APLICACIÓN DE MODELOS DE REGRESIÓN PARA LA PREDICCIÓN DEL PRECIO DE PROPIEDADES EN ALQUILER - AIRBNB PRICE ENGINE

Leandro Gabriel Napolitano

Abstract

En el presente informe, se dispone a analizar una base de datos de publicaciones de propiedades en Airbnb con el objetivo de poder predecir los precios de los hospedajes para algunas ciudades dentro de USA, en base a la aplicación de técnicas de Machine Learning - Supervised Learning, NLP, y una prueba final utilizando algoritmos de Deep Learning.

INTRODUCCIÓN

El auge de la economía compartida ha transformado radicalmente la industria del alojamiento, con plataformas como Airbnb emergiendo como líderes en la facilitación de alquileres a corto plazo. La variabilidad en la fijación de precios de las propiedades en Airbnb se ve influida por una multitud de factores, que van desde la ubicación geográfica hasta las características específicas de la propiedad. En este contexto, la aplicación de técnicas de Machine Learning Supervisado se presenta como una herramienta poderosa para entender y predecir los precios de alquiler.

Este informe se centra en la predicción del precio de propiedades en alquiler en Estados Unidos utilizando un enfoque basado en Machine Learning Supervisado. Me he enfocado específicamente en las propiedades listadas en Airbnb, una plataforma que ha ganado popularidad mundial al permitir a los propietarios ofrecer sus espacios de manera eficiente, y a los viajeros encontrar alojamientos personalizados.

La importancia de esta investigación radica en la capacidad de proporcionar a los propietarios y viajeros información precisa sobre los precios de alquiler esperados. Esto no solo beneficia a los anfitriones para fijar tarifas competitivas y justas, sino que también ayuda a los viajeros a planificar y ajustar sus presupuestos de manera más precisa.

Para abordar esta tarea, empleamos un conjunto de datos extenso que incluye información diversa sobre las propiedades y sus entornos. Utilizamos algoritmos de Machine Learning Supervisado, como regresión lineal y árboles de decisión, para entrenar y validar nuestro modelo. La selección de características y la optimización de hiperparámetros desempeñan un papel crucial en la capacidad predictiva del modelo.

A su vez, también se pone en practica el uso de tecnicas de NLP para poder vectorizar las descripciones y amenities de las propiedades, y de ese modo agregarle un valor extra a la predicción.

DATA SET y PREPROCESAMIENTO

Nos encontramos ante un dataset que contiene un listado de 19.309 publicaciones con 29 variables que muestran algunas características de las propiedades en alquiler en la web de Airbnb.

Variables presentes:

Diccionario dataset Airbnb	
Variable	Significado
id	Identificado de la publicación
property_type	Tipo de Propiedad
room_type	Tipo de Habitación
amenities	Amenities que tiene la propiedad
accommodates	Cantidad de comodidades de la publicacion
bathrooms	Cantidad de baños
bed_type	Tipo de cama
cancellation_policy	Politica de cancelación
cleaning_fee	Si tiene un recargo por limpieza o no
city	Ciudad
description	Descripcion de la publicacion
first_review	Fecha de la primera review
host_has_profile_pic	Si el host tiene foto de perfil o no
host_identity_verified	Si el host es verificado por la página o no
host_response_rate	Frecuencia de respuesta del host
host_since	Fecha desde que el host se inicio en Airbnb
instant_bookable	Si la propiedad se puede reservar de manera instantanea o requiere aprobación del dueño
last_review	Fecha de la ultima review
latitude	Latitud geográfica de la propiedad
longitude	Longitud geográfica de la propiedad
name	Nombre de la publicacion
neighbourhood	Barrio de la propiedad
number_of_reviews	Cantidad de reviews de la publicacion
review_scores_rating	Puntaje de la publicacion
thumbnail_url	URL de la publicacion
zipcode	Codigo postal de la propiedad
bedrooms	Cantidad de cuartos
beds	Cantidad de camas
price (variable a predecir	**Precio por noche de la propiedad**

Para obtener la predicción de los precios de propiedades, previo al EDA se dispuso a razonar las variables que podrian contener una relación directa con los mismos.
Se puede entender que en este tipo de propiedades, la ubicación del barrio y ciudad juegan un rol muy importante, asi como tambien el tipo de propiedad y tamaño. Las variables mencionadas, al ser categóricas, reflejan un problema de muy alta dimensionalidad a la vista, lo cual puede ser perjudicial a la performance final del modelo, tendiendo los resultados hacia el overfitting o sobreajuste del mismo.

