JANVIER 2024

Rapport du Projet : Modélisation Stochastique

Réaliser par :
NAQQAZ Wissal
BEL HADI NISRINE
BELHAID Nassrou-eddine

Encadrer par :
DR.EL MEHDI Rachida

REMERCIREMENT

Chère Professeure Elmehdi Rachida,
Au nom de Nassrou-eddine Belhaid, Nisrin Belhadi et Wissal Neqqaz, nous souhaitons exprimer notre profonde gratitude pour votre encadrement remarquable durant ce semestre consacré au module de Modélisation Stochastique dans la filière **Data Science et Cloud Computing** à l'**ENSA d'Oujda**.
Votre engagement exceptionnel, votre dévouement sans faille et votre expertise ont joué un rôle déterminant dans notre compréhension approfondie des concepts complexes abordés tout au long de ce cours.
Votre manière passionnée et didactique de transmettre le savoir a suscité en nous un intérêt continu pour ce domaine captivant.
Nous sommes sincèrement reconnaissants pour votre disponibilité, vos précieux conseils et votre soutien constant qui ont largement contribué à notre épanouissement académique.

SOMMAIRE

1-INTRODUCTION

1.1-Introduction générale :

La prédiction des prix **des diamants** est un domaine d'intérêt majeur dans le secteur de la joaillerie et de l'industrie des pierres précieuses. L'évaluation précise du prix des diamants repose sur un ensemble complexe de caractéristiques, allant du carat à la taille et aux proportions géométriques. La capacité à estimer avec précision ces valeurs est cruciale pour les acteurs du marché, des acheteurs et des vendeurs aux experts en gemmologie.

1.2-Contexte de l’étude :

En s'appuyant sur des techniques de **régression linéaire** et d'**analyse Stochastique** , cette étude vise à établir des modèles de prédiction robustes et précis. Nous cherchons à explorer comment ces caractéristiques interagissent pour influencer le prix des diamants, identifiant ainsi les facteurs prépondérants dans cette valorisation. L'objectif principal est de développer des modèles qui non seulement fournissent des prédictions précises, mais qui offrent également des informations sur l'importance relative de chaque caractéristique dans la détermination du prix des diamants. Cette analyse aidera à comprendre les tendances du marché et à formuler des recommandations pour les professionnels de l'industrie.

2-DESCRIPTION ET EXPLORATION DES DONNÉES

2.1-Informations sur les variables exogènes / endogènes

Variables Exogènes :

1. **Carat** : Poids du diamant en carats, représentant l'une des caractéristiques majeures influençant traditionnellement le prix.
2. **Depth** : Pourcentage de la profondeur totale par rapport au diamètre moyen.
3. **Table** : Pourcentage de la largeur de la table par rapport au diamètre moyen.
4. **x, y, z** : Dimensions du diamant dans trois directions.

Variable Endogène :

Prix : Le prix du diamant en dollars américains, notre variable cible à prédire.

2.2-Résumé des statistiques descriptives

```
[ ] data.shape
```

```
(53732, 7)
```

```
data.describe()
```

	carat	depth	table	price	x	y	z
count	53732.000000	53732.000000	53732.000000	53732.000000	53732.000000	53732.000000	53732.000000
mean	0.798313	61.747867	57.458916	3936.682759	5.732693	5.736128	3.539614
std	0.473398	1.430490	2.234360	3988.976716	1.120465	1.141012	0.704925
min	0.200000	43.000000	43.000000	326.000000	0.000000	0.000000	0.000000
25%	0.400000	61.000000	56.000000	954.000000	4.720000	4.720000	2.910000
50%	0.700000	61.800000	57.000000	2407.000000	5.700000	5.710000	3.530000
75%	1.040000	62.500000	59.000000	5331.000000	6.540000	6.540000	4.040000
max	5.010000	79.000000	95.000000	18823.000000	10.740000	58.900000	31.800000

2-DESCRIPTION ET EXPLORATION DES DONNÉES

Les caractéristiques des diamants ont été résumées pour faciliter leur compréhension et leur traitement :

- **Carat :** Moyenne de 0.798, avec une variation allant de 0.2 à 5.01.
- **Depth :** En moyenne à 61.75, avec une variation de 43 à 79.
- **Table** : Moyenne de 57.46, variant de 43 à 95.
- **Price :** Valeur moyenne de 3936.68, avec une étendue de 326 à 18823**.**
- **x, y, z :** Dimensions avec des moyennes respectives de 5.73, 5.74 et 3.54.

la présence de zéros dans les dimensions des diamants (x, y, z) semble problématique car elles ne devraient pas être nulles dans un contexte réel

2.3-Traitement des anomalies

```
[ ]  # Compter le nombre de lignes où x, y et z sont égaux à zéro
     zero_values = ((data['x'] == 0) & (data['y'] == 0) & (data['z'] == 0)).sum()

     print(f"Il y a {zero_values} lignes avec des valeurs égales à zéro pour x, y et z.")
```

```
Il y a 6 lignes avec des valeurs égales à zéro pour x, y et z.
```

```
[ ]  # Filtrer les lignes où au moins une des colonnes 'carat', 'depth', 'table', 'price', 'x', 'y' ou 'z' est égale à zéro
     data = data[~(data[['carat', 'depth', 'table', 'price', 'x', 'y', 'z']] == 0).any(axis=1)]

     # Vérifier la nouvelle taille du DataFrame après suppression des lignes
     print(f"Taille du DataFrame après suppression des lignes : {data.shape}")
```

```
Taille du DataFrame après suppression des lignes : (53713, 7)
```

```
[ ]  # Compter le nombre de lignes où x, y et z sont égaux à zéro
     zero_values = ((data['x'] == 0) & (data['y'] == 0) & (data['z'] == 0)).sum()

     print(f"Il y a {zero_values} lignes avec des valeurs égales à zéro pour x, y et z.")
```

```
Il y a 0 lignes avec des valeurs égales à zéro pour x, y et z.
```

2.3-Traitement des anomalies

```
[ ]  #valeurs manquantes
     data.isna().sum()
```

```
carat    0
depth    0
table    0
price    0
x        0
y        0
z        0
dtype: int64
```

```
[ ]  #Showing how many duplicated rows for the whole dataset
     data.duplicated().sum()
```

