

MEMORIA DEL TRABAJO FINAL:

ANÁLISIS Y PREDICCIONES DEL MERCADO INMOBILIARIO ESTADOUNIDENSE



ANTONI BALDÓ Y CARLOS RODRÍGUEZ

ÍNDICE:

1. Nuestra visión.
2. EDA:
 - 2.1. Filas y columnas.
 - 2.2. Análisis.
 - 2.3. Correlaciones.
 - 2.4. Transformaciones y visualizaciones.
3. Red Neuronal:
 - 3.1. Filtro de precios.
 - 3.2. Precisión y pérdidas.
4. Conclusión.
5. Bibliografía.

1. NUESTRA VISIÓN:

La situación económica actual deja un mercado inmobiliario totalmente impredecible debido a la crisis del COVID o de la guerra entre otras razones. Ahora es cuando realmente vamos a notar los verdaderos efectos de una crisis así, por ejemplo ya llevamos desde que 'acabó' la pandemia con la inflación de prácticamente todas las monedas subiendo sin parar, en Italia han llegado a unos puntos de inflación de más del 11%. Lo que desencadena una terrible recesión y una esperanza para todo 2023 y 2024 de un mercado bajista que dejará sin oportunidades a inversores menos capitalizados.

La Reserva Federal de los EE.UU. y la Unión Europea ya han anunciado medidas contra esto en lo que va de año como la subida de los tipos de interés que seguirán subiendo a lo largo del 2023 hasta el 4.5% en EE.UU., y el 2.5% que anunciaba el banco central europeo.

Pero realmente, ¿qué significa esto?, esto quiere decir en pocas palabras que todas las potencias económicas mundiales van alineados con el mismo mensaje, un mensaje de enfriamiento generalizado de la economía.



¿Y qué consecuencias tendrá esto sobre el mercado inmobiliario?. Esto siguiendo los ejemplos del IPC, carburantes o gas, entre otros, subirá todo de precio, las hipotecas, los intereses de las hipotecas etc. En cambio los alquileres bajarán, debido a la poca demanda que hay ya que no hay dinero para alquilar, pero si hay una gran cantidad de oferta.

Average Asking Rent for Multifamily Drops \$9 Nationally in November, Largest One-Month Decline in a Decade, Says Yardi Matrix

Posted on December 13, 2022 by Jeff Shaw in Features, Multifamily



Viendo este panorama nosotros creemos que podemos aportar algo para que las decisiones de comprar una casa e hipotecarse, ya sea para invertir en inmobiliaria o para entrar a vivir, sean mucho más fáciles. Esto lo haremos mediante un buen análisis EDA de nuestro dataset, donde sacaremos las primeras conclusiones. Dado que el mercado inmobiliario global lo marca Estados Unidos hemos decidido tener datos que provienen de dicho país, en este dataset tenemos columnas como el precio de la vivienda, la ciudad y el estado, el año de construcción y el de renovación. Además de esto tenemos, como comentábamos antes las diferentes características, habitaciones, baños, plantas, vistas o los pies cuadrados tanto de la casa, como de la parcela y el sótano. Después del análisis comenzaremos con una red neuronal a filtrar el valor de las viviendas dependiendo de las diferentes características comentadas anteriormente que contenga la misma. Dado que nuestra primera idea de trabajo no llegó a ejecutarse por las complicaciones con el dataset, decidimos centrarnos en algo que nos gustara pero a la vez que fuera retador, que fuera algo difícil de

entender. En este caso elegimos el mercado inmobiliario ya que hacía escasos días del anuncio de la Reserva Federal de los EE.UU. comentado anteriormente. Y dado que es un tema que en algún momento va a ser necesario conocer, decidimos investigar y sacar un dataset que más o menos se ajuste a nuestras ideas.

2. EDA:

2.1. FILAS Y COLUMNAS:

Este conjunto de datos contiene precios de venta de casas para el condado de King, que incluye a Seattle. Incluye viviendas vendidas en 2014 y contiene 4600 filas y 18 columnas.

