pequeño. Para que esto se cumpla es necesario añadir una penalización en *norma nuclear* (este tipo de penalización controla las posibles dependencias lineales entre las filas de la matriz que tomamos en consideración).

Antes de escribir el problema de minimización cabe destacar que supondremos tres cosas:

- Las fuentes de las que emanan las emisiones solo pueden *añadir* contaminantes al aire.
- Cualquier *decaimiento* o *deposito* de contaminante se considerá despreciable y por tanto será absorbido por el término de error.
- Todos los *coeficientes* de emisión son *no negativos* (como consecuencia directa de las dos primeras suposiciones).

El problema entonces, con todas estas hipótesis se puede escribir como

$$\boxed{\min_{\beta_{jk} \geq 0} \frac{1}{nT} \sum_{t=1}^{T} \sum_{i=1}^{n} \left(y_{t,i} - \sum_{j=1}^{p} X_{t,ij}\beta_{j,h(t)} \right)^2 + \lambda_{gl} \sum_{j=1}^{p} ||\beta(j)||_2 + \lambda_{nuc}||\beta||_*} \tag{5.3}$$

donde $\lambda_{gl} \geq 0$ y $\lambda_{nuc} \geq 0$ son parámetros correspondientes respectivamente a la penalización lasso y a la penalización nuclear (notar que este último parámetro mientras más grande sea, la minimización fuerza a que haya una mayor dependencia lineal).

La función objetivo es la suma de funciones convexas, por lo que es convexa. La restricción de no negatividad es también convexa, por lo que se trata de un problema convexo.

El proceso de estimación lo dejamos para el lector interesado ([8], en el apartado 4).

5.3. Modelo geoestadístico

Comentaremos brevemente ahora el uso de técnicas de *estadística espacial* para la predicción de contaminantes. En el artículo de Menezes et al. (2016) [14] se aplicarán herramientas de esta área con el fin de caracterizar la evolución de los niveles de NO_2 en Portugal desde el año 2004 hasta el 2012.

Cabe mencionar que en este caso (con los datos de Portugal) no se disponían sobre el 30 % de las observaciones diarias, por lo que desarrollar métodos de predicción en este aspecto (con el fin de cumplir con las normativas europeas fijadas) es de vital importancia.

Para fijar el contexto matemático donde nos moveremos consideremos una función aleatoria unidimensional $\{Z(s,t) : (s,t) \in \mathbb{R}^d \times \mathbb{R}\}$ donde $s \in \mathbb{R}^d$ es la coordenada *espacial* y $t \in \mathbb{R}$ la *temporal*.

Para este proceso podemos definir respectivamente su función media y covarianza

$$\mu(s,t) = E\big(Z(s,t)\big)$$

$$C\big((s_1,t_1),(s_2,t_2)\big) = Cov\big(Z(s_1,t_1), Z(s_2,t_2)\big)$$

para cualesquiera $(s_1,t_1),(s_2,t_2) \in \mathbb{R}^d \times \mathbb{R}$ y siempre que $Var\big(Z(s,t)\big) < \infty$ para todo $(s,t) \in \mathbb{R}^d \times \mathbb{R}$.

El proceso por lo tanto estará completamente caracterizado conocida para todo z y para todo $(s,t) \in \mathbb{R}^d \times \mathbb{R}$

$$F(s,t;z) = P\big(Z(s,t) \leq z\big)$$

Para poder aplicar técnicas espacio temporales se necesita asumir que la variable objeto de estudio es conocida en cada lugar s y tiempo t. En la práctica, por el contrario, es habitual solamente contar con no más de una realización del proceso aleatorio $Z(s,t)$, por lo que debemos suponer hipótesis sobre los datos que nos permitan hacer inferencia.

Cuando trabajamos con procesos espacio-temporales se suele suponer que el proceso se puede descomponer en tendencia o variación gran escala y residuo (estacionario de media cero) o variación a pequeña escala , esto es,

$$Z(s,t) = \mu(s,t) + \varepsilon(s,t).$$

Lo habitual es caracterizar de forma secuencial ambos sumandos, primeramente la tendencia y posteriormente el residuo. Veamos cómo se estiman éstos.

- Tendencia. El primero, en este trabajo, se estimará mediante un modelo lineal generalizado, para lo que tendremos que relajar la hipótesis de errores no correlados (lo cual nos permitirá obtener una estimación puntual de los parámetros de regresión, mediante estimación por máxima verosimilitud o mediante un criterio de Akaike).

  Concretamente

  1. Especificar la distribución condicional de la variable respuesta $Z(s,t)$ dados los valores de las variables explicativas en el modelo.
  2. Un predictor lineal como función lineal de los regresores

  $$\eta(s,t) = \alpha + \sum_{i=1}^{k} \beta_k X_k(s,t)$$

  con $\alpha, \beta_i \in \mathbb{R}$ (se recomienda emplear *bootstrap paramétrico* para obtener medidas de adecuación de los parámetros estimados) y X_i funciones que dependen de el tiempo, el espacio o ambas.
  3. Una función link suave invertible $g(\cdot)$ que transforma la esperanza de la variable respuesta en el predictor lineal antes expuesto

  $$g\big(\mu(s,t)\big) = \eta(s,t)$$

Con los datos disponibles se llegó a la conclusión que lo adecuado sería modelizar $Z(s,t)$ mediante una distribución gamma (ya que analizando las muestras de NO_2 se pudo ver que su distribución era continua y con una fuerte asimetría como podemos apreciar en la imagen 5.2) y como función link la función logaritmo.

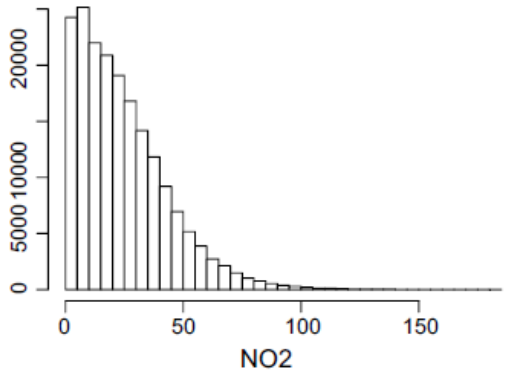


Figura 5.2: Histograma de niveles medios diarios de NO_2. Gráfico sacado del artículo [14].

Como variables explicativas principales se tomaron el tipo de lugar (zona a las afueras de la ciudad, zona industrial o zona de tráfico) y el tipo de ambiente (urbano, suburbano o rural). Esta elección se vio apoyada por un análisis exploratorio previo, mostrado en 5.3.

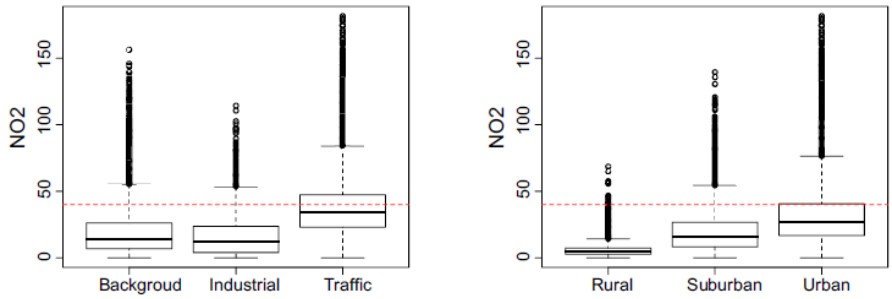


Figura 5.3: Boxplot de los niveles medios diarios de NO_2 según el tipo de lugar y tipo de ambiente. Gráfico sacada del artículo [14].

donde claramente se puede apreciar que los niveles de NO_2 varían tanto por tipo de lugar como por tipo de ambiente.

