PROJET 6

Détecter les faux billets

Création d'un algorithme de détection de faux billets.

DÉROULÉ DE LA PRÉSENTATION

Mission 0

Description des données analyses univariées et bivariées.

01.

02.

Mission 1

Analyse en composantes principales

Mission 2

algorithme de classification de type k-nn

03.

04.

Mission 3

Modélisez les données à l'aide d'une régression logistique

01.

Mission 0

Nous allons rechercher la différence entre les vrais et faux billets en utilisant l'analyse descriptive

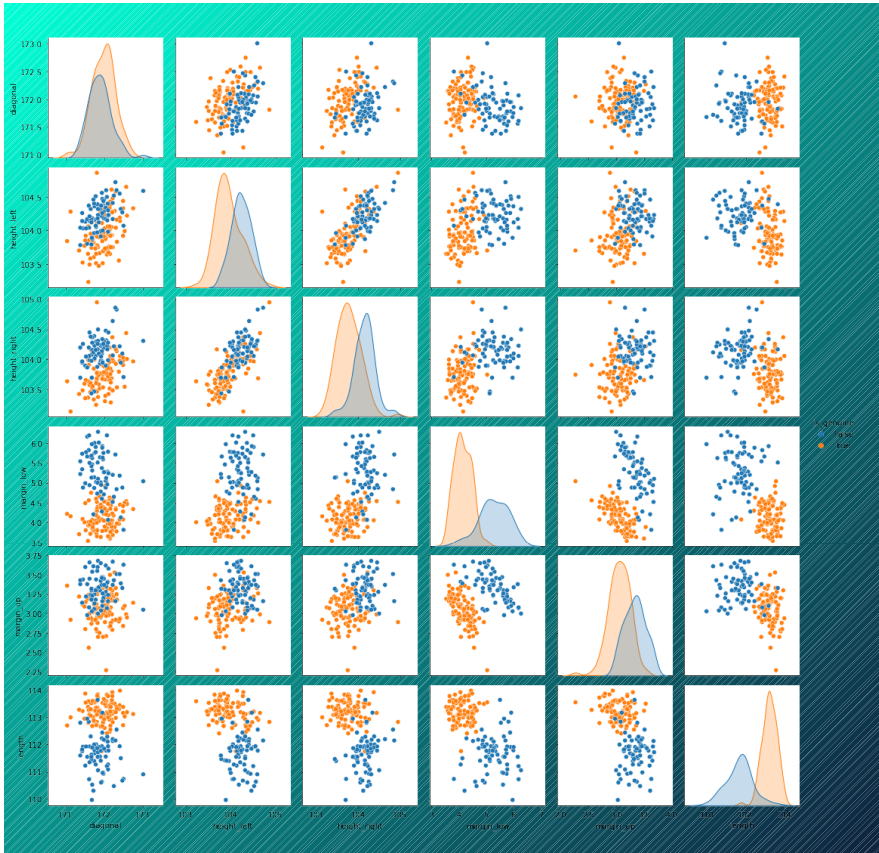


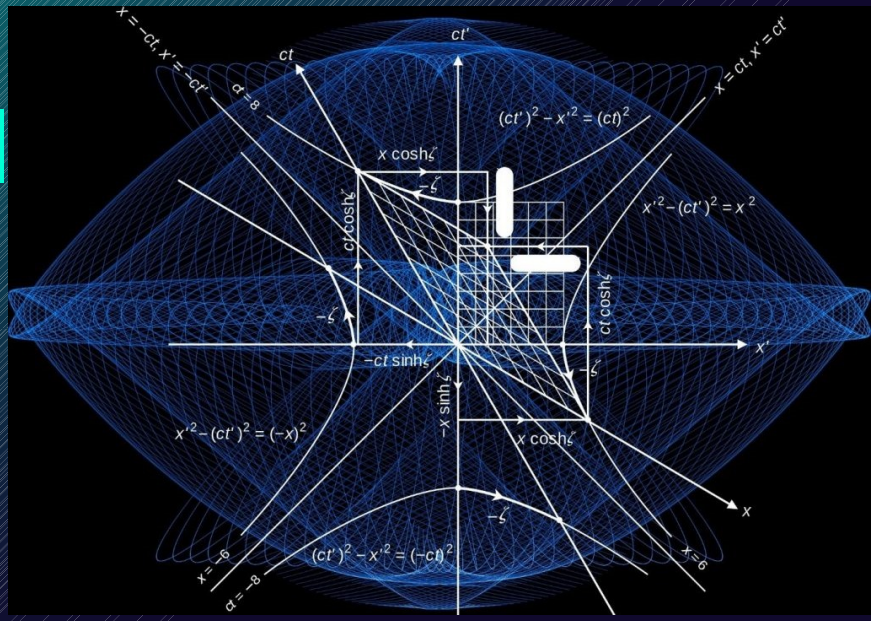
Diagramme en paire

