

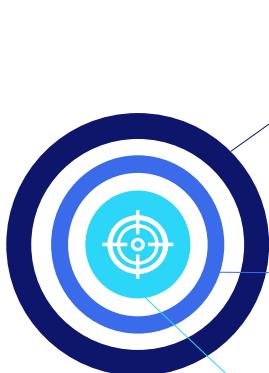
# Modèles de Machine Learning **POUR LA DETECTION DE FAUX BILLETS**

Par Aurélie RIVIERE



# Rappel du contexte & Objectifs

Mission pour l'**ONCFM** : Organisation Nationale de Lutte contre le faux monnayage  
→ Développer un modèle capable de prédire si un billet est authentique à partir de mesures géométriques.



## Approche via les modèles de Machine Learning

Identifier des relations complexes entre variables

## Test de plusieurs modèles

Régression Logistique  
Kmeans (non supervisé)  
KNN  
Random Forest

## Identifier le modèle le plus performant

Et mettre en place une application basée sur ce modèle.



# Présentation des données

## Analyses exploratoires



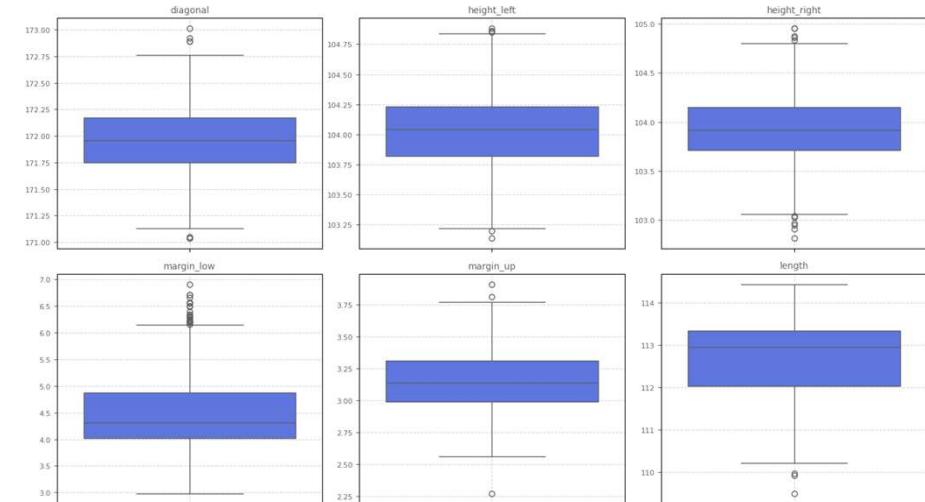
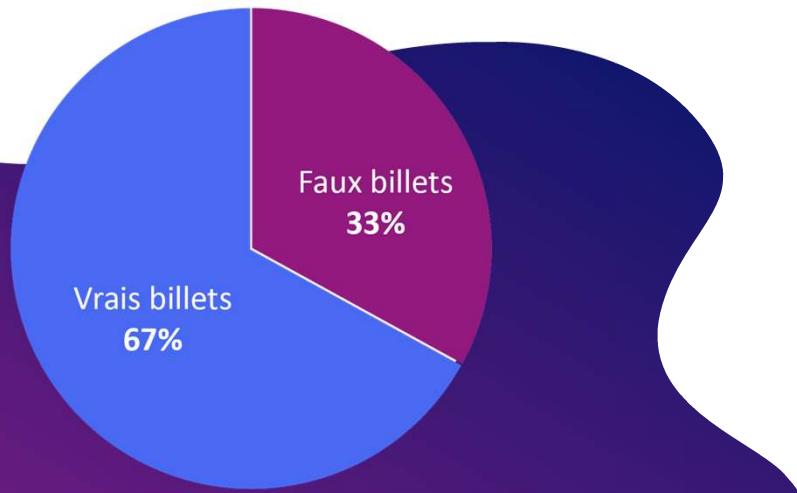
### Jeu de données

1500 billets  
1000 Vrai et 500 faux



### Variables

6 variables géométriques  
1 variable booléenne (VRAI/FAUX)



### Distribution des variables

Distribution globalement homogène  
Quelques outliers, mais cohérentes avec la variabilité des mesures