A continuación haremos una descripción de cada una de las columnas:

- Date: fecha en que se vendió la casa
- Price: precio de la casa
- Bedrooms: Número de Dormitorios/Casa
- Bathrooms: Número de baños/dormitorios
- sqft_living: pies cuadrados de la casa
- sqft_lot: pies cuadrados del lote
- Floors :Total de pisos (niveles) en casa
- Waterfront :Casa que tiene vista a un frente al mar
- View: Ha sido visto
- Condición: Qué tan buena es la condición en general
- sqft_above: pies cuadrados de casa aparte del sótano
- sqft_basement: pies cuadrados del sótano
- yr_built :Año de construcción

- yr_renovated: año en que se renovó la casa
- Street: calle donde se encuentra la casa
- City: ciudad donde se encuentra la casa
- Statezip: código del estado
- Country: país donde se encuentra la casa

2.2. ANÁLISIS:

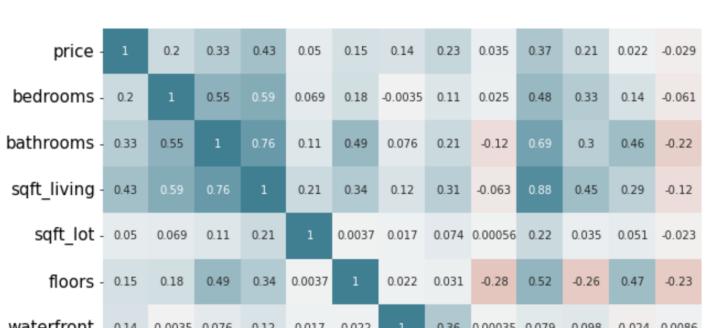
Ahora que sabemos de qué va nuestro dataset y su tamaño, es hora de hacer un análisis preliminar antes de meternos con las transformaciones. Primero observamos que nuestro dataset no contiene ningún valor nulo, por tanto todo el dataset será útil y los filtros saldrán mejor. A continuación miramos qué tipos de datos tenemos, los cuales son todos int o float, menos las últimas columnas que son object ya que contienen el nombre del estado, ciudad y país. Por último hacemos una descripción rápida de las diferentes columnas numéricas para saber más o menos el rango de valores de cada una.

| data.isnull().sum() | data.dtypes | data.describe() |
|---------------------|---------------|-----------------|
| date 0 | date | object |
| price 0 | price | float64 |
| bedrooms 0 | bedrooms | float64 |
| bathrooms 0 | bathrooms | float64 |
| sqft_living 0 | sqft_living | int64 |
| sqft_lot 0 | sqft_lot | int64 |
| floors 0 | floors | float64 |
| waterfront 0 | waterfront | int64 |
| view 0 | view | int64 |
| condition 0 | condition | int64 |
| sqft_above 0 | sqft_above | int64 |
| sqft_basement 0 | sqft_basement | int64 |
| yr_built 0 | yr_built | int64 |
| yr_renovated 0 | yr_renovated | int64 |
| street 0 | street | object |
| city 0 | city | object |
| statezip 0 | statezip | object |
| country 0 | country | object |
| dtype: int64 | dtype: object | |

2.3.

CORRELACIONES:

La matriz de correlación nos muestra que las variables más



correlacionadas son los pies cuadrados de la casa con los pies cuadrados de la casa sin el sótano, lo que nos sorprende especialmente, en lo que sí nos hemos fijado es que los baños y habitaciones, están bastante correlacionados con los pies cuadrados en general. Además también vemos que la variable del año de construcción tiene valores de correlación bajos excepto con los baños, las plantas, los pies cuadrados, lo cual es interesante ya que puede ser que dependiendo del año la moda en la construcción de las casas fuera diferente.

