

INSTITUTO TECNOLÓGICO AUTÓNOMO DE MÉXICO



APRENDIZAJE ESTADÍSTICO
PROF. ALFREDO GARBUNO ÍÑIGO

Predicción de precios habitacionales usando Árboles Aditivos de Regresión Bayesianos

JOSÉ MANUEL MARTÍNEZ DEL CAMPO
GONZÁLEZ

C.U. 181861

9 de junio de 2022

1. Introducción

1.1. Descripción del problema

Este proyecto busca profundizar en el uso de métodos basados en árboles. Específicamente, se plantea el uso de los Árboles Aditivos de Regresión Bayesianos (BART por sus siglas en inglés). Este método, si bien similar a algunos otros vistos en clase como Boosting o Random Forests, busca implementar Árboles Bayesianos.

Se hará el desarrollo del algoritmo de BART tomando como base de datos “The Boston house-price data” (Harrison, et al. 1978). Se hará regresión sobre los datos para predecir el precio de las casas de esta ciudad, tomando en cuenta variables explicativas como el crimen, zona en la que se ubica la casa, y otras 11 variables explicativas.

Para este proyecto se buscó un método que no se hubiera visto en clase. Como tal, los Árboles Aditivos de Regresión Bayesianos son modelos aditivos basados en árboles que se mencionaron brevemente. La principal motivación de este proyecto fue ver su funcionamiento y desempeño como modelos predictivos. Por otro lado, se quería ahondar en los modelos bayesianos que se profundizaron en otras materias. Este proyecto, sin embargo, involucró una rama de la estadística bayesiana que no había explorado: los modelos no paramétricos. La importancia de este trabajo radica en el desarrollo del modelo BART desde una perspectiva matemática y de tener una primera aproximación a los modelos bayesianos no paramétricos. La prueba del desempeño del modelo son también de importancia, pues nos muestra la gran capacidad predictiva que se obtiene cuando se implementa estadística bayesiana.

1.2. Objetivos

- **Datos:** Exponer una idea general de la información histórica y contextual así como la relación que existe con la variable objetivo.
- **Modelo:** Describir a detalle qué es el modelo BART, las partes que lo componen y cómo se obtiene información de las distribuciones posteriores. Después, se buscarán predicciones sobre la variable *precio medio* implementando el modelo BART y otros modelos vistos como *Boosting* y Bosques Aleatorios. Se implementarán técnicas de aprendizaje vistas a lo largo del curso optimizar los hiperparámetros respectivos de cada modelo.
- **Resultados:** Elegir el modelo óptimo y comparar el desempeño y resultados obtenidos con los modelos *Boosting* y Bosques Aleatorios. Informar al lector sobre las variables más relevantes a tomar en consideración al valorar una propiedad habitacional en la ciudad de Boston.

2. Descriptivo del modelo

2.1. Modelos bayesianos no paramétricos

BART es un modelo Bayesiano no paramétrico. Explicaremos los dos conceptos en este apartado.

Primero, los modelos **bayesianos** son aquellos que se apoyan en el teorema de bayes 1 para hacer inferencia. Estos modelos buscan explicar o ajustarse a datos a través de una distribución inicial que explica nuestro conocimiento previo de los parámetros. Por otra parte, se toma en cuenta la función de verosimilitud; la probabilidad de que ocurra la realización que efectivamente ha ocurrido dados los parámetros. (Nonparametric Bayesian Methods: Models, Algorithms, and Applications I, 2017):

$$\mathbb{P}(\text{parámetros}|\text{datos}) \propto \mathbb{P}(\text{datos}|\text{parámetros})\mathbb{P}(\text{parámetros}) \quad (1)$$

Al multiplicar estas dos funciones se tiene una relación proporcional con la función posterior que queremos calcular.

Después, tenemos la parte **no paramétrica** del modelo. Esto se refiere a que el espacio paramétrico está constantemente creciendo, puede que no esté acotado o incluso que sea de dimensión infinita (Orbez, 2010). Para el modelo, sin embargo, se considera un subespacio finito de este espacio paramétrico.