### Identification de valeurs manquantes

37 valeurs manquantes sur margin\_low

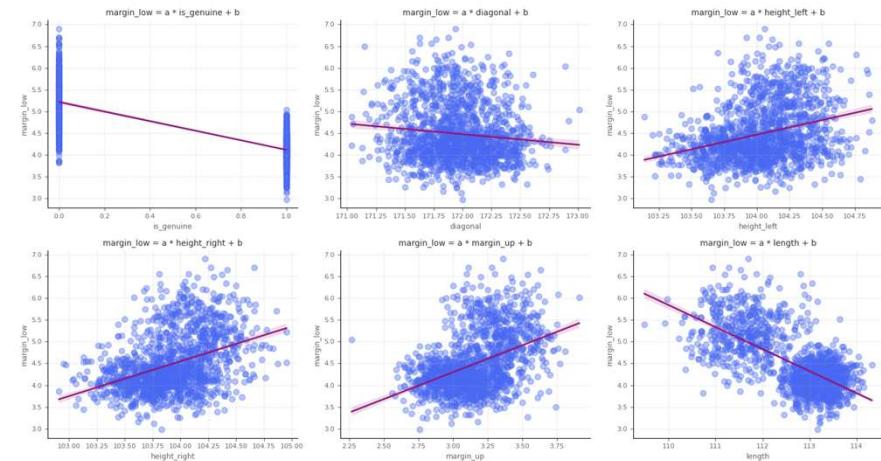
# Gestion des valeurs manquantes par Régression Linéaire

## Observations des correlations entre les variables

Relations essentiellement linéaire entre margin\_low et les autres variables  
Forte multicolinéarité – Indice VIF élevé

## Réduction et sélection des variables prédictives

'is\_genuine' (vrai ou faux billet), 'height\_left' et 'margin\_up'.

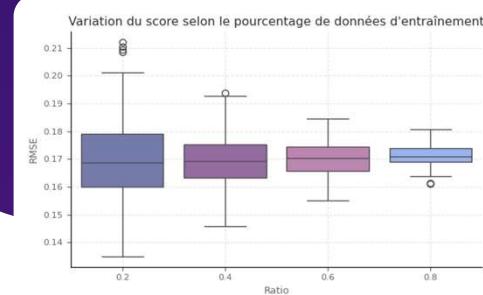


Entrainement du modèle  
Split 80% train / 20% test  
Random seed 42

Performance du modèle  
Score R<sup>2</sup> de 0,67

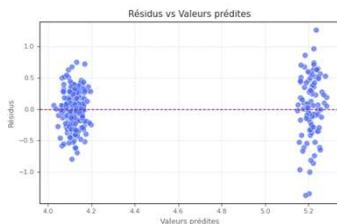
Amélioration du modèle  
Ajout d'un terme croisé (non significatif)  
Influence du Random Seed (positif)

Modèle ajusté  
Score R<sup>2</sup> de 0,682  
MAPE de 6%  
RMSE de 0,13



# Vérification hypothèses regression linéaire

L'analyse des résidus, *différence entre valeurs observées et estimées*, permet de vérifier les hypothèses nécessaires à la régression linéaire.

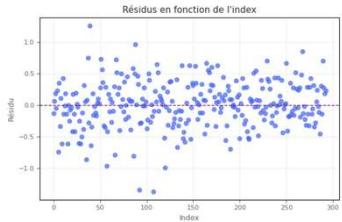


## Linéarité des relations et Homoscédasticité

### Graphique de répartition des résidus en fonction des valeurs prédictes

Dispersion homogène, pas de cone.  
Deux noeuds, dû à la répartition vrai/faux billets

→ Hypothèse de linéarité et d'homoscédasticité confirmée



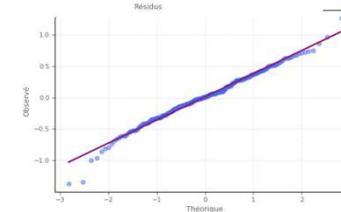
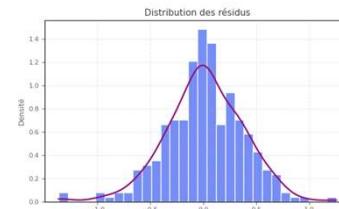
## Indépendance des erreurs

