Projet NLP (Analyse des Sentiments liés au COVID sur Twitter)

Presenté par :

Mouhammed niah

```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
```

```
df=pd.read_csv("Corona_NLP_train.csv",encoding="ISO-8859-1")
```

Exploration

```
df.head()
```

	UserName	ScreenName	Location	TweetAt	OriginalTweet	Sentiment
0	3799	48751	London	16-03-2020	@MeNyrbie @Phil_Gahan @Chrisitv https://t.co/i...	Neutral
1	3800	48752	UK	16-03-2020	advice Talk to your neighbours family to excha...	Positive
2	3801	48753	Vagabonds	16-03-2020	Coronavirus Australia: Woolworths to give elde...	Positive
3	3802	48754	NaN	16-03-2020	My food stock is not the only one which is emp...	Positive
4	3803	48755	NaN	16-03-2020	Me, ready to go at supermarket during the #COV...	Extremely Negative

Definition des variable

```
UserName : Identifiant unique attribué à chaque utilisateur de Twitter. Il permet de distinguer les différents utilisateurs su

ScreenName : Nom d'écran unique associé à chaque utilisateur de Twitter. C'est le nom qui apparaît sur le profil public de l'u

Location : Emplacement géographique déclaré par l'utilisateur sur son profil Twitter. Cela peut inclure le nom d'une ville, d'

TweetAt : Date et heure auxquelles le tweet a été publié sur Twitter. Cette information permet de suivre le moment précis de l

OriginalTweet : Le texte complet du tweet publié par l'utilisateur. Il s'agit du contenu brut du message partagé sur Twitter,

Sentiment : Le sentiment exprimé dans le tweet, catégorisé en cinq classes : "Positive" (positif), "Negative" (négatif), "Neut
```

```
df.shape
```