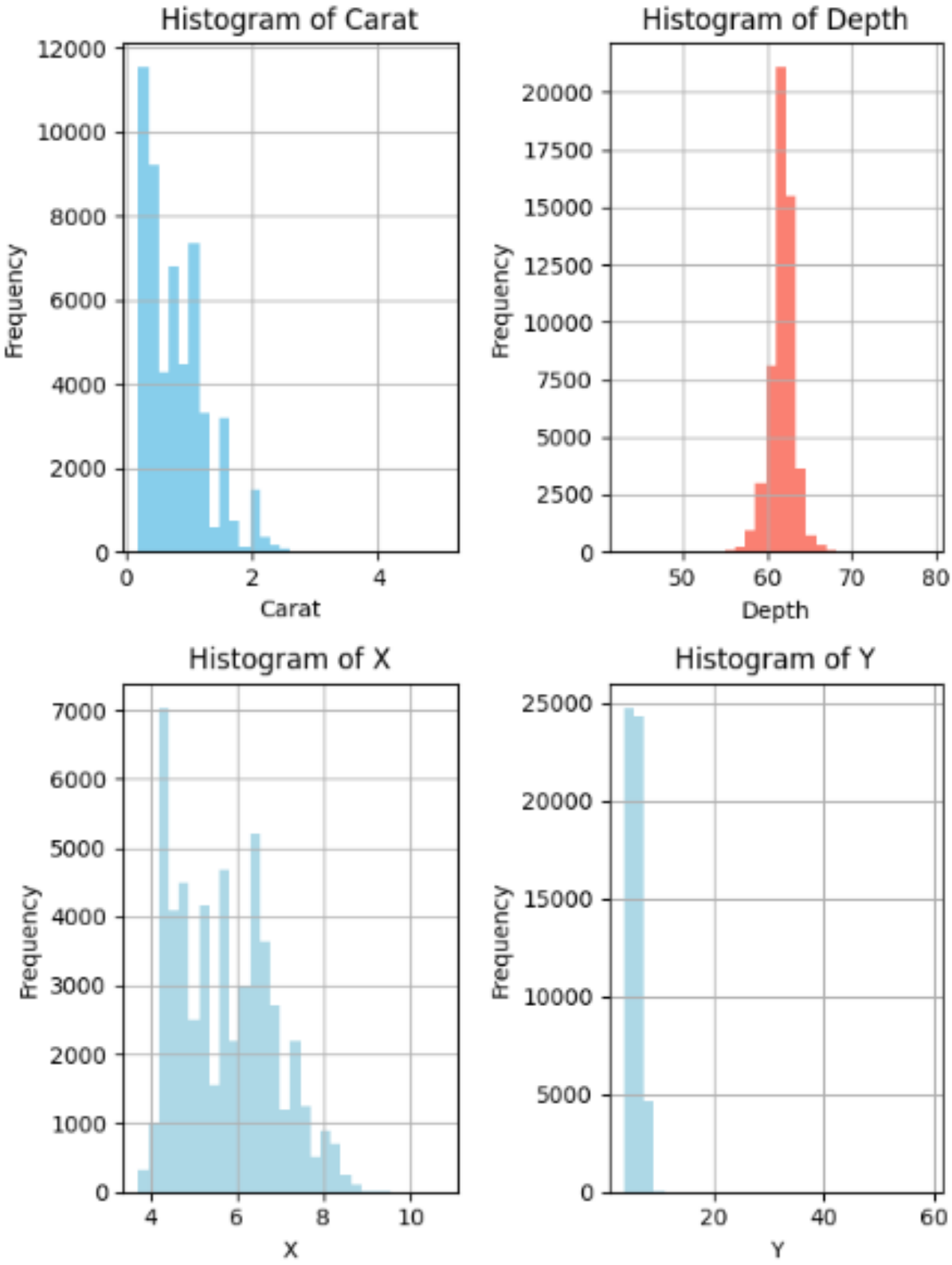
```
208
```

```
[ ]  data=data.drop_duplicates()
```

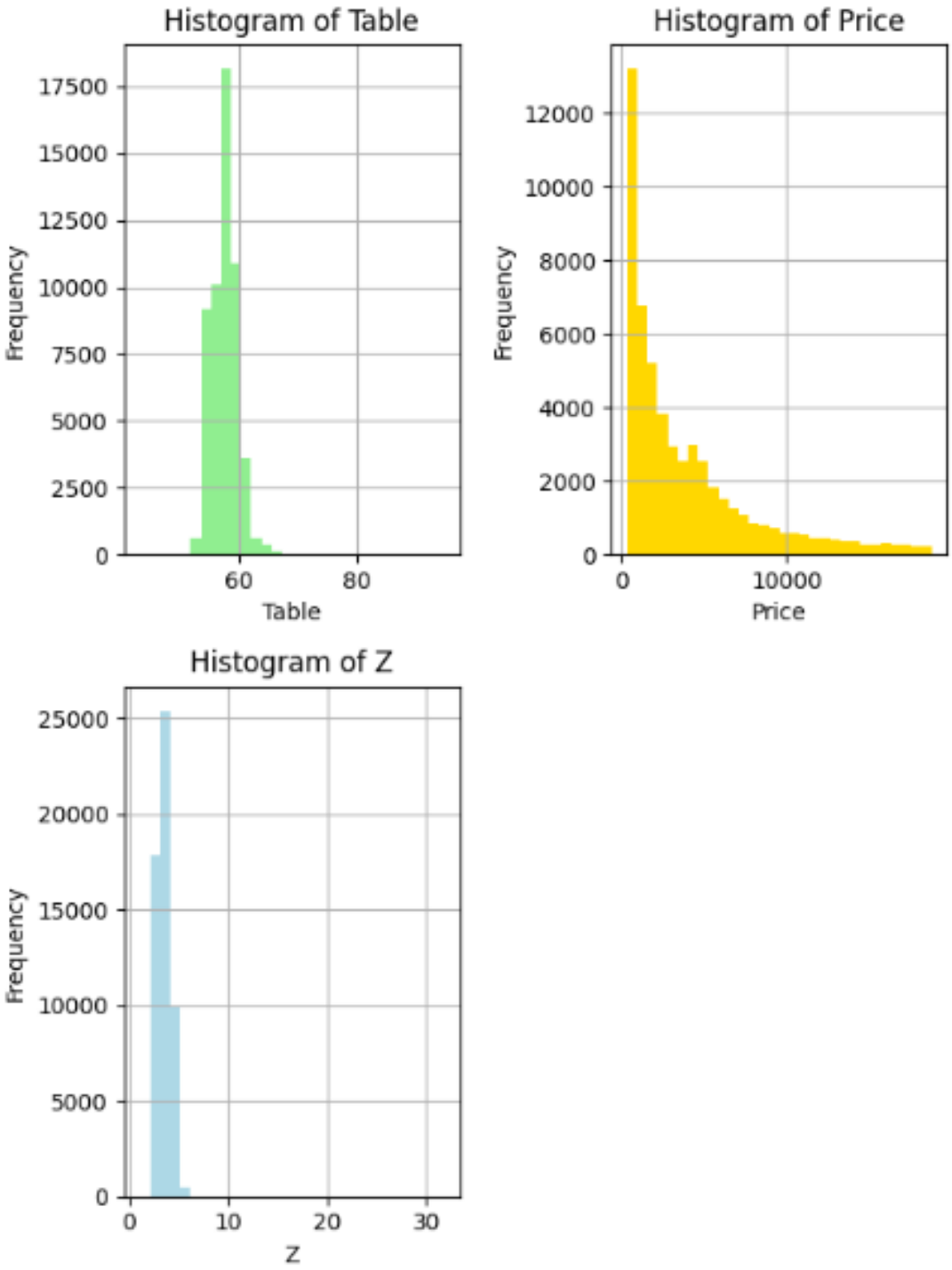
Après avoir effectué une description des données et éliminé les anomalies, l'étape suivante consiste à explorer visuellement **les relations** entre les variables. La visualisation et l'étude **des corrélations** entre les différentes caractéristiques permettront de comprendre les liens et **les dépendances** éventuelles entre elles

