

House Prices - Advanced Regression Techniques

David Muñoz
Paco Sangorrín


El reto

El reto es predecir el precio de venta final de las viviendas. Esta información se almacena en la columna **SalePrice**, es decir esta es nuestra **variable objetivo**.

El reto

Con **79 variables explicativas** que describen (casi) todos los aspectos de las viviendas residenciales de Ames (Iowa), esta competición te reta a predecir el precio final de cada vivienda.

Estamos ante un claro caso de **técnicas avanzadas de regresión** en la que hay que **analizar bien las variables** para quedarnos con aquellas que realmente son las relevantes. Una vez seleccionadas **aplicaremos diversos modelos de regresión ajustando sus hiperparámetros** para determinar cuál es el que mejor predice el precio de la vivienda.



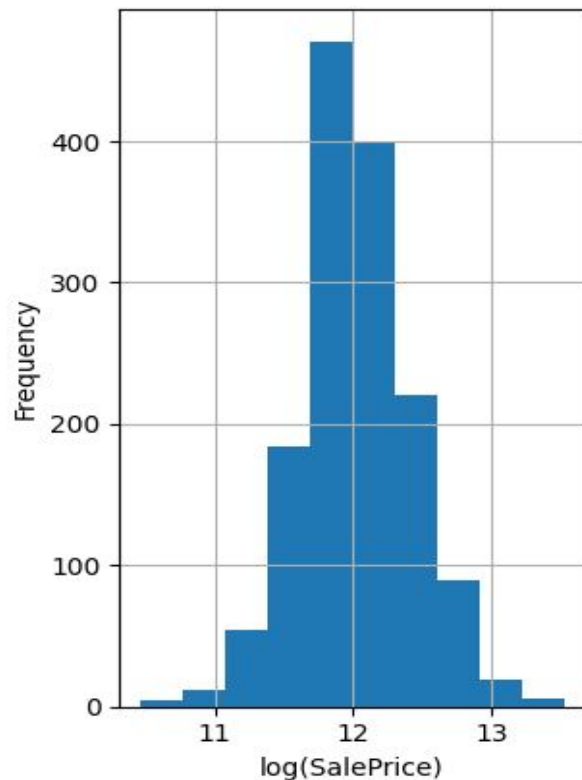
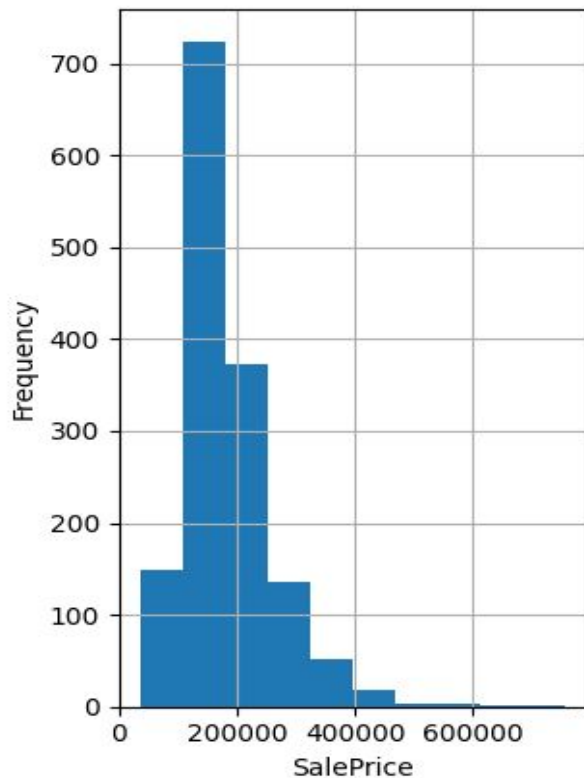


Índice

1. Exploración de variables
2. Modelos probados
3. Comparativa de modelos
4. Modelo final escogido

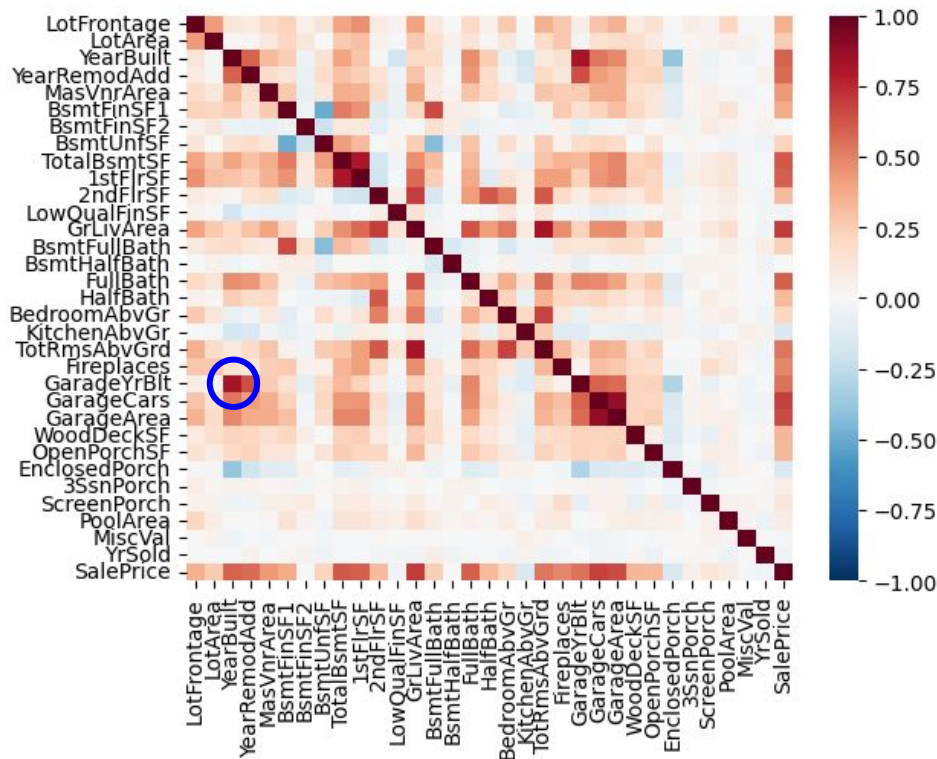
Exploración de variables

Exploración de variables: Objetivo



Tomamos el logaritmo en SalePrice para que su distribución se asemeje lo más posible a una distribución normal

