IM-Analytics Qualification Test: Reporte del Problema del Lapidarista

Aplicante: Ikram V. Saucedo Rocha

1. Introducción

Contexto: Ha habido un robo de diamantes en el banco y es necessria la ayuda de un científico de datos para que por medio de modelos predictivos estime el valor de lo robado ya que es necesario verificar o desmentir la estimación de Krenk quien fue el último en ver dichos diamantes robados.

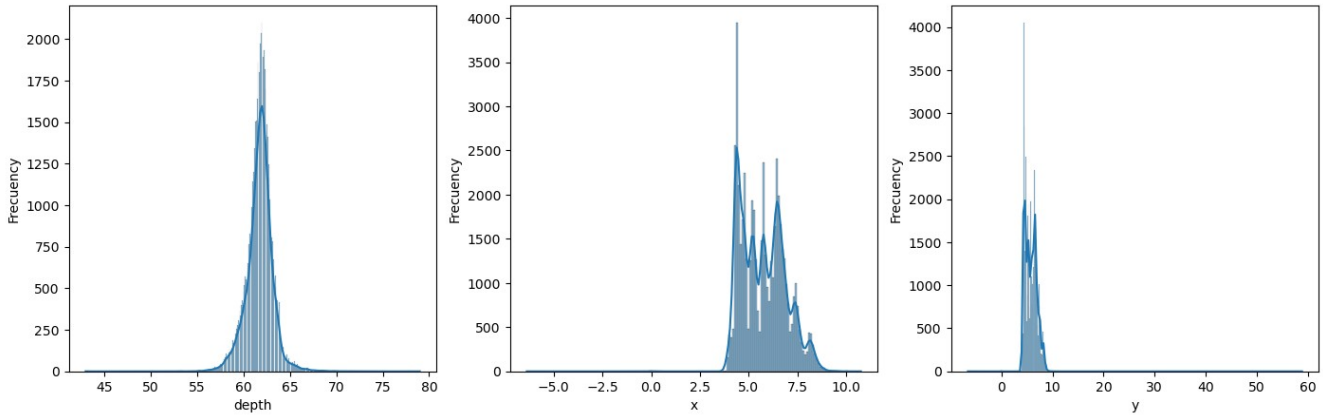
Objetivo: Estimar el valor de los diamantes robados utilizando el dataset proporcionado con datos existentes de diamantes y con este generar un modelo de Machine Learning que permita dar una estimación precisa de la pérdida del banco.

2. Exploración y Limpieza de Datos (EDA)

Carga de Datos: Los datos fueron cargados por medio de la librería pandas en formato csv, se combinó en un solo dataframe las características de los diamantes y las coordenadas de los mismos. Se hizo una exploración rapida y general de los datos observando: estadísticas básicas (valores mínimos, máximos, promedios y cuantiles), existencia de valores nulos por columna e inconsistencias en los tipos de datos, de esto último se encontró que la variable latitud contenía un error en un renglón al contener una letra por lo que se hizo uso de expresiones regulares para eliminar cualquier letra contenida en esta columna, posteriormente se convirtió a tipo flotante.

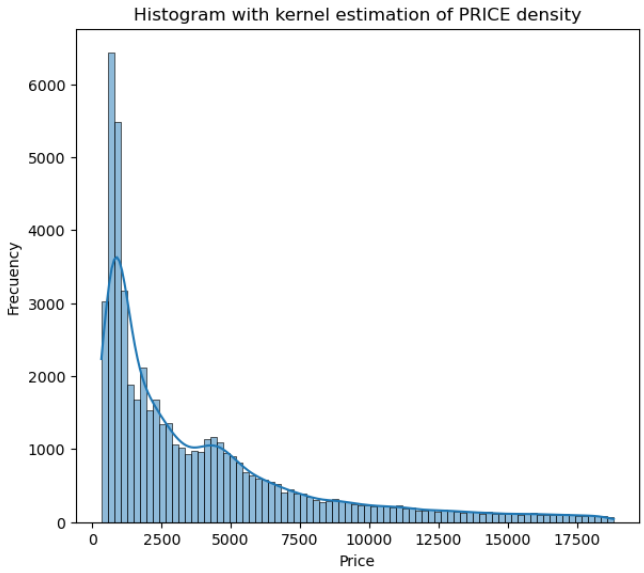
Análisis Descriptivo:

