Desafío 2

Grupo 5

Lesertesseur, Diego
Panizza, Camila
Pellecchia, Franco
Magariños, Néstor
Suarez, Horacio

Digital House
Coding School

OBJETIVO

- Desarrollar un modelo de regresión lineal que permita predecir el precio por metro cuadrado de una propiedad

ESTRUCTURA

i. Dataset
ii. Modelos estimados
iii. Selección del mejor modelo
iv. Interpretación de coeficientes

Dataset

DATASET PROPERATI (1/2)

- El dataset limpio y sin nulos contiene **61.000 observaciones**

Variable	Tipo	Ejemplos
price_usd_per_m2	float64	[1474.0, 1467.0, 1136.0, 464.0, 1006.0, 1074.0]
surface_total_in_m2	float64	[296.5, 318.2, 303.8, 700.0, 397.8, 180.0]
property_type	object	[house, apartment, PH, store]
provincia	object	[Bs.As. G.B.A. Zona Norte, Santa Fe]
ciudad	object	[Tigre, Pilar, Funes, La Plata, Escobar]
departamento	object	[Tigre, Villa Rosa, Village Golf & Tennis]
barrio	object	[Barrio Los Lagos, Barrio Los Alisos]
rooms	float64	[3.0, 2.0, 4.0, 5.0, 1.0, 7.0, 6.0]
banos	float64	[1.0, 2.0, 3.0, 4.0, 5.0, 6.0, 0.0, 8.0, 9.0]

DATASET PROPERATI (2/2)

- El dataset limpio y sin nulos contiene **61.000 observaciones**

Variable	Tipo	Ejemplos
pileta	int64	[0, 1]
cochera	int64	[0, 1]
barrioCerrado	int64	[0, 1]
jacuzzi	int64	[0, 1]
terraza	int64	[0, 1]
quincho	int64	[0, 1]
seguridad	int64	[0, 1]
balcon	int64	[0, 1]
calefacción	int64	[0, 1]

NUEVAS VARIABLES

- Se transformaron las variables categóricas “Tipo de propiedad” y “Localidad” en **variables dummy**
  - La variable localidad es una concatenación de las variables ‘Ciudad’, ‘Departamento’ y ‘Barrio’

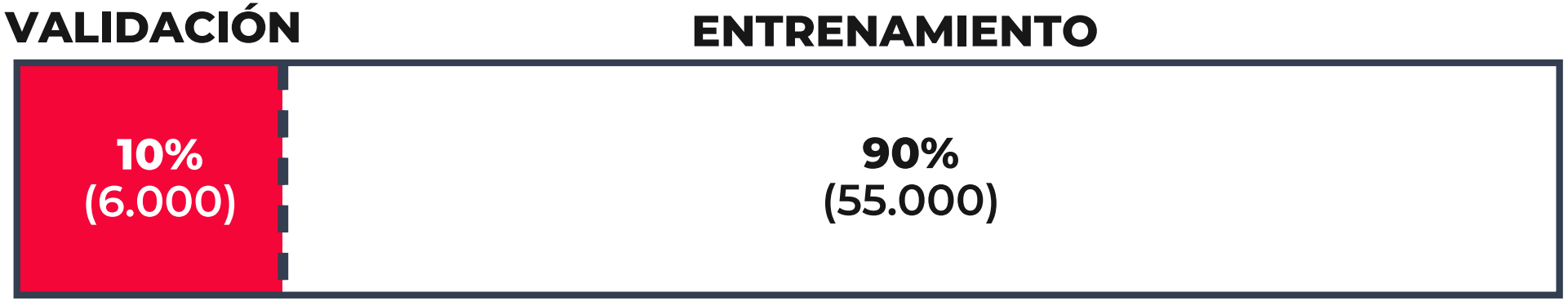
- Se confeccionaron **variables sinergias** para las siguientes combinaciones
  - Casa - Pileta
  - Departamento - Cochera
  - Cuartos - Baños
  - PH - Cochera