Para ejemplificar cómo se puede tener un número infinito de parámetros se puede pensar en el “Efecto Wikipedia”. Al adentrarnos en un *link*, éste nos va a referir a otros diferentes. Si continuamos, no podríamos leer, o en su defecto analizar con un algoritmo, todos los artículos por la cantidad de referencias que existen. Aparte de eso, si tomamos en cuenta que día tras día se escriben nuevos artículos esto lo hace menos viable (Nonparametric Bayesian Methods: Models, Algorithms, and Applications I, 2017). Otro ejemplo surge al tomar en cuenta funciones Gaussianas. En primera instancia, bajo la estadística paramétrica, se busca encontrar el parámetro que mejor ajuste a un conjunto de datos. Por otro lado, bajo la estadística no paramétrica, se intentará centrar una distribución Gaussiana en cada observación y luego combinar estos resultados(Orbanz, 2010).

2.2. Introducción al modelo BART

Este modelo fue propuesto por Chipman en 2010 y fue implementado en R en 2010 como BayesTree por el mismo. Es un trabajo que retoma su propuesta inicial (Chipman, 2002) de un modelo de un único árbol bayesiano, para ahora hacerlo una suma de árboles. Después, fue implementado en la librería *tidymodels* a través de *dbart* (Tidymodels, 2010)

Consideramos el problema general de inferencia sobre una función f para predecir los valores de Y usando un vector p -dimensional de variables explicativas $x = (x_1, x_2, \dots, x_p)$.

$$Y = f(x) + \varepsilon, \quad \varepsilon \sim N(0, \sigma^2). \quad (2)$$

Como se busca hacer predicciones usando observaciones de x , entonces $f(x) = E[Y|x]$ será la media de Y dada una observación. Luego, se aproxima la ecuación 2 con una suma de m árboles bayesianos. Usamos la función definida por Chipman como $h(x) := \sum_{i=1}^m g_i(x)$, donde g_i es un árbol de regresión bayesiano. De tal modo, la regresión se hará sobre la ecuación

$$Y = h(x) + \varepsilon, \quad \varepsilon \sim N(0, \sigma^2). \quad (3)$$

Es importante resaltar que, a diferencia de los Bosques Aleatorios, la suma de árboles se basa en iteraciones del algoritmo *backfitting*. Este algoritmo, a grandes razgos, ajusta un modelo que involucra varios predictores al actualizar el ajuste para cada predictor manteniendo los demás fijos (James, G. et all. 2021). Esto genera una construcción de árboles de forma secuencial, en donde cada árbol b_i explica una parte pequeña y diferente de f .

Como resultado se obtiene una distribución posterior de la estimación de $f(x) = E[Y|x]$ para algún vector de predictores x , así como los intervalos de confianza.

2.3. Elementos del modelo

2.3.1. Suma de árboles

Consideramos ahora los componentes de cada árbol g_i . Tenemos primero un árbol binario T_i que cuenta con un conjunto de parámetros de decisión para sus nodos terminales: $M_i = \{\mu_{i1}, \mu_{i2}, \dots, \mu_{ib_i}\}$. El tamaño del árbol i nos lo dice la variable b_i . Cada μ_{ik} es una decisión binaria de un solo elemento del vector de predictores x ; esto es, corresponde a que $\{x_\ell \in A\}$ para algún $\ell \in \{1, \dots, p\}$, $A \subset \mathcal{X}_\ell$ y \mathcal{X}_ℓ el espacio parametral de x_ℓ .

Como cada árbol de regresión T_i está asociado a una conjunto de parámetros M_i , expresamos la ecuación 3 de manera más explícita como

$$Y = \sum_{i=1}^m g_i(x, T_i, M_i) + \varepsilon, \quad \varepsilon \sim N(0, \sigma^2). \quad (4)$$

Como tenemos una suma de $m > 1$ árboles de regresión, esto representa una interacción entre cada uno de los parámetros μ_{ik} . Entonces, cada uno de estos aportará información para aproximar $f(x)$. BART es un modelo con un gran número de parámetros. Podemos ver de este apartado que la ecuación 4 dependerá de m , de las parejas $(T_1, M_1), \dots, (T_m, M_m)$ y de la varianza de los errores σ^2 . Añadiendo todos los parámetros μ_{ik} para cada uno de los conjuntos M_i .