Exploración de variables: Numéricas

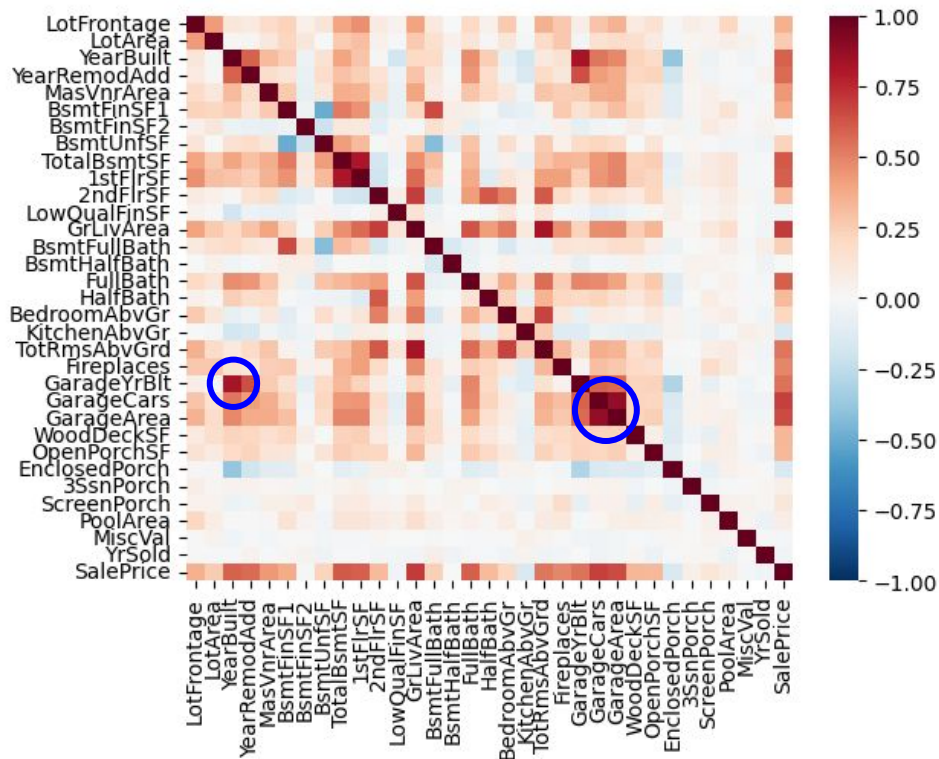


En el mapa de calor vemos que la mayoría de correlaciones son nulas o positivas entre las variables, habiendo pocas con correlación negativa, por lo que, en general, cuando una variable sube, la otra también (correlación positiva) o son independientes (correlación nulas).

Vemos correlaciones fuertes en:

- YearBuilt - GarageYrBlt

Exploración de variables: Numéricas

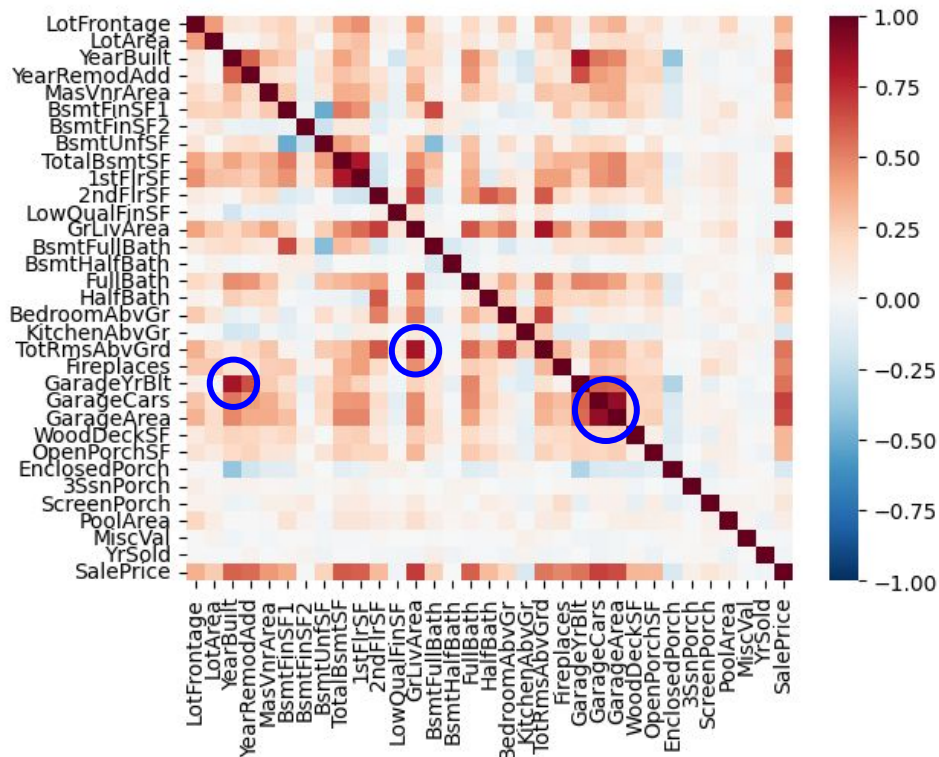


En el mapa de calor vemos que la mayoría de correlaciones son nulas o positivas entre las variables, habiendo pocas con correlación negativa, por lo que, en general, cuando una variable sube, la otra también (correlación positiva) o son independientes (correlación nulas).

Vemos correlaciones fuertes en:

- YearBuilt - GarageYrBlt
- GarageCars - GarageArea

Exploración de variables: Numéricas

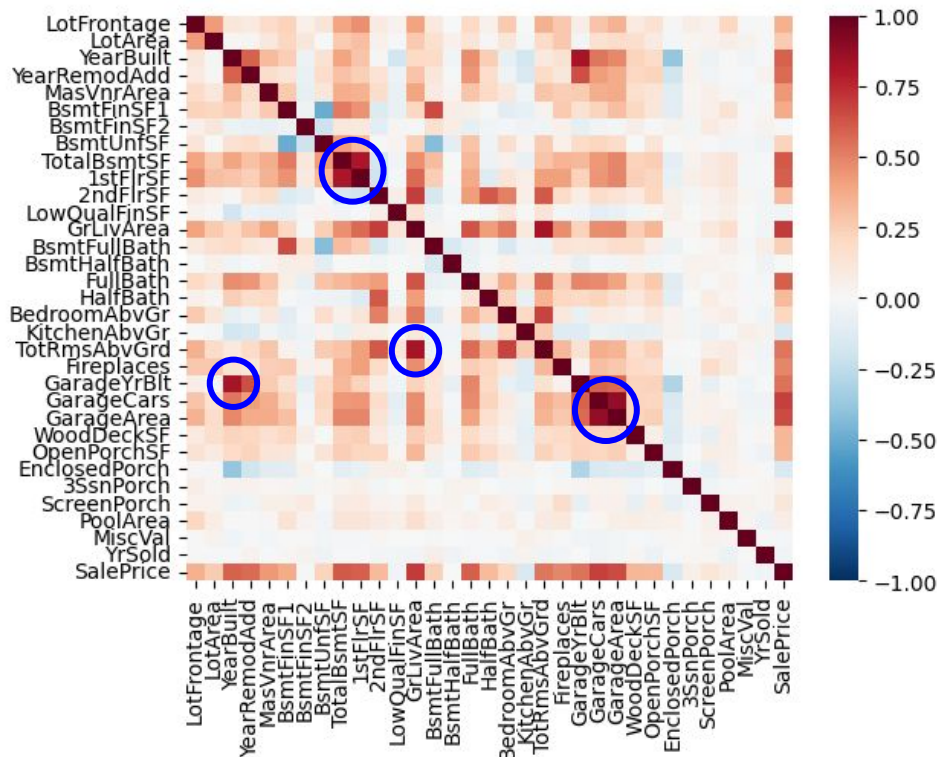


En el mapa de calor vemos que la mayoría de correlaciones son nulas o positivas entre las variables, habiendo pocas con correlación negativa, por lo que, en general, cuando una variable sube, la otra también (correlación positiva) o son independientes (correlación nulas).