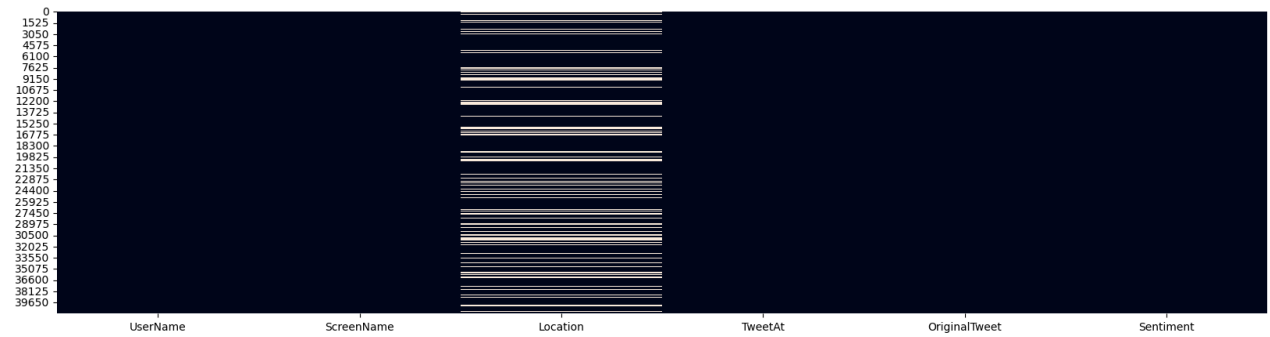
```
(41157, 6)
```

Valeures manquantes

df.isna().sum()/df.shape[0]

```
plt.figure(figsize=(20,5))
sns.heatmap(df.isna(),cbar=False)
```

```
<AxesSubplot:>
```




```
UserName et ScreenName : Aucune valeur manquante dans ces deux colonnes, ce qui signifie que chaque ligne a des informations s

Location : Environ 20.87% des entrées dans la colonne "Location" sont manquantes. Cela indique que près d'un cinquième des uti

TweetAt et OriginalTweet : Aucune valeur manquante dans ces colonnes, ce qui signifie que chaque tweet a été enregistré avec s

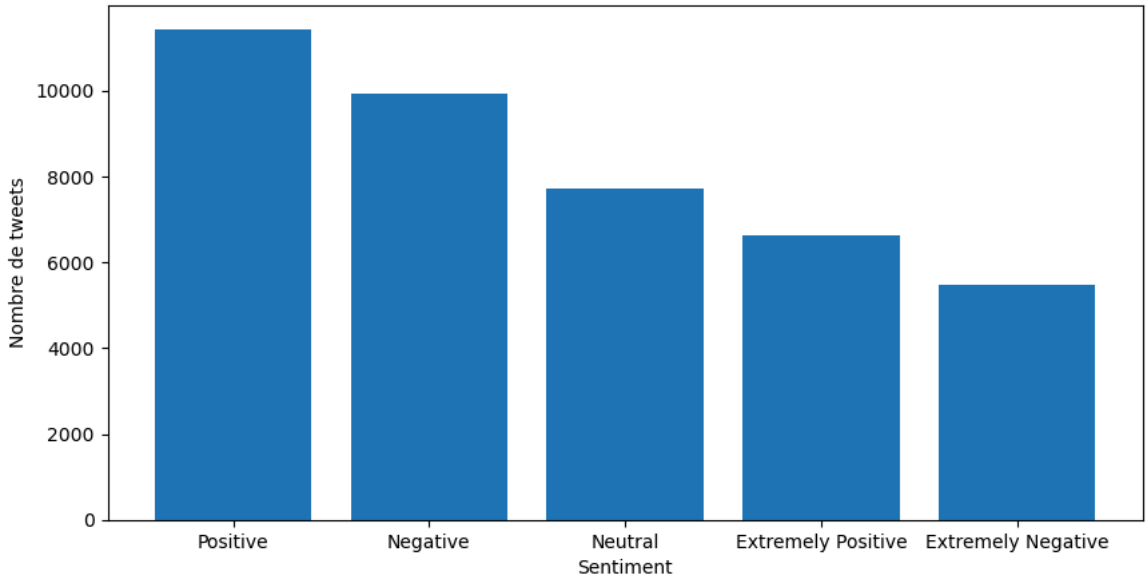
Sentiment : Aucune valeur manquante dans cette colonne, ce qui signifie que chaque tweet a été étiqueté avec un sentiment spéc
```

En résumé, il y a une quantité significative de valeurs manquantes dans la colonne "Location" (environ 21%), ce qui peut être un défi lors de l'analyse, surtout si l'emplacement est une caractéristique cruciale pour l'étude. Pour les autres colonnes, il n'y a pas de valeurs manquantes, ce qui facilitera l'analyse et le traitement ultérieur des données. Il serait peut-être nécessaire de prendre des décisions sur la façon de gérer les valeurs manquantes dans la colonne "Location" en fonction des objectifs spécifiques de l'analyse ou de l'étude.

```
df['Sentiment'].value_counts()
```

```
Positive              11422
Negative               9917
Neutral                7713
Extremely Positive     6624
Extremely Negative     5481
Name: Sentiment, dtype: int64
```

```
import matplotlib.pyplot as plt
sentiment_counts = df['Sentiment'].value_counts()
plt.figure(figsize=(10,5))
plt.bar(sentiment_counts.index, sentiment_counts.values)
plt.xlabel('Sentiment')
plt.ylabel('Nombre de tweets')
plt.show()
```




```
Positive : Il y a 11,422 tweets exprimant un sentiment positif.
Negative : Il y a 9,917 tweets exprimant un sentiment négatif.
Neutral : Il y a 7,713 tweets exprimant un sentiment neutre.
Extremely Positive : Il y a 6,624 tweets exprimant un sentiment extrêmement positif.
Extremely Negative : Il y a 5,481 tweets exprimant un sentiment extrêmement négatif.
```

Ces chiffres montrent la diversité des réactions des utilisateurs de Twitter par rapport à la pandémie COVID-19, allant de la positivité extrême à la négativité extrême.

```
df.shape
```

```
(41157, 6)
```

```
df.loc[1,"OriginalTweet"]
```

```
'advice Talk to your neighbours family to exchange phone numbers create contact list with phone numbers of neighbours
schools employer chemist GP set up online shopping accounts if poss adequate supplies of regular meds but not over order'
```

```
df['Sentiment'].unique()
```

```
array(['Neutral', 'Positive', 'Extremely Negative', 'Negative',
       'Extremely Positive'], dtype=object)
```

Traitement de nos Données

Suppression des ponctuation

La suppression de la ponctuation dans un projet de NLP (Traitement Automatique du Langage Naturel) est une étape préliminaire importante lors de la préparation des données textuelles. Voici quelques raisons pour lesquelles la suppression de la ponctuation est généralement effectuée :

```
Simplification du Texte : La suppression de la ponctuation simplifie le texte en éliminant les caractères non alphabétiques. C

Uniformisation des Mots : Les mots contenant de la ponctuation, comme "word!" et "word", sont traités différemment par les alg

Amélioration de la Cohérence : La suppression de la ponctuation peut améliorer la cohérence des données. Par exemple, "mot." e

Focus sur le Contenu : Dans de nombreux cas, la ponctuation n'ajoute pas de valeur sémantique significative à l'analyse du tex

Réduction de la Dimensionnalité : Dans le cadre de la vectorisation des mots, la suppression de la ponctuation réduit la dimen

Amélioration de la Précision : Certains modèles de NLP peuvent être sensibles à la ponctuation. En la supprimant, vous pouvez

Facilitation de la Tokenisation : La suppression de la ponctuation facilite la découpe du texte en unités discrètes appelées "
```

