DEPARTAMENTO DE COMPUTACION

Facultad de Ciencias Exactas y Naturales - UBA

Trabajo Práctico III

Cuadrados Mınimos Lineales

Métodos Numéricos

Primer Cuatrimestre (modalidad virtual) - 2020

Integrante	LU	Correo electrónico
Alexis Wolfsdorf	529/17	alexiswolfsdorf@gmail.com
Diego Senarruzza	449/17	diegosenarruzza@gmail.com
Tomás Ortner	147/17	tomyortner@gmail.com

Facultad de Ciencias Exactas y Naturales

Universidad de Buenos Aires

Ciudad Universitaria - (Pabellón I/Planta Baja)

Intendente Güiraldes 2160 - C1428EGA

Ciudad Autónoma de Buenos Aires - Rep. Argentina

Tel/Fax: (54 11) 4576-3359

http://www.fcen.uba.ar

Índice

1 Introducción

1.1 Cuadrados Mínimos

Hacía el año nuevo 1801 cuando el padre *Giuseppe Piazzi* descubría a través de su telescopio, un pequeño punto de luz en mitad de la noche. Volvió a realizar la misma observación a la siguiente noche, solo para darse cuenta de que este se encontraba movido respecto de la primer anotación. *Piazzi* seguiría durante 42 días el movimiento de este as de luz, con un total de 19 observaciones del que posteriormente sería el planeta enano llamado *Ceres*.
Muchos astrónomos de la época intentaron calcular la orbita de este nuevo cuerpo celeste, pero no fue hasta un par de años después que el físico, matemático y astrónomo *Friederich Gauss* fue capaz de construir una primera buena aproximación utilizando solamente 3 coordenadas cuidadosamente seleccionadas. Aunque todavía no había revelado su método, *Gauss* había descubierto el método de cuadrados mínimos (cabe destacar que el francés *Adrien-Marie Legendre* desarrolló este mismo método de manera independiente 4 años antes de la publicación de *Gauss*).

Dado un conjunto de pares ordenados, la técnica de **Cuadrados Mínimos** (en su forma mas simple) intenta determinar una función que logre relacionar al segundo valor de estos pares en función del primero, o por lo menos, aproximar lo más posible a nuestros datos. Dado que buscar esto no provee buenas propiedades, lo que vamos a buscar es una función que minimice el cuadrado de la diferencia de nuestros valores. Es decir que dados $\{(x_1, y_1), (x_2, y_2), ..., (x_n, y_n)\}$, buscamos una función $f(x)$ perteneciente a una familia de funciones F que "mejor aproxime" a los datos, o dicho de manera analítica

$$min_{f \in F} \sum_{i=1}^{n} (f(x_i) - y_i)^2 \tag{1}$$

Esta ecuación es la formulación del problema de cuadrados mínimos para dos dimensiones.

Desde un punto de vista estadístico, un requisito implícito para que funcione el método de cuadrados mínimos es que los errores de cada medida estén distribuidos de forma aleatoria. El teorema de Gauss-Márkov prueba que los estimadores mínimos cuadráticos carecen de sesgo y que el muestreo de datos no tiene que ajustarse, por ejemplo, a una distribución normal.

1.2 Problema y Objetivos

Hoy en día, el problema de determinar el movimiento de los cuerpos celestes es un problema resuelto hace algún tiempo, sin embargo el método descrito por *Gauss* es aplicable en infinidad de otros problemas.

En este trabajo desarrollaremos y evaluaremos algunos modelos de clasificación supervisada, a los cuales entrenaremos mediante un *data set* de datos inmobiliarios de México. El *data set* cuenta con múltiples características de cada inmueble y nuestro objetivo será, tomando algunas de todas ellas, poder generar una ecuación que aproxime al precio real.

Para hacer esto utilizaremos el algoritmo de aprendizaje basado en **Regresión Lineal**, el cual recurre a la técnica de **Cuadrados Mínimos Lineales** para su entrenamiento. Construiremos y compararemos distintos modelos que logren exprimir al máximo este algoritmo, combinando distintas técnicas con el fin de que los datos se logren ajustar de manera correcta.

Es importante que los datos a procesar estén bien escogidos, para que permitan visibilidad en las variables que han de ser resueltas. Para esto, realizaremos un análisis de los datos previo a meternos de lleno con la experimentación, en la cual, entre otras cosas, definiremos si hay datos que nos vayan a dificultar la regresión.

2 Análisis exploratorio de datos

La idea de esta primer sección es la de realizar un análisis exploratorio del *dataset* con el que vamos a trabajar. Es decir, queremos observar a los datos desde una perspectiva macroscópica, con el fin de encontrar relaciones entre los distintos **features** (características) que nos ayuden a tomar decisiones en nuestros modelos.

Tenemos un *dataset* con **240.000** datos y 22 columnas. Algunos de estos datos son numéricos, otros son texto e incluso tenemos booleanos, por lo que debemos discernir cuales de estos podemos utilizar de manera practica en un regresor lineal. Además de eso, no todos los campos están siempre completos, esto hay que tenerlo en cuenta si queremos usar una columna que no todos los inmuebles la tienen definida.

Comencemos observando cuales son nuestros features, y con cuantos de ellos contamos:

Feature	Cantidad no nula
Titulo	234.613
Descripcion	238.381
Tipo de propiedad	239.954
Dirección	186.928
Ciudad	239.628
Provincia	239.845
Antigüedad	196.445
Habitaciones	217.529
Garages	202.235
Baños	213.779
Metros cubiertos	222.600
Metros totales	188.533
ID zona	211.379
Latitud	116.512
Longitud	116.512
Fecha	240.000
¿Gimnasio?	240.000
¿Usos múltiples?	240.000
¿Piscina?	240.000
¿Escuelas cercanas?	240.000
¿Centros comerciales cercanos?	240.000
Precio	240.000

Con esto lo que tenemos es una noción cuantitativa de los datos, lo cual nos da un "pantallazo" acerca de las cosas que tenemos que tener en cuenta. Además de eso, es necesario notar que no todos estos datos nos van a servir (al menos de manera directa) a la hora de armar un modelo de regresión.

2.1 Variables Numéricas vs Variables Categoricas

Antes de continuar con el análisis, detengámonos un momento y separemos los datos que tenemos. Llamamos **variables numéricas** a aquellas cuyo dominio son los números, pueden ser tanto reales como enteros. Además del dominio, estas variables comprenden un concepto de distancia, no es lo mismo tener 1 que tener 0,5 o 3. Por ejemplo un inmueble que tenga $20m^2$ será más grande que uno que tenga $10m^2$ y más chico que uno de $50m^2$.