En esta parte, cabe comentar que los autores se dieron cuenta que en los datos estaba presente un comportamiento con tendencia a largo plazo con episodios cíclicos (como se puede ver en 5.4)

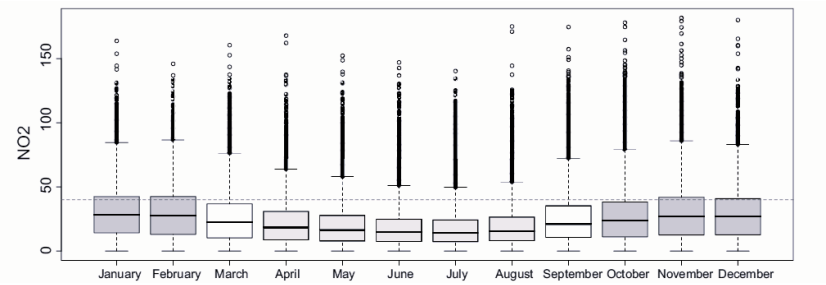


Figura 5.4: Boxplot de los niveles de NO_2 por meses. Figura presente en el artículo [14].

Por ello propusieron cambiar $\eta(s,t)$ en el paso 2, por

$$\eta(s,t) = \alpha + \sum_{i=1}^{k} \beta_i X_i(s,t) + \delta_0 t + \sum_{j=1}^{d} \delta_j s_j + \sum_{b=1}^{l} \left(\phi_{b,1} cos(b\pi t\omega) + \phi_{b,2} sin(b\pi t\omega)\right)$$

con $s = (s_1, \cdots, s_d) \in \mathbb{R}^d$, $\delta_i, \phi_{j,1}, \phi_{j,2} \in \mathbb{R}$ y $\omega \in \mathbb{R}$ (donde ω es la frecuencia fundamental).

- Residuo. La correlación espacio-temporal existente entre los datos, se debe ajustar un semivariograma espacio-temporal válido a los residuos $\varepsilon(s,t) = Z(s,t) - \mu(s,t)$.

Posteriormente con el fin de conseguir una estimación, una vez obtenidas ambas cosas, las predicciones se harán mediante técnicas *kriging*.

Para ver en más profundidad las estimaciones comentadas, se referencia al artículo [14].

5.4. Modelo funcional bidimensional

En el artículo de Oviedo de la Fuente et al. (2020) [11] se retomarán las técnicas propias de *datos funcionales*. En él, se propone un modelo de localización y escala que trata los predictores (concentraciones de SO_2 y NO_x en el tiempo) como funciones mientras que la respuesta es escalar, ya que buscamos obtener la futura concentración de contaminantes.

Pasamos ahora a fijar notación para formular el modelo matemático.

Sea $\{X_i, Y_i\}_{i=1}^n$ un conjunto de observaciones del proceso estocástico $X = (X^1(t), \cdots, X^p(t))$, donde $X^j \in L_2[0,T]$, $j = 1, \cdots, p$ son las covariables predictoras e $Y = (Y_1, Y_2)$, con $Y_j \in \mathbb{R}$, la variable respuesta. En este contexto, se asume el siguiente modelo bivariante de localización y escala

$$\boxed{\begin{pmatrix} Y_1 \\ Y_2 \end{pmatrix} = \begin{pmatrix} \mu_1(X) \\ \mu_2(X) \end{pmatrix} + \Sigma^{1/2}(X) \begin{pmatrix} \varepsilon_1 \\ \varepsilon_2 \end{pmatrix}} \tag{5.4}$$

donde $\Sigma^{1/2}(X)$ representa la descomposición de Cholesky de la matriz de varianzas y covarianzas $\Sigma(X)$

$$\Sigma(X) = \begin{pmatrix} \sigma_1^2(X) & \sigma_{12}(X) \\ \sigma_{12}(X) & \sigma_2(X) \end{pmatrix}$$

de manera que $Var(Y|X) = \Sigma(X) = \Sigma^{1/2}(X)\left(\Sigma^{1/2}(X)\right)^T$.

Cabe destacar que para que no haya problemas de identificación del modelo es necesario suponer que los errores bivariantes $(\varepsilon_1, \varepsilon_2)$ son independientes de las covariables, de media cero, varianza uno y

correlación cero entre ellas. A priori no se fija ninguna distribución del error, pero a lo largo del artículo a la hora de estimar se usan varias distribuciones (como puede ser la distribución de Gauss-Laplace generalizada, modelos lineales generalizados mixtos, etc.)

Además, se definirán regiones de probabilidad incondicional para los errores $(\varepsilon_1, \varepsilon_2)$ como

$$\varepsilon_\tau(k) = \{(\varepsilon_1, \varepsilon_2) \in \mathbb{R}^2 : f(\varepsilon_1, \varepsilon_2) \geq k\}$$

siendo f la función de densidad de los errores bivariantes $(\varepsilon_1, \varepsilon_2)$ y k el cuantil τ de $f(\varepsilon_1, \varepsilon_2)$.

Así pues, dado un X se define la región de incertidumbre τ-ésima para (Y_1, Y_2) conteniendo el $\tau\,\%$ de observaciones como

$$R_\tau(X) = \begin{pmatrix} \mu_1(X) \\ \mu_2(X) \end{pmatrix} + \Sigma^{1/2}(X)\varepsilon_\tau.$$

Con el fin de *familiarizarnos* con la estimación bivariante, procedemos a detallar el ***algoritmo*** empleado por los autores. Denotemos por $\{X_i, (Y_{i1}, Y_{i2})\}$ una muestra de tamaño n, donde $X_i = (X_i^1(t), \cdots, X_i^p(t))$.

1. Representar cada covariable $X^j(t)$ en una base funcional de la forma $X^j(t) \approx \sum_{k=1}^K \psi_k^j \phi_k(t)$, donde ϕ_k $(k = 1, \cdots, K)$ son K funciones de una base funcional (como pueden ser B-splines, wavelets, etc.) e ψ_{il} son o bien los coeficientes de la expansión en una base fija o bien los scores de las componentes principales de la expansión de Karhunen-Loève. Por lo tanto nos quedaríamos con las covariables transformadas

$$\tilde{X}_i = \left((\psi_{i1}^1, \cdots, \psi_{iK}^1); (\psi_{i1}^2, \cdots, \psi_{iK}^2); \cdots; (\psi_{i1}^p, \cdots, \psi_{iK}^p)\right) \;\; i = 1, \cdots, n$$

2. Para $r = 1, 2$ ajustar un modelo aditivo a la muestra $\{\tilde{X}_i, Y_{i1}, Y_{i2}\}_{i=1}^n$ y obtener inicialmente una estimación para las medias

$$\hat{\mu}_r(X_i) = \alpha_r + \sum_{j=1}^p \sum_{k=1}^K \hat{f}_{rk}^j(\psi_{ik}^j)$$

para después estimar $\sigma_r^2(X)$ empleando la muestra $\{\tilde{X}_i, (Y_{ir} - \hat{\mu}_r(X_i))^2\}_{i=1}^n$ como

$$\hat{\sigma}_r^2(X_i) = \exp\left(\hat{\beta}_r + \sum_{j=1}^p \sum_{k=1}^K \hat{g}_{rk}^j(\psi_{ik}^j)\right).$$

Luego, debemos calcular la correlación $\rho(X)$, que se relaciona con la covarianza mediante $\sigma_{12}(X) = \sigma_1(X)\sigma_2(X)\rho(X)$ empleando la muestra $\{X_i, \hat{\delta}_i\}_{i=1}^n$, como sigue

$$\hat{\rho}(X_i) = \tanh\left(\hat{\gamma} + \sum_{j=1}^p \sum_{k=1}^K \hat{m}_k^j(\psi_{ik}^j)\right)$$

siendo

$$\hat{\delta}_i = \frac{(Y_i^1 - \hat{\mu}_1(X_i))(Y_i^2 - \hat{\mu}_2(X_i))}{\hat{\sigma}_1(X_i)\hat{\sigma}_2(X_i)}$$

donde $f_{rk}^i, g_{rk}^i, m_k^i$ son funciones suaves y $\alpha_r, \beta_r, \gamma$ coeficientes, p el número de covariables y K el número de elementos de la base. Para evitar problemas de identificación es preciso suponer que f_j, g_j, m_j tienen media cero. Antes de pasar al siguiente paso, cabe comentar que la elección de las funciones $\exp(\cdot)$ y $\tanh(\cdot)$ no es arbitraria, si no que se eligen así para poder cumplir las restricciones en el espacio de parámetros, esto es, asegurarnos de que $\sigma_r^2(X)$ y $0 \leq \rho(X) \leq 1$.

