Détection de faux billets

Pascal Brochart – Octobre 2024

Contexte

L'ONCFM veut lutter contre le faux-monnayage avec des méthodes d'identification de faux billets

- **Il est convenu de mettre à disposition une application de machine Learning**
- **Analyser avec différents algorithmes et présenter les résultats**
- **Enrichir le modèle d'entrainement avec des scans de billets**
- **Faire de la prédiction de vrais ou faux billets avec un programme de détection autonome**

Analyse exploratoire

Pas de doublons présents dans le jeu de données

Des valeurs manquent dans la colonne marge basse

```
df_billets.info() # 37 valeurs manquantes dans la colonne margin_low
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1500 entries, 0 to 1499
Data columns (total 7 columns):
 #   Column        Non-Null Count  Dtype  
---  ------        --------------  -----  
 0   is_genuine    1500 non-null   bool   
 1   diagonal      1500 non-null   float64
 2   height_left   1500 non-null   float64
 3   height_right  1500 non-null   float64
 4   margin_low    1463 non-null   float64
 5   margin_up     1500 non-null   float64
 6   length        1500 non-null   float64
dtypes: bool(1), float64(6)
memory usage: 71.9 KB
```

```
print(len(df_billets) - len(df_billets.drop_duplicates()), 'doublon') # Aucun doublon
```

```
0 doublon
```

Régression linéaire

Nous utilisons la régression linéaire pour prédire les valeurs manquantes avec toutes les colonnes

Et répétons la séquence jusqu'à ce qu'il ne reste que les données nécessaires

Seules margin_up et is_genuine seront utilisées pour le modèle de prédiction

Le score R2 de 0,617 indique la qualité du modèle

```
reg_multi = smf.ols('margin_low~is_genuine+diagonal+height_left+height_right+margin_up+length
print(reg_multi.summary())
```

```
                            OLS Regression Results
==============================================================================
Dep. Variable:             margin_low   R-squared:                       0.617
Model:                            OLS   Adj. R-squared:                  0.615
Method:                 Least Squares   F-statistic:                     390.7
Date:                Sun, 06 Oct 2024   Prob (F-statistic):          4.75e-299
Time:                        16:56:07   Log-Likelihood:                -774.14
No. Observations:                1463   AIC:                             1562.
Df Residuals:                    1456   BIC:                             1599.
Df Model:                           6
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
Intercept      2.8668      8.316      0.345      0.730     -13.445      19.179
is_genuine[T.True]    -1.1406      0.050    -23.028      0.000      -1.238      -1.043
diagonal      -0.0130      0.036     -0.364      0.716      -0.083       0.057
height_left    0.0283      0.039      0.727      0.468      -0.048       0.105
height_right    0.0267      0.038      0.701      0.484      -0.048       0.102
margin_up     -0.2128      0.059     -3.621      0.000      -0.328      -0.098
length        -0.0039      0.023     -0.166      0.868      -0.050       0.042
==============================================================================
Omnibus:                       21.975   Durbin-Watson:                   2.038
Prob(Omnibus):                  0.000   Jarque-Bera (JB):               37.993
Skew:                           0.061   Prob(JB):                     5.62e-09
Kurtosis:                       3.780   Cond. No.                     1.95e+05
==============================================================================
```

Tests statistiques

Nous effectuons divers tests statistiques pour s'assurer notamment que les variables explicatives du modèle ne mesurent pas le même phénomène et qu'elles suivent une distribution normale

Les 2 derniers tests rejettent l'hypothèse H0 mais on valide cependant le modèle car l'échantillon de test n'est pas de taille suffisante

Les 37 valeurs manquantes de marge basse peuvent être prédites

3.1 - Test de colinéarité des variables

```
# Test de colinéarité des variables
# Tous les coefficients sont inférieurs à 10, il n'y a donc pas de problème de colinéarité
variables = reg_multi.model.exog
[variance_inflation_factor(variables, i) for i in np.arange(1, variables.shape[1])]
```

```
[1.5938854494007757, 1.5938854494007741]
```

3.2 - Test de l'homoscédasticité

```
# Test de l'homoscédasticité
# La p-valeur ici est inférieure à 5%, on rejette l'hypothèse H0 selon laquelle les variances sont constantes
_, pval, __, f_pval = statsmodels.stats.diagnostic.het_breuschpagan(reg_multi.resid, variables)
print('p value test Breusch Pagan:', pval)
```

```
p value test Breusch Pagan: 3.2033559115817906e-36
```