Por otro lado tenemos a las **variables categóricas**, análogamente estas variables no comprenden una noción de distancia, sino que representan una cualidad de lo que describen. Por ejemplo un inmueble puede ser una *casa*, un *edificio* o una *quinta*, por más que utilicemos etiquetas numéricas para diferenciarlas esto no quiere decir que una sea mayor que la otra, el numero es arbitrario y cambiarlo no implicaría nada en la comprensión del mismo.

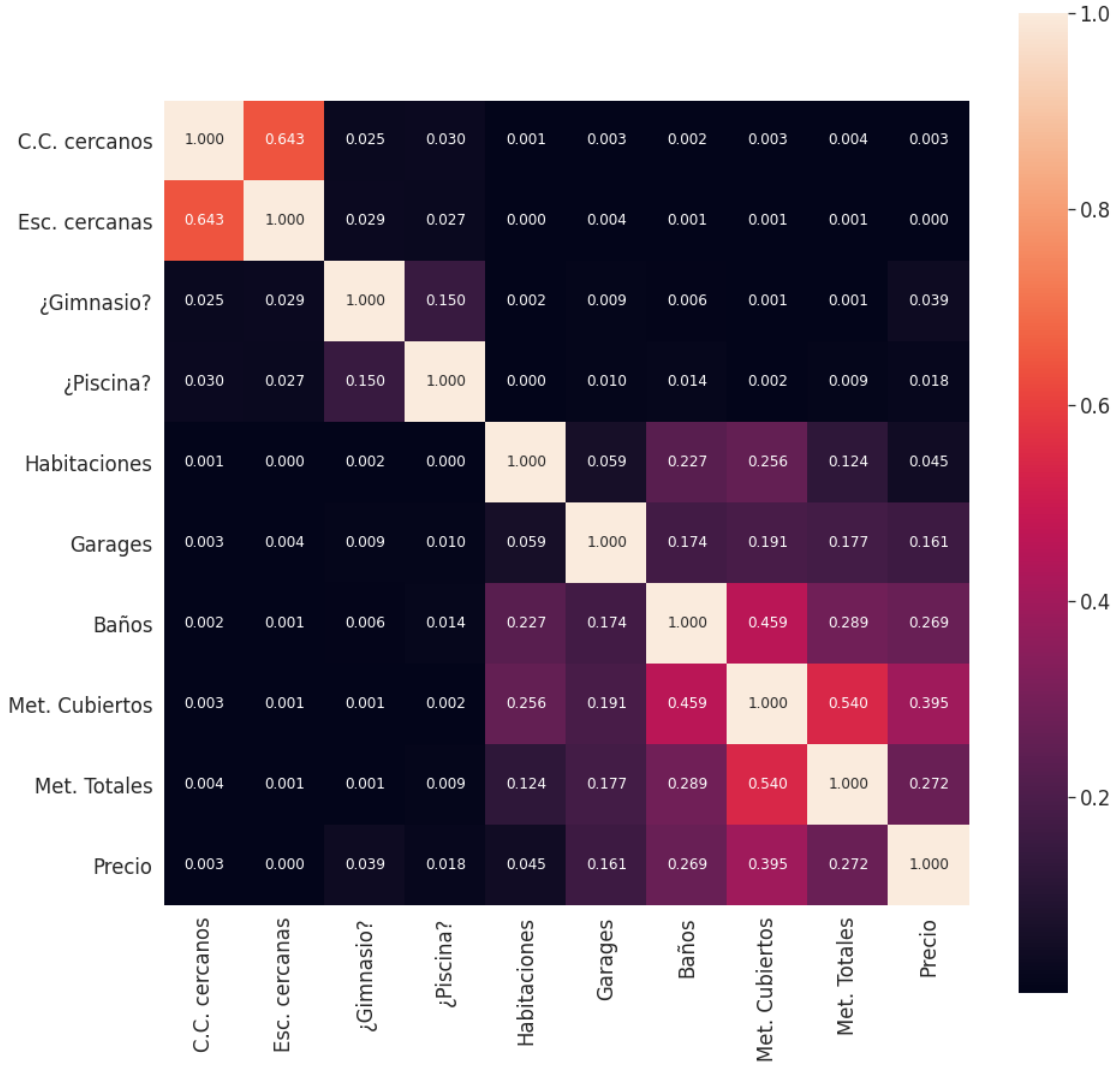
2.2 Matriz de correlación

Una vez que entendemos las diferencias entre los datos que tenemos, podemos avanzar en el proceso de analizarlos. Recordemos lo que pretendemos hacer, queremos construir distintos modelos de regresión lineal que logren predecir o aproximar los precios de los inmuebles, y comparar estas predicciones con los valores reales. Para esto, necesitamos que el precio tenga alguna relación con los demás features para poder hacer la regresión. Dada la cantidad de datos, es imposible que "a ojo" podamos observar cuales de estos pueden llegar a cumplir esas propiedades (si bien intuitivamente podemos saber que por ejemplo, la superficie de una vivienda influye en su precio final).

Utilicemos entonces una **matriz de correlación**. Esta matriz está estrechamente relacionada con una matriz de covarianza (de hecho es una escala de ella).

Es importante mencionar que esto solo lo podemos realizar con variables de tipo numéricas, cosas como la ciudad o el tipo de propiedad (que probablemente tengan gran influencia a la hora de aproximar el precio o los metros de una vivienda) escapan a este análisis. Esos datos no serán desperdiciados, sino que los utilizaremos de otra manera.

Figure 1: Matriz de correlación



Esta matriz no contiene todos los features numéricos, tan solo los que consideramos de mayor relevancia. Para ajustar nuestros modelos de regresión, las características que nos interesan son aquellas que presenten una alta correlación, tal como podemos ver en la matriz las siguientes variables son las que presentan esta propiedad

- Habitaciones
- Garages
- Baños
- Metros cubiertos
- Metros totales
- Precio

Sin embargo no podemos ignorar que existe una fuerte correlación entre las variables **centros comerciales cercanos** y **escuelas cercanas**, lo cual nos lleva a las siguiente preguntas. ¿Existe alguna manera en la que podamos explotar esta correlación? ¿Debemos utilizar únicamente estos datos a la hora de armar un modelo?