3. Calcular los residuos estandarizados

$$\begin{pmatrix} \hat{\varepsilon}_{i1} \\ \\ \hat{\varepsilon}_{i2} \end{pmatrix} = \hat{\Sigma}^{1/2}(X_i) \begin{pmatrix} Y_{i1} - \hat{\mu}_1(X_i) \\ \\ Y_{i2} - \hat{\mu}_2(X_i) \end{pmatrix} \quad i = 1, \cdots, n$$

donde

$$\hat{\Sigma}(X_i) = \begin{pmatrix} \hat{\sigma}_1^2(X_i) & \hat{\sigma}_{12}(X_i) \\ \\ \hat{\sigma}_{12}(X_i) & \hat{\sigma}_2^2(X_i) \end{pmatrix}$$

con $\hat{\sigma}_{12}(X_i) = \hat{\sigma}_1(X_i)\hat{\sigma}_2(X_i)\hat{\rho}(X_i)$.

4. Obtener el estimador de la densidad bivariante $\hat{f}(\varepsilon_1, \varepsilon_2)$ dada por

$$\hat{f}((\varepsilon_1, \varepsilon_2), H) = \frac{1}{n} \sum_{i=1}^{n} K_H \begin{pmatrix} \varepsilon_1 - \hat{\varepsilon}_{i1} \\ \\ \varepsilon_2 - \hat{\varepsilon}_{i2} \end{pmatrix}$$

donde $K(\cdot)$ es una función tipo núcleo con una densidad de probabilidad simétrica y H una matriz 2×2 definida positiva. Para obtener la región de incertidumbre incondicional bivariante en la escala de los residuos

$$\hat{\varepsilon}_\tau\{(\varepsilon_1, \varepsilon_2) \in \mathbb{R}^2 : \hat{f}(\varepsilon_1, \varepsilon_2) \geq \hat{k}\}$$

siendo $\hat{k}$ el τ cuantil empírico de los valores $\hat{f}(\varepsilon_{11}, \varepsilon_{12}), \cdots, \hat{f}(\varepsilon_{n1}, \varepsilon_{n2})$.

Finalmente dado X la región condicional bivariante $R_\tau(X)$ estimada es

$$\hat{R}_\tau(X) = \begin{pmatrix} \hat{\mu}_1(X) \\ \\ \hat{\mu}_2(X) \end{pmatrix} + \hat{\Sigma}^{1/2}(X)\hat{\varepsilon}_\tau.$$

Particularizando al caso que nos compete se consideró

$$\big(Y_1, Y_2\big) = \big(SO(t_0 + t_h), NO(t_0 + t_h)\big)$$

siendo t_0 el tiempo presente medido cada cinco minutos y $SO(t_0), NO(t_0)$ las concentraciones obtenidas con las series medias bihorarias de SO_2 y NO_x en el instante t_0 y t_h el tiempo horizonte de predicción.

Como covariables se emplearon las series de medias bihorarias de esos contaminantes así como sus respectivas derivadas (en verdad una submuestra de ellas, dada por la matriz histórica)

$$X = \big(X^1(t), X^2(t), X^3(t), X^4(t)\big) = \big(SO(t), NO(t), SO'(t), NO'(t)\big)$$

con $t \in [t_0 - t_{lag}, t_0]$ y $\big(NO'(t), SO'(t)\big)$ las primeras derivadas de las funciones que aproximan ambos contaminantes (obtenidas empleando la representación funcional en una base). t_{lag} denota el retraso temporal de los predictores. Se fijó por la normativa vigente en aquella época tomar $t_h = 12$.

Para finalizar este capítulo, cabe destacar que que existen otros artículos en la literatura que tratan el problema de predicción univariante y bivariante.

En el caso de predicción bivariante, podemos destacar el trabajo de Roca Pardiñas et al (2021) [17], en donde se usan técnicas no paramétricas con un modelo de localización y escala análogo al presente en el artículo de Oviedo de la Fuente et al. (2020) [11].

En el caso de predicción univariante, podemos mencionar el artículo de Noel Cressie (2018) [10], que emplea técnicas espacio-temporales para predecir futuros valores de CO_2. Otro artículo interesante en este aspecto, es el de Ma et al. (2021) [9] que emplea un modelo que mezcla técnicas de machine learning con métodos Montecarlo para identificar las fuentes de contaminantes atmosféricos.

Capítulo 6

Ilustración de algunos métodos de predicción con datos reales

Empezaremos comentando que el código utilizado para la generación de las gráficas que mostraremos a continuación se halla en el apéndice. Éste está dividido en secciones según el para qué se utilice cada código; además, éste incluye pequeños comentarios.

Conseguir datasets con mediciones actuales y reales de variables atmosféricas en Internet, es cuando menos no inmediato. Con las nuevas leyes relativas a la protección de datos no es tan sencillo obtener éstos de organismos oficiales o empresas y en caso de conseguirlos (ya sea accediendo a las bases de datos donde están o bien vía API REST[1] de tipo GET, por ejemplo) no se pueden exponer públicamente debido a estas políticas vigentes.

Por este motivo fue interesante considerar la página web https://archive.ics.uci.edu. Ésta es un repositorio de datasets relativos a diferentes proyectos de machine learning, con licencia pública de la Universidad de California, que pueden ser utilizados en entornos educativos. Las temáticas de los datasets son muy diversas, abarcando desde proyectos de ingeniería o ciencias de la salud hasta ciencias de la computación o negocios. Su crecimiento se debe mayoritariamente a que se permite hacer *donaciones* de datasets siempre y cuando se cuenten con los permisos adecuados sobre los mismos.

Nosotros con el fin de ilustrar con el lenguaje `R` la implementación de algunos de los modelos revisados, nos quedaremos con el dataset llamado *Air Quality*. Éste contiene datos que recogen mediciones *horarias* hechas por sensores de concentraciones de diversos contaminantes. Las fechas de recogida de datos están comprendidas entre marzo de 2004 y febrero de 2005. La localización es en una ciudad italiana.

De todas las variables disponibles, nosotros nos quedaremos con las columnas relativas a:

- *Fecha*: `Date`.
- *Hora*: `Time`.
- *Concentración media de CO* (medida en mg/m^3): `CO.GT.`
- *Concentración media de* NO_x (medida en *ppb*): `NOx.GT.`
- *Temperatura* (medida en grados centígrados): `T`.
- *Humedad relativa*: `RH`.
- *Humedad absoluta*: `AH`.

[1] Se puede consultar más sobre esto aquí.

6.1. Pequeño análisis de datos y problema de imputación

Ahora que tenemos los datos haremos un pequeño análisis exploratorio. Para ello en la figura 6.1 empezaremos viendo qué forma tienen los datos de NO_x. Procedimos a quitar aquellas filas que tenían

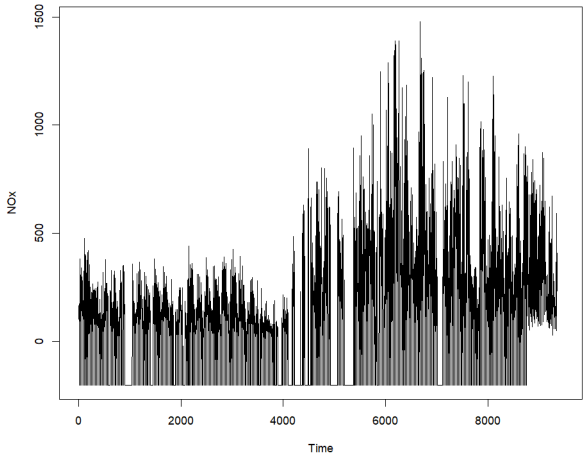


Figura 6.1: NO_x original

valores `NA` para el NO_x (que se correspondían con las últimas 114 filas, que no llegaban al 2 % de los datos totales).

Además de esto, para que el primer día comience a las 00:00 y el último acabe a las 23:00, procedimos a quitar los datos relativos al primer y último día del dataset original.