También vemos que la variable más correlacionada con todas las demás es el precio de la vivienda y la que menos es la de año de renovación, lo cual es nos dice que la mayoría de reformas se debieron hacer por motivos aleatorios ya que si no estaría más relacionada por ejemplo con el año de construcción o con el su precio.

En conclusión creemos que el dataset está bastante preparado para hacer una red neuronal como nosotros la queremos ya que el precio es la variable más correlacionada con las demás, es decir, con las características que marcaran el precio de las viviendas.

2.4. TRANSFORMACIONES Y VISUALIZACIONES:

Lo primero que observamos es que las medidas no están en metros cuadrados, si no que están en pies, por tanto lo primero que haremos será hacer nuevas columnas con todo pasado a metros cuadrados.

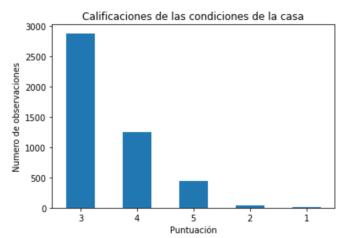
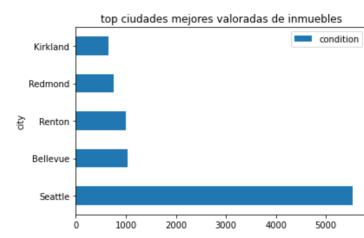
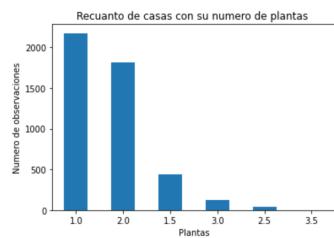
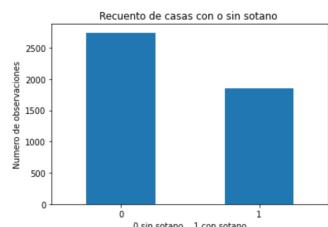
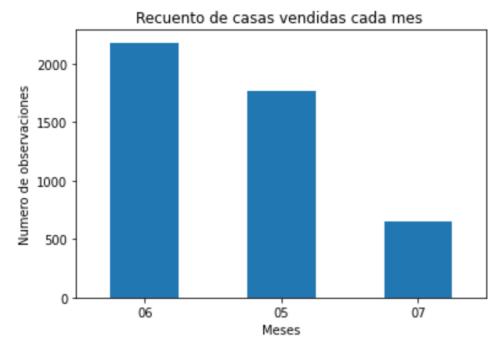
A continuación procedemos a separar la fecha de venta de la casa en año, mes, día y hora, convirtiéndose en 4 columnas más, que luego nos servirán para hacer algunas gráficas. En cuanto al número que sale en el precio

| | |
|---------------|---------|
| date | object |
| price | float64 |
| bedrooms | float64 |
| bathrooms | float64 |
| sqft_living | int64 |
| sqft_lot | int64 |
| floors | float64 |
| waterfront | int64 |
| view | int64 |
| condition | int64 |
| sqft_above | int64 |
| sqft_basement | int64 |
| yr_built | int64 |
| yr_renovated | int64 |
| street | object |
| city | object |
| statezip | object |
| country | object |
| m2_casa | float64 |
| m2_lote | float64 |
| m2_above | float64 |
| m2_basement | float64 |
| año | object |
| mes | object |
| dia | object |
| hora | object |
| dtype: object | |

simplemente está como un número no decimal, pero como el modelo y las gráficas lo leen bien, de momento no lo tocaremos.

Una vez hechas las transformaciones pasamos a analizar mejor lo que tenemos delante mediante algunas gráficas para entender mejor las relaciones y mostrar algo más de información.

Haciendo algunas observaciones vemos que las casas solo se vendieron en los meses mayo, junio y julio de 2014. También vemos algunas características de las casas que luego nos ayudaran a entrenar el modelo, como el número de plantas, el recuento de las que tienen sótano y las que no, las ciudades mejor valoradas y la calificación de las mismas, esta última nos servirá de gran ayuda a la hora de entrenar el modelo.



