OPENCLASSROOMS

PROJET n°12 : Détectez des faux billets avec R ou python

🎯 Objectifs

L'Organisation nationale de lutte contre le faux-monnayage (ONCFM) souhaite mettre en place un algorithme de détection automatique des faux billets en euros.

Contexte :

- 🏦 **L'ONCFM lutte contre la contrefaçon des billets en euros**
- 🔍 **Des différences géométriques existent entre vrais et faux billets**
- 📊 **Ces différences sont difficilement détectables à l'œil nu**

Besoins :

- 🤖 **Automatiser la détection des faux billets**
- 📏 **Utiliser les caractéristiques géométriques**
- 📈 **Maximiser la précision de la détection**

Caractéristiques des données

Dataset :

- 📈 **1500 billets au total**
- ✅ **1000 vrais billets**
- ❌ **500 faux billets**

Variables géométriques (en mm) :

- 📏 **length : longueur du billet**
- 📐 **height_left : hauteur côté gauche**
- 📐 **height_right : hauteur côté droit**
- ↕️ **margin_up : marge supérieure**
- ↕️ **margin_low : marge inférieure**
- ↗️ **diagonal : diagonale du billet**

📊 Chargement et aperçu des données

- 📥 **Chargement du jeu de données `billets.csv` avec `pd.read_csv()`.**
- 🔍 **Examen de la structure : dimensions (`.shape`), types (`.dtypes`), aperçu (`.head()`), statistiques descriptives (`.describe()`).**

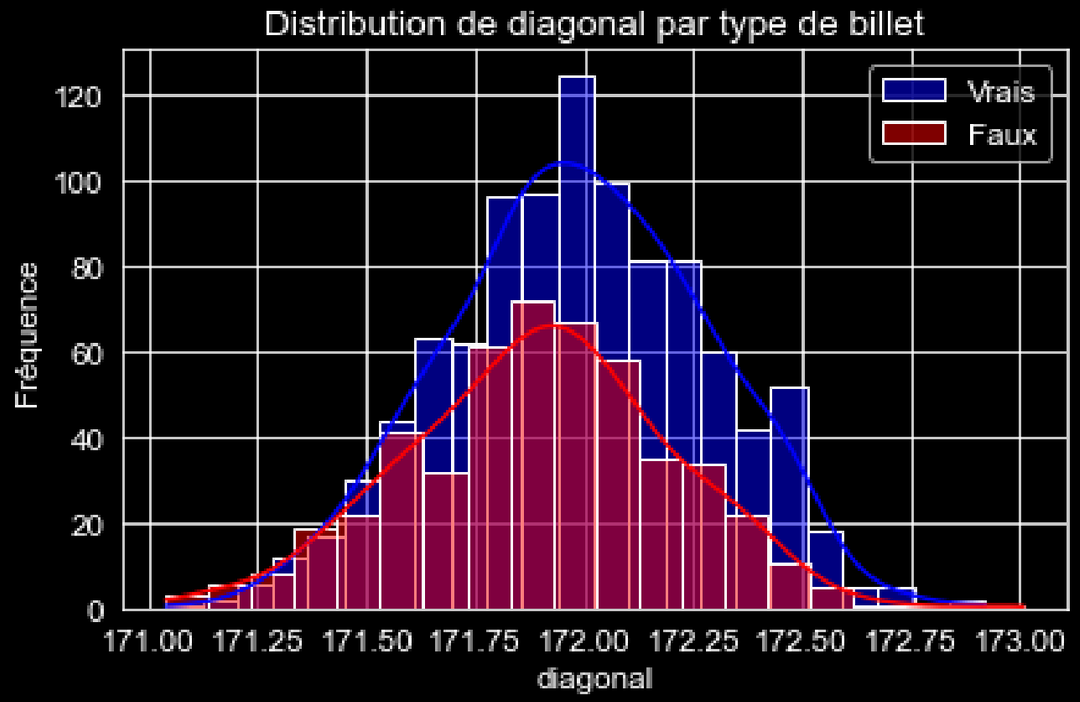
Statistiques descriptives :

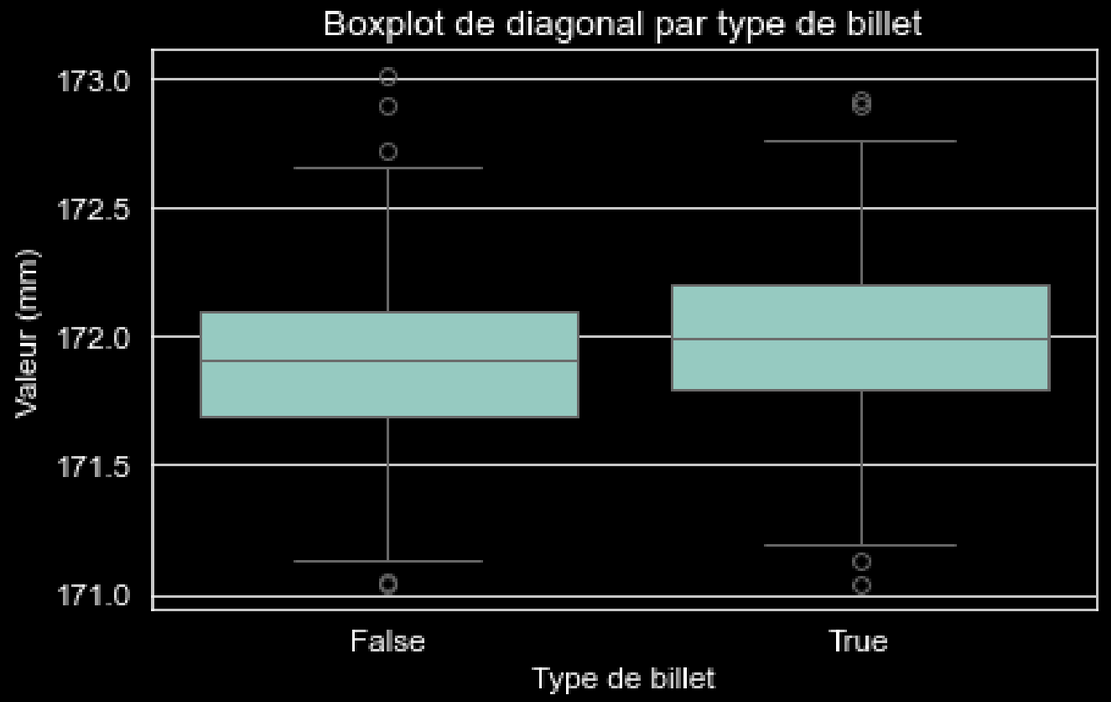
	diagonal	height_left	height_right	margin_low	margin_up	length
count	1500.000000	1500.000000	1500.000000	1463.000000	1500.000000	1500.00000
mean	171.958440	104.029533	103.920307	4.485967	3.151473	112.67850
std	0.305195	0.299462	0.325627	0.663813	0.231813	0.87273
min	171.040000	103.140000	102.820000	2.980000	2.270000	109.49000
25%	171.750000	103.820000	103.710000	4.015000	2.990000	112.03000
50%	171.960000	104.040000	103.920000	4.310000	3.140000	112.96000
75%	172.170000	104.230000	104.150000	4.870000	3.310000	113.34000
max	173.010000	104.880000	104.950000	6.900000	3.910000	114.44000

Caractéristiques des données

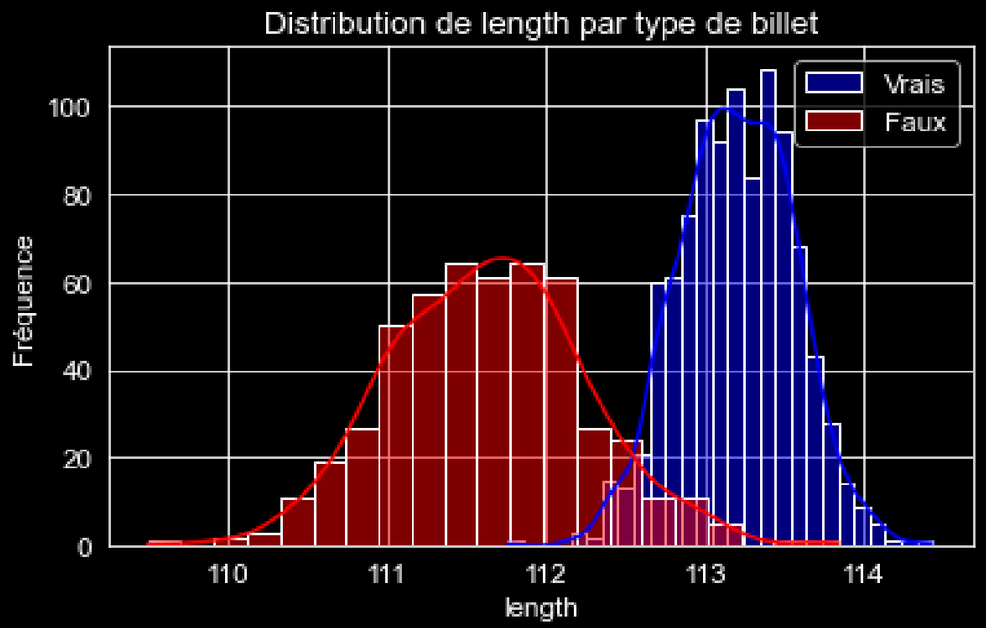
=== RÉSUMÉ GLOBAL DES VALEURS ABERRANTES ===