Con el fin de ver la distribución de estos datos, veremos en 6.2 su histograma correspondiente:

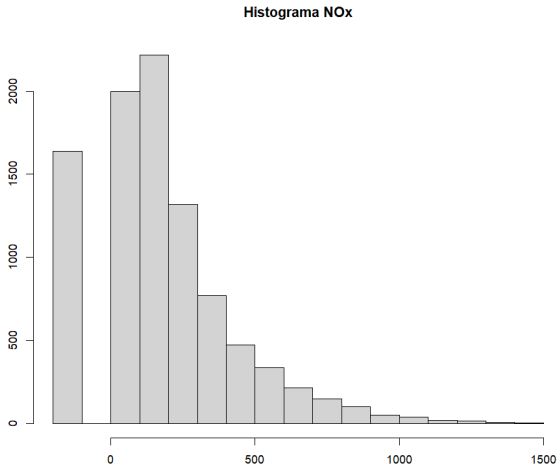


Figura 6.2: Histograma NO_x

Como cabía esperar, por los análisis exploratorios preliminares de los artículos, se trata de una distribución con alta concentración en torno al cero y con valores altos con poca representación. Por otra parte, hay ciertos valores peculiares que toman el valor -200; éstos no son más que una simple imputación para datos faltantes, como bien comentan los autores del dataset.

Para ver cómo trataremos esto, nos quedaremos con los datos relativos al mes de marzo de 2004.

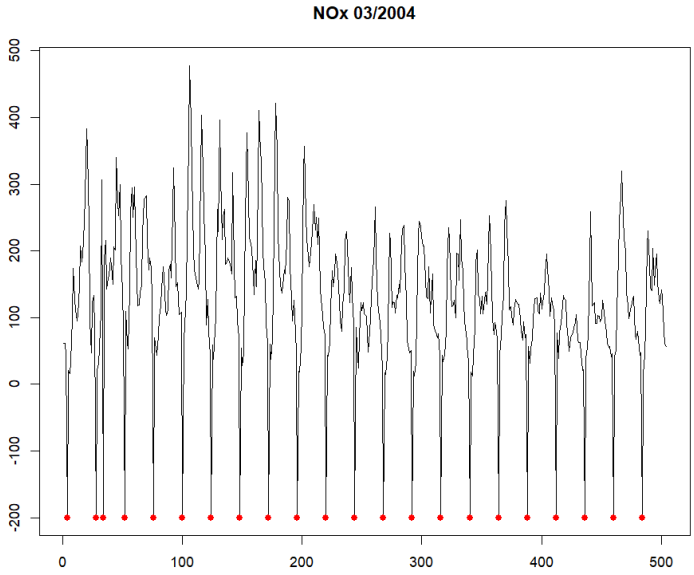


Figura 6.3: Marzo 2004

En la figura 6.3 en color rojo se han representado los valores imputados por los autores. En esta ocasión, se trata de puntos aislados, que en su entorno tienen valores de NO_x no imputados por lo que una solución interesante sería calcular el estimador de Nadaraya Watson y evaluarlo en los puntos con valores -200. Posteriormente, veremos que existe otro caso en el los datos faltantes forman *bloques*, por lo que en su entorno no tienen demasiados puntos, así que la estimación no funcionará tan bien.

Veamos en 6.4 el resultado final de la imputación que se pensó:

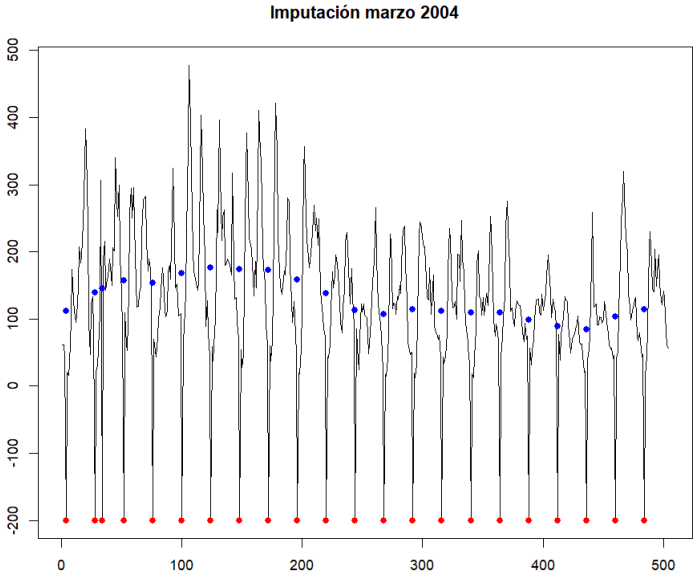


Figura 6.4: Imputación no paramétrica

En azul tenemos la estimación no paramétrica para los valores en rojo. En 6.5 veremos de paso dos histogramas para comprobar la distribución correspondiente a los datos originales y aquellos a los que

se le ha aplicado la estimación no paramétrica:

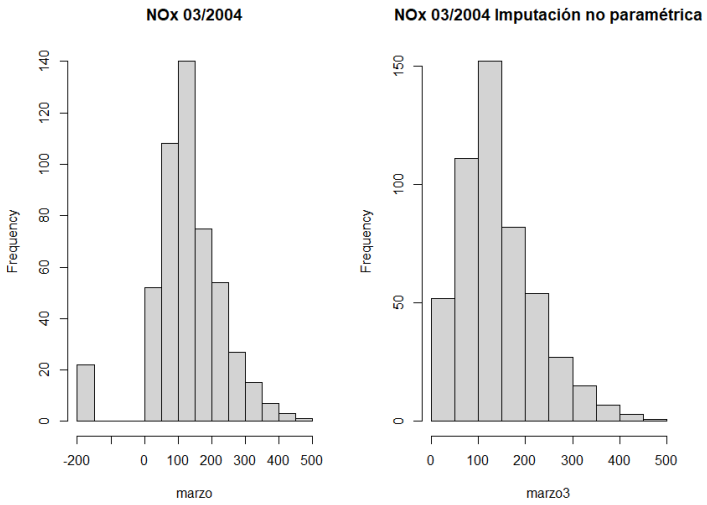


Figura 6.5: NO_x antes y después de la imputación

De la misma manera, podemos ver en 6.6 cómo ha quedado la serie de tiempo imputada:

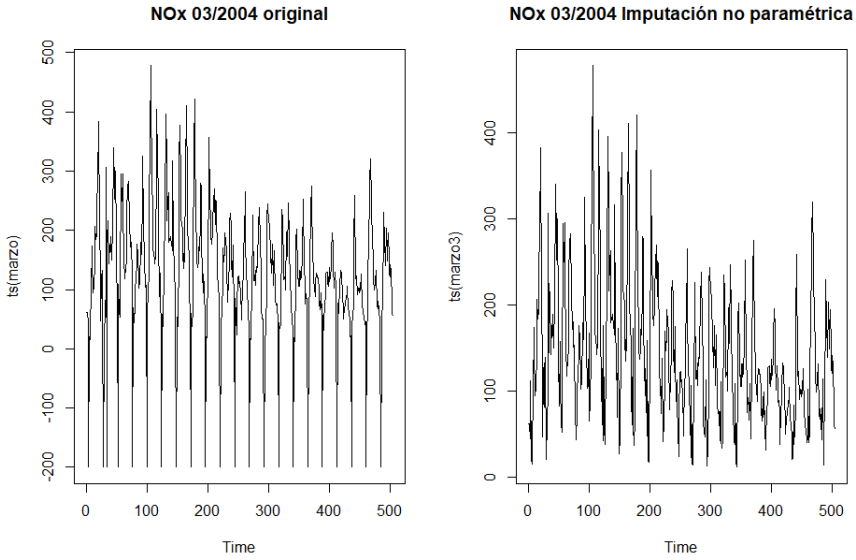


Figura 6.6: Serie de tiempo imputada

Haciendo una prueba de Kolmogorov-Smirnov podemos apreciar que la distribución no ha cambiado.

Esta técnica se ha empleado para el resto de variables que hemos elegido; ahora bien, como comentamos inicialmente no funciona demasiado bien cuando presenta datos imputados (con valor igual a -200) formando bloques. Este comportamiento se puede apreciar en la imagen 6.7, que se corresponde en nuestro dataset, a la imputación correspondiente a la variable que recoge información de CO.