3.3 - Test de la normalité des résidus

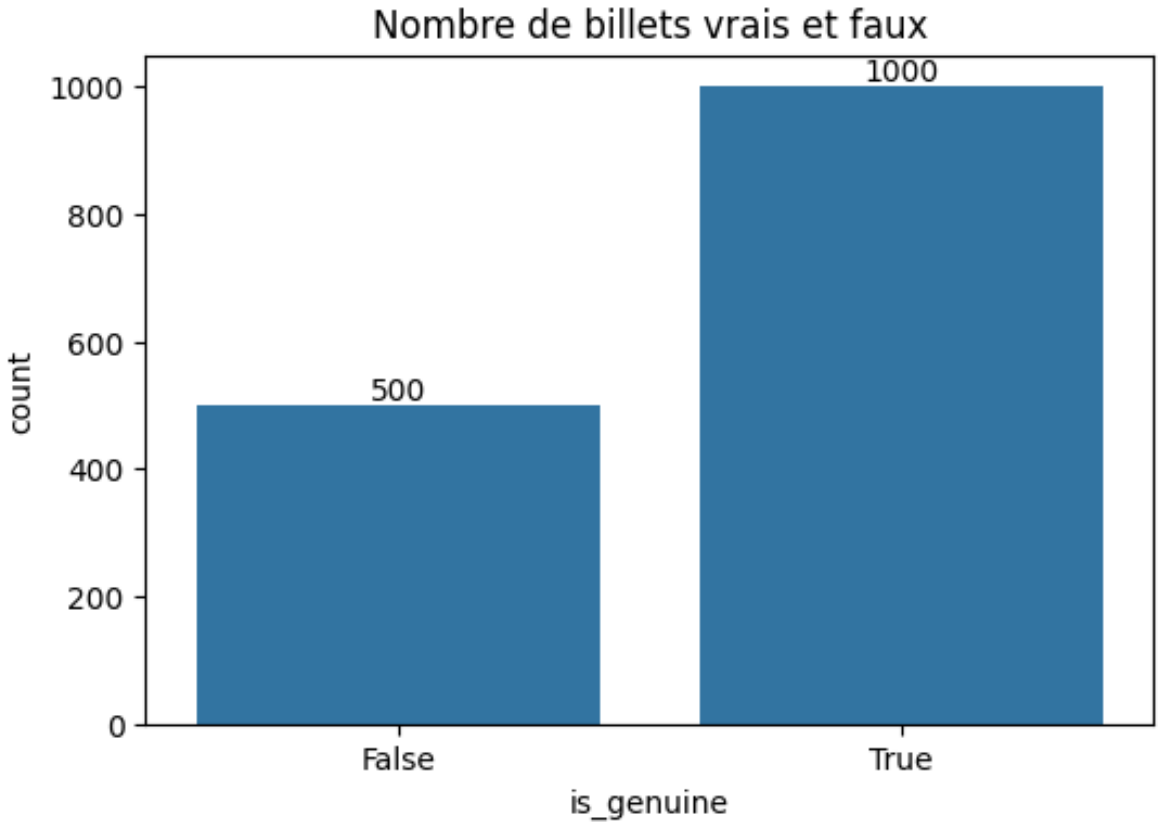
```
# Test de la normalité des résidus
# Ici, l'hypothèse de normalité est remise en cause (p_valeur inférieure à 5%)
shapiro(reg_multi.resid)
```

```
ShapiroResult(statistic=0.9936248064041138, pvalue=6.20942773821298e-06)
```