- La diagonale du faux billet est plus petite que la diagonale du vrai billet.
- La hauteur droite et la hauteur gauche sont corrélées (car le billet est rectangulaire) coefficient de corrélation est de 0.73 et la p - value est < 5%. Ces deux variables n'ont aucune influence sur la nature du billet
- On peut distinguer les vrais billets des faux en mesurant leurs longueurs
- La diagonale et la nature du billet ne sont pas corrélées (variable qualitative et variable quantitative)
  La diagonale est indépendante de la nature du billet car eta² <5%

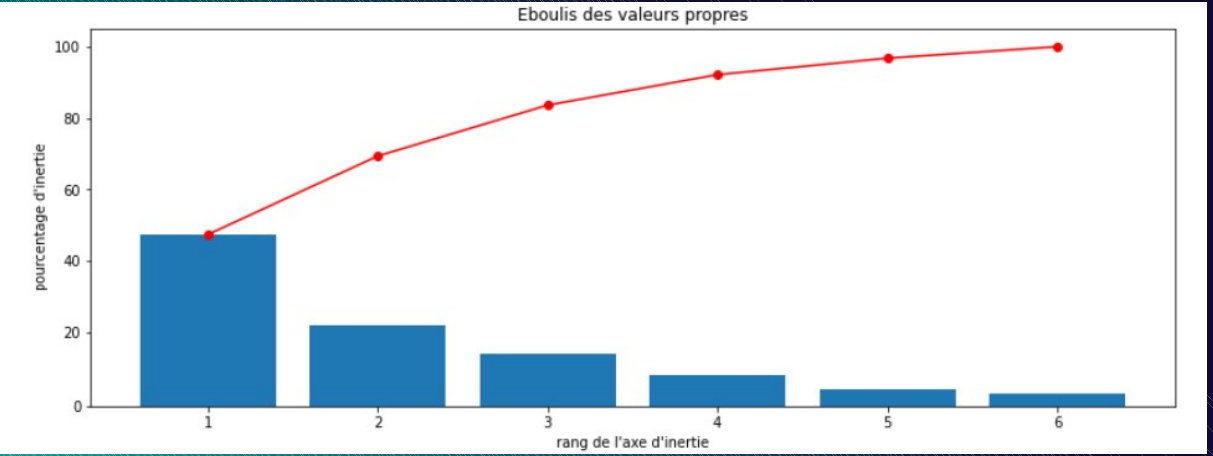
02.