3-INTERPRÉTATION ET PRÉPARATION DES DONNÉES

3.1-Visualisations et interprétation



3.1-Visualisations et interprétation



L'absence de **centrage** des données à zéro peut altérer la représentation visuelle. Lorsque les données ne sont pas **centrées à zéro**, les histogrammes peuvent sembler décalés par rapport à l'axe central, ce qui peut fausser l'interprétation des tendances et des variations

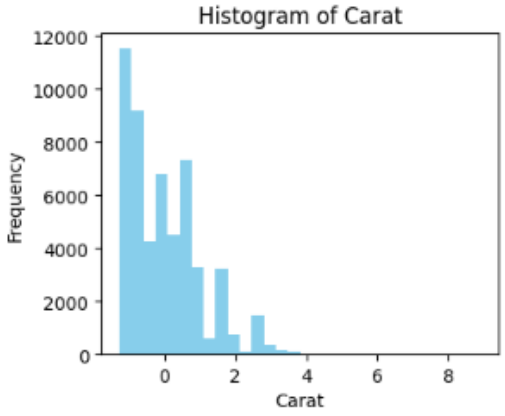
3.2-la normalisation des données

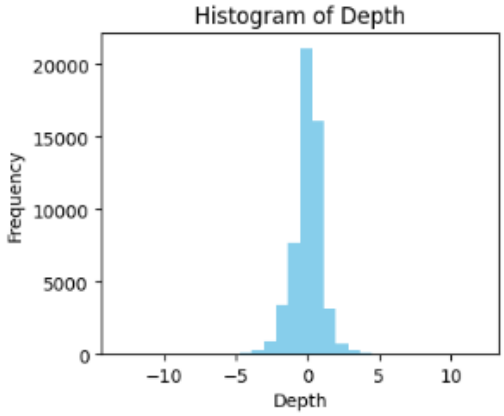
```
[ ]  # Sélectionner les colonnes à normaliser (toutes sauf 'price')
     columns_to_normalize = ['carat', 'depth', 'table', 'x', 'y', 'z']
     # Créer un objet StandardScaler
     scaler = StandardScaler()

     # Normaliser les données sauf la colonne 'price'
     data[columns_to_normalize] = scaler.fit_transform(data[columns_to_normalize])

     # Afficher les premières lignes du DataFrame pour vérification
     print(data.head())
```

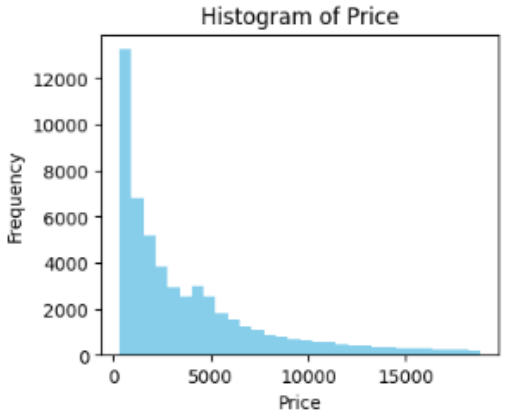
```
      carat     depth     table  price         x         y         z
0 -1.200554 -0.173415 -1.100584    326 -1.594341 -1.541716 -1.582673
1 -1.242822 -1.362054  1.585281    326 -1.647991 -1.664605 -1.753640
2 -1.200554 -3.389733  3.375857    327 -1.504925 -1.462716 -1.753640
3 -1.073750  0.455864  0.242348    334 -1.370801 -1.322272 -1.297729
4 -1.031482  1.085143  0.242348    335 -1.245619 -1.216938 -1.126763
```

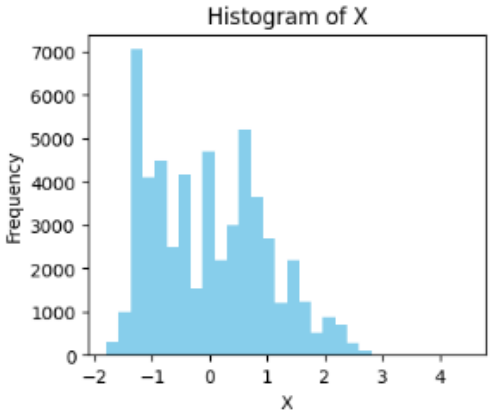
La colonne "**carat**" semble être réduite avec des valeurs négatives, indiquant peut-être des diamants plus petits que la moyenne. La colonne "**depth**" montre des valeurs majoritairement négatives, suggérant des diamants avec des profondeurs moins importantes par rapport à la moyenne. "**Table**" présente également des valeurs majoritairement négatives, ce qui pourrait signifier des diamants avec des tables plus petites que la moyenne. Quant aux colonnes "**x**", "**y**" et "**z**", elles semblent toutes être réduites avec des valeurs négatives, indiquant des dimensions plus petites pour ces diamants spécifiques. La colonne "**price**" semble être la valeur brute du prix, sans normalisation, montrant une fourchette de prix variée pour ces diamants.

