

El modelo general presentado en este trabajo, aunque pesado en notación, resultó ser muy efectivo al llevarlo a la práctica. A lo largo de este capítulo, se hará una exploración intuitiva y visual de sus capacidades. Se remarca que todas las gráficas presentadas, se generaron con el mismo paquete **bpwpm** que realiza la estimación de los parámetros  $\beta$ . Pues, los mismos objetos que las funciones arrojan, pueden ser utilizadas para hacer gráficas que evalúan el modelo y reflejan la intuición subyacente.

Para mostrar los resultados y las capacidades del modelo se presentan seis ejemplos. Los primeros cinco, corresponden a bases de datos simuladas en dos dimensiones, es decir, se tienen dos covariables  $\mathbf{X} \in \mathbb{R}^2$ , con diferentes patrones para la respuesta  $y$  tanto lineales como no lineales. El objetivo, es poder visualizar lo flexibles que son las fronteras de clasificación: la parte no lineal del modelo. Asimismo, al trabajar con bases de datos donde  $\mathcal{X}^2 \subseteq \mathbb{R}^2$ , se puede visualizar la función  $\eta(\mathbf{x})$  en tres dimensiones. El último ejemplo, corresponde a una base de datos reales de cáncer con múltiples variables. Al aumentar la dimensionalidad, el modelo ya no es visualmente intuitivo pero sigue obteniendo excelentes resultados.

A todos los modelos presentados a lo largo de este capítulo se les realizó un análisis de convergencia mirando las medias ergódicas de las cadenas. Sin embargo, únicamente se estudia a detalle para el ejemplo 0.3 de forma que no se saturara más la presentación.

## 0.1. Evaluación del modelo

No obstante, antes de poder presentar los ejemplos, se definen las dos métricas que se usarán para probar la efectividad (y precisión) de los modelos. Ya que se trabaja con modelos de clasificación binaria, una forma intuitiva de medir su efectividad es a través de un simple conteo de *errores y aciertos*. Este conteo, se presenta en una *matriz de confusión* que desglosa la clasificación en sus respectivas categorías binarias. Asimismo, se presenta la función *log-loss* ( $ll$ ) que no solo pondera la clasificación sino la *precisión* de esta, medida a través de la probabilidad  $p_i$  que se le asigna a cada observación  $i$ .

Las matrices de confusión (tabla 1), son un método descriptivo con base en las tablas de contingencia que calcula la frecuencia de los aciertos y errores separando por grupos. Donde  $\hat{y}$  es la variable predicha de la respuesta  $y$  y  $\#$  el símbolo que denota *número de*. Asimismo, se define la precisión del modelo como:

$$\text{precisión} = \frac{\text{Número de clasificacioens correctas}}{\text{Número total de observaciones}}$$

	$\hat{y} = 0$	$\hat{y} = 1$	
$y = 0$	$\#0\text{'s } \checkmark$	$\#0\text{'s clasificados como 1}$	$\#$ de observaciones 0
$y = 1$	$\#1\text{'s clasificados como 0}$	$\#1\text{'s } \checkmark$	$\#$ de observaciones 1
	$\#$ de 0's estimados	$\#$ de 1's estimados	Total de obs. = $n$

Tabla 1: Matriz de confusión

Sin embargo, la matriz de confusión resulta deficiente para comprar modelos completamente diferentes que resultan en la misma clasificación, por ello, se define una métrica más formal. Sea  $\mathbf{y} = (y_1, \dots, y_n)^t$  el vector de respuestas observadas y  $\hat{\mathbf{p}} = (\hat{p}_1, \dots, \hat{p}_n)^t$  el vector de probabilidades ajustadas, donde  $\hat{p}_i = \hat{P}_{\text{modelo}}(y_i = 1 | \mathbf{x}_i)$  es la probabilidad estimada por el modelo de que la observación  $y_i$  sea igual a uno. Asimismo, se puede definir un vector de respuestas ajustadas  $\hat{\mathbf{y}} = (\hat{y}_1, \dots, \hat{y}_n)^t$ , haciendo la predicción en el corte  $\hat{y}_i = 1 \iff \hat{p}_i > 0.5$ .<sup>1</sup>

**Definición 0.1.** La función *log-loss*  $ll : \{0, 1\}^n \times [0, 1]^n \rightarrow \mathbb{R}^+$ :

$$ll(\mathbf{y}, \hat{\mathbf{p}}) = - \sum_{i=1}^n [y_i \ln(\hat{p}_i) + (1 - y_i) \ln(1 - \hat{p}_i)]. \quad (1)$$

La ventaja de usar la función  $ll$ , es que resulta en una métrica que, no sólo mide que tan buena es la clasificación binaria, sino, que toma en cuenta la precisión de la predicción. Esto se debe a que la función es convexa y se penaliza cuando las probabilidades ajustadas están muy lejos de la real. Asimismo, si la predicción fue incorrecta pero la probabilidad fue cercana a 0.5 no se penaliza tanto. Idealmente  $ll = 0$  si se da una clasificación perfecta y conforme crezca, el modelo es peor. En la práctica y bajo un enfoque frecuentista, la función  $ll$  puede ser vista como una función de costos y más recientemente se ha utilizado para entrenar y comparar modelos de clasificación como lo son las redes neuronales ([nielsen2015neural](#)).

1. Este corte, es resultado de la simetría en cero de la función de acumulación normal estándar  $\Phi$ , derivado de la ecuación (??).

## 0.2. Ejemplo 1 - las capacidades del modelo bpwpm

El primer ejemplo que se analizará, busca ejemplificar los componentes del modelo general y sus capacidades. Para ello, se simularon un total de  $n = 350$  observaciones separadas en dos grupos, cada uno con tamaños  $n_0 = 200$  y  $n_1 = 150$  respectivamente ( $n = n_0 + n_1$ ). Los datos se muestrearon de las distribuciones normales bivariadas:

$$\begin{aligned} \text{Grupo 0: } & \mathbf{x}_i \sim N_2 \left( \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \middle| \mu_0 = \begin{pmatrix} 2 \\ 2 \end{pmatrix}, \Sigma_0 = \begin{pmatrix} 0.25 & 0.35 \\ 0.35 & 1 \end{pmatrix} \right) \\ & i=1,\dots,200 \\ \text{Grupo 1: } & \mathbf{x}_i \sim N_2 \left( \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \middle| \mu_1 = \begin{pmatrix} 4 \\ 4 \end{pmatrix}, \Sigma_1 = \begin{pmatrix} 1 & -0.24 \\ -0.24 & 0.64 \end{pmatrix} \right) \\ & i=201,\dots,350 \end{aligned}$$

Las medias  $\mu_j$   $j = \{0, 1\}$  se toman relativamente alejadas y las covarianzas corresponden a las correlaciones  $\rho_0 = 0.7$  y  $\rho_1 = 0.3$  respectivamente. Estos parámetros de simulación se escogen a través de un proceso empírico resultando en una estructura simple donde los grupos están claramente separados y hay poco traslape. Asimismo, el espacio de covariables queda definido en:  $\mathcal{X}^2 \approx [0.3, 7.5] \times [-0.5, 5.9]$ . Se codifica el grupo 0,  $y = 0$ , de color rojo y el grupo 1,  $y = 1$ , de color azul.<sup>2</sup> La base de datos final se presenta en la figura 1.

2. Se recomienda visualizar la versión digital de este trabajo donde se aprecian con más claridad los colores. Disponible en . . .

