

Détection de faux-billets

Organisation Nationale de Lutte contre le Faux-Monnayage

LE Ngoc

ONCFM

Organisation Nationale de Lutte contre le Faux-Monnayage

Organisation publique ayant pour objectif de mettre en place des méthodes d'identification des contrefaçons des billets en euros.
Dans le cadre de cette lutte, nous souhaitons mettre en place un algorithme qui soit capable de différencier automatiquement les vrais des faux billets.

L'objectif est de construire un algorithme qui, à partir des caractéristiques géométriques d'un billet, serait capable de définir si ce dernier est un vrai ou un faux billet.

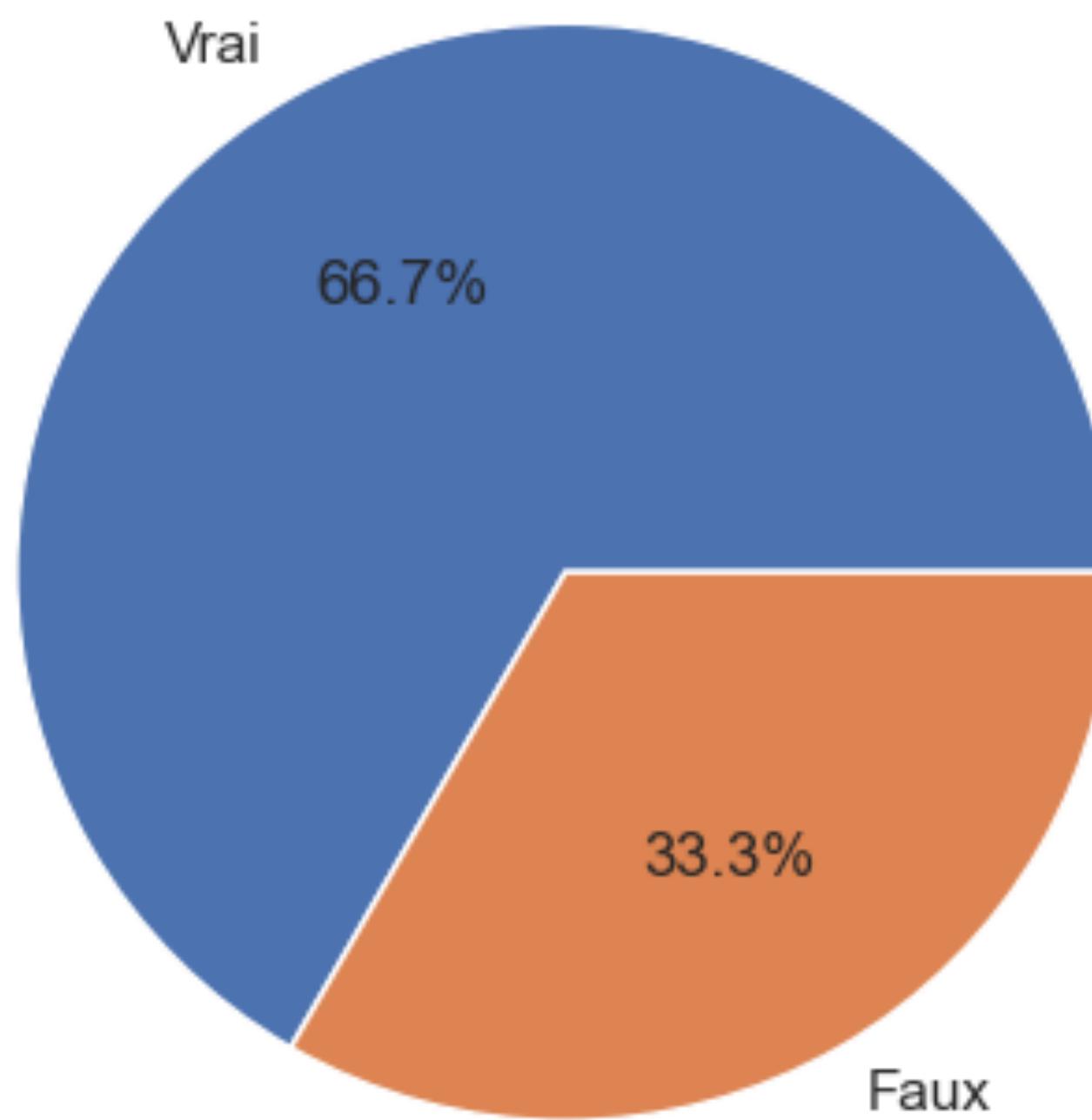
Analyse descriptive

Le jeu de données est composé de **1500 billets**, répartis de la sorte :

- 1000 vrais billets
- 500 faux billets

La variable « ***is_genuine*** » détermine si le billet est vrai ou faux.