2.3 Feature Engineering

Con el fin de aumentar las características disponibles para nuestros modelos, utilizaremos la técnica conocida como **feature engineering** [3]. Esta técnica se describe como el proceso de utilizar conocimiento de la información que ya poseemos (o que provenga de otra fuente), para generar nuevas características. Si bien existen distintas formas de aplicar esto, en particular tendremos en cuenta las siguientes:

- Combinación de características
- Extracción de información de los campos de texto
- Utilización de fuentes externas

Al momento de utilizarlas serán descriptas en los respectivos modelos.

2.4 Segmentación

Si observamos nuevamente nuestro *dataset* y la cantidad de datos que tiene, podemos observar de que los datos podrían presentar una distribución nada homogénea. Es decir que quizás, el precio de un inmueble puede llegar a depender fuertemente de la ciudad en la que se calcula ese precio, o los metros que tiene una casa pueden estar relacionados por el tipo de vivienda.

Bajo esta idea presentamos la **regresión por segmentación**[4]. El sentido de la segmentación es el de dividir a nuestro modelo en segmentos cuyos datos sean lo más homogéneos posibles, y lo más heterogéneos entre dichos segmentos. En los modelos esto se traduce a que, para cada segmento, tendremos una recta de regresión lineal distinta (como una función partida).

$$predict(x) = \begin{cases} segmento_1.predict(x) & \text{if } x \in categoria1 \\ segmento_2.predict(x) & \text{if } x \in categoria2 \end{cases} \tag{2}$$

3 Métricas

Para medir y corroborar nuestros algoritmos no solo utilizaremos las métricas RMSE y RMSLE vistas en clase, si no que además usaremos otras métricas que nos permitan analizar con más profundidad los resultados que nuestro regresor nos está devolviendo

Para las mediciones definiremos dos vectores, el vector X que contiene a los valores reales, y el vector Y que contiene a las predicciones. A continuación, veremos las métricas:

3.1 RMSE

El RMSE (Root Mean Square Error) es el desvío estándar de los errores de las predicciones. Dicho de otra manera, nos dice que tan concentrada está la información alrededor de la mejor recta de aproximación a los valores.

Definiremos $e_i = x_i - y_i$ el error que hay entre los valores reales y las predicciones.

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N} e_i^2} \tag{3}$$

3.2 RMSLE

Cuando la diferencia entre los datos es muy amplia, el RMSE puede arrojar errores demasiados grandes, haciendo que se dificulte el análisis de los mismos. Para achicar la escala del error total, tenemos la alternativa del Root Mean Squared Log Error, que es básicamente aplicar logaritmo a los elementos y luego restarlos. Esto permite que esta métrica solo considere el error relativo entre las predicciones y los valores reales.

$$RMSLE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(log(x_i + 1) - log(y_i + 1))^2} \tag{4}$$

3.3 Coeficiente de determinación

Representa la proporción de la varianza de X y su nomenclatura es R^2. Es un indicador de que tan bueno es el entrenamiento y, por lo tanto, de que tan probable es que el modelo pueda predecir con nuevas muestras, a través de esta proporción de la varianza. Es independiente de la escala de medida de las variables, lo cual es una gran ventaja dado que estas variables son muy grandes y se mueven en intervalos muy amplios. Su mejor resultado es 1, y si el modelo es demasiado malo, puede ser negativo. R^2 es 0 cuando el modelo siempre predice el valor esperado de X, sin importar los valores que le pasen de entrada. Su fórmula es:

$$R^2 = 1 - \frac{\sum_{i=1}^{N}(X_i - Y_i)^2}{\sum_{i=1}^{N}(x_i - \overline{y})^2} \tag{5}$$

3.4 Media del error absoluto

Esta es otra métrica interesante, cuanto mas baja mejor, y además es útil de ver, ya que si es superior al promedio de los valores reales.

4 Experimentación

En esta sección combinaremos las técnicas mencionadas junto con distintas hipótesis para armar modelos de regresión lineal. En particular intentaremos predecir dos cosas: el **precio** de una vivienda, y los **metros totales** en función de distintos features. La idea sera comparar estos modelos (correspondientes a lo que se quiera predecir) e identificar cual es el mejor entre ellos, y la forma que tenemos para comparar es utilizando distintas métricas o *scores* (en la sección anterior explicamos cuales vamos a utilizar).

Con el fin de poder comparar que el grado de generalización de nuestros modelos, separaremos un 2% que no serán tomados en cuenta al momento de entrenar.

4.1 Pre-procesamiento

Antes que nada tenemos que hablar sobre los datos que vamos a usar para cada modelo, en el caso principal donde queremos predecir el precio dado un conjunto de features podemos notar que tenemos un problema: la escala de los datos. Si queremos predecir un precio ¿Cómo sabemos si el resultado es bueno o malo? Un error de $100.000 parece grande, pero para una casa valuada en $3.000.000 no es un error muy grave.
También podemos ver que si seleccionamos los **features** podemos haber elegido columnas del **dataset** que están vacías, como habíamos mencionado, hay muchos datos incompletos, y al ser datos scrapeados del internet de distintas páginas inmobiliarias (O sea datos cargados a mano), hasta los hay mal cargados. Y, no solo eso, sino que hay inmuebles que quizás están bien cargados, solo que son de un conjunto que se aleja mucho de lo que queremos predecir ¿Nos interesa alimentar a nuestro modelo con **outliers**?. La respuesta obvia es que no, los **outliers** darían una información que se aleja mucho del común que queremos predecir y hasta tendrían un gran impacto en nuestra regresión lineal. Para resolver todos estos problemas, hacemos un **pre-procesamiento** de los datos.

Lo que queremos hacer es dejar el **dataset** lo más "prolijo" posible para que nuestro modelo pueda consumir de la forma mas eficiente los datos que le proveemos. Pero esto no es cosa menor, como vimos tenemos mucho por corregir y no parecen nada triviales de hacer; un simple error en esta etapa y podríamos estar quitando del conjunto de entrenamiento datos importantes para el modelo, y lo más peligroso es que estos errores pasarían desapercibidos para el experimento.

4.1.1 Escalado de los datos

Siempre que hablemos de modelar los datos, ya sea con una regresión lineal u otra forma de modelado, hay que tener en cuenta que en general sirve aplicar un escalado/normalización. En nuestro caso es mucho más necesario ya que cuando intentamos predecir precios estos están a una escala de los miles/millones y esto no ayuda.
Por esto decidimos aplicar un escalado a los datos de salida antes de pasarselos a nuestro regresor lineal. Los escalamos sin el uso de la media para que los datos queden entre 0 y 1 (ya que no queremos datos negativos de salida por uso de métricas como RMSLE).