### Graphique de répartition des résidus en fonction de l'index

Absence de motif

→ Hypothèse d'indépendance des erreurs confirmée

## Normalité des résidus



### Histogramme des résidus avec répartition des erreurs

Forme en cloche, proche courbe normale

### Q-Qplot

Suit également la forme normale mais variation aux extrémités

### Vérification de l'hypothèse de normalité avec un test de Shapiro-Wilk

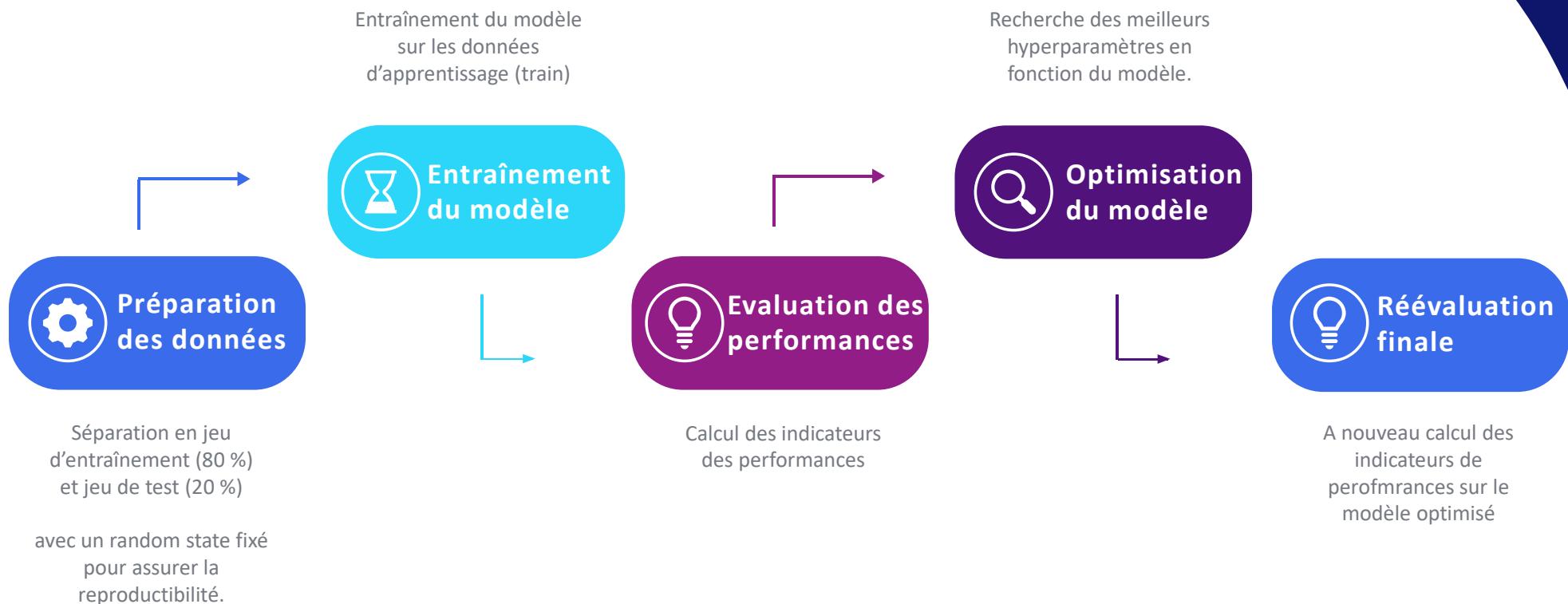
P-value > 0,05

→ Hypothèse de normalité rejetée

Toutes nos hypothèses ne sont pas validées.  
Mais elles sont suffisamment proches d'un modèle normal pour assurer robustesse au modèle dans le cadre d'une utilisation predictive.

→ Prédiction des valeurs manquantes  
à l'aide de notre modèle de régression linéaire

# Méthodologie de modélisation



# Indices de performances des modèles

Indicateurs de performance d'une classification binaire.

## Matrice de confusion

		Observations	
		Vrais faux billets (TN) faux billets prédit faux	Faux vrais billets (FP) faux billets prédit vrai
Prédictions	Vrais faux billets (FN) vrai billets prédit faux	Vrais vrai billets (TP) vrai billets prédit vrai	
	Vrais faux billets (TN) faux billets prédit faux	Faux vrais billets (FP) faux billets prédit vrai	

\*matrice de confusion de sklearn

## Accuracy (score de pertinence)

Pourcentage de prédictions correctes, toutes classes confondues.