- Se introdujo a los metros cuadrados totales de la propiedad elevada al cuadrado para captar un posible **efecto no lineal**

Aclaración: Luego de agregar las nuevas variables, quedan 988 features

PARTICIÓN DEL DATASET

- Se realizó una **división del dataset** en dos partes: dataset de entrenamiento y dataset de validación
- La división fue realizada mediante **selección aleatoria** con **estratificación por deciles de la variable target**



Aclaración: Entre paréntesis se indica la cantidad de observaciones de cada dataset

Modelos estimados

MODELOS

i. Mínimos Cuadrados Ordinarios
ii. Limpieza de variables no significativas
iii. Lasso y Ridge

MÍNIMOS CUADRADOS ORDINARIOS (MCO)

	MCO		
Dep. Variable:	price_usd_per_m2_clean	**R-squared:**	0.714
Model:	OLS	**Adj. R-squared:**	0.710
Method:	Least Squares	**F-statistic:**	152.0
Date:	Tue, 23 Jun 2020	**Prob (F-statistic):**	0.00
Time:	12:35:17	**Log-Likelihood:**	-4.1297e+05
No. Observations:	55152	**AIC:**	8.277e+05
Df Residuals:	54259	**BIC:**	8.357e+05
Df Model:	892		

LIMPIEZA DE VARIABLES NO SIGNIFICATIVAS (1/3)

LIMPIEZA DE VARIABLES NO SIGNIFICATIVAS (1/3)

PROBLEMA

El modelo final seguía conteniendo variables no significativas

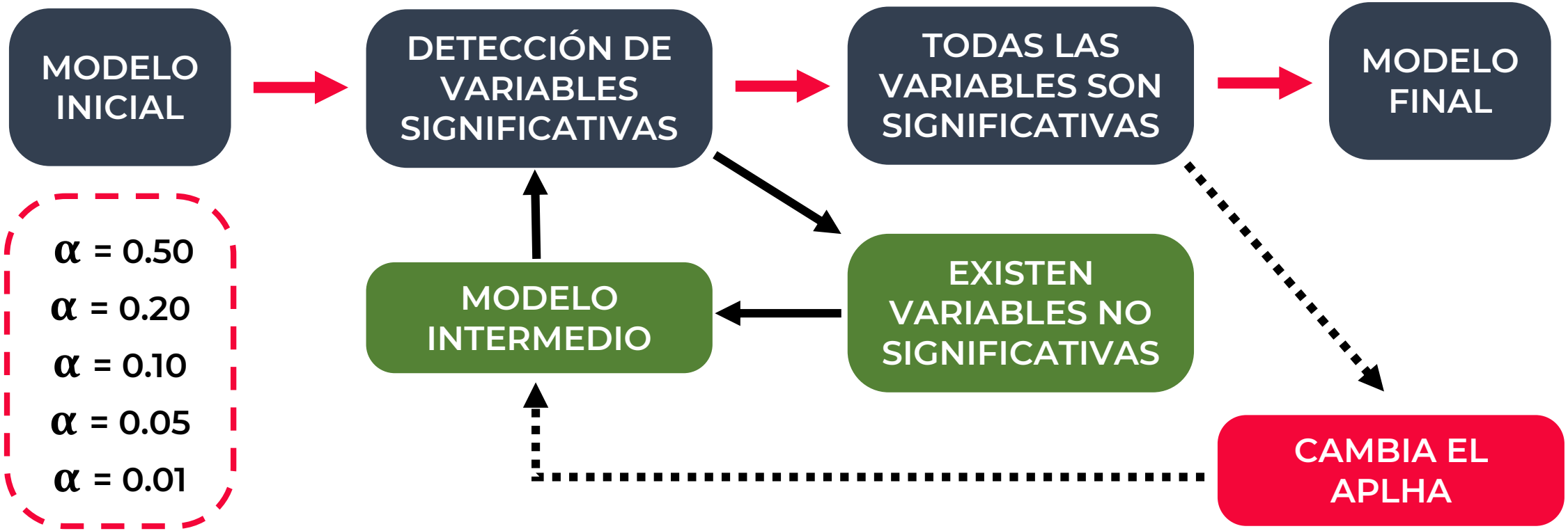
LIMPIEZA DE VARIABLES NO SIGNIFICATIVAS (2/3)

