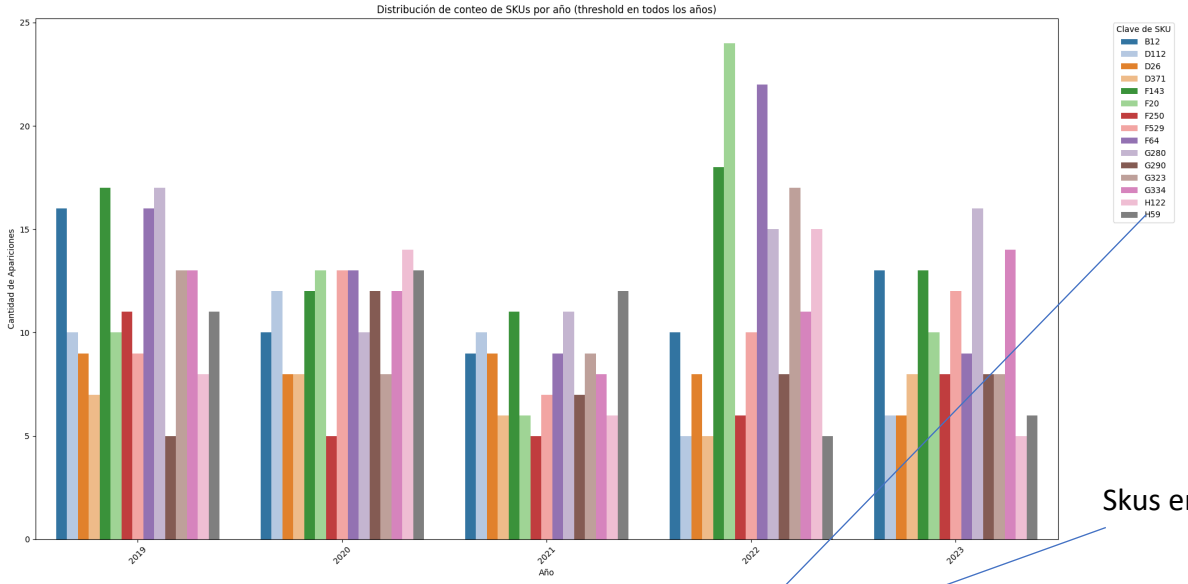
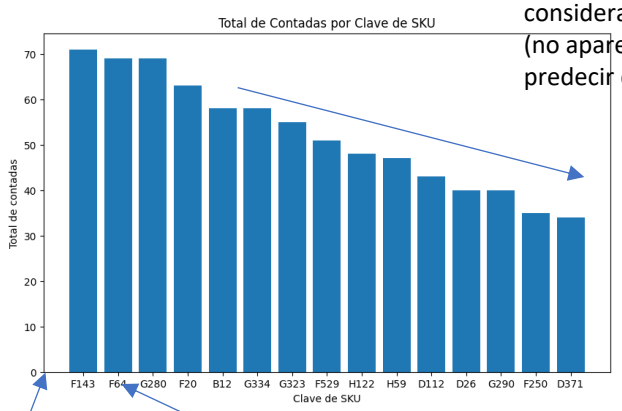
Por qué escogimos un solo modelo?




```
# Filtrar SKUs que aparecen 'threshold' o más veces cada año
threshold = 5
```



Son pocos registros para considerarlos en el ML (no aparecen en amarillo del excel a predecir de todos modos).

Skus en Excel amarillos

GANADOR (UNICO que pasa el criterio de filtración)

```
import numpy as np

# Asumiendo que estos son tus dos arrays (o listas):
array1 = val_final['Clave de SKU'].unique()
array2 = sku_total_counts_sorted['Clave de SKU']

# Calcular la intersección de los dos arrays
intersection = np.intersect1d(array1, array2)
print(intersection)
print('SOLO APARECE UNO')

```

```
['F64']
SOLO APARECE UNO
```

El dataset tenía ciertos problemas

- hay un dato faltante en el indice 4248 en la columna de `df['Unidades vendidas']`
- Se elminará registro

Quiero decir que también en el conjunto de datos no amarillo había información correspondiente al año 2023. Estos los utilicé para entrenar el modelo

Sin embargo, al realizar la intersección de los apredecir (lo borrado en amariilo) realmente sólo F64 apareció.