Mission 1

ANALYSE EN COMPOSANTES PRINCIPALES



Eboulis de valeurs propres

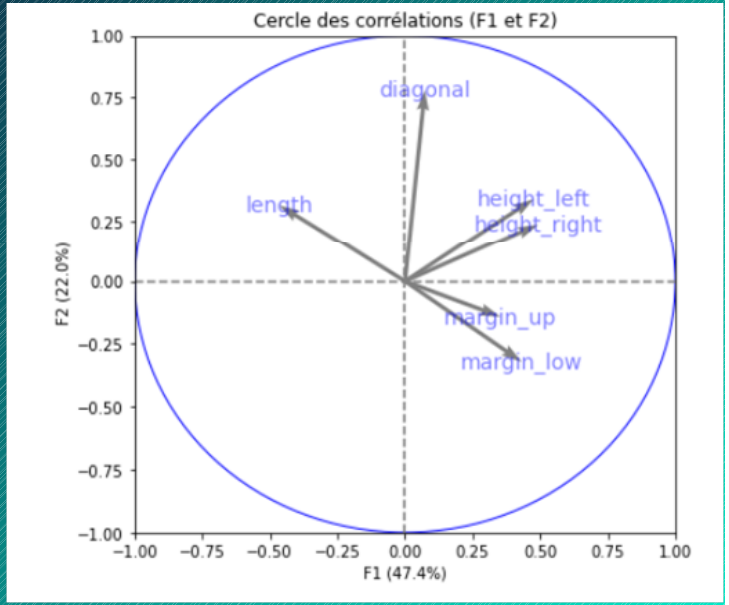


F1 représente 47% des données
F2 en représente 22%.
On peut négliger les autres composantes principales

Cercle des corrélations

On peut interpréter F1 comme étant les caractéristiques du billet.

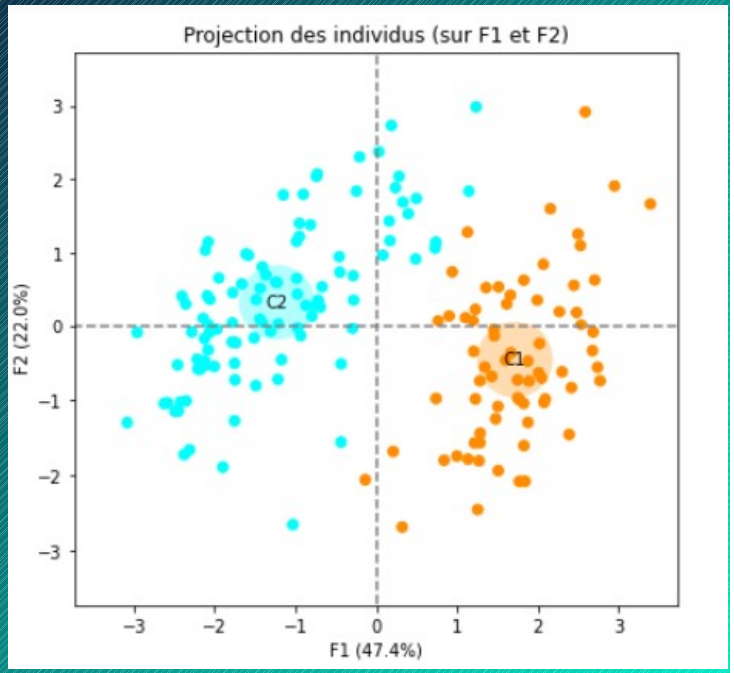
F2 comme étant la diagonale du billet



Projection des individus

Chaque point représente un billet.

