Procesos Gaussianos para problemas de regresión y estimación de la incertidumbre

Victor de la Pompa Porras

Máster en Investigación
e Innovación en TIC

MÁSTERES
DE LA UAM
2017 - 2018

Escuela Politécnica Superior

UAM Universidad Autónoma de Madrid

UNIVERSIDAD AUTÓNOMA DE MADRID
ESCUELA POLITÉCNICA SUPERIOR

TRABAJO FIN DE MÁSTER

Procesos Gaussianos para problemas de regresión y estimación de la incertidumbre

Máster Universitario en Investigación e Innovación en TIC

Autor: DE LA POMPA PORRAS, Víctor

Tutor: DORRONSORO IBERO, José Ramón
Departamento de Ingeniería Informática

Madrid, September 3, 2018

Contents

Índice de Figuras

Índice de Tablas

Índice de Algoritmos

Resumen

En muchos problemas de regresión, surge la necesidad de no sólo predecir un valor, sino dar también un intervalo de confianza o incertidumbre, porque en el problema en cuestión las mediciones realizadas tengan ruido o las observaciones se vean influenciadas por otras que desconocemos o simplemente no podamos medir. Por ello, en este Trabajo Fin de Máster se estudiarán los procesos Gaussianos en regresión pues dan una distribución a la predición, permitiéndonos calcular intervalos de incertidumbre.

En este Trabajo Fin de Máster hemos realizado primero un estudio de algunos algoritmos de aprendizaje automático como la regresión lineal, los perceptrones multicapa, las SVR o la regresión lineal Bayesiana; esto nos ha permitido comparar y comprender mejor a los procesos Gaussianos. Posteriormente, hemos estudiado los procesos Gaussianos en regresión así como su hiperparametrización; además, hemos analizado y ampliado la teoría de los procesos Gaussianos censurados.

Experimentalmente, nos interesa ver cómo de buena es la predicción de los procesos Gaussianos comparándola con otros algoritmos de aprendizaje automático y cómo de buenos son los intervalos que obtenemos con los procesos Gaussianos. Para ello hemos realizado los experimentos en dos problemas: la predicción de energía eólica en Sotavento, que es un parque eólico situado en Galicia y la predicción de energía eólica total en España peninsular.

En resumen, hemos observado que la predicción de los procesos Gaussianos de media cero es un poco peor que la de una SVR; no obstante, hemos visto que la hiperparametrización del núcleo que hemos realizado para los procesos Gaussianos la podemos llevar a la SVR, permitiéndonos usar núcleos más complicados y reducir el coste computacional. En los intervalos de confianza, hemos visto que a los intervalos de incertidumbre del proceso Gaussiano les hacía falta una calibración y con ella obtenemos resultados buenos; también hemos visto que no merece la pena modelizar los intervalos de incertidumbre con los residuos de una SVR pues obtenemos unos resultados peores que con el proceso Gaussiano de manera directa.

Finalmente, como resultado del TFM se ha obtenido la publicación Gaussian Process Kernels for Support Vector Regression in Wind Energy Prediction aceptada en el congreso IDEAL 2018.

Abstract

In many regression problems, we need not only to predict a value, but also give a confidence interval or uncertainty, because in the problem we're trying to solve, the measurements could have noise or the target variables are influenced by variables that we do not know or we simply can not measure. For this reason, in this Master Thesis we have studied Gaussian Processes in regression since they give a distribution to the prediction, allowing us to calculate intervals of uncertainty.

In this Master Thesis we first have made a study of some machine learning algorithms such as linear regression, multilayer perceptrons, SVR or Bayesian linear regression; which has allowed us to compare and better understand Gaussian Processes. Subsequently, we have studied Gaussian Processes in regression as well as their hyperparametrization; in addition, we have analyzed and expanded the theory of censored Gaussian Processes.

Experimentally, we were interested in rating how good is the prediction of Gaussian Processes by comparing it with other machine learning algorithms and how good were the intervals we obtain with Gaussian Processes. For this, we have done the experiments in two problems: the prediction of wind energy production in Sotavento, which is a wind farm located in Galicia and the prediction of the total wind energy production in Peninsular Spain.

In summary, we have observed that the prediction of Gaussian Processes of zero mean is slightly worse than the prediction of a SVR; however, we have seen that the hyperparametrization of the kernel that we have made with Gaussian Processes, can be carried to the SVR, allowing us to use more complicated kernels and reduce the computational cost of SVR. In the confidence intervals, we have seen that the uncertainty intervals of Gaussian process needed a calibration and with it we obtained good results; we have also seen that it is not worth to model the intervals of uncertainty with the residuals of an SVR because we obtain worse results than with Gaussian process directly.

Finally, as a result of this Master Thesis it will be published the following paper: Gaussian Process Kernels for Support Vector Regression in Wind Energy Prediction which is accepted in the IDEAL 2018 conference.

Reconocimientos

Primero, agradecer a mi tutor José Ramón Dorronsoro por la oportunidad de poder investigar en un tema tan interesante, así como por revisar y dirigir este trabajo.

Agradecer también a mi compañero Alejandro Catalina Feliú por la ayuda en la publicación, así como en la discusión y exploración de los resultados de los procesos Gaussianos.

Además, este trabajo ha sido realizado gracias a la ayuda del proyecto FACIL – Ayudas Fundación BBVA a Equipos de Investigación Científica 2017 y a la Cátedra UAM-ADIC de Ciencia de Datos y Aprendizaje Automático por los datos proporcionados.

Finalmente, agradecer a mi familia y a mis compañeros por el apoyo recibido.

Capítulo 1

Introducción

1.1 Motivación del proyecto

En muchos problemas de regresión, surge la necesidad de no sólo predecir un valor, sino dar también un intervalo de confianza o incertidumbre, porque en el problema en cuestión las mediciones realizadas tengan ruido o las observaciones se vean influenciadas por variables que desconocemos o simplemente no podamos medir.

Una de las ventajes de los modelos Bayesianos es que en predicción dan una distribución, lo que nos permite obtener el intervalo de incertidumbre para la predicción; para ello en este TFM se estudiarán los procesos Gaussianos en regresión. Al suponer que la variable observada tiene distribución Gaussiana, podemos calcular los intervalos de incertidumbre con la media y desviación típica que da el modelo.

Al igual que las SVR o la Kernel Ridge Regression, los procesos Gaussianos son métodos que usan una función de núcleo, por lo que resultará interesante la comparación con los anteriores algoritmos.

Debido a causas como el cambio climático, el mundo está avanzando hacia un uso mayor de energías renovables, destacando la expansión de la energía solar y eólica. En esta área, como en todas resulta importante que la predicción de energía sea lo más precisa posible, pero también es importante proporcionar unos intervalos de incertidumbre apropiados.

Por todo esto, en este TFM vamos a comparar la predicción de los procesos Gaussianos con otros métodos más tradicionales como las SVR y los perceptrones multicapa en problemas de producción de energía eólica y además vamos a calcular los intervalos de confianza.

1.2 Objetivos

El objetivo principal del TFM es ver si con los procesos Gaussianos podemos obtener buenas predicciones e intervalos de confianza comparando los resultados con otros algoritmos de aprendizaje automático. En este trabajo se persiguen los siguientes objetivos:

- Estudiar los algoritmos de regresión lineal y no lineal tradicionales.
- Estudiar los procesos Gaussianos en regresión, así como su hiperparametrización.
- Relacionar los procesos Gaussianos con otros algoritmos de Aprendizaje Automático.

- Estudiar la aplicación de los procesos Gaussianos para problemas censurados, así como desarrollar la distribución de la variable observada censurada.
- Comparar los resultados de los procesos Gaussianos con algoritmos de Aprendizaje Automático tradicionales en dos problemas de predicción de producción de energía eólica.
- Estudiar la utilidad de los núcleos de los procesos Gaussianos cuando se usan para construir SVR sobre estos núcleos.
- Valorar los intervalos de incertidumbre que nos proporcionan los procesos Gaussianos sobre la producción de energía eólica así como sobre los residuos de un algoritmo de Aprendizaje Automático.

1.3 Estructura del documento

Además de esta introducción, el trabajo se estructura en los siguientes capítulos:

- **Capítulo 2: Regresión**, donde se describen los algoritmos de regresión lineal tradicionales, los perceptrones multicapa, las SVR, los métodos kernel y la regresión lineal Bayesiana.
- **Capítulo 3: Procesos Gaussianos en regresión**, donde se introduce y detalla los procesos Gaussianos y su hiperparametrización. Además se amplian la teoría de los procesos Gaussianos censurados, a partir del artículo de [Groot and Lucas, 2012].
- **Capítulo 4: Experimentos en energía eólica**, donde mostramos y comparamos los resultados de los procesos Gaussianos con las SVR y los perceptrones multicapa, así como los resultados de estimación de la incertidumbre para los procesos Gaussianos.
- **Capítulo 5: Conclusiones y trabajo futuro**, donde se presentan las conclusiones y el posible trabajo futuro.

Capítulo 2

Regresión

A continuación se explica brevemente la regresión lineal, las SVR y los perceptrones multicapa (MLP). También se explican los métodos kernel para la regresión lineal y la regresión lineal Bayesiana.

2.1 Regresión lineal

En esta sección se explica la regresión lineal, siguiendo el capítulo 3 de [Hastie et al., 2001]. En regresión lineal suponemos que la variable regresora ($\mathbf{y}$) es lineal con respecto a la variable x y presenta un ruido independiente con media cero y varianza σ_n^2, es decir,

$$f(x) = w_0 + x^T\mathbf{w},$$
$$y = f(x) + \varepsilon.$$

Supongamos que tenemos un conjunto $(X, \mathbf{y})$ donde $X_i = x_i$ son las variables predictivas y $\mathbf{y}_i = y_i$ es la variable a predecir. Para reducir los cálculos centramos X e $\mathbf{y}$ de forma que $w_0 = 0$ y el modelo que buscamos sea más sencillo: $f(x) = x^T\mathbf{w}$. Se puede demostrar fácilmente [Hastie et al., 2001] que el $\mathbf{w}$ que minimiza el error cuadrático:

$$e(\mathbf{w}) = \frac{1}{2}\sum_{i=1}^{N}(y_i - x_i^T\mathbf{w})^2,$$

si X^TX es invertible es $w^* = (X^TX)^{-1}X^T\mathbf{y}$. Puesto que la matriz X^TX no es siempre invertible (por ejemplo, en el caso de que haya variables correladas), es necesario añadir un regularizador; en Ridge se añade a la función de error cuadrático un término que penaliza el tamaño del vector $\mathbf{w}$ en L^2, de manera que se tiene que la función de error a minimizar es:

$$e_{\text{Ridge}}(\mathbf{w}) = \frac{1}{2}\sum_{i=1}^{N}(y_i - x_i^T\mathbf{w})^2 + \frac{\lambda}{2}\sum_{j=1}^{p}\|\mathbf{w}\|^2;$$

también se puede demostrar fácilmente que el $\mathbf{w}$ que minimiza el anterior error es (véase el capítulo 3 de [Hastie et al., 2001]),

$$\mathbf{w}^*_{\text{Ridge}} = (X^TX + \lambda I)^{-1}X^T\mathbf{y}.$$

Para ver la diferencia con la regresión lineal tradicional, vamos a hacer la descomposición SVD de la matriz X, usando que dada una matriz cualquiera podemos encontrar dos

matrices ortogonales U, V y una matriz diagonal D de forma que las columnas de U son una base del subespacio de las columnas de X y las columnas de V son una base del subespacio de las filas de X:

$$X = UDV^T;$$

por lo tanto, la predicción $\hat{\mathbf{y}} = X\mathbf{w}$ obtenida en la regresión lineal es

$$\begin{aligned}\hat{\mathbf{y}} = X\mathbf{w}^* &= X(X^TX)^{-1}X^T\mathbf{y} \\ &= UDV^T(VDU^TUDV^T)^{-1}VDU^T\mathbf{y} \\ &= UU^TY = \sum_{j=1}^{p} u_j(u_j^T\mathbf{y}),\end{aligned}$$

y en el caso de la regresión ridge es

$$\begin{aligned}X\mathbf{w}^*_{\text{Ridge}} &= (X^TX + \lambda I)^{-1}X^T\mathbf{y} \\ &= UDV^T(VD^2V^T + \lambda V^TV)^{-1}VDU^T\mathbf{y} \\ &= UDV^T(V(D^2 + \lambda I)V^T)^{-1}VDU^T\mathbf{y} \\ &= UD(D^2 + \lambda I)^{-1}DU^T\mathbf{y} \\ &= \sum_{j=1}^{p} u_j \left(\frac{d_j^2}{d_j^2 + \lambda}\right)(u_j^T\mathbf{y});\end{aligned}$$

por lo tanto, se reducirán mucho las direcciones cuyo autovalor asociado sea mucho menor que la constante λ que acompaña al regularizador, es decir, si $d_j^2/(d_j^2 + \lambda) \ll 1$ la componente u_i de la $\mathbf{y}$ será cercana a cero.

2.2 Regresión no lineal: MLP, SVR

En esta sección se describen brevemente los perceptrones multicapa y las máquinas de vectores soporte siguiendo los capítulos 11 y 12 de [Hastie et al., 2001].

2.2.1 Perceptrones multicapa

Los perceptrones multicapa (MLP) son redes neuronales con la siguiente arquitectura: una capa de entrada, una o más capas ocultas y una capa de salida con conexiones feedforward, es decir, no hay ciclos, y completamente conectadas. En cada capa se realiza una transformación de la salida de las neuronas de la capa inmediatamente anterior; si llamamos z_i^h a la salida de la neurona i-ésima de la capa h y w_{ij}^h al peso que conecta la neurona i de la capa h con la neurona j de la capa $h+1$, se tiene que

$$z_j^{h+1} = f^{h+1}\left(w_{0j}^h + \sum_{i=1}^{D} w_{ij} z_i^h\right),$$

donde $f^{h+1}(\cdot)$ es la función de activación de la capa $h+1$. En la última capa esta función de activación es la identidad para regresión ($f(x) = x$), y en las intermedias tenemos distintas opciones: sigmoide ($f(x) = (1 + \exp(-a))^{-1}$), Relu ($f(x) = \text{máx}(x, 0)$) entre otras.

Para elegir el W^* óptimo dado una muestra $(X, \mathbf{y})$, buscamos que minimice el error cuadrático medio en la muestra:

$$e(W) = \frac{1}{2N} \sum_{i=1}^{N} (y_i - g(x_i, W))^2,$$

donde $g(x, W)$ denota la salida del perceptrón multicapa. La función de error $e(W)$ no es convexa y tiene varios mínimos locales; por lo tanto, no vamos a poder calcular el W^* analíticamente y por ello, lo que se hace es realizar un descenso por gradiente mediante lotes pequeños (mini batch) aplicando el algoritmo de retropropagación o propagación hacia atras (backpropagation); para evitar el problema de caer en mínimos locales lo que se hace es repetir el proceso de entrenamiento con W iniciales distintos y calcular la media de predicción de los modelos obtenidos.

Para evitar el sobreajuste al error cuadrático medio se le suele añadir un regularizador; de manera que buscamos el W^* óptimo que minimice

$$e_R(W) = e(W) + \frac{\lambda}{2} \|W\|^2.$$

Por lo tanto, además del número de capas y el número de neuronas en cada capa, tenemos que elegir el hiperparámetro λ mediante validación cruzada.

2.2.2 Máquinas de vectores soporte para regresión

En las SVR empezamos con un modelo lineal y buscamos minimizar otra función de error regularizada:

$$\sum_{i=1}^{N} [y_i - x_i^T \mathbf{w} - w_0]_\epsilon + \frac{\alpha}{2} \|\mathbf{w}\|^2,$$

donde $[z]_\epsilon = \max(0, |z| - \epsilon)$ es la pérdida ϵ-insensitiva que penaliza los errores que caen fuera del tubo de anchura ϵ alrededor del modelo.

Este problema de optimización es convexo y tiene una única solución; no obstante, el error anterior no es diferenciable, por lo que para encontrar los parámetros óptimos, se construye el siguiente problema primal con restricciones a minimizar:

$$f(\mathbf{w}, w_0, \boldsymbol{\xi}, \boldsymbol{\eta}) = \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^{N} (\xi_i + \eta_i),$$

sujeto a

$$\begin{aligned} -\xi_i - \epsilon &\leq x_i^T \mathbf{w} + w_0 - y_i, \\ \eta_i + \epsilon &\geq x_i^T \mathbf{w} + w_0 - y_i, \\ \eta_i \geq 0, \xi_i &\geq 0, \end{aligned}$$

donde ϵ es la anchura del tubo, siendo junto a C un hiperparámetro del modelo.

Para resolver el problema anterior, se construye el Lagrangiano:

$$\begin{aligned} L(\mathbf{w}, w_0, \boldsymbol{\xi}, \boldsymbol{\eta}, \boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma}, \boldsymbol{\delta}) = \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{N}(\xi_i + \eta_i) \\ - \sum_{i=1}^{N}\alpha_i(x_i^T\mathbf{w} + w_0 - y_i + \xi_i + \epsilon) \\ + \sum_{i=1}^{N}\beta_i(x_i^T\mathbf{w} + w_0 - y_i - \eta_i - \epsilon) - \sum_{i=1}^{N}\gamma_i\xi_i - \sum_{i=1}^{N}\delta_i\eta_i, \end{aligned}$$

sujeto a

$$\alpha_i \geq 0, \beta_i \geq 0, \gamma_i \ \geq 0, \delta_i \geq 0,$$

y se define la función dual:

$$\Theta(\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma}, \boldsymbol{\delta}) = \min_{\mathbf{w}, w_0, \boldsymbol{\xi}, \boldsymbol{\eta}} L(\mathbf{w}, w_0, \boldsymbol{\xi}, \boldsymbol{\eta}, \boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma}, \boldsymbol{\delta});$$

por lo tanto, por construcción:

$$\Theta(\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma}, \boldsymbol{\delta}) \leq L(\mathbf{w}, w_0, \boldsymbol{\xi}, \boldsymbol{\eta}, \boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma}, \boldsymbol{\delta}) \leq f(\mathbf{w}, w_0, \boldsymbol{\xi}, \boldsymbol{\eta}).$$

A continuación, se define el problema dual como:

$$\max_{\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma}, \boldsymbol{\delta} \geq 0} \Theta(\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma}, \boldsymbol{\delta}),$$

que en este caso no depende de $\boldsymbol{\gamma}, \boldsymbol{\delta}$:

$$\Theta(\boldsymbol{\alpha}, \boldsymbol{\beta}) = -\frac{1}{2}\sum_{i,j}(\alpha_i - \beta_j)(\alpha_j - \beta_i)x_i^T x_j - \epsilon\sum_{i=1}^{N}(\alpha_i + \beta_i) + \sum_{i=1}^{N} y_i(\alpha_i - \beta_i),$$

sujeto a

$$\begin{aligned} 0 \leq &\alpha_i \leq C, \\ 0 \leq &\beta_i \leq C, \\ \sum_{i=1}^{N}&\alpha_i = \sum_{i=1}^{N}\beta_i. \end{aligned}$$

Si $\boldsymbol{\alpha}^*, \boldsymbol{\beta}^*$ son las soluciones del dual y $\mathbf{w}^*, w_0^*, \boldsymbol{\xi}^*, \boldsymbol{\eta}^*$ son las soluciones del primal, [Hastie et al., 2001] entonces se tiene que

$$\Theta(\boldsymbol{\alpha}^*, \boldsymbol{\beta}^*) = f(\mathbf{w}^*, w_0^*, \boldsymbol{\xi}^*, \boldsymbol{\eta}^*);$$

para los $\boldsymbol{\alpha}^*$ y $\boldsymbol{\beta}^*$ óptimos el $\mathbf{w}^*$ se calcula como:

$$\mathbf{w}^* = \sum_{i=1}^{N}(\alpha_i^* - \beta_i^*)x_i.$$

Para los i que cunplen que $0 < \alpha_i^*, \beta_i^* < C$, el w_0^* se puede obtener a partir de las siguientes ecuaciones:

$$\begin{aligned} 0 &= \alpha_i^*(x_i^T\mathbf{w}^* + w_0^* - y_i + \epsilon), \\ 0 &= \beta_i^*(x_i^T\mathbf{w}^* + w_0^* - y_i - \epsilon); \end{aligned}$$

por lo tanto, el modelo es:

$$f(x) = w_0 + \sum_{i=1}^{N} (\alpha_i^* - \beta_i^*) x_i^T x.$$

Como tanto para aplicar el modelo como para encontrar los vectores $\boldsymbol{\alpha}$ y $\boldsymbol{\beta}$ óptimos, solo tenemos que aplicar el producto escalar entre los patrones. Podemos extender el resultado a problemas no lineales usando el truco del núcleo, sustituyendo el producto escalar $x_i^T x_j$ por $k(x_i, x_j)$ donde k es una función de núcleo, que suele ser típicamente el núcleo Gaussiano: $k(x, x') = \exp(-\gamma \|x - x'\|^2)$.

2.3 Métodos Kernel

En esta sección se explica la Kernel Ridge Regression, así como la construcción de núcleos a partir de otros núcleos y los núcleos a usar en el capítulo 4, siguiendo el capítulo 6 de [Bishop, 2006].

2.3.1 Kernel Ridge Regression

En Kernel Ridge Regression, primero suponemos que tenemos una función $\phi(\cdot)$ que proyecta los datos x_i a un espacio de dimensión D', y realizamos la regresión lineal Ridge sobre este nuevo espacio, de manera que la función a minimizar pasa a ser:

$$e_{\text{Ridge}}(\mathbf{w}) = \frac{1}{2} \sum_{i=1}^{N} (y_i - \phi(x_i)^T \mathbf{w})^2 + \frac{\lambda}{2} \sum_{j=1}^{p} \|\mathbf{w}\|^2.$$

Si calculamos el gradiente de $e_{\text{Ridge}}(\mathbf{w})$ con respecto a $\mathbf{w}$ e igualamos 0, se tiene que

$$\mathbf{w} = \sum_{i=1}^{N} -\frac{1}{\lambda} \{\mathbf{w}^T \phi(x_i) - y_i\} \phi(x_i) = \Phi^T \mathbf{a},$$

donde $\mathbf{a}_i = -\frac{1}{\lambda} \{\mathbf{w}^T \phi(x_i) - y_i\} \phi(x_i)$ y $\Phi_i = \phi(x_i)$; por lo tanto, si sustituimos en $e_{\text{Ridge}}(\mathbf{w})$, el valor de $\mathbf{w}$ por la expresión anterior, se tiene que

$$\begin{aligned} e_{\text{Ridge}}(\mathbf{a}) &= \frac{1}{2} \sum_{i=1}^{N} \{\mathbf{a}^T \Phi \phi(x_i) - y_i\}^2 + \frac{\lambda}{2} \mathbf{a}^T \Phi \Phi^T \mathbf{a} \\ &= \frac{1}{2} \mathbf{a}^T \Phi \Phi^T \Phi \Phi^T \mathbf{a} - \mathbf{a}^T \Phi \Phi^T \mathbf{y} + \frac{1}{2} \mathbf{y}^T \mathbf{y} + \frac{\lambda}{2} \mathbf{a}^T \Phi \Phi^T \mathbf{a}, \end{aligned}$$

dando lugar a la representación dual de la regresión ridge. Usando el truco del núcleo,

$$k(x, x') = \phi(x)^T \phi(x'),$$

y definiendo $K = \Phi \Phi^T$, es decir, $K_{nm} = \phi(x_n)^T \phi(x_m) = k(x_n, x_m)$ se tiene que

$$\begin{aligned} e_{\text{Ridge}}(\mathbf{a}) &= \frac{1}{2} \mathbf{a}^T \Phi \Phi^T \Phi \Phi^T \mathbf{a} - \mathbf{a}^T \Phi \Phi^T \mathbf{y} + \frac{1}{2} \mathbf{y}^T \mathbf{y} + \frac{\lambda}{2} \mathbf{a}^T \Phi \Phi^T \mathbf{a} \\ &= \frac{1}{2} \mathbf{a}^T K K \mathbf{a} - \mathbf{a}^T K \mathbf{y} + \frac{1}{2} \mathbf{y}^T \mathbf{y} + \frac{\lambda}{2} \mathbf{a}^T K \mathbf{a}; \end{aligned}$$

por lo tanto, si calculamos el gradiente de $e_{\text{Ridge}}(\mathbf{a})$ e igualamos a cero se tiene que

$$0 = \nabla e_{\text{Ridge}}(\mathbf{a}^*) = KK\mathbf{a}^* - K\mathbf{y} + \lambda K\mathbf{a}^*,$$

esto es

$$(KK + \lambda K)\mathbf{a}^* = K\mathbf{y},$$

y por lo tanto,

$$\mathbf{a}^* = (K + \lambda I_N)^{-1}\mathbf{y}.$$

Finalmente, si sustituimos el valor óptimo de $\mathbf{a}$ en el modelo de regresión se tiene que

$$\begin{aligned} y(\hat{x}) &= \phi(\hat{x})^T\mathbf{w}^* = \phi(\hat{x})^T\Phi^T\mathbf{a}^* \\ &= (k(x_1, \hat{x}), ..., k(x_n, \hat{x}))(K + \lambda I_N)^{-1}\mathbf{y}; \end{aligned}$$

por lo tanto, no es necesario calcular ni saber la función $\phi(\cdot)$, basta con usar una función de núcleo apropiada.