Es por eso debido a mi criterio de filtración es que escogí un sku F64 a predecir, descartando los demás skus en la figura de barras presentada.

Hay que recalcar que no podemos hacer un modelo con escasa información, debido aque si el criterio de selección fue considerar la temporalidad también.

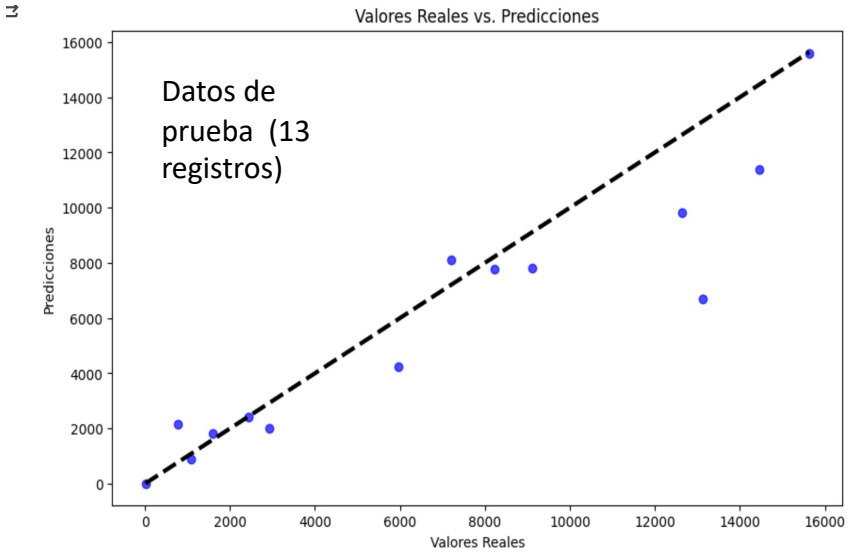
- Entiendo la necesidad de la empresa de querer tomar muchos skus y ver su predicción.

- Considero que es un razonamiento aceptable para la generación de un modelo ML en donde queremos el modelo evite sobre aprender de los pocos datos que se tenga y no pueda generalizar.

Predicción F64

Se buscaron los mejores hiperparametros del modelo XGBOOST en modo regressión
Con el dataset de prueba… (aunque los datos son muy poco, se intenta no sobre ajustar el metodo)

- El dataset de prueba era el 20% (13 registros) y el 80% de entrenamiento
- Se realizó una optimización de hiperparámetros sencilla para buscar los mejores parámetros
- Utilicé todo el conjunto de datos porque los estimadores de XGBoost, especialmente en el contexto de árboles de decisión, son capaces de manejar datos multidimensionales(143 aprox). Establecer un valor más alto para el parámetro 'n_estimators' es parte de esta estrategia, relaciones más complicadas.



RMSE: 2196.662079
R^2: 0.829090

- **Pruba de validación**

Dato de validación (en excel en amarillo)

```
[249] 1  # Make predictions
      2  predictions = xg_reg.predict(VAL_F64)
      3  predictions
```

```
array([1235.1125], dtype=float32)
```

	Año	Precio Regular	Precio Ofertado	Descuento	Incidencia	Vendedores Activos	Ca
9714	2023	429.9	386.9	0.100023	0.0	87148	…

Explicative machine learning

- Por qué el modelo tomó esas desiciones?

- **Para ello utilicé SHAP values, que proviene de la teoría del juego.**
- **Cualquier método de machine learning puede utilizar SHAP para cuantificar la contribución de una variable para cada registro en la clasificación o regresión**
- **Escogí XGBoost debido a su particularidad de ser un método no lineal (si son lineales lo puede capturar), bastante robusto y eficaz**