Por lo cual resulta de real importancia el correcto procesamiento de la información, sumado a una optimización de hiperparametros lo mas adecuada posible, que permita lograr el punto de equilibrio mas optimo en el trade off de varianza vs sesgo.

A su vez, se implementa a modo de prueba final un algoritmo de reducción de la demensionalidad, para comprobar si se logran iguales o mejores resultados generando menor costo computacional y sobreajuste al problema.

BACKGROUND TEÓRICO

Se detalla a continuación la teoría esencial desarrollada a lo largo del informe.

Machine Learning

Es una rama de la inteligencia artificial que empezó a cobrar importancia a partir de los años 80. Se trata de un tipo de IA que ya no depende de unas reglas y un programador, sino que la computadora puede establecer sus propias reglas y aprender por sí misma.

El aprendizaje automático se produce por medio de algoritmos. Un algoritmo no es más que una serie de pasos ordenados que se dan para realizar una tarea.

El objetivo del machine learning es crear un modelo que nos permita resolver una tarea dada. Luego se entrena el modelo usando gran cantidad de datos. El modelo aprende de estos datos y es capaz de hacer predicciones. Según la tarea que se quiera realizar, será más adecuado trabajar con un algoritmo u otro.

Algoritmos de Lenguaje Supervisado

Es el uso de conjuntos de datos etiquetados (en nuesto caso el precio de las propiedades) para entrenar algoritmos que clasifiquen datos o predigan resultados de forma precisa. A medida que los datos se introducen en el modelo, este ajusta sus ponderaciones hasta que dicho modelo se haya ajustado adecuadamente, como parte del proceso de validación cruzada.

El aprendizaje supervisado puede clasificarse en dos tipos de problemas durante la minería de datos:

- La clasificación
- La regresión

Regresión

La regresión se utiliza para comprender la relación entre variables dependientes e independientes.

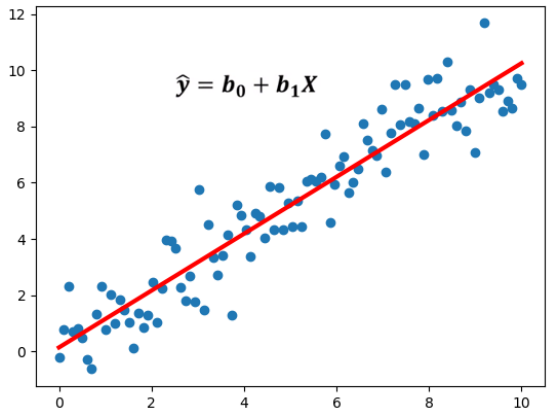
Se utiliza comúnmente para hacer proyecciones, como los ingresos por ventas de un negocio determinado. Regresión lineal y regresión polinomial son algoritmos de regresión populares.

Regresión lineal

La regresión lineal se utiliza para identificar la relación entre una variable dependiente y una o más variables independientes, y normalmente se aprovecha para hacer predicciones sobre resultados futuros.

Cuando solo hay una variable independiente y una variable dependiente, se conoce como regresión lineal simple. A medida que aumenta el número de variables independientes, se denomina regresión lineal múltiple. Para cada tipo de regresión lineal, esta clasificación busca trazar una línea de mejor ajuste, que se calcula mediante el método de mínimos cuadrados.

Sin embargo, a diferencia de otros modelos de regresión, esta línea es recta cuando se traza en un gráfico.



Ejemplo de una recta de regresión lineal

Arboles de Decisión (ensemble)

Un árbol de decisión es un algoritmo de aprendizaje supervisado no paramétrico, que se utiliza tanto para tareas de clasificación como de regresión. Tiene una estructura de árbol jerárquica, que consta de un nodo raíz, ramas, nodos internos y nodos hoja.

El aprendizaje del árbol de decisiones emplea una estrategia de divide y vencerás mediante la realización de una búsqueda codiciosa para identificar los puntos de división óptimos dentro de un árbol. Este proceso de división se repite de forma recursiva de arriba hacia abajo hasta que todos o la mayoría de los registros se hayan clasificado bajo etiquetas de clase específicas. Que todos los puntos de

datos se clasifiquen o no como conjuntos homogéneos depende en gran medida de la complejidad del árbol de decisión. Los árboles más pequeños son más fáciles de obtener nodos hoja puros, es decir, puntos de datos en una sola clase. Sin embargo, a medida que un árbol crece en tamaño, se vuelve cada vez más difícil mantener esta pureza y, por lo general, da como resultado que haya muy pocos datos dentro de un subárbol determinado. Cuando esto ocurre, se conoce como fragmentación de datos y, a menudo, puede resultar en sobreajustes.