2.3.2 Construcción de núcleos

A lo largo de esta subsección se verán distintas formas de construir funciones de núcleo, así como una breve descripción de los núcleos usados en los experimentos con los procesos Gaussianos.

Una manera natural de construir núcleos es elegir una transformación a un espacio de características $\phi(x)$ y definir el núcleo como:

$$k(x, x') = \phi(x)^T\phi(x').$$

También podemos construir funciones de núcleo directamente. Por ejemplo: sean $x, x' \in \mathbb{R}^2$, definimos

$$\begin{aligned} k(x, x') &= (x^T x')^2 = (x_1x'_1 + x_2x'_2)^2 \\ &= x_1^2(x'_1)^2 + 2x_1x'_1x_2x'_2 + x_2^2(x'_2)^2 \\ &= (x_1, \sqrt{2}x_1x_2, x_2)((x'_1)^2, \sqrt{2}x'_1x'_2, (x'_2)^2)^T; \end{aligned}$$

o construir funciones de núcleo a partir de modelos generadores probabilísticos, de forma que x y x' son similares si tienen ambos probabilidad alta, por ejemplo:

$$\begin{aligned} k(x, x') &= p(x)p(x'), \\ k(x, x') &= \sum_i p(x|i)p(x'|i)p(i), \\ k(x, x') &= \int p(x|z)p(x'|z)p(z)dz. \end{aligned}$$

donde $p(\cdot)$ es la función de densidad si la variable es continua, o la función de probabilidad $p(i) = P(I = i)$ si la variable I es discreta.

Además, podemos construir funciones de núcleo a partir de otros; dados tres núcleos k_1, k_2 y k_3, se tiene que las siguientes funciones son núcleos (véase capítulo 6 de [Bishop, 2006]):

$$k(x,x') = ck_1(x,x') \text{ con c una constante no negativa, es decir, } c>0, \tag{2.3.1}$$
$$k(x,x') = f(x)k_1(x,x')f(x'),\ f(x) \text{ cualquier función}, \tag{2.3.2}$$
$$k(x,x') = q(k_1(x,x')),\ q(x) \text{ polinomio con coeficientes no negativos}, \tag{2.3.3}$$
$$k(x,x') = \exp(k_1(x,x')), \tag{2.3.4}$$
$$k(x,x') = k_1(x,x') + k_2(x,x'), \tag{2.3.5}$$
$$k(x,x') = k_1(x,x')k_2(x,x'), \tag{2.3.6}$$
$$k(x,x') = k_3(\phi(x),\phi(x')), \text{donde } \phi(\cdot) \text{ es una proyección a un nuevo espacio} \tag{2.3.7}$$
$$k(x,x') = x^T Ax \text{ donde } A \text{ es una matriz simétrica y definida positiva}, \tag{2.3.8}$$
$$k(x,x') = k_1(x_{\mathcal{F}_1},x'_{\mathcal{F}_1}) + k_2(x_{\mathcal{F}_2},x'_{\mathcal{F}_2}),\ \text{donde } \mathcal{F}_i \text{ son subespacios de } \mathbb{R}^D \tag{2.3.9}$$
$$k(x,x') = k_1(x_{\mathcal{F}_1},x'_{\mathcal{F}_1})k_2(x_{\mathcal{F}_2},x'_{\mathcal{F}_2}). \tag{2.3.10}$$

El núcleo RBF Gaussiano se puede ver como el producto escalar en un espacio $\phi(\mathbb{R}^N)$ de infinita dimensión, pero aquí lo vamos a deducir usando las propiedades anteriores:

$$\begin{aligned} k_{\text{RBF}}(x,x') &= \exp\left(-\frac{\|x-x'\|^2}{2\ell^2}\right) = \exp\left(-\frac{\langle x-x', x-x'\rangle}{2\ell^2}\right) \\ &= \exp\left(-\frac{\langle x,x\rangle}{2\ell^2}\right)\exp\left(\frac{\langle x,x'\rangle}{\ell^2}\right)\exp\left(-\frac{\langle x',x'\rangle}{2\ell^2}\right); \end{aligned} \tag{2.3.11}$$

si llamamos $k_1(x,x') = \langle x,x'\rangle = x^Tx'$ y $f(x) = \exp\left(-\frac{\langle x,x\rangle}{2\ell^2}\right)$ tenemos que

$$k_{\text{RBF}}(x,x') = f(x)\exp\left(\frac{k_1(x,x')}{\ell^2}\right)f(x');$$

por lo tanto, usando (2.3.1), (2.3.4) y (2.3.2) tenemos que k_{RBF} es un núcleo. Otro de los núcleos que hemos usado es el núcleo Rational Quadratic (RQ), el cual tiene la siguiente forma:

$$k_{\text{RQ}}(x,x') = \left(1+\frac{\|x-x'\|^2}{2\alpha\ell^2}\right)^{-\alpha},$$

donde $\alpha>0$ y $\ell>0$; este núcleo se puede ver como una suma infinita de núcleos RBF cada uno de ellos con la variable ℓ de escala distinta, véase capítulo 4 de [Rasmussen and Williams, 2004]. También hemos usado el núcleo Matérn, el cual debe su nombre al trabajo realizado en [Matérn, 1960], cuya expresión es la siguiente:

$$k_{\text{Matérn}}(x,x') = \frac{2^{1-\nu}}{\Gamma(\nu)}\left(\frac{\sqrt{2\nu}\|x-x'\|}{\ell^2}\right)^{\nu} K_\nu\left(\frac{\sqrt{2\nu}\|x-x'\|}{\ell^2}\right),$$

donde $\nu>0$, $\ell>0$, $\Gamma(\cdot)$ es la función gamma y K_ν es una función de Bessel de segunda especie modificada; la constante ν controla cómo de derivable es el proceso Gaussiano con dicho núcleo, véase capítulo 4 de [Rasmussen and Williams, 2004]. Además se cumple que el núcleo Matérn es igual al RBF cuando $\nu\to\infty$. Se puede demostrar que para $\nu = n+1/2$ con $n\in\mathbb{N}$ el núcleo Matérn, se simplifica y se tiene que :

$$\begin{aligned} k^{\nu=n+1/2}_{\text{Matérn}}(x,x') = &\exp\left(-\frac{\sqrt{2(n+1/2)}\|x-x'\|}{\ell^2}\right)\frac{\Gamma(n+1)}{\Gamma(2n+1)}\times \\ &\sum_{i=0}^{n}\frac{(n+i)!}{i!(n-i)!}\left(\frac{\sqrt{8(n+1/2)}\|x-x'\|}{\ell^2}\right)^{n-i}; \end{aligned}$$

véase capítulo 4 de [Rasmussen and Williams, 2004] y sección 9.6 de [Abramowitz et al., 1965]. En particular para $n = 0, 1, 2$ se tiene que:

$$k^{1/2}_{\text{Matérn}}(x, x') = \exp\left(-\frac{\|x - x'\|}{\ell^2}\right),$$
$$k^{3/2}_{\text{Matérn}}(x, x') = \left(1 + \frac{\sqrt{3}\|x - x'\|}{\ell^2}\right)\exp\left(-\frac{\sqrt{3}\|x - x'\|}{\ell^2}\right),$$
$$k^{5/2}_{\text{Matérn}}(x, x') = \left(1 + \frac{\sqrt{5}\|x - x'\|}{\ell^2} + \frac{5\|x - x'\|^2}{\ell^4}\right)\exp\left(-\frac{\sqrt{5}\|x - x'\|}{\ell^2}\right).$$

2.4 Regresión lineal Bayesiana

A continuación se explica la regresión lineal Bayesiana, así como la deducción de los procesos Gaussianos con media cero vistos como una distribución en los pesos.

2.4.1 Regresión lineal con priores Gaussianos

Primero vamos a introducir la regresión lineal Bayesiana; en ella primero suponemos que la variable regresora ($\mathbf{y}$) es lineal con respecto a la variable x y presenta un ruido Gaussiano independiente, es decir,

$$\begin{aligned} y_i &= f(x_i) + \varepsilon_i, \\ f(x_i) &= x_i^T \mathbf{w}, \\ \varepsilon_i &\sim \mathcal{N}(0, \sigma_n^2). \end{aligned}$$

La distribución de $\mathbf{y}|X, \mathbf{w}$ es Gaussiana, pues $y_i|x_i, \mathbf{w} \sim \mathcal{N}(0, \sigma_n^2)$ y

$$\begin{aligned} p(\mathbf{y}|X, \mathbf{w}) &= \prod_{i=1}^{N} p(y_i|x_i, \mathbf{w}) \\ &= \prod_{i=1}^{N} \frac{1}{\sqrt{2\pi}\sigma_n} \exp\left(-\frac{(y_i - x_i^T\mathbf{w})^2}{2\sigma_n^2}\right) \\ &= \frac{1}{(2\pi\sigma_n^2)^{N/2}} \exp\left(-\frac{1}{2\sigma_n^2}\|\mathbf{y} - X\mathbf{w}\|^2\right); \end{aligned}$$

por lo tanto, $\mathbf{y}|X, \mathbf{w} \sim \mathcal{N}(X\mathbf{w}, \sigma_n^2 I)$.

Hasta ahora no hemos hecho ninguna suposición distinta de las tradicionales en regresión. Ahora, si suponemos un prior Gaussiano en las $\mathbf{w}$, es decir, $\mathbf{w} \sim \mathcal{N}(0, \Sigma_D)$ y usamos la regla Bayes para funciones de densidad, tenemos que

$$p(\mathbf{w}|X, \mathbf{y}) = \frac{p(\mathbf{y}|X, \mathbf{w})p(\mathbf{w})}{p(\mathbf{y}|X)}.$$

Para calcular la distribución de $\mathbf{w}|X, \mathbf{y}$, vamos a quitar los términos que no dependen de $\mathbf{w}$ para que los cálculos sean más sencillos, pues $p(\mathbf{w}|X, \mathbf{y})$ es una función de densidad y

debe integrar a uno:

$$\begin{aligned}
p(\mathbf{w}|X,\mathbf{y}) &\propto p(\mathbf{y}|X,\mathbf{w})p(\mathbf{w}) \\
&\propto \exp\left\{-\frac{1}{2}\left[\sigma_n^{-2}(\mathbf{y}-X\mathbf{w})^T(\mathbf{y}-X\mathbf{w})+\mathbf{w}^T\Sigma_D^{-1}\mathbf{w}\right]\right\} \\
&= \exp\left\{-\frac{1}{2}(\mathbf{w}^T(\sigma_n^{-2}X^TX+\Sigma_D^{-1})\mathbf{w}+2\sigma_n^{-2}\mathbf{w}^TX^T\mathbf{y}+\sigma_n^{-2}\mathbf{y}^T\mathbf{y})\right\} \\
&\propto \exp\left\{-\frac{1}{2}(\mathbf{w}^TA\mathbf{w}+2\sigma_n^{-2}(\mathbf{w}^TAA^{-1}X^T\mathbf{y}))\right\} \\
&\propto \exp\left\{-\frac{1}{2}(\mathbf{w}-\sigma_n^{-2}A^{-1}X^T\mathbf{y})^TA(\mathbf{w}-\sigma_n^{-2}A^{-1}X^T\mathbf{y})\right\},
\end{aligned}$$

donde $A=(\sigma_n^{-2}X^TX+\Sigma_D^{-1})$; por lo tanto, $\mathbf{w}|X,\mathbf{y}$ sigue una distribución Gaussiana,

$$\mathbf{w}|X,y\sim\mathcal{N}(\frac{1}{\sigma_n^2}A^{-1}X^T\mathbf{y},A^{-1}).$$

Dado un nuevo dato $\hat{x}$ la función de distribución de la predicción $y(\hat{x})$ es un promedio de los distintos modelos lineales resultantes al variar $\mathbf{w}$ con respecto a su posterior gaussiano, es decir,

$$p(y(\hat{x})|\hat{x},X,\mathbf{y})=\int p(y(\hat{x}),\mathbf{w}|\hat{x},X,\mathbf{y})d\mathbf{w}=\int p(y(\hat{x})|\hat{x},\mathbf{w})p(\mathbf{w}|X,\mathbf{y})d\mathbf{w}. \tag{2.4.1}$$

Para calcular la integral anterior basta usar el siguiente lema y la siguiente proposición, los cuales son fáciles de probar:

Lema 1. *Si*

$$\begin{bmatrix}a\\b\end{bmatrix}\sim\mathcal{N}\left(\begin{bmatrix}\mu_a\\\mu_b\end{bmatrix},\begin{bmatrix}\Sigma_a & C\\C^T & \Sigma_b\end{bmatrix}\right),$$

entonces se tiene que

$$p(b|a)=\frac{p(a,b)}{p(a)},$$

que en este caso se reduce a

$$b|a\sim\mathcal{N}(\mu_b+C^T\Sigma_a^{-1}(a-\mu_a),\Sigma_b-C^T\Sigma_a^{-1}C)=\mathcal{N}(\mu_{b|a},\Sigma_{b|a}).$$

Proposicion 1. *Si consideramos $\mu_{b|a}=C^T\Sigma_a^{-1}a+c,\Sigma_{b|a}$ para calcular*

$$p(b)=\int p(a,b)da=\int p(b|a)p(a)da,$$

se cumple que $b\sim\mathcal{N}(\mu_b,\Sigma_b)$ donde:

$$\mu_b=\mu_{b|a}-C^T\Sigma_a^{-1}(a-\mu_a)=C^T\Sigma_a^{-1}\mu_a+c, \tag{2.4.2}$$

$$\Sigma_b=\Sigma_{b|a}+C^T\Sigma_a^{-1}C=\Sigma_{b|a}+(C^T\Sigma_a^{-1})\Sigma_a(C^T\Sigma_a^{-1})^T. \tag{2.4.3}$$

Usando la Proposición 1 en (2.4.1) con $a=\mathbf{w}|X,\mathbf{y}$ y $b|a=y(\hat{x})|\hat{x},\mathbf{w}$ y teniendo en cuenta que

$$\begin{aligned}
\mathbf{w}|X,\mathbf{y}&\sim\mathcal{N}(\frac{1}{\sigma_n^2}A^{-1}X^T\mathbf{y},A^{-1}),\\
y(\hat{x})|\hat{x},\mathbf{w}&\sim\mathcal{N}(\hat{x}^T\mathbf{w},\sigma_n^2),
\end{aligned}$$

se sigue que

$$y(\hat{x})|\hat{x}, X, \mathbf{y} \sim \mathcal{N}(\hat{x}^T \frac{1}{\sigma_n^2} A^{-1} X^T \mathbf{y}, \sigma_n^2 + \hat{x}^T A^{-1} \hat{x});$$

además, como se cumple que $y(\hat{x}) = f(\hat{x}) + \varepsilon$, donde $\varepsilon \sim \mathcal{N}(0, \sigma_n^2)$, se tiene que

$$\mathbb{E}_{y(\hat{x})}(y(\hat{x})|\hat{x}, X, \mathbf{y}) = \mathbb{E}_{f(\hat{x})}(f(\hat{x})|\hat{x}, X, \mathbf{y}) + \mathbb{E}_{\varepsilon}(\varepsilon) = \hat{x}^T \frac{1}{\sigma_n^2} A^{-1} X^T \mathbf{y}$$
$$\mathbb{V}_{y(\hat{x})}(y(\hat{x})|\hat{x}, X, \mathbf{y}) = \mathbb{V}_{f(\hat{x})}(f(\hat{x})|\hat{x}, X, \mathbf{y}) + \mathbb{V}_{\varepsilon}(\varepsilon) = \hat{x}^T A^{-1} \hat{x} + \sigma_n^2,$$

y se sigue que

$$f(\hat{x})|\hat{x}, X, \mathbf{y} \sim \mathcal{N}(\hat{x}^T \frac{1}{\sigma_n^2} A^{-1} X^T \mathbf{y}, \hat{x}^T A^{-1} \hat{x}).$$

2.4.2 Regresión lineal Bayesiana sobre un espacio proyectado

Hasta ahora no hemos visto ninguna función de núcleo y la regresión se realizaba con la matriz X, pero hay muchas veces que queremos proyectar los datos a una dimensión mayor sin tener que calcularla; para ello utilizamos las funciones de núcleo. Supongamos que conocemos dicha proyección $\phi(x)$ y que nuestro modelo es

$$f(x) = \phi(x)^T \mathbf{w};$$

por lo tanto, si hacemos la regresión Bayesiana en el nuevo espacio $\phi(x)$ de dimensión D', donde suponemos que $\mathbf{w} \sim \mathcal{N}(0, \Sigma_{D'})$, tenemos para un nuevo dato $\hat{x}$ la siguiente predicción:

$$f(\hat{x})|\hat{x}, X, \mathbf{y} \sim \mathcal{N}\left(\frac{1}{\sigma_n^2} \phi(\hat{x})^T A^{-1} \Phi^T \mathbf{y}, \phi(\hat{x})^T A^{-1} \phi(\hat{x})\right), \tag{2.4.4}$$

donde $\Phi = \phi(X)$ y $A = (\sigma_n^{-2} \Phi^T \Phi + \Sigma_{D'}^{-1})$; podemos ver que surge un problema y es que si la dimensión del espacio proyectado es D' y es muy grande, hay que invertir la matriz A que tiene dimensión $D' \times D'$, y el coste de invertir una matriz es cúbico.

Una forma de solucionar este problema es usar el truco del núcleo (kernel trick), suponiendo ahora

$$k(x_i, x_j) = \phi(x_i)^T \Sigma_{D'} \phi(x_j),$$
$$K = \Phi \Sigma_{D'} \Phi^T.$$

Si usamos las definiciones de A y K se obtiene la siguiente igualdad:

$$\begin{aligned} \frac{1}{\sigma_n^2} \Phi^T (K + \sigma_n^2 I) &= \frac{1}{\sigma_n^2} \Phi^T \Phi \Sigma_{D'} \Phi^T + \Sigma_{D'}^{-1} \Sigma_{D'} \Phi^T \\ &= (\frac{1}{\sigma_n^2} \Phi^T \Phi + \Sigma_{D'}^{-1}) \Sigma_{D'} \Phi^T \\ &= A \Sigma_{D'} \Phi^T; \end{aligned}$$

por lo tanto, si multiplicamos por A^{-1} a la izquierda y por $(K + \sigma_n^2 I)^{-1}$ a la derecha en ambos lados de la igualdad, se tiene que

$$\frac{1}{\sigma_n^2} A^{-1} \Phi^T = \Sigma_{D'} \Phi^T (K + \sigma_n^2 I)^{-1}.$$

Si usamos este resultado en (2.4.4), tenemos entonces que la media de $f(\hat{x})|\hat{x}, X, \mathbf{y}$ es

$$\frac{1}{\sigma_n^2}\phi(\hat{x})^T A^{-1}\Phi^T\mathbf{y} = \phi(\hat{x})^T\Sigma_{D'}\Phi^T(K+\sigma_n^2 I)^{-1}\mathbf{y}$$

y si aplicamos el truco del núcleo denotando como $k(X,\hat{x})$ el vector cuyas componentes son $k(x_i,\hat{x})$ se tiene que

$$\mathbb{E}_{f(\hat{x})}(f(\hat{x})|\hat{x}, X, \mathbf{y}) = k(X,\hat{x})^T(K+\sigma_n^2 I)^{-1}\mathbf{y}.$$

Para la varianza, vamos a aplicar la identidad de Woodbury(véase apéndice A.3 de [Rasmussen and Williams, 2004]),

$$(Z+UWV^T)^{-1} = Z^{-1} - Z^{-1}U(W^{-1}+V^TZ^{-1}U)^{-1}V^TZ^{-1}.$$

Sustituyendo $Z^{-1}=\Sigma_{D'}, W^{-1}=\sigma_n^2 I$ y $V=U=\Phi^T$, se tiene

$$\begin{aligned} A^{-1} &= (\Sigma_{D'}^{-1}+\Phi^T\sigma_n^{-2}I\Phi)^{-1} \\ &= \Sigma_{D'} - \Sigma_{D'}\Phi^T(\sigma_n^2 I+K)^{-1}\Phi\Sigma_{D'}; \end{aligned}$$

por lo tanto, se sigue que

$$\begin{aligned} \phi(\hat{x})^T A^{-1}\phi(x) &= \phi(\hat{x})^T\Sigma_{D'}\phi(\hat{x}) - \phi(\hat{x})^T\Sigma_{D'}\Phi^T(\sigma_n^2 I+K)^{-1}\Phi\Sigma_{D'}\phi(\hat{x}) \\ &= k(\hat{x},\hat{x}) - k(X,\hat{x})^T(\sigma_n^2 I+K)^{-1}k(X,\hat{x}). \end{aligned}$$

Finalmente, tenemos que

$$\begin{aligned} f(\hat{x})|\hat{x}, X, \mathbf{y} \sim \mathcal{N}(&k(\hat{x},X)^T(K+\sigma_n^2 I)^{-1}\mathbf{y}, \\ &k(\hat{x},\hat{x}) - k(X,\hat{x})^T(\sigma_n^2 I+K)^{-1}k(X,\hat{x})) \end{aligned}$$

Podemos ver que ahora la matriz que hay que invertir, $(\sigma_n^2 I+K)$, tiene dimensión N, lo cual es conveniente si $N < D'$. Además, tanto en la media como en la matriz de covarianzas no es necesario calcular la función $\phi(\cdot)$; simplemente hay que calcular la matriz de núcleo K y $\phi(\hat{x})^T\Sigma_{D'}\Phi^T = k(X,\hat{x})^T$; por lo tanto, no es necesario calcular ni saber la proyección $\phi(\cdot)$ ni $\Sigma_{D'}$, y podemos usar directamente funciones de núcleo.

Capítulo 3

Procesos Gaussianos en regresión

A continuación se explican los procesos Gaussianos en regresión. Para ello primero se realiza una introducción a los mismos, posteriormente se muestra cómo hacer la estimación de la predicción con una función de covarianzas fija y finalmente se explica cómo obtener los hiperparámetros óptimos [Rasmussen and Williams, 2004] y los procesos Gaussianos censurados [Groot and Lucas, 2012].

3.1 Introducción general

Un proceso Gaussiano es una colección de variables aleatorias, que cumplen que cualquier subconjunto finito de la colección, tiene una distribución Gaussiana. Es decir, sea $\mathbf{Y} = \{y_i\}_{i \in I}$ una colección de variables aleatorias, donde I es un conjunto de índices; si $\mathbf{Y}$ es un proceso Gaussiano, entonces para todo subconjunto finito de índices $\{i_1, i_2, ..., i_n\} \subset I$, se tiene que el vector $(y_{i_1}, y_{i_2}, ..., y_{i_n})^T \sim \mathcal{N}_n(\mu, \Sigma)$ para algún vector $\mu = \mu(i_1, ..., i_N) \in \mathbb{R}^n$ y alguna matriz simétrica y semidefinida positiva $\Sigma = \Sigma(i_1, ..., i_N) \in M_n(\mathbb{R})$.

Debido a que una distribución Gaussiana viene determinada por un vector de medias y una matriz de covarianzas, un proceso Gaussiano queda determinado por la función de media y de covarianzas; esto es con un pequeño abuso del lenguaje:

$$\begin{aligned} f(x) &\sim GP(m(x), k(x, x')), \\ m(x) &= \mathbb{E}[f(x)], \\ k(x, x') &= \mathbb{E}[(f(x) - m(x)(f(x') - m(x'))] = \mathrm{Cov}(x, x'). \end{aligned}$$

Podemos ver un ejemplo de proceso Gaussiano en el ruido Gaussiano independiente, en el cual se supone que

$$\begin{aligned} \varepsilon(x_i) = \varepsilon_i &\sim \mathcal{N}(0, \sigma^2), \\ \mathrm{Cov}(\varepsilon_i, \varepsilon_j) &= 0 \text{ si } i \neq j; \end{aligned}$$

por lo tanto, se tiene que $\varepsilon(x) \sim GP(0, \sigma^2 \delta_{xx'})$. Podemos ver otro ejemplo de proceso Gaussiano en la regresión lineal bayesiana estudiada en la sección 2.4, en la cual se supone una distribución a priori Gaussiana en los pesos, es decir, $\mathbf{w} \sim \mathcal{N}(0, \Sigma_D)$, donde D es la dimensión de x. En este modelo, la variable aleatoria predictora ($f(x) = x^T \mathbf{w}$) es un proceso Gaussiano, cuya función de media y covarianza viene dada como

$$\begin{aligned} \mathbb{E}[f(x)] &= x^T \mathbb{E}(\mathbf{w}) = 0, \\ \mathrm{Cov}(f(x), f(x')) &= x^T \mathbb{E}(\mathbf{w}\mathbf{w}^T) x' = x^T \Sigma_D x'. \end{aligned}$$

Como hemos visto, un proceso Gaussiano, viene determinado por la función de media, $m(x)$, y la de covarianzas, $k(x, x')$; veamos qué restricciones deben de cumplir dichas funciones. Sea $\{x_n\}_{n=1}^N$ un conjunto de puntos en $\mathbb{R}^D$, y sea $f(x) \sim GP(m(x), k(x, x'))$, donde $x, x' \in \mathbb{R}^D$; usando la definición de proceso Gaussiano se tiene que

$$(f(x_1), f(x_2), ..., f(x_N))^T \sim \mathcal{N}_N(\mathbf{m}, K),$$

con $K_{nm} = k(x_n, x_m)$ y $\mathbf{m}_n = m(x_n)$. Para que la expresión anterior tenga sentido K tiene que ser una matriz de covarianzas, es decir, simétrica y semidefinida positiva. Esto quiere decir, que la función $k(x, x')$ no puede ser cualquier función: debe garantizar que se obtienen matrices semidefinidas positivas y simétricas; es decir, ha de ser una función de núcleo (kernel). Por otro lado, no hay ninguna restricción para la función de media $m(x)$, aunque generalmente para aligerar la notación y reducir los desarollos matemáticos se toma como cero.