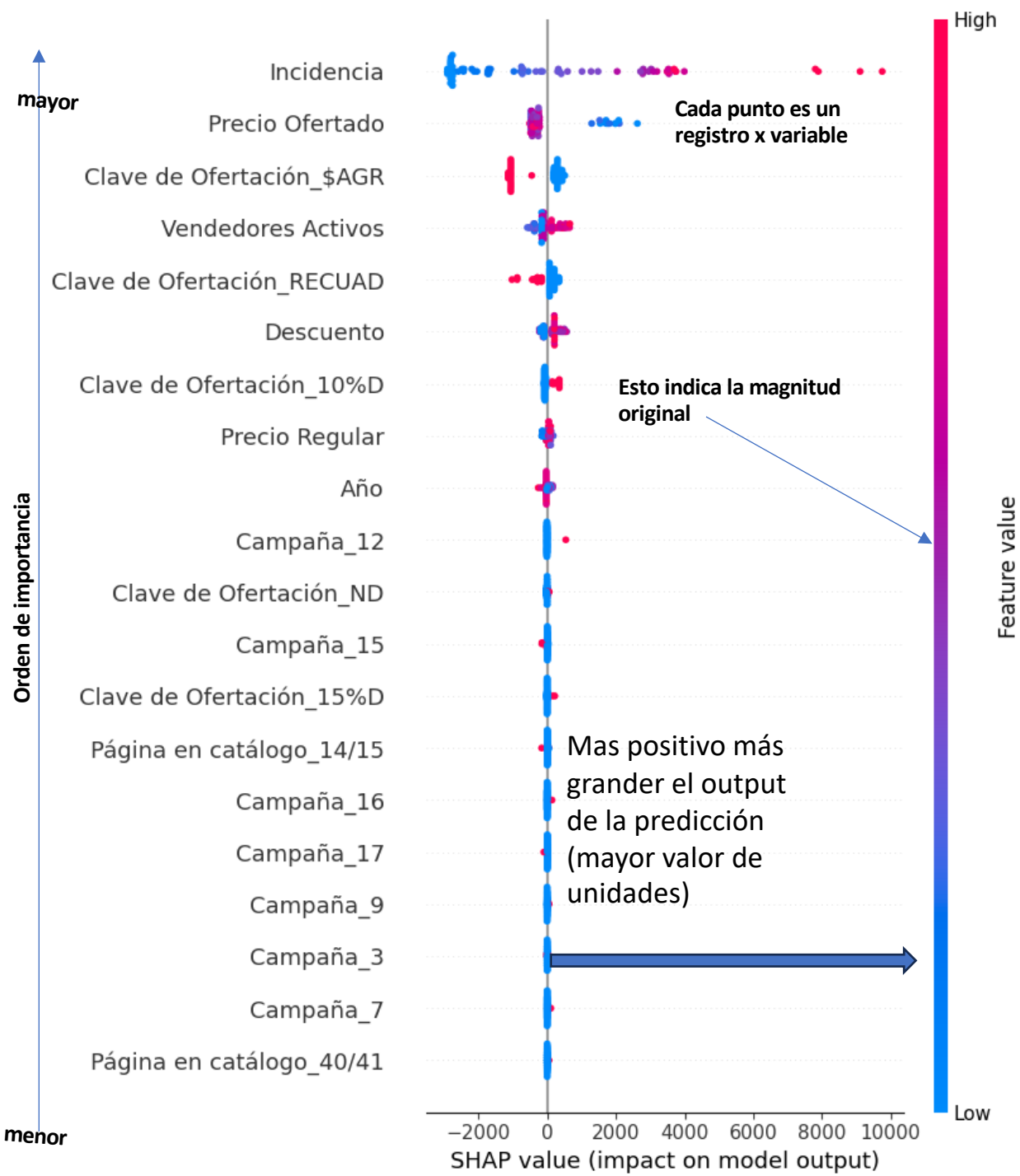
En esta gráfica de explicaciones locales, podemos observar que para el modelo **XGBoost**, el impacto que tienen más peso (SHAP values) por muestra en las variables al menos para este modelo son: **Incidencia**, **Precio Ofertado** y **Clave de Ofertación AGR Vendedores activos**

Si el modelo no tuviera los primeros dos en el top realmente su performance caeria bastante (datos no mostrados)

```
[249]  1  # Make predictions
       2  predictions = xg_reg.predict(VAL_F64)
       3  predictions
```

```
array([1235.1125], dtype=float32)
```