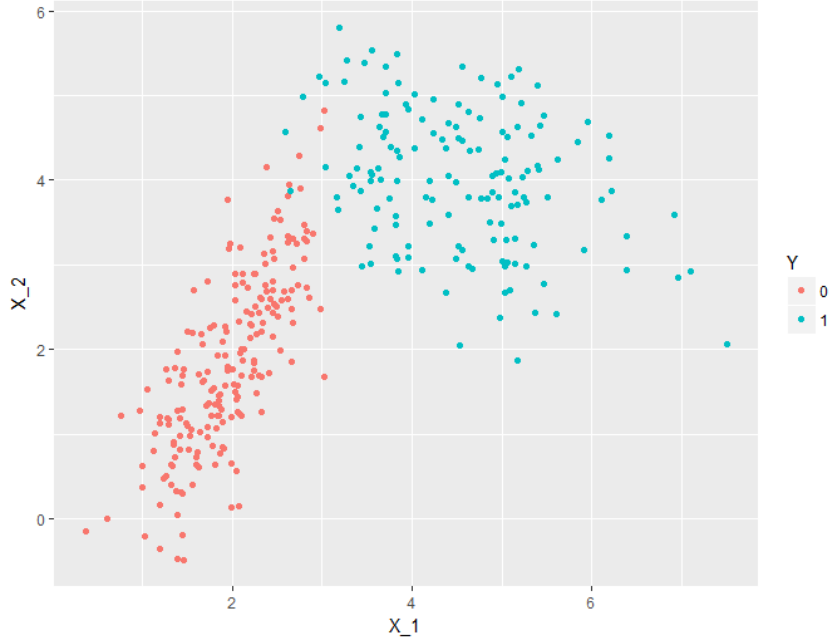


Figura 1: Ejemplo 1

### Tres realizaciones del modelo

El objetivo principal de esta simple base de datos es ejemplificar el tipo de fronteras alcanzables, mostrando una clara separación entre los dos grupos sin sobre-ajustar. Para ello, se corren tres realizaciones del modelo. Para la primera, se escoge una frontera lineal con un solo nodo. La segunda realización, consta de parábolas continuas más no suaves sobre cuatro nodos. Finalmente la tercera realización consta de

splines cúbicos<sup>3</sup> en 3 nodos. En la tabla 2 se resume lo anterior.<sup>4</sup>

Parámetro	Realización 1	Realización 2	Realización 3
$M$	2	3	4
$J$	2	5	3
$K$	1	1	3
$N^*$	2	10	5
$\lambda$	5	21	11

Tabla 2: Ejemplo 1 - tres realizaciones del modelo

Para las tres realizaciones se simularon  $N_{\text{sim}} = 15,000$  valores de  $\beta$  y se opta por no usar periodo de *burn-in* ni suavizamiento para las cadenas, es decir:  $k^* = 0$  y  $k_{\text{thin}} = 0$ . Esto para hacer a los modelos más comparables entre si y dar un vistazo rápido al modelo. La única modificación que se realiza entre las tres realizaciones es que, para la tercera, se estandarizan<sup>5</sup> los datos  $\mathbf{X}$ . Al usar polinomios de orden mayor, en este caso polinomios de tercer grado, el algoritmo puede caer en problemas numéricos pues  $\hat{\eta}$  puede crecer muy rápido fuera de  $\mathcal{X}^d$ ; se expande sobre este tema en el capítulo ??.

En las figuras 2, 3 y 4 se presentan imágenes que ejemplifican las tres realizaciones del modelo respectivamente. En las imágenes 2a, 3a y 4a se visualizan las diferentes

3. Polinomios por partes cúbicos suaves hasta la segunda derivada

4. Se recuerda que  $M - 1$  corresponde al grado de los polinomios,  $J - 1$  es el número de nodos,  $K$  el parámetro que controla la suavidad,  $N^*$  el número de funciones base (por expansión de cada covariable) y  $\lambda$  el número total de parámetros.

5. Se resta la media y se divide entre la desviación estándar muestral de cada covariable.

tipos de fronteras que el modelo logra estimar. Con estas fronteras, se nota claramente como es determinante la elección de  $M$ ,  $J$  y  $K$  en sus formas. El modelo logra estimar tanto fronteras relativamente rígidas como en 2a como fronteras más suaves en las imágenes subsecuentes. Asimismo, para cada realización, se tiene la representación en 3D de cada función  $\hat{\eta}$  que preserva la suavidad (o no) de sus componentes. Asimismo, rescatando las ideas de los GAM, se puede colapsar cada expansión de polinomios por partes en sus correspondientes  $\hat{f}_j$  y visualizarla como la transformación no lineal de cada covariables. Por ejemplo, en la imagen 2c se observa que  $\hat{f}_1(x_1)$  está compuesta por rectas que se conectan en el nodo, mientras que 4d presenta  $\hat{f}_2(x_2)$ , un polinomio cúbico suave hasta la segunda derivada.

	$\hat{y} = 0$	$\hat{y} = 1$	
$y = 0$	198	2	200
$y = 1$	2	148	150
	200	150	350

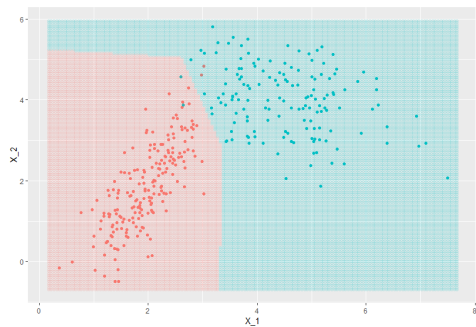
(a) Matriz de confusión

Realización	$ll$
1	0.04088
2	0.03464
3	0.03498

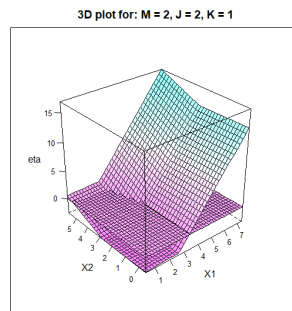
(b)  $\log$ -loss

Tabla 3: Ejemplo 1 - resultados

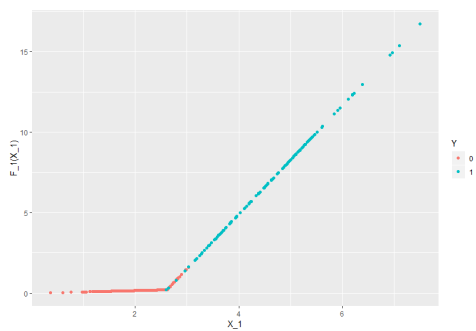
Al estar tratando con una base de datos tan sencilla, no es el enfoque comparar estas realizaciones entre si pues las tres logran exactamente la misma clasificación desglosada en la tabla 3a. Al compartir la matriz de confusión, por ende, las realizaciones también comparten una precisión de 98.9 %. De la matriz y las imágenes, se observa que se clasifican de forma incorrecta solo cuatro observaciones. Sería inverosímil tratar de alcanzar una precisión del 100 % pues implicaría sobre-ajustar el modelo. Para estas cuatro observaciones, no se tiene la suficiente evidencia como para clasificarlas en su categoría contraria. Sin embargo, los modelos se pueden comparar más



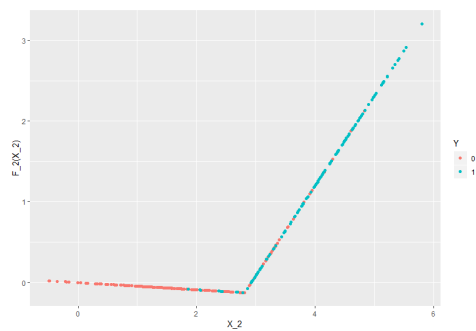
(a) Frontera de predicción



(b) Representación 3D de  $\hat{\eta}$



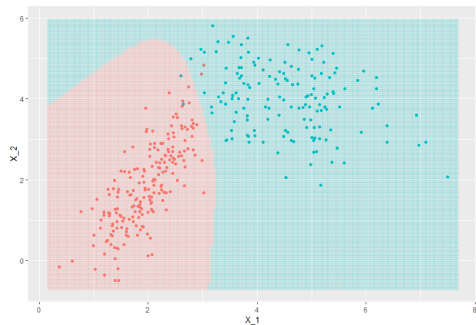
(c)  $\hat{f}_1(x_1)$



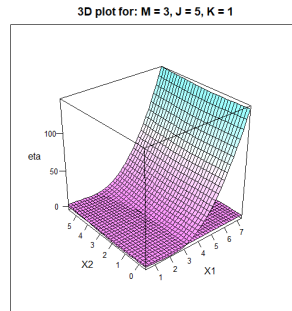
(d)  $\hat{f}_2(x_2)$

Figura 2: Realización 1 - fronteras lineales con un nodo ( $M = 2$ ,  $J = 2$  y  $K = 1$ )