3.2 Estimación

En esta sección se muestra cómo obtener la distribución de predicción de un nuevo ejemplo, a partir de los ejemplos de entrenamiento, teniendo en cuenta que la función de covarianzas es fija siguiendo el capítulo 2 de [Rasmussen and Williams, 2004]. El cálculo de los hiperparámetros óptimos del núcleo de la función de covarianza se detalla en la sección 3.3.

En los problemas de regresión se suele considerar que las observaciones obtenidas presentan un ruido Gaussiano independiente, es decir,

$$y(x) = f(x) + \varepsilon \text{ con } \varepsilon \sim \mathcal{N}(0, \sigma_n^2).$$

Si suponemos que $f(x) \sim GP(m(x), k(x, x'))$, se tiene entonces que las observaciones también son un proceso Gaussiano, con la misma función de media, pero cuya función de covarianza es la suma de la covarianza de $f(x)$ con la del ruido independiente, es decir,

$$y(x) \sim GP(m(x), k(x, x') + \sigma_n^2 \delta_{xx'}).$$

Para aligerar la notación y reducir los desarrollos matemáticos, supongamos que la función de medias es cero ($\forall x\ m(x) = 0$); como $y(x)$ y $f(x)$ son procesos Gaussianos, la distribución conjunta de las observaciones en entrenamiento $(X, \mathbf{y})$ con las objetivo de predicción $(\hat{X}, \hat{\mathbf{f}})$ es una normal multidimensional:

$$\begin{bmatrix} \mathbf{y} \\ \hat{\mathbf{f}} \end{bmatrix} \sim \mathcal{N}\left(0, \begin{bmatrix} K(X, X) + \sigma_n^2 I_N & K(X, \hat{X}) \\ K(\hat{X}, X) & K(\hat{X}, \hat{X}) \end{bmatrix}\right);$$

donde X es una matriz de tamaño $N \times D$, siendo N el número de ejemplos en entrenamiento y D la dimensión en la que viven los datos, $\hat{X}$ es una matriz de tamaño $M \times D$, siendo M el número de ejemplos a predecir; $\mathbf{y}$ es un vector columna de tamaño N y $\hat{\mathbf{f}}$ es un vector columna de tamaño M. Por lo tanto, $K(X, X)$ es una matriz de tamaño $N \times N$, $K(X, \hat{X})$ es una matriz de tamaño $N \times M$, $K(\hat{X}, X)$ es una matriz de tamaño $M \times N$ y $K(\hat{X}, \hat{X})$ es una matriz de tamaño $M \times M$. Además se cumple que $K(X, \hat{X}) = K(\hat{X}, X)^T$ pues la función de núcleo es simétrica.

Usando el Lema 1, podemos calcular la distribución de $\hat{\mathbf{f}}$ condicionando por $X, \hat{X}, \mathbf{y}$, teniéndose que

$$\begin{aligned}\hat{\mathbf{f}}|\hat{X}, X, \mathbf{y} \sim \mathcal{N}(&K(\hat{X}, X)(K(X, X) + \sigma_n^2 I_N)^{-1}\mathbf{y}, \\ &K(\hat{X}, \hat{X}) - K(\hat{X}, X)(K(X, X) + \sigma_n^2 I_N)^{-1}K(X, \hat{X}));\end{aligned} \tag{3.2.1}$$

además, la distribución conjunta de las observaciones en entrenamiento ($\mathbf{y}$) y de las de test con ruido ($\hat{\mathbf{y}}$) es también una normal multidimensional,

$$\begin{bmatrix}\mathbf{y} \\ \hat{\mathbf{y}}\end{bmatrix} \sim \mathcal{N}\left(0, \begin{bmatrix}K(X, X) + \sigma_n^2 I_N & K(X, \hat{X}) \\ K(\hat{X}, X) & K(\hat{X}, \hat{X}) + \sigma_n^2 I_M\end{bmatrix}\right).$$

Por lo tanto, la distribución de $\hat{\mathbf{y}}|\hat{X}, X, \mathbf{y}$ es también normal con:

$$\begin{aligned}\mathbb{E}_{\hat{\mathbf{y}}}(\hat{\mathbf{y}}|\hat{X}, X, \mathbf{y}) &= \mathbb{E}_{\hat{\mathbf{f}}}(\hat{\mathbf{f}}|\hat{X}, X, \mathbf{y}), \\ \mathrm{Cov}(\hat{\mathbf{y}}|\hat{X}, X, \mathbf{y}) &= \mathrm{Cov}(\hat{\mathbf{f}}|\hat{X}, X, \mathbf{y}) + \sigma_n^2 I_M.\end{aligned} \tag{3.2.2}$$

Para el caso en el que hay un único ejemplo en test $(\hat{x}, \hat{f})$, renombrando

$$\begin{aligned}K(X, X) = K, \mathbf{k} = k(X, \hat{x}) = k(\hat{x}, X)^T, \\ \alpha = (K + \sigma_n^2 I)^{-1}\mathbf{y},\end{aligned}$$

se tiene que

$$\begin{aligned}\mathbb{E}_{\hat{f}}[\hat{f}|\hat{x}, X, \mathbf{y}] &= \mathbf{k}^T(K + \sigma_n^2 I)^{-1}\mathbf{y} = \sum_{i=1}^{N} \alpha_i k(x_i, \hat{x}), \\ \mathbb{V}_{\hat{f}}[\hat{f}|\hat{x}, X, \mathbf{y}] &= k(\hat{x}, \hat{x}) - \mathbf{k}^T(K + \sigma_n^2 I)^{-1}\mathbf{k}.\end{aligned} \tag{3.2.3}$$

Con esta notación se puede ver que la media de $\hat{f}|\hat{x}, X, \mathbf{y}$ viene determinada por los puntos de entrenamiento y el de test. También se observa que el valor de la media obtenido es el mismo valor que predice una regresión Ridge, con el truco del núcleo (Kernel Ridge Regression) con $\lambda = \sigma_n^2$ visto en la sección 2.3, para una comparación más extensa entre estos dos métodos véase [Kanagawa et al., 2018].

Además, se puede ver que la varianza depende de las entradas, del ruido (σ_n^2), pero no de las observaciones ($\mathbf{y}$); en esta varianza se pueden identificar dos términos: un término positivo $k(\hat{x}, \hat{x})$ que es la covarianza y un segundo término negativo que representa la "información" que dan las variables de los ejemplos de entrenamiento de la función.

3.2.1 Suavizadores lineales

Los procesos Gaussianos sirven también como suavizadores lineales, pues como se ha visto, en predicción se tiene una distribución para la variable regresora con ruido (y) y para la variable sin ruido o señal (f) .

De manera que si realizamos los cálculos anteriores con $\hat{X} = X$, se tiene que la distribución conjunta de las observaciones $\mathbf{y} = y(X)$, $\mathbf{f} = f(X)$ es Gaussiana y la distribución condicionada $\mathbf{f}|\mathbf{y}, X$ tiene la siguiente distribución:

$$\mathbf{f}|\mathbf{y}, X \sim \mathcal{N}(K(K + \sigma_n^2 I)^{-1}\mathbf{y}, K - K(K + \sigma_n^2 I)^{-1}K).$$

Para ver las propiedades del proceso Gaussiano como suavizador, hay que notar que K es una matriz real, semidefinida positiva y simétrica; por lo tanto, podemos realizar la descomposición SVD, teniendo en cuenta que los elementos de la matriz diagonal son no negativos,

$$K = UDU^T = \sum_{i=1}^{N} \lambda_i u_i u_i^T \text{ con } \lambda_i = d_{ii} \geq 0,$$

y donde $\{u_1, ..., u_N\}$ forman una base ortogonal; por lo tanto, se puede expresar $\mathbf{y}$ en esta base:

$$\mathbf{y} = \sum_{i=1}^{N} \gamma_i u_i = U\gamma \text{ con } \gamma_i = u_i^T \mathbf{y}.$$

Si aplicamos estos resultados a la media de $\mathbf{f}|\mathbf{y}, X$, se obtiene que

$$\begin{aligned}
\mathbb{E}(\mathbf{f}|\mathbf{y}, X) &= UDU^T(U(D + \sigma_n^2 I)U^T)^{-1}U\gamma \\
&= UDU^T U(D + \sigma_n^2 I)^{-1}U^T U\gamma \\
&= UD(D + \sigma_n^2 I)^{-1}\gamma \\
&= \sum_{i=1}^{N} \frac{\gamma_i \lambda_i}{\lambda_i + \sigma_n^2} u_i.
\end{aligned}$$

Por lo tanto, se reducirán mucho las direcciones cuyo autovalor asociado sea mucho menor que el ruido; es decir, si $\lambda_i/(\lambda_i + \ \sigma_n^2) \ll 1$ la componente u_i de la predicción $\mathbf{y}$ tenderá a cero. Estos resultados son muy parecidos a los que se obtienen en la regresión ridge vistos en la sección 2.1. [Hastie et al., 2001] El número efectivo de parámetros, es decir, el número de grados de libertad del suavizador se define allí como

$$\sum_{i=1}^{N} \frac{\lambda_i}{\lambda_i + \sigma_n^2} = \text{traza}(K(K + \sigma_n^2 I)^{-1});$$

si denotamos $K(K + \sigma_n^2 I)^{-1} = \mathbf{h}(X)$, se tiene que

$$\mathbb{E}_{\mathbf{f}}(\mathbf{f}|\mathbf{y}, X) = \mathbf{h}(X)^T \mathbf{y}.$$

A la función $\mathbf{h}(X)$ se le suele llamar función de pesos y en el caso de los procesos Gaussianos, esta función no depende de la variable observada $\mathbf{y}$; por ello, un efecto de los procesos Gaussianos es el de suavizadores lineales.

3.2.2 Procesos Gaussianos con media distinta de cero

Como hemos visto hasta ahora, en los procesos Gaussianos hemos supuesto que la media era cero, pero bien por conveniencia o por interpretabilidad del modelo, en algunos problemas resulta necesario usar una función de media no nula, es decir,

$$f(x) \sim GP(m(x), k(x, x')) \text{ con } m(x) \neq 0 \text{ para algún } x.$$

Para elegir la función de media, se puede usar una función de media determinista $m(x)$, de manera que la distribución conjunta de las observaciones en entrenamiento $(X, \mathbf{y})$ con las objetivo de predicción $(\hat{X}, \hat{\mathbf{f}})$ es:

$$\begin{bmatrix} \mathbf{y} \\ \hat{\mathbf{f}} \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} m(X) \\ m(\hat{X}) \end{bmatrix}, \begin{bmatrix} K(X, X) + \sigma_n^2 I_N & K(X, \hat{X}) \\ K(\hat{X}, X) & K(\hat{X}, \hat{X}) \end{bmatrix} \right); \tag{3.2.4}$$

haciendo los mismos desarrollos que en (3.2.1) tenemos que la predicción $\hat{\mathbf{f}}|\hat{X}, X, \mathbf{y}$ es una normal multidimensional, con la misma covarianza que un proceso Gaussiano de media cero, pero con media:

$$\mathbb{E}_{\hat{\mathbf{f}}}(\hat{\mathbf{f}}|\hat{X}, X, \mathbf{y}) = m(\hat{X}) + K(\hat{X}, X)(K(X, X) + \sigma_n^2)^{-1}(\mathbf{y} - m(X)).$$

En la práctica la función $m(x)$ no se conoce, por lo que se estima a partir de los datos, suponiendo que

$$\begin{aligned} y &= f(x) + \varepsilon, \\ f(x) &= g(x) + \mathbf{h}(x)^T\boldsymbol{\beta}, \\ g(x) &\sim GP(0, k(x, x')), \end{aligned}$$

donde la función $\mathbf{h}(\cdot)$ es fija. A la hora de entrenar el modelo, el vector de parámetros $\boldsymbol{\beta}$ se puede optimizar conjuntamente con los hiperparámetros de la función de covarianzas. Otra opción es suponer que $\boldsymbol{\beta} \sim \mathcal{N}(\mathbf{b}, B)$ obteniéndose un nuevo proceso Gaussiano, véase la sección 2.7 de [Rasmussen and Williams, 2004],

$$f(x) \sim GP(\mathbf{h}(x)^T\mathbf{b}, k(x, x') + \mathbf{h}(x)^T B\mathbf{h}(x')).$$

Sea H la matriz cuya fila i-ésima es $\mathbf{h}(x_i)$ y $\hat{H}$ la matriz cuya fila i-ésima es $\mathbf{h}(\hat{x}_i)$; haciendo los mismos desarrollos que en (3.2.4), con función de media $\mathbf{h}(x)^T\mathbf{b}$ y función de covarianza $k(x, x') + \mathbf{h}(x)^T B\mathbf{h}(x'))$, se tiene que

$$\begin{bmatrix} \mathbf{y} \\ \hat{\mathbf{f}} \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} H\mathbf{b} \\ \hat{H}\mathbf{b} \end{bmatrix}, \begin{bmatrix} K + \sigma_n^2 I + HBH^T & K(X, \hat{X}) + HB\hat{H}^T \\ K(\hat{X}, X) + \hat{H}BH^T & \hat{K} + \hat{H}B\hat{H}^T \end{bmatrix} \right);$$

después de agrupar términos, la distribución condicional se puede escribir como

$$\begin{aligned} \hat{\mathbf{f}}|\hat{X}, X, \mathbf{y} \sim \mathcal{N}(&\mathbb{E}_{\hat{\mathbf{g}}}(\hat{\mathbf{g}}|\hat{X}, X, \mathbf{y}) + R^T\overline{\beta}, \\ &\mathrm{Cov}(\hat{\mathbf{g}}|\hat{X}, X, \mathbf{y}) + R^T[B^{-1} + H^T(K + \sigma_n^2 I)^{-1}H]^{-1}R, \end{aligned}$$

donde $\hat{\mathbf{g}}|\hat{X}, X, \mathbf{y}$ es la variable aleatoria de predicción obtenida con un proceso Gaussiano de media cero y función de covarianzas $k(x, x')$, y

$$\begin{aligned} \overline{\beta} &= [B^{-1} + H^T(K + \sigma_n^2 I)^{-1}H]^{-1}(H^T(K + \sigma_n^2 I)^{-1}\mathbf{y} + B^{-1}\mathbf{b}), \\ R &= \hat{H}^T - H^T(K + \sigma_n^2 I)^{-1}K(X, \hat{X}). \end{aligned}$$

De manera que se puede ver que la covarianza y la media del nuevo proceso Gaussiano (f) se puede expresar como la media y covarianza de un proceso Gaussiano con media 0 (g) más unos términos que provienen de suponer que la media del proceso no es cero. En cualquier caso, en lo que sigue no vamos a considerar este tipo de proceso Gaussiano.

3.2.3 Predicción con los procesos Gaussianos

En las anteriores subsecciones, hemos visto que los procesos Gaussianos en predicción nos dan una distribución Gaussiana para la variable sin ruido $\hat{\mathbf{f}}|\hat{x}, X, \mathbf{y}$ y para la variable con ruido $\hat{\mathbf{y}}|\hat{x}, X, \mathbf{y}$, vistos en (3.2.3), pero generalmente en regresión se busca un valor exacto. Para elegir dicho valor es necesario definir una función de pérdida $\mathcal{L}(y_{verdadero}, y_{predicho})$.

Puesto que el valor real $\hat{y}$ del ejemplo $\hat{x}$ no se conoce, definimos una función de riesgo o pérdida esperada como

$$R_{\mathcal{L}}(y') = \mathbb{E}_Z(\mathcal{L}(Z, y')) = \int \mathcal{L}(z, y')p(z)dz,$$

donde Z es $\hat{\mathbf{f}}|\hat{x}, X, \mathbf{y}$ o $\hat{\mathbf{y}}|\hat{x}, X, \mathbf{y}$. De manera que se define el y óptimo como:

$$y_{\text{óptimo}} = \arg\min_{y'} R_{\mathcal{L}}(y').$$

Si elegimos como función de pérdida la norma $L^1(\mathcal{L} = |\hat{y} - y|)$, el $y_{\text{óptimo}}$ será la mediana de la distribución de predicción y si elegimos como función de pérdida la norma $L^2(\mathcal{L} = (\hat{y} - y)^2)$ el $y_{\text{óptimo}}$ será la media; para el caso de distribuciones Gaussianas, puesto que la media es igual a la mediana, el $y_{\text{óptimo}}$ minimiza el riesgo en L^1 y en L^2.

Por lo tanto, por (3.2.2) el valor óptimo de los riesgos en L^1 y L^2 para $\hat{\mathbf{y}}|\hat{x}, X, \mathbf{y}$ y $\hat{\mathbf{f}}|\hat{x}, X, \mathbf{y}$ es el mismo: $\mathbf{k}^T(K + \sigma_n^2 I)^{-1}\mathbf{y}$.

3.3 Hiperparametrización

Hasta ahora hemos visto cómo hacer regresión con una función de covarianzas fija; no obstante, la matriz de covarianza tiene unos hiperparámetros que dependen del núcleo escogido. Por ejemplo en el caso de un núcleo RBF, se tiene que la función de covarianzas de y es

$$k(x, x') = \sigma_f^2 \exp\left(-\frac{1}{2\ell^2}\|x - x'\|^2\right) + \sigma_n^2 \delta_{xx'},$$

donde σ_f^2, σ_n^2 y ℓ son hiperparámetros. Estos hiperparámetros los agruparemos en un conjunto $\boldsymbol{\theta}$ a lo largo de la explicación de cómo calcular los hiperparámetros óptimos. Estos hiperparámetros se pueden optimizar mediante métodos tradicionales como búsqueda en rejilla con validación cruzada, pero en el caso de los procesos Gaussianos se usan principalmente dos técnicas: selección Bayesiana de modelos y validación cruzada dejando uno fuera(leave-one-out, LOO); véase el Capítulo 3 de [Rasmussen and Williams, 2004].

A continuación vamos a explicar la selección Bayesiana de modelos, la verosimilitud de la selección Bayesiana de modelos como navaja de Ockham y la validación cruzada LOO para los procesos Gaussianos.

3.3.1 Selección Bayesiana de modelos

Es común y sensato especificar de manera jerárquica los modelos en los siguientes niveles:

1. Estructura de los modelos: $\mathcal{H}$.

2. Hiperparámetros: θ. Ej: el número de capas de una red neuronal o la longitud de escala en un RBF.

3. Parámetros: w. Ej: los pesos de una red neuronal.

En cada uno de estos niveles se puede aplicar la regla genérica de Bayes para realizar la inferencia bayesiana:

$$\text{posterior} = \frac{\text{verosimilitud } \times \text{prior}}{\text{verosimilitud marginal}}.$$

El objetivo en la selección Bayesiana de modelos es conocer la distribución del posterior; no obstante, muchas veces no se puede calcular de manera analítica la verosimilitud marginal (la constante de normalización) pues implica calcular una integral que puede resultar intratable. Por ello, o bien aproximamos analíticamente el posterior o buscamos los parámetros, hiperparámetros o modelos que maximizan la verosimilitud.

Aplicando el enfoque bayesiano a los parámetros ($\mathbf{w}$) se tiene que

$$p(\mathbf{w}|\mathbf{y}, X, \boldsymbol{\theta}, \mathcal{H}) = \frac{p(\mathbf{y}, \mathbf{w}|X, \boldsymbol{\theta}, \mathcal{H})}{p(\mathbf{y}|X, \boldsymbol{\theta}, \mathcal{H})} = \frac{p(\mathbf{y}|X, \mathbf{w}, \boldsymbol{\theta}, \mathcal{H})p(\mathbf{w}|\boldsymbol{\theta}, \mathcal{H})}{p(\mathbf{y}|X, \boldsymbol{\theta}, \mathcal{H})},$$
$$p(\mathbf{y}|X, \boldsymbol{\theta}, \mathcal{H}) = \int p(\mathbf{y}, \mathbf{w}|X, \boldsymbol{\theta}, \mathcal{H})d\mathbf{w} = \int p(\mathbf{y}|X, \mathbf{w}, \boldsymbol{\theta}, \mathcal{H})p(\mathbf{w}|\boldsymbol{\theta}, \mathcal{H})d\mathbf{w}.$$

En la sección 2.4, con otro enfoque, vimos los procesos Gaussianos como una distribución en los pesos, y cómo con el posterior de los pesos se podía dar una predicción. En el caso de los hiperparámetros se tiene que

$$p(\boldsymbol{\theta}|\mathbf{y}, X, \mathcal{H}) = \frac{p(\mathbf{y}, \boldsymbol{\theta}|X, \mathcal{H})}{p(\mathbf{y}|X, \mathcal{H})} = \frac{p(\mathbf{y}|X, \boldsymbol{\theta}, \mathcal{H})p(\boldsymbol{\theta}|\mathcal{H})}{p(\mathbf{y}|X, \mathcal{H})},$$
$$p(\mathbf{y}|X, \mathcal{H}) = \int p(\mathbf{y}, \boldsymbol{\theta}|X, \mathcal{H})d\boldsymbol{\theta} = \int p(\mathbf{y}|X, \boldsymbol{\theta}, \mathcal{H})p(\boldsymbol{\theta}|\mathcal{H})d\boldsymbol{\theta}.$$

En el caso de procesos Gaussianos de media cero se tiene que la verosimilitud de los hiperparámetros es $p(\mathbf{y}|X, \boldsymbol{\theta})$ donde

$$\mathbf{y}|X, \boldsymbol{\theta} \sim \mathcal{N}(0, K(X, X) + \sigma_n^2 I) = \mathcal{N}(0, K_{\boldsymbol{\theta}});$$

por lo tanto,

$$\log p(\mathbf{y}|X, \boldsymbol{\theta}) = -\frac{1}{2}\mathbf{y}^T K_{\boldsymbol{\theta}}^{-1}\mathbf{y} - \frac{1}{2}\log\det(K_{\boldsymbol{\theta}}) - \frac{n}{2}\log 2\pi. \tag{3.3.1}$$

Como no vamos a poder calcular el posterior $p(\boldsymbol{\theta}|\mathbf{y}, X, \mathcal{H})$ analíticamente, optaremos por maximizar la verosimilitud con respecto a los hiperparámetros.