De este EDA sacamos como conclusión que todas estas gráficas, además de las tablas hechas en el documento .ipynb, nos han ayudado a entender mejor el dataset sirviéndose de gran ayuda para luego inicializar la red neuronal.

Y hablando más en profundidad del dataset hemos aprendido que es un conjunto de datos bastante atípico pero que muestra muy bien lo que queremos ya que contiene casas de todos los tipos, con precios totalmente diferentes, lo que al principio nos pareció difícil de entender ya que por ejemplo las casas con vistas al mar en la media valían

menos que las que no tenían, o las que tenían sótano valían menos que las que no lo tenían. Entonces al hacer un análisis bastante más exhaustivo nos dimos cuenta que hacer tablas así, generalizando para solamente una característica de la casa no servía de nada ya que al haber muchas más características de la misma casa, una sola no marcaba el precio final de la casa.

| | price | | price | | price |
|------------|--------------|--------|--------------|-----------|--------------|
| waterfront | | sotano | | renovated | |
| 0 | 2.491126e+09 | 0 | 1.374300e+09 | 0 | 1.546028e+09 |
| 1 | 4.790350e+07 | 1 | 1.164730e+09 | 1 | 9.930020e+08 |

3. RED NEURONAL:

3.1. FILTRO DE PRECIOS:

En este caso lo que haremos como he explicado antes será imprimir un valor aproximado de la vivienda dependiendo de sus diferentes características, estas influirán en el precio dependiendo de su correlación con el mismo, estas correlaciones serán las mismas que hemos estado analizando anteriormente.

Dado que nuestro objetivo es desarrollar un modelo que tenga la capacidad de predecir el valor de las viviendas, dividiremos el conjunto de datos en las características y la variable objetivo, y las almacenaremos en las variables ‘features’, que serán las características de las casas las cuales nosotros hemos considerado mirando la gráfica de correlación que tenía más sentido poner, y ‘prices’, que serán los precios de las viviendas, respectivamente.

A continuación deberemos medir la calidad que tendrá el modelo, como hemos visto en clase hay muchos parámetros para medir la calidad de un posible modelo como algún tipo de métrica de rendimiento, sea calculando algún tipo de error, la corrección del ajuste, o alguna otra medición útil. En nuestro caso utilizaremos el coeficiente de determinación, R2, para cuantificar el rendimiento del modelo. El coeficiente de determinación de un modelo es una estadística útil en los análisis de regresión, ya que describe a menudo como es de bueno el modelo haciendo predicciones.

A continuación separamos el dataset en train y test, y observamos si lo hemos hecho correctamente.

Training and testing split was successful.

Ahora entrenaremos el modelo mediante un algoritmo de árbol de decisión, para asegurar que estamos produciendo un buen modelo, entrenaremos el modelo usando el parámetro ‘max_depth’, es el número de preguntas que el algoritmo de del árbol de decisión puede hacer acerca de los datos antes de hacer una predicción.

Adicionalmente, encontraremos que utiliza ShuffleSplit() para una forma alternativa de validación cruzada. La implementación ShuffleSplit() que se puede ver abajo creará 10 (‘n_splits’) sets mezclados, y por cada

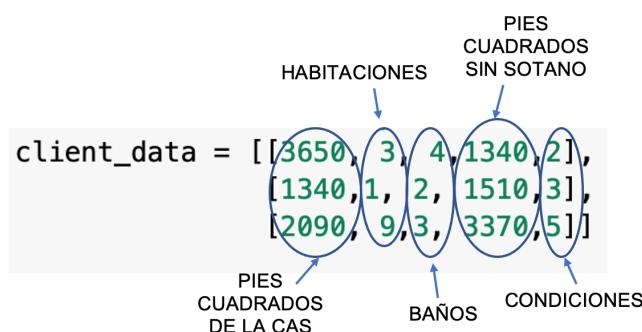
mezcla, el 20% ('test_size') de los datos se usará como el set de validación.