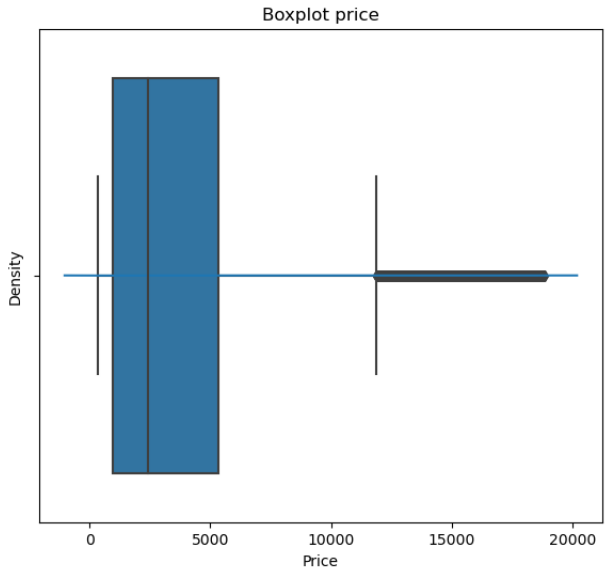
Manejo de valores faltantes. Para la imputación de valores faltantes se tomó en cuenta la distribución de las variables, aquellas que tuvieron una distribución normal se imputaron con la media y aquellas que tuvieron una distribución diferente a la normal, con la mediana.



Solo tres variables presentaron valores nulos (depth, x, y). Como se puede observar, depth cuenta con una distribución normal por lo que se decidió imputar con la media, mientras que las variables x y y se imputaron con la mediana debido a que no cuentan con una distribución normal.

Análisis de las variables numéricas. Nuestro principal interés es saber que variables están más fuertemente correlacionadas con nuestra variable objetivo price, por lo tanto se comenzó analizando la distribución de esta mediante un histograma y un diagrama de caja