2.3.2. Independencia y simetría

Es importante recalcar que, para que el algoritmo sea computacionalmente eficiente, es necesario hacer el supuesto de que los parámetros de las hojas de los árboles, μ_{ij} , son independientes entre sí para cada árbol. Como también σ es independiente de M_i y de T_i , tenemos que las distribuciones iniciales son independientes:

$$\begin{aligned}
\mathbb{P}(T_1, M_1, T_2, M_2, \dots, T_m, M_m, \sigma^2) &= [\prod_{j=1}^m \mathbb{P}(T_j, M_j)] \mathbb{P}(\sigma^2) \\
&= [\prod_{j=1}^m \mathbb{P}(M_j | T_j) \mathbb{P}(T_j)] \mathbb{P}(\sigma^2) \\
&= [\prod_{j=1}^m \prod_{\ell=1}^{b_j} \mathbb{P}(\mu_{j\ell} | T_j) \mathbb{P}(T_j)] \mathbb{P}(\sigma^2)
\end{aligned}$$

Esto nos permitirá hacer un muestreo de Gibbs tomando valores aleatorios de cada una de las siguientes distribuciones iniciales.

2.3.3. Distribuciones iniciales

Para esta sección, se eligen distribuciones iniciales para todos los parámetros mencionados en la sección anterior. Es en esta parte dónde surge el carácter bayesiano del modelo.

Además, buscamos regularizar cada una de las distribuciones, favoreciendo a que los árboles tengan pocos nodos. Esto nos permitirá mantener a los árboles pequeños, haciendo que cada uno de estos mantenga una relevancia al interactuar los unos con los otros. De otro modo, podría tenerse uno solo que acaparara toda la información brindada por [4](#), perdiendo la propiedad aditiva del modelo.

Para el parámetro m , se recomienda usar un valor de 50 (Chipman, 2010). Sin embargo, se puede también usar validación cruzada para encontrar el valor óptimo para cada conjunto de datos. Ahondaremos en esto en la sección de resultados.

■ Distribución de T_i

La distribución inicial de que la longitud del árbol i sea d está dada por

$$\alpha(1+d)^{-\beta}, \quad \alpha \in (0, 1), \beta \in (0, \infty].$$

Esta distribución favorece a que los nodos terminales rondan los valores de 2 y 3. Los valores de α y β afectan que tan agresivo es la preferencia por árboles pequeños. Estos son otros dos parámetros que se pueden encontrar por validación cruzada. . También, se proponen los valores de $\alpha = 0.95$ y $\beta = 2$ para valores de m cercanos o mayores a 50.

■ Distribución de $\mu_{ij} | T_i$

Para la distribución de $\mu_{ij} | T_i$ utilizaremos una distribución normal

$$N(0, \sigma_\mu^2), \quad \sigma_\mu = \frac{1}{2k\sqrt{m}}.$$

Notamos que μ_{ij} son *i.i.d.* dado un árbol T_i . Esto nos genera las distribuciones de cada uno de los elementos de M_i . Estas distribuciones también favorecerán a que los árboles sean pequeños. Entre menor sea σ_μ , menor será el efecto de cada árbol sobre los otros.

Se sugiere $k = 2$ para el hiperparámetro. Se realizará validación cruzada para encontrar un valor óptimo para nuestro problema en el intervalo sugerido $(1, 3)$.

■ Distribución de σ^2

Para la distribución de σ^2 vamos a usar una distribución ji-cuadrada inversa con dos hiperparámetros. de tal forma que

$$\sigma^2 \sim \frac{v\lambda}{\chi_v^2}.$$

Hacer validación cruzada para estos parámetros no está implementado en *tidymodels*. Los valores que toman estos hiperparámetros por *default* son $(v, \lambda) = (3, 0.90)$.