Représentation des vrais et faux billets

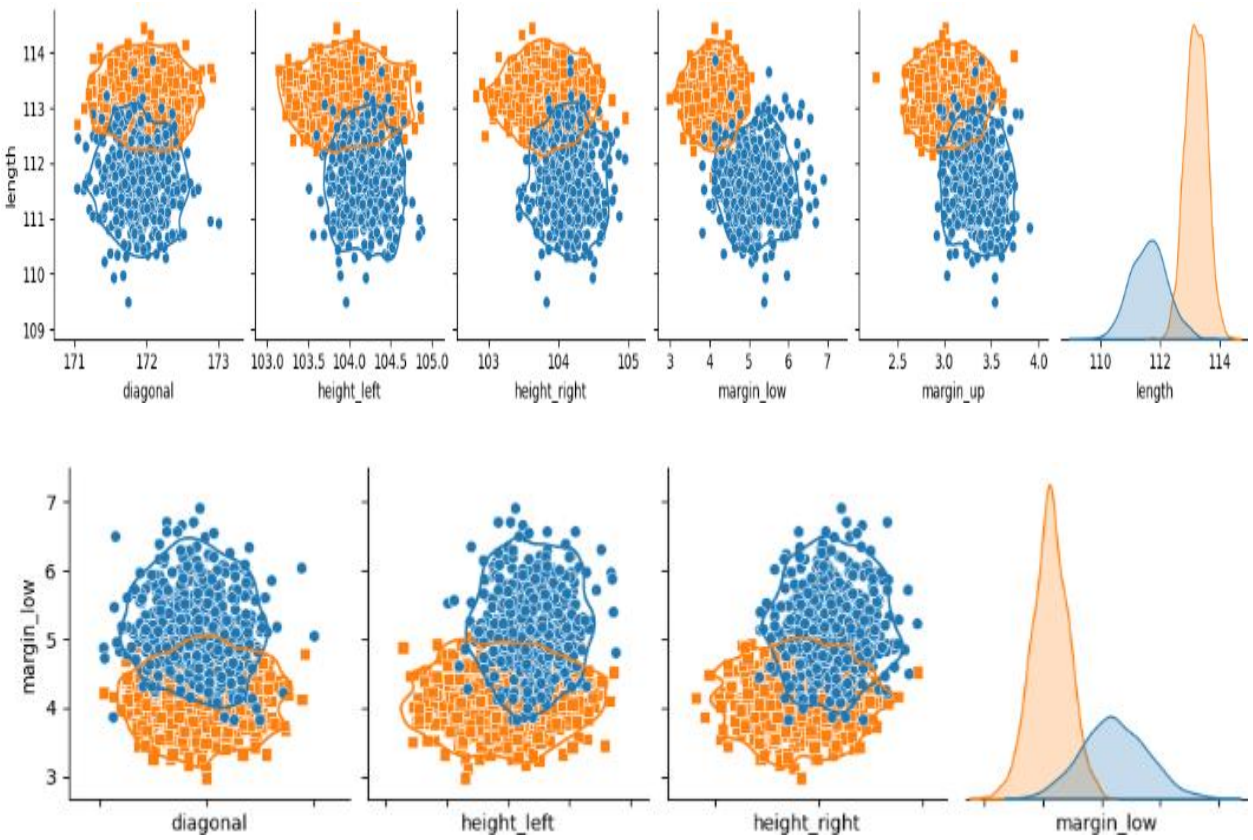
Le jeu de données se compose de:

- **1000 vrais billets**
- **500 faux billets**



Corrélation des variables

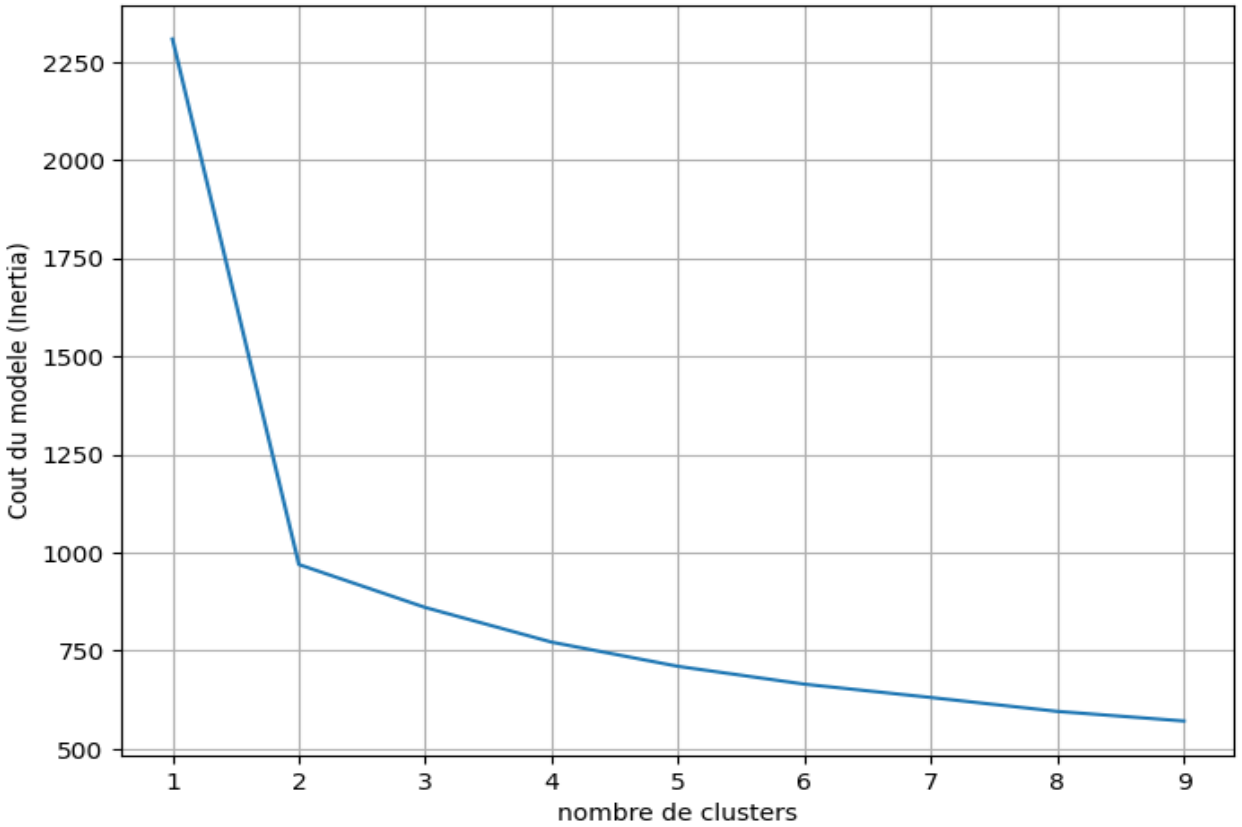
La corrélation entre chaque variables par rapport à la nature du billets permet de mettre en évidence que la longueur des vrais billets en orange est supérieure aux faux en bleus mais la marge basse est plus courte