## Recall (rappel)

Capacité du modèle à détecter les vrais positifs parmi tous les vrais.  
→ « Parmi les vrais billets, combien ont été bien détectés ? »

## Précision

Capacité du modèle à ne prédire positif que lorsqu'il a raison.  
→ « Parmi les billets prédits comme vrais, combien le sont vraiment ? »

## Courbe ROC et AUC (Area Under the Curve)

Courbe qui montre la capacité du modèle à distinguer les classes, en faisant varier le seuil de décision.  
Aire sous la courbe ROC. Elle mesure la probabilité que le modèle classe un vrai billet au-dessus d'un faux.

# Sélection des variables

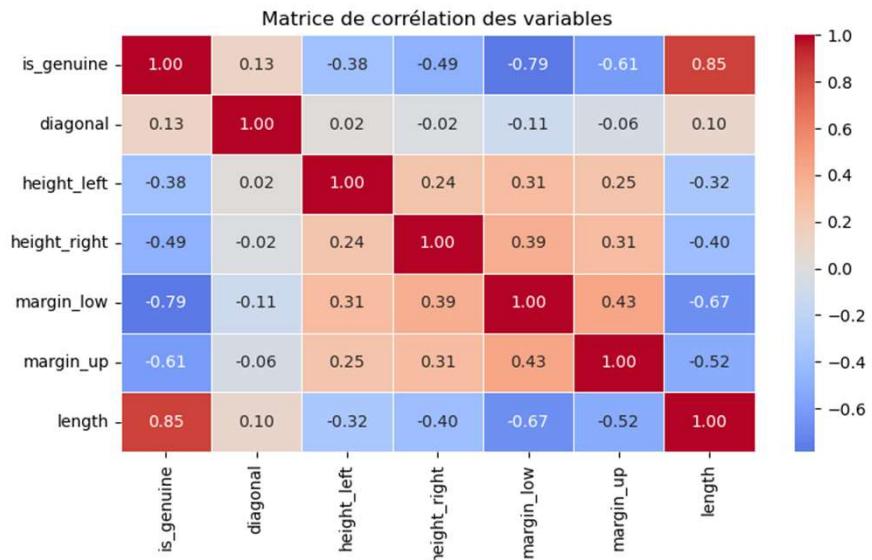
Sélection des variables prédictives les plus pertinentes pour déterminer la variable cible "is\_genuine"

## Corrélation

- Positive forte avec length
- Négative forte avec margin\_low
- Négative forte avec margin\_up
- Très faible corrélation positive avec diagonal
- Faible corrélation négative avec height\_left et height\_right

## Multicolinéarité

- Forte multicolinéarité
- Calcul de l'indice VIF
- Suppression itérative des variables avec le VIF le plus élevé



## Variables prédictives

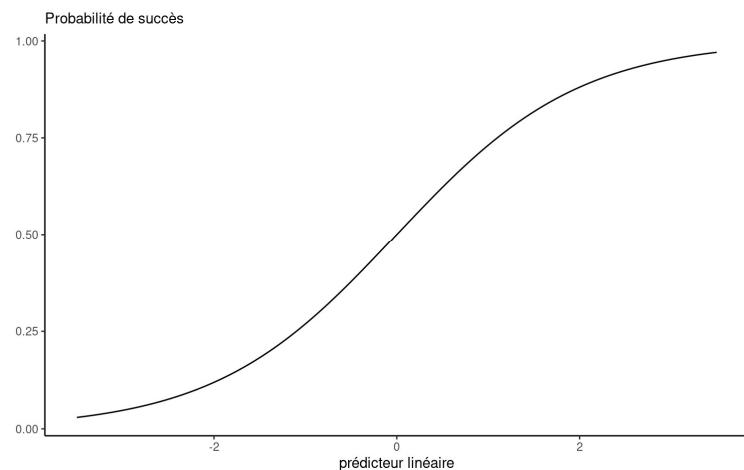
Sélection de 3 variables prédictives :  
'length' - 'margin\_low' - 'margin\_up'

Ces variables permettent de conserver l'essentiel de l'information, tout en limitant la redondance.