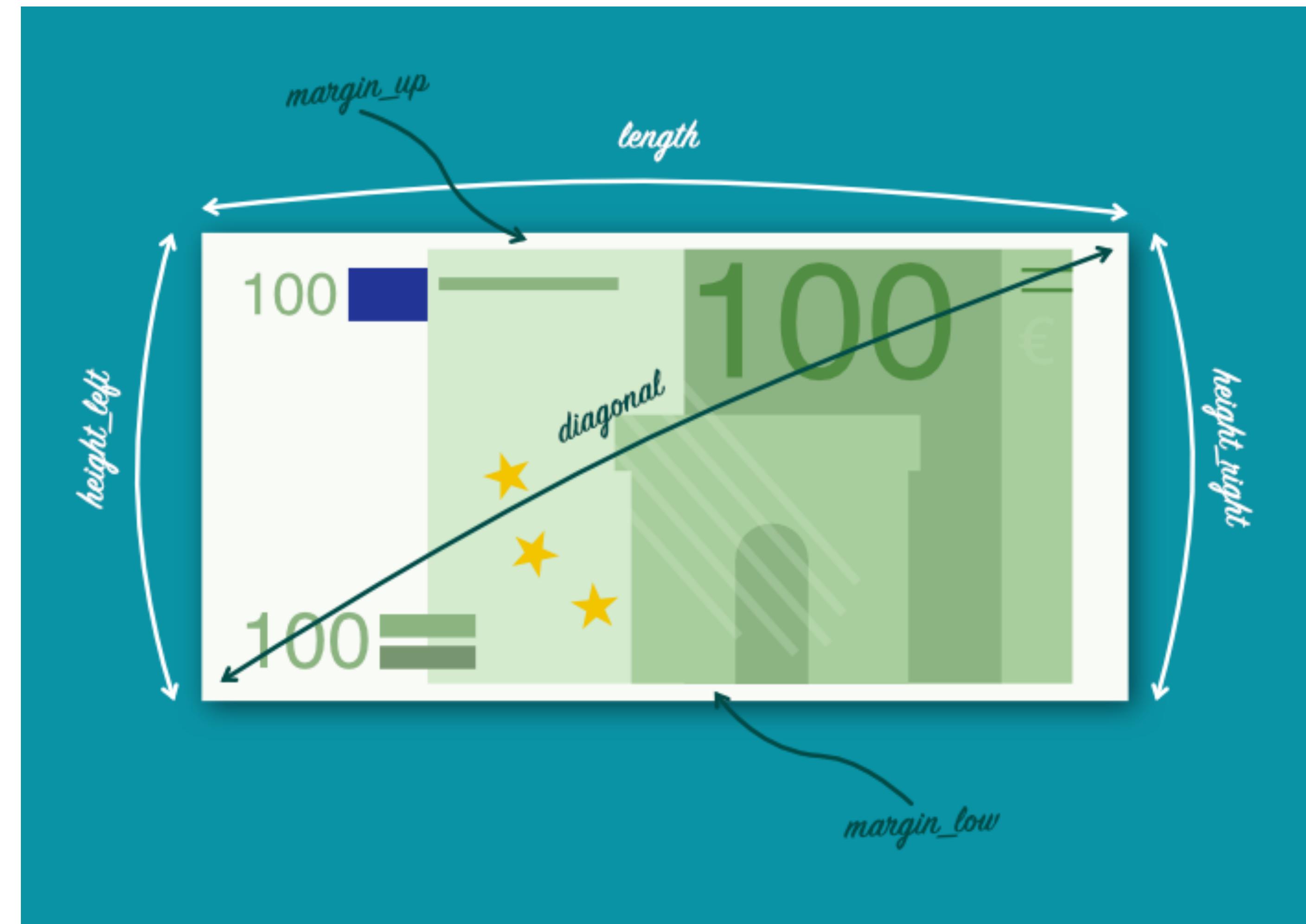
Répartition des billets



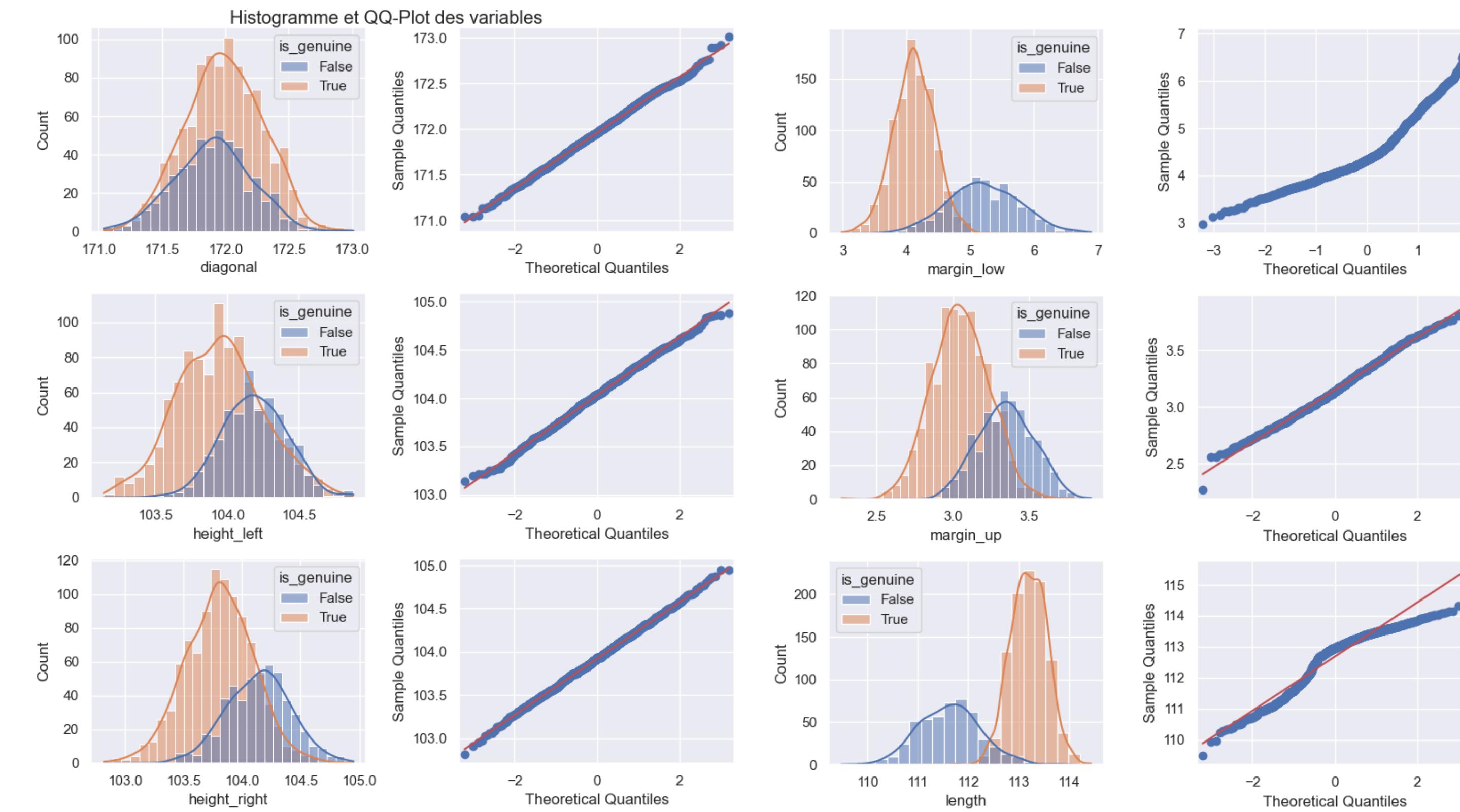
Dimensions des billets

6 variables prédictives :

- ***diagonal***
- ***height_left***
- ***height_right***
- ***margin_low***
- ***margin_up***
- ***length***