4.1.2 Remoción de datos nulos

Esta sección es la más simple de los **pre-procesamientos**, simplemente después de saber qué features y segmentos vamos a usar en el modelo, removemos siempre que el dato inmobiliario tenga alguna de estas columnas vacía (NaN).

4.1.3 Eliminación de Outliers

Los outliers son algo que no siempre tenemos presentes cuando usamos un datasets pero esto no quita que son un problema bastante grande cuando tenemos datos que pueden tener varianza indefinida o pueden estar mal formados. Este es uno de los casos donde estos dos problemas pueden estar presentes y por eso

eliminaremos los **outliers** cuando lo veamos necesario. Ahora, ¿Cómo sabemos si un dato es **outlier**? Lo hacemos mediante el **Z-Score**. Este score sirve para saber que tanto se aleja el dato que estamos midiendo del común de los mismos. Para explicar esto de una forma mas fácil podemos ver la siguiente imagen:

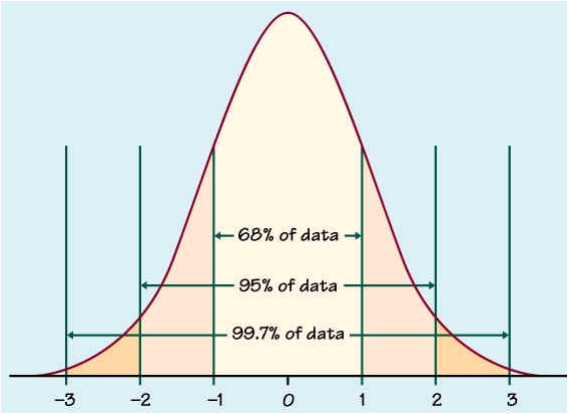


Figure 2: Z-Score

Este score nos da una idea de que tan "**outlier**" es, si el **z-score** tiene modulo mayor a 2 entonces se aleja del 95% de los datos y por lo tanto es posible que este mal cargado o sea un caso muy particular que quizás no nos interese para entrenar a nuestro modelo.

4.2 Primer Modelo

Toca la parte de construir un modelo de regresión que logre explicar (en función de ciertos features), el precio de un inmueble. Para esto debemos partir de algunas hipótesis en función de los datos que hemos analizado previamente.

Aprovechando la buena correlación que tenían algunos de estos features sobre el precio, planteemos las siguientes hipótesis:

1 Serán mas valiosos aquellos inmuebles que cuenten con más metros cuadrados cubiertos en su superficie.

2 De igual manera, los metros cuadrados totales determinan fuertemente el precio del inmueble.

3 Entre más baños tenga un inmueble, más metros tendrá. En consecuencia su precio aumenta en función de esto.

Estos serán los features que describan al regresor lineal de nuestro modelo. Sin embargo, ¿sería correcto suponer que la relación entre estos datos se cumple de manera indistinguible en todo México?, o más en general, ¿en cualquier país?. De manera intuitiva, y por la propia experiencia cotidiana en lo que refiere al ámbito de los inmuebles, sabemos que esto no es así. Por lo tanto, además de apoyarnos en los features mencionados, vamos a apoyarnos en otra hipótesis:

- Sería erróneo suponer que entre los distintos territorios geográficos no existe una variación respecto a los precios de los inmuebles. Es decir que por encima de los metros que pueda tener un inmueble, su precio se relaciona fuertemente con factores regionales, como ser de índole histórico, fronteras con algún país, salida al mar, etc. Estas regiones marcan un límite en los datos que tenemos, separándolos en el espacio de manera categórica.

La idea entonces, es separar a nuestro modelo según estas regiones. Para esto, segmentaremos el modelo según las **provincias**. Cada segmento entrenará un regresor lineal distinto, los cuales en conjunto formarán la ecuación de regresión del modelo general.

Intentaremos medir la eficiencia de nuestro modelo basándonos en algunas métricas. Si bien estas por si solas no describen nada en particular (no hasta que las comparemos con algún otro modelo), particularmente tenemos una noción de lo que esperaríamos encontrar al medirlas.

- **RMSLE**: dada la correlación que tenemos sobre estas variables, esperaríamos ver un valor por lo menos menor al 0.3.
- **R2 Score**: este es quizás una de las métricas más descriptivas, ya que nos devuelve una intuición de la relación entre nuestros datos. Deberíamos llegar a algo superior al 0.5.
- **RMSE**: si bien los precios bastante altos, ese score debería ser relativamente bajo. Lo compararemos con el promedio de los precios en el *dataset*.

Generamos un modelo con estas características. Los resultados se muestran a continuación.

Table 1: Resultados predicción de precios v1

RMSE	RMSLE	R2 Score
783.557.096.384,816	0.346	0.169

Empecemos hablando del *RMSLE*, el cual es un poco superior a 0.3, y aún tenemos margen para mejorar. Por otro lado el cálculo de las covarianzas con *R2 Score* dio bastante peor de lo que esperábamos, por lo que aquí tendremos mucho trabajo por hacer. Por último, el promedio de precios en nuestro *dataset* era de $2.307.149,799$, sin embargo vemos que tenemos un error cuadrático medio inmensamente más alto.

Volveremos a estos resultados más adelante.

4.3 Segundo Modelo

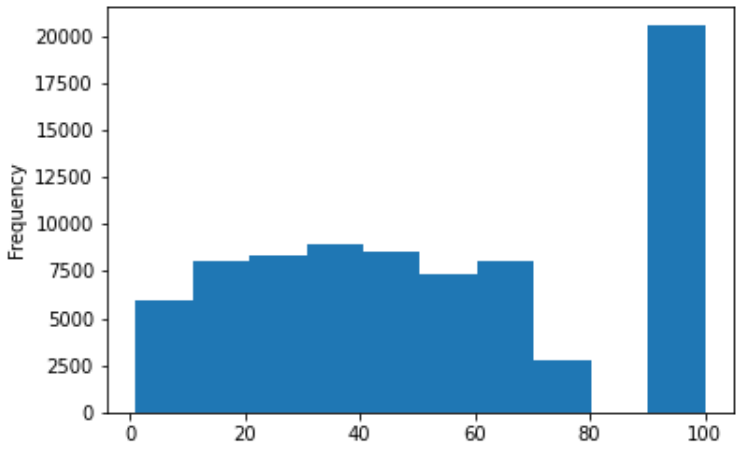
El modelo anterior no da buenos resultados, así que exploraremos otros modelos, intentando utilizar este modelo como base, apoyándonos en que el RMSLE está bastante cerca de nuestro objetivo. Luego aprovecharemos para comparar los otros modelos.