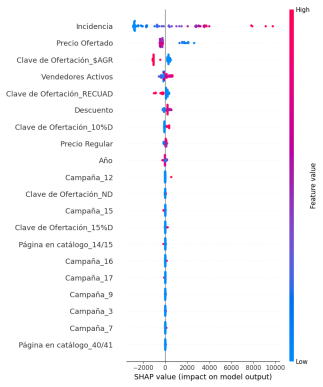
	Año	Precio Regular	Precio Ofertado	Descuento	Incidencia	Vendedores Activos	Ca
9714	2023	429.9	386.9	0.100023	0.0	87148	



MANIFOLD LEARNING + SHAP (CLUSTERING SUPERVISADO)

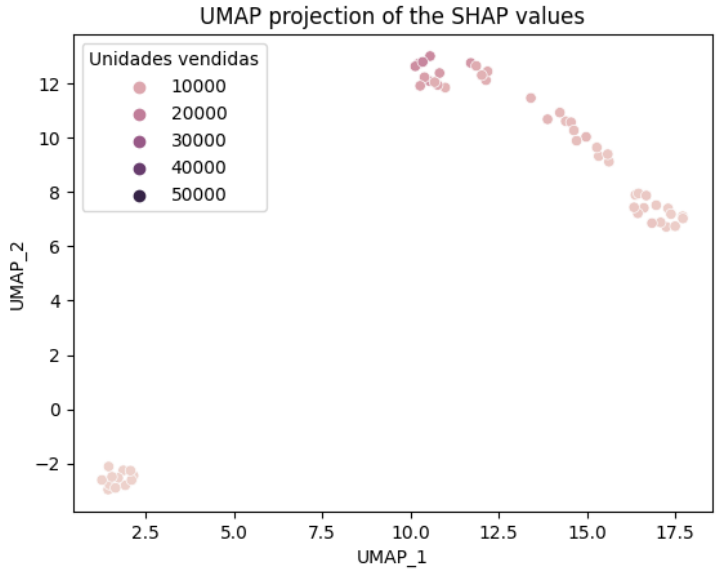
- Si aplicamos un metodo de reducción de dimensiones como UMAP a los pesos podemos tener un espacio explicatorio del modelo.

- En donde se clustericen los datos en función a la importancia del modelo de XGBOOST.

