

# Problem Set 3 – GRUPO 6

# Víctor Dulio Chique

# Víctor Iván Sánchez

# Natalia Castro

# HACIENDO DINERO CON ML: PRECIOS HEDONICOS

1. Introducción:

Las estimaciones de precios de viviendas están basadas en el modelo de precios hedónicos, que consiste en establecer la relación entre el precio de un bien mercadeable, en este caso viviendas urbanas, y las características del bien. Las viviendas son productos diferenciados desde la perspectiva del consumidor, pues los atributos o características de las viviendas les proporcionan utilidad. Por su parte, los productores de estos bienes incurren en costos que dependen de los atributos asignados a estas viviendas. Entonces, la interacción en este mercado entre consumidores y productores determinan la senda de equilibrio del precio de la vivienda.

Las viviendas pueden ser descritas por un conjunto de características estructurales como tamaño, número de habitaciones, numero de baños, número de garajes, tipo de vivienda (casa o apartamento), antigüedad, entre otro, y adicionalmente por un conjunto de características del entorno, que pueden acceso a bienes y servicios públicos como parques, centros comerciales, universidades, transporte público, entre otros.

2. Datos

2.1 Descripción de los Datos

2.2 Estadísticas Descriptivas

Correlaciones

3. Modelo y Resultados

El precio de la vivienda (P) es una función de las características estructurales y los atributos del entorno. Así, estas se representan por un vector de características y atributos , donde denota una de las característica o atributos de la vivienda. Y cada vivienda es denotada por , donde superíndice indica una propiedad distinta con distinto vector de características y atributos . Por lo tanto, el precio de venta de una vivienda es función de las características y atributos de la misma.

En nuestro modelo, el precio dependerá de un conjunto de características estructurales como número de habitaciones, numero de baños y tipo de vivienda (casa o apartamento), y adicionalmente de un conjunto de características del entorno como la distancia al parque, distancia al centro comercial, distancia a la universidad y cercanía a avenida principal.

Random forest (8 predictores):

Este tipo de modelos busca realizar las predicciones a través de varios árboles de decisión que se construyen a partir de un subconjunto aleatorio de un número determinado de variables predictoras. En este ejercicio de utilizaron 8 de estas variables: bedrooms, bathrooms, property\_type, superficie, distancia\_parque, distancia\_comercial, distancia\_avenida\_principal, distancia\_universidad.

Adicionalmente, el modelo de *Random Forest* se entrena sobre un subconjunto de los datos de la variable de interés en este caso: el precio de las casas y apartamentos de Bogotá. El subconjunto de datos de entrenamiento se obtuvo a través de *cross validation* con k=10.

Finalmente, se construyó una grilla con 2, 3, 4 ,5 y 8(bagging) predictores aleatorios. La regla de corte fué “varianza” y los diferentes tamaños de nodos: 1,2,3 y 6.

Como se puede ver en la gráfica XX a medida que se incrementa el número de predictores disminuye el RMSE, sin embargo después de 3 predictores aleatorios comienza a incrementarse.

El mejor número de nodos es 1. El cuadro XX muestra el MEA y MAPE dentro y fuera de muestra.

Chart, line chart

Description automatically generated

|  |  |  |
| --- | --- | --- |
|  | MAPE | MAE |
| Entrenamiento | 0.051 | 29403419 |
| Testeo | 0.121 | 68925784 |

Aunque el MAPE dentro de muestra es bajo no logra mantener el mismo porcentaje en el conjunto de evaluación. Esto se puede deber a *overfitting*. Sin embargo, aunque la variación entre los dos subconjuntos es evidente, una predicción en testeo con un MAPE del 20% es razonable.

Arboles de decisión (8 predictores):

Los árboles de decisión dividen los datos en subconjuntos utilizando particiones recursivas binarias. Se utilizaron 8 variables: bedrooms, bathrooms, property\_type, superficie, distancia\_parque, distancia\_comercial, distancia\_avenida\_principal, distancia\_universidad. Para el conjunto de entrenamiento y evaluación se utilizó *cross validation* con k=10. Se obtuvo el siguiente modelo:

Diagram

Description automatically generated

|  |  |  |
| --- | --- | --- |
|  | MAPE | MAE |
| Entrenamiento | 0.217 | 122791972 |
| Testeo | 0.218 | 122787377 |

El modelo escoge la variable “Superficie” como la más relevante. Para apartamentos con menos de 97m2 predice un precio de $441´000.000. Para las propiedades con un área entre 97m2 y 122 m2 $580´000.000 y las que tienen más de esa área un costo de $824´000.000.

Este es un modelo muy sencillo que puede resultar práctico para tener una visión general de los precios de acuerdo a una variable relativamente fácil de obtener pero que tiene un error de predicción mayor que el modelo de *Random Forest*. Este último sin embargo no es claro en especificar los predictores más importantes pues es escoge 3 de los 8 posibles de forma aleatoria.