En el modelo anterior nos referimos a la región geográfica de un inmueble, y en base a esto realizamos una segmentación que no aporta ningún valor numérico a los regresores, por lo que podríamos preguntarnos, ¿existe forma de cuantificar el precio según el territorio?

A priori con los datos que tenemos no parece que esto pueda ser siquiera considerado. Sin embargo hemos obtenido un ranking de las ciudades más habitables de México, presentado por el propio gobierno mexicano a partir de encuestas [2] (ordenado de mejor a peor). Este ranking no contempla todas las ciudades que tenemos en el *dataset*, por lo que supondremos que aquellas que no se encuentren es porque no son buenas candidatas para habitar, y les pondremos un puntaje de 100, y a las que pertenezcan al ranking, su posición en el ranking.

Si analizamos la distribución de los datos nos queda lo siguiente:

Figure 3: Distribución de inmuebles según ranking de mejor ciudad



Contamos con una buena distribución de los datos, con lo cual podemos afirmar las siguientes hipótesis:

1 Las ciudades más habitables del país tienden a tener un incremento en el precio de sus viviendas o inmuebles. Dado que la ubicación geográfica influye también, al dividir según las provincias cada segmento queda con el ranking de sus respectivas ciudades, lo cual ayuda a determinar.

2 Dado que la correlación con las variables de metros se mantiene, diremos que el precio de los inmuebles queda determinado en función de los metros cubiertos y los metros totales que tenga en su superficie.

En lineas generales esperamos que nuestro modelo se distinga por tener mejores scores que su versión anterior, ya que lo consideramos un refinamiento de este. Esto quiere decir que debemos notar una buena mejora respecto en las métricas.

Generamos un modelo con estas características. Los resultados se muestran a continuación.

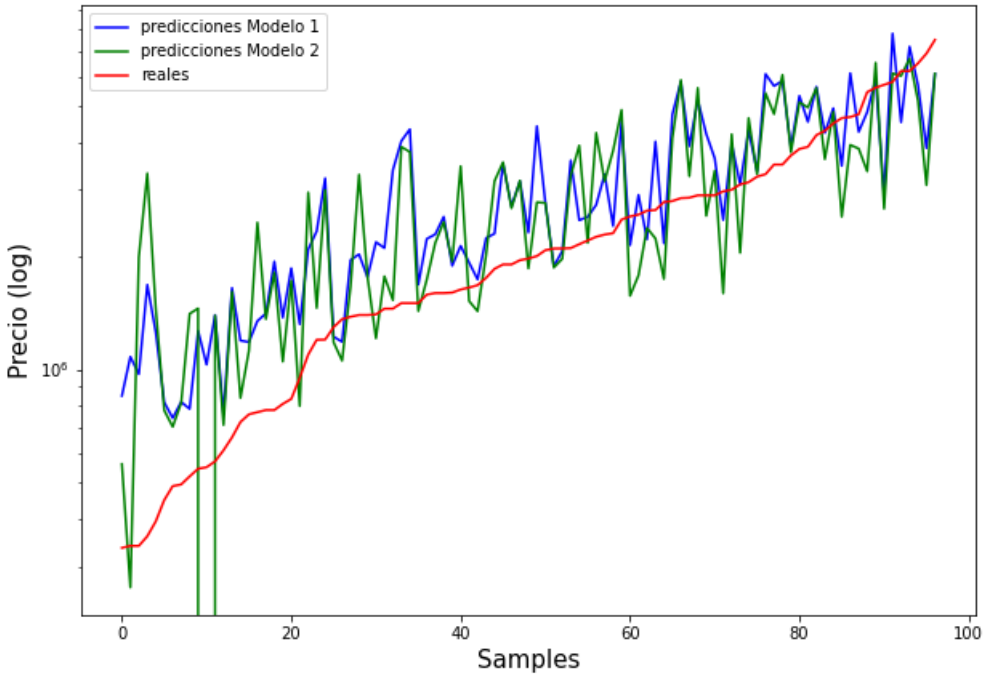
Table 2: Resultados predicción de precios v2

RMSE	RMSLE	R2 Score
1.131.002.886.553,449	0,364	0,033

Al evaluar los resultados obtenidos nos llevamos una ligera decepción. Complejizamos el modelo con la idea de que esto ayudaría en gran medida en el entrenamiento del mismo, sin embargo como podemos ver esto no es necesariamente así. No parece haber una mejora real respecto del primer modelo.

Comparémoslos entonces según su grado de generalización con el subconjunto del *dataset* que no fue utilizado durante los entrenamientos. Ejecutamos la predicción en ambos modelos, posterior al entrenamiento, con un total de 100 *samples*. La cantidad de datos se debe simplemente a que se puedan visualizar de manera correcta, y a que con estos tenemos suficientes para mostrar lo que queríamos. A continuación los resultados de las predicciones, en escala logar ítmica para facilitar su visualización.

Figure 4: Predicción modelo 1 vs modelo 2



Podemos notar una tendencia a similitud entre ambos modelos, ya que ninguno se ajusta perfectamente al gráfico que marca los precios reales. Sin embargo el segundo modelo queda peor parado en este caso, ya que los saltos de diferencia contra los precios reales son muchos mayores que los del primer modelo.

Si tuviésemos que decidir entre alguno de estos dos definitivamente elegiríamos por la primer opción la cual, en apariencia, tiene un deje de mayor simpleza al utilizar datos meramente locales del *dataset*.

4.4 Tercer Modelo

Dado que nuestro primer modelo fue el que mejor nos dio, tomémoslo como punto de comparación para realizar otros ajustes. Mantengamos los mismos features y realicemos una comparación con otra segmentación.

¿Es el territorio geográfico realmente un decisivo cuando hablamos de precios en inmuebles? Viendo los resultados de nuestro primer modelo esto podría parecer así, sin embargo esto no quiere decir que no podamos hallar nuevos factores y llegar, quizás, a mejores resultados.

Previamente utilizando información externa a nuestro *dataset* exploramos la idea de "mejores ciudades", surge entonces la siguiente cuestión. ¿Podríamos nosotros determinar qué es una buena ciudad? Ciertamente no podemos extraer una información similar de los datos que tenemos, y dado que no devolvió los mejores resultados no sería esta la idea. Pero podemos utilizar algún criterio razonable y trabajar con información similar y dividir los inmuebles según otro criterio del estilo "mejor o peor".