Para maximizar la verosimilitud (3.3.1), es necesario calcular su gradiente con respecto a los hiperparámetros $\boldsymbol{\theta}$. Para ello primero vamos a usar los siguientes resultados, que se pueden encontrar en el Apéndice 1 de [Fukunaga, 1990]:

$$\frac{\partial K^{-1}}{\partial \theta_j} = -K^{-1}\frac{\partial K}{\partial \theta_j}K^{-1}, \tag{3.3.2}$$
$$\frac{\partial \log\det(K)}{\partial \theta_j} = \text{traza}(K^{-1}\frac{\partial K}{\partial \theta_j}). \tag{3.3.3}$$

Usando la regla de la cadena y las expresiones anteriores, se tiene que

$$\begin{aligned}\frac{\partial}{\partial \theta_j}\log p(y|X, \boldsymbol{\theta}) &= \frac{\partial}{\partial \theta_j}\left(-\frac{1}{2}\mathbf{y}^T K_{\boldsymbol{\theta}}^{-1}\mathbf{y} - \frac{1}{2}\log|K_{\boldsymbol{\theta}}| - \frac{n}{2}\log 2\pi\right)\\ &= \frac{1}{2}\mathbf{y}^T K_{\boldsymbol{\theta}}^{-1}\frac{\partial K_{\boldsymbol{\theta}}}{\partial \theta_j}K_{\boldsymbol{\theta}}^{-1}\mathbf{y} - \frac{1}{2}\,\text{traza}\left(K_{\boldsymbol{\theta}}^{-1}\frac{\partial K_{\boldsymbol{\theta}}}{\partial \theta_j}\right).\end{aligned}$$

Para reducir los cálculos, si denotamos $\boldsymbol{\alpha} = K_{\boldsymbol{\theta}}^{-1}\mathbf{y}$, se tiene que

$$\mathbf{y}^T K_{\boldsymbol{\theta}}^{-1} \frac{\partial K_{\boldsymbol{\theta}}}{\partial \theta_j} K_{\boldsymbol{\theta}}^{-1}\mathbf{y} = \boldsymbol{\alpha}^T \frac{\partial K_{\boldsymbol{\theta}}}{\partial \theta_j}\boldsymbol{\alpha}.$$

En la expresión anterior el término obtenido es un número real; por lo tanto, su traza es él mismo, de manera que usando la propiedad cíclica de la traza se tiene que

$$\boldsymbol{\alpha}^T \frac{\partial K_{\boldsymbol{\theta}}}{\partial \theta_j}\boldsymbol{\alpha} = \text{traza}\left(\boldsymbol{\alpha}^T \frac{\partial K_{\boldsymbol{\theta}}}{\partial \theta_j}\boldsymbol{\alpha}\right) = \text{traza}\left(\boldsymbol{\alpha}\boldsymbol{\alpha}^T \frac{\partial K_{\boldsymbol{\theta}}}{\partial \theta_j}\right).$$

Finalmente, agrupando términos se tiene que

$$\frac{\partial}{\partial \theta_j} \log p(y|X, \boldsymbol{\theta_j}) = \frac{1}{2}\,\text{traza}\left((\boldsymbol{\alpha}\boldsymbol{\alpha}^T - K_{\boldsymbol{\theta}}^{-1})\frac{\partial K_{\boldsymbol{\theta}}}{\partial \theta_j}\right).$$

Con esta notación podemos ver que para calcular el gradiente es necesario, primero calcular $K_{\boldsymbol{\theta}}^{-1}$, cuyo coste es cúbico en el número de patrones. Como la matriz $K_{\boldsymbol{\theta}}$ cambia en cada paso de la optimización, la inversa también varía, y por lo tanto, hay que calcularla en cada iteración de la optimización. Por otro lado, una vez conocida $K_{\boldsymbol{\theta}}^{-1}$, el cálculo de derivadas tiene un coste lineal con respecto al número de hiperparámetros. Teniendo en cuenta que en cada derivada parcial es necesario calcular la diagonal del producto de dos matrices, cuyo coste es cuadrático, el coste en calcular el gradiente una vez calculada la inversa de $K_{\boldsymbol{\theta}}$ es $O(N^2|\boldsymbol{\theta}|)$, donde $|\boldsymbol{\theta}|$ es el número de hiperparámetros que tiene el modelo.

No obstante, este método presenta el problema de que la verosimilitud puede tener más de un máximo local. Por ello es común realizar el ascenso por gradiente con valores iniciales distintos, eligiendo finalmente los hiperparámetros que obtegan la mayor verosimilitud.

3.3.2 La verosimilitud como navaja de Ockham

En la verosimilitud (3.3.1) se pueden identificar tres términos (véase sección 5.4 de [Rasmussen and Williams, 2004]),

- El primer término ($-\frac{1}{2}\mathbf{y}^T K_{\boldsymbol{\theta}}^{-1}\mathbf{y}$) controla el ajuste a los datos, pues estamos diciendo que el proceso tiene media cero. Como $K_{\boldsymbol{\theta}}$ es definida positiva (pues $K_{\boldsymbol{\theta}}$ es la matriz K, que es semi-definida positiva a la cual se le ha sumado un número positivo a los elementos de la diagonal), $K_{\boldsymbol{\theta}}^{-1}$ también es definida positiva y por lo tanto, este primer término es negativo o cero, en el caso que $\mathbf{y} = 0$.

- El segundo término ($-\frac{1}{2}\log\det(K_{\boldsymbol{\theta}})$) controla la penalización a la complejidad del modelo.

- El tercer término ($-\frac{n}{2}\log 2\pi$) es una constante de normalización.

Por lo tanto, un modelo que se ajuste muy bien a los datos tendrá un primer término muy próximo a cero, pero el segundo término será más negativo. Por ejemplo, en el caso del RBF Gaussiano (2.3.11) si la escala (ℓ) se hace muy grande, el modelo se vuelve más sencillo, haciéndose el segundo término menos negativo, pero el primer término se hace más negativo porque el modelo aproxima peor. En cambio, si la escala se hace muy pequeña, el modelo se hará más complejo; por lo tanto, el segundo término será menor.

Para ver esto, primero usamos la notación $A^\ell = \frac{1}{\sigma_f^2}K^\ell$ para cancelar σ_f^2 en K, donde el superíndice ℓ indica que la matriz depende de ℓ; esta matriz A^ℓ tiene unos en la diagonal; de manera que si hacemos la descomposición SVD de la matriz A^ℓ, podemos reescribir la matriz $K_{\boldsymbol{\theta}}^\ell$ como:

$$K_{\boldsymbol{\theta}}^\ell = K^\ell + \sigma_n^2 I = \sigma_f^2 A^\ell + \sigma_n^2 I = \sigma_f^2 U D^\ell U^T + \sigma_n^2 U U^T = U(\sigma_f^2 D^\ell + \sigma_n^2 I)U^T,$$

donde U es una matriz ortogonal y D^ℓ es una matriz diagonal con términos no negativos; por lo tanto, denotando $D_{ii}^\ell = d_i^\ell$ se tiene que

$$\det(K_{\boldsymbol{\theta}}^\ell) = \det(U)\det(\sigma_f^2 D^\ell + \sigma_n^2 I)\det(U^T) = \prod_{i=1}^{N}(\sigma_f^2 d_i^\ell + \sigma_n^2).$$

En el caso en el que la escala se hace muy grande, la matriz A^ℓ tiende a una matriz de unos de tamaño $N \times N$, cuyos autovalores son N con multiplicidad uno y cero con multiplicidad $N-1$; por lo tanto, $d_1^\infty = N, d_j^\infty = 0\ \forall j > 1$, y se tiene que

$$\det(K_{\boldsymbol{\theta}}^\ell) = \prod_{i=1}^{N}(\sigma_f^2 d_i^\ell + \sigma_n^2) \xrightarrow{\ell\to\infty} (\sigma_f^2 N + \sigma_n^2)(\sigma_n^2)^{N-1}.$$

Dado que $\text{traza}(A^\ell) = \sum_{i=1}^{N} d_i^\ell = N$, pues A^ℓ tiene en la diagonal unos, se tiene que

$$(\sigma_f^2 N + \sigma_n^2)(\sigma_n^2)^{N-1} = (\sigma_f^2 \sum_{i=1}^{N} d_i^\ell + \sigma_n^2)(\sigma_n^2)^{N-1},$$

y se sigue que

$$-\frac{1}{2}\log\det(K_{\boldsymbol{\theta}}) \xrightarrow{\ell\to\infty} -\frac{N-1}{2}\log\sigma_n^2 - \frac{1}{2}\log(\sigma_f^2 \sum_{i=1}^{N} d_i^\ell + \sigma_n^2).$$

En el caso en el que la escala se hace muy pequeña, la matriz A^ℓ tenderá a la identidad, y se tiene que

$$\det(K_{\boldsymbol{\theta}}^\ell) = \prod_{i=1}^{N}(\sigma_f^2 d_i^\ell + \sigma_n^2) \xrightarrow{\ell\to 0} (\sigma_f^2 + \sigma_n^2)^N,$$

y en consecuencia

$$-\frac{1}{2}\log\det(K_{\boldsymbol{\theta}}^\ell) \xrightarrow{\ell\to 0} -\frac{N}{2}\log(\sigma_f^2 + \sigma_n^2).$$

Por lo tanto, se cumple que

$$\lim_{\ell\to 0} -\log\det(K_{\boldsymbol{\theta}}^\ell) \geq -\log\det(K_{\boldsymbol{\theta}}^\ell) \geq \lim_{\ell\to\infty} -\log\det(K_{\boldsymbol{\theta}}^\ell),$$

pues

$$(\sigma_f^2 N + \sigma_n^2)(\sigma_n^2)^{N-1} \leq \prod_{i=1}^{N}(\sigma_f^2 d_i^\ell + \sigma_n^2) \leq (\sigma_f^2 + \sigma_n^2)^N, \tag{3.3.4}$$

dado que para la primera desigualdad basta notar que

$$\begin{aligned}\prod_{i=1}^{N}(\sigma_f^2 d_i^\ell + \sigma_n^2) &= (\sigma_n^2)^N + (\sigma_n^2)^{N-1}\sigma_f^2 \sum_{i=1}^{N} d_i^\ell + \text{términos no negativos} \\ &= (\sigma_f^2 N + \sigma_n^2)(\sigma_n^2)^{N-1} + \text{términos no negativos},\end{aligned}$$

y para la segunda desigualdad véase el Anexo A.

3.3.3 Validación cruzada: Leave-one-out

La validación cruzada consiste en dividir el conjunto de entrenamiento en T particiones y realizar T modelos, donde en cada modelo se usa una partición distinta para validar y el resto para entrenar.

El caso extremo, que se trata aquí es en el que $T = N$, al cual se le llama LOO-CV, que suele ser computacionalmente muy costoso e intratable, pues en muchos algoritmos implica realizar N modelos por cada hiperparámetro. En el caso de procesos Gaussianos, usando propiedades de las matrices podemos expresar el logaritmo de la función predictiva de probabilidad o verosimilitud L_{LOO} únicamente con la matriz de covarianzas y las observaciones con todos los ejemplos de entrenamiento.

Para ello, primero definimos

$$(X_{-i}, \mathbf{y}_{-i}) = (X, \mathbf{y}) \setminus (x_i, y_i);$$

para aligerar la notación, cuando condicionemos con respecto a X_{-i} y x_i pondremos directamente X, es decir,

$$y_i|x_i,\ X_{-i}, \mathbf{y}_{-i} = y_i|\ X, \mathbf{y}_{-i};$$

con esta notación, definimos la verosimilitud L_{LOO}:

$$L_{LOO}(X, \mathbf{y}, \boldsymbol{\theta}) = \sum_{i=1}^{n} \log p(y_i|X, \mathbf{y}_{-i}, \boldsymbol{\theta}).$$

Los hiperparámetros óptimos serán aquellos que maximizan la función L_{LOO}. Teniendo en cuenta los resultados vistos en las expresiones (3.2.1) y (3.2.3), se tiene que

$$\begin{aligned} L_{LOO}(X, \mathbf{y}, \boldsymbol{\theta}) &= \sum_{i=1}^{n} -\frac{1}{2}\left(\log \sigma_i^2 + \frac{(y_i - \mu_i)^2}{\sigma_i^2} + \log(2\pi)\right) \\ &= -\sum_{i=1}^{n} \frac{1}{2}\log \sigma_i^2 - \sum_{i=1}^{n} \frac{(y_i - \mu_i)^2}{2\sigma_i^2} - \frac{n}{2}\log(2\pi), \end{aligned}$$

donde $y_i|X, \mathbf{y}_{-i}, \boldsymbol{\theta}$, es la predicción de un proceso Gaussiano con el conjunto de entrenamiento $(X_{-i}, \mathbf{y}_{-i})$ y de test x_i, y_i, es decir,

$$\begin{aligned} y_i|x_i, X_{-i}, \mathbf{y}_{-i} = y_i|X, \mathbf{y}_{-i} &\sim \mathcal{N}(\mu_i, \sigma_i^2), \\ \mu_i &= k(x_i, X_{-i})K_{-i}^{-1}\mathbf{y}_{-i}, \\ \sigma_i^2 &= (k(x_i, x_i) + \sigma_n^2) - k(x_i, X_{-i})K_{-i}^{-1}k(X_{-i}, x_i), \end{aligned}$$

donde $K_{-i} = k(X_{-i}, X_{-i})$, $k(x_i, X_{-i})$ es un vector fila de dimensión $N-1$ y $k(X_{-i}, x_i)$ es un vector columna de dimensión $N-1$, cumpliéndose que $k(x_i, X_{-i})^T = k(X_{-i}, x_i)$.

En la menos verosmilitud $(-L_{LOO})$ podemos identificar tres términos:

- El primer término $(\sum_{i=1}^{n} \frac{1}{2}\log \sigma_i^2)$ controla la varianza del modelo.
- El segundo término $(\sum_{i=1}^{n} \frac{(y_i-\mu_i)^2}{2\sigma_i^2})$ es un error cuadrático medio ponderado.
- El tercer término $(\frac{n}{2}\log(2\pi))$ es una constante de normalización.

Por lo tanto, al maximizar la verosimilitud, estamos minimizando el error y la varianza del modelo. No obstante, se presenta el dilema sesgo-varianza: si las σ_i^2 son muy pequeñas, el término de la varianza se hace negativo, pero el término correspondiente al error cuadrático medio ponderado se hace más grande. Volviendo a la función de verosimilitud, se puede demostrar que μ_i y σ_i^2, véase sección 5.4 de [Rasmussen and Williams, 2004], se pueden expresar en función de la matriz de covarianzas con todos los ejemplos y el vector $\mathbf{y}$ con todas las observaciones:

$$\mu_i = y_i - \frac{[K_{\boldsymbol{\theta}}^{-1}\mathbf{y}]_i}{[K_{\boldsymbol{\theta}}^{-1}]_{ii}}, \tag{3.3.5}$$

$$\sigma_i^2 = \frac{1}{[K_{\boldsymbol{\theta}}^{-1}]_{ii}}. \tag{3.3.6}$$

Con estos dos resultados, usando (3.3.2) y (3.3.3) y denotando $\boldsymbol{\alpha} = K_{\boldsymbol{\theta}}^{-1}\mathbf{y}$, $Z_j = K_{\boldsymbol{\theta}}^{-1}\frac{\partial K_{\boldsymbol{\theta}}}{\partial \theta_j}$, las derivadas parciales de la media y la varianza con respecto a θ_j son:

$$\begin{aligned}
\frac{\partial \mu_i}{\partial \theta_j} &= \frac{[K_{\boldsymbol{\theta}}^{-1}\frac{\partial K_{\boldsymbol{\theta}}}{\partial \theta_j}K_{\boldsymbol{\theta}}^{-1}\mathbf{y}]_i}{[K_{\boldsymbol{\theta}}^{-1}]_{ii}} - [K_{\boldsymbol{\theta}}^{-1}\mathbf{y}]_i\frac{[K_{\boldsymbol{\theta}}^{-1}\frac{\partial K_{\boldsymbol{\theta}}}{\partial \theta_j}K_{\boldsymbol{\theta}}^{-1}]_i}{[K_{\boldsymbol{\theta}}^{-1}]_{ii}^2} \\
&= \frac{[Z_j\boldsymbol{\alpha}]_i}{[K_{\boldsymbol{\theta}}^{-1}]_{ii}} - \alpha_i\frac{[Z_jK_{\boldsymbol{\theta}}^{-1}]_i}{[K_{\boldsymbol{\theta}}^{-1}]_{ii}^2}, \\
\frac{\partial \sigma_i^2}{\partial \theta_j} &= \frac{[Z_jK_{\boldsymbol{\theta}}^{-1}]_i}{[K_{\boldsymbol{\theta}}^{-1}]_{ii}^2};
\end{aligned}$$

de manera que aplicando la regla de la cadena se tiene que

$$\begin{aligned}
\frac{\partial L_{LOO}}{\partial \theta_j} &= \sum_{i=1}^{N}\frac{\partial \log p(y_i|X,\mathbf{y}_{-i},\boldsymbol{\theta})}{\partial \mu_i}\frac{\partial \mu_i}{\partial \theta_j} + \frac{\partial \log p(y_i|X,\mathbf{y}_{-i},\boldsymbol{\theta})}{\partial \sigma_i^2}\frac{\partial \sigma_i^2}{\partial \theta_j} \\
&= \sum_{i=1}^{N}\frac{y_i-\mu_i}{\sigma_i^2}\frac{\partial \mu_i}{\partial \theta_j} + \left(-\frac{1}{2\sigma_i^2} + \frac{(y_i-\mu_i)^2}{2(\sigma_i^2)^2}\right)\frac{\partial \sigma_i^2}{\partial \theta_j} \\
&= \sum_{i=1}^{N}\frac{1}{[K_{\boldsymbol{\theta}}^{-1}]_{ii}}\left(\alpha_i[Z_j\boldsymbol{\alpha}]_i - \frac{1}{2}(1+\frac{\alpha_i^2}{[K_{\boldsymbol{\theta}}^{-1}]_{ii}})[Z_jK_{\boldsymbol{\theta}}^{-1}]_{ii}\right),
\end{aligned}$$

donde en la última línea se ha sustituido μ_i y σ_i^2 por su valor en las expresiones (3.3.5), (3.3.6) y las derivadas parciales calculadas anteriormente.

En este caso, la complejidad es cúbica, tanto de calcular la inversa de la matriz como calcular las derivadas, por lo que el coste es mayor que en el método bayesiano.

3.4 Procesos Gaussianos censurados

La censura en estadística es el fenómeno que ocurre cuando se conoce parcialmente el valor de una variable. Es decir, no conocemos el valor exacto de una variable, pero tenemos cierta información sobre la variable como por ejemplo, saber que es positiva o que está acotada en un intervalo

Un ejemplo de censura se puede observar en las calificaciones finales de un estudiante, pues en el sistema español de educación las calificaciones deben ser un número decimal entre

cero y diez. Otro ejemplo es la producción de energía en un parque eólico: habrá días que se produzca más y otros días que se produzca menos energía, pero siempre se va a cumplir que la energía producida es mayor o igual que cero y menor o igual que la capacidad total del parque.

Los modelos citados en la sección 2, así como los procesos Gaussianos, son modelos de regresión tradicionales, cuyo objetivo es obtener una predicción de una variable en la recta real, de manera que no incorporan el conocimiento de que la variable está censurada en un intervalo, (en el caso de la energía eólica, las observaciones una vez normalizadas están en el intervalo [0,1]). En esta sección, se explica cómo incorporar esta censura a un proceso Gaussiano, siguiendo a [Groot and Lucas, 2012].

Primero vamos a considerar el problema de regresión con censura y sin ruido; para ello supongamos que la variable observada (y) está en el intervalo $[l, u]$ y supongamos entonces que hay una variable latente que es un proceso Gaussiano, es decir, $f(x) \sim GP(0, k(x, x'))$, que posteriormente se censura, de manera que la variable a predecir es:

$$z = \text{máx}(\text{mín}(f(x), u), l),$$

y la probabilidad asociada que, abusando de la notación, llamaremos verosimilitud es:

$$p_{id}(z|f(x)) = \begin{cases} 1, & \text{si } z = l \wedge f(x) \leq l, \\ & \text{o si } z = f(x) \wedge l < f(x) < u, \\ & \text{o si } z = u \wedge f(x) \geq u \\ 0, & \text{otro caso,} \end{cases}$$

donde p_{id} quiere decir la verosimilitud ideal, es decir, sin ruido. La verosimilitud anterior la podemos reescribir como

$$p_{id}(z|f(x)) = \mathbb{1}_{(-\infty,l]}(f(x))\delta_l(z) + \mathbb{1}_{[u,\infty)}(f(x))\delta_u(z) + \mathbb{1}_{(u,l)}(f(x))\delta_{f(x)}(z).$$

Sin embargo, vamos a considerar que la variable latente (f) está contaminada por un ruido blanco, y posteriormente censurada, es decir,

$$y = \text{máx}(\text{mín}(f(x) + \varepsilon, u), l),$$

donde típicamente se suele considerar que $\varepsilon \sim \mathcal{N}(0, \sigma^2)$. Sea $f_r = f(x) + \varepsilon$, es decir, la variable latente con ruido, sea $\mathcal{N}(z|\mu, \sigma^2)$ la función de densidad de una normal unidimensional con media μ y varianza σ^2 evaluada en z y sea $\Phi(\cdot)$ la función de distribución de una normal unidimensional con media cero y varianza uno; se tiene entonces que

$$\begin{aligned} p(y = l|\ f) &= \int_{\mathbb{R}} p(y = l, f_r|\ f(x))df_r \\ &= \int_{\mathbb{R}} p_{id}(y = l|f_r)p(f_r|f(x))df_r \\ &= \int_{\mathbb{R}} \mathbb{1}_{(-\infty,l]}(f_r)\delta_l(l)\mathcal{N}(f_r - f(x)|0, \sigma^2)df_r \\ &= \delta_l(l) \int_{-\infty}^{l} \mathcal{N}(f_r - f(x)|0, \sigma^2)df_r \\ &= \delta_l(l) \int_{-\infty}^{l-f} \mathcal{N}(f_r|0, \sigma^2)df_r \\ &= \delta_l(l)\Phi\left(\frac{l - f(x)}{\sigma}\right) = \left(1 - \Phi\left(\frac{f(x) - l}{\sigma}\right)\right)\delta_l(y); \end{aligned} \tag{3.4.1}$$

donde en la segunda línea se ha usado el hecho de que la censura es posterior al ruido y $\delta(\cdot)$ es la delta de Dirac. Realizando los mismos cálculos para el caso $y = u$ se tiene que

$$p(y = u|\ f(x)) = \Phi\left(\frac{f(x) - u}{\sigma}\right)\delta_u(y). \tag{3.4.2}$$

Finalmente, para el caso en el que $y = f_r'$ con $f_r' \in (l, u)$ se tiene que

$$\begin{aligned} p(y = f_r'|f(x)) &= \int \mathbb{1}_{(u,l)}(f_r')\delta(f_r' - f_r)p(f_r|\ f(x))df_r \\ &= \mathbb{1}_{(u,l)}(f_r')\mathcal{N}(f_r'|f(x), \sigma^2) = \mathbb{1}_{(u,l)}(y)\mathcal{N}(y - f(x)|0, \sigma^2). \end{aligned} \tag{3.4.3}$$

Por (3.4.1), (3.4.2) y (3.4.3) se tiene que la medida de Lebesgue-Stieltjes dm asociada a la función de distribución de $y|f(x)$ se puede representar como:

$$\begin{aligned} dm = &\left(1 - \Phi\left(\frac{f(x) - l}{\sigma}\right)\right)d\delta_l(y) + \Phi\left(\frac{f(x) - u}{\sigma}\right)\delta_u(y) \\ &+ \mathbb{1}_{(u,l)}(y)\mathcal{N}(y - f(x)|0, \sigma^2)dy, \end{aligned} \tag{3.4.4}$$

donde dy es la medida de Lebesgue; esto quiere decir, que la variable observada (y) condicionada por la variable latente (f) sigue una distribución continua en el intervalo (l, u) y discreta en l y u.

La y no es un proceso Gaussiano ya que es una transformación no lineal de la variable latente con ruido; por lo tanto, no podemos usar los resultados vistos anteriormente. La f es una variable latente, por lo tanto; el conjunto de entrenamiento va a estar formado por la variable observada y las variables predictivas, es decir, el conjunto $(X, \mathbf{y}) = \{(x_i, y_i)\}_{i=1}^N$ y no vamos a conocer las $f(x_i)$. Por ello, primero vamos a tener que calcular la distribución del posterior $\mathbf{f}|X, \mathbf{y}$, donde $\mathbf{f} = (f(x_1), ..., f(x_N))^T$, pues la predicción de la variable latente $\hat{f}$ para un nuevo ejemplo $\hat{x}$ es:

$$p(\hat{f}|\hat{x}, X, \mathbf{y}) = \int p(\hat{f}, \mathbf{f}|\hat{x}, X, \mathbf{y})d\mathbf{f} = \int p(\hat{f}|\hat{x}, \mathbf{f})p(\mathbf{f}|X, \mathbf{y})d\mathbf{f};$$

por los resultados vistos en la sección 3.2, el primer término $p(\hat{f}|\hat{x}, \mathbf{f})$ lo conocemos. Para calcular el segundo término $p(\mathbf{f}|X, \mathbf{y})$, usamos la regla de Bayes:

$$p(\mathbf{f}|X, \mathbf{y}) = \frac{p(\mathbf{f}, \mathbf{y}|X)}{p(\mathbf{y}|X)} = \frac{1}{Z}p(\mathbf{f}|X)p(\mathbf{y}|\mathbf{f}), \tag{3.4.5}$$

donde el primer término Z es la constante de normalización o evidencia, es decir,

$$Z = p(\mathbf{y}|X) = \int p(\mathbf{f}, \mathbf{y}|X)d\mathbf{f} = \int p(\mathbf{f}|X)p(\mathbf{y}|\mathbf{f})d\mathbf{f};$$

el segundo término $p(\mathbf{f}|X)$ se corresponde con el prior, el cual en el caso del proceso Gaussiano, por definición, tiene la siguiente distribución:

$$\mathbf{f}|X \sim \mathcal{N}(\mathbf{f}|0, K),$$

donde $K_{ij} = k(x_i, x_j)$; si asumimos que las observaciones son muestras independientes e idénticamente distribuidas se tiene que la verosimilitud factoriza, y se puede escribir de la siguiente forma (veáse [Groot and Lucas, 2012] o sección 22.3 de [Greene, 2012]),

$$\begin{aligned} p(\mathbf{y}|\mathbf{f}) &= \prod_{i=1}^N p(y_i|\ f(x_i)) \\ &= \prod_{y_i=l}\left[1 - \Phi\left(\frac{f(x_i) - l}{\sigma}\right)\right]\prod_{l<y_i<u}\mathcal{N}(y_i - f(x_i)|0, \sigma^2)\prod_{y_i=u}\left[\Phi\left(\frac{f(x_i) - u}{\sigma}\right)\right]. \end{aligned}$$

Notar que si hacemos u tender a ∞ y l tender a $-\infty$, los términos no Gaussianos de la verosimilitud desaparecen y obtenemos el modelo de procesos Gaussianos tradicional. Debido a estos términos no Gaussianos, el posterior $\mathbf{f}|X, \mathbf{y}$ de la expresión (3.4.5) es intratable analíticamente; por lo tanto, es necesario aproximarlo. Para aproximar este posterior hay distintos métodos: bien deterministas como la aproximación de Laplace, o bien de propagación de la esperanza (expectation propagation) o bien de muestreo como Markov Chain Monte Carlo entre otros; véase capítulos 10 y 11 de [Bishop, 2006].