→ conditions équitables pour tous les algorithmes testés

# Modèle de Régression Logistique

Modèle de classification binaire de référence



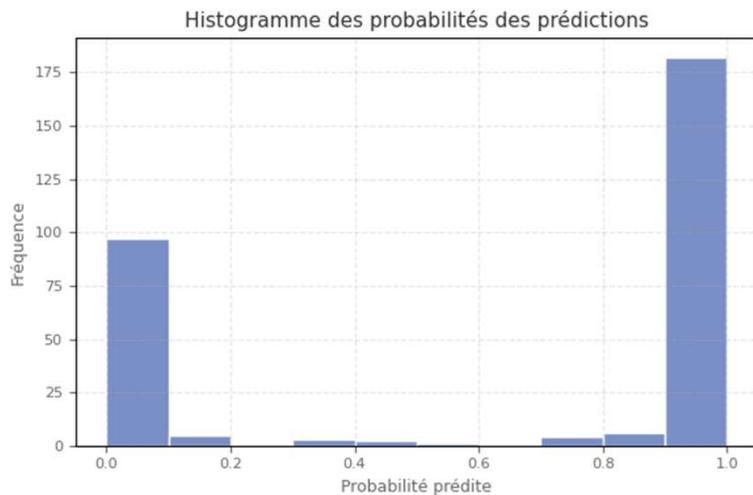
- Modèle statistique simple et robuste
- Estime la probabilité qu'un billet soit authentique ou non
- Particulièrement adapté à la classification binaire (vrai/faux)
- Benchmark initial pour évaluer nos autres modèles
- Interprétable et facile à ajuster via un seuil de décision

# Résultats Régression Logistique



## Histogramme des probabilités

- Forte concentration autour de 0 et 1
- Le modèle distingue nettement les deux classes.



## Evaluation du modèle

- Accuracy : 99%
- Matrice de Confusion [[107 3] [ 0 190]].



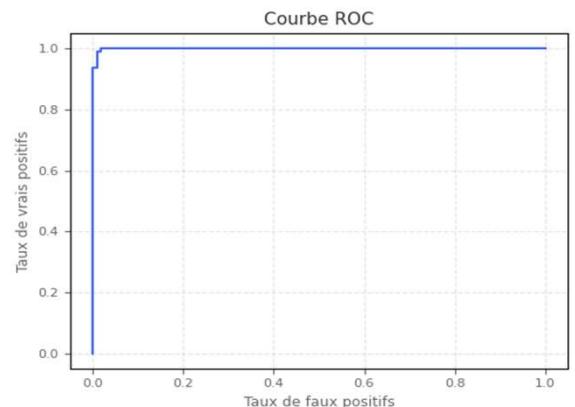
## Seuil de séparation des classes

- Test de différents seuils de séparation des classes
- Meilleurs identification des classes à Seuil 0,7



## Performances du modèle

Accuracy : 0.993  
Précision : 0.990  
Rappel : 1.000  
AUC-ROC : 0.999  
F1-Score : 0.995  
Matrice de confusion  
[[108 2] [ 0 190]]



## Courbe ROC

- Excellente capacité de discrimination du modèle

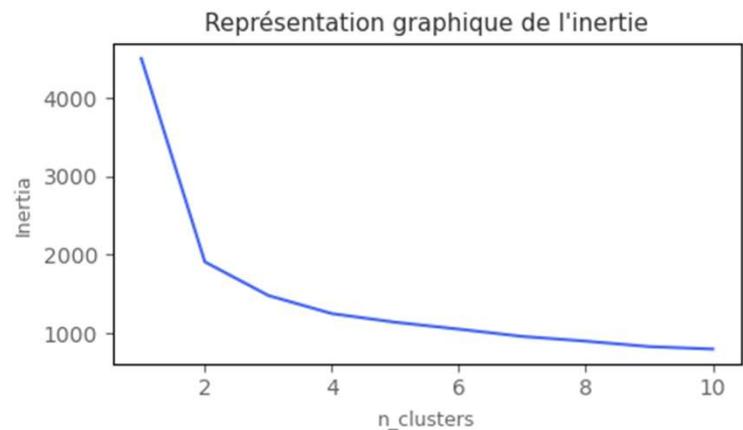
→ Un modèle simple, performant et interprétable, idéal comme point de référence pour évaluer les autres approches

# Segmentation Kmeans

Une approche non supervisée

- Algorithme de clustering non supervisé
- Regroupe les données selon leurs similarités.
- Ne nécessite aucune étiquette pour fonctionner.
- Données standardisées pour garantir une pondération équitable des variables.
- Entraînement sur l'ensemble du dataset (pas de split train/test).