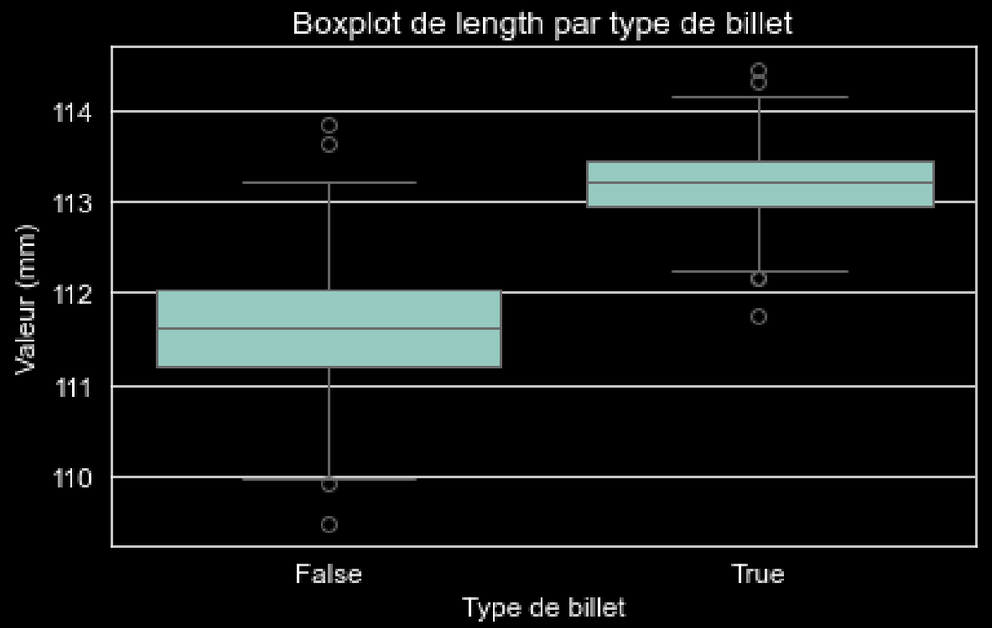
Aspect	Conclusion
Variable la plus affectée	diagonal (6 valeurs aberrantes)
Répartition authentique/contrefait	Authentiques: 7, Contrefaits: 8
Impact sur la détection	À utiliser comme indicateur potentiel de contrefaçon