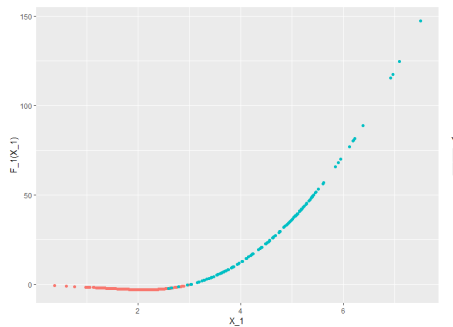




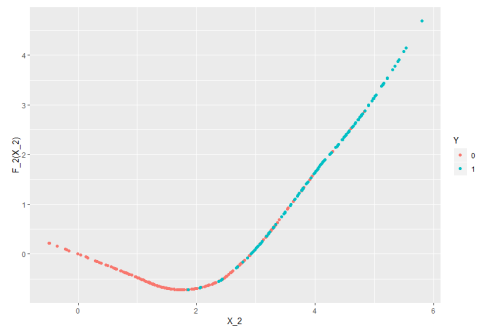
(a) Frontera de predicción



(b) Representación 3D de  $\hat{\eta}$



(c)  $\hat{f}_1(x_1)$

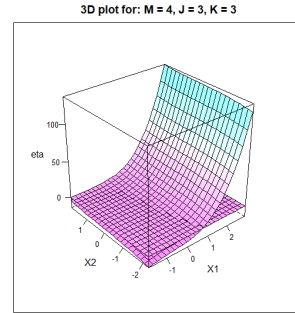


(d)  $\hat{f}_2(x_2)$

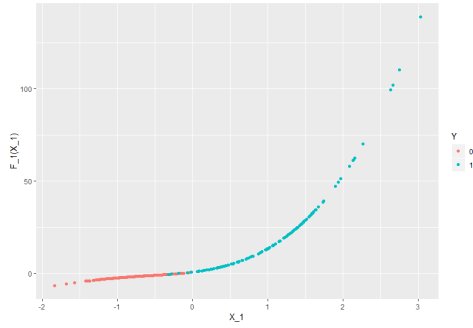
Figura 3: Realización 2 - parábolas continuas mas no suaves ( $M = 3$ ,  $J = 5$  y  $K = 1$ )



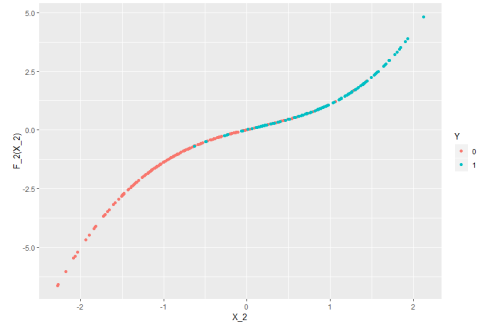
(a) Frontera de predicción



(b) Representación 3D de  $\hat{\eta}$



(c)  $\hat{f}_1(x_1)$



(d)  $\hat{f}_2(x_2)$

Figura 4: Realización 3 - *splines* cúbicos ( $M = 4$ ,  $J = 3$  y  $K = 3$ )

a fondo por medio de la métrica  $ll$  presentada en la tabla 3b. Aunque muy similares, la definición de la métrica indica que la realización dos es la mejor por un pequeño margen pues es la más cercana a cero. Claro está, bajo esta comparación no se toma en cuenta el número de parámetros y la complejidad del modelo en si.

Cabe mencionar que para esta sencilla base de datos particular, usar un modelo complejo como el *bpwpm* no es del todo necesario pues la base podría ser clasificada con la misma precisión por un modelo que use un predictor lineal en las covariables. Sin embargo, se usa el modelo para ejemplificar las fronteras flexibles. Asimismo, presentar las formas funcionales que toman las funciones  $f_j$  tampoco aportaría mucho pues están compuestas de muchos términos aditivos que no vale la pena presentar. De igual forma, dado que este es un ejemplo introductorio la estimación de los parámetros, se realizó *dentro de la muestra* (*in-sample*) esto quiere decir, que el modelo se entrena con las mismas observaciones contra las que se busca predecir.<sup>6</sup>

### 0.3. Ejemplo 2 - comparación contra un GLM

Aprovechando la familiaridad de la base de datos anterior, se decidió modificarla para que existieran dos regiones de clasificación separadas. Se tomaron aproximadamente trece puntos, más allá de  $x_1 \approx 5.5$  y se cambia su clasificación. En la imagen 6a se presenta la base de datos modificada.

6. El efecto que esto puede tener es que se sobre-ajuste o se hagan predicciones demasiado acertadas. De cualquier forma el paquete es suficientemente flexible como para permitir un entrenamiento previo y una predicción *fuera de muestra* (*out-of-sample*).

Con afán de comparar las predicciones del modelo *bpwpm* presentado en este trabajo, primeramente se corre un modelo probit lineal sobre esta base de datos. Es decir, se estiman<sup>7</sup> los parámetros  $\beta = (\beta_0, \beta_1, \beta_2)^t$  del modelo:

$$p_i = P(y_i = 1) = \mathbb{E}[y|\mathbf{x}_i] = \Phi(\eta(\mathbf{x}_i)) \Rightarrow$$

$$\Phi^{-1}(p_i) = \beta_0 + \beta_1 x_{i,1} + \beta_2 x_{i,2} \quad \forall i = 1, \dots, n \quad (2)$$

De donde se obtienen los resultados presentados en la tabla 4 y la figura 5.

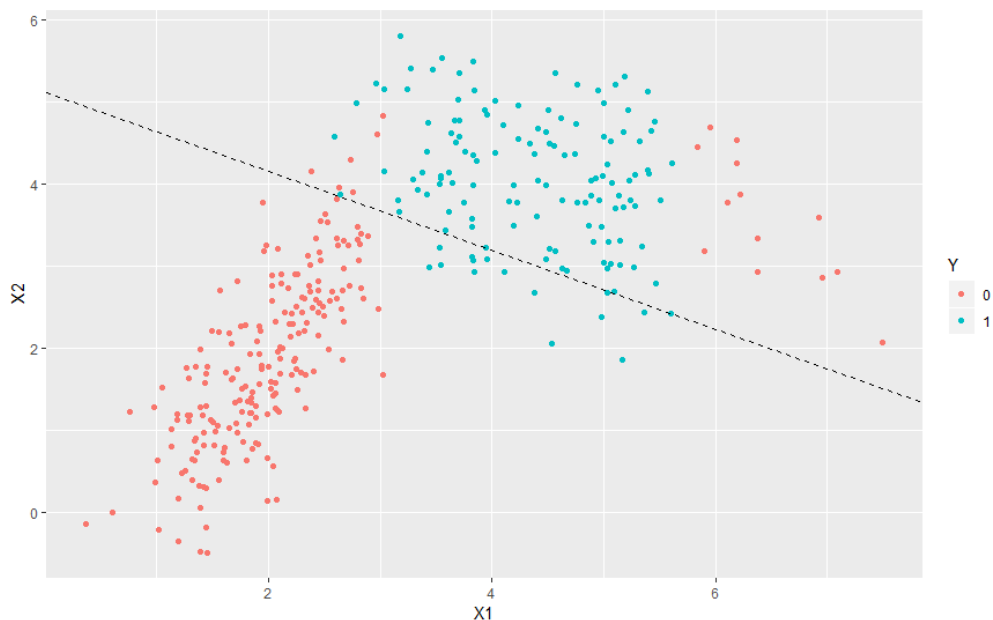


Figura 5: frontera de predicción para modelo probit lineal

7. La estimación se realiza bajo el paradigma frecuentista usando el método de mínimos cuadrados a través de la función `glm(..., family = binomial(link = 'probit'))` en R.