→ Objectif : observer la capacité du K-means à recréer cette séparation naturellement

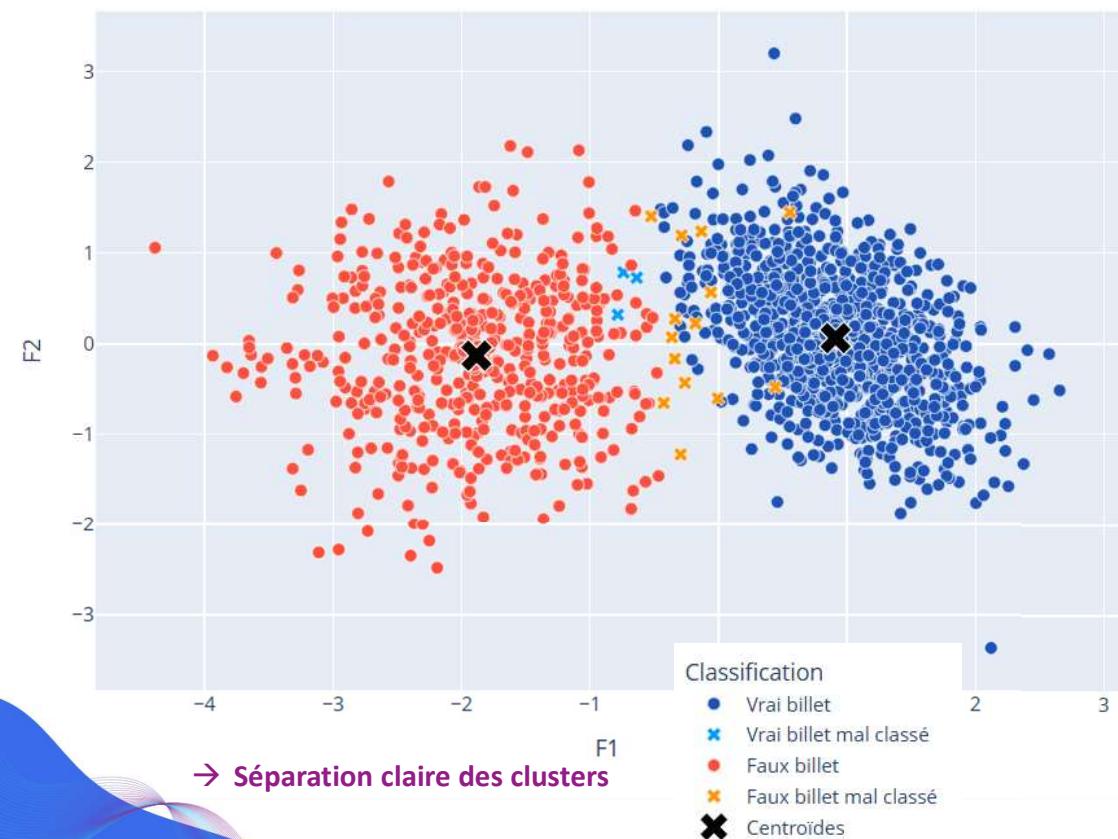


➤ Validation du nombre de clusters avec la méthode du coude :

- Confirme la présence de 2 groupes naturels  
→ Vrais / Faux billets.

# Projection des centroides

## Segmentation Kmeans



### ➤ Analyse en composantes principales (ACP)

- Réduction de dimensionnalité pour projection en deux dimensions

### ➤ Projection des centroides

→ Séparation nette entre vrais et faux billets  
→ Centroïdes bien positionnés, peu d'erreurs de classification

### ● Performances du modèle

→ Silhouette Score : 0,531 (bonne compacité des clusters)

Comparaison avec les vraies étiquettes. Adaptée pour permettre la comparaison avec les autres modèles

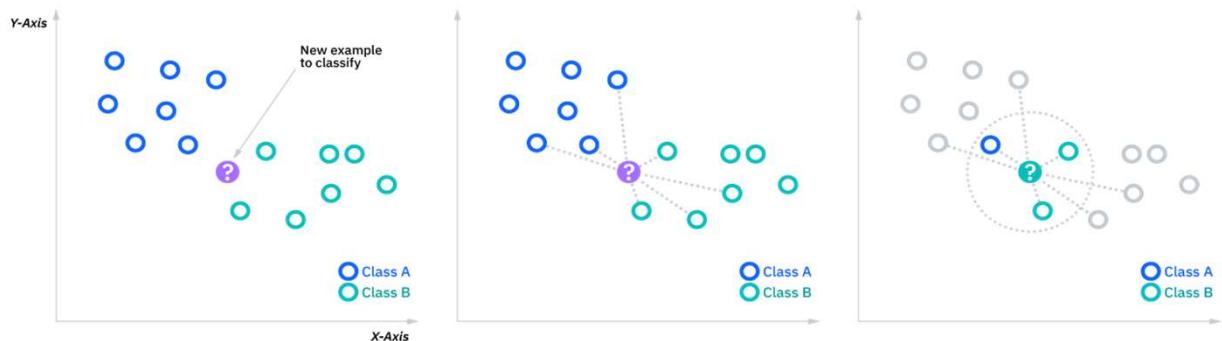
→ Accuracy : 98,9 %

→ Matrice de confusion : [[107 3] [0 190]]