LIMPIEZA DE VARIABLES NO SIGNIFICATIVAS (2/3)

PROBLEMA

Estamos eliminando variables que figuran como no significativas por la presencia de otras variables que hacen ruido en el modelo

LIMPIEZA DE VARIABLES NO SIGNIFICATIVAS (3/3)



LIMPIEZA DE VARIABLES NO SIGNIFICATIVAS (3/3)

step	alpha	previo	posterior
1	0.5	988	675
2	0.5	675	645
3	0.5	645	629
4	0.5	629	621
5	0.5	621	617
6	0.5	617	616
7	**0.5**	**616**	**616**
8	0.2	616	501
9	0.2	501	499
10	**0.2**	**499**	**499**
11	0.1	499	425

step	alpha	previo	posterior
12	0.1	425	404
13	0.1	404	399
14	**0.1**	**399**	**399**
15	0.05	399	353
16	0.05	353	346
17	0.05	346	345
18	**0.05**	**345**	**345**
19	0.01	345	272
20	0.01	272	269
21	0.01	269	267
22	**0.01**	**267**	**267**

Ver código

MCO CON VARIABLES SIGNIFICATIVAS AL 1%

MCO solo con variables significativas

Dep. Variable:	price_usd_per_m2_clean	**R-squared:**	0.709
Model:	OLS	**Adj. R-squared:**	0.707
Method:	Least Squares	**F-statistic:**	500.5
Date:	Tue, 23 Jun 2020	**Prob (F-statistic):**	0.00
Time:	16:54:15	**Log-Likelihood:**	-4.1348e+05
No. Observations:	55152	**AIC:**	8.275e+05
Df Residuals:	54884	**BIC:**	8.299e+05
Df Model:	267		

REGULARIZACIÓN
RIDGE Y LASSO

- Se corrieron ambos modelos con **cross-validation** para alcanzar el alpha óptimo [alphas=np.linspace(**0.001, 10, 100**)]

- Solo se utilizaron solo las variables detectadas anteriormente como significativas

Modelo	Alpha óptimo	R^2
Lasso	0,001	0,709
Ridge	0,001	0,709

Aclaración Con las variables iniciales ambos modelos arrojaban R^2=0.71 y Lasso solo anulaba 153 variables