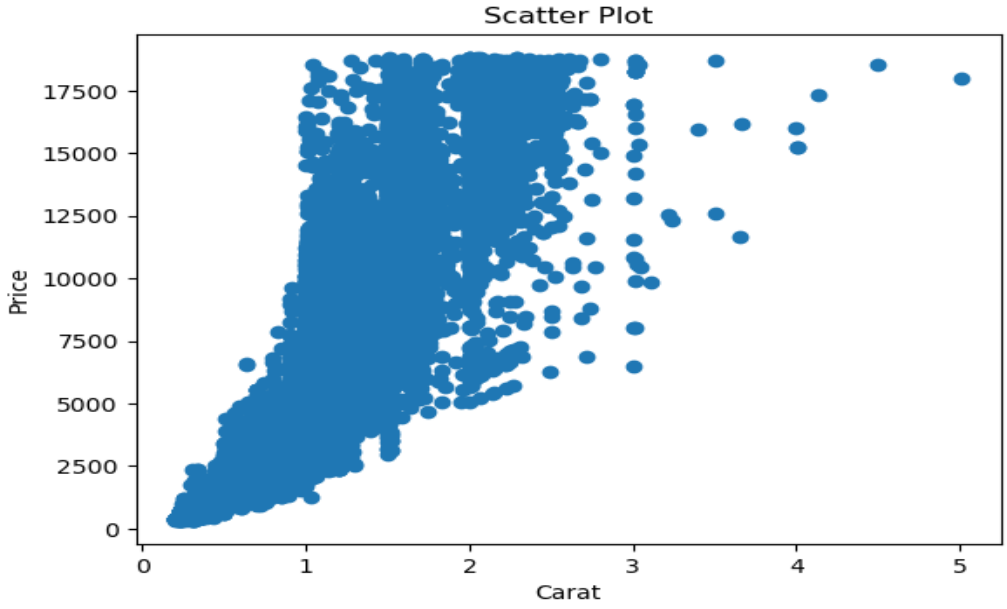


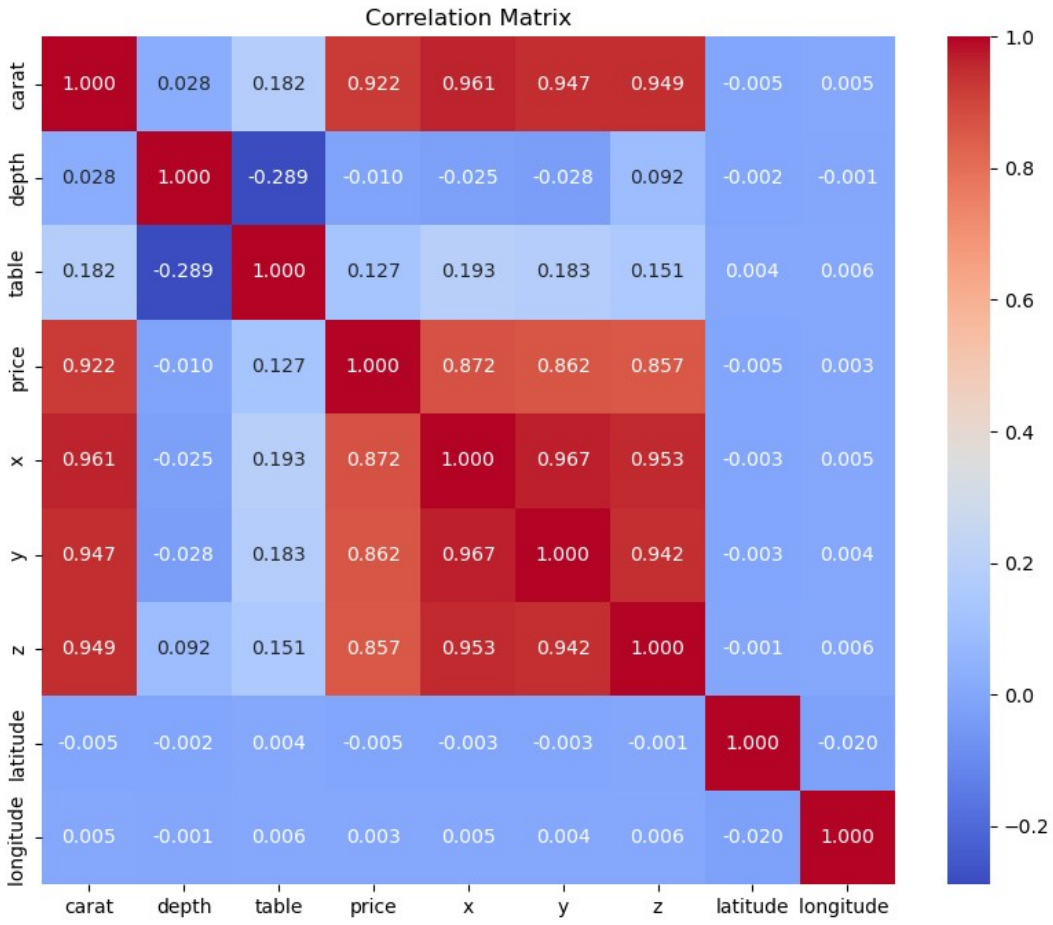


Donde aparentemente precios mayores a 12000 podrían considerarse como atípcos, una forma de trabajar con esto podría ser normalizar esta variable y quizá las demás variables numéricas pero dado que el modelo que se utilizó no es sensible a datos no estandarizados/normalizados, no haría gran diferencia para la eficiencia del modelo pero sería necesario en caso de utilizar por ejemplo, regresión lineal múltiple o algún otro modelo más sensible a relaciones lineales.