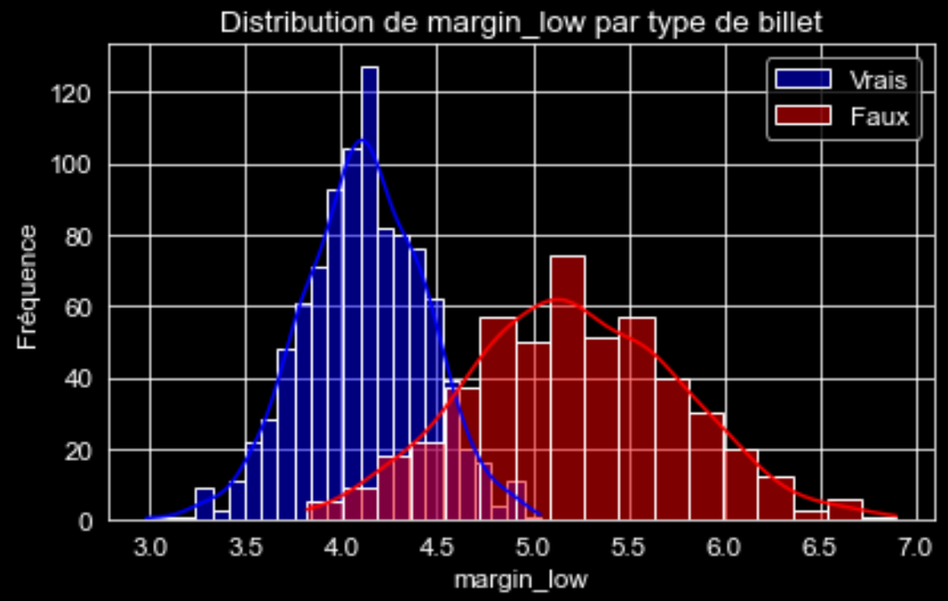


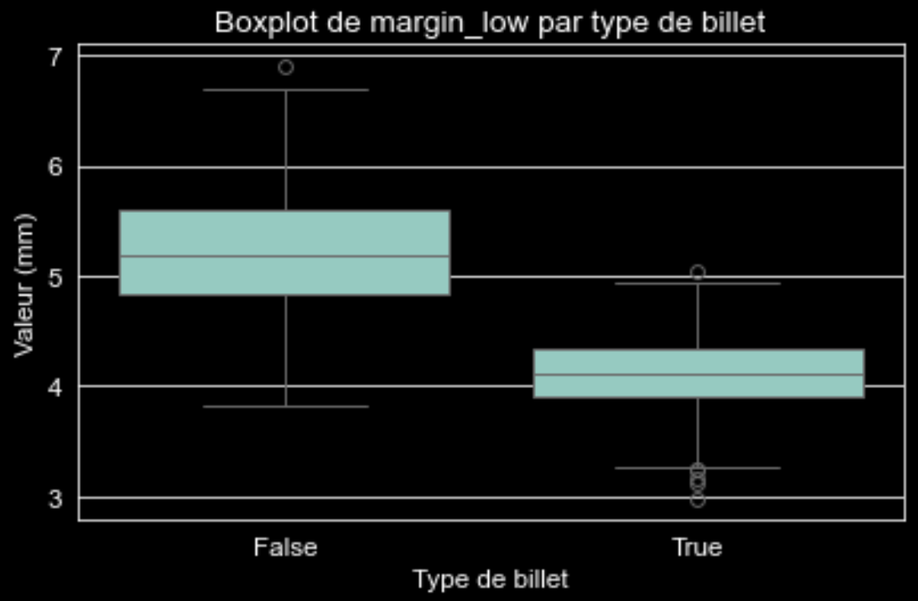


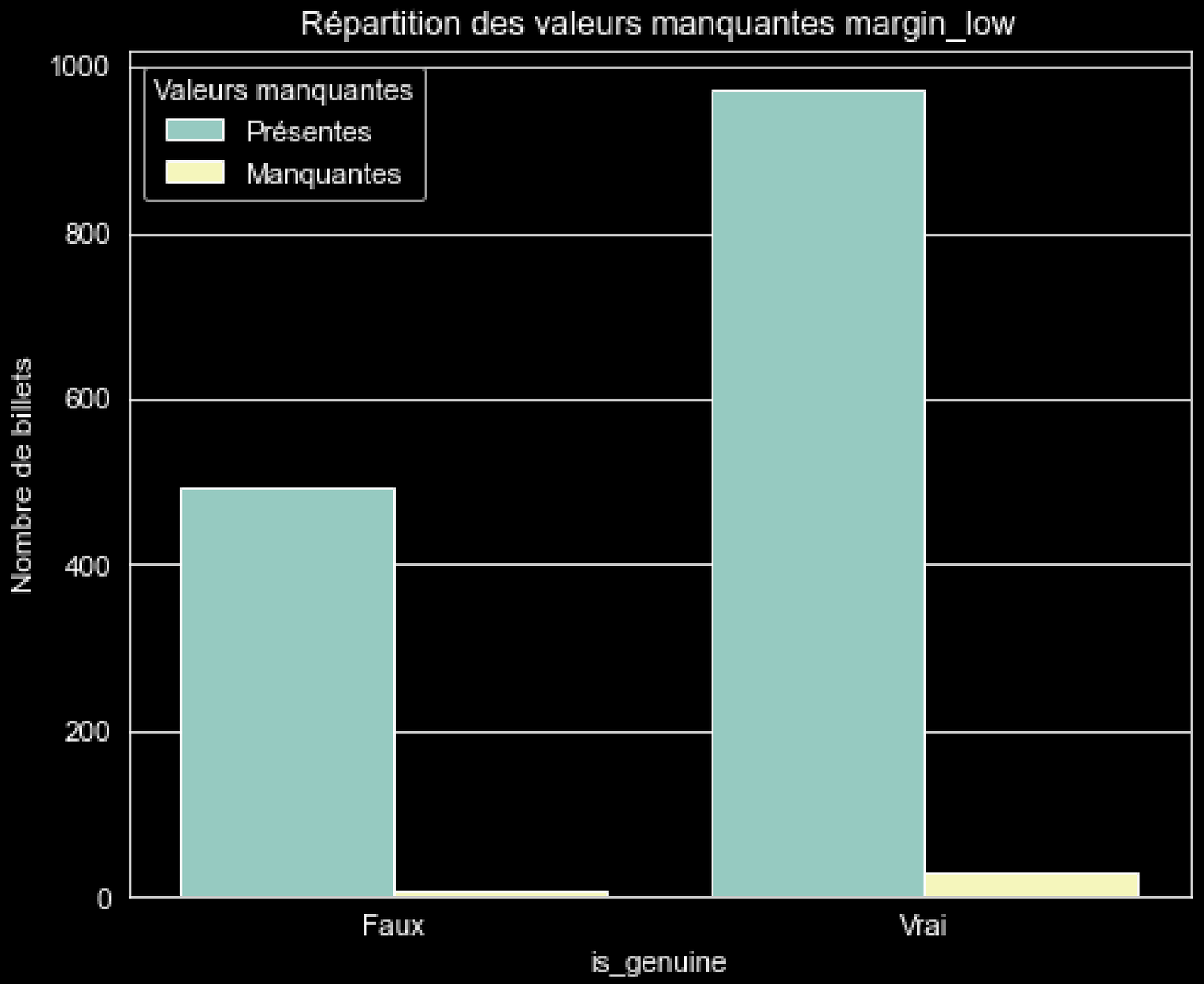
Caractéristiques des données

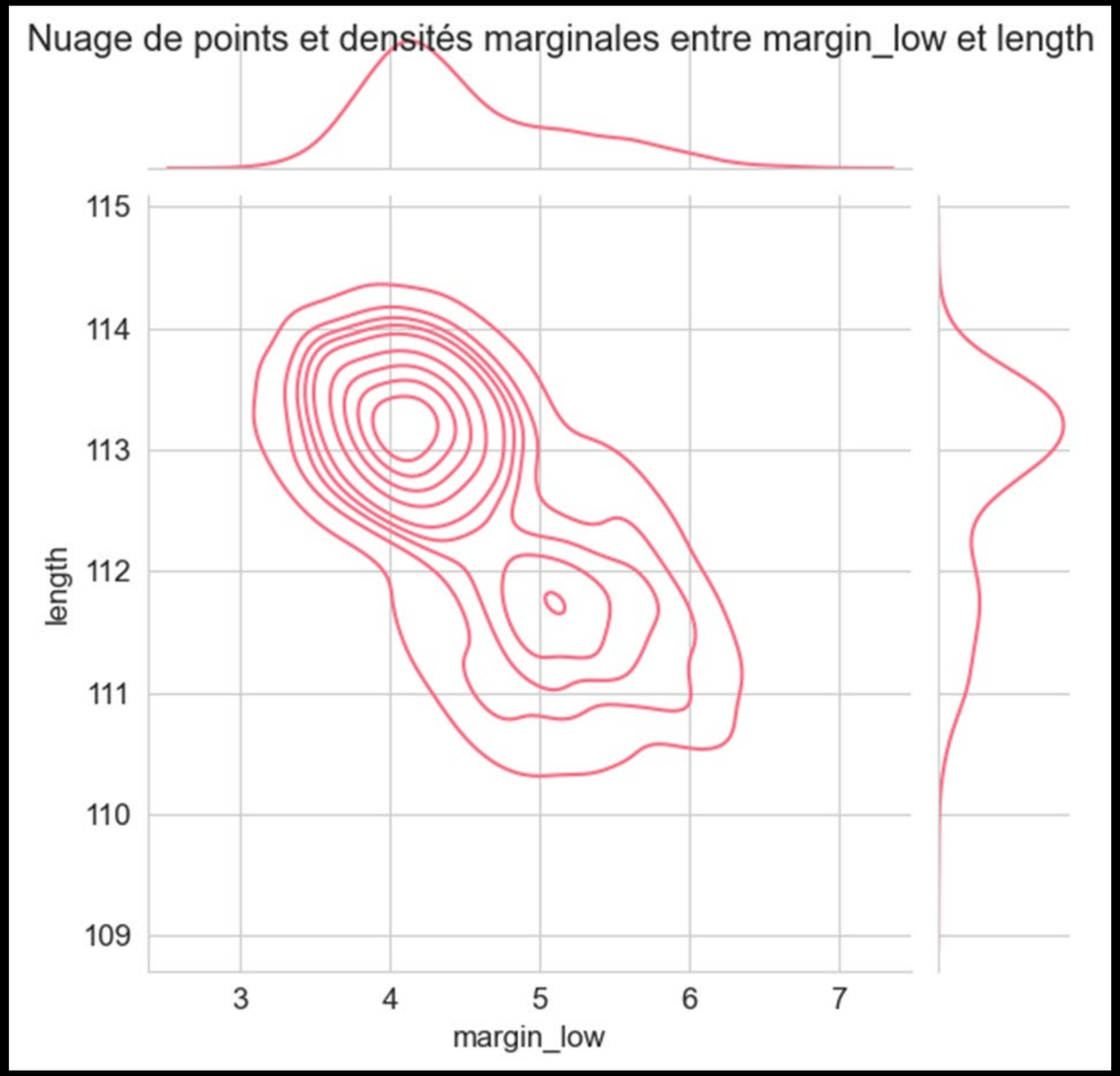


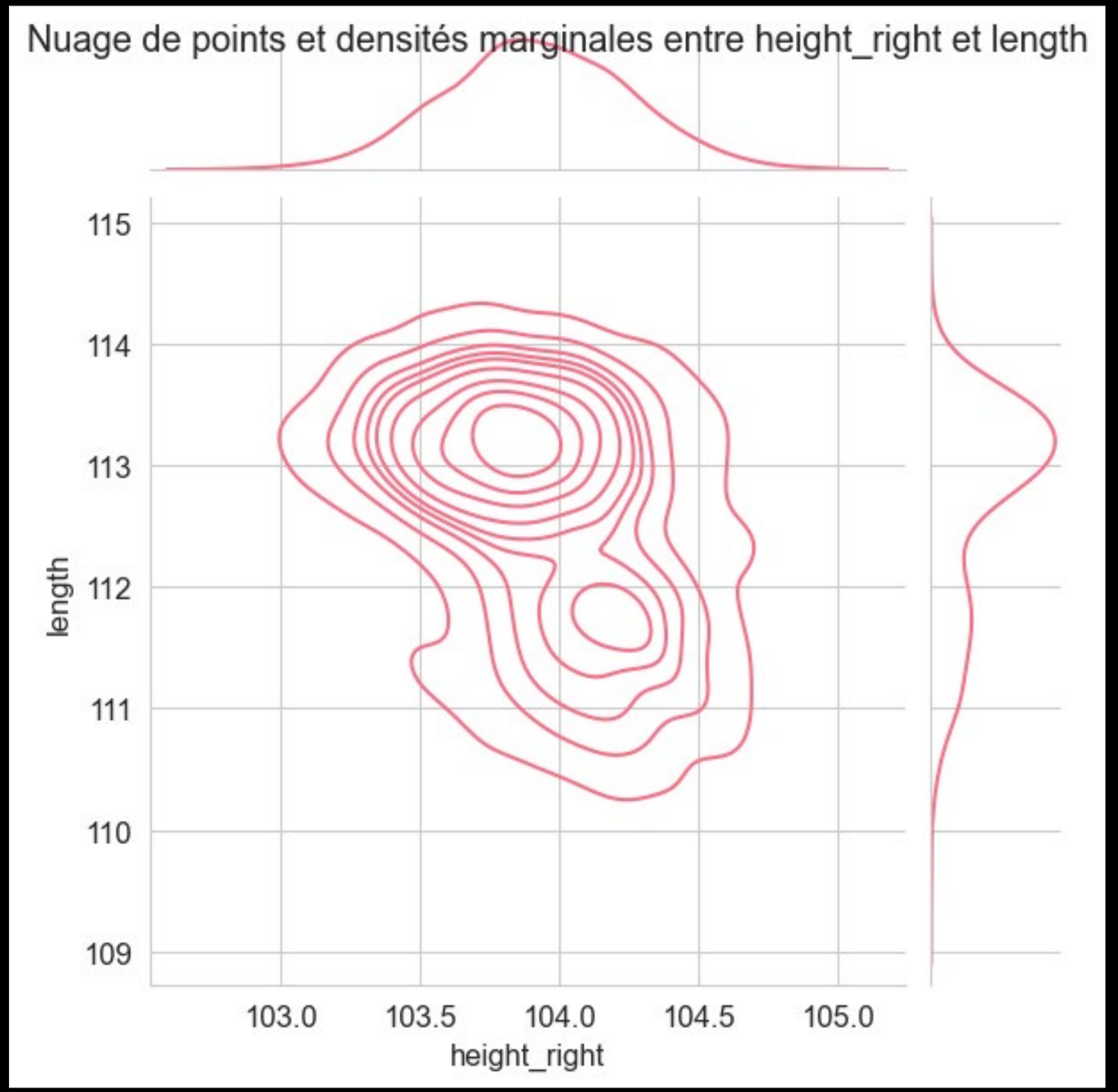


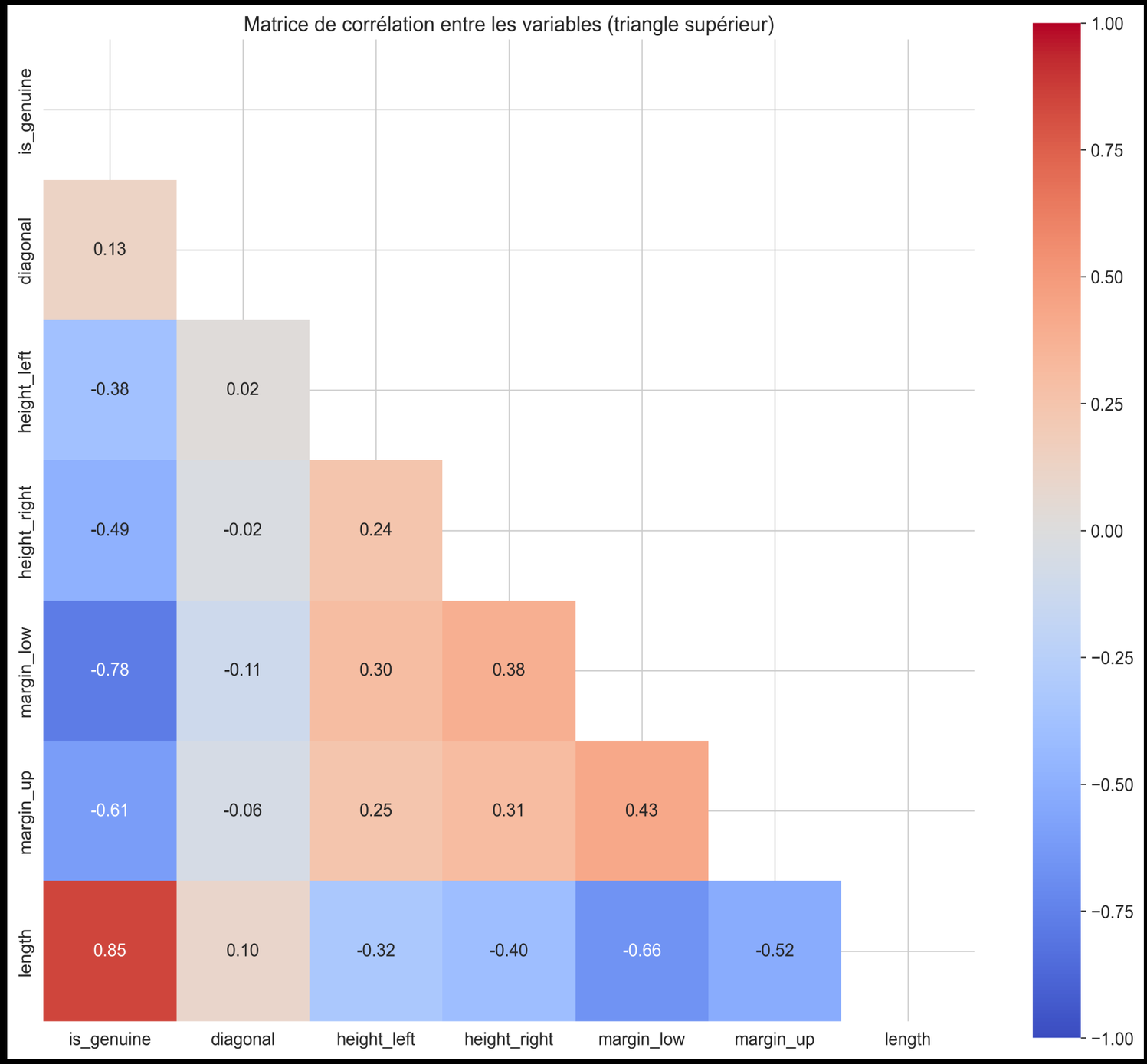





Répartition des valeurs manquantes margin_low
Valeurs manquantes
Présentes
Manquantes
1000
800
600
400
200
0
Nombre de billets
Faux
Vrai
is_genuine


Nuage de points et densités marginales entre margin_low et length
115
114
113
112
111
110
109
length
3
4
5
6
7
margin_low


Nuage de points et densités marginales entre height_right et length
115
114
113
112
111
110
109
length
103.0
103.5
104.0
104.5
105.0
height_right


Matrice de corrélation entre les variables (triangle supérieur)
is_genuine
diagonal
height_left
height_right
margin_low
margin_up
length
0.13
-0.38
0.02
-0.49
-0.02
0.24
-0.78
-0.11
0.30
0.38
-0.61
-0.06
0.25
0.31
0.43
0.85
0.10
-0.32
-0.40
-0.66
-0.52
is_genuine
diagonal
height_left
height_right
margin_low
margin_up
length
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00

Relation avec margin_low Dans le but de prévoir les valeurs manquantes