Parámetro	Estimado	Info. predicción	
$\hat{\beta}_0$	-4.67	Est. Puntual	No aplica
$\hat{\beta}_1$	0.45	Precisión	90 %
$\hat{\beta}_2$	0.91	<i>log-loss</i>	0.28072

	$\hat{y} = 0$	$\hat{y} = 1$	
$y = 0$	194	19	213
$y = 1$	16	121	137
	210	140	350

Tabla 4: Resultados para modelo probit lineal

Por ende, el modelo resultante es:

$$\phi^{-1}(\hat{p}_i) = \hat{\eta}(\mathbf{x}) = -4.67 + 0.45x_{i,1} + 0.91x_{i,2}, \quad (3)$$

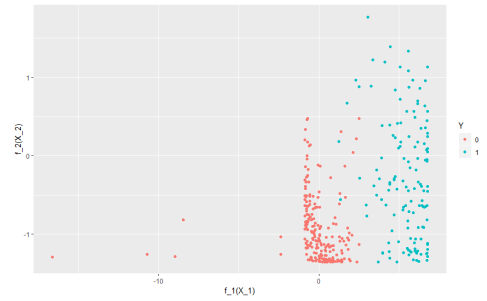
de donde se puede obtener explícitamente la ecuación de la frontera de predicción igualando (3) a 0.5:

$$\begin{aligned}
\phi(\hat{\eta}(\mathbf{x}_i)) &\equiv 0.5 && \Longleftrightarrow \\
\hat{\eta}(\mathbf{x}_i) &= 0 && \Longleftrightarrow \\
0.45x_{i,1} + 0.91x_{i,2} &= 4.67 && (4)
\end{aligned}$$

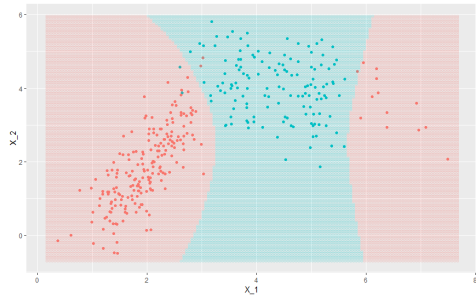
Ahora, se corre el modelo *bpwpm* con los parámetros escogidos resumidos en la tabla 5. Asimismo se presentan los resultados en la tabla 6 e imágenes en la figura 6.



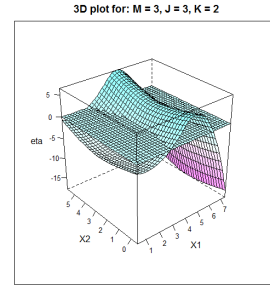
(a) Base del ejemplo 1 modificada



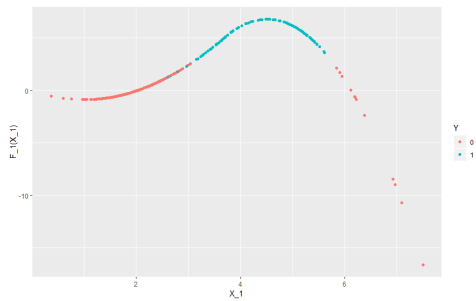
(b) Transformación no lineal



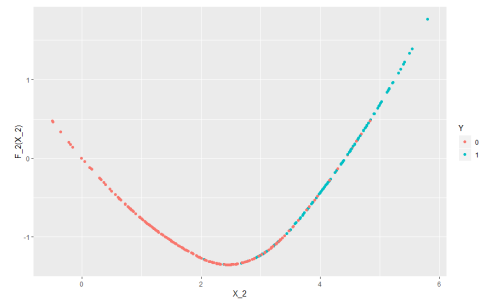
(c) Frontera de predicción



(d) Representación 3D de  $\hat{\eta}$



(e)  $\hat{f}_1(x_1)$



(f)  $\hat{f}_2(x_2)$

Figura 6: ejemplo 2 - regiones disjuntas de clasificación ( $M = 3$ ,  $J = 3$  y  $K = 2$ )

Parámetros		Parámetro Sim.
$M = 3$	$N^* = 4$	$N_{\text{sim}} = 10,000$
$J = 3$	$\lambda = 9$	$k^* = 7,500$
$K = 2$	$n = 350$	$k_{\text{thin}} = 0$

Tabla 5: ejemplo 3 - región parabólica

$$\begin{aligned}
\phi^{-1}(\hat{p}_i) &= \hat{\eta}(\mathbf{x}) = \hat{f}_0 + \hat{f}_1(x_{i,1}) + \hat{f}_2(x_{i,2}) \\
&= \underbrace{\hat{f}_0}_{\hat{\beta}_0} \\
&\quad + \underbrace{\hat{\beta}_1 x_{i,1} + \hat{\beta}_2 x_{i,1}^2 + \hat{\beta}_3 (x_{i,1} - \hat{\tau}_{1,1})_+^2 + \hat{\beta}_4 (x_{i,1} - \hat{\tau}_{1,2})_+^2}_{\hat{f}_1(x_{i,1})} \\
&\quad + \underbrace{\hat{\beta}_5 x_{i,2} + \hat{\beta}_6 x_{i,2}^2 + \hat{\beta}_7 (x_{i,2} - \hat{\tau}_{2,1})_+^2 + \hat{\beta}_8 (x_{i,2} - \hat{\tau}_{2,2})_+^2}_{\hat{f}_2(x_{i,2})}
\end{aligned} \tag{5}$$

- Hacer comparación, haciendo cross referencing, concluir que me la pelan - Puntualizar la transformación a FSpace

### 0.3.1. Análisis de convergencia

Este ejemplo es interesante pues, como se ve en la imagen ??, la *sabana* que antes era creciente a medida que  $x_1$  crecía, ahora se vuelve a curvar, volviéndose negativa otra vez y clasificando bien la segunda sección roja. Una vez más, se tienen esos

Info. predicción	
Est. Puntual	Media posterior
Precisión	98.6 %
<i>log-loss</i>	0.04505

$\beta$	Valor	
$\hat{\beta}_0$	-2.03	} $\hat{f}_0$
$\hat{\beta}_1$	-1.74	
$\hat{\beta}_2$	0.90	} $\hat{f}_1(x_{i,1})$
$\hat{\beta}_3$	-0.07	
$\hat{\beta}_4$	-3.68	
$\hat{\beta}_5$	-1.01	
$\hat{\beta}_6$	0.13	} $\hat{f}_1(x_{i,1})$
$\hat{\beta}_7$	0.31	
$\hat{\beta}_8$	-0.25	

	$\hat{y} = 0$	$\hat{y} = 1$	
$y = 0$	210	2	200
$y = 1$	2	135	137
	212	138	350

$\mathcal{P}$	Valor	
$\hat{\tau}_{1,1}$	2.07	} Nodos
$\hat{\tau}_{1,2}$	3.69	
$\hat{\tau}_{2,1}$	2.00	
$\hat{\tau}_{2,2}$	3.52	