En este trabajo se detalla cómo usar la propagación de la esperanza, pues este método realiza una proyección de cada factor en la familia exponencial. Dado que la Gaussiana pertenece a esta familia, su proyección, es ella misma, y para la función de distribución de una Gaussiana se ha visto que propagación de la esperanza obtiene mejores aproximaciones para el modelo de procesos Gaussianos en clasificación [Nickisch and Rasmussen, 2008].

3.4.1 Propagación de la esperanza

A continuación se detalla de manera resumida, el método de propagación de la esperanza [Minka, 2001]; para ello se ha seguido la sección 10.7 de [Bishop, 2006]. Supongamos que queremos efectuar la siguiente aproximación $\mathcal{Q}(\mathbf{z})$ a una distribución $\mathcal{P}(\mathbf{z})$,

$$\mathcal{P}(\mathbf{z}) = \frac{1}{Z}\prod_{i=1}^{K} f_i(\mathbf{z}) \approx \mathcal{Q}(\mathbf{z}) \coloneqq \frac{1}{Z_{EP}}\prod_{i=1}^{K} \widetilde{f}_i(\mathbf{z}),$$

donde los factores $f_i(\mathbf{z})$ son funciones generales no negativas (aquí no son las f_i del proceso Gaussiano). En inferencia variacional, [Bishop, 2006] capítulo 10, buscamos que $\mathcal{Q}$ minimice la divergencia de Kullback-Leibler ($KL(\mathcal{Q}||\mathcal{P})$); sin embargo, en propagación de la esperanza buscamos que $\mathcal{Q}$ minimice $KL(\mathcal{P}||\mathcal{Q})$, pues se busca que la aproximación Q tome valores altos donde $\mathcal{P}$ sea altamente densa, ya que la expresión de la divergencia de Kullback-Leibler es

$$KL(\mathcal{P}||\mathcal{Q}) = -\int \mathcal{P}(\mathbf{z}) \ \log\frac{\mathcal{Q}(\mathbf{z})}{\mathcal{P}(\mathbf{z})} d\mathbf{z}.$$

Además, buscamos que $\mathcal{Q}$ pertenezca a la familia exponencial; esta familia de distribuciones de probabilidad cumple que es cerrada frente a la operación producto. Dentro de esta familia, vamos a imponer que la aproximación sea una Gaussiana; por lo tanto, idealmente lo que tenemos que buscar es

$$\mathcal{Q}(\mathbf{z}) = \underset{R\in\text{Gaussianas}}{\arg\min} \quad KL(\mathcal{P}(\mathbf{z})||R(\mathbf{z}));$$

en este caso, se puede demostrar que para minimizar la divergencia de Kullback-Leibler [Minka, 2001], basta con que se cumpla

$$\mathbb{E}_{\mathcal{P}}(\mathbf{z}) = \mathbb{E}_{\mathcal{Q}}(\mathbf{z}) \text{ y } \mathbb{E}_{\mathcal{P}}(\mathbf{z}\mathbf{z}^T) = \mathbb{E}_{\mathcal{Q}}(\mathbf{z}\mathbf{z}^T).$$

No obstante, como no sabemos la distribución de $\mathcal{P}(\mathbf{z})$, no vamos a poder calcular sus momentos. Como $\mathcal{P}(\mathbf{z})$ es un producto de K factores, una opción que podemos hacer es aproximar cada factor por separado minimizando $KL(f_i(\mathbf{z})||\widetilde{f}_i(\mathbf{z}))$; no obstante, esto puede dar lugar a que el producto de los factores no sea una buena aproximación de la distribución $\mathcal{P}(\mathbf{z})$ que queremos calcular.

Para evitar este problema, el algoritmo de propagación de la esperanza lo que hace es calcular de manera iterativa la aproximación $\widetilde{f}_i$ del factor f_i, teniendo en cuenta la aproximación

ya realizada sobre los otros factores:

$$\widetilde{f}_i(\mathbf{z}) = \arg\min_g KL\left(\frac{1}{Z_i} f_i(\mathbf{z})\mathcal{Q}^{\setminus i}(\mathbf{z})\Big|\Big|\frac{1}{Z_i}\mathcal{Q}^{\setminus i} g(\mathbf{z})\right),$$

donde $g(\mathbf{z})$ es una Gaussiana sin normalizar y $\mathcal{Q}^{\setminus i}(\mathbf{z})$ es la función de cavidad,

$$\mathcal{Q}^{\setminus i}(\mathbf{z}) \propto \prod_{j\neq i} \widetilde{f}_j(\mathbf{z}) \propto \frac{\mathcal{Q}(\mathbf{z})}{\widetilde{f}_i(\mathbf{z})}, \tag{3.4.6}$$

y la constante de normalización Z_i es:

$$Z_i = \int f_i(\mathbf{z})\mathcal{Q}^{\setminus i}(\mathbf{z})d\mathbf{z}. \tag{3.4.7}$$

Generalmente, encontrar el $\widetilde{f}_i(\mathbf{z})$ que minimiza la divergencia anterior es bastante complicado, por ello vamos a realizarlo en dos pasos: primero buscamos la aproximación A_i Gaussiana, que es:

$$A_i(\mathbf{z}) = \arg\min_{R\in \text{Gaussianas}} KL\left(\frac{1}{Z_i} f_i(\mathbf{z})\mathcal{Q}^{\setminus i}(\mathbf{z})\Big|\Big|R(\mathbf{z})\right), \tag{3.4.8}$$

y a continuación obtenemos que el factor aproximado que estamos buscando es

$$\widetilde{f}_i = Z_i \frac{A_i(\mathbf{z})}{\mathcal{Q}^{\setminus i}(\mathbf{z})}. \tag{3.4.9}$$

Repetimos este proceso desde $i = 1$ hasta K, siendo K el número de factores a aproximar, hasta que converjan las aproximaciones.

3.4.2 Propagación de la esperanza en procesos Gaussianos censurados

Los resultados que hay que aplicar para realizar propagación de la esperanza en los procesos Gaussianos censurados son muy similares a los que se obtienen en clasificación binaria con procesos Gaussianos usando propagación de la esperanza, sección 3.6 [Rasmussen and Williams, 2004], pues en clasificación, el posterior es también intratable, ya que la verosimilitud en clasificación no es Gaussiana:

$$p(\mathbf{y}|\ \mathbf{f}) = \prod_{i=1}^{N} \Phi(y_i f_i) = \prod_{y_i=1} \Phi(f_i) \prod_{y_i=-1} (1-\Phi(f_i)),$$

donde y_i es la etiqueta de clase cuyo valor es 1 o -1. De manera que en clasificación tienen que aproximar la función $\Phi(\cdot)$ mediante Gaussianas, que es justamente lo mismo que tenemos que hacer con los factores no Gaussianos de la verosimilitud del proceso Gaussiano censurado, pues los términos que tenemos que aproximar son:

$$1-\Phi\left(\frac{f(x_i)-l}{\sigma}\right) \text{ y } \Phi\left(\frac{f(x_i)-u}{\sigma}\right).$$

Usando la regla de Bayes y que la verosimilitud $p(\mathbf{y}|\mathbf{f})$ factoriza se tiene que

$$p(\mathbf{f}|X,\mathbf{y}) = \frac{1}{Z} p(\mathbf{f}|X)p(\mathbf{y}|\mathbf{f}) = \frac{1}{Z} p(\mathbf{f}|X) \prod_{i=1}^{N} p(y_i|f(x_i)),$$

y $p(\mathbf{f}|X) = \mathcal{N}(0, K)$; como buscamos $Q(\mathbf{f}) \approx p(\mathbf{f}|X, \mathbf{y})$ Gaussiana, no tenemos que aproximar los factores que ya son Gaussianos, por lo tanto, sólo tenemos que aproximar los factores $p(y_i|\ f(x_i))$ no Gaussianos, que vamos a aproximar por una Gaussiana sin normalizar, es decir,

$$p(y_i \mid f(x_i)) \approx \widetilde{Z}_i \mathcal{N}(f(x_i)|\widetilde{\mu}_i, \widetilde{\sigma}_i^2),$$

teniendo en cuenta que para el conjunto de factores $\{f(x_i) : l < y_i < u\}$ su aproximación es el mismo factor $(\mathcal{N}(f(x_i)|y_i, \sigma^2))$,

$$Q(\mathbf{f}) \coloneqq \frac{1}{Z_{EP}} p(\mathbf{f}|X) \prod_{i=1}^{N} \widetilde{Z}_i \mathcal{N}(f(x_i)|\widetilde{\mu}_i, \widetilde{\sigma}_i^2).$$

Antes de aplicar el algoritmo de propagación de la esperanza, tenemos que calcular primero la distribución de la aproximación $Q(\mathbf{f})$; para ello calculamos primero el producto de las aproximaciones de la verosimilitud:

$$\prod_{i=1}^{N} \widetilde{Z}_i \mathcal{N}(f(x_i)|\widetilde{\mu}_i, \widetilde{\sigma}_i^2) = \mathcal{N}(\mathbf{f}|\widetilde{\boldsymbol{\mu}}, \widetilde{\Sigma}) \prod_{i=1}^{N} \widetilde{Z}_i,$$

donde $\widetilde{\boldsymbol{\mu}}_i = \widetilde{\mu}_i$ y $\widetilde{\Sigma}$ es una matriz diagonal cuyo elemento i-ésimo es $\widetilde{\sigma}_i^2$. A continuación, enunciamos un resultado auxiliar que vamos a usar repetidamente (véase Apéndice A.2 de [Rasmussen and Williams, 2004]):

$$\mathcal{N}(\mathbf{z}|\boldsymbol{\mu}_1, \Sigma_1) \cdot \mathcal{N}(\mathbf{z}|\boldsymbol{\mu}_2, \Sigma_2) = C\mathcal{N}(\mathbf{z}|\boldsymbol{\mu}, \Sigma), \tag{3.4.10}$$

donde $\mathbf{z} \in \mathbb{R}^d$ y

$$\begin{aligned}
\Sigma &= (\Sigma_1^{-1} + \Sigma_2^{-1})^{-1}, \\
\boldsymbol{\mu} &= \Sigma(\Sigma_1^{-1}\boldsymbol{\mu}_1 + \Sigma_2^{-1}\boldsymbol{\mu}_2), \\
C &= \sqrt{\frac{|\Sigma|}{(2\pi)^d |\Sigma_1||\Sigma_2|}} \exp\left[-\frac{1}{2}(\boldsymbol{\mu}_1^T \Sigma_1^{-1} \boldsymbol{\mu}_1 + \boldsymbol{\mu}_2^T \Sigma_2^{-2} \boldsymbol{\mu}_2 - \boldsymbol{\mu}^T \Sigma^{-1} \boldsymbol{\mu})\right].
\end{aligned}$$

Por lo tanto, en el producto $p(\mathbf{f}|X)\mathcal{N}(\mathbf{f}|\widetilde{\boldsymbol{\mu}}, \widetilde{\Sigma})$, donde $p(\mathbf{f}|X) = \mathcal{N}(\mathbf{f}|0, K)$, se tiene que

$$\mathcal{N}(\mathbf{f}|0, K)\mathcal{N}(\mathbf{f}|\widetilde{\boldsymbol{\mu}}, \widetilde{\Sigma}) = C\mathcal{N}(\mathbf{f}|\boldsymbol{\mu}, \Sigma),$$

donde

$$\begin{aligned}
\Sigma &= (K^{-1} + \widetilde{\Sigma}^{-1})^{-1}, \\
\boldsymbol{\mu} &= \Sigma\widetilde{\Sigma}^{-1}\widetilde{\boldsymbol{\mu}};
\end{aligned}$$

por lo tanto, hemos visto que

$$Q(\mathbf{f}) = \left(\frac{C}{Z_{EP}} \prod_{i=1}^{N} \widetilde{Z}_i\right) \mathcal{N}(\mathbf{f}|\boldsymbol{\mu}, \Sigma).$$

Puesto que $Q(\mathbf{f})$ es una función de densidad, debe integrar a uno, y por lo tanto se ha de

cumplir que

$$\begin{aligned}1 &= \int_{\mathbb{R}^N} Q(\mathbf{f})d\mathbf{f} \\ &= \int_{\mathbb{R}^N} \left(\frac{C}{Z_{EP}}\prod_{i=1}^{N}\widetilde{Z}_i\right)\mathcal{N}(\mathbf{f}|\boldsymbol{\mu},\Sigma)d\mathbf{f} \\ &= \frac{C}{Z_{EP}}\prod_{i=1}^{N}\widetilde{Z}_i \int_{\mathbb{R}^N}\mathcal{N}(\mathbf{f}|\boldsymbol{\mu},\Sigma)d\mathbf{f} \\ &= \frac{C}{Z_{EP}}\prod_{i=1}^{N}\widetilde{Z}_i;\end{aligned}$$

por lo tanto, las constantes $\widetilde{Z}_i, C, Z_{EP}$ se han de cancelar y no es necesario calcularlas.

Con estos resultados ya podemos empezar a realizar los cálculos para el algoritmo de propagación de la esperanza según la subsección 3.4.1. Para ello tenemos que calcular la distribución de la función de cavidad (3.4.6), teniendo en cuenta que en esta subsección $Q^{\setminus i}(\mathbf{z})$ es $Q^{\setminus i}(f(x_i))$, la cual es una normal de acuerdo con los resultados de la sección 3.6 de [Rasmussen and Williams, 2004]:

$$Q^{\setminus i}(f(x_i)) \propto \int p(\mathbf{f}|X)\prod_{i\neq j}\widetilde{Z}_j\mathcal{N}(f(x_j)|\widetilde{\mu}_j,\widetilde{\sigma}_j^2)d\mathbf{f}_{-i} = \mathcal{N}(f(x_i)|\mu_{\setminus i},\sigma^2_{\setminus i}),$$

donde, denotando $\Sigma_{ii} = \sigma_i^2$ y $\boldsymbol{\mu}_i = \mu_i$, se tiene que

$$\begin{aligned}\sigma^2_{\setminus i} &= (\sigma_i^{-2} - \widetilde{\sigma}_i^{-2})^{-1}, \\ \mu_{\setminus i} &= \sigma^2_{\setminus i}(\sigma_i^{-2}\mu_i - \widetilde{\sigma}_i^{-2}\widetilde{\mu}_i).\end{aligned}$$

Como tenemos tres tipos de factores que aproximar: $\Phi\left(\frac{f(x_i)-u}{\sigma}\right)$ correspondientes a $y_i = u$, $1-\Phi\left(\frac{f(x_i)-l}{\sigma}\right)$ correspondientes a $y_i = l$ y $\mathcal{N}(f(x_i)|y_i,\sigma^2)$ correspondientes a $y_i \in (l,u)$; vamos a ir calculándolos de uno en uno.

No vamos a tener que calcular la aproximación del factor $\mathcal{N}(f(x_i)|y_i,\sigma^2)$ pues es Gaussiano, y como hemos mencionado antes, la aproximación que se obtiene es él mismo ya que el factor aproximado que hemos llamado $\widetilde{f}_i$ en la sección 3.4.1 es:

$$\mathcal{N}(f(x_i)|y_i,\sigma^2) = \underset{g\in\text{Gaussianas no normalizadas}}{\arg\min} KL\left(\frac{1}{Z_i}Q^{\setminus i}\mathcal{N}(f(x_i)|y_i,\sigma^2)\Big|\Big|\frac{1}{Z_i}Q^{\setminus i}g(f(x_i))\right);$$

por lo tanto, para los ejemplos $y_i \in (l,u)$ se tiene que

$$\widetilde{\mu}_i = y_i \text{ y } \widetilde{\sigma}_i^2 = \sigma^2.$$

Para los otros factores vamos a tener que realizar más cálculos; vamos a continuar con la aproximación de los factores correspondientes a $y_i = u$. Para ello, tenemos que calcular Z_i como en (3.4.7), teniendo en cuenta que la $f_i(\mathbf{z})$ es $\Phi\left(\frac{f(x_i)-u}{\sigma}\right)$; para calcular la Z_i hay que resolver la siguiente integral:

$$Z_i = \int_{\mathbb{R}}\Phi\left(\frac{f(x_i)-u}{\sigma}\right)\mathcal{N}(f(x_i)|\mu_{\setminus i},\sigma^2_{\setminus i})df(x_i),$$

para lo que vamos a usar los resultados de la sección 3.9 [Rasmussen and Williams, 2004], que nos dicen que:

$$\int_{\mathbb{R}} \Phi\left(\frac{f(x_i)-u}{\sigma}\right)\mathcal{N}(f(x_i)|\mu_{\setminus i},\sigma^2_{\setminus i})df(x_i) = \Phi\left(\frac{\mu_{\setminus i}-u}{\sigma\sqrt{1+\sigma^2_{\setminus i}/\sigma^2}}\right) = \Phi(z_i^u),$$

con

$$z_i^u = \frac{\mu_{\setminus i}-u}{\sigma\sqrt{1+\sigma^2_{\setminus i}/\sigma^2}}.$$

A continuación tenemos que buscar $A_i = \mathcal{N}(f(x_i)|\hat{\mu}_i,\hat{\sigma}_i^2)$ que minimice la divergencia de Kullback-Leibler $KL\left(\frac{1}{Z_i}Q^{\setminus i}\Phi\left(\frac{f(x_i)-u}{\sigma}\right)||A_i\right)$ como en (3.4.8). Para ello, vamos a usar los resultados de la sección 3.9 [Rasmussen and Williams, 2004], que nos dicen que

$$\hat{\mu}_i = \mu_{\setminus i} + \frac{\sigma^2_{\setminus i}\mathcal{N}(z_i^u|0,1)}{\Phi(z_i^u)\sigma\sqrt{1+\sigma^2_{\setminus i}/\sigma^2}},$$
$$\hat{\sigma}_i^2 = \sigma^2_{\setminus i} - \frac{\sigma^4_{\setminus i}\mathcal{N}(z_i^u|0,1)}{(\sigma^2_{\setminus i}+\sigma^2)\Phi(z_i^u)}\left(z_i^u + \frac{\mathcal{N}(z_i^u|0,1)}{\Phi(z_i^u)}\right).$$

Con estos resultados hemos calculado la A_i y la Z_i cuando la variable regresora y_i tiene el valor u; no obstante, como los resultados del Rasmussen que hemos usado antes para calcular Z_i y la aproximación A_i son válidos para $\sigma \in \mathbb{R}$, vamos a usarlos para σ negativo. Aplicando que $\Phi(-z) = 1-\Phi(z)$ se tiene que

$$1-\Phi\left(\frac{f(x_i)-l}{\sigma}\right) = \Phi\left(-\frac{f(x_i)-l}{\sigma}\right) = \Phi\left(\frac{f(x_i)-l}{-\sigma}\right);$$

por lo tanto, para calcular Z_i y A_i para los factores correspondientes a $y_i = l$ cuya expresión es $1-\Phi\left(\frac{f(x_i)-l}{\sigma}\right)$, podemos usar los mismos resultados que antes, sustituyendo σ por $-\sigma$ y u por l:

$$\begin{aligned} Z_i &= \int_{\mathbb{R}}\left(1-\Phi\left(\frac{f(x_i)-l}{\sigma}\right)\right)\mathcal{N}(f(x_i)|\mu_{\setminus i},\sigma^2_{\setminus i})df(x_i) \\ &= \int_{\mathbb{R}}\Phi\left(\frac{f(x_i)-l}{-\sigma}\right)\mathcal{N}(f(x_i)|\mu_{\setminus i},\sigma^2_{\setminus i})df(x_i) \\ &= \Phi\left(\frac{\mu_{\setminus i}-l}{-\sigma\sqrt{1+\sigma^2_{\setminus i}/\sigma^2}}\right) = 1-\Phi(z_i^l) \end{aligned}$$

donde $z_i^l = \frac{\mu_{\setminus i}-l}{\sigma\sqrt{1+\sigma^2_{\setminus i}/\sigma^2}}$, y $A_i = \mathcal{N}(f(x_i)|\hat{\mu}_i,\hat{\sigma}_i^2)$ con

$$\hat{\mu}_i = \mu_{\setminus i} - \frac{\sigma^2_{\setminus i}\mathcal{N}(z_i^l|0,1)}{1-\Phi(z_i^l)\sigma\sqrt{1+\sigma^2_{\setminus i}/\sigma^2}},$$
$$\hat{\sigma}_i^2 = \sigma^2_{\setminus i} - \frac{\sigma^4_{\setminus i}\mathcal{N}(z_i^l|0,1)}{(\sigma^2_{\setminus i}+\sigma^2)(1-\Phi(z_i^l))}\left(-z_i^l + \frac{\mathcal{N}(z_i^l|0,1)}{1-\Phi(z_i^l)}\right).$$

Con esto hemos calculado la aproximación A_i para los factores correspondientes a $y_i = l$ y $y_i = u$; ahora, siguiendo (3.4.9) es necesario calcular la aproximación del factor i, $\widetilde{Z}_i\mathcal{N}(f(x_i)|\widetilde{\mu}_i, \widetilde{\sigma}_i^2)$, que es,

$$\widetilde{Z}_i\mathcal{N}(f(x_i)|\widetilde{\mu}_i, \widetilde{\sigma}_i^2) = Z_i\frac{A_i}{Q^{\setminus i}},$$

con

$$\begin{aligned}\widetilde{\sigma}_i^2 &= (\hat{\sigma}_i^{-2} - \sigma_{\setminus i}^{-2})^{-1},\\ \widetilde{\mu}_i &= \widetilde{\sigma}_i^2(\hat{\sigma}_i^{-2}\hat{\mu}_i - \sigma_{\setminus i}^{-2}\mu_{\setminus i}).\end{aligned}$$

Finalmente iteramos este proceso hasta que converjan las aproximaciones anteriores.