'length':
Corrélation de Pearson: -0.67 (p-value < 0.05)
Corrélation de Spearman: -0.59 (p-value < 0.05) => forte corrélation monotone négative significative.
Corrélation Xicorr (ordered): 0.56 (différence ordered-unordered = 0.59, p-value < 0.05).

'height_right':
Corrélation de Pearson: 0.39 (p-value < 0.05) 'length'.
Corrélation de Spearman: 0.40 (p-value < 0.05)
Corrélation Xicorr (ordered): 0.60 (différence ordered-unordered = 0.12, p-value < 0.05)

Relation avec « is_genuine » Afin de prévoir si un biais de banque est une contrefaçon

'length':
Corrélation de Pearson: -0.85 (p-value < 0.05)
Corrélation de Spearman: -0.87 (p-value < 0.05)
Corrélation Xicorr (ordered): 0.70 (différence ordered-unordered = 0.59, p-value < 0.05).

'margin_low' :
Corrélation de Pearson: -0.78 (p-value < 0.05)
Corrélation de Spearman: -0.67 (p-value < 0.05)
Corrélation Xicorr (ordered): 0.56 (différence ordered-unordered = 0.59, p-value < 0.05**)**

'margin_up' :
Corrélation de Pearson: -0.61 (p-value < 0.05)
Corrélation de Spearman: -0.52 (p-value < 0.05)
Corrélation Xicorr (ordered): 0.70 (différence ordered-unordered = 0.25, p-value < 0.05).