Tabla 6: ejemplo 3 - resultados



Métrica	$\hat{\beta}_0$	$\hat{\beta}_1$	$\hat{\beta}_2$	$\hat{\beta}_3$	$\hat{\beta}_4$
Mínimo	-6.79	-6.63	-0.39	-2.11	-27.40
Primer Cuartíl	-2.54	-2.22	0.66	-0.48	-3.72
Media	-2.03	-1.73	0.90	-0.07	-3.68
Mediana	-1.98	-1.69	0.85	-0.09	-3.28
Tercer Cuartíl	-1.44	-1.17	1.05	0.27	-2.86
Máximo	0.85	1.56	4.21	10.38	-1.07
Desviación Estandar	0.89	0.87	0.45	0.72	2.61

Métrica	$\hat{\beta}_5$	$\hat{\beta}_6$	$\hat{\beta}_7$	$\hat{\beta}_8$
Mínimo	-5.59	-1.64	-1.89	-2.47
Primer Cuartíl	-1.49	-0.04	-0.05	-0.69
Media	-1.00	0.13	0.31	-0.21
Mediana	-0.97	0.13	0.27	-0.27
Tercer Cuartíl	-0.44	0.31	0.63	0.13
Máximo	2.44	1.47	6.67	11.0
Desviación Estandar	0.87	0.28	0.60	0.94

Tabla 7: resúmenes numéricos para las cadenas de  $\beta$

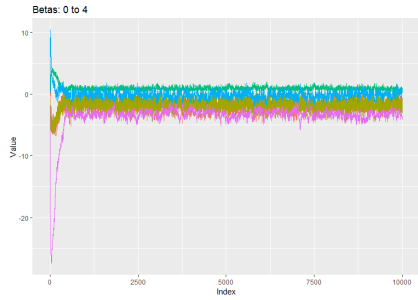
pocos puntos que no quedan bien clasificados, incluyendo uno nuevo cerca de las coordenadas cartesianas (5.8, 2.3). Para estos datos, se debe usar un nodo adicional cerca de la segunda región, ya que la curvatura, deriva de él. El parámetro  $K$  en este ejemplo no es muy relevante como se ve en la imagen ??, nuevamente porque  $\hat{f}_1(x_1)$  pareciera ser suficientemente suave sin tener que restringir el modelo. Finalmente, se enfatiza que vuelve a suceder lo mismo que pasó con el ejemplo 6, donde la información se podía resumir únicamente con las primeras dos dimensiones.

## 0.4. Ejemplos 3 a 5 - otros resultados interesantes

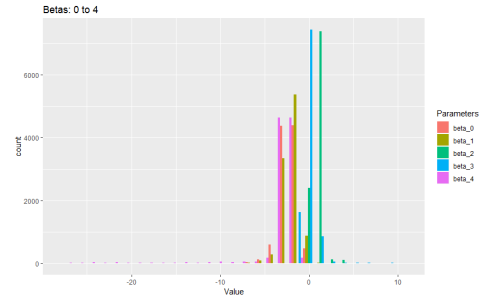
Los ejemplos presentados a continuación, son más expositivos que analíticos, es decir, se enfatizan los resultados más que los detalles matemáticos como se hizo en la sección anterior. Estos ejemplos y bases de datos simuladas, buscan sobre todo, poner a prueba las capacidades no lineales del modelo y estresar las interacciones entre las dimensiones. Al estar tratando con regiones de clasificación más complejas, la predicción sería imposible para un GLM.

### Ejemplo 3 - región parabólicos

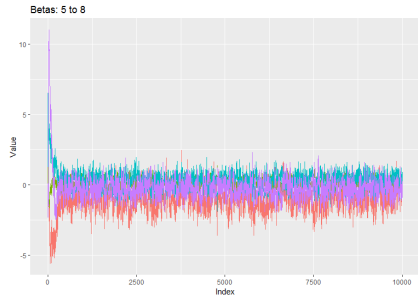
Para este ejemplo, se generaron  $n = 400$  datos en  $\mathbb{R}^2$  usando coordenadas polares al tomar ángulos con un rango entre  $[-1, 1]$ . Posteriormente se tomaron diferentes radios para diferenciar cada grupo y finalmente se les sumó ruido blanco a los puntos



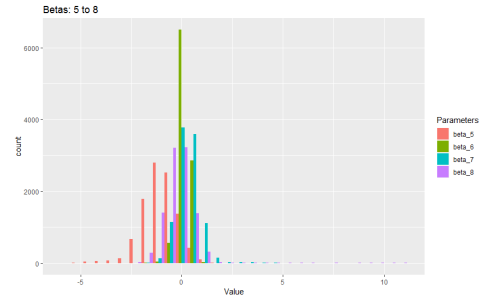
(a) Trazas de  $\hat{\beta}_0$  a  $\hat{\beta}_4$



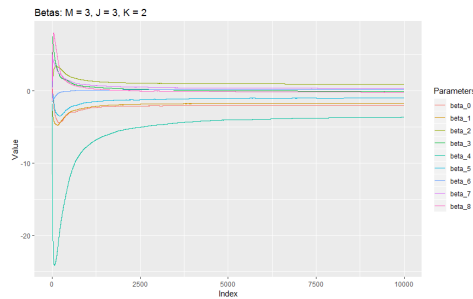
(b) Histogramas de  $\hat{\beta}_0$  a  $\hat{\beta}_4$



(c) Trazas de  $\hat{\beta}_5$  a  $\hat{\beta}_8$



(d) Histogramas de  $\hat{\beta}_5$  a  $\hat{\beta}_8$



(e) Media ergódica

Figura 7: ejemplo 2 - análisis de convergencia

para que existiera una región de confusión. Las simulación derivó en un patrón de datos cuya frontera es curva, inclusive parabólica. Dadas las características de los datos, se piensa que usar polinomios por partes parabólicos y suaves ( $M = 3$  y  $K = 2$ ) es una buena opción para modelarlos. Los parámetros escogidos para la realización final del modelo, se presentan en la tabla 8. Asimismo, los resultados e imágenes se presentan en la tabla 9 y la figura 8 respectivamente.

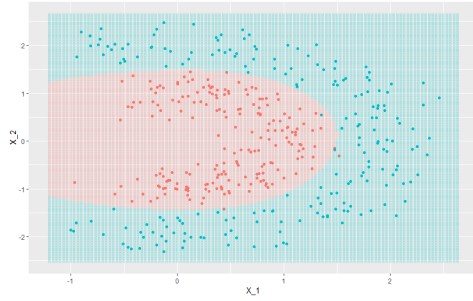
Parámetros		Parámetro Sim.
$M = 3$	$N^* = 5$	$N_{\text{sim}} = 10,000$
$J = 4$	$\lambda = 11$	$k^* = 2,500$
$K = 2$	$n = 400$	$k_{\text{thin}} = 0$

Tabla 8: Ejemplo 3 - región parabólica

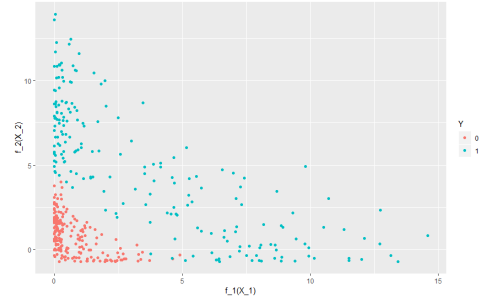
Info. predicción				
		$\hat{y} = 0$	$\hat{y} = 1$	
Est. Puntual	Media posterior	198	2	200
Precisión	99.2 %	1	199	200
<i>log-loss</i>	0.04352	199	201	400