3.4.3 Predicción

Una vez que ya hemos aproximado el posterior $p(\mathbf{f}|X, \mathbf{y})$, podemos calcular la distribución de la predicción. Sea $\hat{x}$ un nuevo ejemplo para el que queremos calcular la distribución de la predicción; usando primero marginalización y después la aproximación $Q(\mathbf{f})$ de la subsección 3.4.2, se tiene que

$$\begin{aligned}p(\hat{f}|\hat{x}, X, \mathbf{y}) &= \int p(\hat{f}, \mathbf{f}|\hat{x}, X, \mathbf{y})d\mathbf{f}\\ &= \int p(\hat{f}|\hat{x}, \mathbf{f})p(\mathbf{f}|X, \mathbf{y})d\mathbf{f}\\ &\approx \int p(\hat{f}|\hat{x}, \mathbf{f})Q(\mathbf{f})d\mathbf{f},\end{aligned}$$

donde $Q(\mathbf{f})$ es la aproximación calculada anteriormente. Para resolver la integral anterior, vamos a usar la Proposición 1 que nos permitía calcular la distribución de b sabiendo que $b|a$ y a son Gaussianas, teniendo en cuenta que en este caso

$$\begin{aligned}p(b|a) = p(\hat{f}|\hat{x}, \mathbf{f}) &= \mathcal{N}(\hat{f}|\mathbf{k}^T K^{-1}\mathbf{f}, k(\hat{x}, \hat{x}) - \mathbf{k}^T K^{-1}\mathbf{k}),\\ p(a) = Q(\mathbf{f}) &= \mathcal{N}(\mathbf{f}|\mu, \Sigma),\end{aligned}$$

donde $\mathbf{k}_i = k(x_i, x_i)$; por lo tanto, usando los resultados (2.4.2) y (2.4.3) se tiene que

$$p(\hat{f}|\hat{x}, X, \mathbf{y}) \approx \mathcal{N}(\hat{f}|\hat{\mu}, \hat{\sigma}^2) = q(\hat{f}|\hat{x}, X, \mathbf{y}),$$

donde

$$\begin{aligned}\hat{\mu} &= \mathbf{k}^T(K + \widetilde{\Sigma})^{-1}\widetilde{\boldsymbol{\mu}},\\ \hat{\sigma}^2 &= k(\hat{x}, \hat{x}) - \mathbf{k}^T(K + \widetilde{\Sigma})^{-1}\mathbf{k},\end{aligned}$$

Con esto, hemos visto la predicción para la variable latente; para calcular la distribución de la variable observada tenemos que incorporar la verosimilitud mediante marginalización, es decir,

$$\begin{aligned}p(\hat{y}|\hat{x}, X, \mathbf{y}) &= \int_{\mathbb{R}} p(\hat{y}, \hat{f}|\hat{x}, X, \mathbf{y})d\hat{f}\\ &= \int_{\mathbb{R}} p(\hat{y}|\hat{f})p(\hat{f}|\hat{x}, X, \mathbf{y})d\hat{f}\\ &\approx \int_{\mathbb{R}} p(\hat{y}|\hat{f})q(\hat{f}|\hat{x}, X, \mathbf{y})d\hat{f}\\ &= q(\hat{y}|\hat{x}, X, \mathbf{y}).\end{aligned}$$

Puesto que $p(\hat{y}|\hat{f})$ es una función definida a trozos, el posterior aproximado $q(\hat{y}|\hat{x}, X, \mathbf{y})$ también es una función definida a trozos; por lo tanto, hay que separar en tres casos la integral. Para los casos $y = l$ o $y = u$, usando los resultados de la sección 3.9 [Rasmussen and Williams, 2004], se tiene que

$$\begin{aligned} q(\hat{y} = l|\hat{x}, X, \mathbf{y}) &= \int_{\mathbb{R}} \left(1 - \Phi\left(\frac{f - l}{\sigma}\right)\right) \delta_l(y) \mathcal{N}(\hat{f}|\hat{\mu}, \hat{\sigma}^2) d\hat{f} \\ &= \left(1 - \Phi\left(\frac{\hat{\mu} - l}{\sqrt{\sigma^2 + \hat{\sigma}^2}}\right)\right) \delta_l(y), \end{aligned} \tag{3.4.11}$$

$$\begin{aligned} q(\hat{y} = u|\hat{x}, X, \mathbf{y}) &= \int_{\mathbb{R}} \Phi\left(\frac{f - u}{\sigma}\right) \delta_u(y) \mathcal{N}(\hat{f}|\hat{\mu}, \hat{\sigma}^2) d\hat{f} \\ &= \Phi\left(\frac{\hat{\mu} - u}{\sqrt{\sigma^2 + \hat{\sigma}^2}}\right) \delta_u(y), \end{aligned} \tag{3.4.12}$$

y para el caso $l < y < u$ se tiene que

$$\begin{aligned} q(\hat{y} = y|\hat{x}, X, \mathbf{y}) &= \int_{\mathbb{R}} \mathcal{N}(y|\hat{f}, \sigma^2) \mathcal{N}(\hat{f}|\hat{\mu}, \hat{\sigma}^2) d\hat{f} \\ &= \int_{\mathbb{R}} \mathcal{N}(\hat{f}|\ y, \sigma^2) \mathcal{N}(\hat{f}|\hat{\mu}, \hat{\sigma}^2) d\hat{f} \\ &= \int_{\mathbb{R}} C\mathcal{N}(\hat{f}|\mu_*(y), \sigma_*^2(y)) d\hat{f} = C(y), \end{aligned}$$

donde hemos aplicado (3.4.10) al producto $\mathcal{N}(\hat{f}|\ y, \sigma^2)\mathcal{N}(\hat{f}|\hat{\mu}, \hat{\sigma}^2)$ de manera que $C(y), \mu_*(y), \sigma_*^2(y)$ son:

$$\begin{aligned} \sigma_*^2(y) &= (\sigma^{-2} + \hat{\sigma}^{-2})^{-1} = \frac{\sigma^2 \hat{\sigma}^2}{\sigma^2 + \hat{\sigma}^2}, \\ \mu_*(y) &= \sigma_*^2(\sigma^{-2} y + \hat{\sigma}^{-2} \hat{\mu}) = \frac{\sigma^2 \hat{\mu} + \hat{\sigma}^2 y}{\sigma^2 + \hat{\sigma}^2}, \\ C(y) &= \sqrt{\frac{\sigma_*^2}{2\pi\sigma^2\hat{\sigma}^2}} \exp\left[-\frac{1}{2}\left(\frac{y^2}{\sigma^2} + \frac{\hat{\mu}^2}{\hat{\sigma}^2} - \frac{\mu_*^2}{\sigma_*^2}\right)\right] \\ &= \sqrt{\frac{1}{2\pi(\sigma^2 + \hat{\sigma}^2)}} \exp\left[-\frac{1}{2}\left(\frac{y^2}{\sigma^2} + \frac{\hat{\mu}^2}{\hat{\sigma}^2} - \frac{(\sigma^2\hat{\mu} + \hat{\sigma}^2 y)^2}{\sigma^2\hat{\sigma}^2(\sigma^2 + \hat{\sigma}^2)}\right)\right] \\ &= \sqrt{\frac{1}{2\pi(\sigma^2 + \hat{\sigma}^2)}} \exp\left[\frac{\sigma^2\hat{\sigma}^2(y - \hat{\mu})^2}{2\sigma^2\hat{\sigma}^2(\sigma^2 + \hat{\sigma}^2)}\right] = \mathcal{N}(y|\hat{\mu}, \sigma^2 + \hat{\sigma}^2); \end{aligned} \tag{3.4.13}$$

por (3.4.11), (3.4.12) y (3.4.13), se tiene que la medida de Lebesgue-Stieltjes (dm) asociada a la función de distribución aproximada de $\hat{y}|\hat{x}, X, \mathbf{y}$ se puede escribir como

$$dm = \left(1 - \Phi\left(\frac{\hat{\mu} - l}{\sqrt{\sigma^2 + \hat{\sigma}^2}}\right)\right) d\delta_l(y) + \Phi\left(\frac{\hat{\mu} - u}{\sqrt{\sigma^2 + \hat{\sigma}^2}}\right) d\delta_u(y) + \mathbb{1}_{(l,u)}(y) \mathcal{N}(y|\hat{\mu}, \sigma^2 + \hat{\sigma}^2) dy,$$

donde dy es la medida de Lebesgue y $\delta(\cdot)$ es la delta de Dirac. La anterior expresión se parece a la expresión (3.4.4) que habíamos obtenido antes; por lo tanto, podemos decir lo mismo: la aproximación de la variable predictiva ($\hat{y}$) condicionada por la matriz de datos X, el vector de observaciones $\mathbf{y}$ y el dato $\hat{x}$ sigue una distribución continua en el intervalo (l, u) y discreta en l y u.

Ahora tenemos que calcular la media y la mediana de la distribución $\hat{y} = y|\hat{x}, X, \mathbf{y}$ para obtener la predicción que minimiza el riesgo en L^2 y en L^1 respectivamente. Para calcular la mediana, basta notar que se cumplen las siguientes desigualdades:

$$1 - \Phi\left(\frac{\hat{\mu} - l}{\sqrt{\sigma^2 + \hat{\sigma}^2}}\right) \geq 0.5 \iff \hat{\mu} \leq l,$$
$$\Phi\left(\frac{\hat{\mu} - u}{\sqrt{\sigma^2 + \hat{\sigma}^2}}\right) \geq 0.5 \iff \hat{\mu} \geq u;$$

por lo tanto, la mediana cuando $\hat{\mu} \leq l$ o $\hat{\mu} \geq u$ es l y u respectivamente; para el caso $\hat{\mu} \in (l, u)$, se tiene que

$$\begin{aligned} P(\hat{y} \leq z|\hat{x}, X, \mathbf{y}) &= p(\hat{y} = l|\hat{x}, X, \mathbf{y}) + p(l < \hat{y} \leq z|\hat{x}, X, \mathbf{y}) \\ &= 1 - \Phi\left(\frac{\hat{\mu} - l}{\sqrt{\sigma^2 + \hat{\sigma}^2}}\right) + \int_l^z \mathcal{N}(y|\hat{\mu}, \sigma^2 + \hat{\sigma}^2)dy \\ &= 1 - \Phi\left(\frac{\hat{\mu} - l}{\sqrt{\sigma^2 + \hat{\sigma}^2}}\right) + \Phi\left(\frac{z - \hat{\mu}}{\sqrt{\sigma^2 + \hat{\sigma}^2}}\right) - \Phi\left(\frac{l - \hat{\mu}}{\sqrt{\sigma^2 + \hat{\sigma}^2}}\right) \\ &= \Phi\left(\frac{z - \hat{\mu}}{\sqrt{\sigma^2 + \hat{\sigma}^2}}\right), \end{aligned}$$

donde $z \in (l, u)$ y en la tercera línea se ha usado que $1 - \Phi(x) = \Phi(-x)$. La mediana en este caso es $\hat{\mu}$ pues

$$\Phi\left(\frac{z - \hat{\mu}}{\sqrt{\sigma^2 + \hat{\sigma}^2}}\right) \geq 0.5 \iff z \geq \hat{\mu},$$
$$\Phi\left(\frac{z - \hat{\mu}}{\sqrt{\sigma^2 + \hat{\sigma}^2}}\right) \leq 0.5 \iff z \leq \hat{\mu}.$$

Si juntamos todos los resultados se tiene que

$$\text{mediana}(\hat{y}|\hat{x}, X, \mathbf{y}) = \text{máx}(\text{mín}(\mathbb{E}_{\hat{f}}(\hat{f}|\hat{x}, X, \mathbf{y}), u), l),$$

es decir, la mediana de la predicción de la variable observada la obtenemos haciendo un clip en el intervalo $[l, u]$ de la media de la variable latente predicha.

Para calcular la media de $\hat{y}|\hat{x}, X, \mathbf{y}$, hay que separar la integral $\mathbb{E}_{\hat{y}}(q(\hat{y}|\hat{x}, X, \mathbf{y})))$ en tres:

$$\begin{aligned} \mathbb{E}_{\hat{y}}(q(\hat{y}|\hat{x}, X, \mathbf{y}))) &= \int_{\mathbb{R}} y\left[1 - \Phi\left(\frac{\hat{\mu} - l}{\sqrt{\sigma^2 + \hat{\sigma}^2}}\right)\right] d\delta(y - l) + \int_{\mathbb{R}} y\Phi\left(\frac{\hat{\mu} - u}{\sqrt{\sigma^2 + \hat{\sigma}^2}}\right) d\delta(y - u) \\ &+ \int_{\mathbb{R}} y\mathbb{1}_{(l,u)}(y)\mathcal{N}(y|\hat{\mu}, \sigma^2 + \hat{\sigma}^2)dy \\ &= l\left[1 - \Phi\left(\frac{\hat{\mu} - l}{\sqrt{\sigma^2 + \hat{\sigma}^2}}\right)\right] + u\Phi\left(\frac{\hat{\mu} - u}{\sqrt{\sigma^2 + \hat{\sigma}^2}}\right) + I, \end{aligned}$$

donde I es:

$$\begin{aligned} I &= \int_l^u y\mathcal{N}(y|\hat{\mu}, \sigma^2 + \hat{\sigma}^2)dy \\ &= \int_l^u (y - \hat{\mu})\mathcal{N}(y|\hat{\mu}, \sigma^2 + \hat{\sigma}^2)dy + \int_l^u \hat{\mu}\mathcal{N}(y|\hat{\mu}, \sigma^2 + \hat{\sigma}^2)dy \\ &= -(\sigma^2 + \hat{\sigma}^2)\int_l^u -\frac{(y - \hat{\mu})}{\sigma^2 + \hat{\sigma}^2}\mathcal{N}(y|\hat{\mu}, \sigma^2 + \hat{\sigma}^2)dy + \hat{\mu}\int_l^u \mathcal{N}(y|\hat{\mu}, \sigma^2 + \hat{\sigma}^2)dy \\ &= -(\sigma^2 + \hat{\sigma}^2)\int_l^u \frac{d(\mathcal{N}(y|\hat{\mu}, \sigma^2 + \hat{\sigma}^2))}{dy}dy + \hat{\mu}\left(\Phi\left(\frac{u - \hat{\mu}}{\sqrt{\sigma^2 + \hat{\sigma}^2}}\right) - \Phi\left(\frac{l - \hat{\mu}}{\sqrt{\sigma^2 + \hat{\sigma}^2}}\right)\right) \\ &= (\sigma^2 + \hat{\sigma}^2)(\mathcal{N}(l|\hat{\mu}, \sigma^2 + \hat{\sigma}^2) - \mathcal{N}(u|\hat{\mu}, \sigma^2 + \hat{\sigma}^2)) + \hat{\mu}\left(\Phi\left(\frac{u - \hat{\mu}}{\sqrt{\sigma^2 + \hat{\sigma}^2}}\right) - \Phi\left(\frac{l - \hat{\mu}}{\sqrt{\sigma^2 + \hat{\sigma}^2}}\right)\right). \end{aligned}$$

Por lo tanto, se tiene que

$$\mathbb{E}_{\hat{y}}(q(\hat{y}|\hat{x}, X, \mathbf{y}))) = (l - \hat{\mu})\Phi\left(\frac{l - \hat{\mu}}{\sqrt{\sigma^2 + \hat{\sigma}^2}}\right) + u\Phi\left(\frac{\hat{\mu} - u}{\sqrt{\sigma^2 + \hat{\sigma}^2}}\right) + \hat{\mu}\Phi\left(\frac{u - \hat{\mu}}{\sqrt{\sigma^2 + \hat{\sigma}^2}}\right)$$
$$+ (\sigma^2 + \hat{\sigma}^2)(\mathcal{N}(l|\hat{\mu}, \sigma^2 + \hat{\sigma}^2) - \mathcal{N}(u|\hat{\mu}, \sigma^2 + \hat{\sigma}^2)).$$

Capítulo 4

Experimentos en energía eólica

4.1 Predicción con procesos Gaussianos en regresión

En esta sección consideraremos dos problemas de predicción de energía renovable: la predicción de energía eólica en el parque Sotavento, situado en la zona noroeste de España, y de la producción de energía eólica en toda la Peninsula Ibérica.

La variable de regresión en ambos casos va a ser la producción de energía eólica horaria durante los años 2013, 2014 y 2015, cuyo valor se ha normalizado al intervalo $[0, 1]$. Las producciones de energía eólica de Sotavento se pueden descargar de su página web y las de la Península Ibérica han sido proporcionadas por Red Eléctrica de España, REE.

Como características para los dos problemas hemos usado como variables predictivas los datos obtenidos mediante un modelo numérico de predicción meteorológica (Numerical Weather Prediction, NWP) para el año 2013, 2014 y 2015, los cuales han sido descargados del repositorio MARS del ECMWF's (European Center for Medium Weather Forecasts). Más información sobre el sistema NWP del ECMWF se puede encontrar en las referencias [Catalina and Dorronsoro, 2017] y [Díaz-Vico et al., 2017]

Hemos usado una rejilla 8×15 con una resolución de $0.25°$ para Sotavento, y una rejilla 18×29 con resolución $0.5°$ para la Península Ibérica; en cada punto de la rejilla se tienen las siguientes variables:

- Las componentes U y V de la velocidad del viento, así como sus módulos, a 10 y 100 metros.
- La presión superficial.
- Temperatura a 2 metros.
- La conversión de viento a potencia a 10 y 100 metros, usando una curva genérica de potencia que satura a 20 m/s y tiene un punto de corte a 25 m/s.

Por lo tanto, para Sotavento tenemos $8 \times 15 \times 10 = 1200$ variables y para Red Eléctrica $18 \times 29 \times 10 = 5220$ variables; además, hemos normalizado las variables NWP con media cero y desviación típica 1. Por cada año se tiene aproximadamente 8760 patrones.

Para estos problemas vamos a usar como referencia los resultados obtenidos en [Catalina and Dorronsoro, 2017] para las Máquinas de Vectores Soporte para Regresión (SVR) y el

perceptrón multicapa (MLP) y los vamos a comparar con los resultados que se obtienen con los procesos Gaussianos. Como se ha mencionado antes, la producción de energía está siempre en el intervalo $[0, 1]$; puesto que estos algoritmos no incorporan dicha información, puede darse el caso de que la predicción para un nuevo ejemplo sea mayor que uno, o menor que 0, en cuyo caso se hace un clip para que la predicción esté en el intervalo $[0, 1]$. Es decir, si $\hat{y}$ es la predicción obtenida por uno de los anteriores algoritmos, la predicción que se usa para calcular el error es

$$y_{clip} = \text{máx}(\text{mín}(\hat{y}, 1), 0).$$

Una alternativa en teoría más sólida sería utilizar los procesos censurados descritos en la sección 3.4; esta opción se deja para trabajo futuro.

En los procesos Gaussianos, hemos probado los siguientes núcleos:

- RBF: $k(x, x') = \theta \exp\left(-\frac{1}{2\ell^2}\|x - x'\|^2\right)$.
- 2 RBF: $k(x, x') = \theta_1 \exp\left(-\frac{1}{2\ell_1^2}\|x - x'\|^2\right) + \theta_2 \exp\left(-\frac{1}{2\ell_2^2}\|x - x'\|^2\right)$.
- RationalQuadratic: $k(x, x') = \theta\left(\frac{1+\|x-x'\|^2}{2\alpha\ell^2}\right)^{-\alpha}$.
- Matérn 0.5 : $k(x, x') = \theta \exp\left(-\frac{\|x-x'\|}{\ell^2}\right)$.
- Matérn 1.5 : $k(x, x') = \theta\left(1 + \frac{\sqrt{3}\|x-x'\|}{\ell^2}\right)\exp\left(-\frac{\sqrt{3}\|x-x'\|}{\ell^2}\right)$.
- Matérn 2.5 : $k(x, x') = \theta\left(1 + \frac{\sqrt{5}\|x-x'\|}{\ell^2} + \frac{5\|x-x'\|^2}{3\ell^4}\right)\exp\left(-\frac{\sqrt{5}\|x-x'\|}{\ell^2}\right)$.

El núcleo RBF tiene 2 hiperparámetros: θ y ℓ, el núcleo RationalQuadratic tiene tres hiperparámetros: θ, ℓ y α, el núcleo Matérn en un principio, tiene dos hiperparámetros: ν y ℓ, pero como en la implementación de Scikit-learn, el valor de ν es fijo, el único hiperparámetro que se optimiza es ℓ.

Para probar los procesos Gaussianos, se ha utilizado la clase `GaussianProcessRegressor` de Scikit-learn. En esta implementación, la hiperparametrización de los procesos Gaussianos se realiza mediante la selección Bayesiana de modelos como se ha explicado en la sección 3.3.1. Puesto que la verosimilitud puede tener más de un máximo, en Scikit-learn se realiza el ascenso por gradiente con valores iniciales distintos; el número de veces que se realiza depende de la variable `num_restarts` que nosotros hemos puesto a 5. Además, en Scikit-learn la implementación es sólo para procesos Gaussianos de media cero; si se pone la variable `normalize_y` a `True`, se centra la variable observada en entrenamiento, es decir, se usa en entrenamiento la variable

$$z_i = y_i - m,$$

y en predicción se da la variable $\hat{z}_{GP} + m$ donde m es la media del vector $\mathbf{y}$ en entrenamiento y $\hat{z}_{GP}$ es la predicción del proceso Gaussiano de media cero del ejemplo $\hat{x}$ habiendo entrenado con las z_i. Nosotros hemos puesto a `False` esta variable y hemos puesto como m la media de producción eólica de los años 2013 y 2014. Idealmente lo que deberíamos hacer es restar la media histórica de la producción en el parque o en la península, pero no disponemos de dichos datos, pues claramente la producción de energía no tiene media cero.

Los parámetros de los núcleos usados en Scikit-learn asociados a la relación anterior son los siguientes:

Núcleo	**Sotavento**			**Red Eléctrica**		
	MAE (%)	LV_1	LV_2	**MAE(%)**	LV_1	LV_2
RBF	8.215	10941.307	22112.357	3.645	22303.168	46227.591
2 RBF	7.296	11254.609	22703.845	3.570	22356.146	**46413.249**
RationalQuadratic	7.322	**11281.791**	**22746.760**	3.522	**22369.858**	46348.488
Matérn 0.5	**7.139**	11031.393	22045.194	**3.305**	19537.323	40456.777
Matérn 1.5	7.473	11232.848	22684.126	3.451	22158.323	45745.373
Matérn 2.5	7.652	11166.729	22572.923	3.511	22321.393	46202.942

Tabla 4.1.1: Resultados obtenidos en los problemas de Sotavento y Red Eléctrica en validación. LV_1 quiere decir logaritmo de la verosimilitud con conjunto de entrenamiento el primer año (2013) y LV_2 logaritmo de la verosimilitud con conjunto de entrenamiento los dos primeros años (2013 y 2014).

- `ConstantKernel(constant_value =` θ`)`.
- `WhiteKernel(noise_level =` σ_n^2`)`.
- `RBF(length_scale =` ℓ`)`.
- `RationalQuadratic(alpha =` α`, length_scale =` ℓ`)`.
- `Matern(length_scale =` ℓ`, nu =` x`)`: donde $x \in \{0.5, 1.5, 2.5\}$.

Es facil construir los núcleos que hemos usado, multiplicando y sumando las funciones de scikit-learn anteriores, además para permitir que la constante de ruido se hiperparametrice es necesario sumar un `WhiteKernel` al núcleo que consideremos, pues si no la constante de ruido del proceso Gaussiano (la variable `alpha` en Scikit-learn) tiene como valor 10^{-10} por defecto y no se hiperparametriza. Nosotros hemos puesto esta variable a cero y hemos sumado a cada núcleo un `WhiteKernel`, de manera que por ejemplo el núcleo con un RBF que finalmente hemos usado es:

$$\texttt{ConstantKernel(constant\_value =}\ \theta\texttt{)*RBF(length\_scale =}\ \ell\texttt{)} + \texttt{WhiteKernel(noise\_level =}\ \sigma_n^2\texttt{)}.$$

En un principio, para elegir qué núcleo usar en test se nos pueden ocurrir distintas opciones:

1. Entrenar un proceso Gaussiano por separado, con cada núcleo con conjunto de entrenamiento el año 2013 y de validación el año 2014, quedándonos con el núcleo que menor error (MAE) tenga en validación.
2. Entrenar cada proceso Gaussiano por separado con conjunto de entrenamiento el año 2013 y 2014, eligiendo el núcleo que dé lugar a una mayor verosimilitud.
3. Entrenar un proceso Gaussiano en el que sumamos todos los núcleos, y quedarnos con los núcleos que tengan una constante mayor.

Uno de los problemas de la opción 3, es que hay que optimizar muchos hiperparámetros; además, es más probable que la verosimilitud tenga más de un máximo; por lo tanto, es más difícil que acabemos en el máximo global. Hemos decidido usar principalmente la primera opción, pues como podemos ver en la Tabla 4.1.1, tanto en Sotavento como en Red Eléctrica el núcleo con menor MAE en validación (en negrita) es distinto del que tiene menor verosimilitud en el primer año; por lo tanto, algo parecido puede pasar en

Algoritmo	MAE en Sotavento (%)	MAE en Red Eléctrica (%)
MLP (DARE)	5.86	2.76
SVR (DARE)	**5.80**	**2.54**
GP Val	6.006	2.654
GP LV_2	6.211	2.796

Tabla 4.1.2: Resultados obtenidos en los problemas de Sotavento y Red Eléctrica en test. SVR (DARE) y MLP(DARE) son los resultados obtenidos en [Catalina and Dorronsoro, 2017]. GP Val es el proceso Gaussiano con el núcleo que tiene menor error en validación (Matérn 0.5 en ambos casos) y GP LV_2 es el GP con el núcleo con mayor verosimilitud entrenando con 2013 y 2014 (RationalQuadratic para Sotavento y 2 RBF para Red Eléctrica).

test. No obstante, para ver si pasa lo mismo en test, vamos a considerar también el núcleo que tenga mayor verosimilitud con el conjunto de entrenamiento el año 2013 y 2014, es decir, mayor LV_2, el cual resulta también tener mayor LV_1 en ambos problemas. En la opción 1 escogemos el núcleo Matérn para Sotavento y Red Eléctrica pues es el que menor MAE tiene en validación y en la opción 2 escogemos el núcleo RationalQuadratic para Sotavento y 2 RBF para Red Eléctrica pues son los núcleos con mayor verosimilitud con entrenamiento 2013 y 2014, es decir, mayor LV_2.