Por medio de un diagrama de dispersión (price vs carat) y una matriz de correlación podemos inferir lo siguiente:

- Las variables que tienen mayor correlación con el precio del diamante son carat y las dimensiones del mismo (x,y,z).
- Las variables que no parecen tener correlación significativa con el precio son: depth, table, latitude y longitude.





Limpieza y análisis de las variables categóricas. Las variables categóricas (cut, color, clarity) contenian diversos errores de escritura, se eliminaron los caracteres especiales dejando únicamente los alfanuméricos por medio de la creación de una función. A continuación se puede ver un antes y después de aplicar dicha función

```
df_diam.cut.unique()
```

```
array(['Ideal', 'Premium', 'Good', 'Very Good', 'Fair', 'Very Goo!d',
       "P*'remium", 'I#deal', '#Very Good', 'P?remium', "*'Ideal",
       '!Good', 'Pre!mium', 'Pr?emium', 'Very Go#od', 'Ide&al', 'Ide!al',
       'Id!eal', '&Premium', 'Go?od', 'G#ood', "Very *'Good", "Ide*'al",
       'V&ery Good', '&Ideal', 'Very G#ood'], dtype=object)
```

```
df_diam.cut.unique()
```

```
array(['Ideal', 'Premium', 'Good', 'VeryGood', 'Fair'], dtype=object)
```

La relación entre dichas variables y price se analizó mediante un ANOVA, donde según el p-value obtenido para cada una de estas mostró que las variables cut y clarity están fuertemente relacionadas con nuestra variable objetivo. Así mismo, se codificaron manualmente las categorias de las variables debido a que son de tipo ordinal y al usar álgún otro método usual con alguna librería de python como LaberEncoder no se respetaría la jerarquía de estas.

3. Modelo Predictivo

Selección del Modelo. Se optó por seleccionar el modelo XGBoost por las siguientes razones:

- Tiene la capacidad de manejar relaciones complejas y grandes volúmenes de datos.
- Menos sensible a datos no normalizados o estandarizdos.
- Su velocidad de ejecución es muy buena.
- Es capaz de manejar valores nulos (aunque en este caso particular ya no contamos con valores nulos)

División de Datos. Se realizó una división de los datos en conjuntos de prueba y entrenamiento a través de train_test_split de la librería sklearn. Se tomó el 30% para prueba y 70% para entrenamiento.

Optimización de hiperparámetros. Aunque el modelo con parámetros por defaul ya de por sí mostró muy buenos resultados, se realizó una busqueda aleatoria de hiperparámetros para eficientar las predicciones del modelo por medio de RandomizedSearchCV, cuidando que el espacio de búsqueda fuera pertinente para evitar el sobreajuste. Los hiperparámetros elegidos fueron

```
Best parameters found:  {'subsample': 0.8, 'reg_lambda': 1, 'reg_alpha': 5,
'n_estimators': 200, 'max_depth': 5, 'learning_rate': 0.05, 'gamma': 0,
'colsample_bytree': 1.0}
```

