

```
Baseline LinearRegression  
R2 score train 0.8957008271017817  
MAE train 1198.4678961500135  
MSE train 2705129.423081414  
RMSE train 1644.7277656443373
```

```
R2 score test 0.899438024100912  
MAE test 1460.7567168117605  
MSE test 3174097.3539761053  
RMSE test 1781.5996615334504
```

Parece haber un poco de overfitting, pero podría ser variabilidad estadística.

```
LinearRegression MinMax -> No afecta al ser una transformación lineal  
R2 score train 0.8957008271017817  
MAE train 1198.4678961500133  
MSE train 2705129.423081414  
RMSE train 1644.7277656443373
```

```
R2 score test 0.899438024100912  
MAE test 1460.7567168117603  
MSE test 3174097.353976106  
RMSE test 1781.5996615334507
```

```
Ridge alpha=0.1 MinMax  
R2 score train 0.8956542818544961  
MAE train 1199.2163914277749  
MSE train 2706336.632251323  
RMSE train 1645.0947183221163
```

```
R2 score test 0.8986591607392291  
MAE test 1464.7443890153074  
MSE test 3198681.080711023  
RMSE test 1788.48569485781
```

De hecho Ridge alpha = 0.1 no mejora la performance.

Ahora probamos a hacer un barrido en alphas:

Ridge alpha={0.0001->1000} MinMax

MSE test:

```
55] np.min(ridge_error)
   ✓ 0.0s
· np.float64(30939176.903210662)

56] np.argmin(ridge_error)
   ✓ 0.0s
· np.int64(0)

· ^
  alphas[int(np.argmin(ridge_error))]
57] ✓ 0.0s
· np.float64(0.0001)
```

El mejor error nos lo da para la alpha = 0.001, lo cual quiere decir que no estamos aplicando regularización, cosa que ya intuimos de la diferencia entre MSE train y test del LinearRegressor.

Para Ridge, Lasso y ElasticNet, puede mejorar un Scaler la performance del modelo?

Sí, aplicar una transformación **MinMaxScaler** puede ser útil para mejorar el rendimiento de un modelo **Ridge Regression**. Esto se debe a que **Ridge Regression** incluye una penalización basada en la magnitud de los coeficientes, y las diferencias en la escala de las variables pueden desbalancear el impacto de la regularización.

¿Por qué MinMaxScaler?

La transformación **MinMaxScaler** escala los valores de las características a un rango definido, típicamente $[0, 1]$. Esto ayuda a que las características tengan una escala uniforme, lo que:

1. **Facilita la convergencia del optimizador:** Los métodos de optimización en modelos de regresión como Ridge son más eficientes cuando las características están en escalas comparables.
2. **Mejora el desempeño de la regularización L_2 :** La penalización de Ridge está directamente relacionada con los coeficientes de las características, y escalarlas evita que las variables con mayores magnitudes dominen el modelo.

CONTAMINACIÓN ENTRE TRAIN Y TEST

La fórmula general de MinMaxScaler para transformar una característica x es:

$$x' = \frac{x - \min(x)}{\max(x) - \min(x)}$$

Polinomio grado 2 MinMax -> Mejora, poco overfitting

```
R2 score train 0.986104518792419  
MAE train 408.97219513550624  
MSE train 360396.67447012226  
RMSE train 600.3304710491733
```

```
R2 score test 0.9869181490609605  
MAE test 526.1794444043729  
MSE test 412910.228537899  
RMSE test 642.5809120553605
```

Polinomio grado 3 MinMax -> Mejora, poco overfitting

```
R2 score train 0.9916695295589968  
MAE train 317.7015419249498  
MSE train 216061.16397546846  
RMSE train 464.82379884798115
```

```
R2 score test 0.9906674629185213  
MAE test 388.6817048651651  
MSE test 294568.40909660456  
RMSE test 542.741567503913
```

Polinomio grado 4 MinMax-> No mejora en train, y overfitting

```
R2 score train 0.9914394563593426  
MAE train 367.7  
MSE train 222028.4  
RMSE train 471.19889643334267
```

```
R2 score test 0.8320000554436232  
MAE test 998.3  
MSE test 5302682.0  
RMSE test 2302.75530614957
```

No tiene sentido aplicar regularización al polinomio de grado 4, porque ya de salida tiene peores resultados que el polinomio de grado 3.

Aún así probamos con Ridge y el spread de alphas:
27495867.593038023