Procesamiento de Lenguaje Natural (NLP)

El procesamiento del lenguaje natural (NLP) hace referencia a la rama de la informática (y más específicamente, a la rama de la inteligencia artificial o IA encargada de dar a los ordenadores la capacidad de comprender textos y palabras habladas de la misma manera que los seres humanos.

NLP combina la lingüística computacional (modelado basado en reglas del lenguaje humano) con modelos estadísticos, de machine learning y deep learning. Juntas, estas tecnologías permiten a los ordenadores procesar el lenguaje humano en forma de datos de texto o voz y "comprender" su significado completo, junto con la intención y el sentimiento del orador o escritor.

Metricas para la Evaluacion de Modelos de Regresión

A la hora de evaluar la performance de nuestro modelo, se disponen distintas metricas que nos permiten asegurar el rendimiento óptimo, y a su vez realizar comparaciones con otros modelos.

A continuacion se detallan las metricas utilizadas a lo largo del informe

Error absoluto medio (MAE)

Esta métrica es una medida de la diferencia entre dos valores, es decir, nos permite saber que tan diferente es el valor predicho y el valor real u observado. Para que un error con valor positivo no cancele a un error con error negativo usamos el valor absoluto de la diferencia. Como nos interesa conocer el comportamiento del error de todas las observaciones y no solamente de una, entonces obtenemos el promedio de los valores absolutos de la diferencia.

$$MAE = \frac{\sum_{i=1}^{n} |y_i - x_i|}{n}$$

Error medio cuadrado (MSE)

Esta métrica es muy útil para saber que tan cerca es la línea de ajuste de nuestra regresión a las observaciones. Al igual que en caso anterior evitamos que un error con valor positivo anule a uno con valor negativo, pero en lugar de usar el valor absoluto, elevamos al cuadrado la diferencia.

$$MSE = \frac{\sum_{i=1}^{n} (y_i - x_i)^2}{n}$$

Raíz del error medio cuadrado (RMSE)

Como la métrica anterior nos da el resultado en unidades cuadradas, para poder interpretarlo más fácilmente sacamos la raíz cuadrada y de esta manera tenemos el valor en las unidades originales.

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n} (y_i - x_i)^2}{n}}$$

R2, R cuadrada

R2 es el coeficiente de determinación, nos indica que tanta variación tiene la variable dependiente que se puede predecir desde la variable independiente. En otras palabras que tan bien se ajusta el modelo a las observaciones reales que tenemos. Cuando usamos R2 todas las variables independientes que estén en nuestro modelo contribuyen a su valor.

El mejor valor posible que tenemos con R2 es 1 y el peor es 0. Una desventaja que tiene es que asume que cada variable ayuda a explicar la variación en la predicción, lo cual no siempre es cierto. Si adicionamos otra variable, el valor de R2 se incrementa o permanece igual, pero nunca disminuye. Esto puede hacernos creer que el modelo esta mejorando, pero no necesariamente es así.

$$R^2 = 1 - \frac{\sum (y_i - x_i)^2}{\sum (y_i - \mu_y)}$$

ANÁLISIS EXPLORATORIO DE DATOS – EDA

El dataset inicialmente cuenta con 19309 samples, 28 features y 1 variable a predecir.

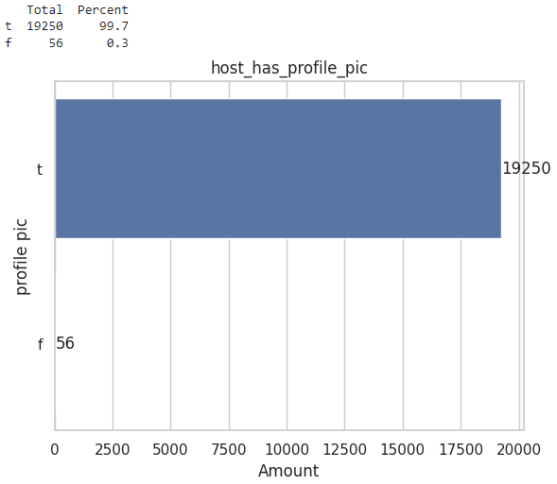
A su vez, presenta un porcentaje significativo de nulos en varias de sus features.