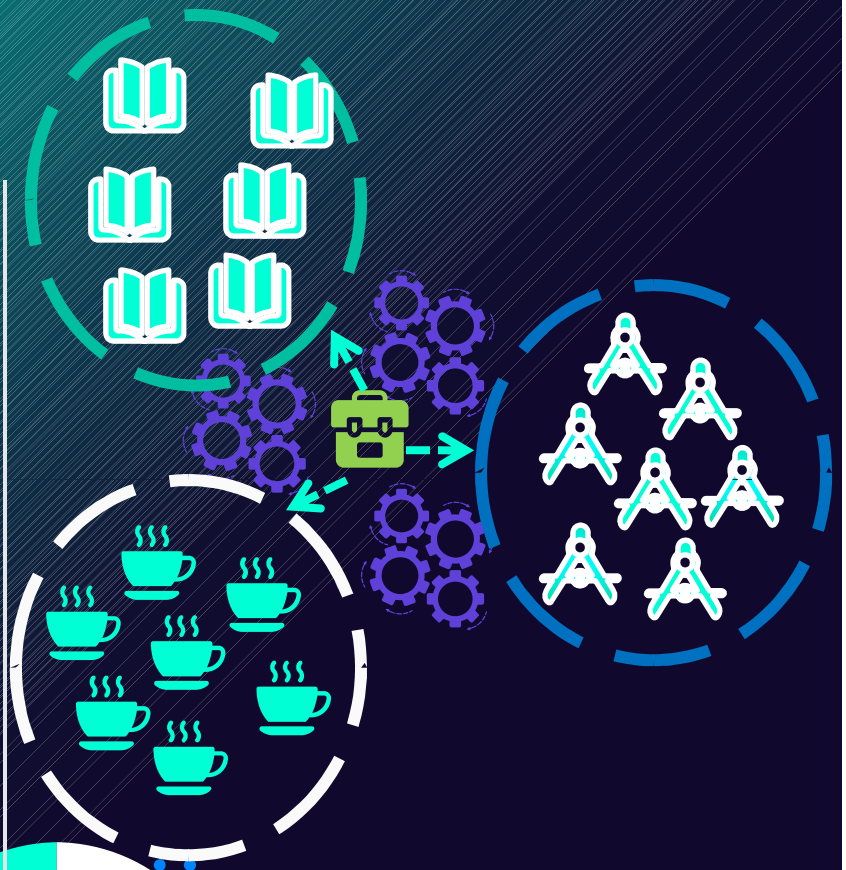
La projection des individus nous montre deux clusters , billets vrai (turquoise) et billets faux (orange)



03.

Mission 2

ALGORITHME DES K PLUS PROCHES VOISINS (k-NN)



Modèle de k-NN

Charger le modèle depuis sklearn

```
from sklearn import neighbors
```

Initialisation du modèle

```
neighbors.KNeighborsClassifier
```

Entraîner les données X et y

```
X = X_projected
y= data['is_genuine'].values
```

Taille du test

20% du jeu de données

Détermination du meilleur k

À l'aide d'un graphique

Entraînement du modèle

```
classifier_knn.fit(X_train, y_train)
```

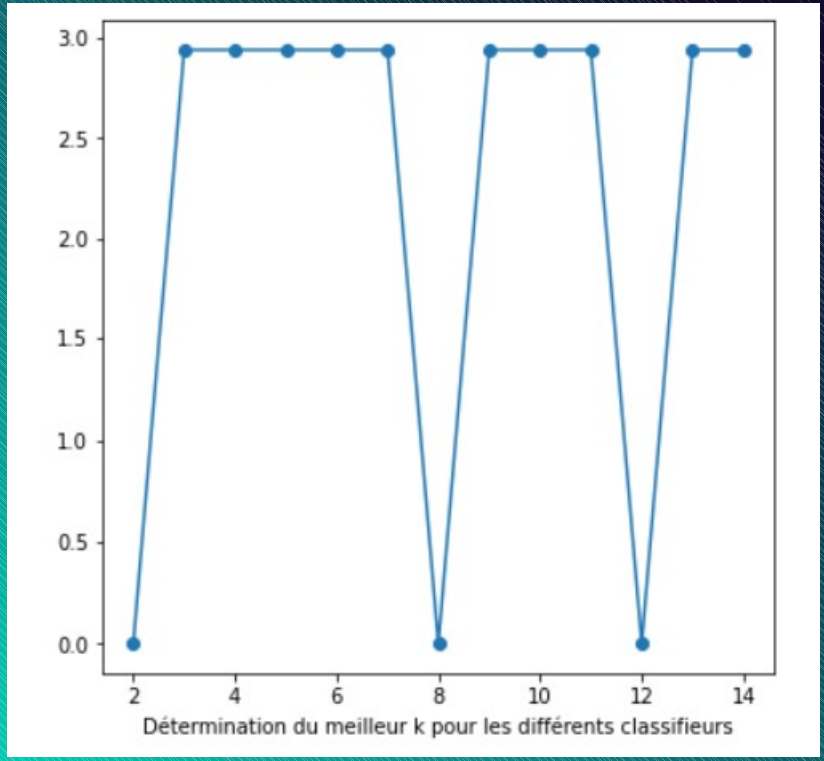
Évaluation de la performance

```
classifier_knn.score(X_test, y_test)
```