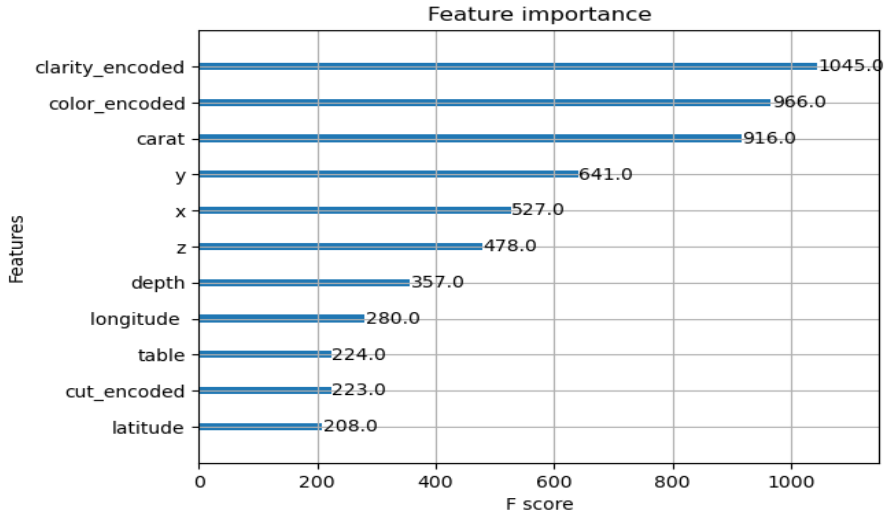
NOTA: Se realizó una prueba con One Hot Encoding para las variables categóricas lo que resultó en un sobreajuste severo (verificado con validación cruzada) debido posiblemente al aumento en la complejidad del modelo, una razón más para preferir una codificación manual de variables categóricas cuando el problema lo permitao

4. Resultados y Hallazgos

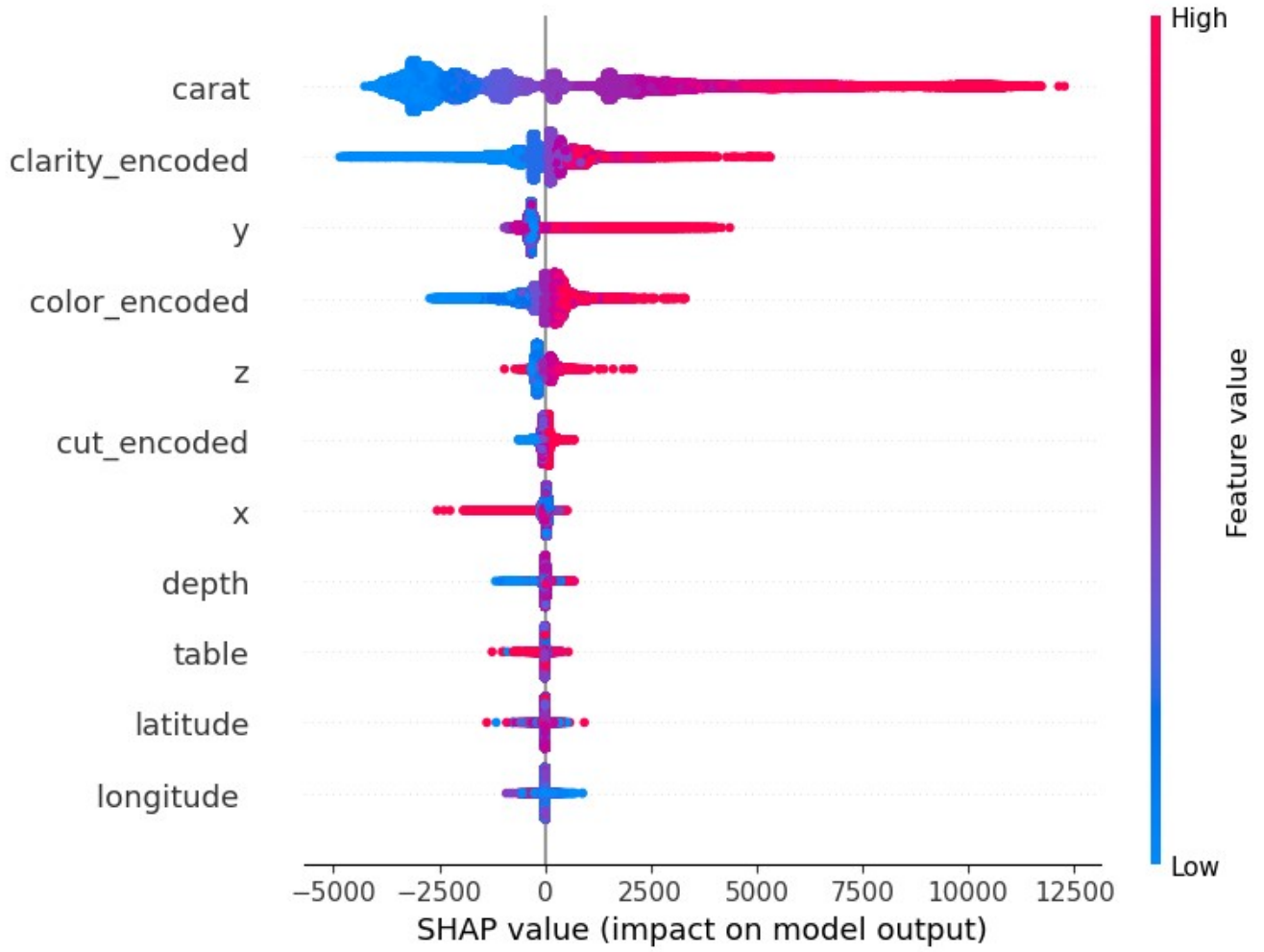
Resultados del Modelo. Las metricas del modelo fueron:

```
Mean Squared Error: 269348.53
R2_score:  0.9826300766454481
```