**Modelos de Regularización**

También se han planteado modelos de regularización para contrastarlo con el mejor modelo. El modelo que se planteo es el siguiente, con distintas especificaciones:

La variable dependiente es el precio de la vivienda (casa o apartamento) medido en pesos colombianos de la ciudad de Bogotá.

La primera variable predictora es el tipo de propiedad, es decir si es casa o apartamento, no se le hizo ningún tratamiento pues es una dummy; la otra variable es el área de la propiedad medido en metros cuadrados, estos datos fueron obtenidos de la descripción del anuncio, lográndose reducir el número de observaciones faltantes de la base train de 30079 a 8798, luego estos datos fueron imputados mediante el algoritmo K-Nearest Neighbors (KNN), considerando los 6 vecinos más cercanos; la variable habitaciones resulta del valor máximo entre bedrooms y rooms; la variable baños tuvo 10071 valores faltantes y fue imputada mediante KNN con seis vecinos cercano, y las variables de distancia más cercanas al parque, centro comercial, Av. Principal y universidad fueron calculadas usando información geográfica como las coordenadas en formato latitud y longitud.

Adicionalmente en la base train, mediante la función boxplot se detectaron y excluyeron los outliers de la variable dependiente y las variables independientes, considerando como todo valor que está fuera de los bigotes, es decir las líneas que se determinan como el tercer cuartil + 1.5 veces el rango intercuartílico y el primer cuartil -1.5 veces el rango intercuartílico.

En la base de datos test, con el propósito de no perder observaciones, en el caso del área los outliers fueron reemplazados por el promedio dentro del rango 500 y 30 metros cuadrados y los NAs fueron imputados con KNN.

En este grupo de modelos se plantean diversas especificaciones como lineal, semilogarítmico y polinómicos. Los resultados de predicción que mejor se ajustan provienen de una especificación lineal, cuyos coeficientes se muestran en la siguiente tabla:

Modelo de Regresión



El grado de penalización de los modelos de regularización está controlado por el hiperparametro lambda, cuyos óptimos son los siguientes:

Hiperparametros



El mejor modelo es el Lasso, porque dentro de la muestra tiene la menor métrica RMSE.

RMSE



Se realiza la predicción con la base de datos test y se obtiene los precios predichos con la siguiente distribución:



El escore obtenido con dichas predicciones del modelo Lasso en kaggle es un MAE de 282445492.70880, que es superado por el modelo Super Learner. Por lo tanto, en este caso, los modelos de regularización no superan las predicciones de modelo Super Learner.

Los otros modelos estimados de este grupo son modelos no lineales, es decir donde el precio está expresado en logaritmos y otro con los predictores en polinomios. Los resultados se muestran en los anexos. El modelo semi-logaritmico, si bien es lo recomendable para reducir la magnitud de la variable dependiente, el problema se da en las predicciones, pues llega a predecir por ejemplo precios por encima de $ 4 mil millones, lo cual empeora la métrica MAE. Por su parte, el modelo con polinomios de las variables dependientes e interacciones entre ellas, arroja valores predichos del precio negativos para el test, lo que supone un MAE peor fuera de la muestra.

3. Conclusiones y Recomendaciones

4. Bibliografía

Anexos

**Modelo Lasso con variable precio en logaritmos**

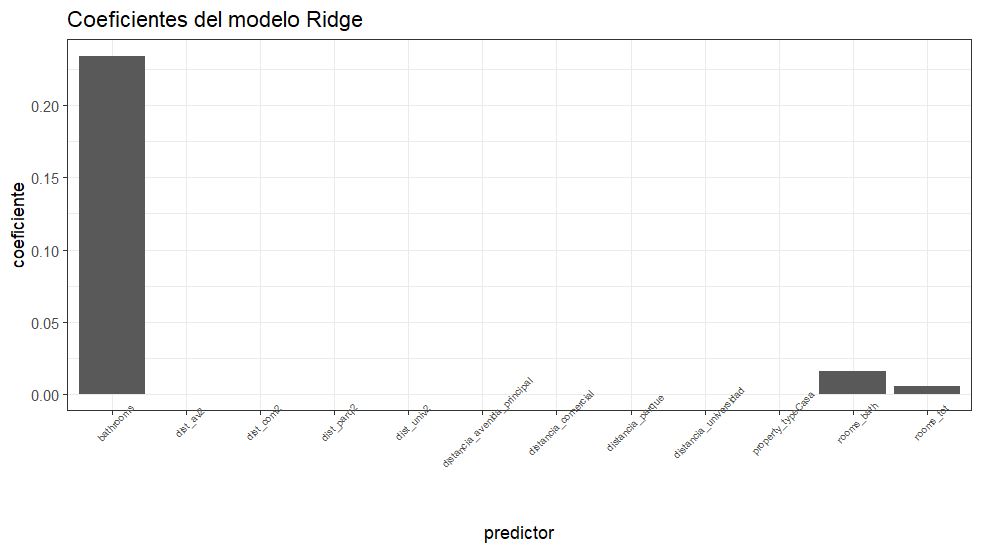
train <- Bogota\_train %>%

select(lprice, property\_type, rooms\_tot, bathrooms, distancia\_avenida\_principal,