Prédiction du modèle

```
classifier_knn.predict(X_test)
```

Évaluation de la performance : Détermination du meilleur k



Le nombre de voisin optimal est de 8

Pour ce k = 8 et 20% des données test on obtient un score égale à 1

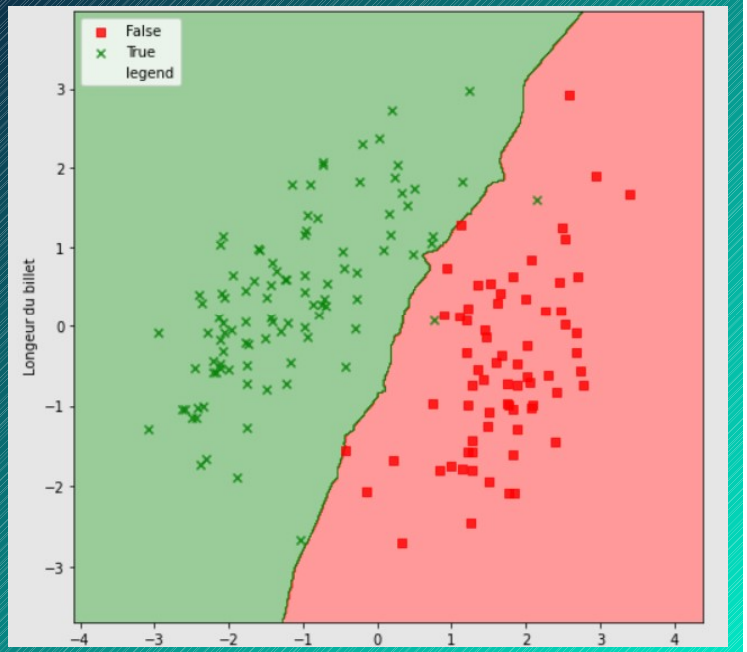
Prédiction du model : Matrice de confusion

	False	True	Prédiction
False	70	0	Is_genuine
True	2	98	

70 billets faux parfaitement prédits
98 billets vrais correctement prédits

Affichage de la région limite

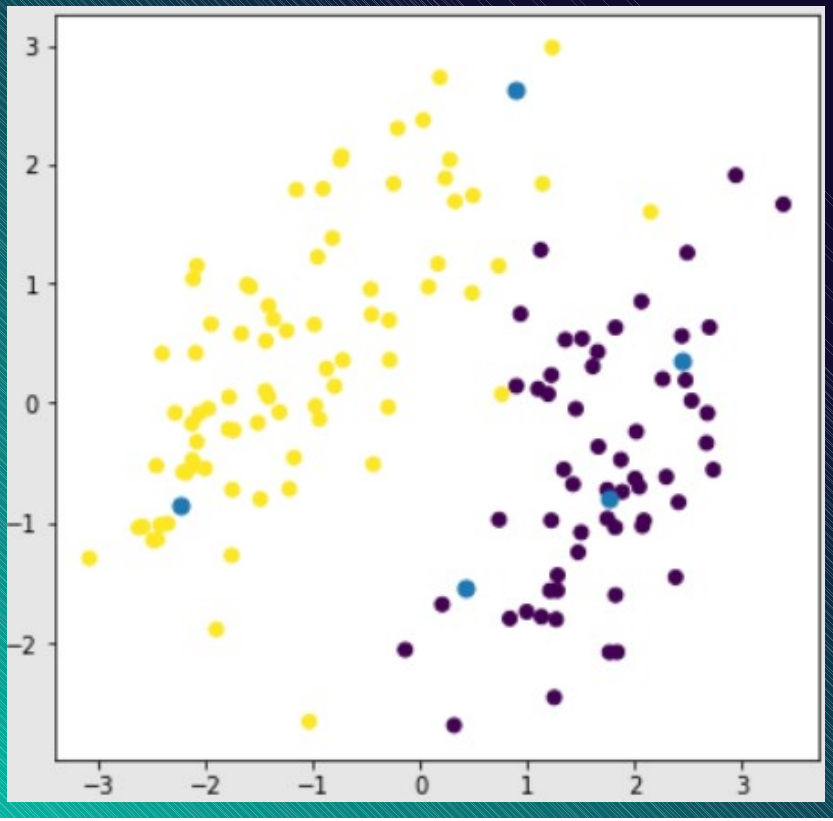
On retrouve bien les deux billets vrais dans la région des faux : Faire une double vérification pour les billets détectés faux par ce modèle