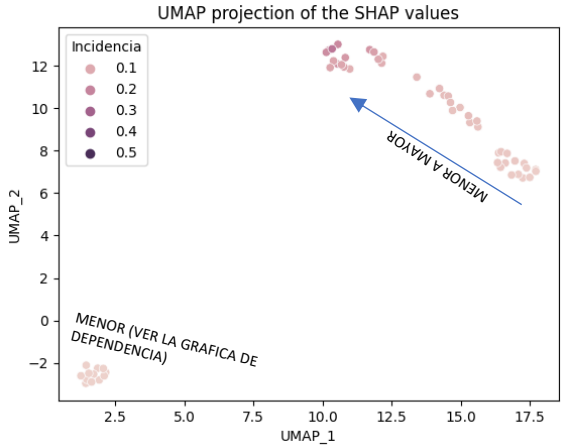


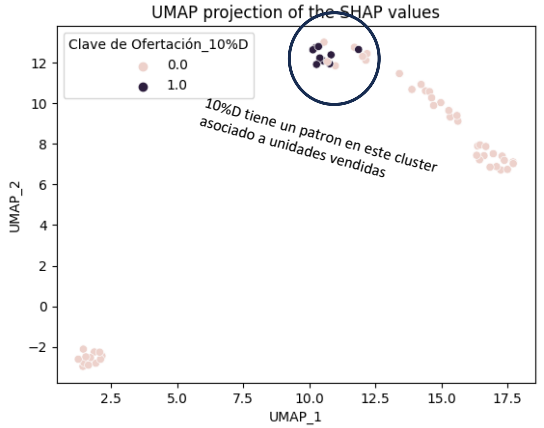
SHAP VALUES

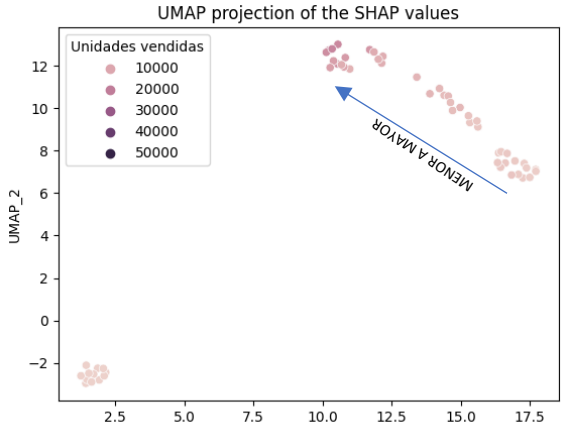
UMAP

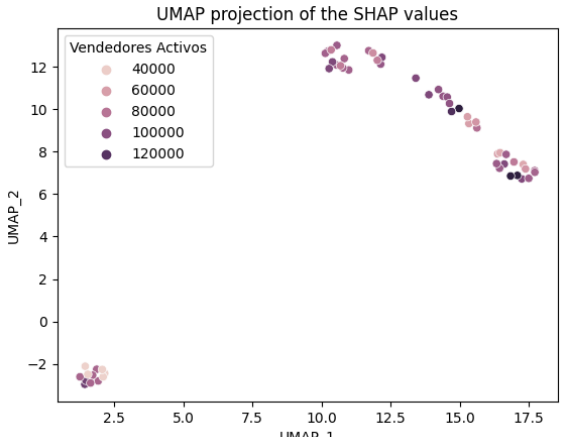


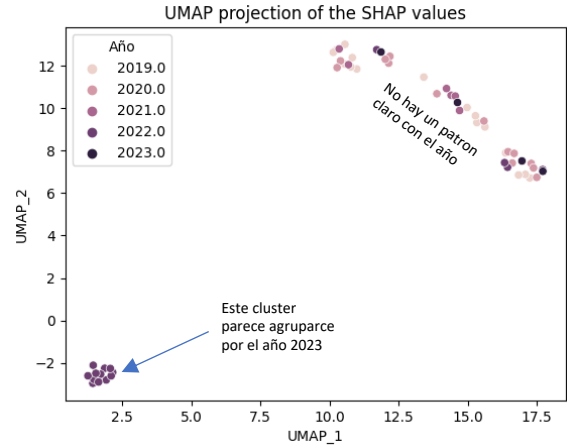
Supervised clustering= (shaps_XGBOOST (SUPERVISADO)+ umap (no supervisado)