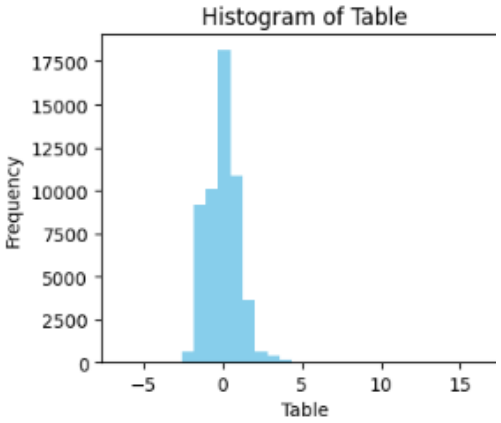
3.2-la normalisation des données

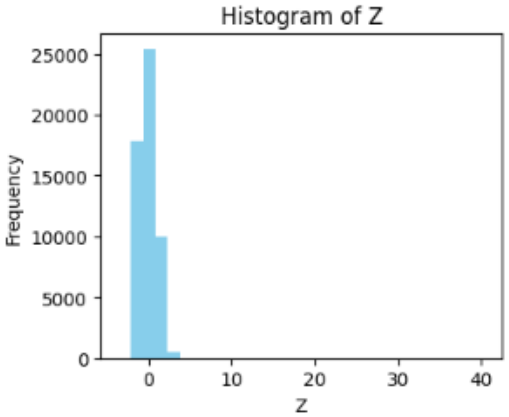


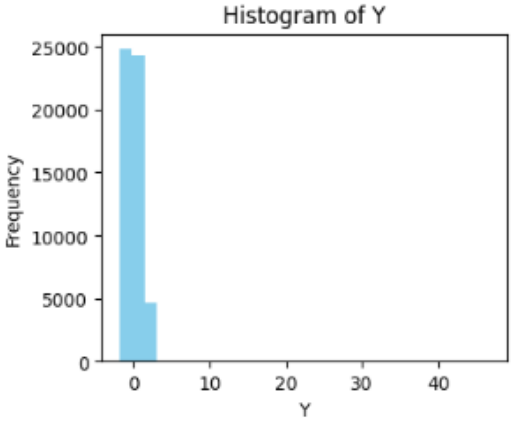








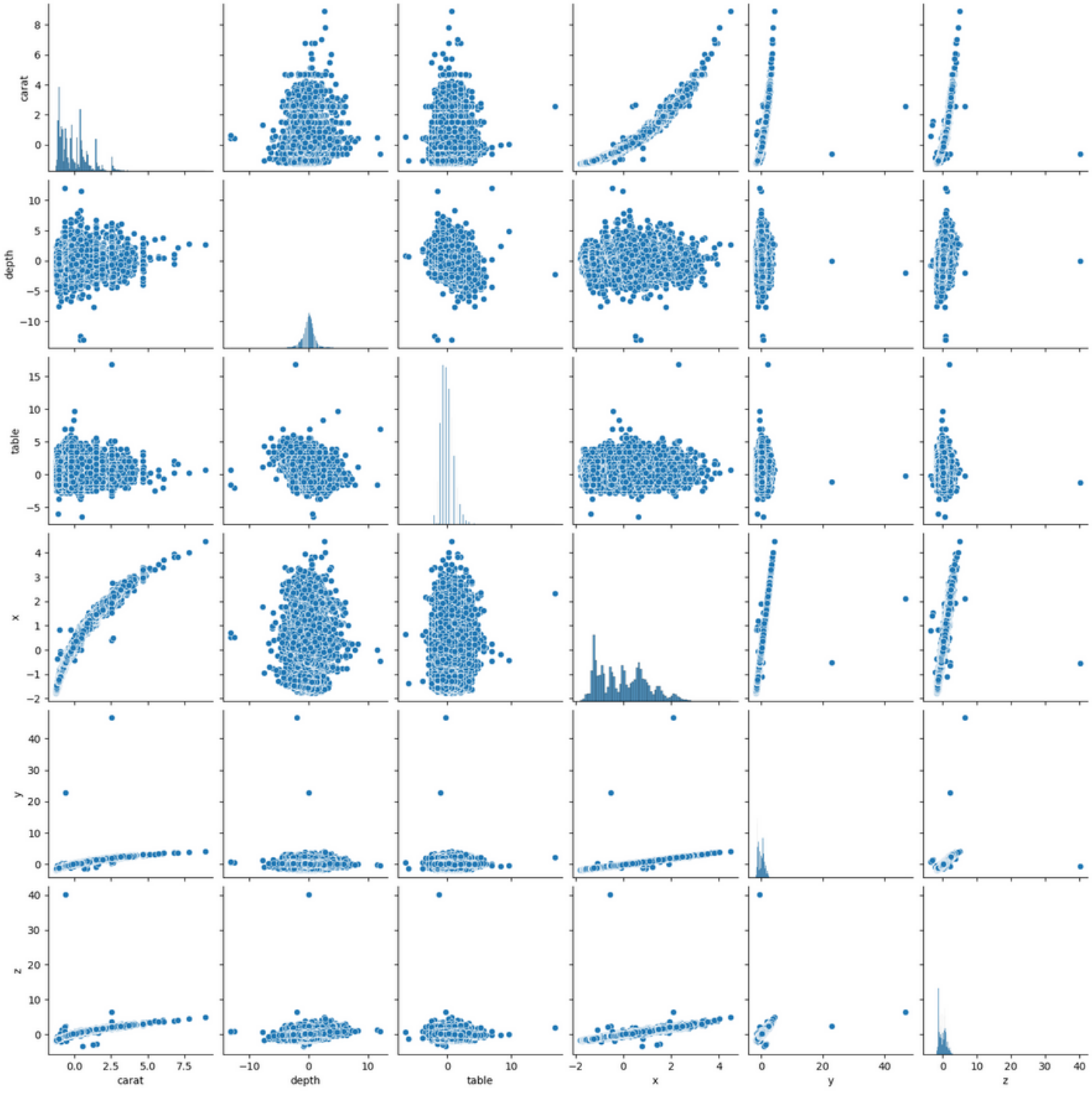




3.3-Etude de la corrélation

Ces graphiques montrent la dispersion des données et suggèrent des corrélations ou des tendances entre les variables

MULITI-PLOT



Dans le multi-plot, une tendance claire est observée entre le poids en carats des diamants et leurs dimensions physiques (x, y, z), suggérant une relation directe entre ces variables.