Una vez el modelo está entrenado, averiguamos cual es la profundidad máxima que devuelve el modelo.

Parameter 'max_depth' is 3 for the optimal model.

Como observamos la profundidad máxima es 3 para nuestro modelo.

Por último mediante las matrices donde les pondremos los diferentes valores para las características, imprimimos el valor de las diferentes casa:



Precio de venta previsto para el cliente 1º casa: \$867,331.24

Precio de venta previsto para el cliente 2º casa: \$411,152.86

Precio de venta previsto para el cliente 3º casa: \$518,096.22

En conclusión este modelo nos ha parecido muy interesante ya que no era como los demás si no que se basaba en un árbol de decisión para entrenar al modelo y la predicción, observando y comparando, salía bastante parecida al precio real.

Al ser un modelo relativamente sencillo podría ser de gran ayuda a aquellos que estén buscando una casa con características específicas,

ya que también es posible re-entrenarlo gracias a que está hecho a partir del árbol de decisión que comentábamos anteriormente .

3.2. PRECISIÓN Y PÉRDIDAS:

Una vez hecho el filtro para los precios de las viviendas, haremos un pequeño análisis de la precisión y las pérdidas, para ver así si el dataset estaría preparado para hacer predicciones del precio de la vivienda y poder mirar el posible precio de las casas en futuro con las diferentes características que hemos analizado anteriormente.

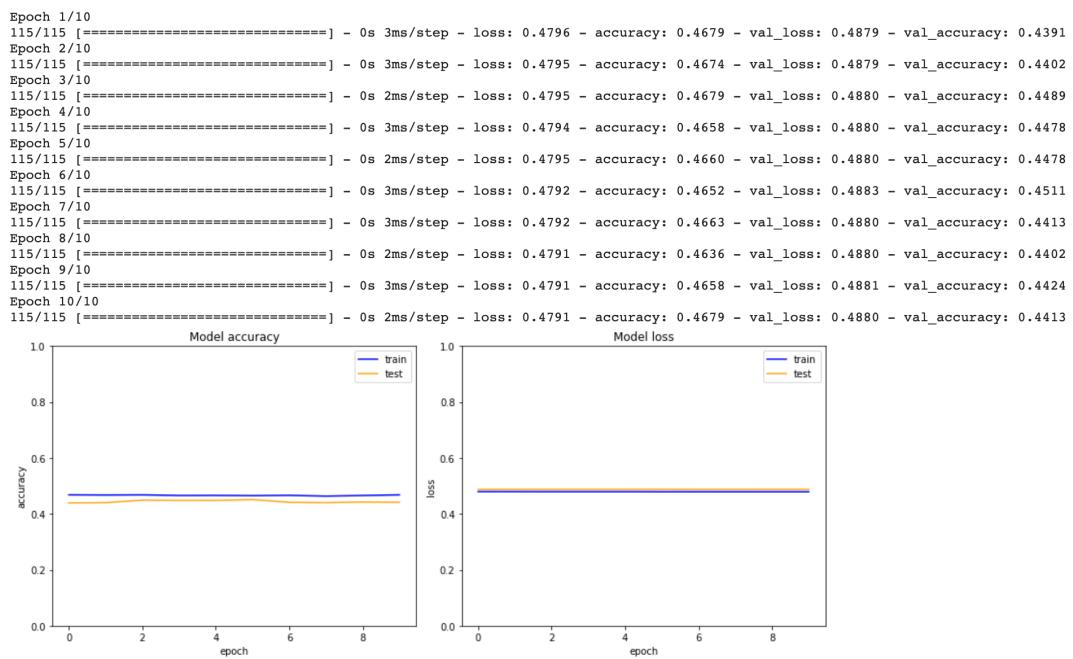
Para validar el modelo es necesario asegurar que éste funcionará correctamente para los datos reales, de manera que capten el problema y generalicen las predicciones correctamente, por esto diseñaremos con Keras, un modelo adaptado a nuestro problema. El valor de los hiperparámetros dependerá en gran medida del problema estudiado.