Analyse univariée

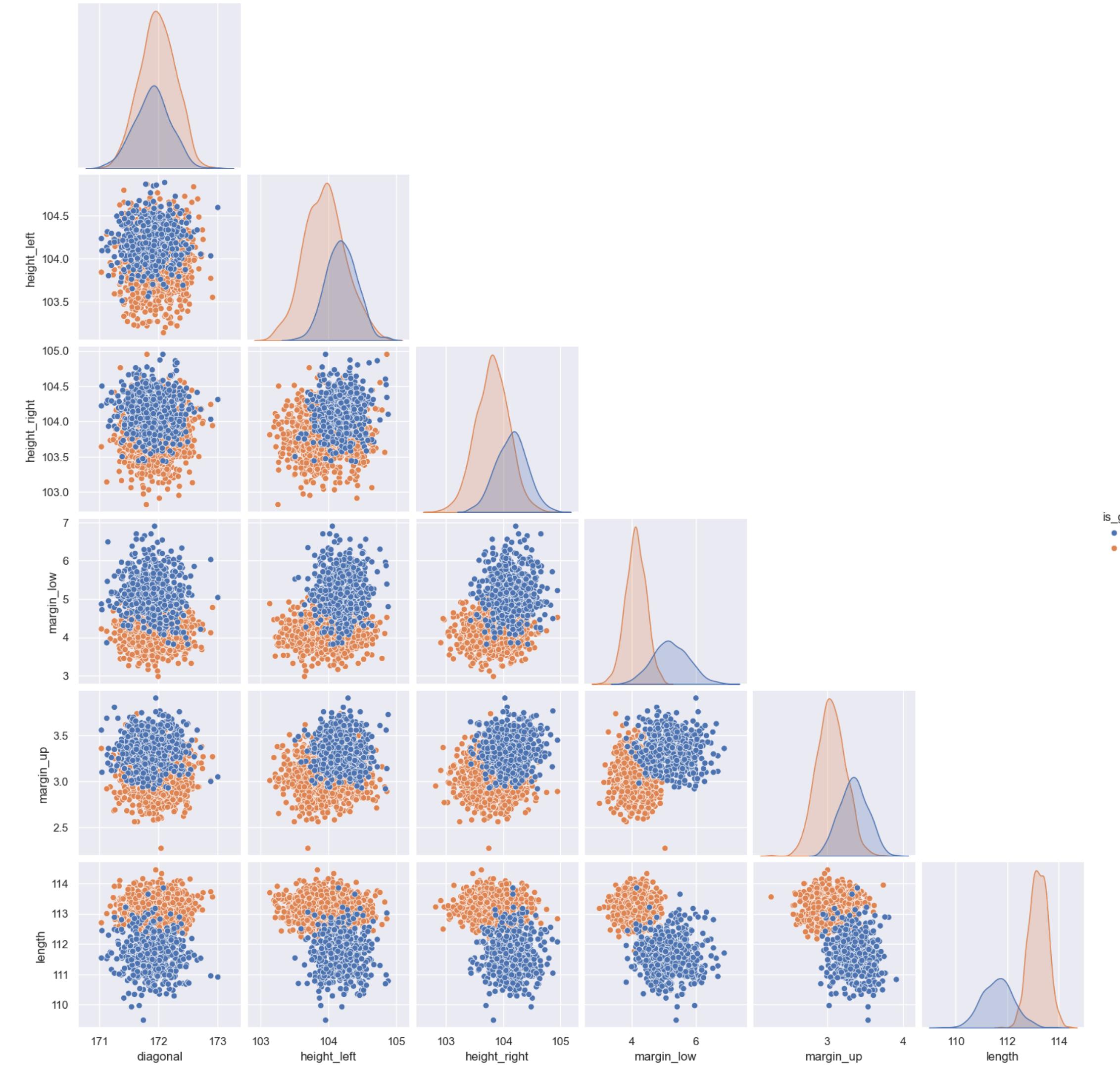


Histogramme et QQ plot pour analyser la distribution des variables.
Les variables suivent une loi normale si on considère les différences entre les vrais et les faux billets.

Analyse multi-variée

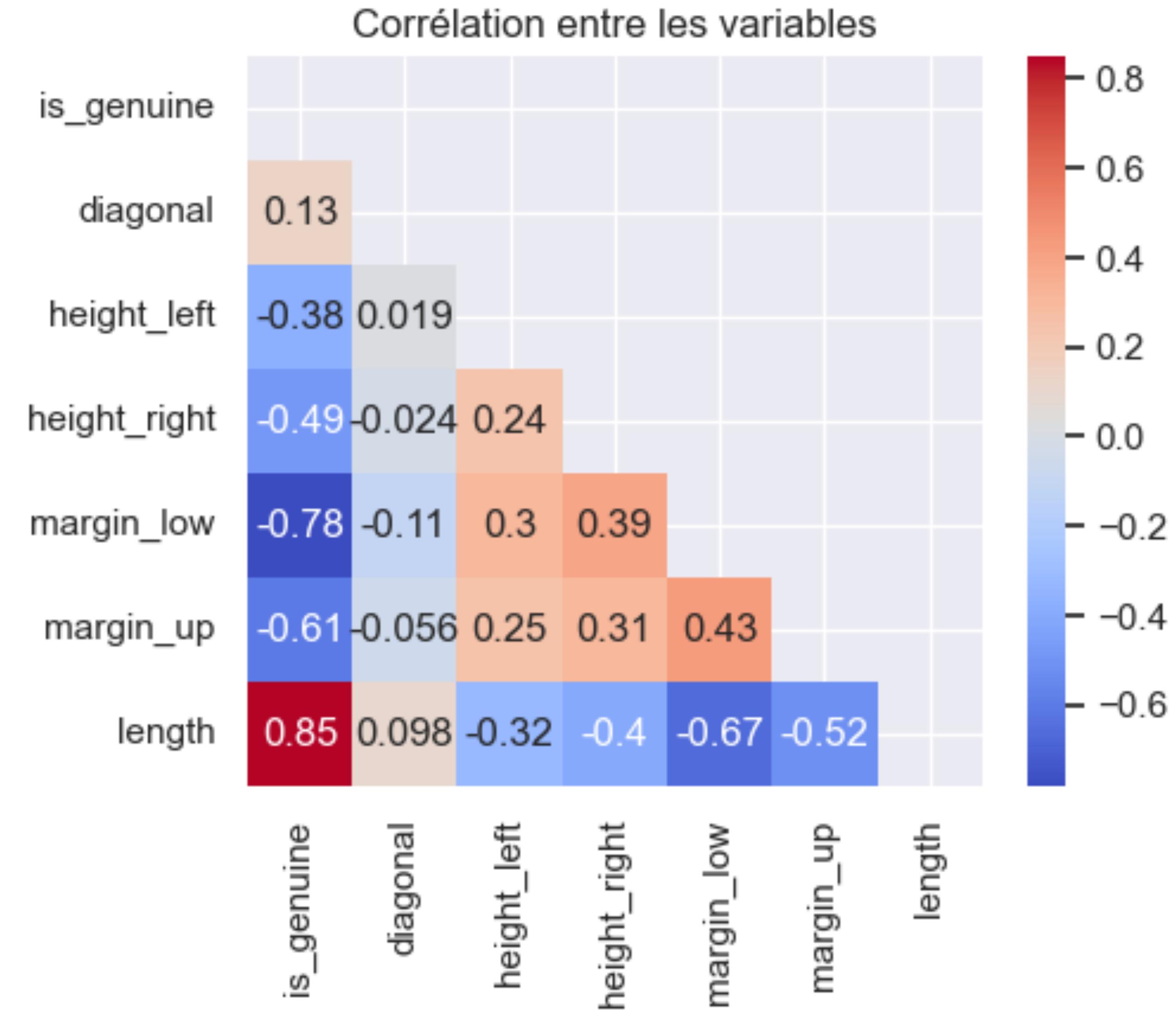
Analyse des variables par paire :

- il y a une distinction significative entre les vrais et les faux billets sur la variable « **length** »,
- la longueur en relation avec les marges inférieurs et supérieures semble être un indicateur de la véracité des billets.



Analyse multi-variée

L'analyse des corrélations entre les variables semble confirmer l'importance des variables « **length** », « **margin_low** » et « **margin_up** » dans la prédiction de la validité des billets.



Valeurs manquantes

Seule la dimension « ***margin_low*** » comporte des valeurs manquantes.

Il nous faut réaliser une **régression linéaire** pour remplir ces manquants.

is_genuine	0
diagonal	0
height_left	0
height_right	0
margin_low	37
margin_up	0
length	0

Régression linéaire

Conditions préalables de la régression linéaire :

- **Linéarité**, il doit exister une relation linéaire entre les variables dépendantes et indépendantes.
- **Normalité**, les résidus doivent être distribués normalement.
- **Homoscédasticité**, la variance des résidus doit être constante.
- **Indépendance**, les résidus doivent être indépendants les uns des autres.
- **Absence de multicolinéarité**, les variables indépendantes ne doivent pas être corrélées entre elles.

Vérification de la linéarité des relations

Entre les variables prédictives et la variable cible « **margin_low** ».

Projection des variables, et traçage de la ligne de régression.

La relation entre les variables est bien linéaire.