Tabla 9: Ejemplo 3 - resultados

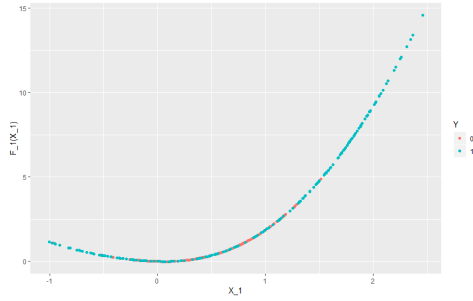
Esta es una realización particularmente interesante pues con un total  $\lambda = 11$  parámetros se logra una precisión alta además de obtener convergencia relativamente rápido ( $N_{\text{sim}} = 10,000$  y  $k^* = 2,500$ ). Analizando el modelo de forma gráfica, se observa claramente que la segunda transformación  $\hat{f}_2(x_2)$  (imagen 8d) captura la parte parabólica. A la vez, la primera transformación  $\hat{f}_1(x_1)$  (imagen 8c) le da poco peso a la región donde hay confusión entre los los grupos pero posteriormente crece en donde hay certidumbre. Asimismo, se presenta el espacio de la transformación no



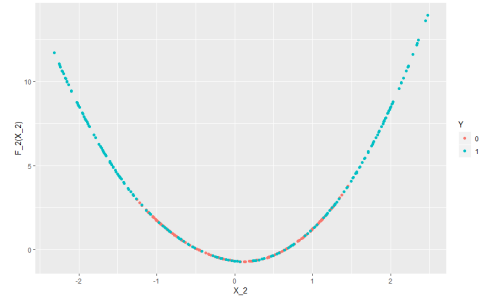
(a) Frontera



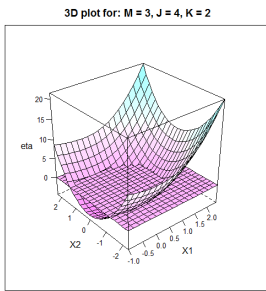
(b) Transformación no lineal



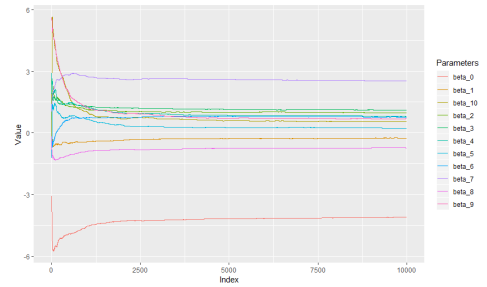
(c)  $\hat{f}_1(x_1)$



(d)  $\hat{f}_2(x_2)$



(e) Representación 3D de  $\hat{\eta}$



(f) Medias ergódicas

Figura 8: ejemplo 3 - parábolas suaves ( $M = 3$ ,  $J = 4$  y  $K = 2$ )

lineal en la imagen 8b en donde se observa que el grupo rojo cero, se concentra en la esquina inferior izquierda, representando la posible separación lineal en este espacio transformado.

## Ejemplo 4 - región ovalada

Para esta base de datos en particular se busca replicar algo similar a la imagen del capítulo introductorio ?? de la página ?. Avanzando con las regiones no lineales, se obtuvo una base de datos pequeña del curso en línea de ML de **andrew2018ml** que presenta una frontera de clasificación ovalada.<sup>8</sup> Esta base de datos se usa para entrenar modelos saturados logit con regularización, **hastie2008elements**, logrando predecir fronteras curvas con modelos tradicionalmente lineales al incluir las interacciones entre las covariables. Por lo tanto, se decidió probarlo también con el modelo presentado para contrastar.

El modelo una vez más, fue ajustado usando parábolas suaves las cuales resultaron ser excelentes herramientas. Los parámetros escogidos para la realización final del modelo, se presenta en la tabla 10 con resultados e imágenes en la tabla ?? y figura 9 respectivamente.

Para esta realización del modelo, se buscó estresar su flexibilidad al incluir el menor número de términos posibles usando un solo nodo ( $J = 2$  y  $\lambda = 7$ ) y cadenas cortas ( $N_{\text{sim}} = 2,000$ ). Aunque una precisión de 78.8% no resulte tan atractiva, es

8. Este curso, se ofrece de forma gratuita en la página de Coursera.

Parámetros		Parámetro Sim.
$M = 3$	$N^* = 3$	$N_{\text{sim}} = 2,000$
$J = 2$	$\lambda = 7$	$k^* = 500$
$K = 2$	$n = 118$	$k_{\text{thin}} = 0$

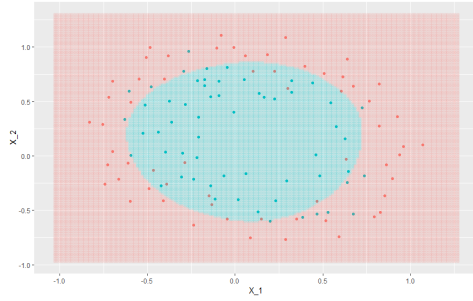
Tabla 10: Ejemplo 4 - región ovalada

Info. predicción				
		$\hat{y} = 0$	$\hat{y} = 1$	
Est. Puntual	Media posterior	48	12	60
Precisión	78.8 %	13	45	58
<i>log-loss</i>	0.4714	61	57	118

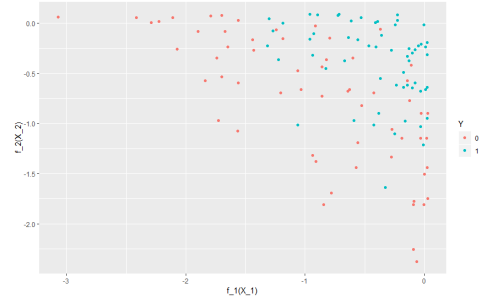
Tabla 11: Ejemplo 4 - resultados

la precisión que se presenta en el curso en línea y permanece constante aún si se aumenta  $\lambda$ . La métrica  $ll$  mejora (marginalmente) sobre la presentada en el curso, sin embargo, se logra una reducción significativa en el número de parámetros pues el modelo saurado de **andrew2018ml** inicia con 28 parámetros.<sup>9</sup>. Asimismo, como se observa en la figura 9f las cadenas convergen rápidamente. Todo el poder del modelo, recae en la forma funcional de las funciones  $\hat{f}_j$  al poder estimar regiones irregularmente curvas, con muy pocas observaciones y parámetros.

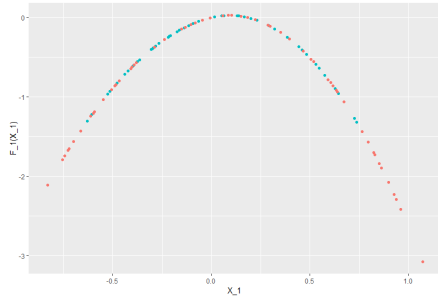
9. Dada la regularización, muchos de estos se desvanecían.



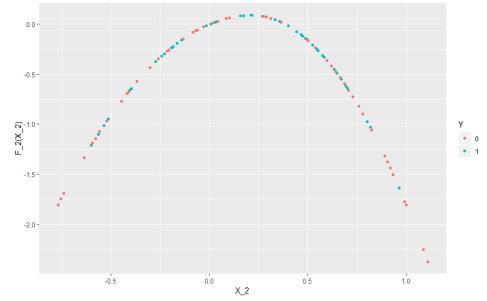
(a) Frontera



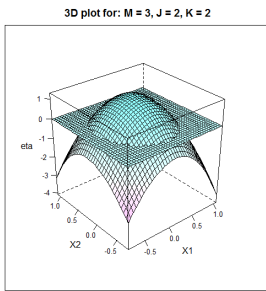
(b) Transformación no lineal



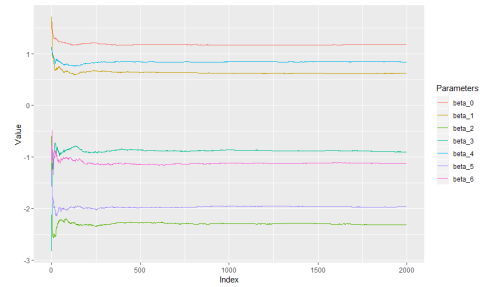
(c)  $\hat{f}_1(x_1)$



(d)  $\hat{f}_2(x_2)$



(e) Representación 3D de  $\hat{\eta}$



(f) Medias ergódicas

Figura 9: ejemplo 4 - parábolas suaves en un nodos ( $M = 3$ ,  $J = 2$  y  $K = 2$ )