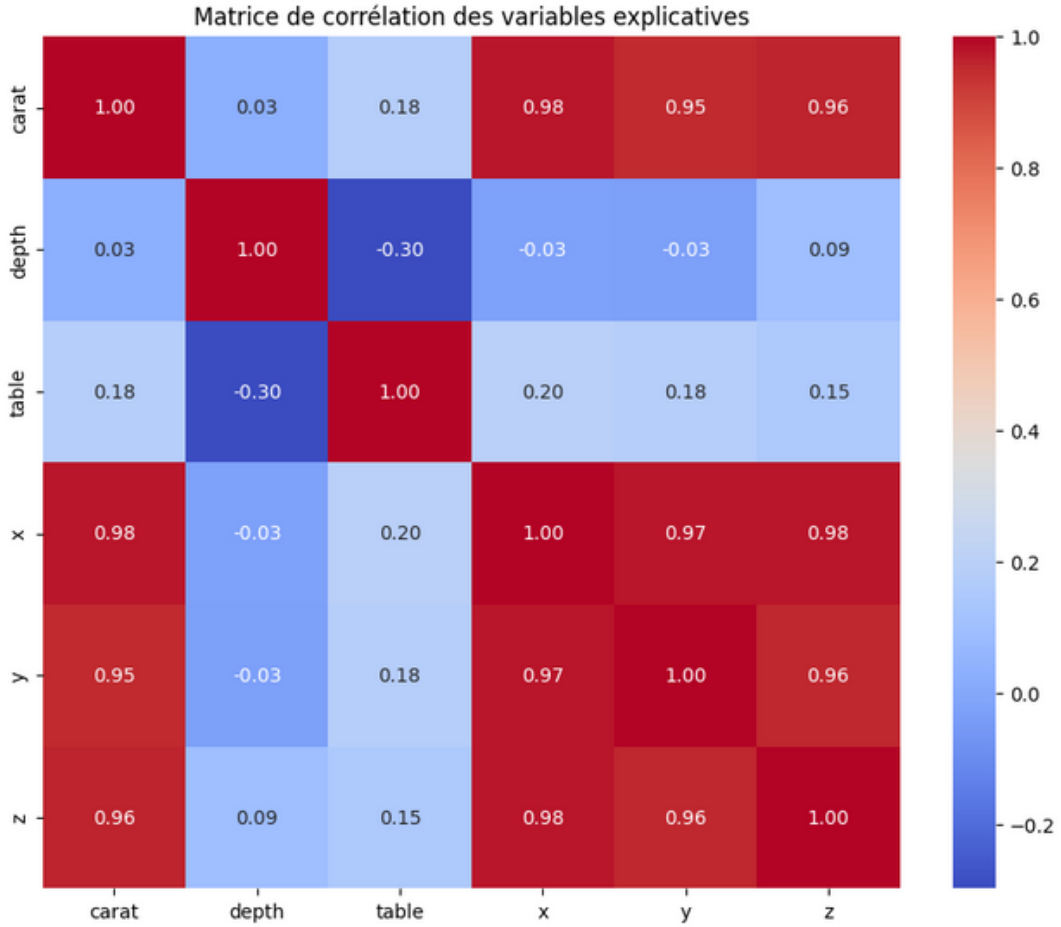
3.3-Etude de la corrélation

MATRICE DE CORRÉLATION

Matrice de corrélation des variables explicatives :

	carat	depth	table	x	y	z
carat	1.000000	0.028039	0.180927	0.977848	0.953937	0.960996
depth	0.028039	1.000000	-0.297512	-0.025095	-0.029137	0.094885
table	0.180927	-0.297512	1.000000	0.195268	0.183622	0.151473
x	0.977848	-0.025095	0.195268	1.000000	0.974784	0.975346
y	0.953937	-0.029137	0.183622	0.974784	1.000000	0.956559
z	0.960996	0.094885	0.151473	0.975346	0.956559	1.000000

- La matrice de corrélation souligne des associations clés : le poids en carats des diamants présente une corrélation forte et positive avec leurs dimensions physiques (x, y, z), approchant souvent la valeur maximale de 1.
- Les dimensions x et y, ainsi que x et z, démontrent également une corrélation positive élevée.
- En revanche, la profondeur (depth) semble avoir une relation légèrement négative avec la table, suggérant une tendance où une profondeur accrue est liée à une table plus petite.



3.3-Etude de la corrélation

CORRÉLATION DES VARIABLES EXPLICATIVES AVEC LA VARIABLE CIBLE

```
[ ]  # Liste des noms des variables explicatives (excluant la variable cible 'price')
     variables_explicatives = ['carat', 'depth', 'table', 'x', 'y', 'z']

     # Calcul de la corrélation entre chaque variable explicative et la variable cible 'price'
     correlations_with_price = data[variables_explicatives].corrwith(data['price'])

     # Affichage des corrélations avec la variable cible 'price'
     print("Corrélations avec la variable 'price':")
     print(correlations_with_price)
```

```
Corrélations avec la variable 'price':
carat    0.921500
depth   -0.011036
table    0.126472
x        0.887096
y        0.867615
z        0.867961
dtype: float64
```

Les carats ainsi que les dimensions physiques (x, y, z) présentent une forte corrélation avec le prix des diamants, sous-entendant une liaison directe entre ces caractéristiques et la valeur commerciale des diamants. Cependant, la profondeur (depth) semble ne pas avoir de corrélation marquée avec le prix.

4-MODELISATION STOCHASTIQUE

La **modélisation statistique** est une manière simplifiée et formalisée mathématiquement de s'approcher de la réalité et, en d'autres termes, de décrire les processus qui génèrent vos données. De façon optionnelle, elle permet de faire des prédictions à partir de cette approximation. Le modèle statistique est l'équation mathématique utilisée.

La **modélisation stochastique** est un terme générique dont le but principal est de représenter un comportement par des modèles probabilistes. La plupart du temps ce que nous cherchons à modéliser présente un aléas (telle que la précipitation par exemple) mais ce n'est pas toujours le cas.

Modèles Stochastiques et Déterministes:

Les modèles sont généralement soir stochastiques, soit déterministes, bien que des modèles composites existent. Les modèles déterministes sont basés sur une loi connue ou hypothétique de la physique, des mathématiques ou d'une quelconque autre discipline, de sorte que des valeurs d'input données produisent toujours le même résultat. Par contre, le modèle stochastique accepte une certaine distribution de probabilité associée à des inputs donnés, dans les processus au sein du modèle et donc dans l'output, de sorte que le même input peut amener à différentes valeur d'output.

4.1-Entraînement de la régression multiple en utilisant différentes combinaisons

1er régression

```
[ ] #split data into feature and target
    X = data[['carat', 'depth','table','x','y','z']]
    y = data['price']
```

```
[ ] #split data into test and train
    X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
```

```
[ ] #model training
    model = LinearRegression()
    model.fit(X_train, y_train)
```

▾ LinearRegression
LinearRegression()

1. Choisir 'carat', 'depth','table','x','y','z' des variables explicatives de la variable cible
2. Division du jeu de données
3. Entrainement du modèle et afficher les résultats

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                  price   R-squared:                       0.860
Model:                            OLS   Adj. R-squared:                  0.860
Method:                 Least Squares   F-statistic:                 5.482e+04
Date:                Mon, 08 Jan 2024   Prob (F-statistic):               0.00
Time:                        23:31:43   Log-Likelihood:                -23481.
No. Observations:               53713   AIC:                         4.698e+04
Df Residuals:                   53706   BIC:                         4.704e+04
Df Model:                           6
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
Intercept   8.049e-16      0.002   4.98e-13      1.000      -0.003       0.003
carat          1.3096      0.008    163.975      0.000       1.294       1.325
depth         -0.0734      0.002    -36.072      0.000      -0.077      -0.069
table         -0.0575      0.002    -33.222      0.000      -0.061      -0.054
x             -0.4005      0.013    -31.133      0.000      -0.426      -0.375
y              0.0247      0.007      3.360      0.001       0.010       0.039
z             -0.0079      0.009     -0.895      0.371      -0.025       0.009
==============================================================================
Omnibus:                    14043.727   Durbin-Watson:                   1.281
Prob(Omnibus):                  0.000   Jarque-Bera (JB):           415872.514
Skew:                           0.639   Prob(JB):                         0.00
Kurtosis:                      16.572   Cond. No.                         18.4
==============================================================================
```

4.1-Entraînement de la régression multiple en utilisant différentes combinaisons

Interpretation de la 1er régression

- Le coefficient de détermination (R^2) ajusté est de 0,860, indiquant que 86 % de la variance dans le prix des diamants est expliquée par les caractéristiques incluses dans le modèle
- Les variables 'carat', 'depth' et 'table' présentent des coefficients significativement différents de zéro (p-value < 0.05), suggérant une influence statistiquement significative sur le prix des diamants.
- les coefficients associés aux dimensions 'x', 'y' et 'z' semblent également significatifs, bien que leur impact sur le prix puisse être plus modéré, étant donné que les p-values pour 'y' et 'z' sont légèrement supérieures à 0.05.

4.2-Evaluation du modele

Interpretation de la 2 eme régression

- Étant donné que la probabilité de l'interception est élevée et celle de z est élevée également, nous allons réajuster le modèle en l'absence de l'interceptt on premiere lieu

2eme regression sans l'intercept