Para esto utilizaremos 4 capas ocultas con 25, 50, 10 y 20 neuronas sucesivamente, además de la función relu para las capas intermedias, que ayuda a las redes neuronales a formar modelos de aprendizaje profundo, y la función softmax para la capa de salida.

A continuación compilamos el modelo con una función de pérdida binary crossentropy, la cual penaliza a las predicciones correctas, y la función de optimización Adam, que acelerara el proceso de optimización. Con una velocidad de aprendizaje de 0.2, se ha comprobado que esta será la más óptima para este apartado, ya que si se agranda esta no podría acertar el mínimo, pero si se disminuye el aprendizaje tardará demasiado. Y por último, la métrica accuracy que nos dirá el los casos que el modelo ha acertado.

| Layer (type) | Output Shape | Param # |
|------------------|--------------|---------|
| dense_28 (Dense) | (None, 25) | 300 |
| dense_29 (Dense) | (None, 50) | 1300 |
| dense_30 (Dense) | (None, 10) | 510 |
| dense_31 (Dense) | (None, 20) | 220 |
| dense_32 (Dense) | (None, 1741) | 36561 |

Ahora al ser simplemente un modelo preliminar, le ponemos que entrene el modelo en 10 épocas y que visualice la precisión y la pérdida, para train y test.



Primeramente observamos que la arquitectura es buena ya que mantiene prácticamente durante todas las instancias el mismo patrón, lo que nos dice que tanto las capas ocultas como las neuronas están bien definidas, y no hay ni sobreentrenamiento ni falta de él mismo. Además la velocidad de aprendizaje también se muestra constante, sin oscilaciones, por lo tanto a falta de mirar el resto del modelaje, concluyamos que está bien.

Por último, analizando las diferencias entre categorical_crossentropy y binary_crossentropy, nos quedariamos con esta última ya que es la más adecuada para problemas de este tipo.

En conclusión creemos que si seria un buen conjunto de datos para hacer un buen modelo que realiza la predicción del precio de las viviendas.

4. CONCLUSIÓN:

Dado que hemos ido haciendo diferentes conclusiones a lo largo del documento en los diferentes apartados aquí nos centraremos en los retos que ha supuesto hacer este proyecto y para que creemos que será útil.

Primeramente queremos recalcar que todo el proceso ha sido una muy buena experiencia, sobre todo por toda la recogida de información y la posterior puesta en común de esta para conectarlo todo en un mismo trabajo. Ha sido muy nutritivo y hemos aprendido mucho más, de lo que ya lo habíamos hecho.

Por último creemos que hemos creado una metodología bastante útil ya que esta sirve para, por una parte segmentar por características la casa que buscas y por otra parte para concluir su precio, además este proyecto deja la puerta abierta a nuevos experimentos con el mismo dataset, gracias a esa segunda parte del apartado de la red neuronal, que ratifica la calidad del mismo.

5. BIBLIOGRAFÍA:

<https://elpais.com/economia/negocios/2022-11-13/el-bce-ante-el riesgo-de-recesion.html>

<https://www.eleconomista.es/economia/noticias/12110411/01/23.html>

<https://www.bbc.com/mundo/noticias-61571916>

<https://es.euronews.com/2022/10/27/el-bce-sube-los-tipos-de-interes-hasta-el-2>

<https://es.euronews.com/2022/09/22/la-libra-y-el-euro-alcanzan-minimos-historicos>

<https://www.20minutos.es/sexta-subida-consecutiva-de-los-tipos-de-interes-en-ee-uu>

<https://rebusinessonline.com/multifamily-rents-drop-9-nationally-the-largest-one-month>

<https://www.cienciadedatos.net/documentos/py35-redes-neuronales-python.html>

<https://medium.com/datos-y-ciencia/proyecto-machine-learning-predicci>