Caractéristiques des données

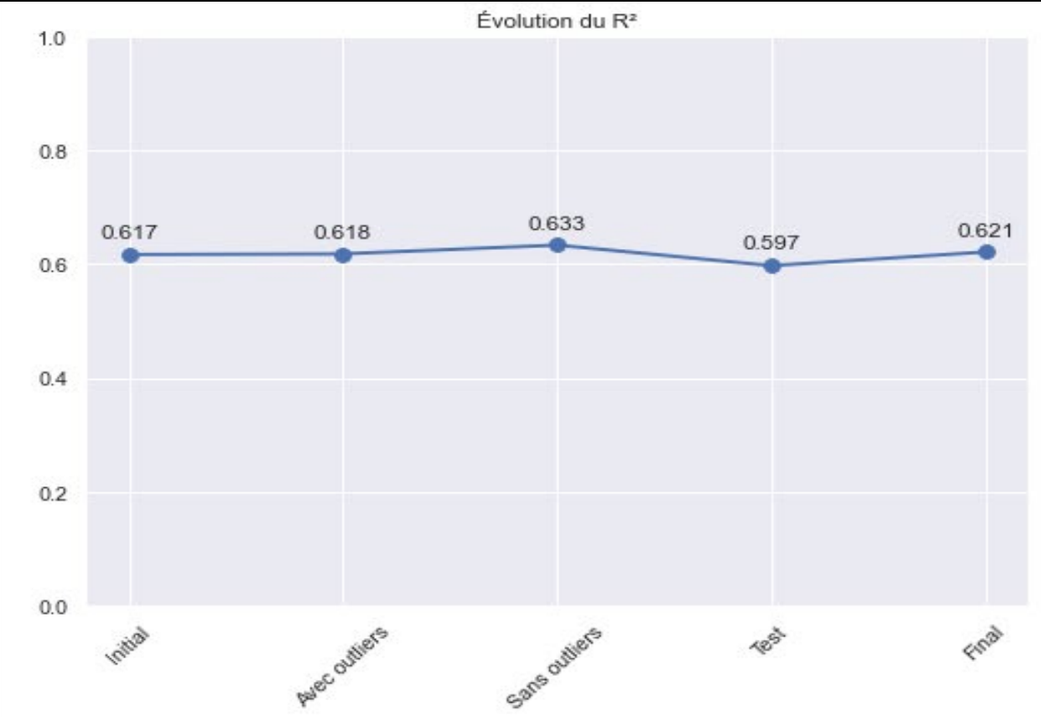
Préparation des Données pour l'Analyse

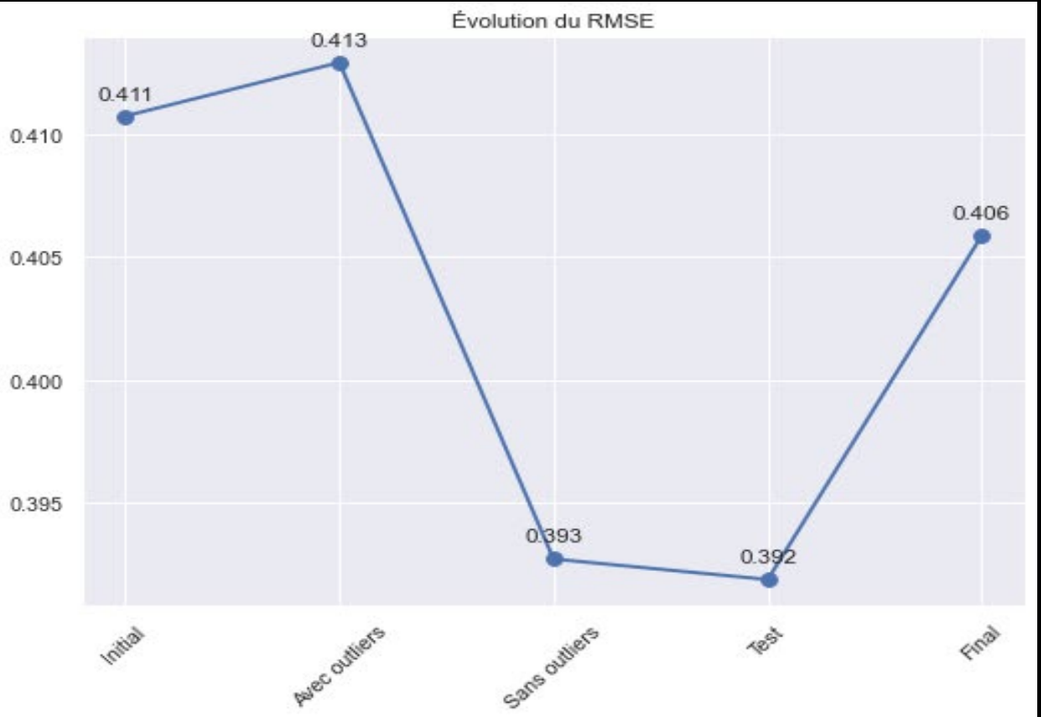
Division Train/Test

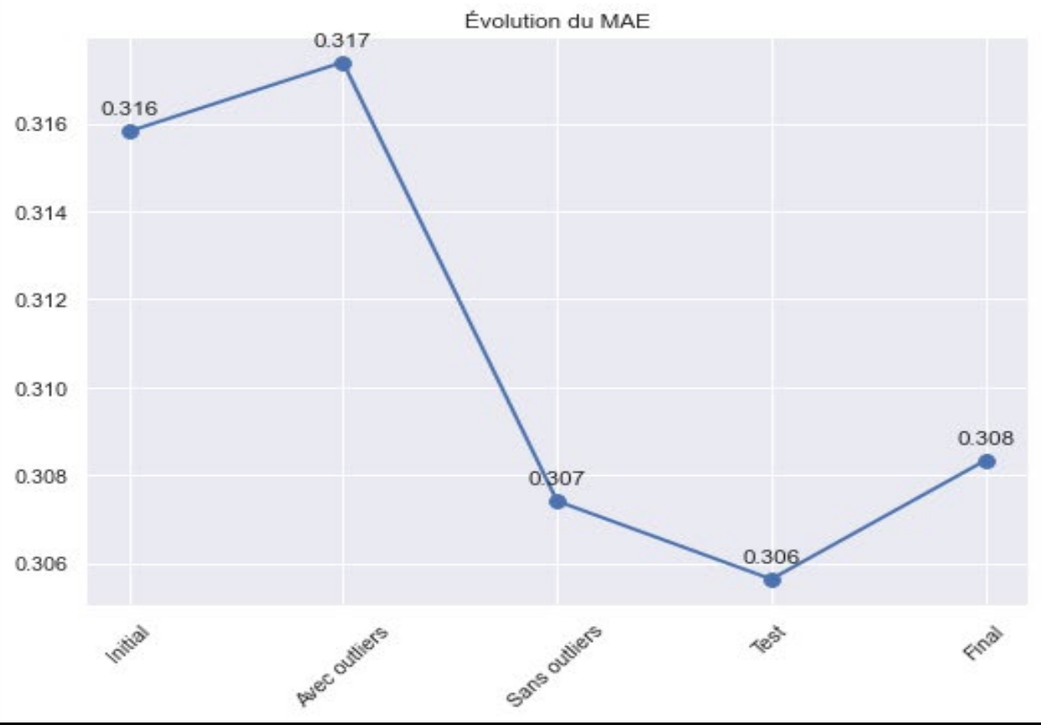
- ⚖️ **Stratification :**
  - **Maintien des proportions vrai/faux billets**
  - **Évitement du biais d'échantillonnage**

Standardisation

- 📏 **Standardisation RobustScaler :**
  - **Utilisation médiane/IQR vs moyenne/écart-type**
  - **Robustesse aux valeurs extrêmes**