Si recordamos la matriz de correlaciones (1), teníamos dos variables que parecen tener una relación estrecha entre si. El hecho de que un inmueble cuente con centros comerciales cerca, parece tener mucho que ver con el hecho de tener escuelas cercanas. Parece bastante lógico esto, ya que zonas *urbanas* suelen cumplir con esta condición (ciudades o pueblos muy habitados).

- Un inmueble que posea escuelas y centros comerciales cercanos verá incrementado su precio, ya que lo más factible es que se encuentre dentro de las ciudades en el ranking que utilizamos anteriormente. Al dividir el modelo con este nuevo dato, generaremos una segmentación en dos espacios, aquellos que consideraremos *urbanos* y los que no.

Analicemos un poco más colateralmente los inmuebles mexicanos. El país se encuentra situado geográficamente al norte de *Centro América* y al sur de *EEUU*, y se ve atravesado por el **Trópico de Cáncer**. El mismo delimita hacia el lado sur una región mas bien tropical, en donde las temperaturas son más elevadas que en el lado norte. No solo eso, al ser un clima tropical, en esta zona hay altas temperaturas durante todo el año. ¿Estarán los precios sujetos a la climatización que posean estos inmuebles?

En un intento por responder a esta pregunta, parece lógico afirmar lo siguiente:

- El trópico de cáncer se encuentra en la latitud $23^o50'$ aproximadamente, por lo cual aquellas viviendas que se encuentren por debajo deberán contar con correspondiente climatización para tener un precio adecuado en el mercado. Apoyándonos en la información que nos brinda el *dataset*, dividiremos a las viviendas entre aquellas que estén por debajo de este trópico y posean piscina (las cuales consideramos como "calurosas"), y aquellas que o no vivan por debajo, o no posean piscina (ya que nos interesa diferenciar específicamente a las calurosas)

Nuevamente compararemos este modelo contra el primero que generamos. Recordemos que la idea detrás de todo esto es lograr encontrar aquellos factores que influyen en el precio de una vivienda y, dado que en principio no conocemos cuales pueden ser estos, intentamos utilizar el razonamiento para deducir cuales podrían ser buscando salirnos un poco de lo que parecería mas obvio.

Ahora si, ejecutamos nuestro tercer modelo y estos fueron los resultados obtenidos.

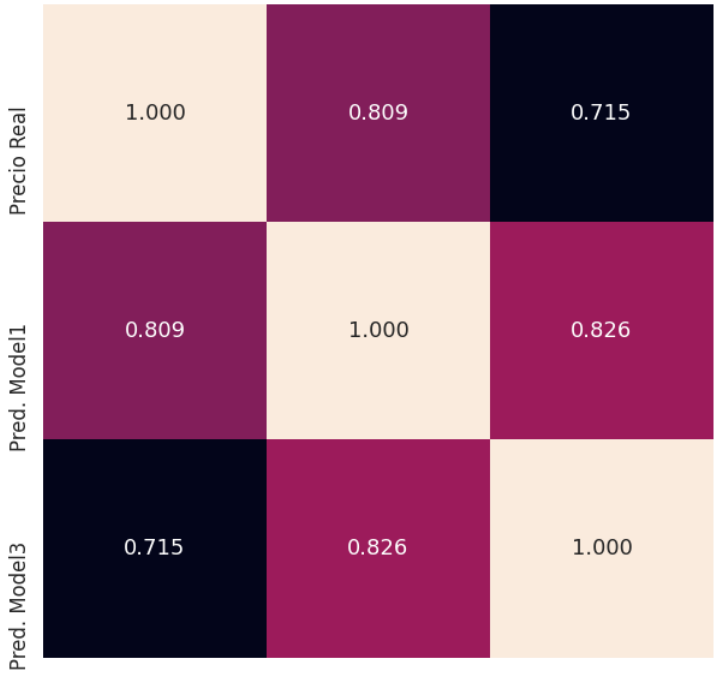
Table 3: Resultados predicción de precios v1

RMSE	RMSLE	R2 Score
2.340.202.131.425,650	0,392	0,265

Los scores parecen seguir sin mejorar respecto del resultado inicial, a pesar de tener (en principio) una noción más básica de como calcular los precios, el primer modelo parece saber generalizar los suficientemente bien.

Realicemos un análisis de generalización comparando la correlación que nos devuelve cada una de las predicciones, vs el precio real. Nuevamente utilizamos ese 2% de datos que nos separamos al comienzo, y que ningún modelo conoce.

Figure 5: Predicción modelo 1 vs Predicción modelo 3



Como intuíamos a partir de las métricas la relación entre estos dos modelos es muy similar, sin embargo se notada la superioridad del primer modelo tanto en relación con los scores (la parte del entrenamiento), como la parte allegada a la generalización de los modelos (darle datos reales y ver que tan bien lo hace).

Si bien pueden parecer modelos muy similares (los 3 que hemos visto), hasta ahora la simplicidad de nuestro primer *aproach* parece ser la que mejor logra resolver este problema, al menos hasta ahora.

4.5 Cuarto Modelo

Al observar los resultados de los modelos anteriores uno podría llegar a pensar que, quizás, estos no cuentan con la información suficiente para resolver una aproximación a los precios de los inmuebles. Mirando en retrospectiva, en general hemos utilizado las variables numéricas como las features de nuestro modelo, y las categóricas para lograr una segmentación. Con lo cual surge la pregunta, ¿Estamos aprovechando toda la información que nos provee el *dataset*? Como respuesta a este interrogante, nos dimos cuenta de que existen ciertos campos que potencialmente podrían tener muy buena información relacionada al precio de una vivienda. Estos son los campos de texto **Titulo y Descripción**, los cuales proveen una numerosa cantidad de información distribuida y condensada en un sistema casi imposible de entender para las computadoras, al cual nosotros (los humanos) llamamos **lenguaje**.

¿Podemos lograr que una computadora o, en este caso, un modelo comprenda este lenguaje tan ajeno a el?. La respuesta es que si, gracias a los avances en computación hoy en día nos es posible desarrollar algoritmos de manera tal que la computadora logre interpretar nuestro lenguaje. Así es, nos hemos embarcado en el apasionante mundo del **Procesamiento del lenguaje natural** [5] (**NLP** por sus siglas en ingles). Como buenos estudiantes de Métodos Numéricos, conocemos distintas técnicas para extraer información de donde parece que no la hay. Lo primero a realizar entonces es un estudio a través de las distintas clases de *NLP* existentes, con el fin de entender cual de todas estas nos puede ayudar a extraer la información necesaria que contenga relación con el dato que queremos predecir (el precio).