	Total	Percentage
host_response_rate	4296	22.2
review_scores_rating	4134	21.4
first_review	3954	20.5
last_review	3954	20.5
thumbnail_url	2402	12.4
neighbourhood	1458	7.6
zipcode	225	1.2
bathrooms	35	0.2
beds	24	0.1
bedrooms	17	0.1
instant_bookable	0	0.0

Los nulos correspondientes a las primeras 4 features de la tabla que se detalla arriba, se trataban de casos en los cuales el host y/o la propiedad publicada corresponde a una nueva publicacion en la web, por lo cual aun no cuentan con reputacion y scoring. Por otro lado, las url en nuestro caso de estudio no fueron utilizadas ya que no aportan informacion útil.
Por ultimo, cabe mencionar la importancia de los nulos en la variable **neighborhood**, que resulta ser una variable con mucha relevancia con respecto a las variaciones de precio.

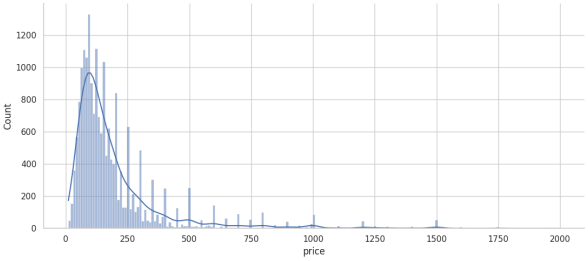
Luego de intentar con técnicas de imputación de nulos, finalmente se tomó como decision final eliminar las samples que tienen valores nulos y no realizar imputaciones artificiales, ya que influían significativamente de forma negativa en las predicciones finales.

En cuanto a las features, mediante el análisis exploratorio se eliminaron aquellas features que: presentaban clases desbalanciadas, o no agregaban valor al modelo.

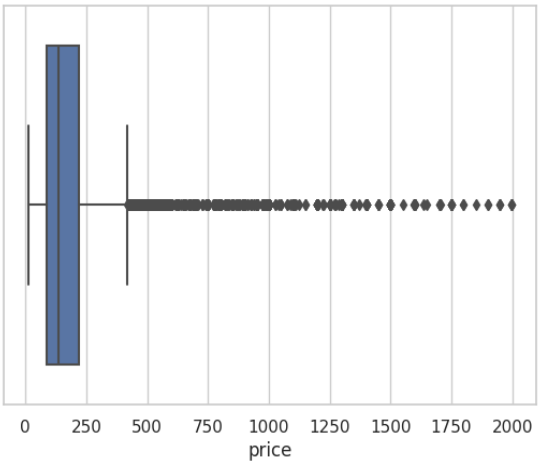


Ejemplo: variable 'host_has_profile_pic' eliminada por altos niveles de desbalanceo de clases.

En cuanto a la variable **precio**, en la distribucion de los datos se pudo observar una asimetria positiva significativa, con un cierto volumen de **outliers**



Distribución de la variable target 'price'



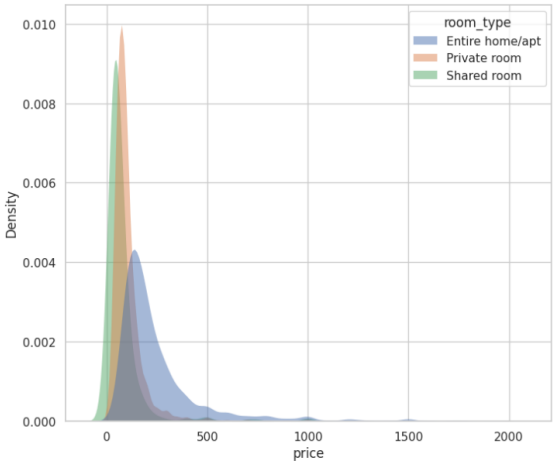
Presencia de outliers hacia la derecha (altos niveles de precios)

Se procedió mediante el método IQR a la eliminación de outliers.