2.4. Extracción de la información de la distribución posterior

La extracción de la distribución posterior se hace por medio de un muestreo de Gibbs. El algoritmo Gibbs es un método de Montecarlo basado en cadenas de Markov (MCMC por sus siglas en inglés) que nos ayuda a iterar sobre variables o funciones para encontrar aproximaciones de una distribución. El algoritmo toma valores al azar siguiendo una distribución inicial, y luego ejecuta una función para obtener el valor posterior. Esto lo ejecuta varias veces, para así tomar

En nuestro caso, el algoritmo Gibbs lo que hará será modificar [6](#) una por una cada una de las $2m - 1$ distribuciones; manteniendo las otras $2m$ distribuciones fijas [5](#). Cada iteración del algoritmo Gibbs implica ajustar todas las distribuciones. Definimos las otras distribuciones excluyendo (T_j, M_j) como

$$\mathcal{R}_i = \sum_{i \neq j} (T_i, M_i) \tag{5}$$

Las distribuciones que se modificarán van en el siguiente orden:

$$\begin{aligned}
1 : & T_1 | \mathcal{R}_1, \sigma \\
2 : & M_1 | T_1, \mathcal{R}_1, \sigma \\
3 : & T_2 | \mathcal{R}_2, \sigma \\
4 : & M_2 | T_2, \mathcal{R}_2, \sigma \\
& \vdots \\
2m-1 : & T_m | \mathcal{R}_m, \sigma \\
2m : & M_m | T_m, \mathcal{R}_m, \sigma \\
2m+1 : & \sigma^2 | T_1, M_1, T_2, \dots, T_m, M_m
\end{aligned} \tag{6}$$

Se un árbol T_i se modificamediante las acciones de podar un nodo, crecer un nodo o cambiar la regla de un nodo. Al modificar cada uno manteniendo el resto fijo, la suma de árboles va evolucionando y ajustando los datos que falten por explicar.

2.5. Comparación con *Boosting* y Bosques Aleatorios

Una de las principales diferencias entre el modelo BART y *boosting* es que, para un número fijo de árboles m , BART usa un algoritmo de *backfitting* con un muestreador Gibbs, para iterar a través de los árboles e ir haciendo el ajuste. El hiperparámetro k toma el papel del parámetro de *boosting* que reduce la influencia que tiene un árbol sobre los otros. Ambos parámetros afectan la importancia que se le puede dar a un árbol

En cuanto a bosques aleatorios, la diferencia es que los árboles van ajustandose considerando los demás árboles. Por lo tanto, se buscará ir describiendo todos los datos, no tomar conjuntos de datos aleatoriamente.

3. Análisis exploratorio de los datos

3.1. Contexto de los datos

Boston, situada en el estado de Massachussets, es una de las ciudades más importantes y antiguas en Estados Unidos. Con más de 7.5 millones de habitantes, la vivienda en esta ciudad y su zona conurbada es muy cotizada.

Además, es una de las ciudades más caras para vivir en Estados Unidos. Una casa promedio en la actualidad ronda los \$787,320 dólares, con una apreciación anual de 5.51% por año (Schiller, 2022). En contraste, el precio promedio de las casas en el país es de \$374,900; casi la mitad.

Se registró un gran aumento de los delitos violentos desde comienzos de los años 1990. Esto fue un motivo para algunos barrios de la ciudad fueran marcados como zonas altamente conflictivas y que, naturalmente, los inquilinos prefirieran residir en zonas más seguras.

En la actualidad, Boston está entre el 19 % de las ciudades más inseguras de Estados Unidos.

Boston tiene una inmensa zona conurbada. La ciudad cuenta con apenas el 10 % de la población, mientras que el 90 % viven en las zonas aledañas. Esto provoca largos trayectos de transporte, haciendo muy cotizadas las zonas céntricas. Sin embargo, el transporte público es uno de los mejores en Estados Unidos, haciendo prescindible el uso del automóvil; a diferencia de la mayoría de las ciudades del país. Esto beneficia a personas que no pueden solventar un automóvil; las cuales usualmente viven en las zonas más alejadas (Schiller, 2022).