Vemos correlaciones fuertes en:

- YearBuilt - GarageYrBlt
- GarageCars - GarageArea
- TotRmsAbvGrd - GrLivArea

Exploración de variables: Numéricas

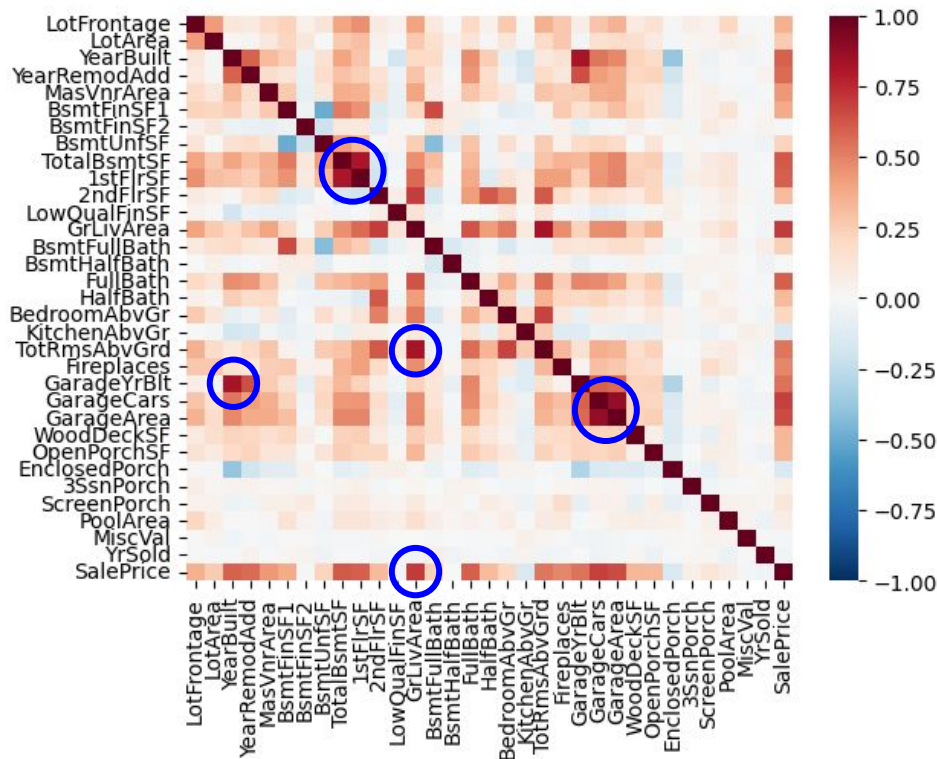


En el mapa de calor vemos que la mayoría de correlaciones son nulas o positivas entre las variables, habiendo pocas con correlación negativa, por lo que, en general, cuando una variable sube, la otra también (correlación positiva) o son independientes (correlación nulas).

Vemos correlaciones fuertes en:

- YearBuilt - GarageYrBlt
- GarageCars - GarageArea
- TotRmsAbvGrd - GrLivArea
- TotalBsmtSF - 1stFlrSF

Exploración de variables: Numéricas



En el mapa de calor vemos que la mayoría de correlaciones son nulas o positivas entre las variables, habiendo pocas con correlación negativa, por lo que, en general, cuando una variable sube, la otra también (correlación positiva) o son independientes (correlación nulas).

Vemos correlaciones fuertes en:


- YearBuilt - GarageYrBlt
- GarageCars - GarageArea
- TotRmsAbvGrd - GrLivArea
- TotalBsmtSF - 1stFlrSF
- SalePrice - GrLivArea

Exploración de variables. Numéricas: Valores perdidos

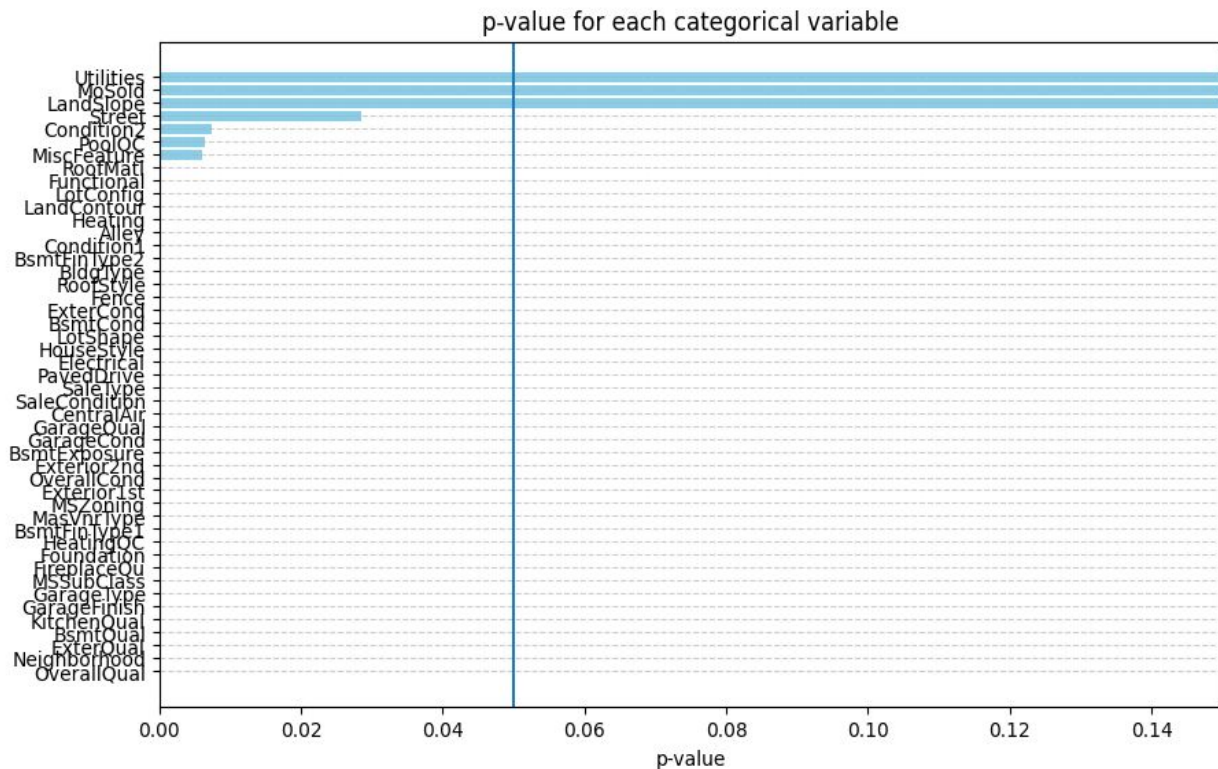
- En **LotFrontage** (Linear feet of street connected to property) cambiamos los NaN por 0, porque entendemos que hay distancia cero.
- La variable **GarageYrBuilt**, al tener valores perdidos y estar muy correlacionada con YearBuilt (0.83) que tiene todos los valores, la descartamos.
- **MasVnrArea** (Masonry veneer area in square feet). Los 8 valores perdidos los ponemos a 0. La distribución parece cuadrar con los que tienen valor 0.