```
#entrainement de model
model = smf.ols("price ~ carat + depth + table + x + y + z-1", data=data1)
results = model.fit()
print(results.summary())
```

```
                                 OLS Regression Results
================================================================================================
Dep. Variable:                   price   R-squared (uncentered):                   0.860
Model:                             OLS   Adj. R-squared (uncentered):              0.860
Method:                  Least Squares   F-statistic:                          4.408e+04
Date:                 Tue, 09 Jan 2024   Prob (F-statistic):                        0.00
Time:                         08:55:46   Log-Likelihood:                         -18865.
No. Observations:                42970   AIC:                                    3.774e+04
Df Residuals:                    42964   BIC:                                    3.779e+04
Df Model:                            6
Covariance Type:             nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
carat          1.3203      0.009    147.377      0.000       1.303       1.338
depth         -0.0748      0.002    -33.692      0.000      -0.079      -0.070
table         -0.0575      0.002    -29.674      0.000      -0.061      -0.054
x             -0.4094      0.014    -30.210      0.000      -0.436      -0.383
y              0.0193      0.007      2.615      0.009       0.005       0.034
z             -0.0040      0.009     -0.453      0.651      -0.021       0.013
==============================================================================
Omnibus:                    11139.976   Durbin-Watson:                   1.980
Prob(Omnibus):                  0.000   Jarque-Bera (JB):           285990.664
Skew:                           0.674   Prob(JB):                         0.00
Kurtosis:                      15.567   Cond. No.                         17.4
==============================================================================

Notes:
[1] R² is computed without centering (uncentered) since the model does not contain a constant.
[2] Standard Errors assume that the covariance matrix of the errors is correctly specified.
```

4.3-Sélection des caractéristiques pertinentes et amélioration du modèle

Interpretation de la 3 eme régression

- ous allons réajuster le modèle en l'absence de l'intercept et z

3 eme regression sans intercept et z

```
#entrainement de model
model = smf.ols("price ~ carat + depth + table + x + y -1", data=data1)
results = model.fit()
print(results.summary())
```

```
                                 OLS Regression Results
================================================================================
Dep. Variable:                  price   R-squared (uncentered):                   0.860
Model:                            OLS   Adj. R-squared (uncentered):              0.860
Method:                 Least Squares   F-statistic:                          5.290e+04
Date:                Tue, 09 Jan 2024   Prob (F-statistic):                        0.00
Time:                        08:55:51   Log-Likelihood:                         -18865.
No. Observations:               42970   AIC:                                  3.774e+04
Df Residuals:                   42965   BIC:                                  3.778e+04
Df Model:                           5
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
carat          1.3202      0.009    147.401      0.000       1.303       1.338
depth         -0.0753      0.002    -38.459      0.000      -0.079      -0.071
table         -0.0575      0.002    -29.671      0.000      -0.061      -0.054
x             -0.4128      0.011    -36.453      0.000      -0.435      -0.391
y              0.0188      0.007      2.577      0.010       0.005       0.033
==============================================================================
Omnibus:                    11139.707   Durbin-Watson:                   1.980
Prob(Omnibus):                  0.000   Jarque-Bera (JB):           286008.102
Skew:                           0.674   Prob(JB):                         0.00
Kurtosis:                      15.567   Cond. No.                         13.4
==============================================================================

Notes:
[1] R² is computed without centering (uncentered) since the model does not contain a constant.
[2] Standard Errors assume that the covariance matrix of the errors is correctly specified.
```

Le coefficient de détermination ajusté (R ajusté) est resté inchangé, et seuls les paramètres présentant une probabilité presque nulle d'être égaux à zéro ont été conservés, selon le test de Student sur les paramètres. Ainsi, les paramètres significatifs sont désormais : carat, table, depth, x, y.

4.4-Identifier les points leviers (distance de cook)

- Le levier, en statistiques, fait référence à la mesure de l'influence d'une observation individuelle sur la forme d'une courbe de régression. En d'autres termes, il évalue la capacité d'un point de données à influencer la position ou l'inclinaison de la ligne de régression. Un point de levier élevé indique une forte influence sur le modèle statistique, tandis qu'un point de levier faible a une influence moindre. Le levier est souvent utilisé pour identifier les observations qui peuvent avoir un impact disproportionné sur les résultats d'une analyse de régression

  - Nous avons identifié la présence de 42 970 points de levier.

```
print(leviers)
```

```
[6.23713881e-05 3.77938240e-05 8.29478127e-05 ... 6.51590730e-05
 5.40184819e-05 5.87110777e-05]
```

```
print(len(leviers))
```

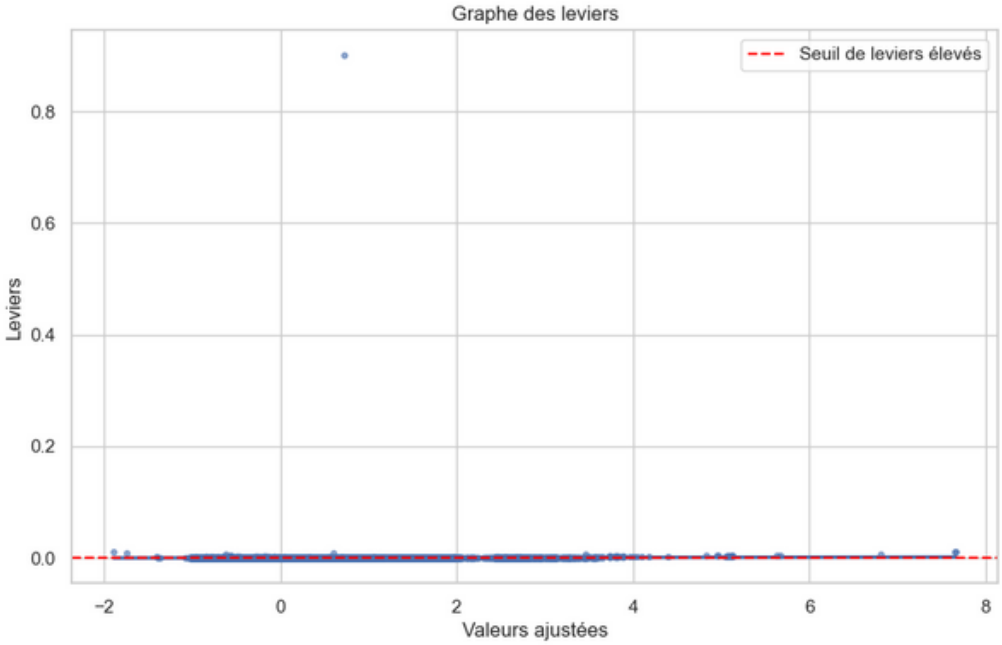
```
42970
```

4.4-Identifier les points leviers (distance de cook)

- Après avoir calculé le seuil de levier, il devient évident qu'il est nécessaire d'éliminer les points dont le levier est supérieur à ce seuil.