Méthode K-Means

Avant d'exécuter l'algorithme de clustering non supervisé K-Means nous effectuons la méthode du coude pour déterminer le nombre de clusters nécessaires pour l'initialisation

On utilise le point de retournement de la courbe



Régression logistique

La régression logistique est un modèle de statistiques qui utilise la fonction logistique comme l'équation entre x (variables explicatives) et y (variable à expliquer)

Elle permet aussi de donner une probabilité avec une valeur comprise entre 0 et 1

Pour un modèle performant, les données seront séparées comme suit:

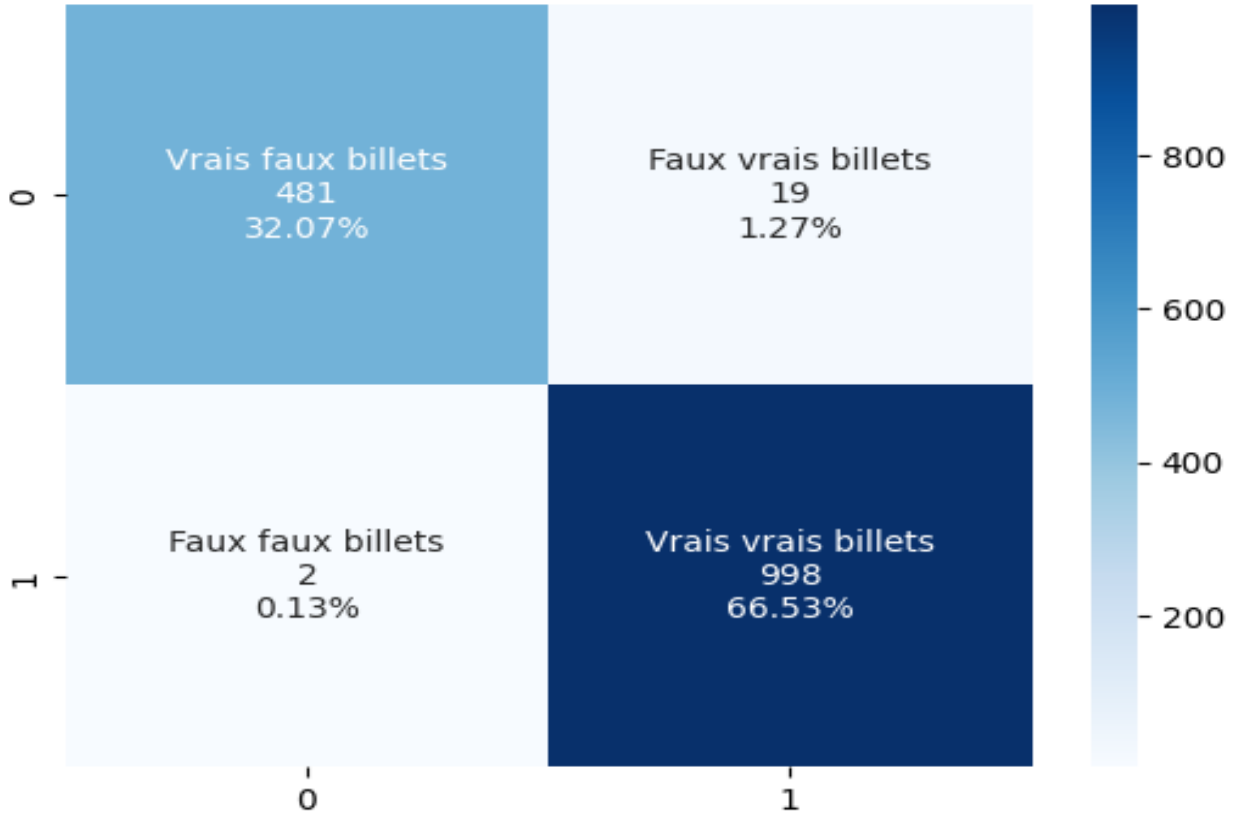
- **80% pour la partie entrainement**
- **20% pour la partie test**