3.2. Análisis exploratorio

La base de datos “The Boston house-price data” (D. Harrison, et all. 1978) contiene 14 variables y tiene 506 observaciones. La variable que se quiere explicar es *medv*, el valor medio de la propiedad habitacional . En el siguiente histograma se pueden ver las observaciones registradas en la base de datos para esta variable. En el cuadro 1 se muestra una tabla con los nombres de las variables y sus significados.

A partir del análisis exploratorio de los datos, se determinó que no era necesario remover datos. Esto debido a que la base de datos tenía valores de en todas las observaciones y a pesar de que se encontraron algunos valores atípicos en las variables, estos se conservaron al ejecutar el modelo. La única variable que presentó una gran cantidad (93 %) de observaciones como 0 fue *chas*. Esto es por la naturaleza del clasificador binario, por lo que se decidió conservar esta variable.

Nombre de la variable	Explicación
CRIM	Crimen per cápita por barrio
ZN	Proporción de áreas residenciales por 25,000 pies cuadrados
INDUS	Proporción de industrias por acre por barrio
CHAS	Charles River dummy variable (1 si está en el río; 0 e.o.c.)
NOX	Concentración de óxido nítrico (partes por 10 millones)
RM	Promedio de habitaciones
AGE	Proporción de casas antiguas (1940) ocupadas por sus dueños
DIS	Distancia a los 5 centros de empleo
RAD	Accesibilidad a periféricos
TAX	Impuestos en \$10,000
PTRATIO	Media de la tasa de docente-estudiante por barrio
B	$1000(Bk - 0.63)^2$; con Bk la proporción de afroamericanos en un barrio
LSTAT	% estatus socioeconómico
MEDV	Precio medio en \$1000's

Cuadro 1: Tabla con las variables de la base de datos

El histograma de la variable objetivo *medv* mostrado en la figura 1 nos muestra una distribución normal con algunas observaciones atípicas. El precio de las propiedades se concentra en valores cercanos a \$20,000. Tenemos pocas casas con valor mayor a \$30,000. El valor mínimo de una casa es de \$5,000, mientras que el máximo es de \$50,000. Se puede observar un amplio rango de precios, cuya media es de \$22,532.

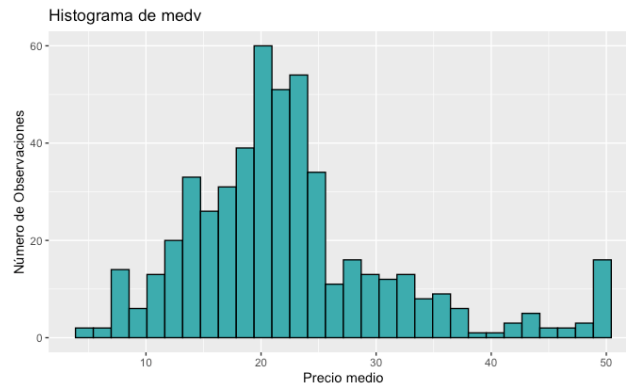


Figura 1: Histograma de medv usando ggplot2

En la figura 2 se muestran diagramas de dispersión de las variables que se consideraron a primera instancia que más pudieran afectar al valor de la propiedad. Entre estas se vio que, naturalmente, el crimen del barrio es un factor que afecta negativamente al valor de las casas; sin embargo, hay propiedades con valor de \$50,000 situadas en barrios con índices de crímenes por arriba de la media (3.6). También se observan muchas entradas cercanas a cero, y una tendencia a que en los barrios más peligrosos los precios bajen considerablemente. Por otra parte, se ve que la proporción de zona habitacional afecta de manera positiva al valor de las casas. La mayoría de las casas por debajo de los \$20,000 tienen un porcentaje de zona habitacional de 0, mientras que para las casas con un valor superior a los \$30,000 el porcentaje aumenta. También se exploró la variable *rm*, el promedio de habitaciones por propiedad del barrio. Se observó una tendencia positiva y lineal muy clara con respecto al *medv*.