### Ejemplo 5 - *yin-yang*, limitaciones del modelo

Para finalizar con las bases de datos simulados, el modelo se llevó al límite de sus capacidades sobre un patrón de puntos, intuitivo al ojo humano, pero difícil de identificar por un algoritmo.<sup>10</sup> Los datos tratan de simular un *yin-yang* que se puede observar en la figura 10a. La simple simulación de la base de datos representó un reto donde se conjuntaron varias áreas de la matemática aplicada. En el software GeoGebra, se generó el diagrama presentado en la figura 10b que consiste de las siguientes desigualdades cartesianas:

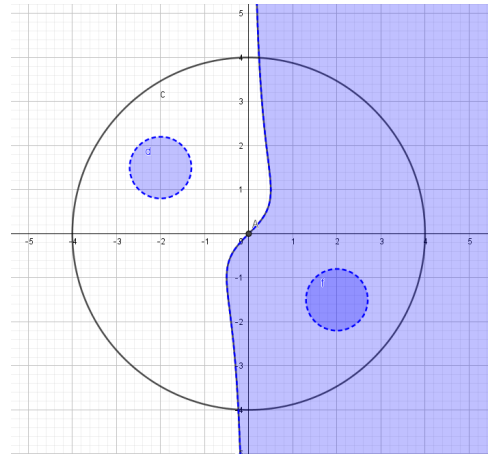
$$\begin{aligned}x^2 + y^2 &< 16, \\(x + 2)^2 + (y - 1.5)^2 &< 0.49, \\(x - 1.5)^2 + (y + 2)^2 &< 0.49, \\x &< \frac{y}{(1 + y^2)}.\end{aligned}$$

Una vez dibujadas las ecuaciones, se generó una base de datos de aproximadamente  $n \approx 800$  observaciones con una distribución uniforme dentro del círculo usando coordenadas polares. A estos puntos se les asignó la categoría cero, posteriormente, se cambió la categoría a los puntos que cumplieran con las desigualdades. Posteriormente, se le añadió algo de ruido normal a cada punto para darle aleatoriedad a la base de datos pero manteniendo el patrón. Finalmente, se escala la base para centrarla en cero.

10. O al menos el presentado en este trabajo



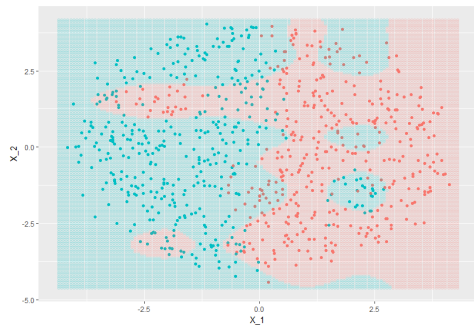
(a) Datos simulados



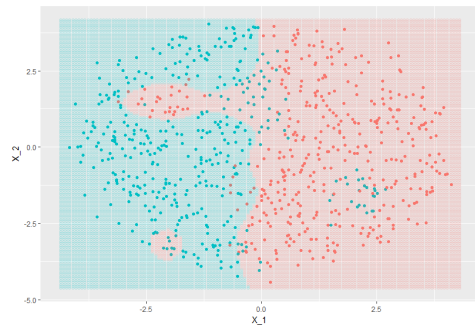
(b) Salida del software GeoGebra

Figura 10: Patrón yin-yang

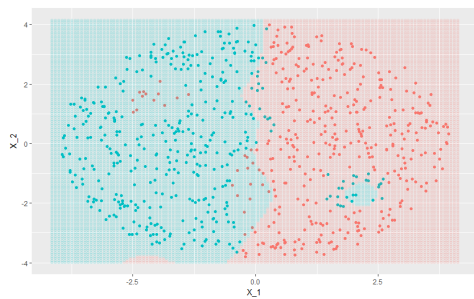
Se corrieron un sinnfín de realizaciones del modelo, tratando de calibrar los parámetros y captar de la mejor manera posible el patrón. Sin embargo y aunque el modelo casi siempre lograba una precisión de cerca de 85 %, no se logra la clasificación esperada identificando los puntos de color dentro de las áreas contrarias. De cualquier forma se observa como el algoritmo está tratando de encontrar este patrón. En la figura 11 se pueden ver fronteras de algunos de los mejores modelos. Para las dos primeras imágenes 11c y 11b, se observa como el modelo está tratando de encontrar las regiones anidadas, sin embargo, nunca se logra de forma precisa. En la imagen 11a, el modelo detecta relativamente bien la curva que separa las regiones, sino que detecta de forma aislada, el círculo azul de la esquina inferior derecha. Finalmente 11d, muestra una, de las muchas representaciones 3D que se hicieron al tratar de ajustar esta base de datos. Precisamente en esta última imagen esconde el porqué no se logró hacer la estimación correcta: la dependencia implícita entre los nodos.



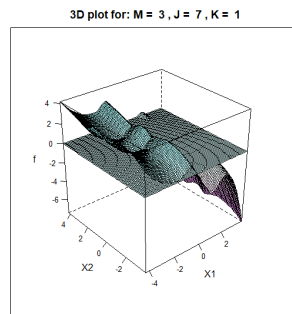
(a) Sobre-ajuste



(b) Mejor modelo



(c) Falta de precisión



(d) Gráfico 3D para uno de los modelos

Figura 11: fronteras de varios modelos para datos yin-yang

Estos nodos, en realidad están dividiendo el espacio bi-dimensional en una cuadrícula donde las interacciones son difíciles de discernir. Conforme aumenta el número de nodos, más complejo se vuelve el modelo. Es por ello, que los picos y valles se repiten en un patrón uniforme. Asimismo, dada la naturaleza global de los polinomios y esta interacción, el modelo tiene esta estructura decreciente siempre, derivando que los picos y los valles nunca alcancen las regiones extremas en polos opuestos. De igual forma, la uniformidad y simetría impar, inherente a esta base de datos, llevó a que la estimación de los parámetros fuera óptima dentro de las capacidades del modelo. Otra desventaja de esta base, es que estos modelos se tuvieron que correr con un número grande de nodos  $J \approx 20$ , derivando en un número de parámetros aún más alto.

## 0.5. Ejemplo 6 - el modelo en la práctica

Hasta ahora, todos los resultados de este trabajo han sido sobre bases de datos simuladas. Claramente se forman imágenes atractivas por construcción, sin embargo, no se está prediciendo nada en realidad pues se utiliza una metodología *dentro de muestra* para enfatizar las posibles fronteras del modelo. Por lo tanto y como último ejemplo, se presenta la base de datos de cáncer de mama de la Universidad de Wisconsin. Esta base de datos, es citada en varios trabajos de los años noventa, donde se tratan de hacer clasificaciones binarias usando una serie de procedimientos más robustos que los tradicionales GLM, **mangasarian1990pattern** y **bennett1992robust**.