Dado que no parece trivial la idea de convertir una o varias oraciones en un simple campo numérico, es que creamos un *pipeline* mediante el cual pasan los distintos párrafos con el fin de transformarlo en un dato que nuestro regresor lineal pueda utilizar como un simple feature más. Es decir que en definitiva, la idea es realizar feature engineering sobre estos campos de texto, separamos la generación de este nuevo dato en 3 etapas.

1. Lo primero que necesitamos hacer, es pensar de que manera nos conviene procesar estos campos de texto. La decisión más trivial quizás (aunque muy usada) es la de realizar un *Bag of Words* [6] (**BoW**) tomando las descripciones de los inmuebles como el corpus de texto y, de esta forma, convertir el texto en una representación vectorial (One-Hot Encoding) dada por una matriz rala, en donde cada uno de los índices es la representación de un token en el corpus.

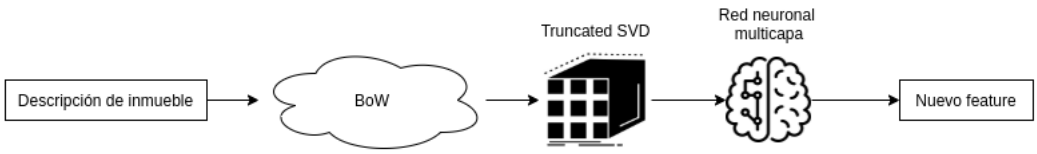
2. Como siguiente paso podríamos simplemente cuantificar estos datos, es decir, transformarlos a números mediante alguna operación algebraica... pero podemos hacer un poco mejor. Podemos extraer las componentes principales de este *BoW* (**CountVectorized**), con lo cual estaríamos mirando tan solo aquellos *tokens* que más hacen variar al dato que queremos predecir. Para realizar esto nos apoyamos en la clase *TruncatedSVD* que nos brinda *SkLearn*.

3. Con esto estamos listos para realizar una última transformación. Decidimos usar una *Red Neuronal Multicapa* [7] que pueda aprender como transformar la matriz de componentes principales, en un único dato que será finalmente aquel que nuestro regresor lineal usará.

Esta idea parece ser muy interesante, sin embargo todo este **pipeline** tiene muchas variables y parámetros involucrados. ¿Cuantos tokens tiene que usar el BoW?, ¿Con cuantas componentes principales nos quedamos? ¿Que tamaño deben tener los layers de la red neuronal?, ¿Función de aprendizaje?. En definitiva, se crea una combinatoria no demasiado fácil de manejar. Para realizar la búsqueda de estos **hiper-parámetros** nos apoyamos fuertemente en las clases y métodos que provee el proyecto *SkLearn*. Gracias a esto pudimos realizar un *GridSearch* mediante, el cual obtuvimos los mejores *hiper-parámetros* para generar este nuevo feature.

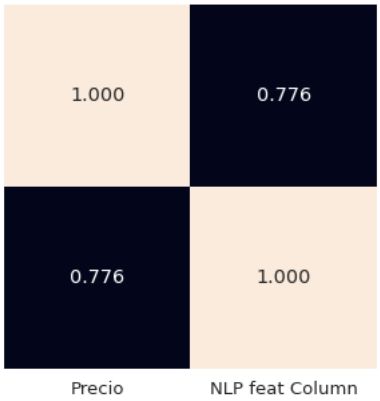
En resumen, el pipeline por el cual atraviesa el texto puede entenderse con el siguiente esquema:

Figure 6: Procesamiento del Pipeline NLP



Con el fin de evidenciar la ayuda que nos va a proveer este nuevo feature, lo que hicimos fue generarlo para todo el *dataset* original. Así como previamente comparamos los features originales para ver su relación con el precio (1), veamos la relación que este nuevo dato tiene para con el precio de un inmueble.

Figure 7: Correlación precio-NLP feature



Como podemos notar se consigue un muy buen resultado con una correlación mayor a cualquier otro feature antes visto. Esto debe de ser ya que normalmente las descripciones de los inmuebles están hechas para darle la mayor información a los posibles compradores de por qué la casa vale lo que vale. Gracias a este pre-procesamiento de los campos de texto se pudo extraer parte de esta información que de otra forma sería imposible procesarla.

Dado que parece ser que existe una muy buena correlación entre el precio un inmueble y este feature, podemos conjeturar la siguiente hipótesis.

- Los títulos y descripciones de los artículos contienen información de suma importancia para el comprador. Estos textos suelen ser cortos y limitados en caracteres, por lo que deben condensar una gran cantidad de información acerca del inmueble en resumidas palabras (obviando muchas veces artículos o conectores de la lengua). Dada esta condensación de palabras, es posible y factible extraer información de tipo numérica, que tenga alta relación con el precio. Es decir que podemos predecir el precio de un inmueble en función de los campos de texto que acompañan a la publicación de venta del mismo.

4.5.1 Aplicando NLP

Dado la disimilitud entre estos modelos con los primeros, vamos a realizar una comparativa entre dos distintas propuestas de modelos que combinan esta técnica de *NLP*. Los modelos propuestos a continuación son quizás demasiado distintos entre si, o comprenden un espacio de búsqueda distinto cuanto menos. No será la finalidad de este experimento la de encontrar que modelo es "mejor", sino la de comparar el nivel de generalización que ambos poseen.

4.5.2 NLP Modelo 1

La primer propuesta se basa en el buen rendimiento que parece tener el primer modelo que generamos. Bajo las mismas hipótesis, sostenemos que el precio de un inmueble está relacionado con la cantidad de *metros cubiertos* y *metros totales* que este pose. Además. La cantidad de baños se relaciona con los metros de una vivienda, por lo tanto también influirán en el precio final.

Vamos a continuar con la idea de que la región delimita el precio, por lo tanto vamos a segmentar según la provincia en la que cada inmueble se encuentre.

4.5.3 NLP Modelo 2

La segunda propuesta se basa en conceptualizar otra noción de calculo de precio, otro punto de vista. Para empezar mantendremos la idea de que los metros que un inmueble posee hacen que crezca el precio, sin embargo, dada la correlación que tienen las dos variables de métodos, quizás estemos agregando ruido al cálculo si utilizamos ambas. Por este motivo solo tendremos en cuenta los *metros cubiertos* (que poseen una mayor correlación). Volvemos nuevamente con la noción de *mejor ciudad* de la mano del ranking que habíamos obtenido previamente, dado que el fracaso en su primer uso quizás pudo deberse a otros factores.