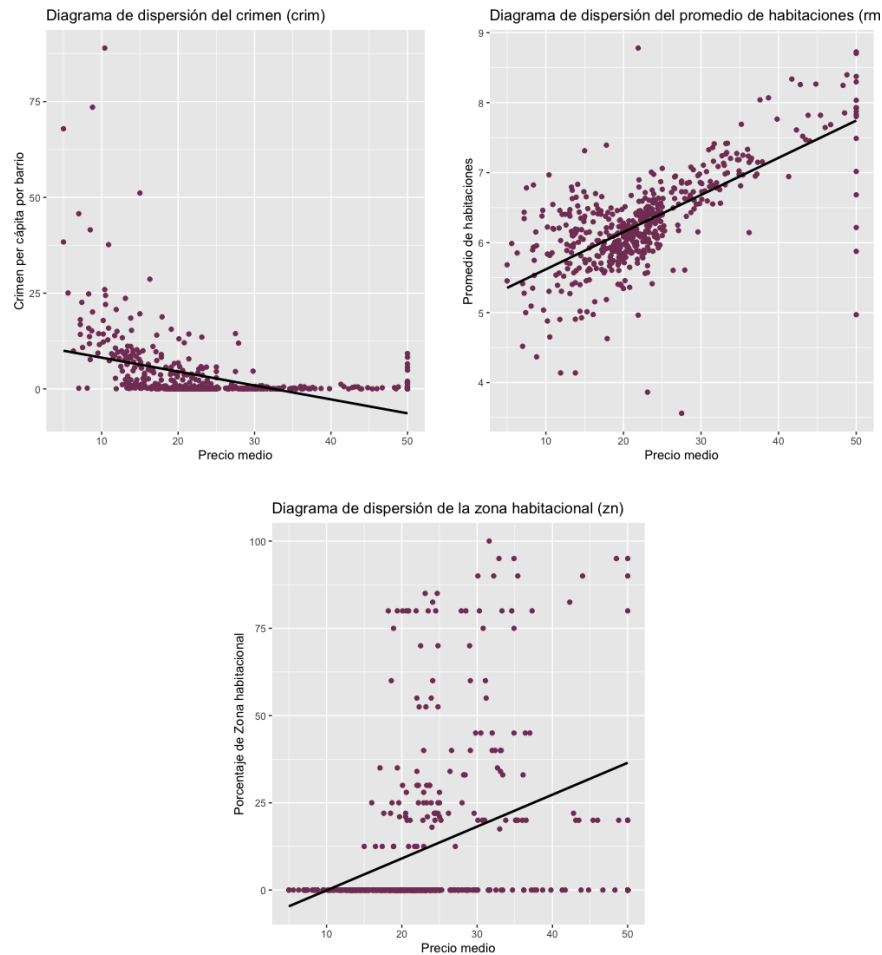


Figura 2: Diagramas de dispersión usando ggplot2(ref)

Por último, observamos la matriz de correlación en la figura 3. Al fijarnos en la variable

medv, podemos ver que las variables con las que mayor correlación tiene son *zn* y *lstat*: el número de habitaciones promedio y el estatus socioeconómico del barrio. Por otro lado, vemos que la variable que menos explica el precio es *chas*. Como vimos, esta variable no tiene muchas entradas diferentes de cero; por lo que explicará poco el precio.