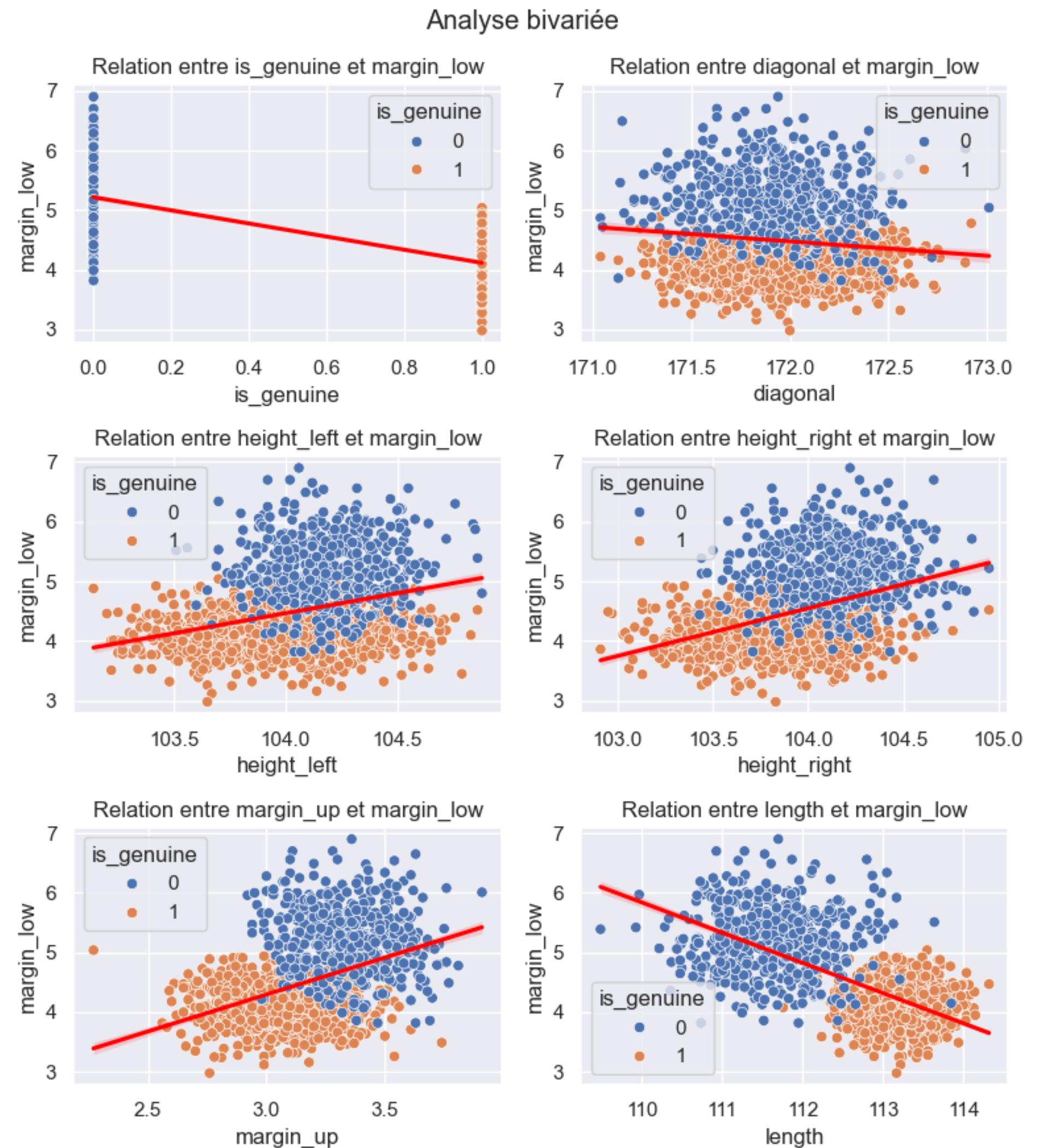
Test RESET de Ramsey :
F-Statistic : 0.1428
P-Value : 0.8669

Examiner si l'ajout de transformations polynomiales des variables explicatives améliore la qualité du modèle.

Hypothèse nulle : le modèle est correctement spécifié.

Avec P-Value > 5%, on ne rejète pas l'hypothèse nulle.

L'ajout de transformations de variables explicatives n'a pas permis de réduire la variance de l'erreur.



Régression linéaire multiple

Modèle qui cherche à établir une relation linéaire entre la variable expliquée et les variables explicatives.

Standardisation des données.

Coefficients :

- « **is_genuine** » explique presque à elle seule la variance de la variable cible.
- quasiment aucune action des autres variables, mise à part une faible participation de « **margin_up** ».

Variables	Coefficients
diagonal	0.001987
height_left	0.001984
height_right	0.005073
margin_up	-0.046796
length	-0.002473
is_genuine	-1.143903

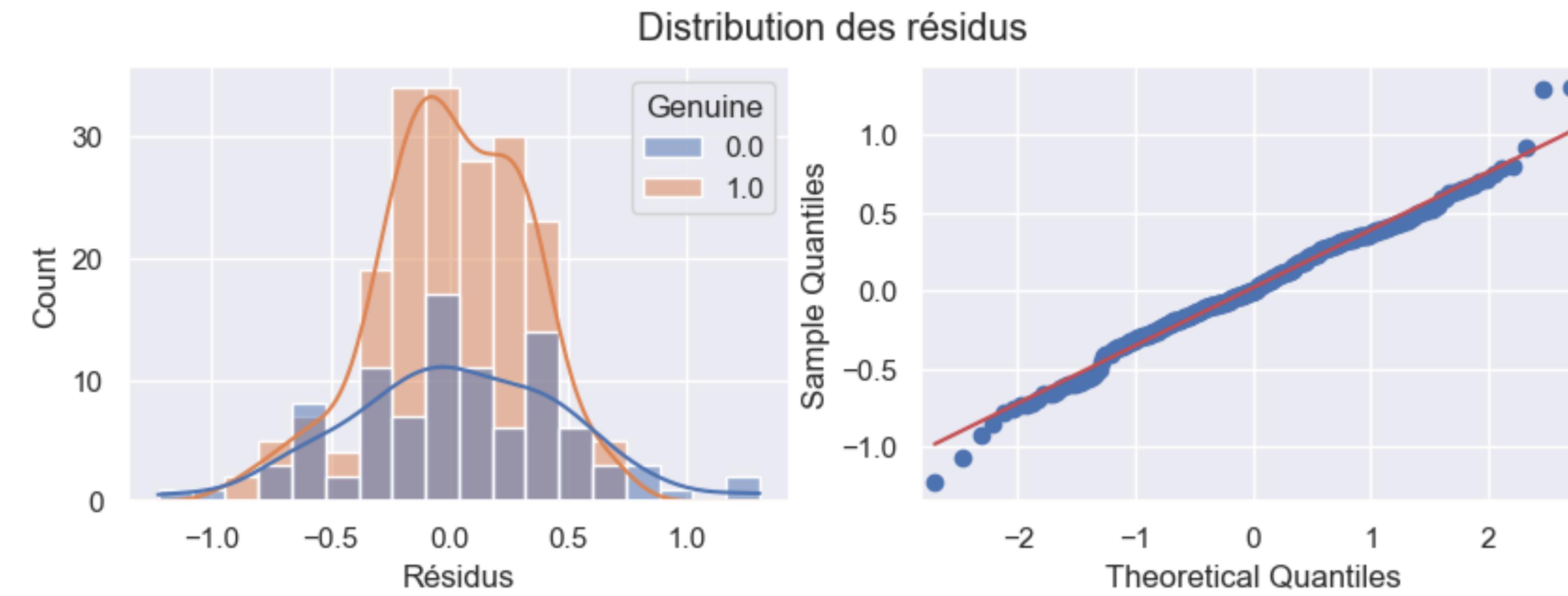
Résultats :

- **R²**, coefficient de détermination : 0.67
- **MAPE**, moyenne de la valeur absolue des erreurs en % : 0.06

Nous obtenons un coefficient de détermination de 0.67, ce qui signifie que 67% de la variance de la variable dépendante est expliquée par les variables indépendantes.

Normalité des résidus

Les résidus sont normalement distribués.