Exploración de variables: Categóricas. Valores nulos.

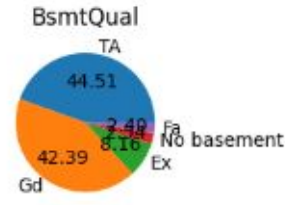
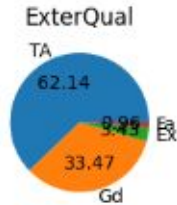
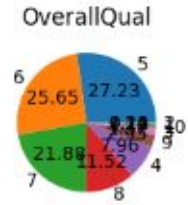
- **PoolQC**: Es que no hay piscina.
 - **MiscFeature**: Significa que no hay ninguna otra feature extra en la casa por lo que ponemos un None.
 - **Alley**: Significa que no hay Alley Access.
 - **Fence**: No hay valla
 - **MasVnrType**: No hay tipo de revestimiento de mampostería
 - **FireplaceQu**: No hay chimenea.
 - **GarageType, GarageCond, GarageQual, GarageFinish**: No hay garage ni nada de las variables asociadas al garage.
 - **BsmtExposure, BsmtQual, BsmtFinType1, BsmtFinType2, BsmtCond**: No basement
 - **Electrical**: El valor perdido es perdido de verdad. El valor para $\log(\text{SalePrice})$ es de 12.02, muy cerca de la media de la categoría 'SBrkr', por lo que lo ponemos ahí.
- 

Exploración de variables: Categóricas. Test ANOVA

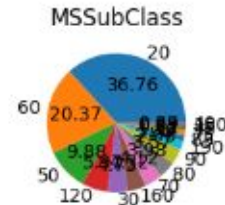
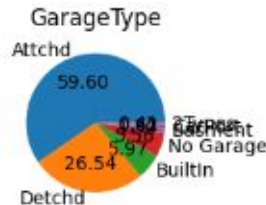
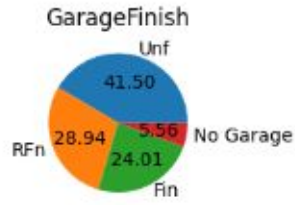
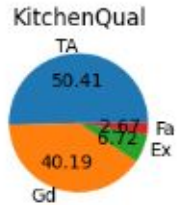


- Observamos que al aplicar el Test ANOVA las variables menos informativas
 - Utilities
 - Mosold
 - LandSlope


Exploración de variables: Categóricas



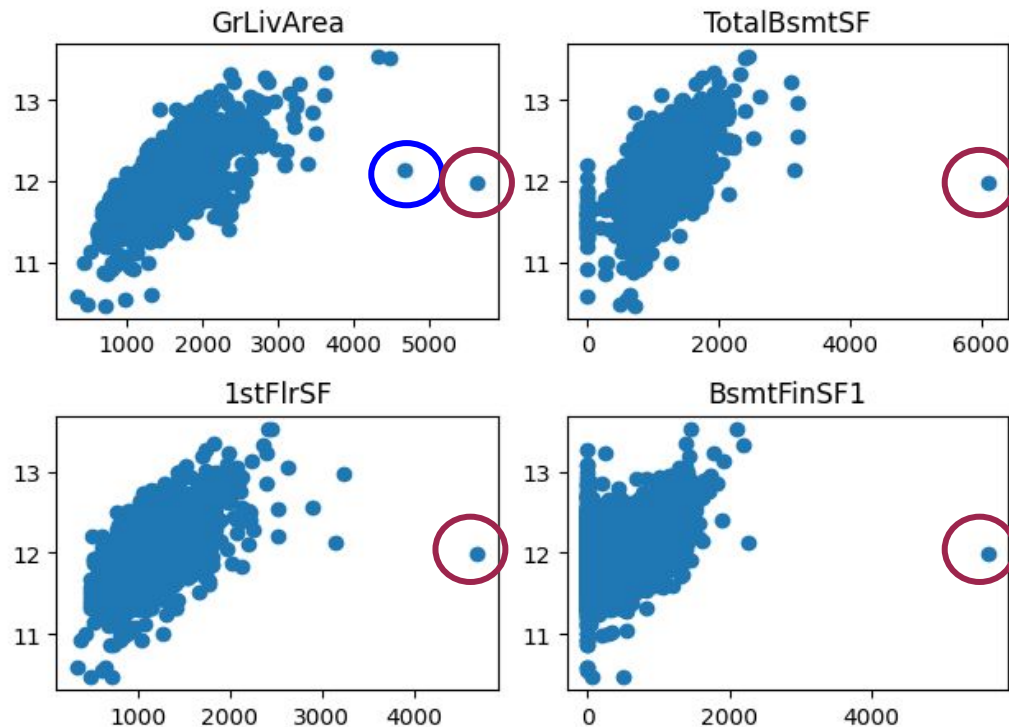
- Distribución porcentual de las categorías de cada variable categórica



Exploración de variables: Categóricas. Conclusiones

- **Utilities:** Está súper descompensada. La descartamos
 - **MoSold:** No se presentan diferencias significativas entre las categorías con SalePrice
 - **LandSlope:** No se presentan diferencias significativas entre las categorías con SalePrice y está muy descompensada. La descartamos.
 - **Street:** Muy descompensada. La descartamos.
 - **Condition2:** Muy descompensada. El 99% es una categoría. La descartamos.
 - **PoolQC:** Muy descompensada. La descartamos.
 - **MiscFeature:** Muy descompensada. La descartamos.
 - **RoofMatl:** Muy descompensada. La descartamos.
 - **Functional:** Muy descompensada. La descartamos.
 - **LotConfig:** Probaremos a hacer solo dos categorías
 - Las variables **MSSubClass**, **OverallCond**, **OverallQual** las consideraremos numéricas en lugar de categóricas.
- 