Por último probaremos con una segmentación un poco mas acotada. Lo más seguro es que las mejores ciudades que se encuentren en el ranking sean capitales de las provincias mexicanas, y, dada la cantidad de edificios que suelen encontrarse en las ciudades, no es muy seguro que los inmuebles allí cuenten con muchos baños. Dividiremos entonces al *dataset* acorde a la cantidad de baños que posean los inmuebles.

Con el fin de apoyar al ranking de ciudades dividiremos los datos según si una casa se considera *urbana* o no, tal cual lo utilizamos en el modelo 3.

Como hemos mencionado anteriormente, la finalidad de este experimento no es la de comparar los scores obtenidos sino la de su grado de generalización. Sin embargo, no resta en nada observar el resultado de las métricas de cada uno. Cambiamos el *RMSE* (que veníamos usando) por la media del error absoluto, simplemente porque nos devuelve un valor quizás mas tangible.

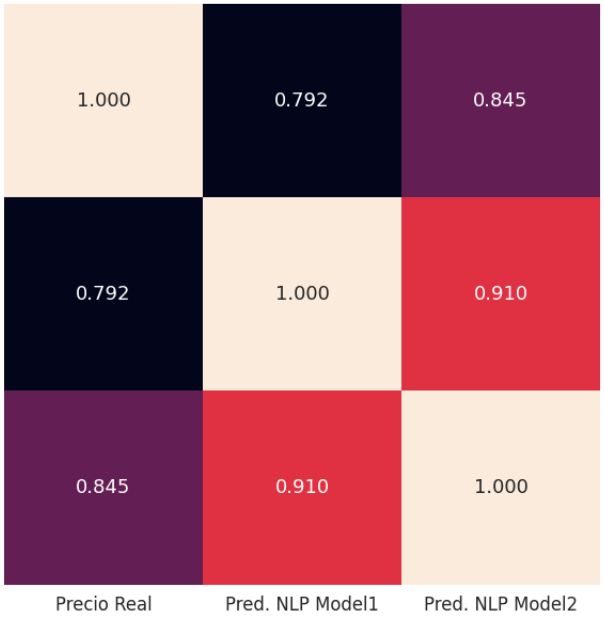
Dado que el *NLP Modelo 1* extiende con este nuevo feature al modelo que venia funcionando mejor (el primero de todos) esperaríamos que sepa generalizar mucho mejor los resultados que su rival.

Table 4: Resultados predicción de precios, NLP Model1 vs NLP Model2

	RMSLE	R2 Score	Mean Absolute Error
NLP Modelo 1	0,192	0,189	855.486,454
NLP Modelo 2	0,383	0,317	765.342,483

El segundo modelo devuelve una mayor correlación con los datos predichos. Analicemos la correlación que se obtiene al predecir datos ajenos al entrenamiento:

Figure 8: Correlación Precio Real vs Precio-NLPModel1 vs Precio-NLPModel2



Los resultados no parecen arrojarnos demasiada luz a la hora de determinar cual de estos dos modelos generaliza mejor los resultados. Pero si nos sirven para refutar la idea que teníamos acerca de que, una extensión del primero de todos los modelos daría un mejor resultado. Si bien la diferencia es muy chica, existe y parece darnos un mejor resultado.

Contrario a lo que nos venía ocurriendo, parece que una combinación de features pensada desde otra perspectiva, hace que podamos determinar el precio con un poco más de fiabilidad. En definitiva, el *NLP Modelo 2* nos está dando un mejor nivel de generalización.

5 Conclusiones

A lo largo de este trabajo hemos recorrido una serie de modelos con el fin de intentar predecir, a partir de un *dataset*, los precios de los inmuebles en México. Dado que en el mundo del mercado inmobiliario las ecuaciones que rigen estos precios pueden ser muy complejas, determinarlo utilizando regresiones lineales no es una tarea sencilla. Experimentamos con distintos modelos bajo la idea de ir variando la definición objetiva que se tiene del valor de una vivienda, es decir que para armar cada uno de ellos tuvimos que pensar desde diversos puntos de vista.

Utilizamos distintas técnicas para intentar ajustar de la mejor manera posible los datos que teníamos, desde utilizar segmentación hasta realizar distintos tipos de *feature engineering* (algunos mucho más complejos que otros).

De los experimentos y comparaciones hechas, podemos extraer una conclusión un tanto sencilla. Un modelo óptimo no es necesariamente un modelo robusto, ya que quizás peca de complejo (o simple) a la hora de generalizar datos que no conoce. También puede ocurrir, que los datos utilizados contengan más ruido del que uno cree, o que por el contrario esto no pase pero que no sean realmente un conjunto representativo.

Sin embargo, si podemos asegurar algunas cosas. Se hace evidente en los distintos experimentos que los metros que una vivienda posee se encuentran fuertemente relacionados con su precio. También parecen influir en gran medida el hecho de que la vivienda se encuentre en una ciudad o localidad urbanizada. Sumado a esto parece ser que los anuncios publicados para la venta de inmuebles, contienen una buena cantidad de información condensada en su corpus. En los modelos donde utilizamos esta información propusimos una forma en la que se puede utilizar, pero no es la única.

5.1 Reflexión y trabajo a futuro

La finalidad del trabajo era la de armar y comparar distintos modelos que logren determinar el precio de una vivienda, pero al comparar los scores tampoco nos encontramos con una diferencia demasiado abrumadora. Como hemos podido ver en la comparativa del Modelo 2 vs Modelo 1 (4) la distribución de las predicciones no parece ajustarse demasiado bien al precio real, lo que nos da una noción de que queda mucho para mejorar.

Hoy en día existen múltiples técnicas que nos permiten resolver el problema que se presenta en este trabajo (entre ellas las redes neuronales). Una idea a futuro podría ser la de comparar los resultados aquí obtenidos contra resultados de otros métodos. También podría extenderse la gama de modelos presentados, y aplicar todas las combinaciones posibles en un intento de *GridSearch*.

6 Referencias

[1] El problema de determinación de las órbitas
https://biblioteca.unirioja.es/tfe_e/TFE004266.pdf

[2] Ranking de mejores ciudades para vivir en México
https://gabinete.mx/index.php/es/ciudades-mas-habitables-2019

[3] Feature Engineering
https://en.wikipedia.org/wiki/Feature_engineering

[4] Regresión Segmentada
https://es.wikipedia.org/wiki/Regresion_segmentada

[5] Natural Language Processing
https://en.wikipedia.org/wiki/Natural_language_processing

[6] Bag of Words
https://en.wikipedia.org/wiki/Bag-of-words_model

[7] Red Neuronal Multicapa
https://en.wikipedia.org/wiki/Multidimensional_network