En résumé, la suppression de la ponctuation est une étape de prétraitement essentielle qui simplifie et prépare le texte pour les étapes ultérieures de l'analyse et du traitement automatique du langage naturel.

```
import string
```

```
string.punctuation
```

```
'!"#$%&\'()*+,-./:;<=>?@[\\]^_`{|}~'
```

```
def suprime_ponctuation(text):
    new_text="".join([ch for ch in text if ch not in string.punctuation])
    return new_text
```

```
df['Sans_ponc_OriginalTweet']=df['OriginalTweet'].apply(lambda x : suprime_ponctuation(x))
```

```
df.head()
```

	UserName	ScreenName	Location	TweetAt	OriginalTweet	Sentiment	Sans_ponc_OriginalTweet
0	3799	48751	London	16-03-2020	@MeNyrbie @Phil_Gahan @Chrisitv https://t.co/i...	Neutral	MeNyrbie PhilGahan Chrisitv httpstcoiFz9FAn2Pa...
1	3800	48752	UK	16-03-2020	advice Talk to your neighbours family to excha...	Positive	advice Talk to your neighbours family to excha...
2	3801	48753	Vagabonds	16-03-2020	Coronavirus Australia: Woolworths to give elde...	Positive	Coronavirus Australia Woolworths to give elder...

==========

```
import re
def tokenizer(text):
    words = re.split('\W+', text)
    return words
```

```
df['list_mot']=df['Sans_ponc_OriginalTweet'].apply(lambda x : tokenizer(x))
```

```
df.head()
```

	UserName	ScreenName	Location	TweetAt	OriginalTweet	Sentiment	Sans_ponc_OriginalTweet	list_mot
0	3799	48751	London	16-03-2020	@MeNyrbie @Phil_Gahan @Chrisitv https://t.co/i...	Neutral	MeNyrbie PhilGahan Chrisitv httpstcoiFz9FAn2Pa...	[MeNyrbie, PhilGahan, Chrisitv, httpstcoiFz9FA...
1	3800	48752	UK	16-03-2020	advice Talk to your neighbours family to excha...	Positive	advice Talk to your neighbours family to excha...	[advice, Talk, to, your, neighbours, family, t...

StopWord

La suppression des stopwords dans un projet de NLP (Traitement Automatique du Langage Naturel) est essentielle pour plusieurs raisons :

```
Réduction du Bruit : Les stopwords, tels que "le", "et", "de", sont des mots couramment utilisés mais n'apportent pas beaucoup

Économie d'Espace et de Temps : Les stopwords apparaissent fréquemment dans un texte, donc les stocker ou les traiter peut con

Amélioration de la Précision : Lorsque l'on se concentre sur les mots significatifs plutôt que sur les stopwords, les modèles

Conservation du Sens : Supprimer les stopwords n'affecte généralement pas le sens global du texte, car ces mots n'apportent pa
```

```
import nltk
```

```
nltk.download('stopwords')
```

```
[nltk_data] Downloading package stopwords to
[nltk_data]     /home/mouhammed/nltk_data...
[nltk_data]   Package stopwords is already up-to-date!
True
```

```
stopwork_en=nltk.corpus.stopwords.words('english')
```

```
stopwork_en
```

```
['i',
 'me',
 'my',
 'myself',
 'we',
 'our',
 'ours',
 'ourselves',
 'you',
 "you're",
 "you've",
 "you'll",
 "you'd",
 'your',
 'yours',
 'yourself',
 'yourselves',
 'he',
 'him',
 'his',
 'himself',
 'she',
 "she's",
 'her',
 'hers',
 'herself',
 'it',
 "it's",
 'its',
 'itself',
 'they',
 'them',
 'their',
 'theirs',
 'themselves',
 'what',
 'which',
 'who',
 'whom',
 'this',
 'that',
 "that'll",
 'these',
 'those',
 'am',
 'is',
 'are',
 'was',
 'were',
 'be',
 'been',
 'being',
 'have',
 'has',
 'had',
 'having',
 'do',
 'does',
```

```
def suprime_stopword(liste):
    result=[word for word in liste if word not in stopwork_en]
    return result
```

```
df['Sans_stopword']=df['list_mot'].apply(lambda x: suprime_stopword(x))
```

```
df.head()
```