Otro de los motivos por el cual la opción 1 es preferible es porque tiene menor coste computacional, ya que tenemos que ejecutar seis procesos Gaussianos con 8760 patrones en entrenamiento en 2013 y un proceso Gaussiano con 2×8760 para construir el modelo final con conjunto de entrenamiento 2013 y 2014. Por contra, en la opción 2 tenemos que ejecutar seis procesos Gaussianos con 2×8760 patrones. Como hemos visto en la sección 3.3, la hiperparametrización tiene coste cúbico en el número de patrones de entrenamiento, de manera que si llamamos C al coste de entrenar un proceso Gaussiano con 8760 patrones y suponemos que es el mismo para todos los núcleos, el coste de entrenar un proceso Gaussiano con 2×8760 patrones es de 2^3C y se tiene que:

$$\text{Coste}_{\text{Opción 1}} = 6C + 2^3C = 14C,$$
$$\text{Coste}_{\text{Opción 2}} = 6 \cdot 2^3C = 48C,$$

es decir, el coste de la opción 2 es 3.4 veces mayor.

En los resultados de [Catalina and Dorronsoro, 2017], en las SVR, primero, han realizado una búsqueda en rejilla de los hiperparámetros (ancho del núcleo, el valor de la constante C y la anchura del tubo ϵ) con conjunto de entrenamiento el año 2013 y con conjunto de validación el año 2014, seleccionando los hiperparámetros cuyo MAE en 2014 fuera el menor. Posteriormente, han vuelto a entrenar con los hiperparámetros óptimos del paso anterior, con conjunto de entrenamiento los años 2013 y 2014, calculando el error de test con el año 2015. En el MLP realizaron el mismo proceso de validación que en las SVR, para el valor de la regularización en L^2 de los pesos, han usado un perceptrón multicapa con 2 capas ocultas con 100 neuronas por capa en Sotavento y 4 capas ocultas con 1000 neuronas por capa en Red Eléctrica.

En la Tabla 4.1.2 se muestran los resultados obtenidos anteriormente mediante SVR y MLP en [Catalina and Dorronsoro, 2017] así como con los procesos Gaussianos: para el núcleo que menor MAE tenía en validación (Matérn 0.5 en ambos casos) y para el núcleo que tenía mayor verosimilitud con conjunto de entrenamiento 2013 y 2014 (RationalQuadratic en Sotavento y 2 RBF en Red Eléctrica).

Algoritmo	Sotavento		Red Eléctrica	
	MAE Validación(%)	MAE Test(%)	MAE Validación(%)	MAE Test(%)
MLP (DARE)	-	5.86	-	2.76
SVR (DARE)	-	5.80	-	**2.54**
SVR Matérn 0.5 ℓ CV	6.96	5.698	3.174	2.551
SVR Matérn 0.5 ℓ GP	6.993	**5.693**	3.198	2.554
SVR Matérn 0.5 ℓ, θ GP	6.995	5.711	3.174	2.547
GP Matérn 0.5	7.139	6.006	3.305	2.654

Tabla 4.2.1: Resultados obtenidos en los problemas de Sotavento y Red Eléctrica en test. SVR (DARE) y MLP(DARE) son los resultados obtenidos en [Catalina and Dorronsoro, 2017]. SVR Matérn 0.5 es la SVR con el núcleo Matérn con $\nu = 0.5$.

Podemos observar que los procesos Gaussianos obtienen mejores resultados en Red Eléctrica que el MLP pero peores en Sotavento; sin embargo, no mejoran en ningún caso los resultados de la SVR. Además, podemos observar que la opción de elegir el núcleo como aquel que tiene menor MAE en validación es mejor que la de elegir el núcleo con mayor verosimilitud, pues el MAE en test resulta menor tanto en Sotavento como en Red Eléctrica.

Por lo tanto, la elección de núcleo mediante el MAE en validación obtiene mejores resultados, que la elección de núcleo mediante máxima verosimilitud con verosimilitud Bayesiana, lo cual es esperable y además tiene menor coste.

Estos resultados pueden deberse entre otras cosas, al hecho de que la producción de energía eólica no se corresponde a un proceso Gaussiano más un ruido Gaussiano pues, como se ha mencionado anteriormente, el valor a predecir está entre $[0, 1]$ y los procesos Gaussianos nos dan una distribución en la recta real. Una alternativa, es usar los procesos Gaussianos censurados descritos en la sección 3.4, pues en Sotavento el número de ejemplos cuya producción es cero es de aproximadamente un 10 % y no hay ningún ejemplo con producción máxima.

Por otro lado, en Red Eléctrica la censura superior sería uno, pero la censura inferior sería 0.01 aproximadamente, siendo este el mínimo histórico de producción; no obstante, si consideramos que la censura es en el intervalo $[0, 1]$ y aplicamos el enfoque de los procesos Gaussianos censurados, se tiene que el posterior $\mathbf{f}|X, \mathbf{y}$ no es necesario aproximarlo en este caso, pues todos los términos de la verosimilitud serían Gaussianos, ya que no hay ningún patrón con producción cero o uno, y podríamos aplicar directamente los resultados que hemos visto en la subsección 3.4.3. De hecho, en Red Eléctrica estamos aplicando de manera indirecta los resultados de predicción del proceso Gaussiano censurado en $[0, 1]$, pues la mediana de $\hat{y}|X, \mathbf{y}$ es el clip de la media de la aproximación $\hat{f}|\hat{x}, X, \mathbf{y}$ en $[0, 1]$, que es justamente la predicción que estamos dando en el proceso Gaussiano en regresión para este problema.

Los hiperparámetros del proceso Gaussiano Matérn 0.5 en validación son aproximadamente: $\sigma_f^2 = 0.571^2, \ell = 547, \sigma_n^2 = 10^{-5}$ para Sotavento y $\sigma_f^2 = 1.3^2, \ell = 7.12 * 10^4, \sigma_n^2 = 10^{-5}$ en Red Eléctrica. En test son los siguientes: $\sigma_f^2 = 0.733^2, \ell = 880, \sigma_n^2 = 10^{-5}$ en Sotavento y $\sigma_f^2 = 1.37^2, \ell = 9.16 * 10^4, \sigma_n^2 = 10^{-5}$ en Red Eléctrica.

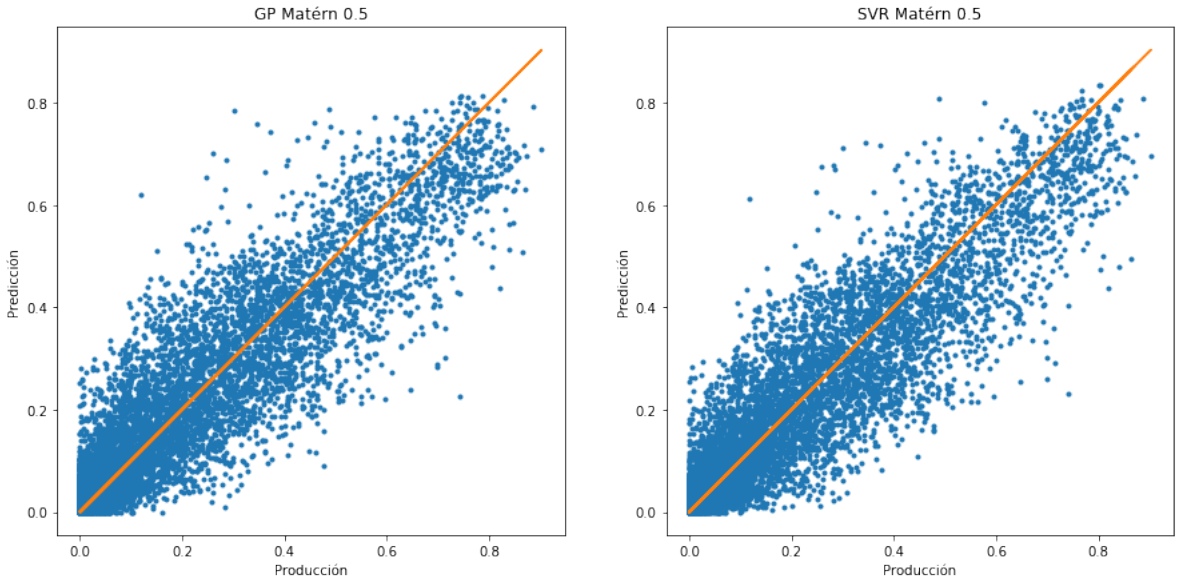

Figura 4.2.1: Gráficos de producción contra predicción para el problema de Sotavento para GP Matérn 0.5 y SVR Matérn 0.5 CV

4.2 SVR sobre núcleos usados en los procesos Gaussianos

En esta sección consideraremos usar una SVR usando el núcleo Matérn en vez del Gaussiano para los problemas de Sotavento y Red Eléctrica, pues como hemos visto en la sección 4.1, la diferencia entre un proceso Gaussiano con núcleo Matérn 0.5 y un proceso Gaussiano con núcleo RBF (Gaussiano) es significativa. Por ello cabe la posibilidad de que se obtengan mejores resultados en las SVR con otros núcleos.

Para realizar estos experimentos se ha usado la clase SVR de Scikit-learn, poniendo como variable de kernel el núcleo Matérn implementado en Scikit-learn en la librería de los procesos Gaussianos. A la hora de entrenar esta SVR se han seguido los mismos pasos que en [Catalina and Dorronsoro, 2017]; de manera que buscamos primero los hiperparámetros óptimos entrenando con el año 2013 y validando sobre 2014, posteriormente entrenamos con 2013 y 2014 con estos hiperparámetros para predecir sobre 2015, a este modelo lo hemos llamado SVR Matérn 0.5 ℓ CV.

Puesto que la SVR tiene dos hiperparámetros (C y ϵ), el núcleo Matérn tiene dos hiperparámetros (ν, ℓ) y hemos fijado el ν a 0.5, lo que hemos hecho es realizar la búsqueda en rejilla del ϵ, C, ℓ, eligiendo los hiperparámetros con menor error en validación en 2014, cuyos resultados se muestran en la Tabla 4.2.1. Los resultados de esta sección y de la sección 4.1 han llevado a publicar el artículo [de la Pompa et al.,].

Podemos observar, que la SVR con el núcleo Matérn obtiene mejores resultados que los obtenidos en el proceso Gaussiano. Además obtiene mejores resultados en Sotavento que la SVR y el MLP de [Catalina and Dorronsoro, 2017] y muy ligeramente peor que la SVR de [Catalina and Dorronsoro, 2017] en Red Eléctrica pero mejor que el MLP.

En la sección 4.1 hemos visto que el mejor núcleo era el Matérn con $\nu = 0.5$ y con un ℓ y θ que da el proceso Gaussiano, resulta interesante preguntarse si estos dos hiperparámetros los podríamos llevar a la SVR para así tener únicamente que estimar dos hiperparámetros. Para ello, hemos usado los hiperparámetros óptimos del GP con el año 2013 y realizado el

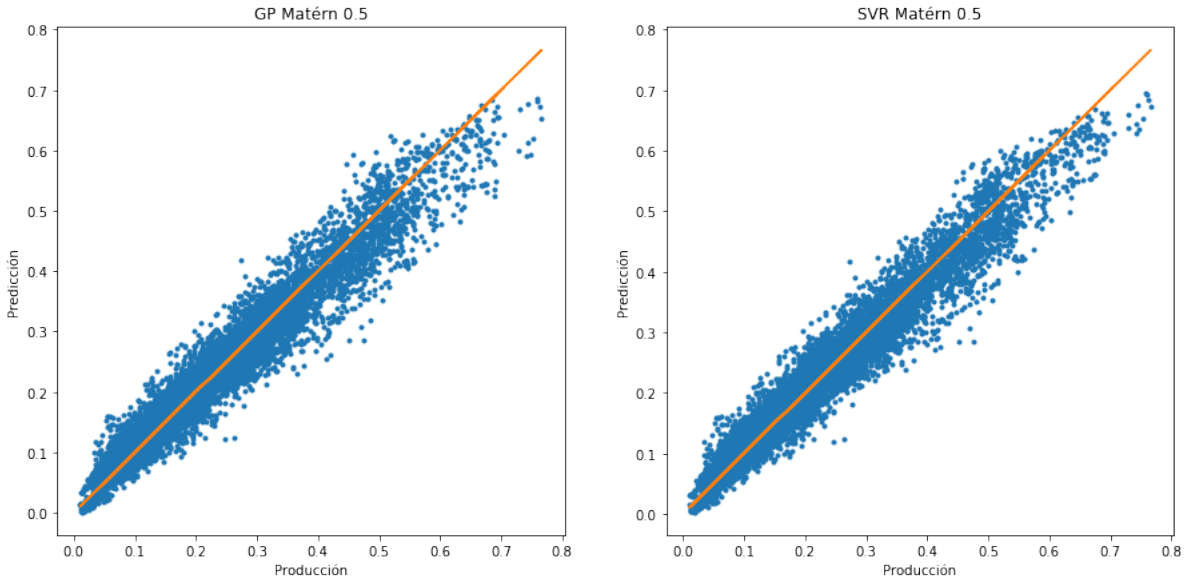

Figura 4.2.2: Gráficos de producción contra predicción para el problema de Red Eléctrica para GP Matérn 0.5 y SVR Matérn 0.5 CV

Algoritmo	Sotavento				Red Eléctrica			
	C	ϵ	ℓ	θ	C	ϵ	ℓ	θ
SVR Matérn 0.5 ℓ CV	0.25	2.115e-04	138.56	1	16.0	1.445e-04	36991.78	1
SVR Matérn 0.5 ℓ GP	1.0	2.066e-07	546.95	1	16.0	2.258e-06	71152.30	1
SVR Matérn 0.5 ℓ, θ GP	4.0	8.264e-07	546.95	0.326	16.0	2.258e-06	71152.30	1.682

Tabla 4.2.2: Hiperparámetros de los modelos SVR Matérn obtenidos.

mismo proceso de validación anterior pero solo con C y ϵ. En teoría el θ no es necesario en la SVR, pues si multiplicamos por θ el C^* nuevo debería ser aproximadamente $\frac{C^*}{\sqrt{\theta}}$; no obstante, hemos probado la SVR con el θ y ℓ que nos da el proceso Gaussiano y solamente con el ℓ correspondiéndose con SVR Matérn 0.5 ℓ, θ GP y SVR Matérn 0.5 ℓ GP respectivamente.

Podemos observar, que no hay diferencias significativas entre usar o no el θ y que en ambos casos, se obtiene un error parecido al que obtiene la SVR Matérn 0.5 calculando el valor de ℓ mediante validación. Esto, significa que podemos reducir de tres a dos el número de hiperparámetros a estimar mediante una búsqueda en rejilla, que recordemos tiene coste exponencial; por lo tanto, reducimos el tiempo de ejecución. Además, surge una vía a explorar que es probar núcleos más sofisticados, con más hiperparámetros en los procesos Gaussianos y usar los hiperparámetros estimados por estos en la SVR.

En la Figura 4.2.1 y la Figura 4.2.2 se muestran los gráficos producción contra predicción para el GP y la SVR Matérn 0.5 CV para los problemas de Sotavento y Red Eléctrica, donde podemos ver que en ambos modelos tienden a subestimar producciones mayores que el 70 %.

Hemos usado inicialmente la siguiente rejilla para ϵ: $4.0^{-k} * std(y_{train})$ para k de 1 a 6. Para la constante C hemos usado la siguiente rejilla: 4.0^k para k de -5 a 5. Y para la length scale en el SVR Matérn 0.5 ℓ CV hemos usado la siguiente rejilla: $\frac{1}{\sqrt{4.0^k/D}}$ para k entre -3 y 2. En algunos casos ha salido alguno de los tres hiperparámetros en el borde del intervalo, por

Confianza	Sotavento	Red Eléctrica
20	28.727	17.548
50	58.101	43.472
80	79.230	70.419
90	85.799	79.998
95	89.557	86.014

Tabla 4.3.1: Resultados de los intervalos de incertidumbre en los problemas de Sotavento y Red Eléctrica en el año 2014 de manera directa, es decir, sin calibrar.

lo que hemos explorado un poco más (sobre todo en el ϵ). Los hiperparámetros obtenidos finalmente son los mostrados en la Tabla 4.2.2.

4.3 Incertidumbre en los GP para regresión eólica

En esta sección veremos los resultados de los intervalos de incertidumbre sobre los procesos Gaussianos: de manera directa y cambiando la media de la predicción.

4.3.1 Intervalo de incertidumbre directos sobre GPs

Como hemos visto en la sección 3.2, los procesos Gaussianos nos dan una distribucción de la predicción $(\hat{y}|\hat{x}, X, \mathbf{y}) \sim \mathcal{N}(\hat{y}|\hat{\mu}, \hat{\sigma}^2)$; por lo tanto, podemos construir los intervalos de confianza teniendo en cuenta que

$$P\left(\left|\frac{\hat{y}|\hat{x}, X, \mathbf{y} - \hat{\mu}}{\hat{\sigma}}\right| \leq \Phi^{-1}(1-\alpha/2)\right) = 1-\alpha,$$

donde $\Phi^{-1}(\cdot)$ es la inversa de la función de distribución de una normal unidimensional con media cero y varianza uno; por lo tanto, para cada ejemplo tenemos la banda de incertidumbre:

$$[\hat{\mu} - \hat{\sigma}\Phi^{-1}(1-\alpha/2), \hat{\mu} + \hat{\sigma}\Phi^{-1}(1-\alpha/2)].$$

Como se ha mencionado en la sección 4.1, las predicciones están entre $[0, 1]$; por lo tanto, hay que hacer clip también de los intervalos de incertidumbre. Definimos entonces los intervalos de incertidumbre como:

$$[\text{máx}(\text{mín}(\hat{\mu} - \hat{\sigma}\Phi^{-1}(1-\alpha/2), 1), 0), \text{máx}(\text{mín}(\hat{\mu} + \hat{\sigma}\Phi^{-1}(1-\alpha/2), 1), 0)]. \quad (4.3.1)$$

En la Tabla 4.3.1 se muestran los resultados obtenidos en 2014 con los procesos Gaussianos: para el núcleo que menor MAE tenía en validación en la sección 4.1 (Matérn 0.5 en ambos casos); en cada fila se muestra la confianza que tiene el intervalo, así como el porcentaje de ejemplos que caen en dicho intervalo. Podemos observar que en Red Eléctrica estamos subestimando la incertidumbre, pues el número de ejemplos que caen en la banda de incertidumbre con confianza α es menor que dicha confianza y en Sotavento estamos subestimando en 80, 90 y 95 pero sobreestimando en 20 y 50; esto quiere decir que la distribución real en Sotavento es un poco más picuda en la media y con colas más pesadas. Esto puede deberse entre otras cosas a que se cumpla que las variables

$$z_i = \frac{\hat{y}_i|\hat{x}, X, \mathbf{y} - \hat{\mu}_i}{\hat{\sigma}_i^2} \sim \mathcal{N}(0, 1),$$

Algoritmo 1: Modelo calibrado para una confianza s fija

Entrenamiento del modelo calibrado
Entrenar GP_{2013}
Predecir $\hat{\mu}_{2014}$ y $\hat{\sigma}^2_{2014}$
Minimizar $i_{err}(\delta_s)$ en δ_s mediante CV obteniendo δ^*_s
Test del modelo calibrado
Entrenar GP_{2014}
Predecir $\hat{\mu}_{2015}$ y $\hat{\sigma}^2_{2015}$
Devolver modelo $\hat{\mu}_{2015} \pm \delta^*_s \hat{\sigma}^2_{2015}$

Confianza	Sotavento			Red Eléctrica		
	Calibrado	δ	$1-\alpha$ (%)	Calibrado	δ	$1-\alpha$ (%)
20	20.180	0.170	13.507	20.193	0.293	23.060
50	49.682	0.530	40.386	50.357	0.800	57.615
80	80.294	1.336	81.861	79.652	1.629	89.677
90	89.950	1.997	95.421	90.124	2.232	97.440
95	94.865	2.660	99.218	94.996	2.828	99.532

Tabla 4.3.2: Resultados de los intervalos de incertidumbre en los problemas de Sotavento y Red Eléctrica en el año 2014, habiendo sido calibrados con el año 2014; además se muestran los deltas obtenidos así como los $1-\alpha$ correspondientes.

pero no sean independientes y a que estamos haciendo un clip del intervalo de confianza; por lo tanto, es necesario realizar una calibración de estos intervalos para ajustar mejor su anchura, para lo cual vamos a realizar una busqueda en rejilla para el $\delta(s)$. Puesto que calcular el α óptimo en los intervalos de (4.3.1) puede resultar bastante complicado, lo que vamos a hacer es ajustar el $\delta_s(\alpha) = \Phi^{-1}(1-\alpha/2)$. Para ello, entrenamos un proceso Gaussiano con el núcleo Matérn con $\nu = 0.5$, pues es el que menor error tiene en validación, con conjunto de entrenamiento el año 2013; posteriormente calculamos los intervalos de incertidumbre sobre el año 2014 y mediante validación en 2014 buscamos el δ_s que minimice:

$$i_{err}(\delta_s) = |\{\,\% \text{ de residuos en 2014 } \in I_{\delta_s}\} - s|,$$

donde s es la confianza (20, 50, 80, 90, 95) y I_{δ_s} se corresponde con los intervalos de incertidumbre:

$$[\text{máx}(\text{mín}(\hat{\mu} - \delta_s\hat{\sigma}, 1), 0), \text{máx}(\text{mín}(\hat{\mu} + \delta_s\hat{\sigma}, 1), 0)].$$

El algoritmo de calibración se describe en Algoritmo 1. La rejilla que hemos usado en el $\delta(s)$ es la siguiente: $\delta(s)_{init} * k$ donde k va de 0.1 hasta 1.5 con paso fijo 1.4/50 y $\delta(s)_{init} = \Phi^{-1}(1-(1-s)/2)$.

En la Tabla 4.3.2 se muestran los resultados obtenidos en 2014 tras realizar la calibración sobre 2014, así como los $\delta_s(\alpha)$ obtenidos, y el $1-\alpha$ correspondiente a dichos deltas.

Posteriormente, calculamos los intervalos de incertidumbre con los deltas anteriores sobre el año 2015, para el proceso Gaussiano con conjunto de entrenamiento 2013 y para el proceso Gaussiano con conjunto de entrenamiento 2013 y 2014, cuyos resultados se muestran en la Tabla 4.3.3. El modelo GP 1 año tiene unos intervalos de incertidumbre mejores que el GP 2 años; no obstante, el modelo GP 2 años tiene un error (MAE) menor pues ha sido entrenado con dos años y tiene más información; por lo tanto, es preferible el segundo modelo.

Podemos observar que la calibración en 2014 nos sirve para el año 2015, pues los intervalos son mejores que sin hacer la calibración, y el error en la incertidumbre está entre un 1 %

Confianza	Sotavento		Red Eléctrica	
	GP 1 año	GP 2 años	GP 1 año	GP 2 años
20	20.700	22.295	19.716	20.336
50	51.314	51.704	51.346	52.109
80	82.593	82.846	82.916	83.429
90	92.932	92.679	93.150	93.031
95	97.338	97.235	97.081	97.034

Tabla 4.3.3: Resultados de los intervalos de incertidumbre en los problemas de Sotavento y Red Eléctrica en el año 2015, habiendo sido calibrados con el año 2014.

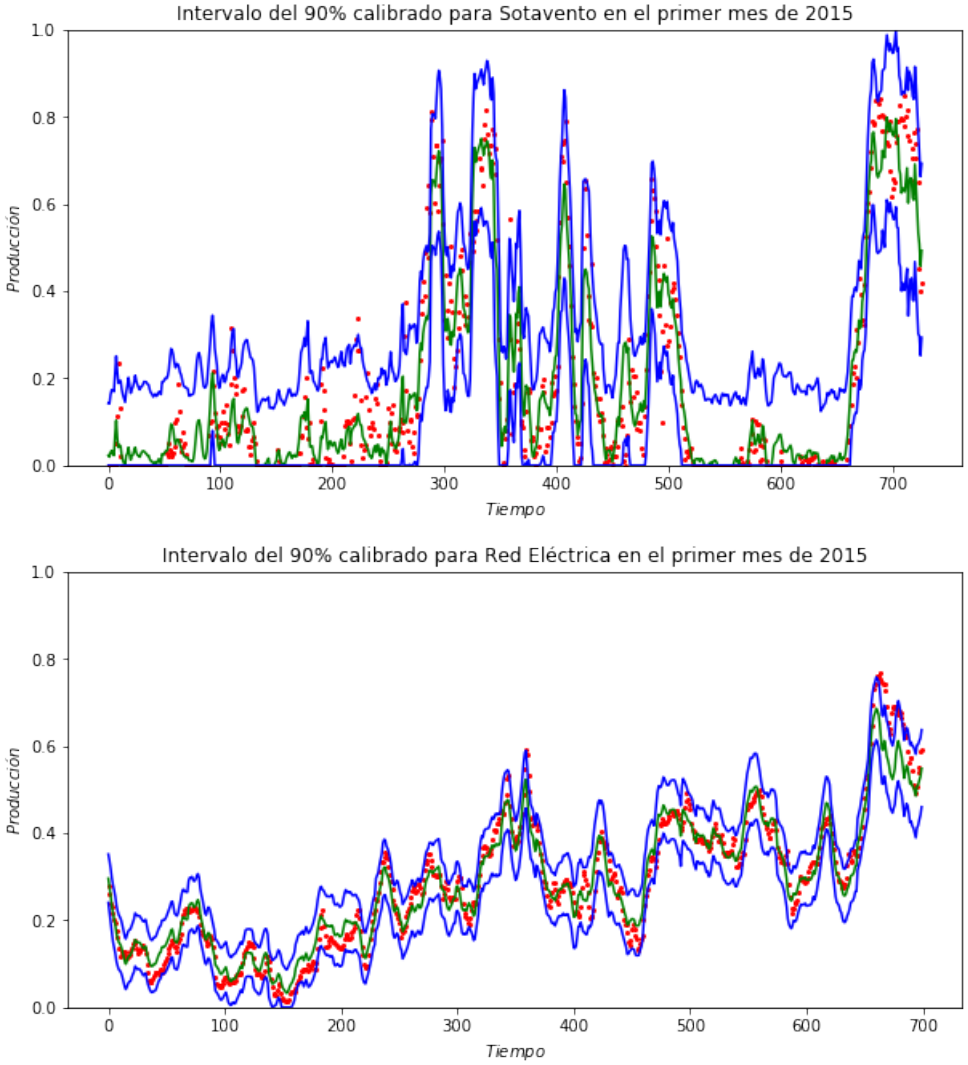

Figura 4.3.1: Intervalos de incertidumbre calibrados sobre el primer mes de 2015 de Sotavento y Red Eléctrica

y un 3 %; por lo tanto, podemos realizar una estimación razonable de los intervalos con el modelo calibrado. Notar que hemos hecho la calibración año a año; a la hora de llevarlo a la práctica, sería más apropiado realizar esta calibración más a menudo, por ejemplo mensualmente.