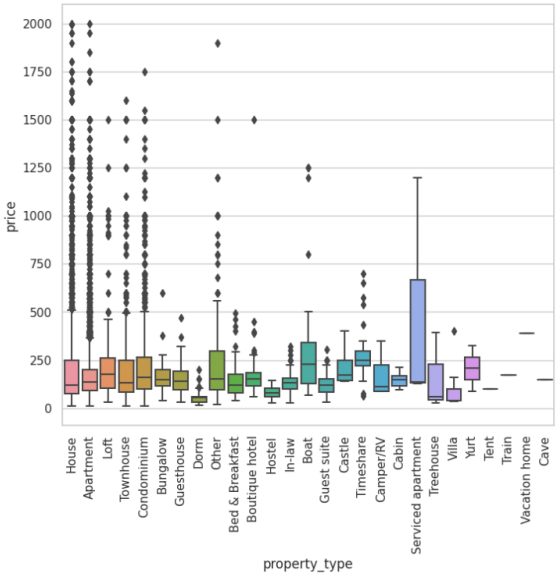
En su esquema final, el dataset quedó compuesto por 21 variables dependientes, 1 variable independiente, y 11988 samples.

A continuación se muestran algunas variables que demuestran agregar variabilidad al target precio

room_type: podemos observar el incremento de precios en el alquiler de apartamentos completos, asi como también que los rooms compartidos brindan los precios mas económicos del mercado.



property_type: Los tipos de propiedad influyen significativamente en los niveles de precios en cada propiedad.



Variables de texto: *'description', 'name', 'amenities'*

La información que se encuentra en las variables mencionadas, resulta de gran utilidad, y hubiera sido un gran desaprovechamiento de la misma, si no se hubiesen realizado técnicas de vectorización para la generación de embedding de datos.

Por lo tanto, se procedió a vectorizar dicha información, concatenando en primer lugar las tres columnas en una sola, luego se aplicaron tecnicas de stemming, lematizing, quitando caracteres especiales, acentos, digitos, stopwords, se utilizaron tecnicas para expandir las contractions,
Por ultimo, se tokenizó cada corpus, y se vectorizaron mediante la técnica **TFIDF** para generar los embeddings necesarios por cada sample.

Recategorización de clases desbalanceadas

En ciertas variables, se observaron clases que estaban desbalanceadas en cuanto al tamaño de muestras, con respecto a las clases mayoritarias. Por lo tanto, se tomó la decision de recategorizar como “Other” aquellas clases que su suma resultaba menor a una semilla impuesta.

De ese modo, se pudo lograr una reducción de la dimensionalidad del dataset.

A modo de ejemplo, en la variable **neighbourhood** se colocó una semilla de 15, por lo que todo barrio que su suma haya dado menor a la semilla, fue recategorizado a “Other”.
Las otras variables afectadas por esta técnica fueron: **property_type**, con una semilla de 100, y **cancellation_policy**, con una semilla de 110.

IMPLEMENTACIÓN DE ALGORITMOS - EVALUACIONES

Luego de obtener los embeddings, y realizar los encodings de las variables categoricas, se obtuvo una matriz de muestras de **11988x1187,** por lo cual el riesgo de overfitting resulta de alta importancia.

Se realizó el entrenamiento de diversos modelos, tanto de regresion lineal, como de bagging, gradient boosting y por ultimo una red neuronal.

Los resultados obtenidos son los siguientes:

Model	MAE	MSE	RMSE	R2 Train	R2 Test
Linear Regression	36.85	2381.43	48.8	0.71	0.6
CV Ridge Linear Regression	36.85	2381.56	48.8	0.71	0.6
CV Lasso Linear Regression	34.85	2191.35	46.81	0.69	0.63
XGBoost	31.86	1965.88	44.34	0.94	0.67
ADABoost	38.84	2571.57	50.71	0.72	0.56
Gradient Boosting	32.66	2019.39	44.94	0.83	0.66
Random Forest	39.5	2744.13	52.38	0.83	0.53
MLP Neural Network	-	2487.58	-	-	-

Se puede observar que XGBoost brindó la mejor performance, lo cual es entendible en base a la complejidad de su entrenamiento, y que es muy bueno para modelos de alta dimensionalidad.
Cabe destacar la muy buena performance de los modelos de regresión, destacan debido a la simpleza y rápidez de sus operaciones para obtener los pesos óptimos.
Por último, la red neuronal que se utilizó a modo de prueba no logró resultados interesantes, siendo uno de los modelos con peor performance.