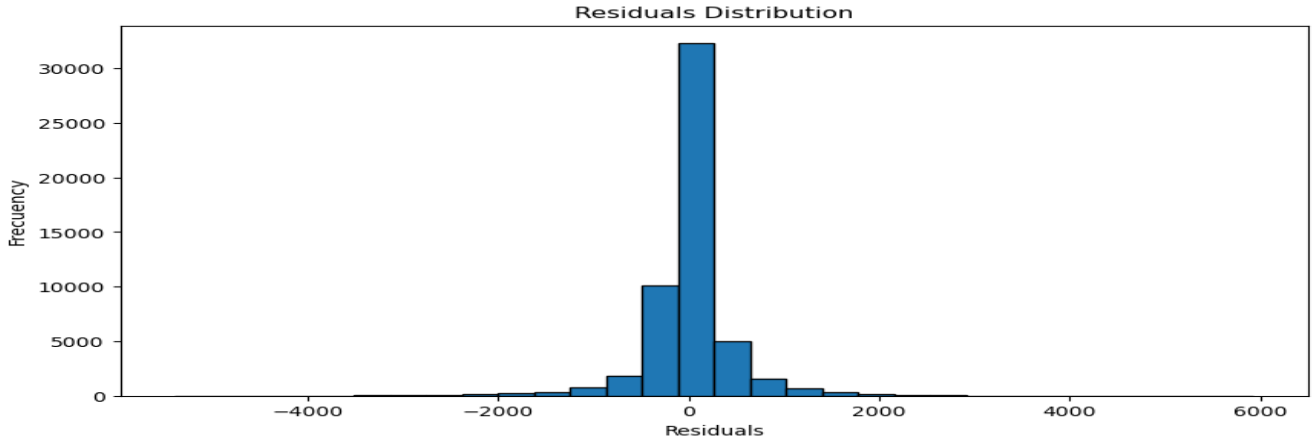
Donde las tres variables más relevantes para la predicción del precio de los diamantes fueron clarity, color y carat. Tal como muestra la siguiente gráfica