Conclusions

diagonal	height left	height right	margin low	margin up	length	id	proba	prédiction
171.76	104.01	103.54	5.21	3.30	111.42	A 1	0.0	False
171.87	104.17	104.13	6.00	3.31	112.09	A 2	0.0	False
172.00	104.58	104.29	4.99	3.39	111.57	A 3	0.0	False
172.49	104.55	104.34	4.44	3.03	113.20	A 4	1.0	True
171.65	103.63	103.56	3.77	3.16	113.33	A 5	1.0	True

Affichage de la prédiction



04.

Mission 3

MODÉLISER LES DONNÉES À L'AIDE D'UNE RÉGRESSION LOGISTIQUE

Modèle de regréssion logistique

Charger le modèle depuis sklearn

```
from sklearn.linear_model
    import LogisticRegressio
```

Initialisation du modèle

LogisticRegression

Entraîner les données X et y

```
X = X_projected
y= data['is_genuine'].values
```

Taille du test

40% du jeu de données

Entraînement du modèle

```
classifier_logistic.fit(X_train, y_train)
```

Évaluation de la performance

```
classifier_logistic.score(X_test, y_test)
```

Prédiction du model

```
classifier_logistic.predict(X_test)
```

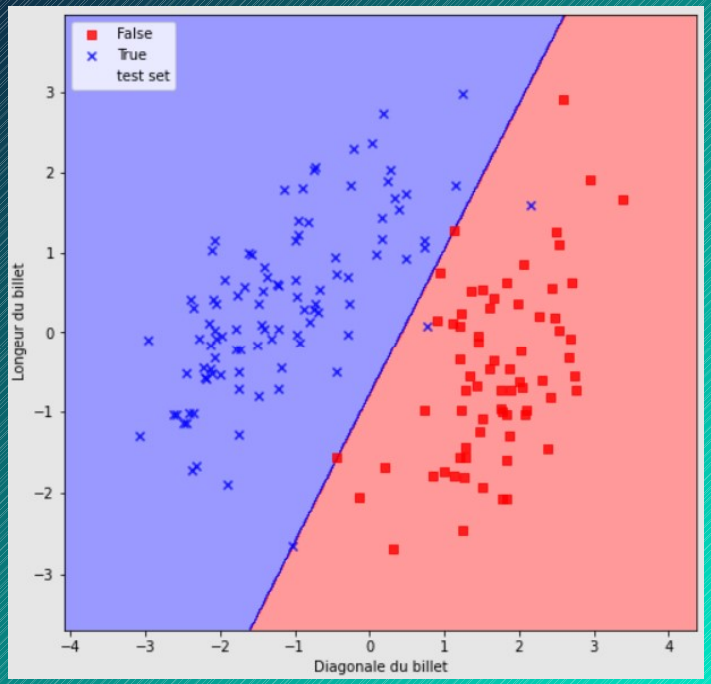
Prédiction du model : Matrice de confusion