Matrice de confusion

La matrice de confusion permet de mesurer la qualité d'un système de classification

Dans le cas d'un problème binaire comme ici elle ne comporte que 4 valeurs

Le pourcentage d'erreur pour le clustering K-Means par rapport à l'authenticité des billets est indiqué dans les quadrants en haut à droite et en bas à gauche



Comparaison et solution retenue

Nous calculons la métrique de performance « Accuracy » pour les deux méthodes qui consiste à donner la proportion de résultats vrais parmi le nombre total de cas examinés:

- **Méthode K-Means: 98,6%**
- **Régression logistique: 99%**

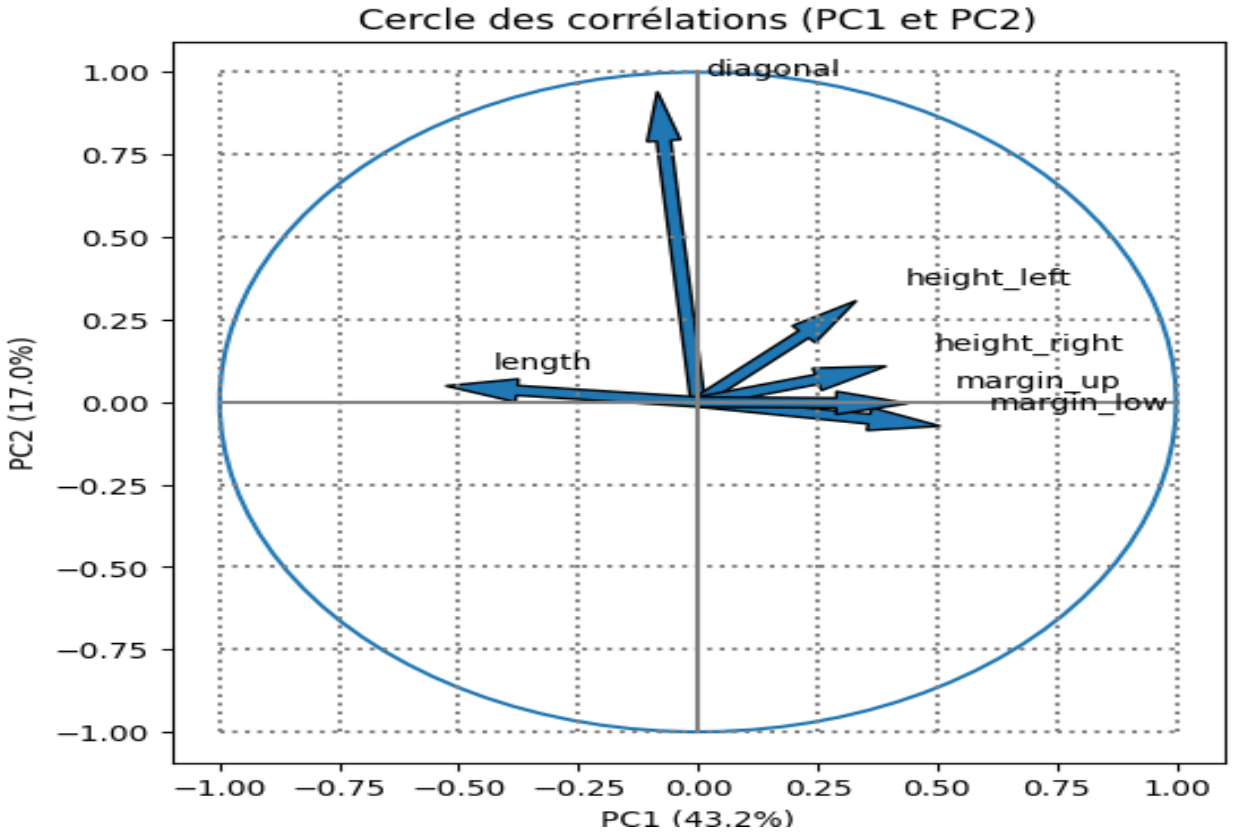
La régression logistique donne un meilleur résultat et si l'on ajoute un plus faible nombre de faux billets détectés vrais alors il apparait plus pertinent de choisir cette méthode pour le programme de détection