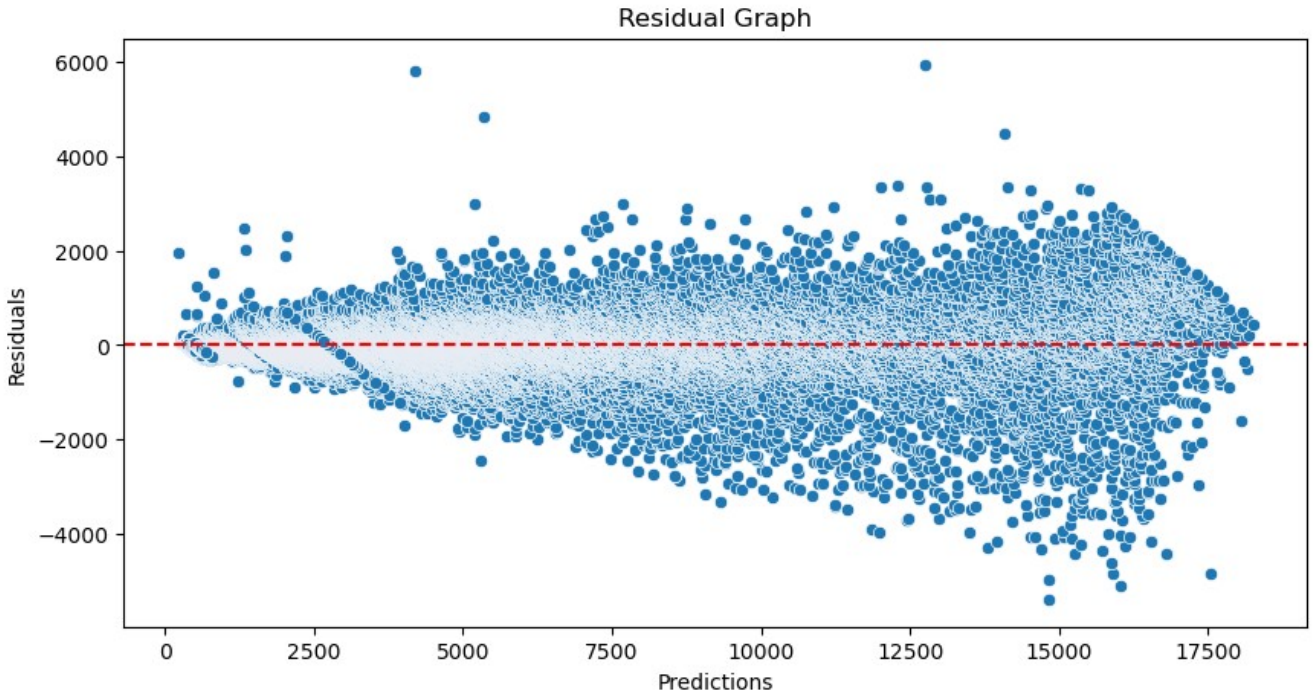


La siguiente gráfica muestra como varía la salida del modelo según las variables.



Distribución de Errores. Como se observa en las siguientes gráficas la distribución normal de los errores sugiere que el modelo está correctamente ajustado y que los residuos están centrados en torno a cero, indicando que no hay sesgo.





5. Conclusiones

- **Valor Estimado de los Diamantes Robados**: Tras el análisis y el entrenamiento del modelo predictivo, se puede estimar el valor total de los diamantes robados dependiendo de las características proporcionadas por Krenk, basándose en sta estimación se basa en los valores más representativos del dataset de referencia.
- **Limitaciones**: Aunque el modelo ha mostrado un buen rendimiento, existen algunas limitaciones que deben ser consideradas. Entre ellas se encuentran la presencia de outliers, que aunque no afectan significativamente el rendimiento del modelo XGBoost, podrían alterar los resultados en otros contextos, así mismo probablemente sea conveniente el uso de una regularización más estricta sobre todo en los casos donde se obtenga un error sospechosamente muy bajo lo que podría indicar sobreajuste. Además, el dataset utilizado, aunque extenso, podría no ser completamente representativo de todos los tipos de diamantes, lo cual impacta en la generalización del modelo.
- **Recomendaciones**: Para futuras mejoras, se recomienda ampliar el dataset con más muestras de diamantes que reflejen una mayor diversidad de características. Asimismo, sería útil optimizar aún más los hiperparámetros del modelo, quizás utilizando técnicas más avanzadas como Bayesian Optimization o bien usar modelos alternativos para comparar el rendimiento, tales como Random Forest o CatBoost, incluso con el uso de algoritmos genéticos para la optimización de hiperparámetros en caso de que el costo computacional no sea excesivo.