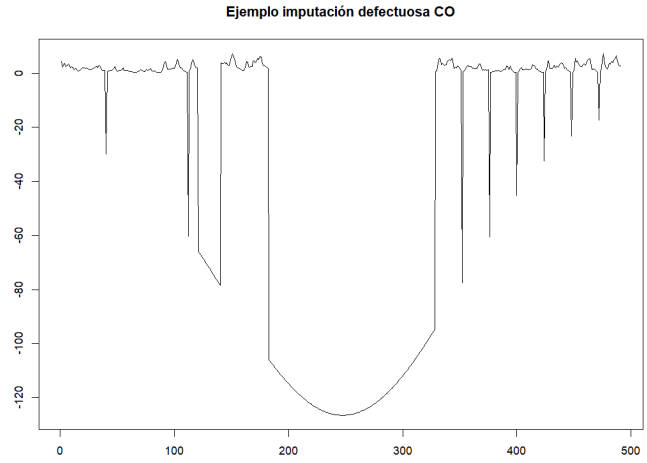


Figura 6.7: Imputación no paramétrica en el caso de que formen bloques los datos faltantes

Es por ello que debemos tener cuidado a la hora de elegir la muestra de entrenamiento y test.

6.2. Elección de la muestra de entrenamiento y test

En nuestro caso, nuestra muestra de entrenamiento se corresponderá al mes de marzo de 2004 y como muestra test tomaremos el día 1 de abril de 2004, ambas tomadas de los datos de NO_x. Las representamos en la figura 6.8.

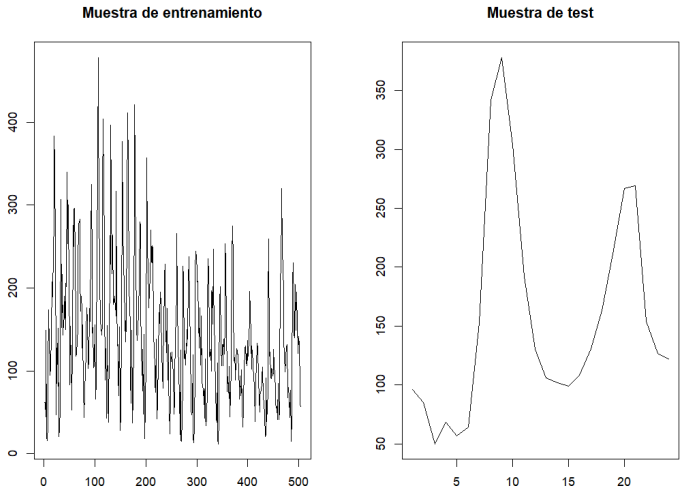


Figura 6.8: Muestra de entrenamiento y muestra test

6.3. Implementación de modelos y resultados

6.3.1. Box-Jenkins

Para empezar, veremos, cómo la aproximación con modelos Box-Jenkins no es la adecuada.

Para ello emplearemos las funciones automáticas de ajustes de series de tiempo de `R`, en particular la rutina `auto.arima()`.

La hipótesis fundamental sobre la que se sustentan los modelos clásicos Box-Jenkins es que las innovaciones sean gaussianas (para lo cual se analizan los residuos).

Para ello podemos representar un boxplot con ellos y apreciar gráficamente en 6.9 que no se ajustan a la recta $y = x$. Haciendo las pruebas correspondientes (Jarque Bera y Shapiro) se puede apreciar que analíticamente se corrobora que falla esta hipótesis:

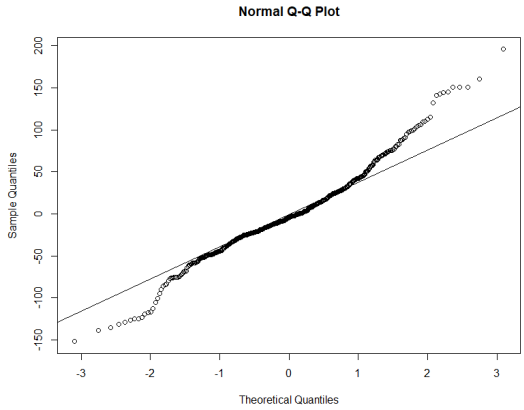


Figura 6.9: Falta de normalidad en los residuos

Esto nos llevará a que si empleamos dicho modelo para predecir, lo haga bastante mal, como podemos apreciar en 6.10:

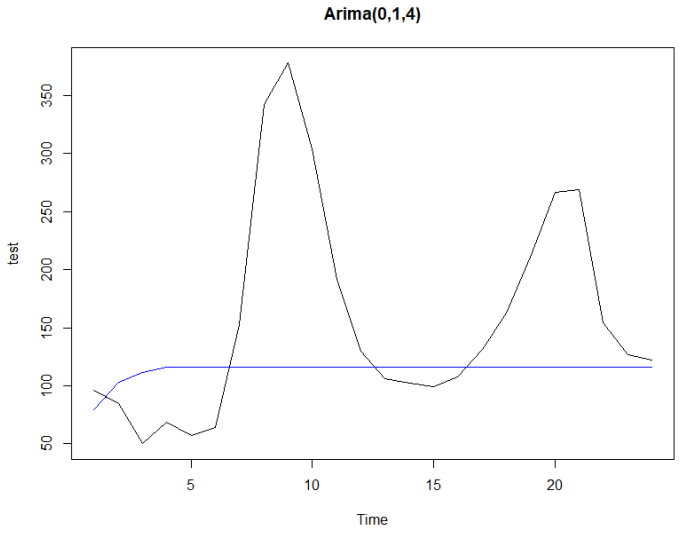


Figura 6.10: Predicción usando modelo ARIMA ajustado

6.3.2. No paramétrico

En nuestro caso en particular buscaremos estimar la función de autorregresión $\varphi(\cdot)$ de X_{n+1} en base a X_n. Para ello, emplearemos el estimador de Nadaraya Watson. Podemos ver en 6.11 el diagrama de dispersión correspondiente junto con su estimación no paramétrica.

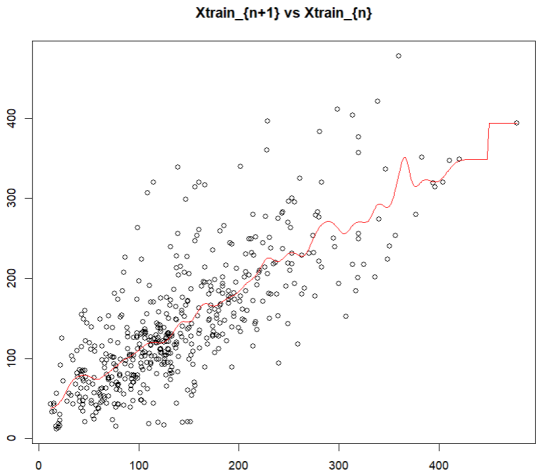


Figura 6.11: Diagrama de dispersión de X_{n+1} (eje Y) vs X_n (eje X); en rojo, la estimación no paramétrica Nadaraya Watson

Para obtener las predicciones que querramos, simplemente tendremos que evaluar el estimador en los valores deseados, esto es:

$$\hat{X}_{n+1} = \hat{\varphi}(X_n^{test})$$

Podemos ver en 6.12 cómo queda el ajuste con respecto a la muestra test:

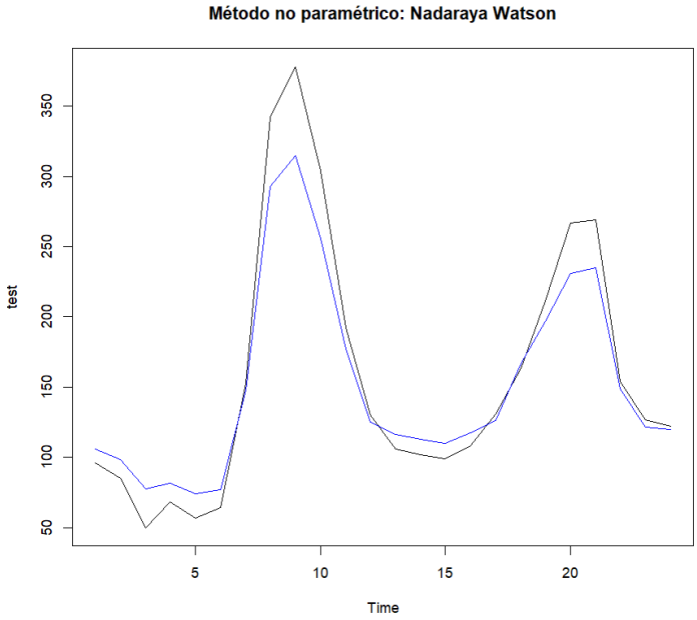


Figura 6.12: Predicción no paramétrica

Como comentario final, en la rutina de R, quizás los valores que tome la muestra test no estén presentes en el soporte del estimador Nadaraya Watson que hemos obtenido para la muestra de entrenamiento. En este caso, optamos por sustituir la estimación de ese valor por aquella estimación del valor que esté más próximo al valor correspondiente de la muestra de entrenamiento.