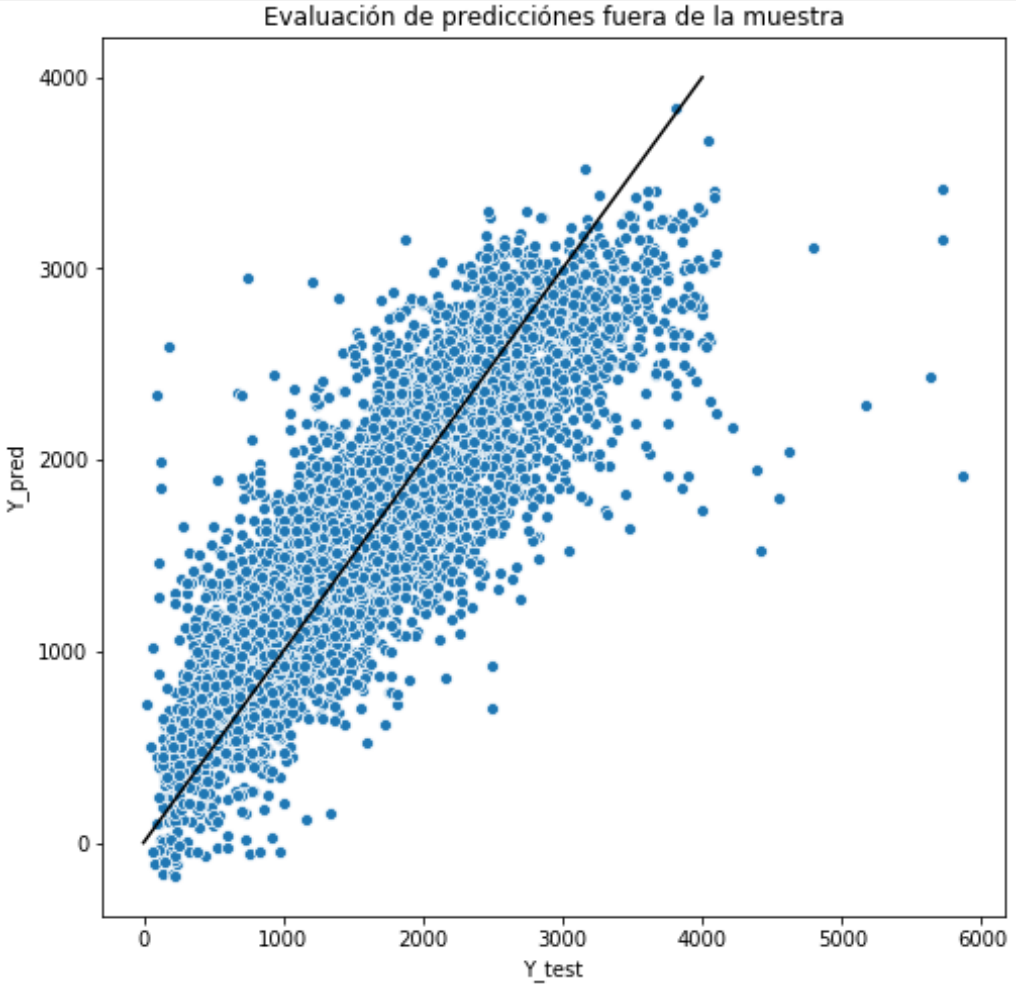
Selección del mejor modelo

SELECCIÓN DEL MEJOR MODELO

- Los modelos con regularización **Ridge y Lasso no presentan mejoras**. Esto resulta un indicio de que no existe overfiting
- El modelo de **Mínimos Cuadrados Ordinarios** con 190 features y un R^2=0.70 resulta **el más adecuado para la predicción**

PREDICCIÓN
FUERA DE LA MUESTRA

$R^2 = 0.70$



Interpretación de coeficientes

INTERPRETACIÓN DE COEFICIENTES

PRINCIPALES

Variable	Beta
const	1300,4
apartment	250,9
house	150,9
store	714,1
surface_total_in_m2	-4,6
surface_total_in_m2__ squared	0,004
rooms_clean	25,2
banos	42,9

AMENITIES

Variable	Beta
Pileta	126,4
Jacuzzi	187,1
Quincho	39,8
Gimnasio	112,0
Seguridad	56,3
Balcon	67,6
Calefaccion	115,0
Jardín	-86,8

SINERGIA

Variable	Beta
house_pileta	80,7
apartment_cochera	154,3
ph_cochera	74,6

¡Muchas gracias!

Digital House
Coding School

Anexo

LIMPIEZA DE VARIABLES NO SIGNIFICATIVAS

```
list_alpha=[0.50,0.20,0.10,0.05,0.01]

variables_significativas = data_features.columns
variables_significativas_previo= variables_significativas.shape[0]
step=1

for alpha in list_alpha:
    variables_significativas_posterior=0
    while variables_significativas_posterior<variables_significativas_previo:
        Xtrain = Xtrain[variables_significativas]
        variables_significativas_previo= variables_significativas.shape[0]

        model_sm = sm.OLS(ytrain,Xtrain).fit()

        variables_significativas = pd.DataFrame(data = [x for x in model_sm.summary().tables[1].data[1:] if float(x[4]) < alpha],\
                                                columns = model_sm.summary().tables[1].data[0])['']

        variables_significativas_posterior=variables_significativas.shape[0]
        print(step, alpha,variables_significativas_previo,variables_significativas_posterior)
        step=step+1
```

Volver