# KNN K Nearest Neighbors

## Algorithme supervisé de classification

→ Prédit la classe d'un billet en observant les k billets les plus proches dans l'espace des variables

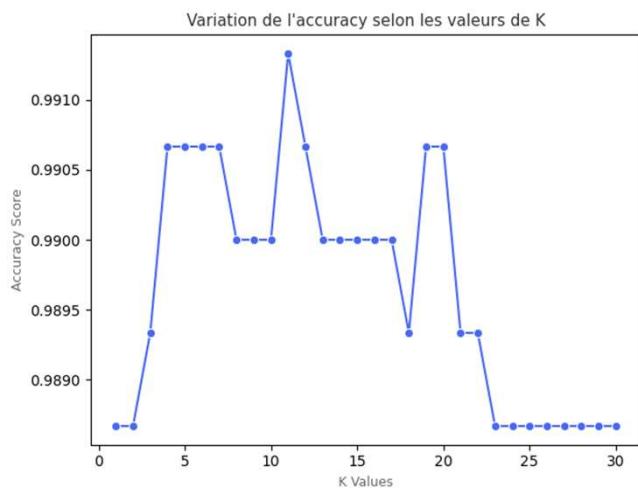


- Basé sur la proximité entre les individus
- Classe attribuée selon la majorité des voisins les plus proches
- Données standardisées (le modèle est sensible aux échelles)

# Optimisation et Résultats modèle KNN

## ➤ Optimisation de K (nombre de voisins)

- Test de variation de l'accuracy selon les valeurs de k
  - Projection des accuracy sur un graphique
- Meilleur score obtenu pour k = 11



## Performances du modèle

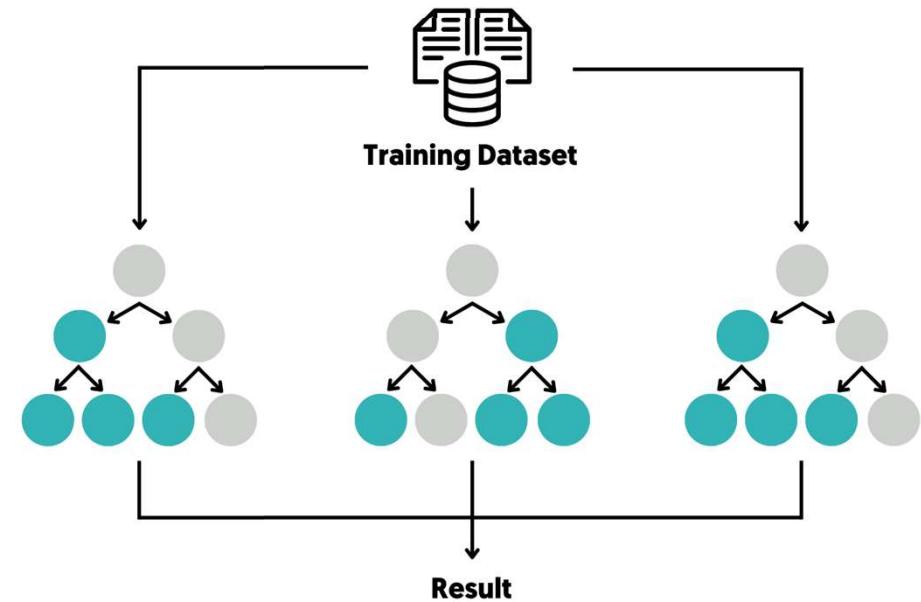
- Accuracy : 0.993
- Precision : 0.990
- Recall : 1.000
- F1 Score : 0.995
- AUC : 1.000

→ Matrice de confusion : [[109 1] [ 0 190]]  
→ 1 faux positif, aucun faux négatif

→ modèle simple mais efficace, qui combine précision, rappel parfait et une excellente robustesse face aux erreurs critiques.

