Exploración y Curación de Datos - Documentación

Grupo 28

Integrantes:

- **Gustavo Alvarez Lupu**
- **Laura Hayas**
- **Maria Emilia Santacruz**
- **Nicolás Ambrosis**

Docente: Laura Montes

A continuación detallamos los trabajos realizados en los entregables 1 y 2:

1. **Criterios de exclusión de filas**

En el punto 2.2.2 del entregable 1, excluimos los 'zipcode' que tenían una frecuencia en el dataframe inferior a 20, quedándonos de esa manera con el 96% de los datos..

2. **Interpretación de las columnas presentes**

Las columnas definitivas con las que trabajamos fueron las siguientes, relacionadas a la información de las propiedades:

- Suburb: región
- Rooms: número de habitaciones
- Type: tipo de propiedad. h - casa, cabaña; u - monoambiente, duplex; t - complejo de viviendas, otras.
- Price: precio en dólares.
- Distance: distancia al centro de la ciudad.
- Postcode: código postal.
- Landsize: metros del terreno.
- Propertycount: cantidad de propiedades por región (suburb)
- BuildingArea: metros construidos.
- YearBuilt: año de construcción.

Para el análisis de variables relevantes dividimos el total de características en tres grupos:

a. *Geografia subyacente a la propiedad*

En relación a la ubicación de los inmuebles, consideramos relevantes las variables **suburb**, **distance**, **postcode**, **propertycount**, ya que son variables que pueden tener una influencia mayor en el precio de una propiedad, por un lado más de practicidad. No consideramos que **address**, **regionname**, **councilarea** sean

columnas relevantes para el análisis, ya que es un valor distinto para cada caso y no se puede agrupar para hacer un análisis más general de los precios de las propiedades.

b. Características físicas de la propiedad

Respecto a lo que hace a la propiedad en sí, se puede considerar como relevante **type**, **rooms**, **landsize**, **buildinarea** y **yearbuilt** ya que son variables que pueden tener una mayor influencia en el precio de la propiedad. Por otro lado, consideramos que **bedroom2** y **bathroom** no son relevantes ya que pueden ser redundantes con la variable **rooms**, y se considera suficiente la información con esta última variable.

Respecto a la variable **date**, la excluimos del análisis porque tiene un valor distinto para cada propiedad, y no la consideramos tan relevante por el rango de tiempo en que realizamos el análisis.

c. Precio de la propiedad

Price es fundamental mantenerla dentro del dataset ya que en etapas posteriores del proyecto será necesaria para entrenar y validar el modelo de predicción de precios. Por otro lado, **method** y **sellerg** no consideramos que sean relevantes.

3. Transformaciones realizadas

En ambos entregables, trabajamos con dos conjuntos de datos, uno de ellos ‘Melbourne’, de la compentencia Kaggle sobre estimación de precios de ventas de propiedades en Melbourne, Australia. Adicionalmente, para contar con más información a la hora de estimar con mayor precisión el valor del vecindario de cada propiedad, se trabajó con un dataset similar: las publicaciones de la plataforma AirBnB en Melbourne en el año 2018, Airbnb’.

a) En el Entregable Parte 1:
   i) Seleccionamos las columnas relevantes del df de Melbourne de acuerdo al apartado anterior, y eliminamos las restantes;
   ii) Agrupamos el df de Airbnb por zipcode y calculamos el precio promedio, tomando sólo estas dos variables para continuar el análisis;
   iii) Utilizando la columna zipcode del df de Airbnb, hicimos un Merge con la columna postcode del df de Melbourne. Con el resultado de esta unión, generamos un nuevo df con el nombre ‘Merge_df.csv’, que utilizamos para trabajar en la segunda parte del entregable.
b) En el Entregable Parte 2, partiendo del df ‘Merge_df.csv’:
   i) Eliminamos las columnas 'BuildingArea' y 'YearBuilt': para trabajarlas de manera separada y la columna 'zipcode' porque aportaba información redundante;
   ii) Convertimos el tipo de dato de ‘postcode’ a ‘str’, por considerar que se trata de una variable categórica;

iii) Se codificó todo el df con la clase DictVectorizer, obteniendo una matriz;
iv) Concatenamos la matriz obtenida en el punto anterior con las columnas 'BuildingArea' y 'YearBuilt' que habíamos eliminado previamente;
v) Estandarizamos la matriz con el método StandardScaler;
vi) Aplicamos una instancia de IterativeImputer con un estimador KNeighborsRegressor para imputar los valores faltantes de las variables 'BuildingArea' y 'YearBuilt'. En un primer escenario, se completaron los valores faltantes sólo considerando esas dos variables. en un segundo escenario, se utilizó todo el df para completar los valores faltantes de las dos variables;
vii) Se compararon gráficamente las distribuciones de ambos escenarios, con la distribución original (antes de las imputaciones);
viii) Normalizamos la matriz obtenida en el segundo escenario con el método MinMaxScaler;
ix) Aplicamos la reducción de dimensionalidad con el método PCA;
x) Graficamos la varianza explicada en función del número de componentes;
xi) Seleccionamos los primeros 10 componentes y las agregamos al conjunto de datos procesado en un nuevo df: '**df_final'.**