Analyse de la composante principale

L'ACP permet de réduire les dimensions d'un jeu de données dans un nouvel espace

En choisissant deux composantes principales on pourra visualiser les billets dans des graphiques en deux dimensions

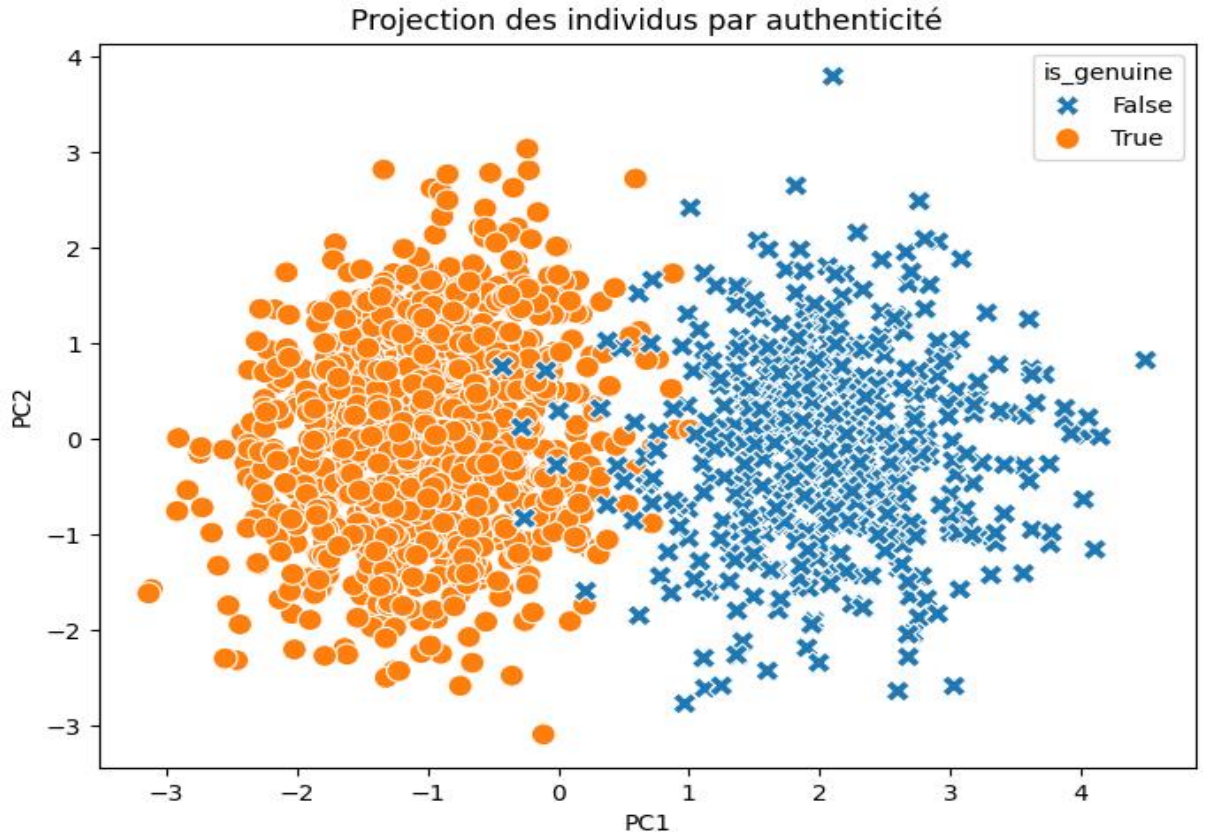
Le cercle des corrélations ci-joint représente la contribution des variables aux composantes principales