Exploración de variables: Outliers



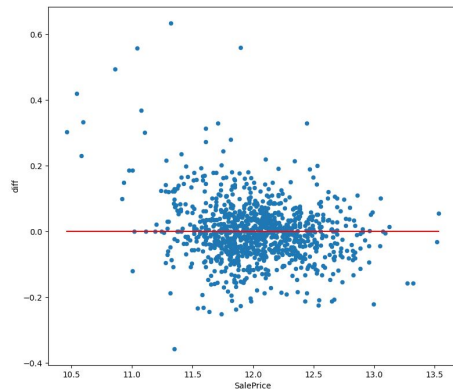
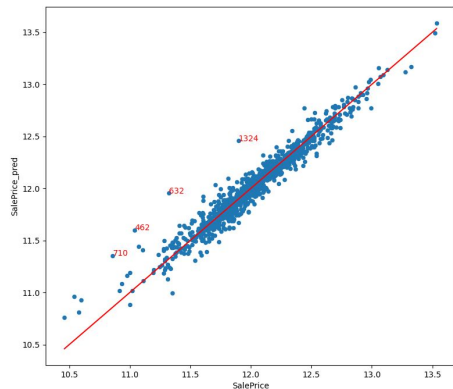
- Hay claramente dos outliers en las variables GrLivArea, Total BsmtSF, 1stFlrSF y BsmtFinSF1



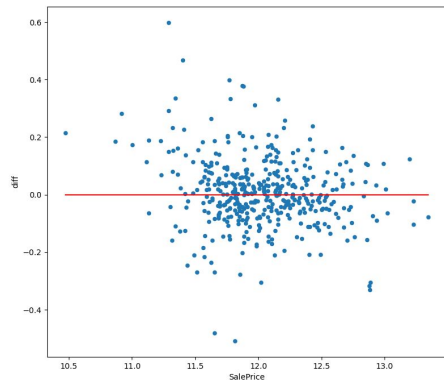
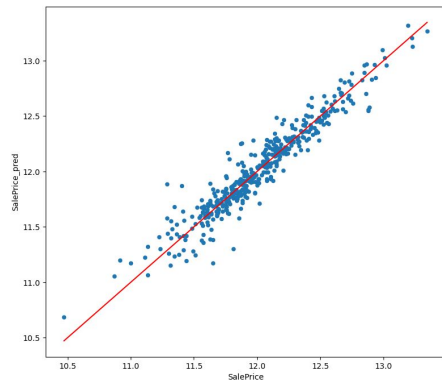
Modelos probados

Modelos: Regresión lineal

TRAIN



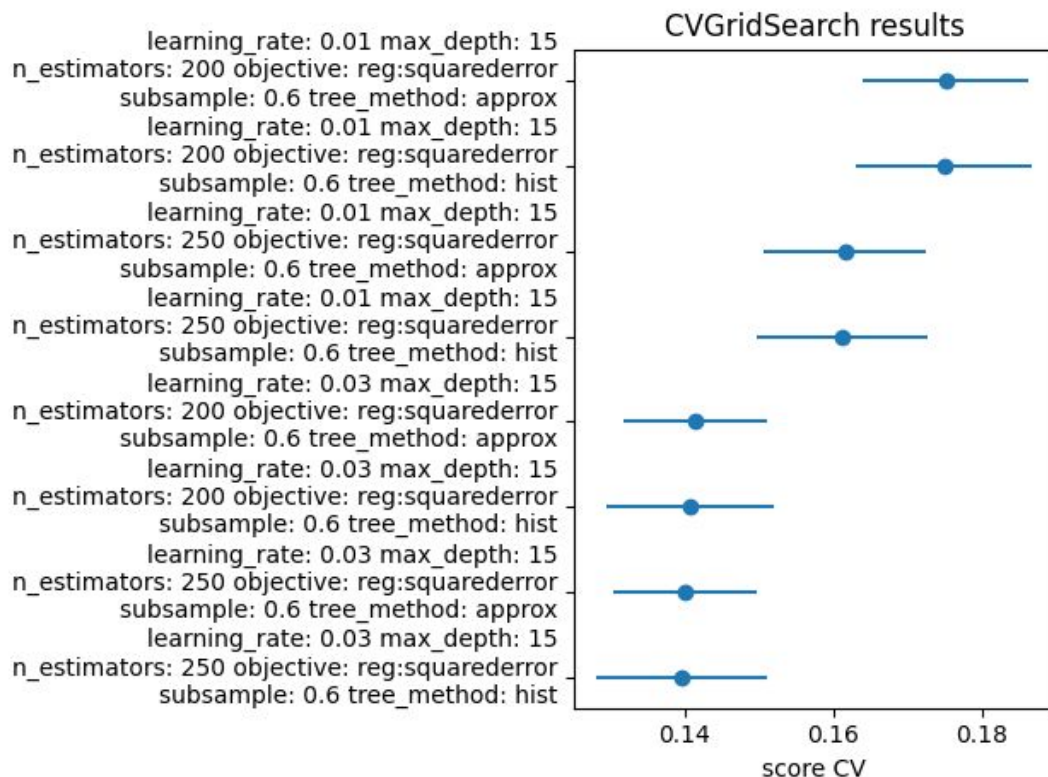
TEST



- TRAIN RMSE: 0.0925147803759712
- TEST RMSE: 0.12230550765589478

El modelo es consistente entre
entrenamiento y test

Modelos: XGBoost

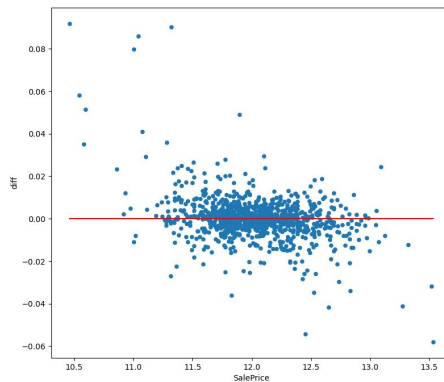
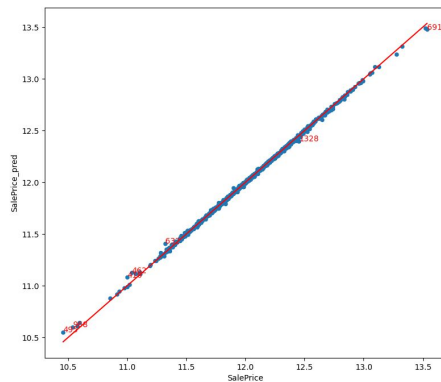


Mejores hiperparámetros

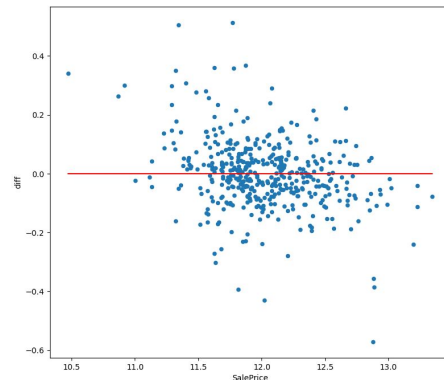
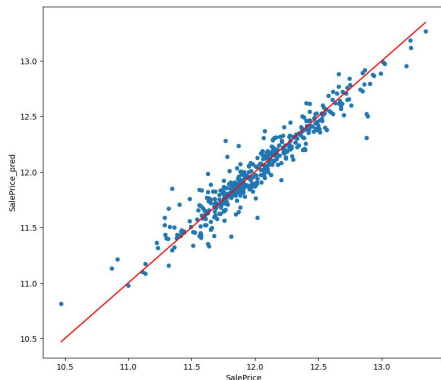
'learning_rate': 0.03,
'max_depth': 15,
'n_estimators': 250,
'subsample': 0.6,
'tree_method': 'hist'