Prédiction / Is genuine	False	True
False	68	2
True	2	98

68 billets faux parfaitement prédits
98 billets vrais correctement prédits

Affichage de la région limite

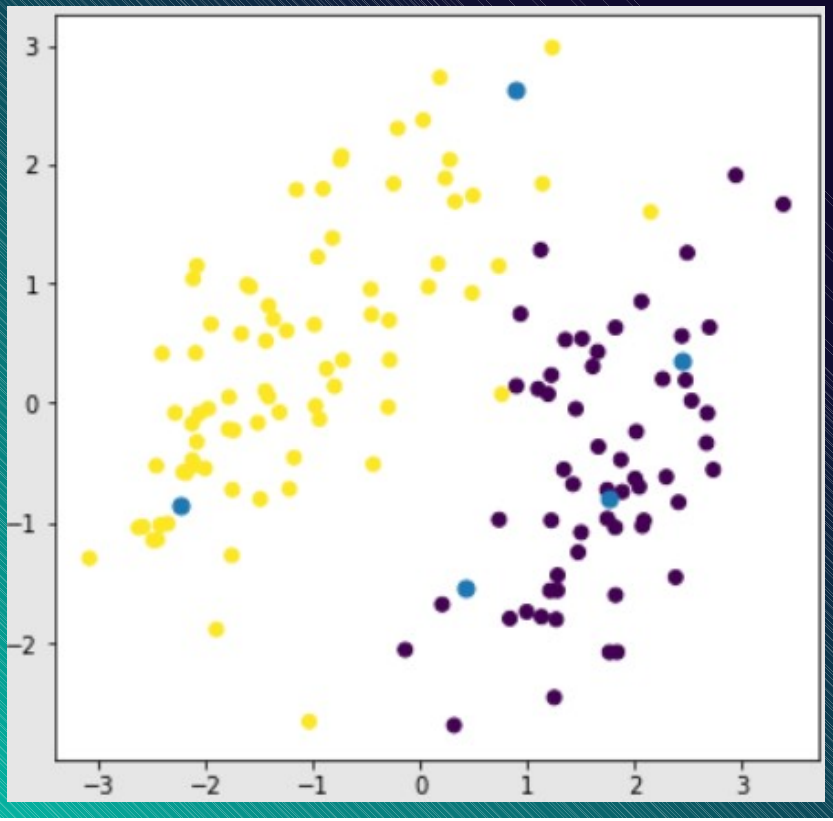
On retrouve bien les deux billets vrais dans la région des faux et à la limite de la région on trouve les deux faux billets. Donc il est nécessaire de calculer la probabilité.



Conclusions

diagonal	height left	height right	margin low	margin up	length	id	proba	prédiction
171.76	104.01	103.54	5.21	3.30	111.42	A 1	0.12	False
171.87	104.17	104.13	6.00	3.31	112.09	A 2	0.015	False
172.00	104.58	104.29	4.99	3.39	111.57	A 3	0.013	False
172.49	104.55	104.34	4.44	3.03	113.20	A 4	0.91	True
171.65	103.63	103.56	3.77	3.16	113.33	A 5	0.99	True

Affichage de la prédiction



Conclusions

Modèle de k-nn

c'est un modèle qui demande beaucoup de ressources pour le mettre en place donc très couteux.

Modèle de régression logistique

C'est un modèle facile à mettre en place qui permet de prédire la probabilité d'un évènement

Bibliographie

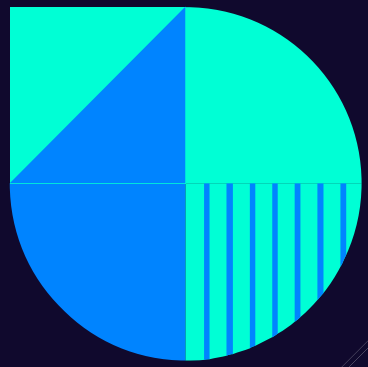
https://www.youtube.com/watch?v=DvupLD0LXb8&t=2208s

https://openclassrooms.com/fr/courses/4525306-initiez-vous-a-la-statistique-inferentielle

https://openclassrooms.com/fr/courses/4525326-realisez-des-modelisations-de-donnees-performantes

https://scikitlearn.org/stable/modules/generated/sklearn.neighbors.KNeighborsClassifier.html#sklearn.neighbors.KNeighborsClassifier

https://openclassrooms.com/fr/courses/4011851-initiez-vous-au-machine-learning/4022441-entrainez-votre-premier-k-nn



MERCI POUR VOTRE ATTENTION

DES QUESTIONS ?

My france Levasseur