Projection des individus par authenticité

Le graphique ci-contre représente tous les billets dans un espace en deux dimensions avec les données de l'ACP

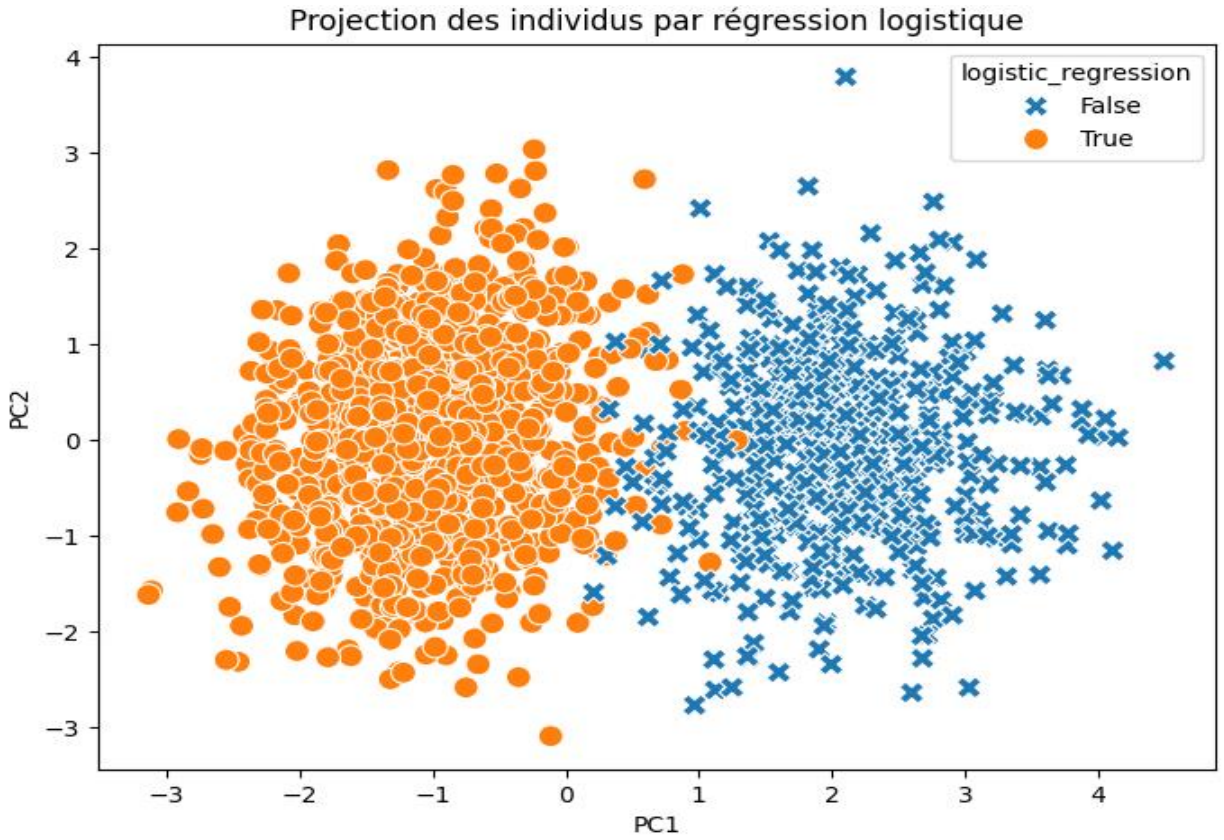
Nous observons une frontière très incertaine avec quelques faux billets en bleus dans l'espace des vrais billets en orange



Projection des individus par régression logistique

Ici la même représentation avec la méthode de la régression logistique

On note quelques erreurs de prédictions mais globalement le résultat est satisfaisant