```
]: print(seuil_leviers )
```

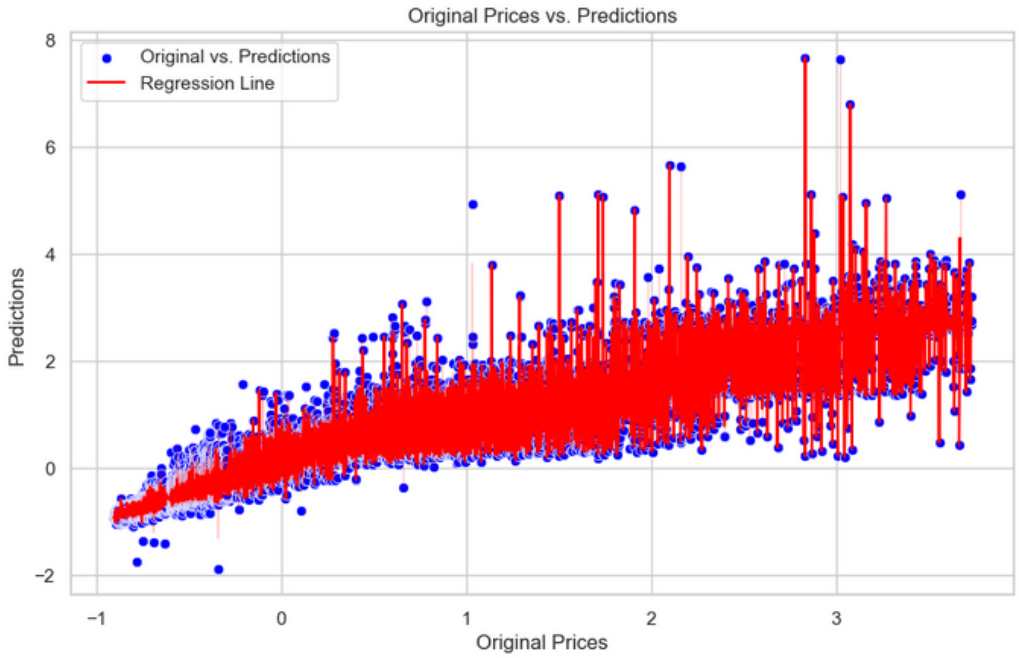
```
4.6544100535257155e-05
```



- En se fondant sur le graphique et la limite définie, nous avons éliminé quelques observations de notre ensemble de données. Ensuite, en effectuant une autre régression uniquement avec les variables pertinentes, et en excluant ces points, nous avons généré un diagramme de dispersion illustrant la différence entre les prédictions et les valeurs réelles.

4.4-Identifier les points leviers

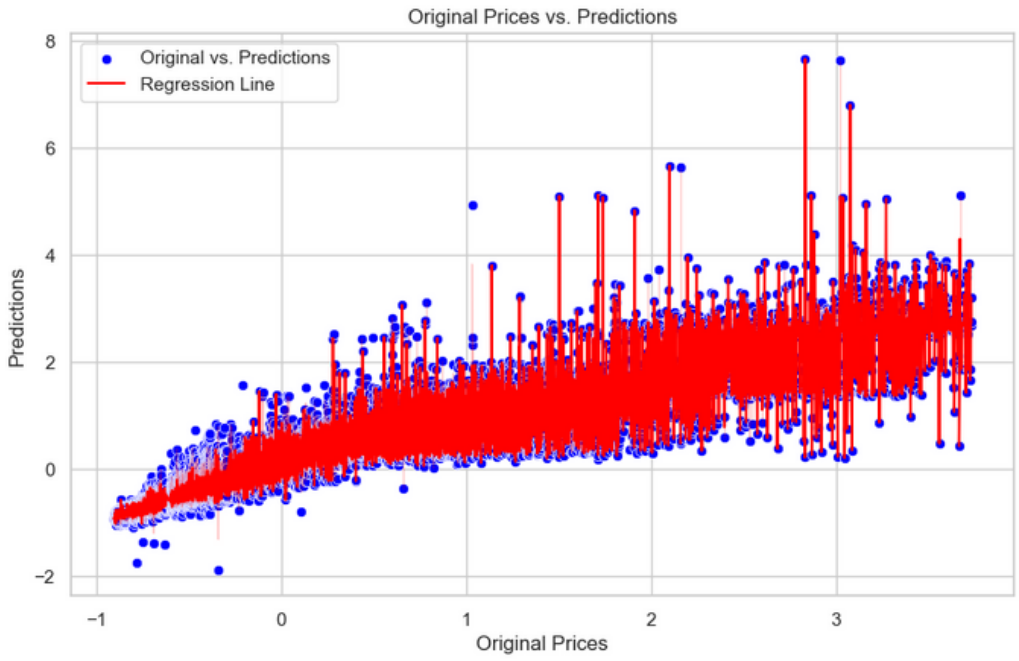
- L'erreur entre la réalité et la prédiction était minime, soit 0.18813736432144557.



5-MISE EN OEUVRE DU MODELE

5.1-Prediction

Nous avons déjà partitionné notre ensemble de données en ensembles d'entraînement et de test. À présent, nous allons utiliser la partie du jeu de données réservée au test pour entraîner notre modèle en excluant l'intercept et le z. Ensuite, nous comparerons les valeurs prédites par le modèle ajusté avec les valeurs réelles.

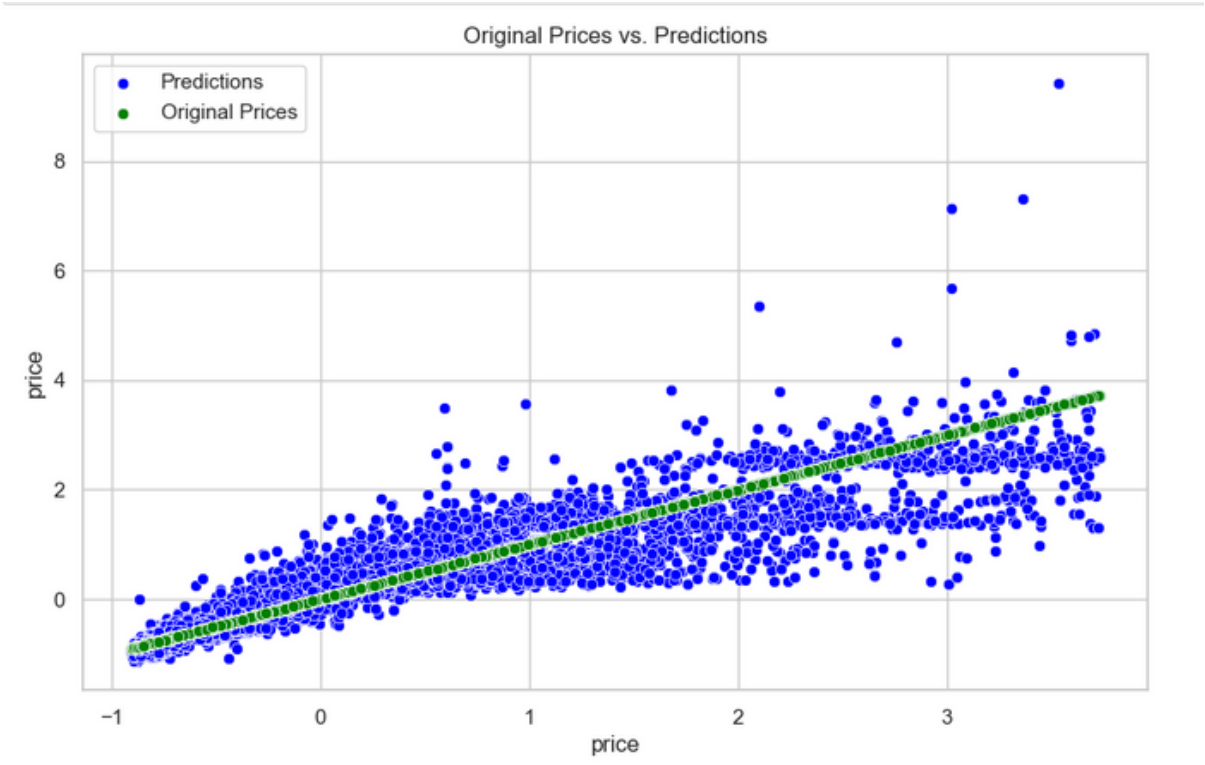