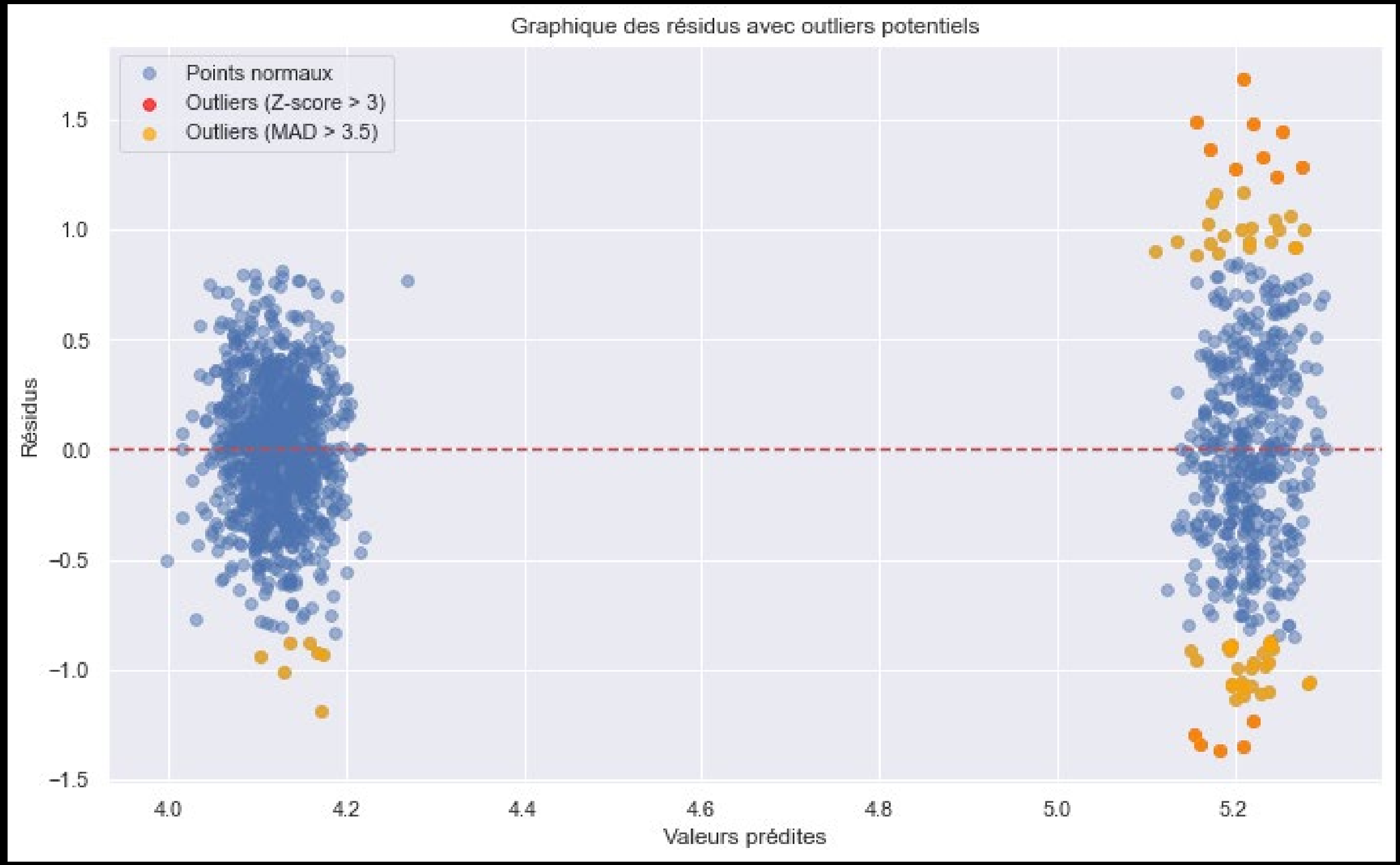

Évolution du R²
1.0
0.8
0.6
0.4
0.2
0.0
0.617
0.618
0.633
0.597
0.621
Initial
Avec outliers
Sans outliers
Test
Final


Évolution du RMSE
0.413
0.411
0.410
0.405
0.400
0.395
0.406
0.393
0.392
Initial
Avec outliers
Sans outliers
Test
Final


Évolution du MAE
0.317
0.316
0.316
0.314
0.312
0.310
0.308
0.306
0.308
0.307
0.306
Initial
Avec outliers
Sans outliers
Test
Final


Évolution du Variance expliquée
1.0
0.8
0.6
0.4
0.2
0.0
0.617
0.618
0.633
0.610
0.621
Initial
Avec outliers
Sans outliers
Test
Final

Caractéristiques des données



Le Traitement des Valeurs Manquantes

Facteurs d'inflation de la variance (VIF) :

	Variable	VIF
0	diagonal	169341.929149
1	height_left	112879.408271
2	height_right	100054.247709
4	length	22676.458356
3	margin_up	260.785987

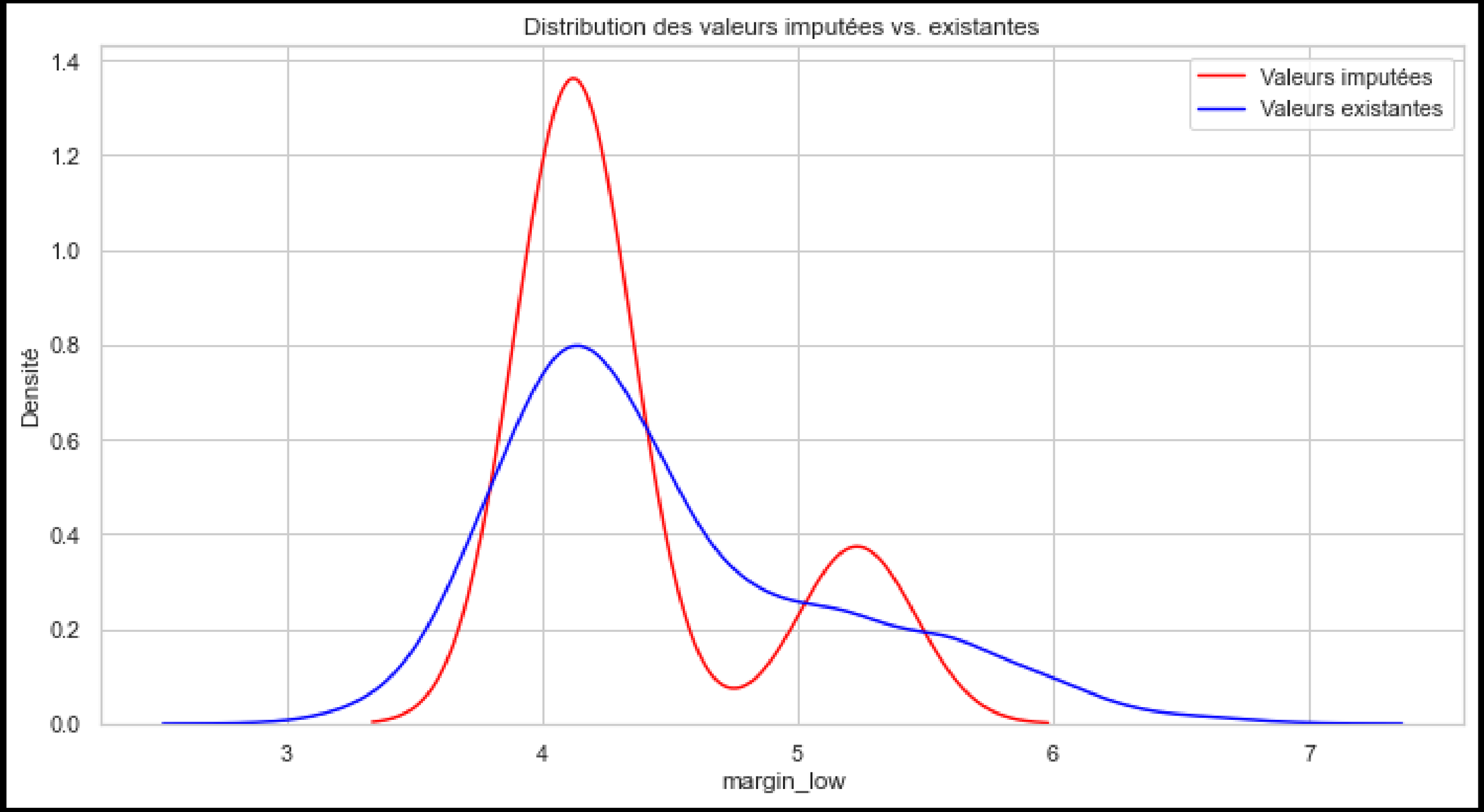
- Mesurent la multicolinéarité
- Un VIF > 10 indique déjà une forte multicolinéarité

Coefficients de régression :

	Variable	Coefficient
0	diagonal	-0.111060
1	height_left	0.184122
2	height_right	0.257138
3	margin_up	0.256188
4	length	-0.409103

Les coefficients de régression montrent l'impact de chaque variable sur margin_low

- length a le plus fort impact (-0.409103)
- diagonal a le plus faible impact (-0.111060)


Distribution des valeurs imputées vs. existantes
Valeurs imputées
Valeurs existantes
Densité
1.4
1.2
1.0
0.8
0.6
0.4
0.2
0.0
3
4
5
6
7
margin_low

Caractéristiques des données

Analyse des hypothèses :