6.3.3. Semiparamétrico

Procedemos a implementar ahora la corrección semiparamétrica.

Con respecto a la aproximación anterior en este caso solo tendremos que sumarle a la predicción para el tiempo $n+1$ no paramétrica, la predicción una etapa por delante de la serie residual correspondiente a la muestra de entrenamiento hasta tiempo n.

Veamos en 6.13 qué nos queda con respecto a la muestra test:

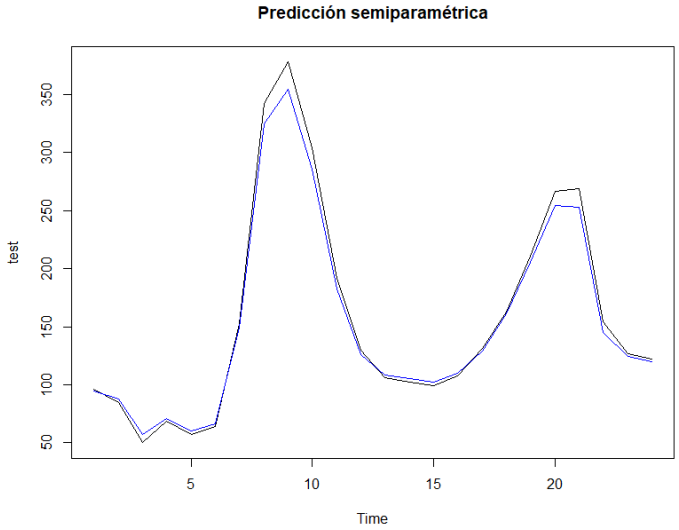


Figura 6.13: Predicción semiparamétrica

6.3.4. Redes neuronales

En este caso, tendremos que emplear la librería `forecast` de R y dentro de ésta la función `nnetar()` útil para predecir series de tiempo univariantes. Ésta nos permite entrenar una red neuronal *feed-forward* con una única capa oculta con inputs las propias series *laggeadas*. Tiene como parámetros principales:

- `y`: un vector numérico de tipo *ts*.
- `size`: número de nodos en la capa oculta.
- `xreg`: vector o array de regresores externos. Es importante que sea numérico y tenga el mismo número de filas que `y`.
- `repeats`: número de redes neuronales a ajustar con pesos aleatorios iniciales. Éstos se promedian para obtener predicciones.

Admite también parámetros de la función `nnet()`, que se emplea para entrenar redes neuronales con una única capa oculta. De todos los parámetros que admite esta última función que nosotros acabamos de mencionar, utilizaremos:

- `maxit`: número máximo de iteraciones durante el entrenamiento. Recordemos que al final nosotros al entrenar una red neuronal vamos modificando los pesos entre las distintas conexiones entre neuronas con el fin de cometer el menor error posible a la hora de predecir lo que nos interesa. Éste parámetro nos permite controlar la convergencia o divergencia hacia lo que estamos buscando predecir.

- `abstol`: que permite parar el entrenamiento siempre que tengamos un error de ajuste por debajo de éste.

En nuestro caso, fijamos `abstol` a 1e-07. Para el resto de parámetros, decidimos dejar los que cometen un menor error ajustándose a la muestra de entrenamiento, obtenidos vía experimentación.

Lo suyo, sería hacer un *grid* e ir comparando los errores cometidos por cada combinación de valores de los parámetros de la rutina (en nuestro caso particular por ejemplo tomando `size`, `repeats`, `maxit`) al ajustar la muestra de entrenamiento, quedándonos con aquel que cometa menor error. Éste proceso, sin embargo, lleva bastante tiempo de computación, con lo que habría que tener cuidado de construir un grid con demasiados puntos.

Veamos en 6.14 cómo queda nuestra predicción de la muestra test:

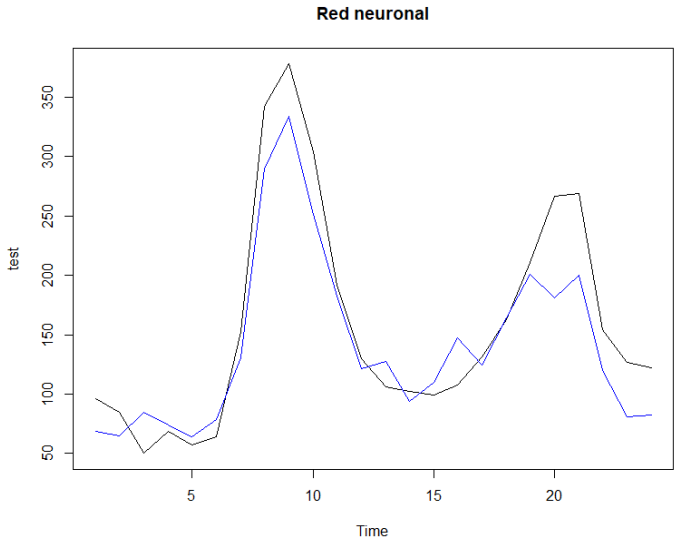


Figura 6.14: Predicción red neuronal

6.3.5. Comparación final

Para terminar, hagamos una gráfica 6.15 conjunta de los métodos de predicción que hemos expuesto hasta el momento:

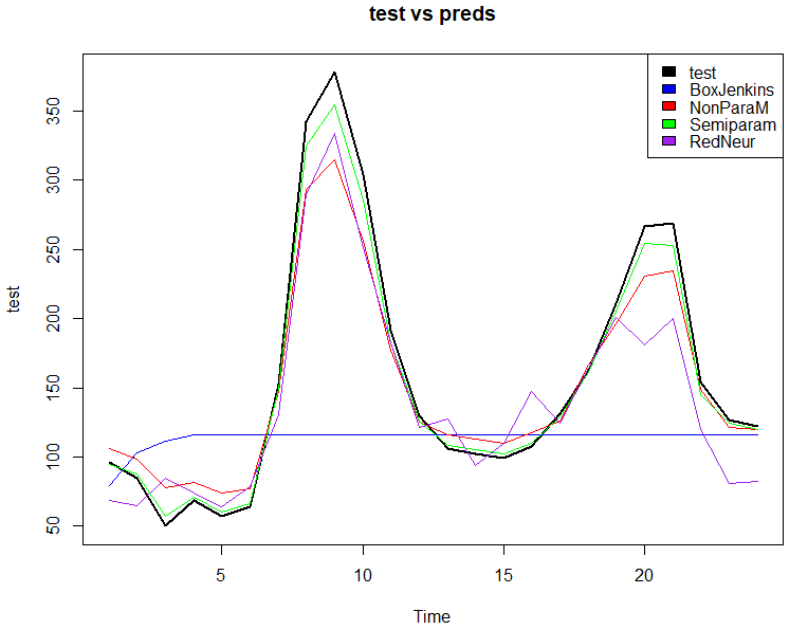


Figura 6.15: Comparación de métodos

Por último veamos una tabla con el RMSE y MAE para cada método:

Método	RMSE	MAE
Box-Jenkins	67.63924	99.01723
No paramétrico	17.79915	24.00141
Semiparamétrico	6.651436	9.075109
Red Neuronal	27.9957	35.44523

El método que mejor funciona para este ejemplo de muestra test en este caso es el modelo semiparamétrico; se puede apreciar que hace pequeñas correciones al método no paramétrico. Por otra parte, el modelo Box-Jenkins lo hace bastante mal, que era lo esperado ya que no cumple las hipótesis necesarias para su utilización. La red neuronal funciona bastante bien para predecir el primer episodio con valores elevados, sin embargo en el segundo comete bastantes errores.