```
47595     -0.516170        -0.551245

[10743 rows x 2 columns]
```

L'erreur s'est élevée à seulement 0.2221511805574603.

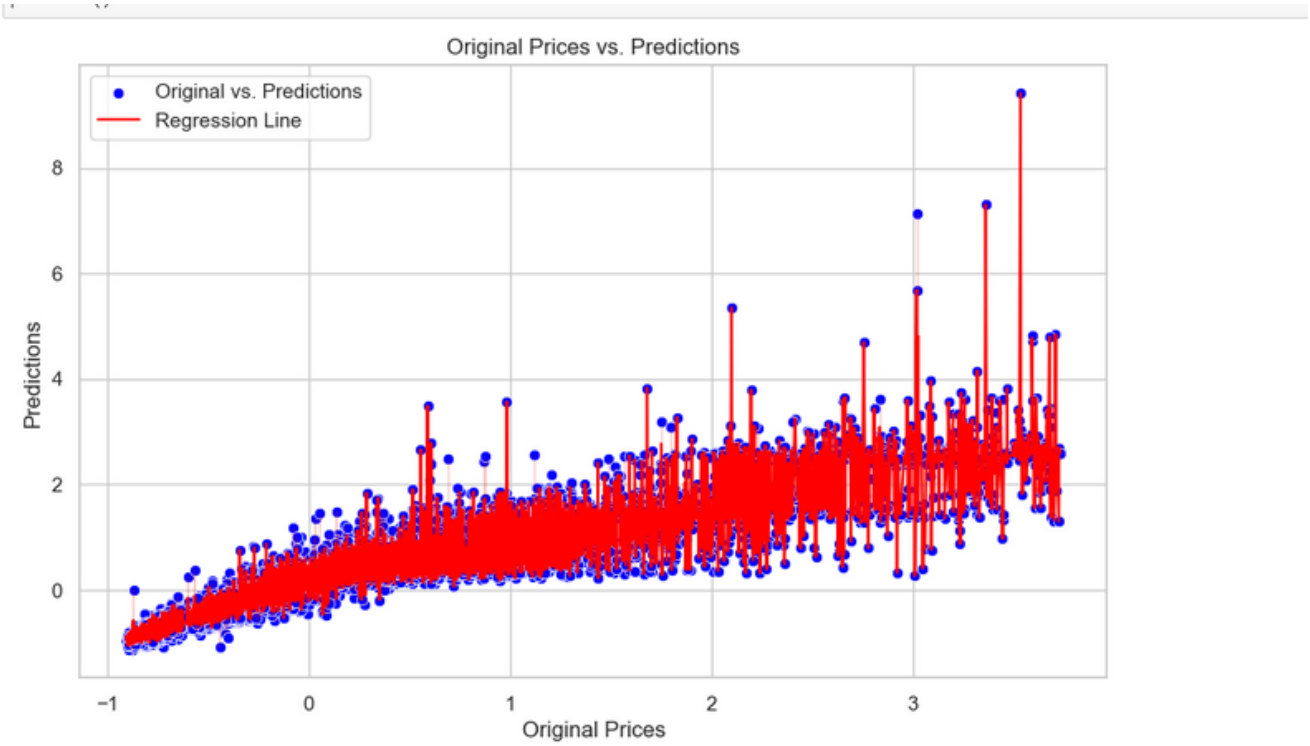
5.2-Graphe et analyse

Le graphique comparant les valeurs prédites aux valeurs réelles montre que la droite verte de régression passe par le centre du nuage de points. Cela indique que le modèle est bien ajusté, car la ligne de régression capture efficacement la tendance générale des données, minimisant ainsi l'écart entre les valeurs prédites et les valeurs réelles. En d'autres termes, le modèle semble bien représenter la relation entre les variables, ce qui renforce la confiance dans sa capacité à faire des prédictions précises.



5.2-Graphe et analyse

La bande rouge qui s'étend de la droite de régression vers les valeurs réelles représente les résidus ou les erreurs du modèle. Les résidus sont les écarts entre les valeurs prédites par le modèle et les valeurs réelles dans le jeu de données. Dans ce contexte, nous avons calculé une erreur moyenne de 0.2221511805574603, ce qui indique la magnitude moyenne des résidus.



6-CONCLUSION

6.1-Résumé

Le modèle de régression multiple suggère que le poids du diamant en carats, le pourcentage de la profondeur totale, le pourcentage de la largeur de la table, et les dimensions du diamant dans les directions x et y sont des facteurs significatifs pour expliquer la variation des prix des diamants. Cela signifie que ces caractéristiques spécifiques du diamant ont un impact statistiquement significatif sur son prix.
Cependant, la variable z, bien que incluse dans le modèle, ne semble pas contribuer de manière significative à l'explication de cette variation des prix.

6.2-Bibliographie

- https://seos-project.eu/modelling/modelling-c07-p01.fr.html
- https://help.xlstat.com/fr/6724-what-statistical-modeling
- Licence 3 MIASHS - Université de Bordeaux:chapitreII Régression linéaire multiple
- WikiStat:Introduction à la Régression linéaire multiple