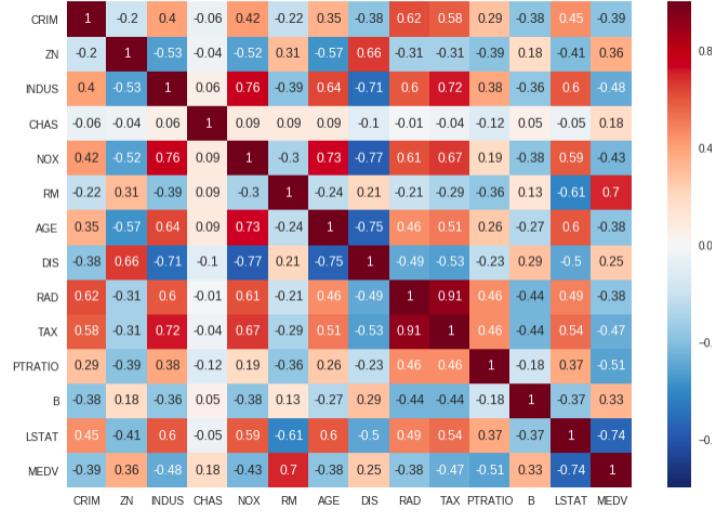


Figura 3: Matriz de correlación

4. Resultados

Primero, se realizó una separación de los datos en un conjunto de prueba y de entrenamiento, con el 80 % de los datos.

Para la implementación de BART consideramos dos formatos. El primero es BART-cv, la versión de BART en la que hacemos validación cruzada para encontrar los hiperparámetros (m, k, α, β) que optimizan el ajuste a los datos. Se consideraron los posibles valores para realizar la validación cruzada:

m	k	α	β
10,50,110,150	1,2,3,4	0.05,0.35,0.70,0.95	1,2,3,4
10,60,110,170,210	1,2,3,4,5	0.95	1,2,3,4,5

Cuadro 2: Parámetros con los que se hizo validación cruzada

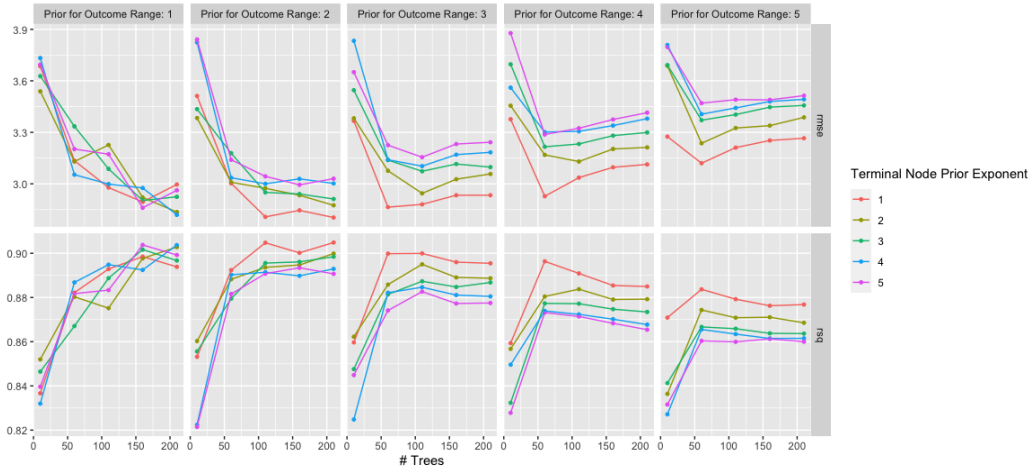


Figura 4: Validación cruzada de BART

Con la figura 4 se muestra la comparación de los modelos con los diferentes parámetros usando la métrica de $rmse$ y la R^2 . Se obtuvo que el mejor rendimiento de BART con validación cruzada fue con los parámetros $m = 170$, $k = 2$, $\beta = 1$, $\alpha = 0.95$.

Los resultados basados en la métrica $RMSE$ se muestran en el cuadro 3. Se puede ver una amplia mejora con respecto a los otros métodos basados en árboles.:

Método	RMSE
<i>BART-cv</i>	1.938
<i>BART-default</i>	2.313
<i>Bagging</i>	3.422
Bosques Aleatorios con <i>cv</i>	3.183
<i>Boosting-cv</i>	3.339
Árbol de Regresión	4.783

Cuadro 3: Tabla con las variables de la base de datos

Se intentó ver la relevancia de cada una de las variables en nuestro método predictivo. Esto no fue posible por que no estaba implementada esa función para el motor *dbarts*. Sería interesante poder verlo en alguna otra de las librerías que implementan BART.