Détection de la multicolinéarité

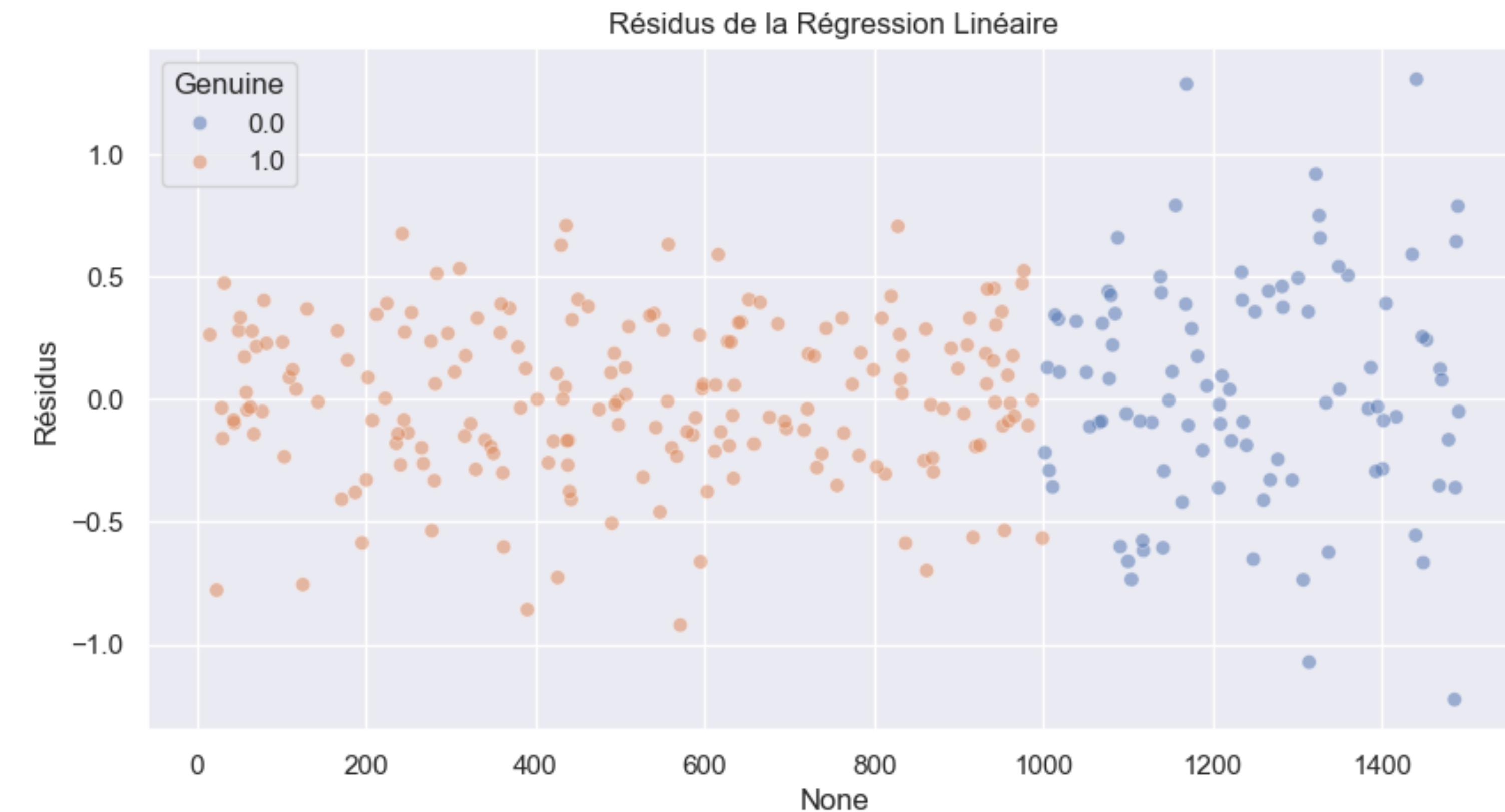
La multicolinéarité affecte la stabilité et l'interprétabilité des coefficients de régression. Le facteur d'inflation de la variance (VIF) quantifie la variance d'un coefficient de régression en raison des corrélations entre les variables prédictives.

Aucune valeur supérieure à 5, le niveau de multicolinéarité est modéré.

	Variables	VIF
0	diagonal	1.028199
1	height_left	1.173711
2	height_right	1.320836
3	margin_up	1.596090
4	length	3.613255
5	is_genuine	4.725732

Homoscédasticité des résidus

La variance des résidus est constante.



Indépendance des résidus

Test de Durbin-Watson :

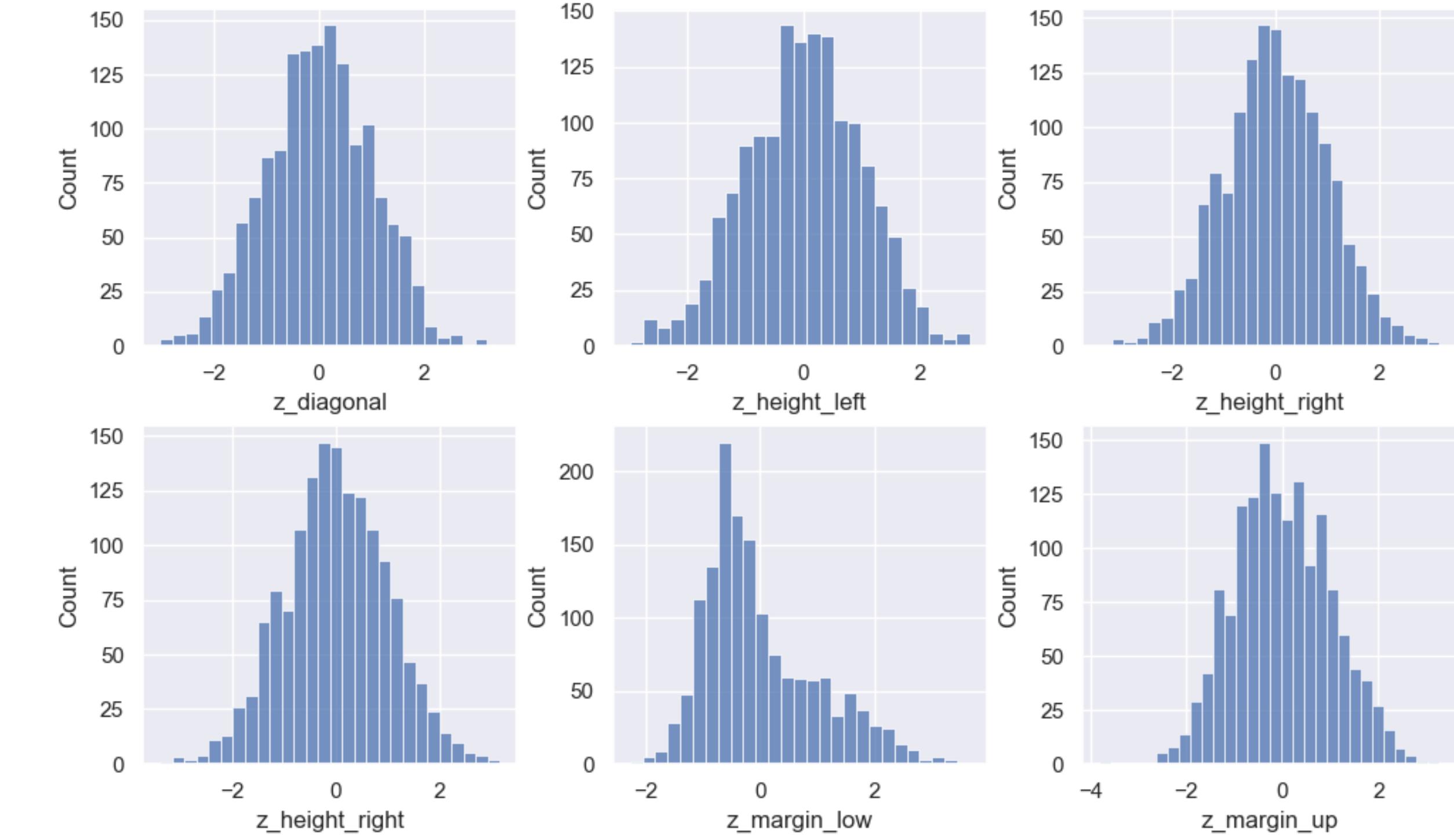
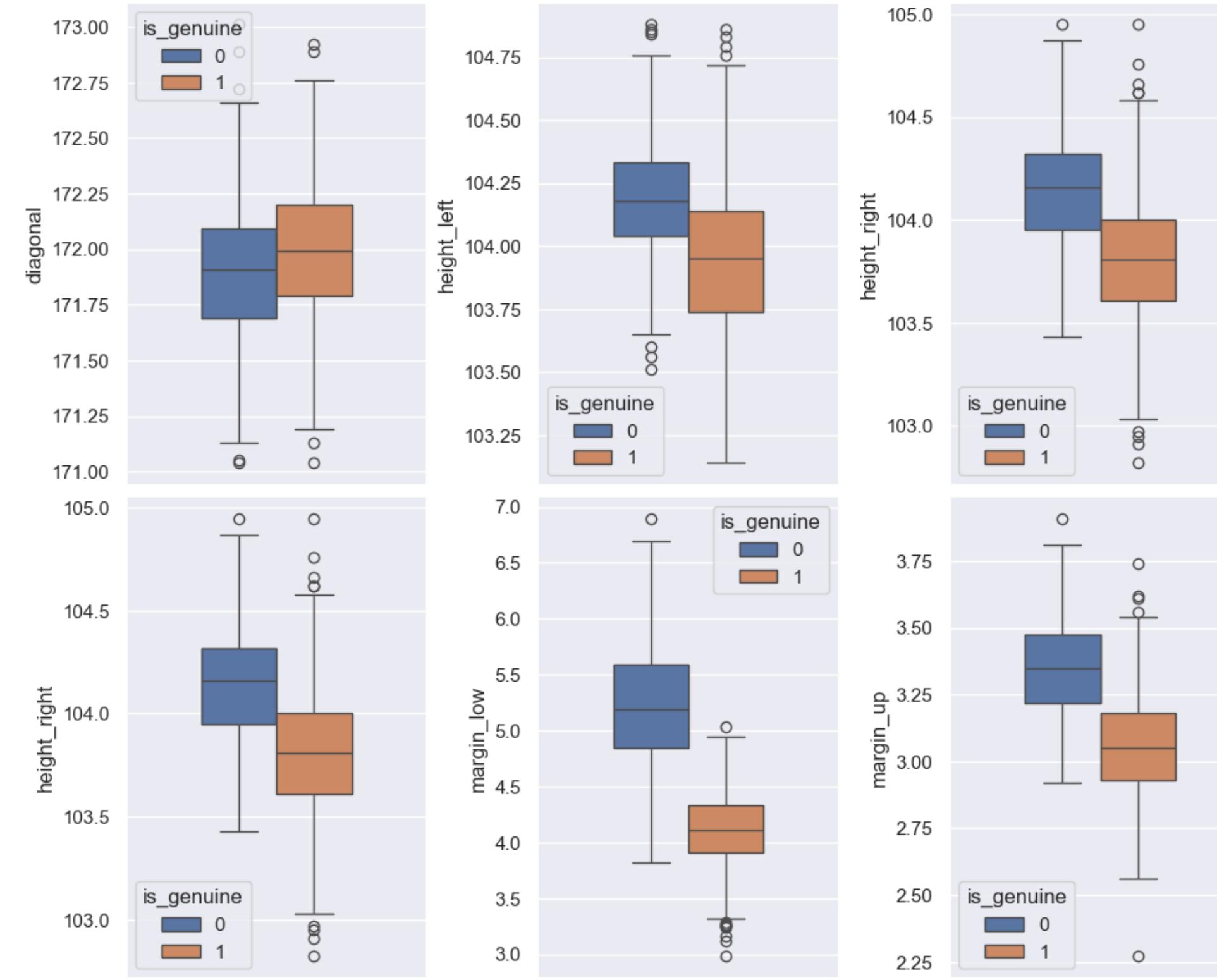
Test la présence d'autocorrélation dans les résidus.