De manera general y sin entrar en el detalle biológico de las variables como tal, se presenta un análisis exploratorio preliminar que se lleva a cabo para seleccionar las que se consideren relevantes. La base de datos cuenta con  $n = 699$  observaciones de las cuales el 34.5% representan pacientes infectados con tumores malignos representados por el color rojo (etiqueta cero). Se cuenta con diez variables (dimensiones) médicas sobre las características de los tumores como lo son: el tamaño, la uniformidad de la pared celular y la cromatina.<sup>11</sup> En la figura 12, se muestran los gráficos de puntos pareados para todas las posibles combinaciones de covariables además de cierta información adicional. Este proceso se lleva a cabo para tratar de

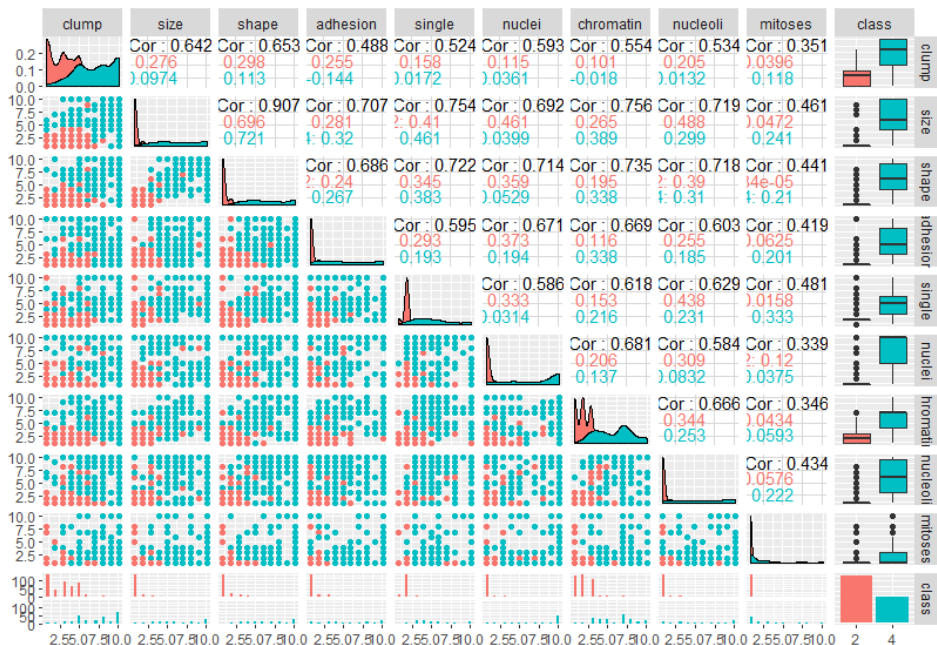


Figura 12: análisis exploratorio para selección de variables

11. Forma en la que se presenta la cadena de ADN en el núcleo celular.

seleccionar las covariables relevantes y estudiar un poco los datos. Asimismo, las covariables están codificadas en una escala a 10 puntos, por lo tanto, la representación gráfica de los datos se ve más como una cuadrícula que como un espacio real de variables. Derivado de esta exploración previa, se seleccionan las covariables *clump*, *size* y *chromatin*<sup>12</sup> debido a que parecieran ser las que mejor separan el espacio. En la figura 13 se presentan dos gráficos de puntos con algo de ruido para hacer notar que las regiones son un poco más complejas de lo que podría parecer a simple vista, además de que se tienen puntos idénticos con clasificaciones contrarias. Sin embargo, a simple vista se detecta cierto patrón en los datos. Para poder hablar de



(a) Variables *clump* y *chromatin*



(b) Variables *clump* y *size*

Figura 13: gráficos de puntos con ruido para separar las observaciones

predicción como tal, tiene que existir una base de datos contra la cual probar las estimaciones del modelo. Por lo tanto, la base original se parte en dos: un conjunto de entrenamiento con el 60% de las observaciones ( $n_{\text{train}} = 409$ ) y un conjunto de prueba con las observaciones restantes ( $n_{\text{test}} = 274$ ) sobre las que se evaluará el

12. Estas covariables corresponden a el espesor de los tumores, su tamaño y la textura de la cromatina en las células respectivamente.

modelo.<sup>13</sup> La realización final de entrenamiento del modelo se resume en la tabla 12, se escogen segmentos de recta continuos sobre tres nodos. Los resultados numéricos sobre la base de datos de prueba se presentan en la tabla 13 y el análisis de convergencia a través de las medias ergódicas en la figura 14a. Asimismo, se presenta cada  $\hat{f}_j$   $j = 1, 2, 3$  en las figuras 14b, 14c y 14d respectivamente.

Parámetros		Parámetro Sim.
$M = 2$	$N^* = 4$	$N_{\text{sim}} = 50,000$
$J = 4$	$\lambda = 13$	$k^* = 10,000$
$K = 1$	$n = 409$	$k_{\text{thin}} = 0$

Tabla 12: ejemplo 6 - datos médicos reales

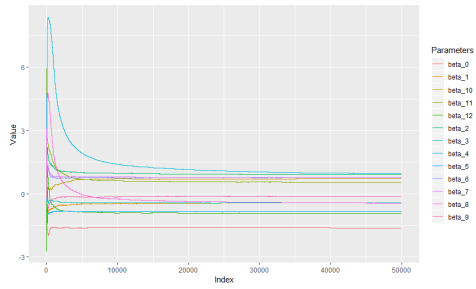
Info. predicción				
		$\hat{y} = 0$	$\hat{y} = 1$	
Est. Puntual	Media posterior	169	9	178
Precisión	95.6 %	3	93	96
$\log\text{-loss}$	0.1561	172	102	274

Tabla 13: datos médicos - resultados

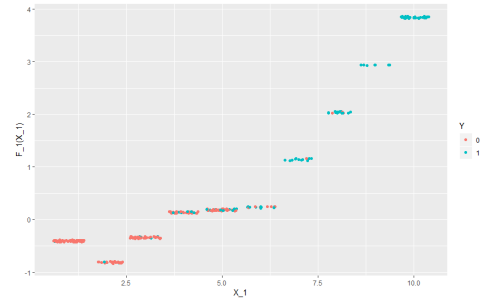
Inclusive, haciendo una predicción fuera de muestra los resultados son buenos logrando una precisión del 95.6 %. Asimismo, se resalta que inclusive en dimensiones ( $d = 3$ ) más altas si se escogen los parámetros correctos  $M$ ,  $J$  y  $K$ , el número total de parámetros ( $\lambda = 13$ ) no necesita aumentar demasiado para lograr una buena separación del espacio.<sup>14</sup> Derivado también del número de covariables es que no se

13. La diferencia de 16 observaciones entre la suma de entrenamiento y prueba, contra las 699 originales, se debe a que estas estaban incompletas y por lo tanto se descartan.

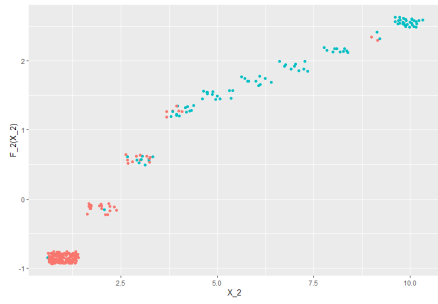
14. Se corrieron otras realizaciones del modelo aumentando el número de covariables  $d$ . Sin embargo, no logró aumentar significativamente la precisión del modelo.



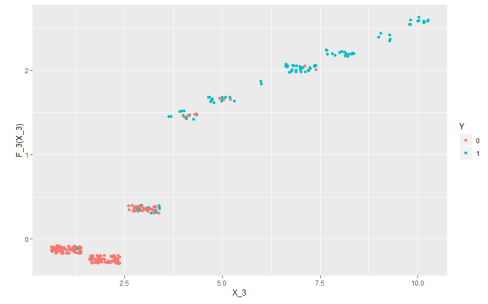
(a) Medias ergódica



(b)  $\hat{f}_1(x_1)$



(c)  $\hat{f}_2(x_2)$



(d)  $\hat{f}_3(x_3)$

Figura 14: media ergódica y funciones  $\hat{f}_j(x_j)$   $j = 1, 2, 3$



pueden hacer una visualización en el plano cartesiano  $\mathbb{R}^2$  como en los ejemplos anteriores. No obstante, la convergencia es clara y los resultados buenos, incluso usando segmentos de recta y un número de nodos pequeño. e hace notar que la codificación de las covariables usando una escala de diez puntos no es óptima para un modelo que se construye pensando en un espacio real de covaraibles  $\mathcal{X}^d$ , sin embargo, no parece afectar la estimación de los parámetros.