Modelos: XGBoost

TRAIN



TEST



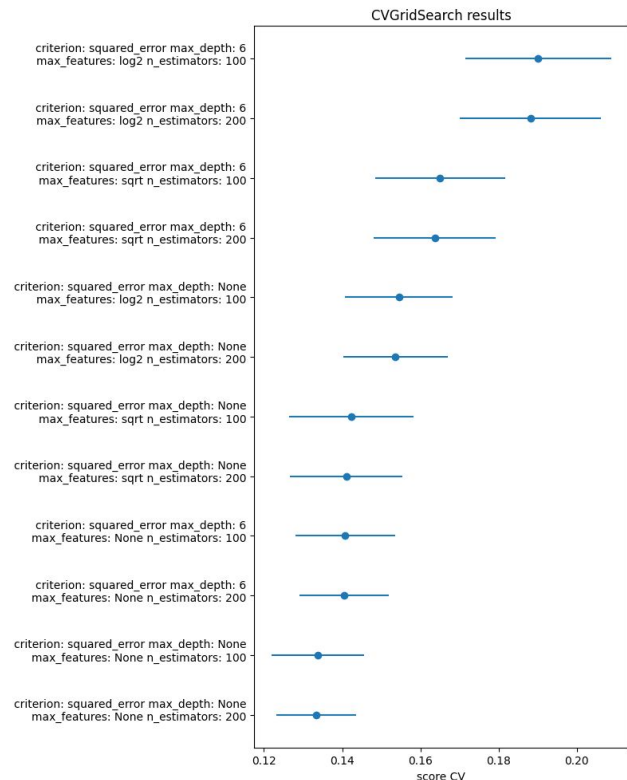
- TRAIN RMSE: 0.0107338998173822
- TEST RMSE: 0.11962607863350601

Se aprecia un claro overfitting.

Modelos: Random Forest

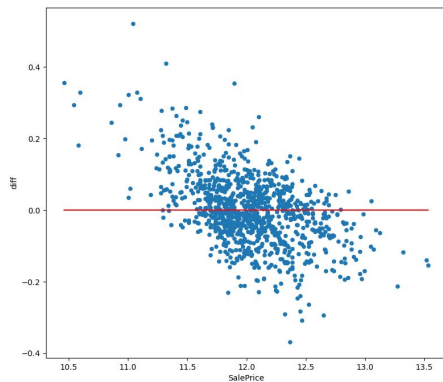
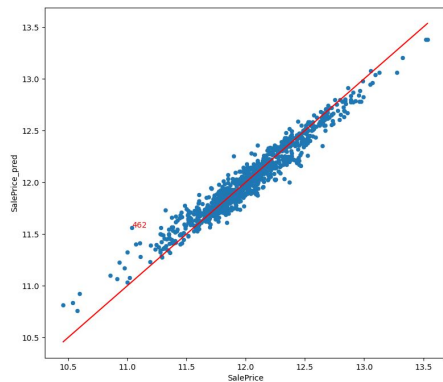
Mejores hiperparámetros

'random_state': 42,
'criterion': 'squared_error',
'max_depth': None,
'max_features': None,
'n_estimators': 200

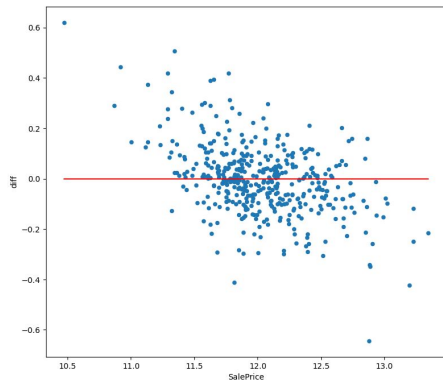
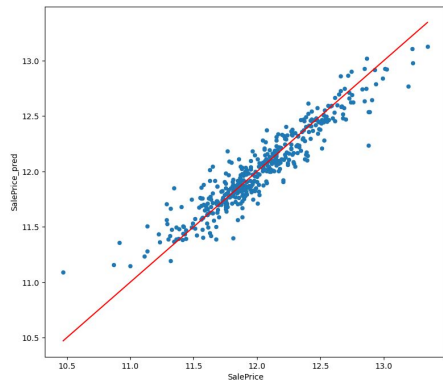


Modelos: Random Forest

TRAIN



TEST



- TRAIN RMSE: 0.0999023163879834
- TEST RMSE: 0.14070783357538005

Ya no hay tanto overfitting.

Modelos: Regularización Lasso (L1)

$$\vec{\hat{\beta}} = \min_{\vec{\beta}} \left[(\vec{y} - X\vec{\beta})^T (\vec{y} - X\vec{\beta}) + \alpha ||\vec{\beta}||_1 \right]$$

Mejores hiperparámetros

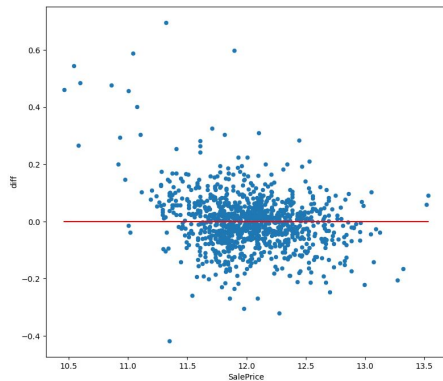
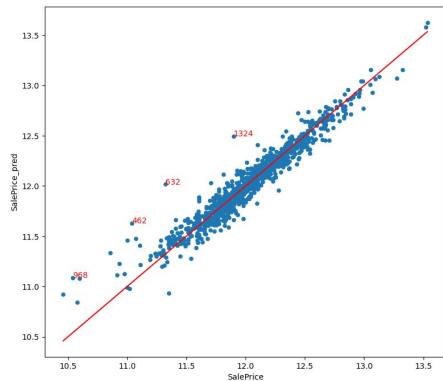
Mejor $\alpha = 0.0005$

Ha seleccionado 113 variables y ha eliminado las restantes 144 variables

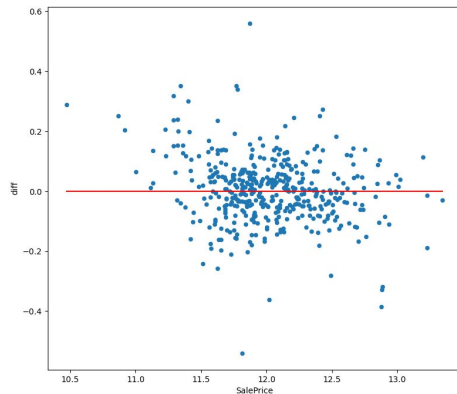
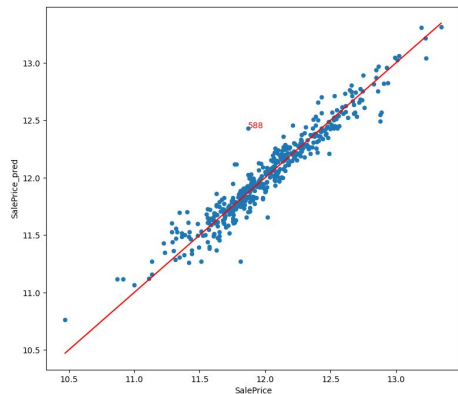