Statistique : 1.85

Une valeur proche de 2 indique qu'il n'y a pas d'autocorrélation.

Tous les pré-requis de la régression linéaire ont été vérifiés. Les résultats sont bons.
Nous pouvons procéder au remplacement des valeurs manquantes par le modèle de régression linéaire.

Analyse des outliers

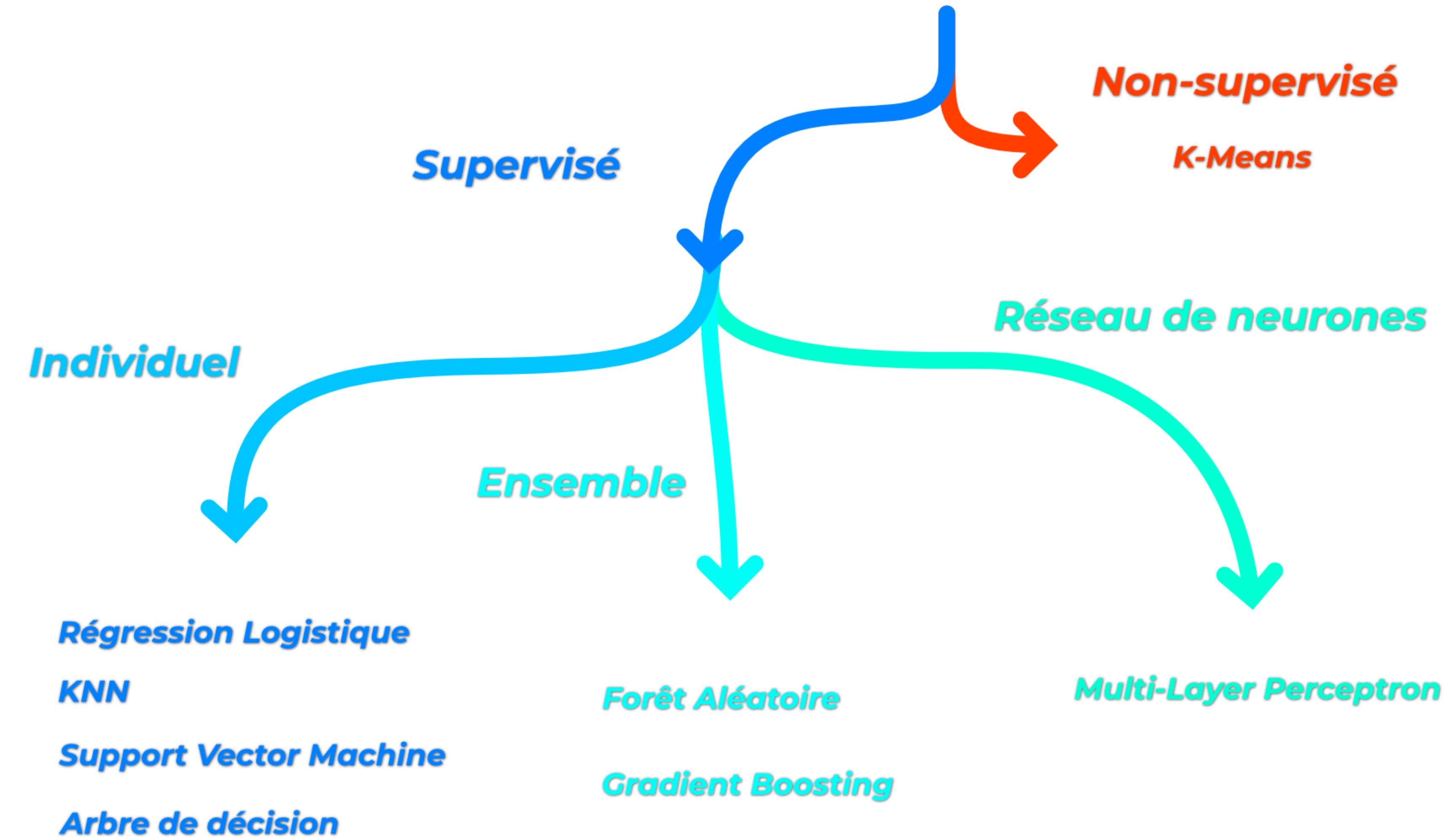


La plupart des valeurs sont à moins de 2 écarts-type de la moyenne.