5. Conclusiones

5.1. Resumen

En resumen, lo que se hizo fue indagar en cómo funciona un modelo bayesiano no paramétrico. Se revisó la teoría detrás de este método y se tuvo una primera aproximación a la estadística bayesiana no paramétrica.

Se implementó el modelo BART en el lenguaje de programación R usando *tidymodels*, para finalmente hacer validación cruzada y obtener un modelo que predijera sobre la base de datos de *Boston-housing*.

Se pudo ver una amplia mejora tanto en el modelo usando validación cruzada como en el que usaba los parámetros sugeridos. Es importante notar que al realizar la validación cruzada, el poder predictivo aumenta pero el costo computacional es notablemente mayor. Para balancear esto, se sugiere tomar únicamente 3 valores para el parámetro m : 50, 100 y 200; y uno o dos valores para los demás parámetros.

5.2. Posibles líneas de trabajo e investigación futuras

BART es un algoritmo que puede tener muchos usos, y como se vio en la comparación con otros algoritmos, puede tener mejoras de predicción y de implementación.

Posibles trabajos posteriores a este podrían ser la comparación del algoritmo BART con redes neuronales y otros algoritmos con mayor poder predictivo. Se observó que BART usando *cross-validation* tuvo un rendimiento considerablemente mejor al de los otros algoritmos escogidos; sin embargo sería interesante ver la comparativa con algoritmos más complejos. También, comparar el rendimiento de la librería de BayesTree con la más reciente *bartMachine* [Kapelner (2018), implementado en R como *bartMachine* por Kepler, Bleich (2004)] (h)

Adicionalmente se podrían utilizar diferentes bases de datos y hacer una comparación del rendimiento. Existen infinidad de bases de datos con características similares a la de Boston. Se podría hacer una selección y comparar qué tan bien le va a BART con otros datos.

Otro de los posibles proyectos que también se podrían explorar es el uso de BART en problemas de clasificación. Esta herramienta está también implementada en *tidymodels*, es un problema que podría seguir de manera natural.

Por último, sería interesante ahondar en otros algoritmos bayesianos no paramétricos. Estos tienen mucho camino por recorrer y que explorar, por lo que se me hizo un buen lugar para comenzar a indagar sobre la construcción de un trabajo de tesis.

Referencias

- [1] Bayesian additive regression trees via dbarts — details.bart_dbarts. (2010). Tidymodels. https://parsnip.tidymodels.org/reference/details_bart_dbarts.html
- [2] G. James, D. Witten, T. Hastie, R. Tibshirani (2021)- An Introduction to Statistical Learning - with Applications in R-(Springer Texts in Statistics) Springer Science+Business Media .pdf
- [3] Hugh A. Chipman, H. A., George, E. I. and McCulloch, R. E. (2002). Bayesian treed models. Machine Learning 48 299–320.

- [4] Hugh A. Chipman, Edward I. George, Robert E. McCulloch. "BART: Bayesian additive regression trees." *Ann. Appl. Stat.* 4(1):266 - 298, March 2010. <https://doi.org/10.1214/09-AOAS285>
- [5] Harrison, D., Rubinfeld, D. L. (1978). The Boston house-price data (Versión 1) [Hedonic prices and the demand for clean air]. <http://lib.stat.cmu.edu/datasets/boston>
- [6] Kapelner, A., Bleich, J. (2013). bartMachine: Machine learning with Bayesian additive regression trees. arXiv preprint [arXiv:1312.2171](https://arxiv.org/abs/1312.2171).
- [7] Nonparametric Bayesian Methods: Models, Algorithms, and Applications I. (2017, 25 enero). [Vídeo]. YouTube. <https://www.youtube.com/watch?v=I7bgrZjoRhM>
- [8] Orbanz, P. (2010). Bayesian Nonparametric Models. Harvard. Recuperado 5 de junio de 2022, de <https://groups.seas.harvard.edu/courses/cs281/papers/orbanz-teh-2010.pdf>
- [9] Schiller, A. (2022, 6 abril). Boston, MA Appreciation Rate Trends Housing Market Data. NeighborhoodScout. <https://www.neighborhoodscout.com/ma/boston/real-estate>