1. **Linéarité : Respectée.**
2. **Normalité des résidus : Non respectée (Test Shapiro-Wilk, p-value = 1.509e-04).**
3. **Homoscédasticité : Non respectée (Test Breusch-Pagan, p-value ≈ 9.02e-27).**
4. **Indépendance des erreurs : Respectée (Durbin-Watson = 1.957).**
5. **Absence de multicolinéarité : Non respectée (VIF max = 170738.73).**

Aspect	Évaluation	Interprétation
Performance globale	R^2 final = 0.6213, RMSE final = 0.4059	À améliorer
Stabilité du modèle	MAE = 0.3083	Instable
Robustesse aux outliers	Outliers détectés = 0.9%	Acceptable
Respect des hypothèses	DW = 2.04, Het. p-value = 9.27e-34	Non validé
Impact sur les données	Var. exp. = 0.6213	À améliorer

Entrainement et optimisation des modèles

📊 **Régression logistique** 🎯 **K-means** 🔍 **KNN** 🌳 **Random Forest**

Préparation des Données pour l'Analyse

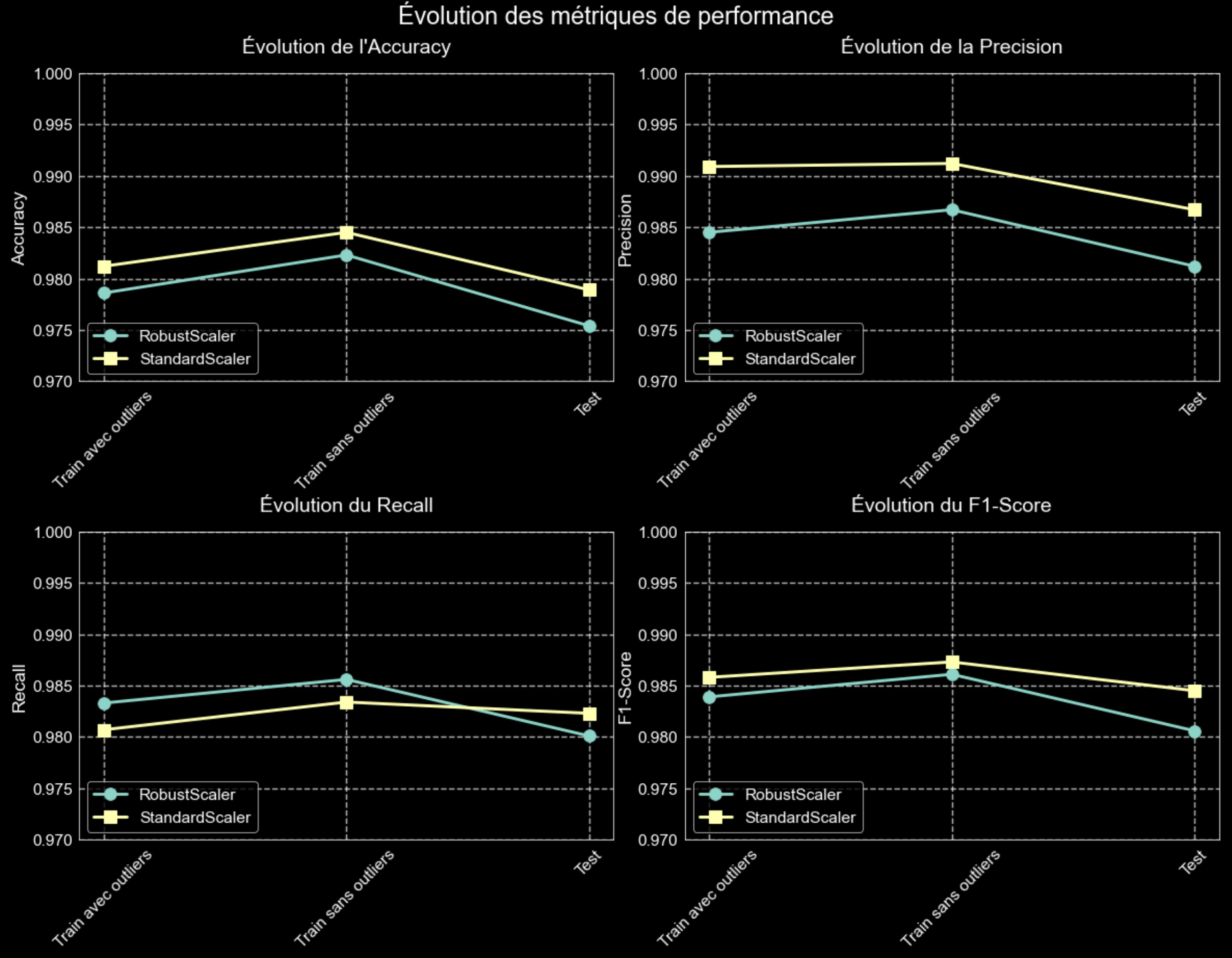
Division Train/Test

- ⚖️ **Stratification :**
  - **Maintien des proportions vrai/faux billets**
  - **Évitement du biais d'échantillonnage**

Standardisation

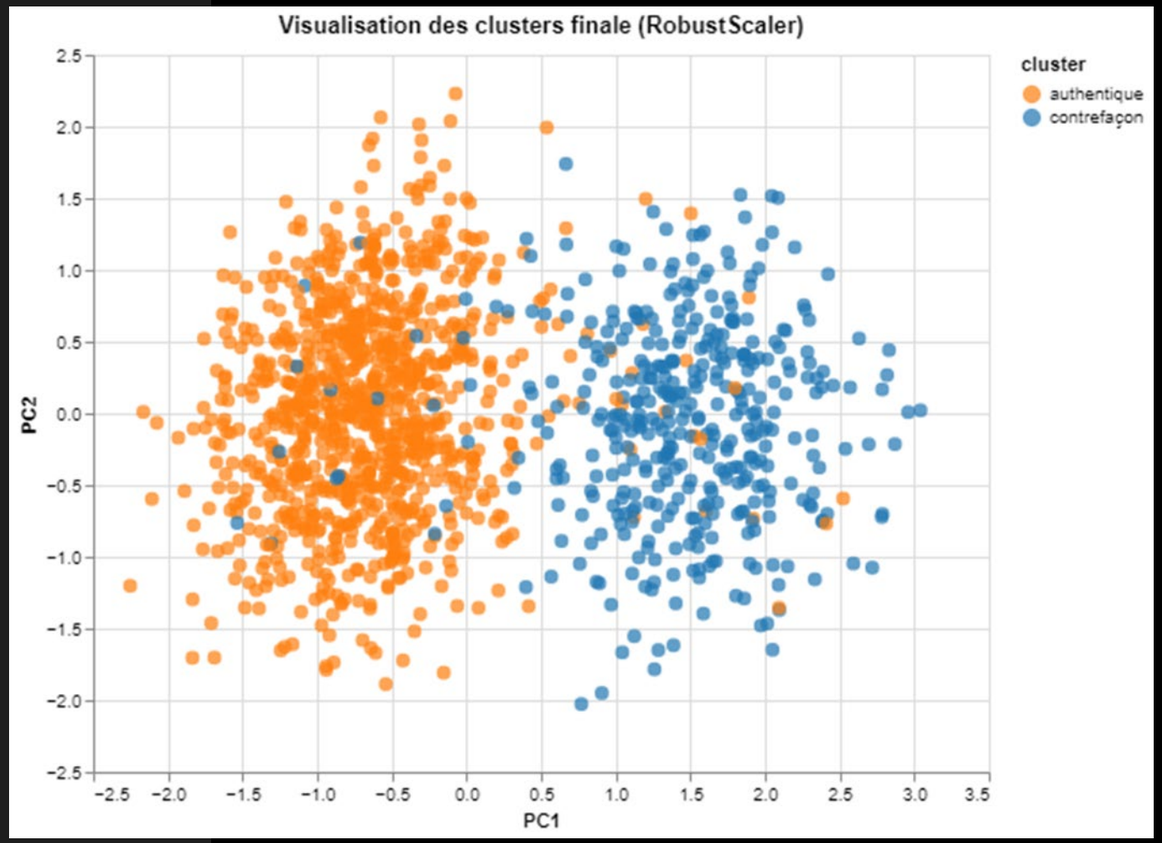
- 🔪 **Standardisation RobustScaler :**
  - **Utilisation médiane/IQR vs moyenne/écart-type**
  - **Robustesse aux valeurs extrêmes**