Modelos: Regularización Lasso (L1)

TRAIN



TEST



- TRAIN RMSE: 0.1006519060165124
- TEST RMSE: 0.11037112866301352

El modelo es consistente entre
entrenamiento y test

Modelos: Regularización Ridge (L2)

$$\hat{\vec{\beta}} = \min_{\vec{\beta}} \left[(\vec{y} - X\vec{\beta})^T (\vec{y} - X\vec{\beta}) + \alpha ||\vec{\beta}||_2 \right]$$

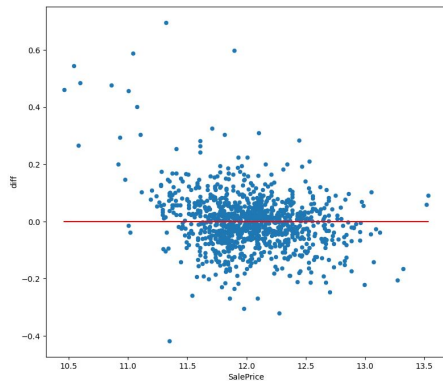
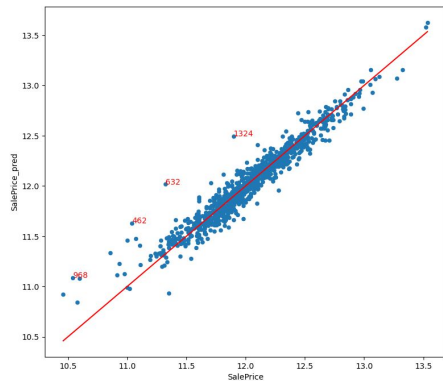
Mejores hiperparámetros

Mejor $\alpha = 20$

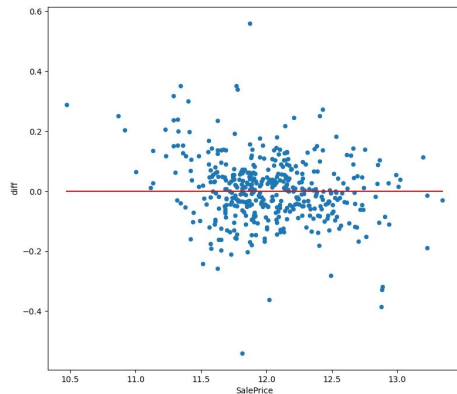
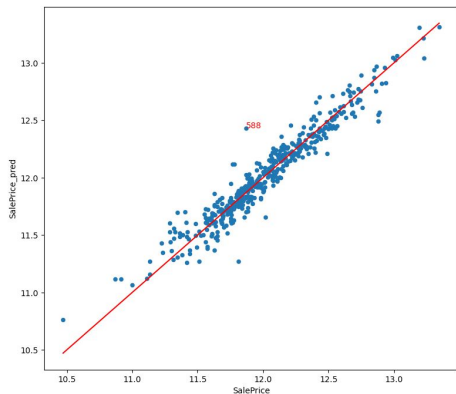


Modelos: Regularización Ridge (L2)

TRAIN



TEST



- TRAIN RMSE: 0.10086999337588502
- TEST RMSE: 0.1122128126862731

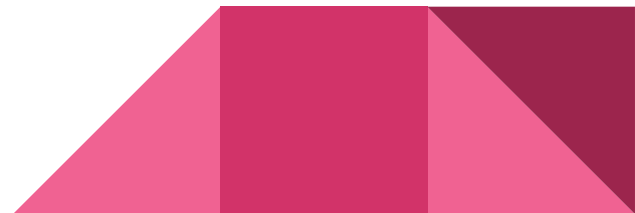
El modelo es consistente entre
entrenamiento y test

Modelos: Regularización ElasticNet (L1 y L2)

$$\vec{\hat{\beta}} = \min_{\vec{\beta}} \left[(\vec{y} - X\vec{\beta})^T (\vec{y} - X\vec{\beta}) + \alpha \left(L_1 \|\vec{\beta}\|_1 + (1 - L_1) \|\vec{\beta}\|_2 \right) \right]$$

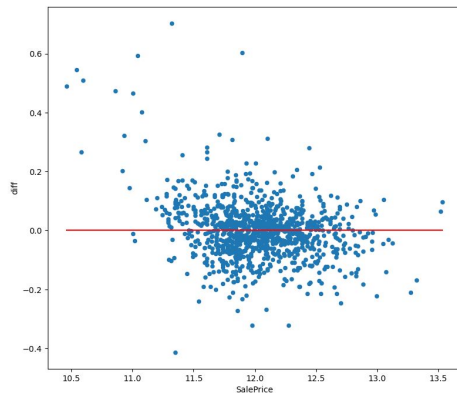
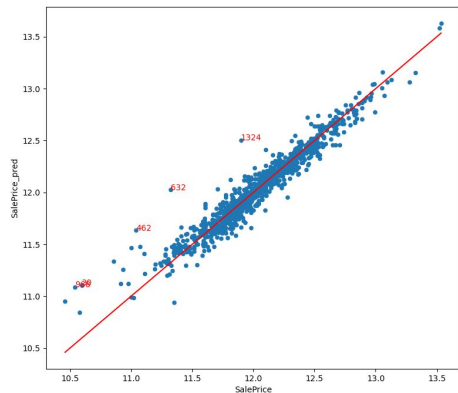
Mejores hiperparámetros

$\alpha = 0.001$
 $L1 = 0.55$

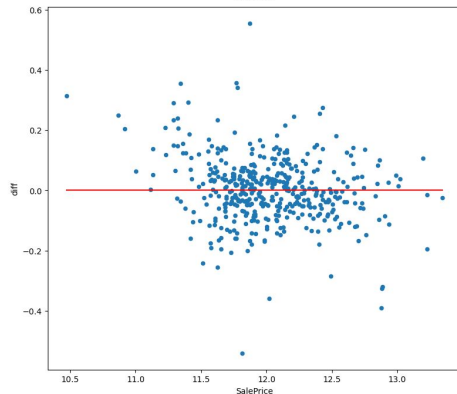
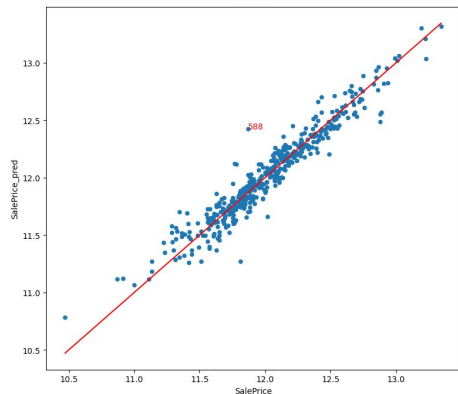