	UserName	ScreenName	Location	TweetAt	OriginalTweet	Sentiment	Sans_ponc_OriginalTweet	list_mot	Sans_stopwor
0	3799	48751	London	16-03-2020	@MeNyrbie @Phil_Gahan @Chrisitv https://t.co/i...	Neutral	MeNyrbie PhilGahan Chrisitv httpstcoiFz9FAn2Pa...	[MeNyrbie, PhilGahan, Chrisitv, httpstcoiFz9FA...	[MeNyrbie PhilGahar Chrisitv httpstcoiFz9FA.
1	3800	48752	UK	16-03-2020	advice Talk to your neighbours family to excha...	Positive	advice Talk to your neighbours family to excha...	[advice, Talk, to, your, neighbours, family, t...	[advice, Talk neighbours family exchange, p.
2	3801	48753	Vagabonds	16-03-2020	Coronavirus Australia: Woolworths to	Positive	Coronavirus Australia Woolworths to give elder...	[Coronavirus, Australia, Woolworths, to,	[Coronavirus Australia Woolworths

==========

Stemming

Le Stemming dans un projet de NLP (Traitement Automatique du Langage Naturel) :

```
Réduction des Mots à Leur Racine : Le stemming consiste à réduire les mots à leur racine ou à leur forme de base. Par exemple,

Amélioration de la Cohérence : En réduisant les mots à leur forme de base, le texte devient plus cohérent. Les mots apparentés

Réduction de la Dimensionnalité : En utilisant le stemming, on réduit le nombre de mots différents dans le texte. Cela peut êt

Traitement des Variantes : Le stemming est particulièrement utile lorsque l'on traite des textes provenant de sources différen

Attention aux Erreurs : Cependant, il est important de noter que le stemming peut parfois produire des racines qui ne sont pas
```

```
ps=nltk.PorterStemmer()
```

```
def stemming(liste):
    result=[ps.stem(word) for word in liste]
    return result
```

```
df['liste_stemming']=df['Sans_stopword'].apply(lambda x:stemming(x))
```

```
df.head()
```

	UserName	ScreenName	Location	TweetAt	OriginalTweet	Sentiment	Sans_ponc_OriginalTweet	list_mot	Sans_stopwor
0	3799	48751	London	16-03-2020	@MeNyrbie @Phil_Gahan @Chrisitv https://t.co/i...	Neutral	MeNyrbie PhilGahan Chrisitv httpstcoiFz9FAn2Pa...	[MeNyrbie, PhilGahan, Chrisitv, httpstcoiFz9FA...	[MeNyrbie PhilGahar Chrisitv httpstcoiFz9FA.
1	3800	48752	UK	16-03-2020	advice Talk to your neighbours family to excha...	Positive	advice Talk to your neighbours family to excha...	[advice, Talk, to, your, neighbours, family, t...	[advice, Talk neighbours family exchange, p.
2	3801	48753	Vagabonds	16-03-2020	Coronavirus Australia: Woolworths to	Positive	Coronavirus Australia Woolworths to give elder...	[Coronavirus, Australia, Woolworths, to,	[Coronavirus Australia Woolworths

==========

Lemmatization

La Lemmatisation dans un Projet de NLP (Traitement Automatique du Langage Naturel) :

```
Recherche de la Forme Canonique : Contrairement au stemming, qui se contente de réduire les mots à leur racine, la lemmatisati

Préservation de la Sémantique : La lemmatisation préserve la sémantique des mots. En utilisant la lemmatisation, on obtient de

Contexte Grammatical : La lemmatisation prend en compte le contexte grammatical d'un mot dans une phrase. Elle tient compte de
```

```
Qualité des Résultats : En raison de sa compréhension approfondie des mots et de leur contexte, la lemmatisation produit génér

Utilisation dans les Applications Complexes : La lemmatisation est particulièrement utile dans les applications NLP complexes
```

```
wn = nltk.WordNetLemmatizer()
```

```
def lemmatiz(liste):
    result=[wn.lemmatize(word) for word in liste]
    return result
```

```
df['liste_lemmatization']=df['liste_stemming'].apply(lambda x:stemming(x))
```

```
df.head()
```