distancia\_comercial, distancia\_parque, distancia\_universidad, rooms\_bath,

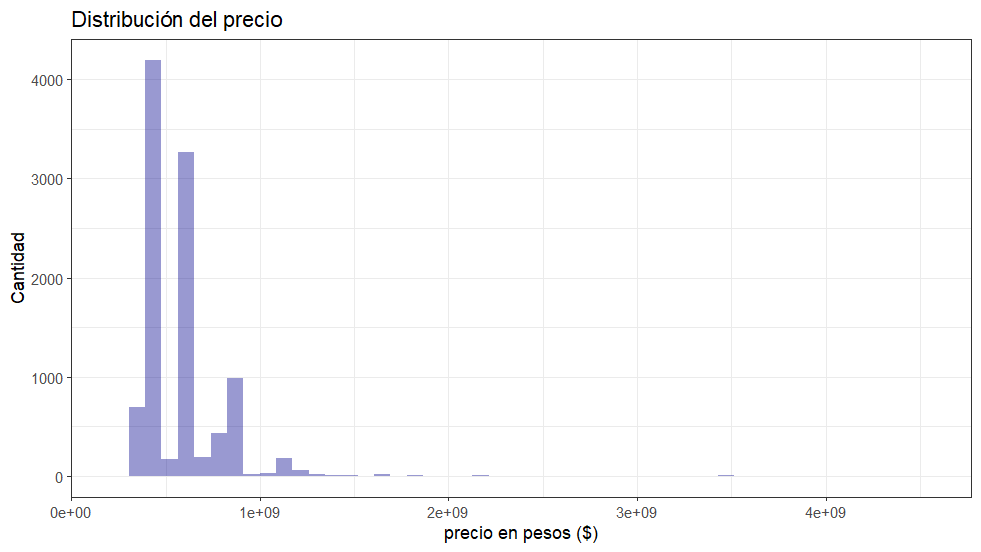
dist\_av2, dist\_com2, dist\_parq2, dist\_univ2)

Lambda optimo: 9.69804377996242e-05"



"Error (mse) de entrenamiento: 0.0874021221908704"

"Error (mae) de entrenamiento: 0.239060247990877"



**Modelo Lasso con polinomios**

OLS RIDGE LASSO ELASTIC\_NET

(Intercept) 591372140.6 591372140.6 591372140.59 591372140.6

property\_typeCasa -30761762.9 -27671041.9 -30468958.28 -30353812.9

`poly(Superficie, 2)1` 91779606.0 59134294.2 89308622.98 88606128.6

`poly(Superficie, 2)2` -85156464.6 -47340141.7 -78480095.63 -76386494.9

rooms\_tot -29503545.3 -17639518.3 -28165170.91 -27587415.6

bathrooms 54106319.6 49748845.1 53703186.13 53654590.6

`poly(Superficie, 2)1:rooms\_tot` 50964122.5 50675645.2 52044514.67 52994058.6

`poly(Superficie, 2)2:rooms\_tot` 19389049.8 2427501.8 18100666.15 16456599.7

`poly(Superficie, 2)1:bathrooms` 10180131.7 36947714.1 11116079.36 10612037.0

`poly(Superficie, 2)2:bathrooms` 5628467.7 -13004017.4 373812.81 0.0

`bathrooms:poly(distancia\_parque, 2)1` 143694.6 615552.4 11128.64 0.0

`bathrooms:poly(distancia\_parque, 2)2` -1261591.7 -1399073.8 -989937.24 -873658.9

`bathrooms:poly(distancia\_comercial, 2)1` -7522750.9 -7529736.1 -7337572.18 -7237881.6

`bathrooms:poly(distancia\_comercial, 2)2` -8690855.3 -8449249.3 -8515198.76 -8446796.2

`bathrooms:poly(distancia\_avenida\_principal, 2)1` 9010529.7 8554239.6 8684796.39 8557105.4

`bathrooms:poly(distancia\_avenida\_principal, 2)2` -2948503.2 -3008093.2 -2752174.13 -2678022.8

`bathrooms:poly(distancia\_universidad, 2)1` -1167006.3 -1371331.4 -942099.92 -846661.4

`bathrooms:poly(distancia\_universidad, 2)2` -4901361.6 -5165554.3 -4726439.11 -4658935.7

OLS RIDGE LASSO EN

RMSE 151007264 151577050 150950062 151591374