Programme de détection avec streamlit

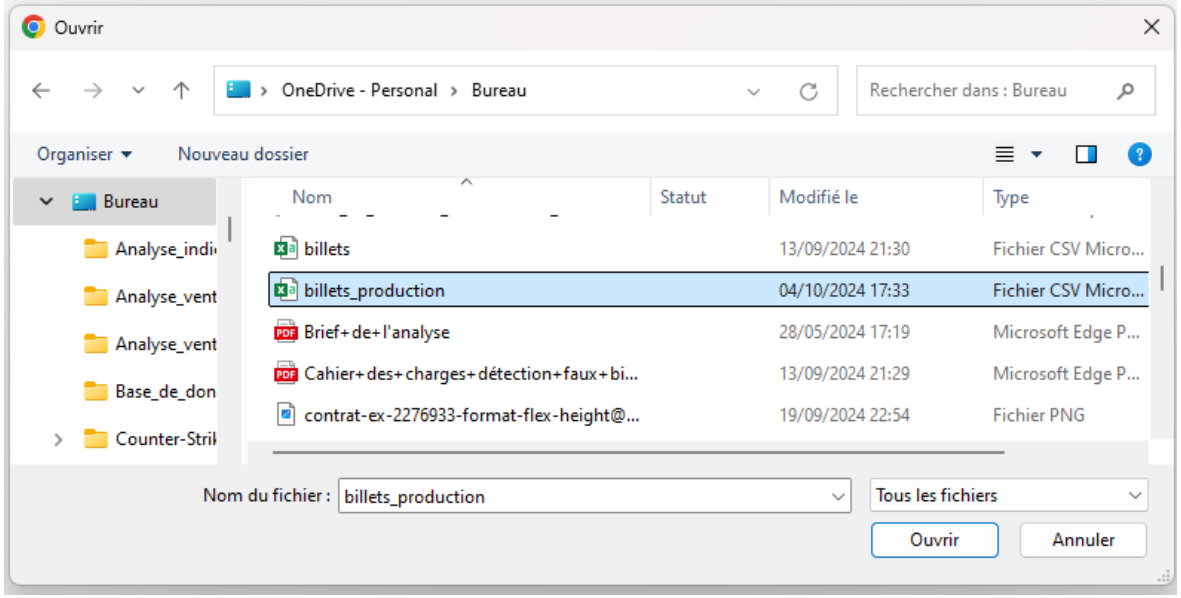
Programme de détection de faux billets

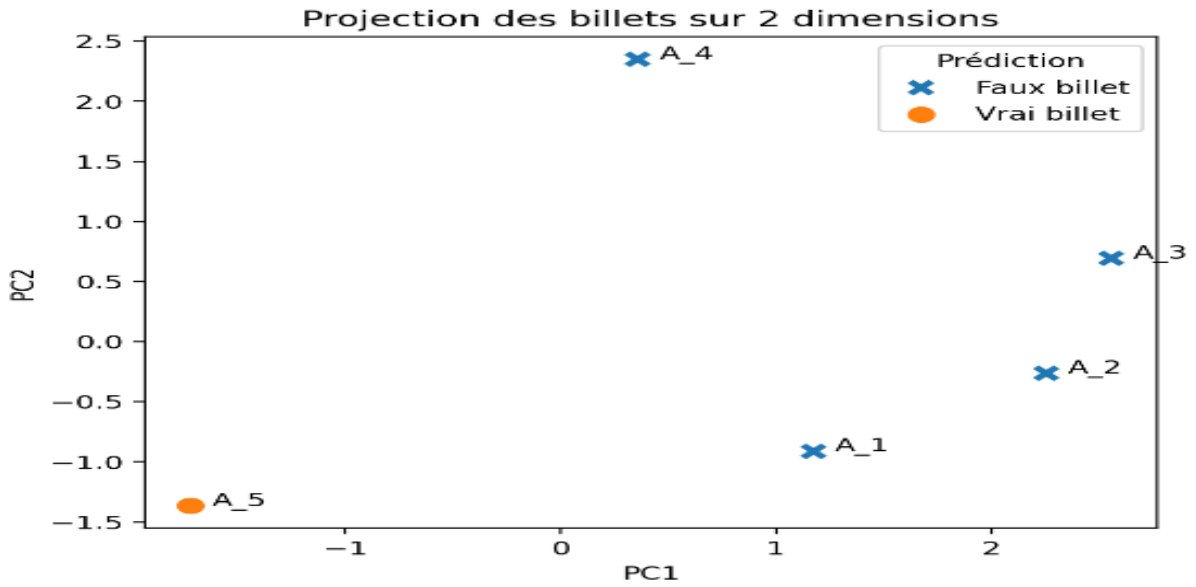
Choose a file

Drag and drop file here

Limit 200MB per file

Browse files





	Prédiction	Probabilité faux billet	Probabilité vrai billet	id
0	Faux billet	0.9998	0.0002	A_1
1	Faux billet	1	0	A_2
2	Faux billet	1	0	A_3
3	Faux billet	0.8162	0.1838	A_4
4	Vrai billet	0	1	A_5