En la Figura 4.3.1 se muestran los intervalos de incertidumbre del 90 % en azul, sobre aproximademente el primer mes de Sotavento y Red Eléctrica. En verde se muestra la

Confianza	Sotavento			Red Eléctrica		
	Calibrado	δ	$1-\alpha$ (%)	Calibrado	δ	$1-\alpha$ (%)
20	20.261	0.141	11.225	19.788	0.286	22.506
50	49.913	0.491	37.687	49.631	0.761	55.347
80	80.259	1.336	81.861	80.284	1.593	88.879
90	90.170	1.997	95.421	90.172	2.138	96.751
95	95.016	2.660	99.218	95.127	2.660	99.218

Tabla 4.3.4: Resultados de los intervalos de incertidumbre en los problemas de Sotavento y Red Eléctrica en el año 2014 usando la media del SVR Matérn 0.5, habiendo sido calibrados con el año 2014; además se muestran los deltas obtenidos así como los $1-\alpha$ correspondientes.

Confianza	Sotavento	Red Eléctrica
20	22.628	19.657
50	52.645	51.334
80	83.626	84.215
90	93.161	93.305
95	97.338	96.986

Tabla 4.3.5: Resultados de los intervalos de incertidumbre centrados en la media de la predicción de la SVR en los problemas de Sotavento y Red Eléctrica en el año 2015, habiendo sido calibrados con el año 2014.

predicción del proceso Gaussiano y en rojo el valor real. Podemos concluir que los intervalos son buenos; no obstante, están centrados en una predicción que es un poco inferior a la de las SVR.

4.3.2 Intervalos de incertidumbre en los GP centradas en la predicción de la SVR

Como hemos visto, en la secciones 4.1 y 4.2, los procesos Gaussianos en regresión obtienen peores resultados que la SVR; ahora que hemos visto que hay que realizar una calibración de los intervalos, nos preguntamos qué pasa si en los intervalos de la forma

$$[\text{máx}(\text{mín}(\hat{\mu} - \delta_s\hat{\sigma}, 1), 0), \text{máx}(\text{mín}(\hat{\mu} + \delta_s\hat{\sigma}, 1), 0)],$$

donde $\hat{\mu}$ es la predicción del proceso Gaussiano, cambiamos la predicción del proceso Gaussiano por la predicción del SVR Matérn 0.5; es decir, usamos los intervalos de la forma:

$$[\text{máx}(\text{mín}(\hat{y}_{SVR} - \delta_s\hat{\sigma}_{GP}, 1), 0), \text{máx}(\text{mín}(\hat{y}_{SVR} + \delta_s\hat{\sigma}_{GP}, 1), 0)].$$

Para ello, vamos a realizar los mismos pasos que en 4.3, realizando de nuevo la calibración sobre 2014, pero con la media centrada en la predicción de la SVR Matérn 0.5 que ha sido entrenada con 2013 y $\hat{\sigma}$ la que nos da el proceso Gaussiano Matérn 0.5, cuyos resultados se muestran en la Tabla 4.3.4.

En la Tabla 4.3.5 se muestran los resultados en test de los intervalos de incertidumbre centrados en la predicción del SVR Matérn 0.5 entrenada con los años 2013 y 2014. Podemos observar que el error en la incertidumbre está entre un 1 % y un 4 %, siendo muy parecidos a los que hemos obtenido con la predicción del proceso Gaussiano, salvo para el 80 y 90 que son un poco peores tanto en Sotavento como en Red Eléctrica. Por lo tanto, si nos interesa más la incertidumbre que la predicción, sería mejor utilizar el modelo GP calibrado puro.

4.4 Incertidumbre a partir de los residuos de una SVR

Como hemos visto en la subsección 3.2.2, si conocemos la función de media $m(x)$, la predicción de un proceso Gaussiano con dicha media es:

$$\begin{aligned}\hat{\mathbf{f}}|\hat{x}, X, \mathbf{y} \sim \mathcal{N}(m(\hat{x}) + k(\hat{x}, X)(K + \sigma_n^2)^{-1}(\mathbf{y} - m(X)) \\ k(\hat{X}, \hat{X}) - k(\hat{x}, X)(K + \sigma_n^2 I_N)^{-1}k(X, \hat{x});\end{aligned}$$

por lo tanto, si llamamos $\mathbf{e}_i = y(x_i) - m(x_i)$ se tiene que

$$\begin{aligned}\mathbb{E}_{\hat{\mathbf{f}}}(\hat{\mathbf{f}}|\hat{X}, X, \mathbf{y}) &= m(\hat{x}) + k(\hat{x}, X)(K + \sigma_n^2)^{-1}\mathbf{e} \\ &= m(\hat{x}) + e_{\mathrm{GP}}(\hat{x}),\end{aligned}$$

donde $e_{\mathrm{GP}}(\hat{x})$ quiere decir la predicción del proceso Gaussiano con media cero, con variable observada $\mathbf{e}$; por lo tanto, si tomamos como $m(x)$ la predicción de un algoritmo de Aprendizaje Automático, las e_i son los residuos y el proceso Gaussiano estaría modelando los residuos. Obviamente, al obtener una distribución en los residuos, obtenemos una distribución en la producción.

Para realizar los mismos pasos que en la seccion 4.3, necesitamos tener tres años con residuos. Para ello, una vez fijado los hiperparámetros de la SVR y el núcleo en el proceso Gaussiano, vamos a realizar el siguiente esquema, resumido en el Algoritmo 2:

1. Calcular los residuos en 2013: e_{2013}.
2. Entrenar un GP con variable predictiva e_{2013}: GP_{2013}.
3. Calcular los residuos en 2014: e_{2014}.
4. Predecir los residuos en 2014 con el GP_{2013}: $\hat{e}_{2014} = \hat{\mu}_{2014} \pm \hat{\sigma}_{2014}$.
5. Calcular $\delta(s)_{2014}$ de forma que el porcentaje de residuos que caen en el intervalo de confianza sea próximo a s:

$$\#\{e_{2014} \in [\hat{\mu}_{2014} - \delta(s)\hat{\sigma}_{2014}, \hat{\mu}_{2014} + \delta(s)\hat{\sigma}_{2014}]\}/N_{2014} \approx s.$$

6. Calcular los residuos en 2015: e_{2015}.
7. Entrenar un GP con variable predictiva e_{2014}: GP_{2014}.
8. Predecir los residuos en 2015 con el GP_{2014}: $\hat{e}_{2015} = \hat{\mu}_{2015} \pm \hat{\sigma}_{2015}$
9. Calcular en test el porcentaje de residuos que caen en el intervalo de confianza s:

$$\#\{e_{2015} \in [\hat{\mu}_{2015} - \delta(s)_{2014}\hat{\sigma}_{2015}, \hat{\mu}_{2015} + \delta(s)_{2014}\hat{\sigma}_{2015}]\}/N_{2015}.$$

Para obtener los hiperparámetros óptimos de la SVR con núcleo Gaussiano hemos realizado una validación cruzada con 6 folds sobre 2013, posteriormente hemos obtenido los residuos del año 2014 y 2015 entrenando con el año 2013 y 2014 respectivamente, con los hiperparámetros óptimos calculados anteriormente.

Como solo tenemos los datos del año 2013, 2014 y 2015 y necesitamos tres años de residuos. En 2013 hemos calculado los residuos usando la función `cross_val_predict` de Scikit-learn.

Algoritmo 2: Obtención de intervalos calibrados en los residuos

#Calcular los residuos en los dos primeros años
Calcular e_{2013}
Obtener GP_{2013}
Calcular e_{2014}
Predecir $\hat{e}_{2014} = \hat{\mu}_{2014} \pm \hat{\sigma}_{2014}$
Calcular el modelo calibrado
Calcular δ_s* de forma que: $\#\{e_{2014} \in [\hat{\mu}_{2014} - \delta_s^* \hat{\sigma}_{2014}, \hat{\mu}_{2014} + \delta_s^* \hat{\sigma}_{2014}]\}/N_{2014} \approx s$
Test
Calcular e_{2015}
Entrenar GP_{2014}
Predecir $\hat{e}_{2015} = \hat{\mu}_{2015} \pm \hat{\sigma}_{2015}$
Test: $\#\{e_{2015} \in [\hat{\mu}_{2015} - \delta_s^* \hat{\sigma}_{2015}, \hat{\mu}_{2015} + \delta_s^* \hat{\sigma}_{2015}]\}/N_{2015}$

Núcleo	Sotavento Log verosimilitud	Red Eléctrica Log verosimilitud
RBF	11109.356	**22230.564**
2 RBF	11111.521	22095.064
RationalQuadratic	11138.957	22230.562
Matérn 0.5	10753.276	19230.907
Matérn 1.5	11125.099	21819.545
Matérn 2.5	**11142.695**	22075.316

Tabla 4.4.1: Resultados obtenidos mediante un proceso Gaussiano en los residuos de la SVR en Sotavento y Red Eléctrica en validación (2014).

Confianza	Sotavento	Red Eléctrica
20	30.323	19.621
50	60.738	47.594
80	81.485	74.744
90	88.158	84.429
95	91.789	89.898

Tabla 4.4.2: Resultados de los intervalos de incertidumbre en los residuos de Sotavento y Red Eléctrica en el año 2014.

Para la elección del núcleo, hemos decidido escoger el núcleo con mayor verosimilitud entrenando con los residuos del 2013, cuyos resultados se muestran en la Tabla 4.4.1, eligiendo Matérn 2.5 para los residuos de Sotavento y RBF para los residuos de Red Eléctrica.

El MAE en 2015 del modelo $y_{SVR} + \hat{e}_{GP}$ es de **6.51** en Sotavento y **3.00** en Red Eléctrica, siendo estos resultados peores que los obtenidos en las secciones 4.1 y 4.2, si bien es cierto que aquí el modelo que estamos considerando solo tiene en cuenta el año anterior y en la secciones 4.1 y 4.2 el modelo se entrena con los dos años anteriores.

En la Tabla 4.4.2 se muestran los resultados del intervalo de incertidumbre sobre 2014; vemos que al igual que en la sección 4.3, los intervalos en Sotavento para confianza 20 y 50 son bastante malos y en Red Eléctrica los de 20 y 50 están mas o menos bien pero en el resto subestimamos la incertidumbre.

En la Tabla 4.4.3, se muestran los resultados en 2014 tras la calibración en 2014, donde podemos ver que estamos haciendo bien la calibración del δ.

En la Tabla 4.4.4, se muestran los resultados en 2015 tras la calibración en 2014. Podemos observar que el error en la incertidumbre está entre un 1 % y un 3 % en Sotavento y entre

Confianza	**Sotavento**			**Red Eléctrica**		
	Calibrado	δ	$1-\alpha$ (%)	**Calibrado**	δ	$1-\alpha$ (%)
20	20.065	0.163	12.938	19.752	0.257	20.280
50	50.145	0.491	37.687	50.274	0.723	53.012
80	80.167	1.227	78.004	79.926	1.446	85.191
90	89.731	1.762	92.199	89.671	1.950	94.886
95	94.958	2.380	98.269	95.199	2.492	98.730

Tabla 4.4.3: Resultados de los intervalos de incertidumbre en los residuos de la SVR en Sotavento y Red Eléctrica en el año 2014, habiendo sido calibrados con el año 2014, además se muestran los deltas obtenidos así como los $1-\alpha$ correspondientes.

Confianza	**Sotavento**	**Red Eléctrica**
20	21.492	21.098
50	52.828	54.396
80	83.924	85.037
90	93.230	93.984
95	97.740	98.034

Tabla 4.4.4: Resultados de los intervalos de incertidumbre en los residuos de la SVR en Sotavento y Red Eléctrica en el año 2015, habiendo sido calibrados con el año 2014.

un 1 % y un 5 % en Red Eléctrica.

Estos resultados son un poco peores que los obtenidos en la sección 4.3; por lo tanto, podemos concluir que la obtención de intervalos de incertidumbre modelizando los residuos de una SVR por un proceso Gaussiano no es lo más apropiado.

Capítulo 5

Conclusiones y trabajo futuro

En este trabajo, se ha visto brevemente la regresión lineal tradicional, la regresión lineal bayesiana, los métodos kernel, los perceptrones multicapa y las SVR y más en detalle los procesos Gaussianos, permitiéndonos comprender mejor a estos últimos comparándolos con los modelos anteriormente citados.

Hemos visto que los procesos Gaussianos de media cero y el Kernel Ridge Regresion son muy parecidos, pues la predicción es la misma si dejamos fijos los hiperparámetros de la función de núcleos; no obstante, como los procesos Gaussianos nos dan una distribución, no solo obtenemos un valor de predicción si no también una varianza. También, hemos visto brevemente los procesos Gaussianos con media distinta de cero, quedando para trabajo futuro una mayor descripción y experimentación con estos.

Hemos visto dos formas de calcular los hiperparámetros óptimos en los procesos Gaussianos: mediante la selección Bayesiana de modelos y mediante validación cruzada LOO, teniendo ambos coste lineal con respecto al número de hiperparámetros. Experimentalmente, hemos probado únicamente la selección Bayesiana de modelos, luego queda como trabajo futuro buscar otra librería donde se implemente el cálculo de hiperparámetros mediante validación cruzada LOO y comprobar si existe alguna diferencia significativa en los resultados.

Hemos analizado y desarrollado los procesos Gaussianos censurados, detallando y ampliando la teoría proporcionada en [Groot and Lucas, 2012], obteniendo una distribución para la variable censurada, lo cual nos permite entre otras cosas dar intervalos de incertidumbre sobre esta nueva variable, pero no hemos hecho un estudio experimental.

Experimentalmente, hemos probado en problemas de regresión en energía eólica los procesos Gaussianos con distintas funciones de núcleo y hemos observado que no son competitivos con las SVR; no obstante, hemos visto que el núcleo óptimo del proceso Gaussiano en una SVR nos puede permitir mejorar los resultados de ésta. De hecho, hemos observado que podemos usar los hiperparámetros del núcleo calculados por el proceso Gaussiano en la SVR, pues los resultados son muy parecidos, lo que supone una reducción de 3 a 2 dimensiones en la búsqueda en rejilla de la SVR, conllevando una reducción significativa de tiempo de ejecución, puesto que la hiperparametrización tiene coste lineal en los procesos Gaussianos. Queda como trabajo futuro probar núcleos más sofisticados en los procesos Gaussianos y ver si éstos los podemos llevar a la SVR, así como probar esta misma técnica en otros problemas de regresión.

Hemos realizado el cálculo directo de intervalos de incertidumbre en los mismos problemas y hemos visto que si no hacemos nada, generalmente estamos subestimando la incertidumbre.

Para arreglar este problema, hemos propuesto una calibración de los intervalos y hemos visto que los resultados del modelo calibrado son bastante buenos.

Hemos probado también a modelar los residuos de una SVR mediante un proceso Gaussiano para intentar mejorar los intervalos de incertidumbre, donde hemos concluido que no merece la pena pues los resultados de los intervalos y el modelo son peores que modelando la producción mediante un proceso Gaussiano.

Además de lo dicho anteriormente, para trabajo futuro queda probar los mismos experimentos en energía solar y probar e implementar los procesos Gaussianos censurados en los mismos problemas teniendo en cuenta que habrá que estudiar cómo hacer inferencia aproximada pues computacionalmente es muy costoso.

Glosario de acrónimos

- **GP**: Gaussian Process (Proceso Gaussiano)
- **NWP**: Numerical Weather Prediction (modelo numérico de predicción meteorológica)
- **KRR**: Kernel Ridge Regresion (Regresion Ridge como método de núcleo)
- **SVR**: Support Vector Regresion (Máquinas de vectores soporte para regresión)
- **MLP**: MultiLayer Perceptron (Perceptrón multicapa)
- **CV**: Cross-Validation (Validación cruzada)

Anexo A

Multiplicadores de Lagrange

En este anexo vamos a completar los desarrollos de la subsección 3.3.2 sobre como la verosimilitud Gaussiana puede actuar como navaja de Ockham. Para ello, tenemos que ver que bajo las restricciones $\sum_{i=1}^{N} d_i = N$, y $d_i \geq 0 \ \forall i$, que se cumple que

$$\prod_{i=1}^{N}(\sigma_f^2 d_i + \sigma_n^2) \leq (\sigma_f^2 + \sigma_n^2)^N.$$

Para ello definimos la siguiente función,

$$\log f(\mathbf{d}) = \sum_{i=1}^{N} \log(\sigma_f^2 d_i + \sigma_n^2),$$

donde $\mathbf{d} = (d_1, ..., d_N)$; para que las cuentas sean más fáciles vamos a buscar el máximo de la función $\log f(\mathbf{d})$, usando únicamente la restricción $\sum_{i=1}^{N} d_i = N$, para ello usamos los multiplicadores de Lagrange, denotando

$$g(\mathbf{d}) = \sum_{i=1}^{N} d_i - N,$$

se tiene que el máximo de $\log f(\mathbf{d})$ con la restricción anterior se alcanza resolviendo el siguiente sistema de ecuaciones:

$$\begin{cases} g(\mathbf{d}) = 0 \\ \nabla \log f(\mathbf{d}) - \lambda \nabla g(\mathbf{d}) = 0; \end{cases}$$

por lo tanto, tenemos que calcular las derivadas parciales de $\log f$ y g con respecto a d_i:

$$\frac{\partial \log f(\mathbf{d})}{\partial d_i} = \frac{\sigma_f^2}{\sigma_f^2 d_i + \sigma_n^2},$$
$$\frac{\partial g(\mathbf{d})}{\partial d_i} = 1;$$

por lo tanto, se sigue que

$$\begin{cases} \sum_{i=1}^{N} d_i - N = 0 \\ \dfrac{\sigma_f^2}{\sigma_f^2 d_i + \sigma_n^2} - \lambda = 0 \text{ para } 1 \leq i \leq N, \end{cases} \tag{A.0.1}$$

teniendo en cuenta que

$$\frac{\sigma_f^2}{\sigma_f^2 d_i + \sigma_n^2} - \lambda = \frac{\sigma_f^2 - \lambda\sigma_f^2 d_i - \lambda\sigma_n^2}{\sigma_f^2 d_i + \sigma_n^2},$$

y que

$$\frac{\sigma_f^2}{\sigma_f^2 d_i + \sigma_n^2} - \lambda = 0 \iff \sigma_f^2 - \lambda\sigma_f^2 d_i - \lambda\sigma_n^2 = 0,$$

se sigue que podemos despejar d_i:

$$d_i = \frac{\sigma_f^2 - \lambda\sigma_n^2}{\lambda\sigma_f^2}. \tag{A.0.2}$$

Sustituyendo en (A.0.1) se tiene que

$$\sum_{i=1}^{N} \frac{\sigma_f^2 - \lambda\sigma_n^2}{\lambda\sigma_f^2} = N,$$

de manera que podemos despejar λ:

$$\sigma_f^2 - \lambda\sigma_n^2 - \lambda\sigma_f^2 = 0 \implies \lambda = \frac{\sigma_f^2}{\sigma_f^2 + \sigma_n^2}.$$

Sustituyendo el valor de λ obtenido en (A.0.2) se tiene que

$$\begin{aligned} d_i &= \frac{\sigma_f^2 - \sigma_n^2 \frac{\sigma_f^2}{\sigma_f^2+\sigma_n^2}}{\sigma_f^2 \frac{\sigma_f^2}{\sigma_f^2+\sigma_n^2}} \\ &= \frac{\sigma_f^2(\sigma_f^2 + \sigma_n^2) - \sigma_n^2\sigma_f^2}{(\sigma_f^2)^2} \\ &= \frac{\sigma_f^2 + \sigma_n^2 - \sigma_n^2}{\sigma_f^2} = 1. \end{aligned}$$

Puesto que el logaritmo es una función monótona creciente, si $\mathbf{d}'$ maximiza $\log f(\cdot)$ también maximiza $f(\cdot)$; por otro lado, no hemos impuesto la restricción de que las d_i fueran no negativas; no obstante, puesto que la solución obtenida cumple estas restricciones hemos acabado. Para justificar que es un máximo, basta notar que el mínimo se alcanza en $d_1 = N$, $d_j = 0\ \forall j > 1$, pues hemos visto en (3.3.4) que se cumple la siguiente igualdad:

$$(\sigma_f^2 N + \sigma_n^2)(\sigma_n^2)^{N-1} \leq \prod_{i=1}^{N} (\sigma_f^2 d_i + \sigma_n^2).$$

Por lo tanto, concluimos que

$$\prod_{i=1}^{N} (\sigma_f^2 d_i + \sigma_n^2) \leq (\sigma_f^2 + \sigma_n^2)^N.$$

Bibliografía

[Abramowitz et al., 1965] Abramowitz, M., Stegun, I. A., and Miller, D. (1965). Handbook of Mathematical Functions With Formulas, Graphs and Mathematical Tables (National Bureau of Standards Applied Mathematics Series No. 55). *Journal of Applied Mechanics*.

[Bishop, 2006] Bishop, C. M. (2006). *Pattern Recognition And Machine Learning*.

[Catalina and Dorronsoro, 2017] Catalina, A. and Dorronsoro, J. R. (2017). NWP ensembles for wind energy uncertainty estimates. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 10691 LNAI:121–132.

[de la Pompa et al.,] de la Pompa, V., Catalina, A., and Dorronsoro, J. R. Gaussian Process Kernels for Support Vector Regression in Wind Energy Prediction. aceptado en la conferencia ideal 2018.

[Díaz-Vico et al., 2017] Díaz-Vico, D., Torres-Barrán, A., Omari, A., and Dorronsoro, J. R. (2017). Deep Neural Networks for Wind and Solar Energy Prediction. *Neural Processing Letters*.

[Fukunaga, 1990] Fukunaga, K. (1990). Statistical Pattern Stastical Pattern Recognition. *Pattern Recognition*.

[Greene, 2012] Greene, W. W. H. . (2012). *Econometric analysis*.

[Groot and Lucas, 2012] Groot, P. and Lucas, P. (2012). Gaussian Process Regression with Censored Data Using Expectation Propagation. *Sixth European Workshop on Probabilistic Graphical Models,*, pages 115–122.

[Hastie et al., 2001] Hastie, T., Tibshirani, R., and Friedman, J. (2001). The Elements of Statistical Learning. *The Mathematical Intelligencer*, 27(2):83–85.

[Kanagawa et al., 2018] Kanagawa, M., Hennig, P., Sejdinovic, D., and Sriperumbudur, B. K. (2018). Gaussian Processes and Kernel Methods: A Review on Connections and Equivalences. pages 1–64.

[Matérn, 1960] Matérn, B. (1960). Spatial Variation. *Reports of the Forest Research Institute of Sweden*.

[Minka, 2001] Minka, T. P. (2001). A family of algorithms for approximate bayesian inference. *Ph.D. Thesis*.

[Nickisch and Rasmussen, 2008] Nickisch, H. and Rasmussen, C. E. (2008). Approximations for Binary Gaussian Process Classification. *Journal of Machine Learning Research*, 9:2035–2078.

[Rasmussen and Williams, 2004] Rasmussen, C. E. and Williams, C. K. I. (2004). *Gaussian processes for machine learning.*, volume 14.