Modelos: Regularización ElasticNet (L1 y L2)

TRAIN



TEST

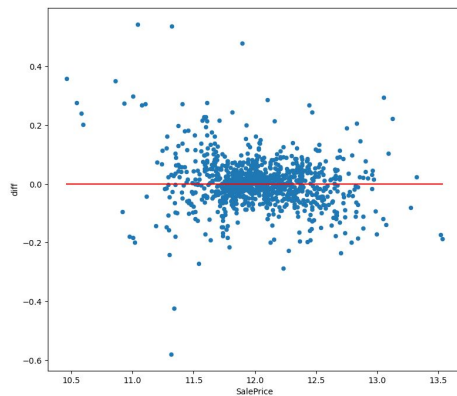
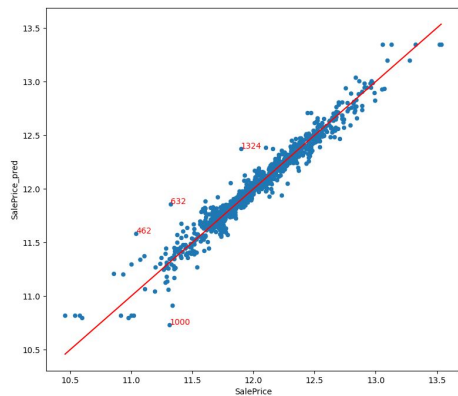


- TRAIN RMSE: 0.10145996319218037
- TEST RMSE: 0.11029365392796532

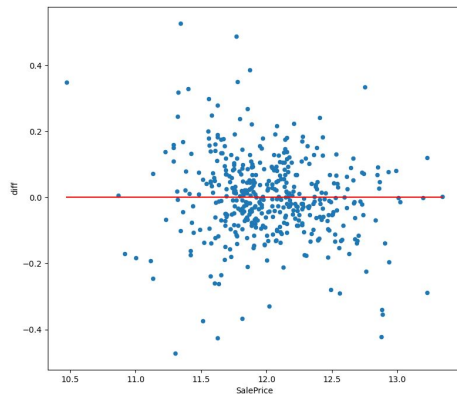
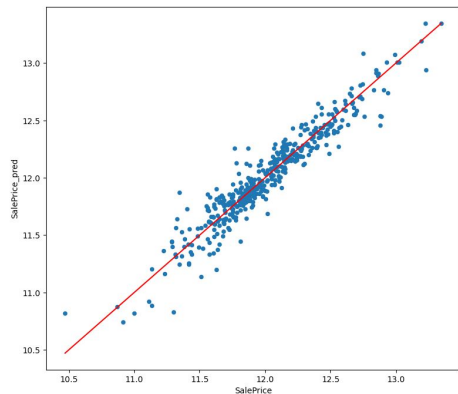
El modelo es consistente entre
entrenamiento y test

Modelos: Ensemble

TRAIN



TEST



Estimadores:

- XGB
- Random Forest
- Ridge
- ElasticNet

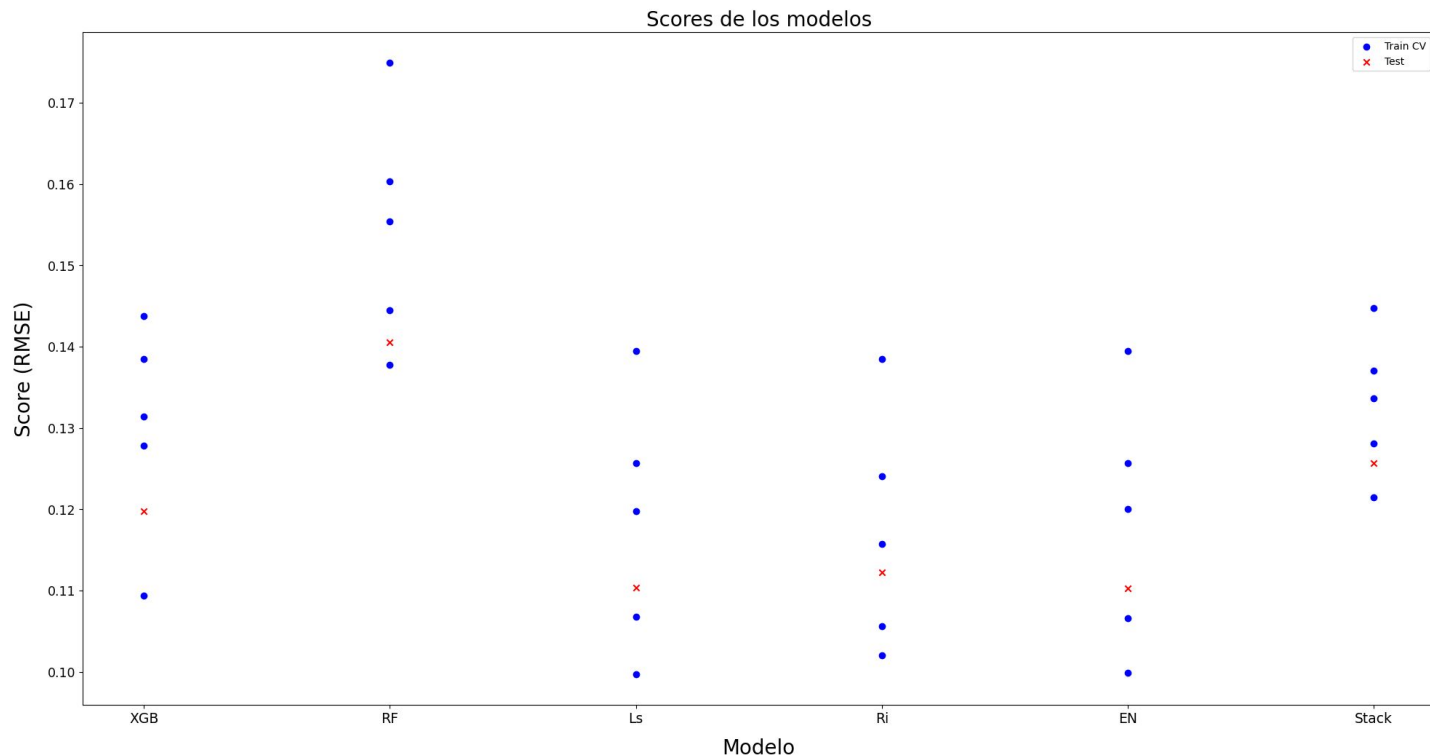
Estimador final:

- XGB

- TRAIN RMSE: 0.08591334045444235
- TEST RMSE: 0.12567562743883187

Comparativa de modelos

Modelos: Comparativa



Validación
cruzada 5-fold
en dataset de
entrenamiento
en azul

Score en
dataset de test
en rojo



Modelo final escogido

Modelo final escogido



0.13327

Seleccionamos el modelo Ridge (R_i). Tiene el promedio parecido a L_s y EN pero su dispersión es menor.

[Tabla](#)