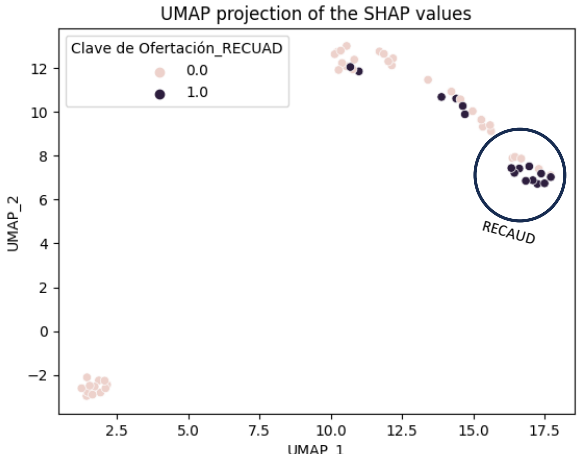


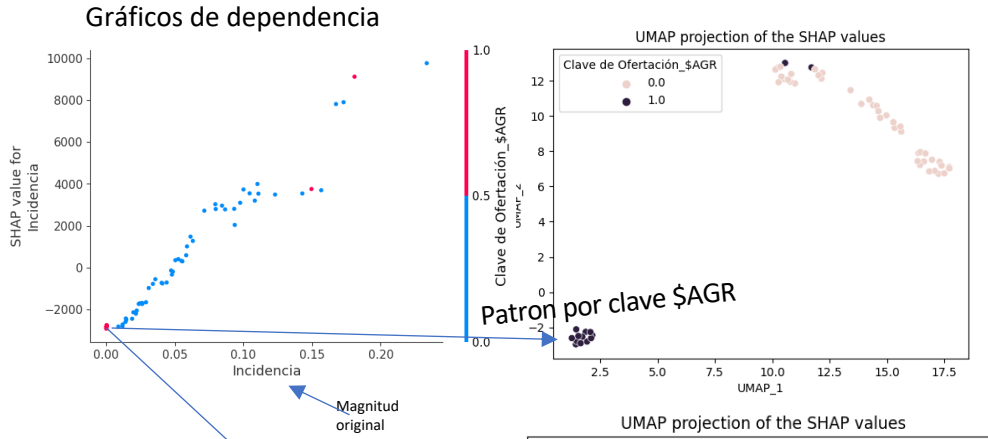


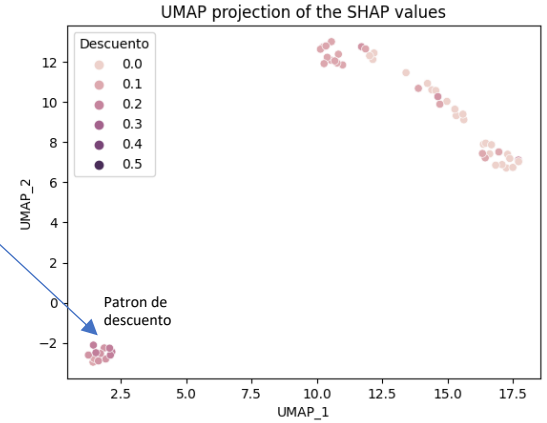


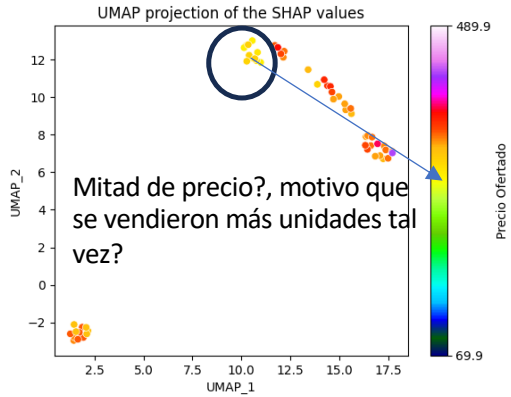












El espacio UMAP que vemos es por la simitud de los pesos para la regresión

- Visualmente podemos ver varios patrones y comportamientos
- Differentes clusters como estos pueden afectar la toma de desiciones en la regresión.

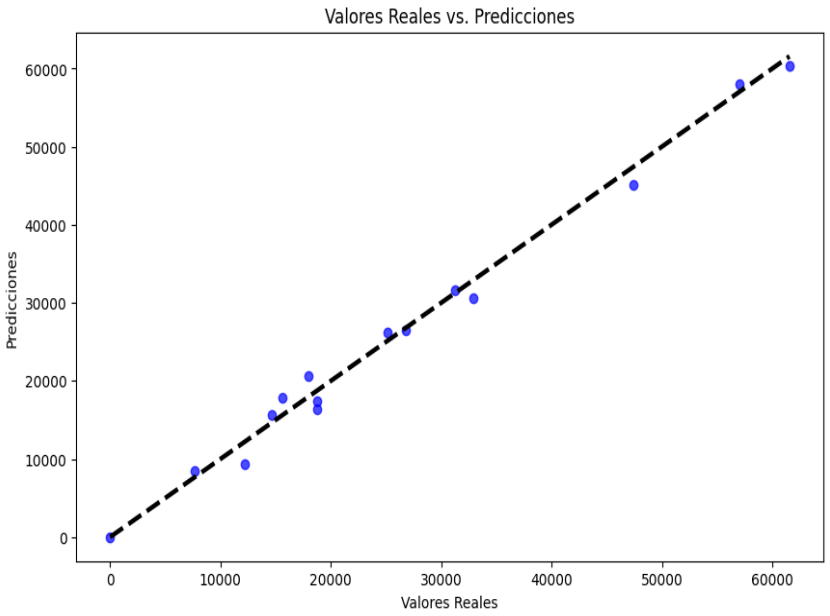
OTRO MODELO

Modelo F143

Este es SKU MAS FRECUENTE pero no aparece en el 2023 a predecir (en amarillo), pero lo hice de cualquier forma. Igual los más presentes siempre son **Incidencia vendedores Activos Precio ofertado**

Un método de reducción de los pesos podemos ver como se agrupan las muestras y podemos visualizar algunas variables

Modelo F143



RMSE: 1695.672556
R^2: 0.990131