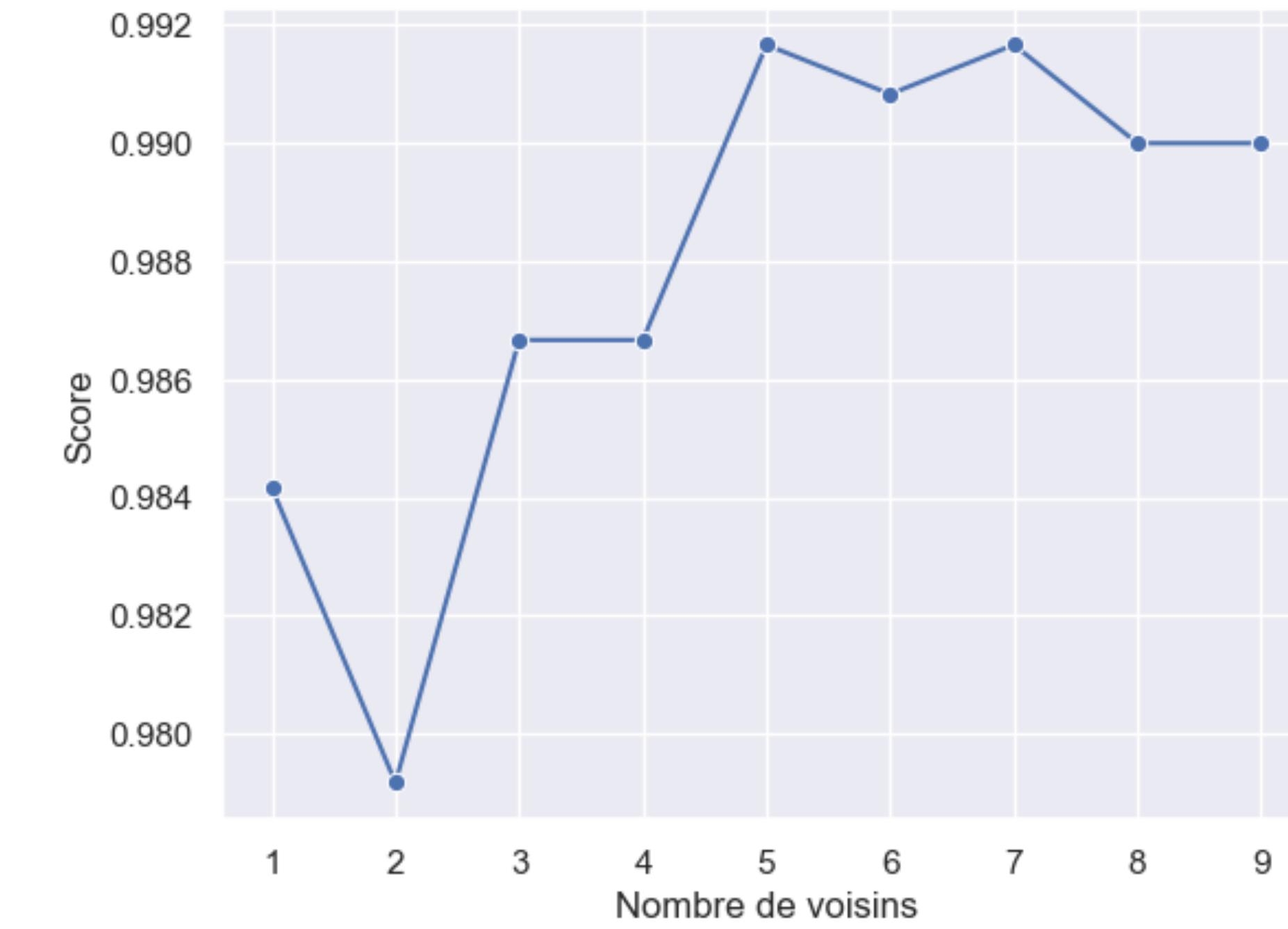
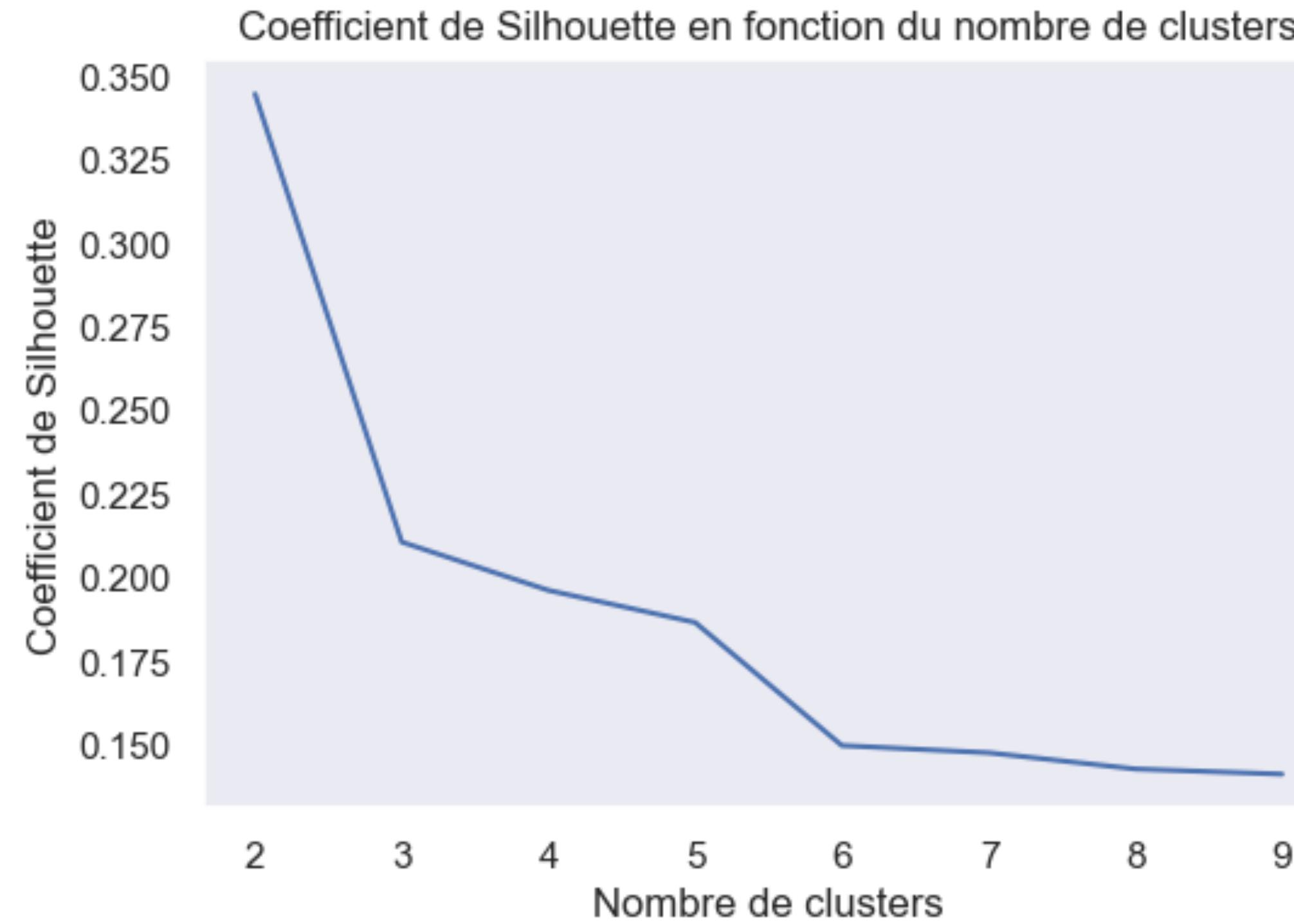
Quelques valeurs sont à plus de 2 écarts-type mais aucun au-delà de 3.