Bibliografía

[1] Atakan Kurt, A. B. Oktay, Forecasting air pollutant indicator levels with geographic models 3days in advance using neural networks (2010), Expert Systems with Applications, Volume 37, Issue 12, Pages 7986-7992, ISSN 0957-4174.

[2] Conde-Amboage, M., González-Manteiga, W. & Sánchez-Sellero, C. (2016). Predicting trace gas concentrations using quantile regression models. Stochastic Environment Research and Risk Assessment.

[3] Fernández de Castro, B.M., Guillas, S. & González-Manteiga, W. (2005). Functional Samples and Boostrap for Predicting Sulfur Dioxide Levels. Technometrics, 47, 212-222.

[4] Fernández de Castro, B.M., Prada-Sánchez, J.M., González-Manteiga, W., Febrero-Bande, M., Bermúdez-Cela, J.L. & Hernández Fernández J.J. (2003). Prediction of SO2 levels using neural networks. Journal of the Air and Waste Management Association, 53, 532-538.

[5] García-Jurado, I., González-Manteiga, W., Prada-Sánchez, J.M., Febrero-Bande, M. & Cao, R. (1995). Predicting using Box-Jenkins, Nonparametric and Bootstrap Techniques. Technometrics, 37, 303-310.

[6] González Manteiga, W., Febrero Bande, M. Piñeiro Lamas, M. (2009). Multidimensional Semiparametric Prediction with Cointegration in errors for Pollution Indicators, Proceedings of The 57th Session of the ISI International Statistical Institute.

[7] González-Manteiga, Wenceslao, Febrero-Bande, Manuel and Oviedo de la Fuente, Manuel, Semiparametric prediction models for variables related with energy production (2016) .

[8] Hwang, Youngdeok, Emre Barut, and Kyongmin Yeo. Statistical-Physical estimation of pollution emission (2018). Statistica Sinica 28, no. 2, 921-40.

[9] Ma, D., Gao, J., Zhang, Z. et al. Identifying atmospheric pollutant sources using a machine learning dispersion model and Markov chain Monte Carlo methods (2021). Stoch Environ Res Risk Assess 35, 271-286.

[10] Noel Cressie (2018) Mission CO2ntrol: A Statistical Scientist's Role in Remote Sensing of Atmospheric Carbon Dioxide, Journal of the American Statistical Association, 113:521, 152-168.

[11] Oviedo-de La Fuente, M.; Ordóñez, C.; Roca-Pardiñas, J. Functional Location-Scale Model to Forecast Bivariate Pollution Episodes (2020). Mathematics 2020, 8, 941.

[12] Prada-Sánchez, J.M. & Febrero-Bande, M. (1997). Parametric, Non-Parametric and Mixed approaches to prediction of sparsely distributed pollution incidents: a case study. Journal of Chemometrics, 11, 13-32.

[13] Prada-Sánchez, J.M., Febrero-Bande, M., Cotos-Yáñez, T., González-Manteiga, W., Bermúdez-Cela, J.L. & Lucas-Domínguez T. (2000). Prediction of SO2 pollution incidents near a power station using partially linear models and a historical matrix of predictor-response vectors. Environmetrics, 11, 209-225.

[14] Raquel Menezes & Helena Piairo & Pilar García Soidán & Inés Sousa, Spatial-temporal modellization of the NO2 concentration data through geostatistical tools (2016) , Statistical Methods & Applications, Springer; S. Italiana di Statistica, vol. 25(1), pages 107-124.

[15] Roca Pardiñas, J., Cadarso Suárez, C. & González Manteiga, W. (2005). Testing for interactions in generalized additive models: Application to SO2 pollution data. Statistics and Computing, 15, 289-299.

[16] Roca Pardiñas, J., González Manteiga, W., Febrero Bande, M., Prada Sánchez, J.M. & Cadarso Suárez C. (2004). Predicting binary time series of SO2 using generalized additive models with unknown link function. Environmetrics, 15, 729-742.

[17] Roca-Pardiñas, J., Ordóñez C. & Lado-Baleato, O. Nonparametric location-scale model for the joint forecasting of SO2 and NOx pollution episodes (2021). Stoch Environ Res Risk Assess 35, 231-244.

[18] Thombs, L. A., and Schucany, W. R. (1990). Bootstrap Prediction Intervals for Autoregression, Journal c:f the American Stutisticul Association, 85,48-492.

[19] Apuntes de la materia *Series de Tiempo* (UDC) del MTE.

[20] Apuntes de la materia *Regresión generalizada y modelos mixtos* (USC) del MTE.

[21] Apuntes de la materia *Procesos estocásticos* (USC) del MTE.

[22] Apuntes de la materia *Técnicas de Remuestreo* (UDC) del MTE.

[23] Apuntes de la materia *Estadística Espacial* (UVigo) del MTE.

Apéndice

Análisis exploratorio

```
#Lectura de datos disponibles en el repositorio mencionado
datos=read.csv2("AirQualityUCI.csv")
head(datos)
names(datos)

#Representación NOx
plot(ts(datos$NOx.GT.),ylab="NOx")

#Datos de NOx con NA
which(is.na(datos$NOx.GT.))

#Porcentaje datos NA
sum(is.na(datos$NOx.GT.))/length(datos$NOx.GT.)

#Recorte comentado
dataset=datos[-c(which(is.na(datos$NOx)),which(datos$Date=="04/04/2005"),
which(datos$Date=="10/03/2004")),c("Date","Time","NOx.GT.","CO.GT.","T","RH","AH")]

#Histograma nuevos datos
hist(dataset$NOx.GT.,main="Histograma NOx",xlab="",ylab="")

#Datos de marzo de 2004
marzo=dataset$NOx.GT.[which(substr(dataset$Date,4,10)=="03/2004")]
plot(ts(marzo),main="NOx 03/2004")
points(which(marzo==-200),marzo[which(marzo==-200)],col="red",pch=19)

#Estimación no paramétrica para los datos de marzo de 2004
marzo3=marzo
time=1:length(marzo3)
plot(time,marzo)
hCV<-(np.cv(as.matrix(cbind(marzo,time)),kernel="gaussian")$h.opt)[2,1]
nw<-ksmooth(time,marzo,kernel="normal",bandwidth=hCV)
lines(nw,col="red")

#Imputación al mes de marzo con regresión no paramétrica
marzo3[which(marzo3==-200)]=nw$y[which(marzo3==-200)]

#Visualización: imputación mes de marzo  Nadaraya Watson vs original
plot(time,marzo,type="l",main="Imputación marzo 2004")
points(which(marzo==-200),marzo[marzo==-200],col="red",pch=19)
points(which(marzo==-200),nw$y[which(marzo==-200)],col="blue",pch=19)

#Visualización 2: imputación mes de marzo Nadaraya Watson vs original histograma
par(mfrow=c(1,2))
hist(marzo,main="NOx 03/2004")
hist(marzo3,main="NOx 03/2004 Imputación no paramétrica")
par(mfrow=c(1,1))

#Visualización 3: imputación mes de marzo Nadaraya Watson vs original serie de tiempo
par(mfrow=c(1,2))
plot(ts(marzo),main="NOx 03/2004 original")
plot(ts(marzo3),main="NOx 03/2004 Imputación no paramétrica")
par(mfrow=c(1,1))

#Misma distribución mes de marzo antes y después de la imputación
ks.test(marzo,marzo3)
```