# Modèle Random Forest

- Modèle d'ensemble basé sur des arbres de décision
- Chaque arbre s'entraîne sur un sous-échantillon aléatoire (données et variables)
- Entrainement du modèle avec les paramètres par défaut
  - Permet d'établir une base de référence
- Premier résultats excellents
  - variables bien discriminantes
  - Le modèle capture efficacement la structure des données



# Optimisation et Résultats Random Forest

## ➤ Test de sensibilité au nombre d'arbres et à leur profondeur

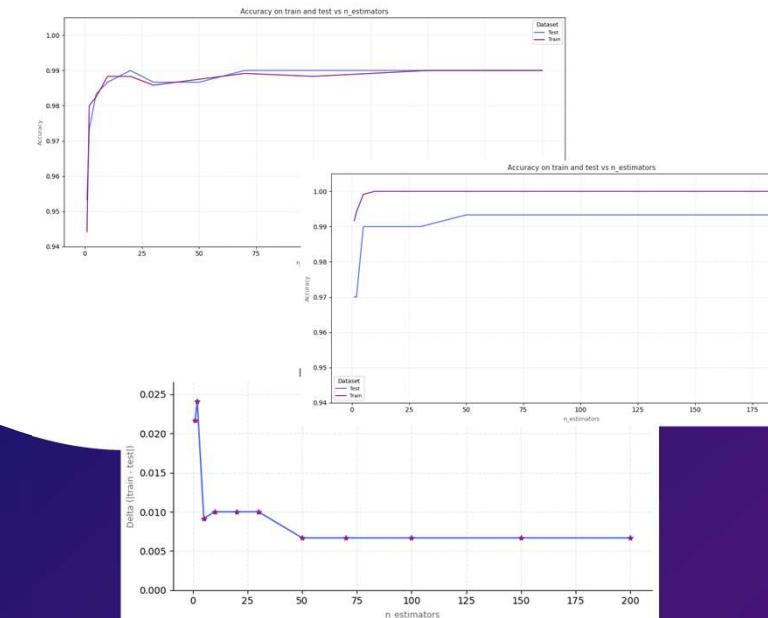
- ▶ Score rapidement stable

## ➤ Optimisation via GridSearchCV

- ▶ Validation croisée sur 60 combinaisons
- ▶ Meilleurs paramètres : 150 arbres, profondeur libre, `max_features='sqrt'`

## Performance du modèle

- Accuracy : 0.993
- Precision : 0.990
- Recall : 1.000
- F1 Score : 0.995
- Matrice de confusion : [[108 2] [ 0 190]]



→ Excellente performance, optimisée avec le GridSearchCV. Mais légère tendance au faux positif.

# Comparaison Des Modèles

	Accuracy	Précision	Rappel	F1-Score	AUC-ROC	Silhouette Score	Matrice de confusion
Régression Logistique	0,993	0,99	1,0	0,995	0,999	-	[108 2] [0 190]
K-Means	0,989	-	-	-	-	0,531	[107 3] [0 190]
KNN	0,997	0,995	1,0	0,997	0,995	-	[109 1] [0 190]
Random Forest	0,993	0,99	1,0	0,995	1,0	-	[108 2] [0 190]

# Sélection **Meilleur Modèle**

---

**Critères de sélection :**

- Meilleurs scores de performance globale
- Minimisation des erreurs critiques (faux positifs)
- Pertinence dans le contexte métier

**Modèle retenu :**

**KNN (K-Nearest Neighbors)**

# Application du modèle sur les données de production

Test et adaptation du modèle sur un ensemble de données de production.

	diagonal	height_left	height_right	margin_low	margin_up	length	id	is_genuine
0	171.76	104.01	103.54	5.21	3.30	111.42	A_1	0
1	171.87	104.17	104.13	6.00	3.31	112.09	A_2	0
2	172.00	104.58	104.29	4.99	3.39	111.57	A_3	0
3	172.49	104.55	104.34	4.44	3.03	113.20	A_4	1
4	171.65	103.63	103.56	3.77	3.16	113.33	A_5	1

## Exemple de fichier de production

Fichier type production fournit pour analyser la structure des données

## Sélection des variables prédictives

Sélection des variables utiles et standardization des données en vue de l'utilisation du modèle

## Prédictions du modèle

Prédictions sur les données de production

## Exportation du modèle

Le modèle et le scaler ont été exporté pour utilisation dans une application

# Application De Détection de faux billets