Les données ne comportent pas de valeurs aberrantes ou atypiques.

Algorithmes de classification



K-Means & KNN



Coefficient de silhouette pour déterminer le nombre optimal de clusters : 2.

Score suivant le nombre de voisins. Nombre de voisins optimal : 5.

Hyperparamètres et validation croisée

Utilisation de GridSearchCV et RandomizedSearchCV pour à la fois :

- effectuer une validation croisée
- trouver les meilleurs paramètres des modèles

Enregistrement des temps d'entraînement et de prédiction.

	Modèles	Nombre de fits	Temps d'entraînement par fit (s)	Temps de prédiction (s)
0	Régression Logistique	50	0.050290	0.000072
1	K-Means	5	0.020034	0.000278
2	K-Nearest Neighbors	30	0.004695	0.005293
3	Arbre de décision	50	0.001868	0.000060
4	Forêt aléatoire	50	0.008207	0.001612
5	Gradient Boosting	50	0.006424	0.000284
6	Support Vector Machine	50	0.003636	0.002092
7	Multi-Layer Perceptron	50	0.039353	0.000117

Evaluation des modèles

	Modèles	Accuracy	Precision	Recall	F1-score	ROC-AUC	Score Train	Score Test
0	Régression Logistique	0.986667	0.986808	0.986667	0.986691	0.987559	0.990833	0.986667
1	K-Means	0.973333	0.973333	0.973333	0.973333	0.971400		
2	K-Nearest Neighbors	0.990000	0.990005	0.990000	0.989991	0.988345	0.991667	0.990000
3	Arbre de décision	0.966667	0.967262	0.966667	0.966783	0.967968	0.975000	0.966667
4	Forêt aléatoire	0.993333	0.993333	0.993333	0.993333	0.992850	0.992500	0.993333
5	Gradient Boosting	0.986667	0.986667	0.986667	0.986667	0.985700	0.997500	0.986667
6	Support Vector Machine	0.990000	0.990005	0.990000	0.989991	0.988345	0.993333	0.990000
7	Multi-Layer Perceptron	0.993333	0.993333	0.993333	0.993333	0.992850	0.992500	0.993333

Ce jeu de données (assez simple) nous permet d'obtenir pour tous les modèles, de très bon scores. Avec le modèle le moins efficace, nous obtenons tout de même 0.96 d'Accuracy. La marge d'amélioration grâce au réglage des hyperparamètres est assez faible. Il n'y a **pas d'overfitting**.

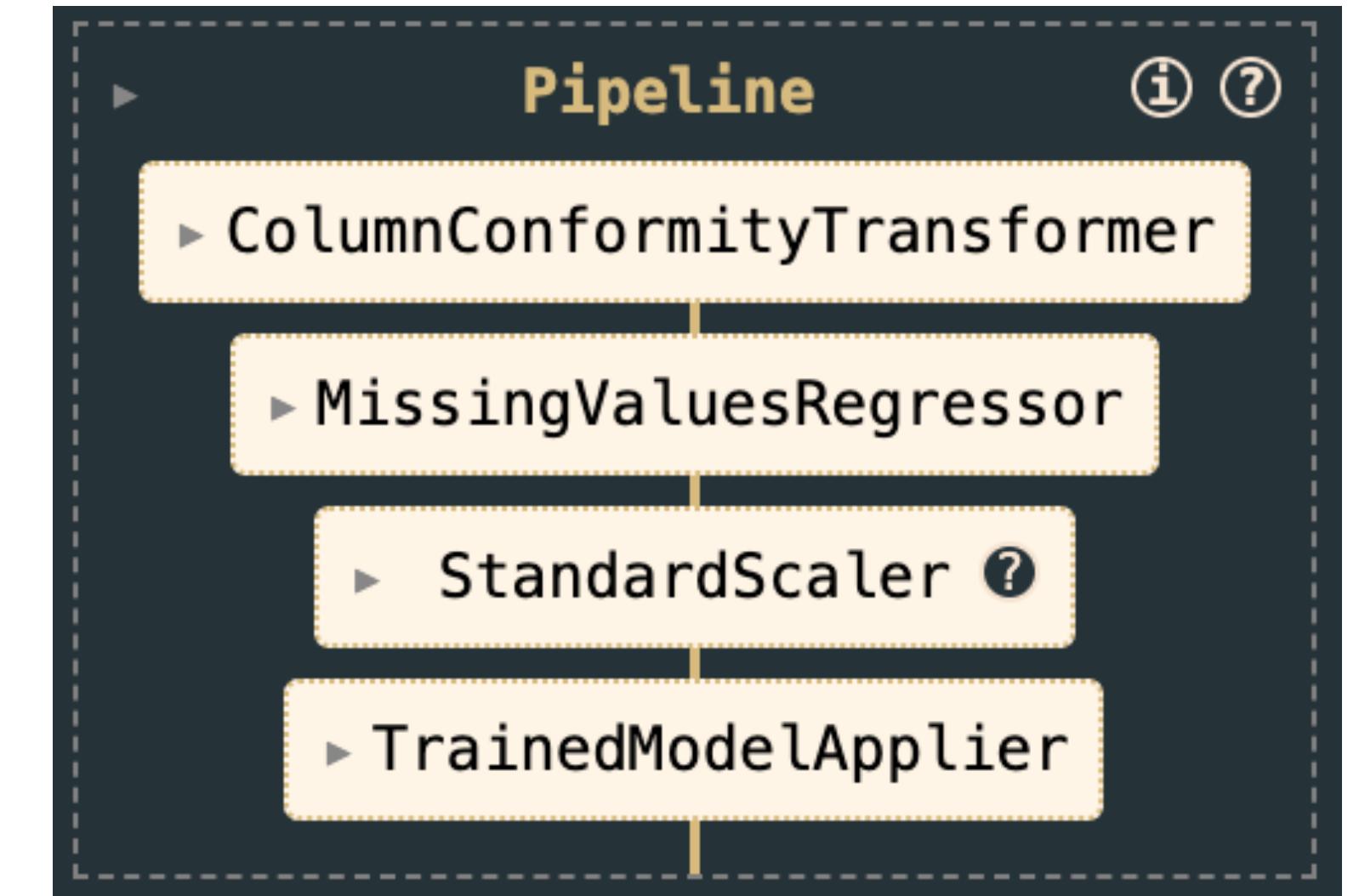
Les meilleurs scores sont obtenus par la **Forêt Aléatoire** et le **Multi-Layer Perceptron**.

Nous retenons la **Forêt Aléatoire**, pour sa meilleure résistance au sur-apprentissage, et à sa meilleure interprétabilité.

Modèle de production

Pipeline de mise en production :

- Traitement de la conformité des variables
- Traitement des données manquantes par régression linéaire
- Standardisation des données
- Application du modèle pré-entraîné



Résultats des prédictions

Analyse de l'authenticité des billets				
	ID	Prédiction de l'authenticité	Probabilité Vrai	Probabilité Faux
0	A_1	Faux billet	0.082850	0.917150
1	A_2	Faux billet	0.083494	0.916506
2	A_3	Faux billet	0.015797	0.984203
3	A_4	Vrai billet	0.986041	0.013959
4	A_5	Vrai billet	0.998316	0.001684