Imputación no paramétrica a cada una de las variables utilizadas

```
#######IMPUTACIONES Y CREACIÓN DEL NUEVO DATASET (tarda bastante en ejecutar esta parte)
dataset3=dataset

#NOx
a=dataset3$NOx.GT.
time=1:length(a)
hCVa<-(np.cv(as.matrix(cbind(a,time)),kernel="gaussian")$h.opt)[2,1]
nwa<-ksmooth(time,a,kernel="normal",bandwidth=hCVa)

#CO
b=dataset3$CO.GT.
time=1:length(b)
hCVb<-(np.cv(as.matrix(cbind(b,time)),kernel="gaussian")$h.opt)[2,1]
nwb<-ksmooth(time,b,kernel="normal",bandwidth=hCVb)

#Temperature
c=dataset3$T
time=1:length(c)
hCVc<-(np.cv(as.matrix(cbind(c,time)),kernel="gaussian")$h.opt)[2,1]
nwc<-ksmooth(time,c,kernel="normal",bandwidth=hCVc)

#Relative humidity
d=dataset3$RH
time=1:length(d)
hCVd<-(np.cv(as.matrix(cbind(d,time)),kernel="gaussian")$h.opt)[2,1]
nwd<-ksmooth(time,d,kernel="normal",bandwidth=hCVd)

#Absolute humidity
e=dataset3$AH
time=1:length(e)
hCVe<-(np.cv(as.matrix(cbind(e,time)),kernel="gaussian")$h.opt)[2,1]
nwe<-ksmooth(time,e,kernel="normal",bandwidth=hCVe)


dataset3$NOx.GT.[which(dataset3$NOx.GT.==-200)]=nwa$y[which(dataset3$NOx.GT.==-200)]
dataset3$CO.GT.[which(dataset3$CO.GT.==-200)]=nwb$y[which(dataset3$CO.GT.==-200)]
dataset3$T[which(dataset3$T==-200)]=nwc$y[which(dataset3$T==-200)]
dataset3$RH[which(dataset3$RH==-200)]=nwd$y[which(dataset3$RH==-200)]
dataset3$AH[which(dataset3$AH==-200)]=nwe$y[which(dataset3$AH==-200)]

#Escritura del nuevo dataset con las imputaciones correspondientes
write.csv(dataset3, "finaldataset.csv")
```

Creación de la muestra test y muestra de entrenamiento

```
#Ejemplo de imputación defectuosa
plot(data$CO.GT.[710:1200],main="Ejemplo imputación defectuosa CO",type="l")

#######DECLARACIÓN DE MUESTRA DE ENTRENAMIENTO Y MUESTRA TEST
data=read.csv("finaldataset.csv")
data=data[,-1]
#Train:mes de marzo
train=ts(data$NOx.GT.[substr(data$Date,4,10)=="03/2004"])
par(mfrow=c(1,2))
plot(train,main="Muestra de entrenamiento",ylab="",xlab="")
#Test: siguientes 24 h
test=ts(data$NOx.GT.[substr(data$Date,1,10)=="01/04/2004"])
plot(test,main="Muestra de test",ylab="",xlab="")
par(mfrow=c(1,1))
```

Box-Jenkins

```
#BOX JENKINS
library(fpp2)
library(tseries)
auto.arima(train)
ajuste <- Arima(train, order=c(0,1,4),include.mean=FALSE,lambda=NULL)
ajuste

#Todos los coeficientes son significativos
abs(ajuste$coef) < 1.96*sqrt(diag(ajuste$var.coef))

#Residuos
checkresiduals(ajuste)
tsdiag(ajuste)

#Contraste de media nula
t.test(ajuste$residuals, mu=0)

#Normalidad de los residuos
qqnorm(ajuste$residuals)
qqline(ajuste$residuals)

jarque.bera.test(ajuste$residuals)
shapiro.test(ajuste$residuals)

#Innovaciones obviamente no gaussianas, como era de esperar
#Box Jenkins forecast
bjp<-forecast(ajuste,length(test))
#Representación gráfica predicción vs test
plot(test,main="Arima(0,1,4)")
lines(as.numeric(bjp$mean),col="blue")
```

Modelo no paramétrico

```
#NO PARAMÉTRICO
library(ks)
library(PLRModels)
library(KernSmooth)
library(MASS)
plot(train[-length(train)],train[-1],main="Xtrain_{n+1} vs Xtrain_{n}",xlab="",ylab="")
#Cálculo de la ventana óptima vía validación cruzada
hCV<-(np.cv(as.matrix(cbind(train[-1],train[-length(train)])),kernel="gaussian")$h.opt)[2,1]
#Ya tenemos la estimación de la función de autorregresión. Por cómo lo hemos construido
#tenemos que f(x_{n-1})=x_{n}
nw1<-ksmooth(train[-length(train)],train[-1],kernel="normal",bandwidth=hCV)
lines(nw1,col="red")

#En caso de no tener el valor concreto de la muestra test entre los valores del soporte de entrenamiento,
#aproximamos la predicción buscada por la evaluación de la función de autoregresión en el valor
#cuya distancia a la muestra de entrenamiento en valor absoluto es más pequeña
npp=numeric(length(test))
for(i in 1:length(test)){
        npp[i]=nw1$y[which.min(abs(nw1$x-test[i]))]
}
plot(test,main="Método no paramétrico: Nadaraya Watson")
lines(as.numeric(npp),col="blue")
```

Modelo semiparamétrico

```
#SEMIPARAMÉTRICO
ex=train[-length(train)]
#Evaluamos X_{n-1} en la estimación de la función de regresión
px=numeric(length(ex))
for(i in 1:length(px)){
        px[i]=nw1$y[which.min(abs(nw1$x-ex[i]))]
}

#Serie residual para el tiempo inicial
rs=ex-px
#plot(ts(rs))
```

```
ajuste=auto.arima(rs)


as=numeric(length(test))
spp=numeric(length(test))
#Necesitamos valores de arranque
as[1]=as.numeric(forecast(ajuste,1)$mean)
spp[1]=as[1]+npp[1]
#Aquí necesitamos ir añadiendo más puntos a la muestra de entrenamiento para poder obtener la serie residual
#en tiempo n
for(i in 2:length(test)){
        ex=c(train[-length(train)],test[1:i])
        hCV<-(np.cv(as.matrix(cbind(ex[-1],ex[-length(ex)])),kernel="gaussian")$h.opt)[2,1]
        nw1<-ksmooth(ex[-length(ex)],ex[-1],kernel="normal",bandwidth=hCV)
        px=numeric(length(ex))
        for(j in 1:length(px)){
                px[j]=nw1$y[which.min(abs(nw1$x-ex[j]))]
        }
        #Serie residual
        rs=ex-px
        #plot(ts(rs))
        #Ajuste de la serie residual en tiempo n
        ajuste=auto.arima(rs)
        as[i]=as.numeric(forecast(ajuste,1)$mean)
        spp[i]=as[i]+npp[i]
}

plot(test,main="Predicción semiparamétrica")
lines(as.numeric(spp),col="blue")
```

Red neuronal

```
#RED NEURONAL
library(forecast)
#Se fija la semilla para reproducibilidad
set.seed(35643419)
#Debemos crear un dataset con las variables predictoras y a predecir
neuraldataset=cbind(x=train[-length(train)],y=train[-1])

q2=nnetar(y=neuraldataset[,2],size=33,xreg=neuraldataset[,1],maxit=250,repeats=40,abstol=1.0e-7)

#Predicción
nt2=forecast(q2,xreg=test)

#Representación final
plot(test,main="Red neuronal")
lines(as.numeric(nt2$mean),col="purple")
```

Comparación final conjunta y obtención de errores

```
#######GRÁFICO CONJUNTO
plot(test,col="black",main="test vs preds",type="l",lwd=2)
lines(as.numeric(bjp$mean),col="blue")
lines(npp,col="red")
lines(spp,col="green")
lines(as.numeric(nt2$mean),col="purple")
legend(x="topright",legend=c("test","BoxJenkins","NonParaM","Semiparam","RedNeur"),
fill=c("black","blue","red","green","purple"))

#######ERRORES: MSE, MAE
#BoxJenkins
ebj=as.numeric(test)-as.numeric(bjp$mean)
sqrt(mean(ebj^2))
mean(abs(ebj))

#No paramétrico
enp=as.numeric(test)-npp
sqrt(mean(enp^2))
mean(abs(enp))
```

```
#Semiparamétrico
esp=as.numeric(test)-spp
sqrt(mean(esp^2))
mean(abs(esp))

#Red neuronal
enn=as.numeric(test)-as.numeric(nt2$mean)
sqrt(mean(enn^2))
mean(abs(enn))
```