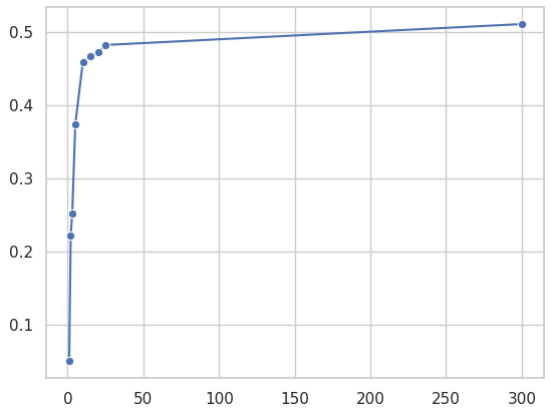
EVALUACIÓN CON REDUCCION DE LA DIMENSIONALIDAD (PCA)

En este ultimo apartado, se procedió a evaluar técnicas de reducción de la dimensionalidad, para comprobar si el modelo puede lograr mejor performance, o al menos mantener la performance anterior, pero con el beneficio de trabajar con menor dimensionalidad, y reducir el overfitting

Se realizaron pruebas utilizando la técnica PCA (Principal Component Analysis),

testeando con diferentes componentes, y a su vez entrenando al modelo Gradient Boosting. Los resultados fueron:

PCA - G.B. - Components	MAE	MSE	RMSE	R2 Train	R2 Test
300	39.87	2888.98	53.75	0.88	0.51
25	41.07	3057.7	55.29	0.76	0.48
20	41.65	3118.75	55.84	0.73	0.47
15	41.9	3149.16	56.12	0.71	0.46
10	42.32	3195.87	56.53	0.69	0.45
5	45.54	3696.57	60.80	0.58	0.37
3	50.41	4415.75	66.45	0.47	0.25
2	51.81	4591.08	67.75	0.41	0.22
1	59.05	5605.11	74.86	0.22	0.05



R2 Score vs nº componentes (PCA - GB)

A medida que se reduce la cantidad de componentes, el rendimiento del modelo decrece, lo cual es coherente. Con los componentes utilizados, no se vio una performance muy satisfactoria, sin embargo con 300 componentes se logró un R1 Score de 0.51, 16 puntos menos que con nuestro mejor modelo (XGboost), y con menos de un 5% en cuanto a cantidad de variables, lo cual denota un buen potencial. Debido al elevado tiempo de ejecucion, no se evaluaron opciones con myor cantidad de componentes.

CONCLUSIONES

A partir del análisis realizado, se pudo concluir que pese a la elevada dimensionalidad, el conjunto de datos contiene una consistencia adecuada en su información, logrando resultados destacables teniendo en cuenta los niveles de variación que pueden sufrir los precios de alquileres en una economía. Se debe tener en cuenta que hay muchos otros factores que pueden afectar al precio y no se llevó a cabo un análisis, como puede ser por ejemplo la estacionalidad del conjunto de datos en el momento de su extracción, la cercanía de las propiedades a establecimientos de gran relevancia, o transporte público, etc. Sin embargo, los resultados obtenidos son mas que satisfactorios, aunque debido a la gran diferencia entre los resultados de entrenamiento y de testeo, entiendo que probablemente haya tecnicas aun por explotar que permitan optimizar la performance.
En cuanto a la reducción de la dimensionalidad, se podria evaluar una mejora utilizando algoritmos para problemas no lineales, ya que PCA conlleva la limitación de la linealidad.
Otro punto a mejorar, es optimizar el tratamiento de los valores nulos para conservar la mayor cantidad de informacion posible, ya que la perdida de datos en la transformación fue significativo. Un mejor tratamiento de los nulos puede brindarnos mayor consistencia en los resultados.

REFERENCIAS

https://www.ibm.com/mx-es/topics/supervised-learning

https://medium.com/@nicolasarrioja/m%C3%A9tricas-en-regresi%C3%B3n-5e5d4259430b

https://www.ibm.com/es-es/topics/natural-language-processing

https://pandas.pydata.org/docs/

https://seaborn.pydata.org/

https://scikit-learn.org/stable/

https://www.statlearning.com/

https://www.deeplearningbook.org/

https://powerunit-ju.com/wp-content/uploads/2021/04/Aurelien-Geron-Hands-On-Machine-Learning-with-Scikit-Learn-Keras-and-Tensorflow_-Concepts-Tools-and-Techniques-to-Build-Intelligent-Systems-OReilly-Media-2019.pdf