🎯 K-means

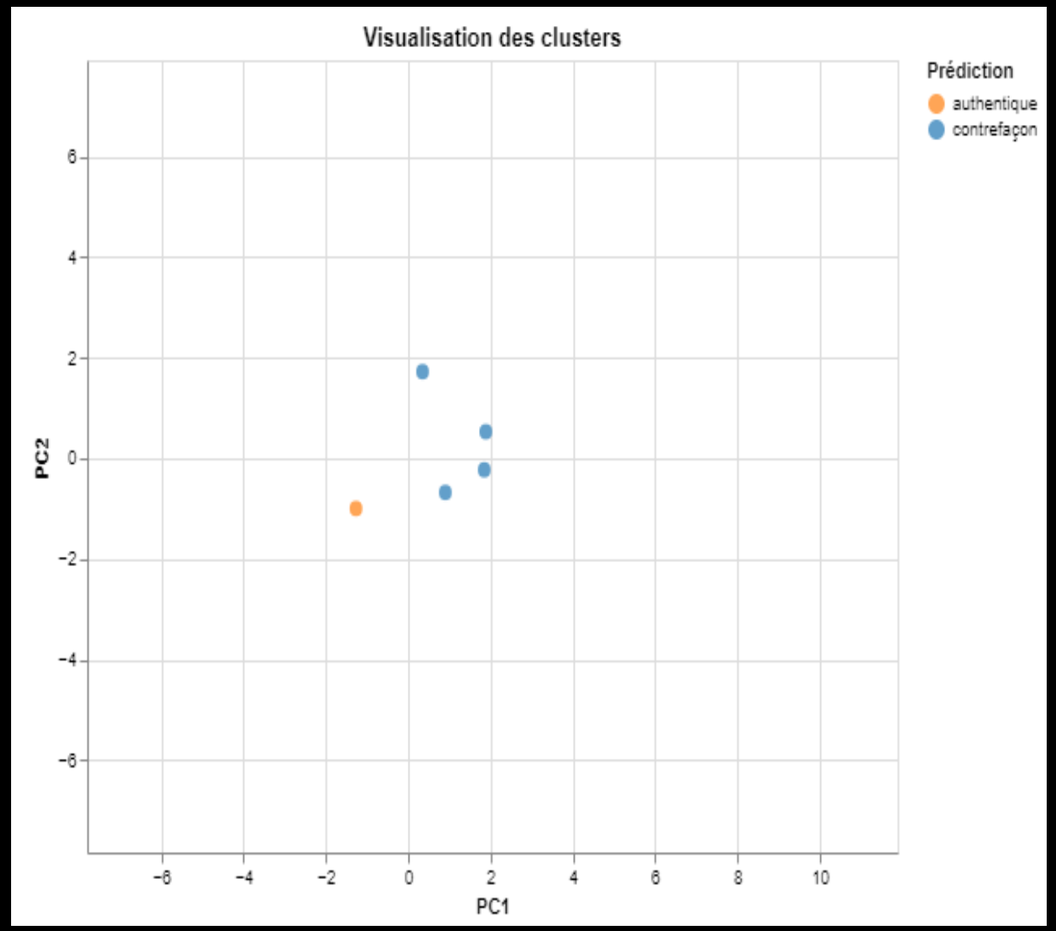

Évolution des métriques de performance
Évolution de l'Accuracy
Évolution de la Precision
Évolution du Recall
Évolution du F1-Score
Accuracy
Precision
Recall
F1-Score
1.000
0.995
0.990
0.985
0.980
0.975
0.970
RobustScaler
StandardScaler
Train avec outliers
Train sans outliers
Test

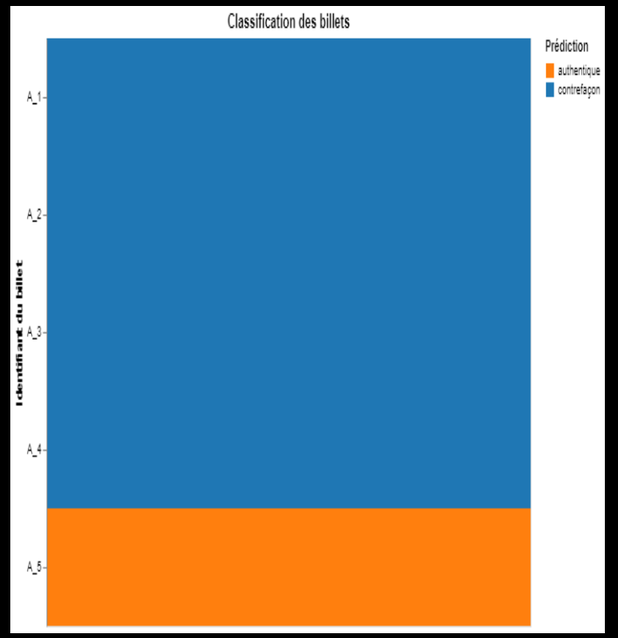
Caractéristiques des données

```
Composition finale des clusters:
cluster_robust  is_genuine
0               0            0.971175
                1            0.028825
1               1            0.986444
                0            0.013556
Name: proportion, dtype: float64

Corrélations entre les variables et les composantes principales:
                PC1       PC2       PC3       PC4       PC5       PC6
diagonal    -0.081490  0.929158 -0.299911 -0.170319 -0.104931 -0.007673
height_left  0.323019  0.340332  0.871096  0.132090  0.057396  0.016798
height_right 0.396923  0.101107 -0.324579  0.837797  0.151615  0.044578
margin_low   0.563487 -0.089958 -0.110963 -0.216992 -0.601602  0.503058
margin_up    0.419580  0.001968 -0.149337 -0.407272  0.762400  0.233501
length      -0.487756  0.050198  0.106206  0.196681  0.139749  0.830711

Modèles et configuration sauvegardés avec succès dans le dossier 'kmeans':
- Kmeans_production.pkl
- ACP_production.pkl
- Robust_scaler_production.pkl
- model_config.pkl

Contenu de la configuration:
Nombre de composantes: 2
Nombre de clusters: 2
Noms des clusters: {0: 'contrefaçon', 1: 'authentique'}
Colonnes utilisées: ['diagonal', 'height_left', 'height_right', 'margin_low', 'margin_up', 'length']
```



Caractéristiques des données





=== RÉSULTATS DÉTAILLÉS ===

ID	Prédiction
A_1	contrefaçon
A_2	contrefaçon
A_3	contrefaçon
A_4	contrefaçon
A_5	authentique

Phase 1: entraînement et optimisation des modèles

📊 **Régression logistique** 🔍 **KNN** 🌳 **Random Forest**

Processus d'Optimisation Double :

- 🔍 **RandomizedSearch :**
  - **Exploration large de l'espace**
  - **Évitement des minima locaux**
  - **Échantillonnage probabiliste efficace**
- 🎯 **GridSearch :**
  - **Affinage autour des meilleurs paramètres**
  - **Recherche exhaustive locale**
  - **Optimisation fine finale**

Entraînement du modèle logistic_regression
Fitting 5 folds for each of 2 candidates, totalling 10 fits

Matrice de confusion:
[[97 2]
 [2 192]]

Rapport de classification:

	precision	recall	f1-score	support
0	0.98	0.98	0.98	99
1	0.99	0.99	0.99	194
accuracy			0.99	293
macro avg	0.98	0.98	0.98	293
weighted avg	0.99	0.99	0.99	293

Entraînement du modèle random_forest
Fitting 5 folds for each of 8 candidates, totalling 40 fits

Matrice de confusion:
[[97 2]
 [2 192]]

Rapport de classification:

	precision	recall	f1-score	support
0	0.98	0.98	0.98	99
1	0.99	0.99	0.99	194
accuracy			0.99	293
macro avg	0.98	0.98	0.98	293
weighted avg	0.99	0.99	0.99	293

Importance des features:

	feature	importance
5	length	0.383003
8	diagonal_length_ratio	0.270626
3	margin_low	0.207760
4	margin_up	0.066927
7	margin_ratio	0.030352
2	height_right	0.020050
1	height_left	0.012085
0	diagonal	0.005023
6	height_ratio	0.004175

Entraînement du modèle knn
Fitting 5 folds for each of 12 candidates, totalling 60 fits

Matrice de confusion:
[[95 4]
 [0 194]]

Rapport de classification:

	precision	recall	f1-score	support
0	1.00	0.96	0.98	99
1	0.98	1.00	0.99	194
accuracy			0.99	293
macro avg	0.99	0.98	0.98	293
weighted avg	0.99	0.99	0.99	293