	UserName	ScreenName	Location	TweetAt	OriginalTweet	Sentiment	Sans_ponc_OriginalTweet	list_mot	Sans_stopwor
0	3799	48751	London	16-03-2020	@MeNyrbie @Phil_Gahan @Chrisitv https://t.co/i...	Neutral	MeNyrbie PhilGahan Chrisitv httpstcoiFz9FAn2Pa...	[MeNyrbie, PhilGahan, Chrisitv, httpstcoiFz9FA...	[MeNyrbie PhilGahar Chrisitv httpstcoiFz9FA.
1	3800	48752	UK	16-03-2020	advice Talk to your neighbours family to excha...	Positive	advice Talk to your neighbours family to excha...	[advice, Talk, to, your, neighbours, family, t...	[advice, Talk neighbours family exchange, p.
2	3801	48753	Vagabonds	16-03-2020	Coronavirus Australia: Woolworths to give elde...	Positive	Coronavirus Australia Woolworths to give elder...	[Coronavirus, Australia, Woolworths, to, give,...	[Coronavirus Australia Woolworths give, eld.

```
def grouper(liste):
    return " ".join(liste)
```

```
df['clean_tweet']=df['liste_lemmatization'].apply(lambda x:grouper(x))
```

```
df.head()
```

	UserName	ScreenName	Location	TweetAt	OriginalTweet	Sentiment	Sans_ponc_OriginalTweet	list_mot	Sans_stopwor
0	3799	48751	London	16-03-2020	@MeNyrbie @Phil_Gahan @Chrisitv https://t.co/i...	Neutral	MeNyrbie PhilGahan Chrisitv httpstcoiFz9FAn2Pa...	[MeNyrbie, PhilGahan, Chrisitv, httpstcoiFz9FA...	[MeNyrbie PhilGahar Chrisitv httpstcoiFz9FA.
1	3800	48752	UK	16-03-2020	advice Talk to your neighbours family to excha...	Positive	advice Talk to your neighbours family to excha...	[advice, Talk, to, your, neighbours, family, t...	[advice, Talk neighbours family exchange, p.
2	3801	48753	Vagabonds	16-03-2020	Coronavirus Australia: Woolworths to give elde...	Positive	Coronavirus Australia Woolworths to give elder...	[Coronavirus, Australia, Woolworths, to, give,...	[Coronavirus Australia Woolworths give, eld.
3	3802	48754	NaN	16-03-2020	My food stock is not the only one which is	Positive	My food stock is not the only one which is emp...	[My, food, stock, is, not, the, only, one, whi	[My, food, stock one, empty PLEASE, don

```
data=df[['UserName','ScreenName','Location','clean_tweet','Sentiment']]
```

```
data.head()
```

	UserName	ScreenName	Location	clean_tweet	Sentiment
0	3799	48751	London	menyrbi philgahan chrisitv httpstcoifz9fan2pa ...	Neutral
1	3800	48752	UK	advic talk neighbour famili exchang phone numb...	Positive
2	3801	48753	Vagabonds	coronaviru australia woolworth give elderli di...	Positive
3	3802	48754	NaN	my food stock one empti plea dont panic there ...	Positive
4	3803	48755	NaN	me readi go supermarket covid19 outbreak not i...	Extremely Negative

```
Commencez à coder ou à générer avec l'IA.
```

==========

Modelisation

Vectorization

La Vectorisation dans un Projet de NLP (Traitement Automatique du Langage Naturel) :

```
Transformation des Mots en Vecteurs : La vectorisation dans le contexte du NLP consiste à convertir des mots ou des phrases en

Utilisation d'Algorithmes d'Apprentissage : La vectorisation est essentielle car de nombreux algorithmes d'apprentissage autom

Différentes Méthodes de Vectorisation : Il existe différentes techniques de vectorisation, notamment la TF-IDF (Term Frequency

Capture de la Sémantique : Les méthodes de vectorisation basées sur le contexte, comme Word2Vec, capturent également la sémant

Réduction de la Dimensionalité : La vectorisation peut également impliquer des techniques de réduction de la dimensionalité, t

Amélioration de la Précision : La vectorisation permet d'améliorer la précision des modèles NLP en fournissant une représentat
```

```
from sklearn.feature_extraction.text import CountVectorizer,TfidfVectorizer
```

```
vectorisation=TfidfVectorizer(max_features=20000)
```

```
vect=vectorisation.fit_transform(data['clean_tweet'])
```

```
vect.toarray()
```

```
array([[0., 0., 0., ..., 0., 0., 0.],
       [0., 0., 0., ..., 0., 0., 0.],
       [0., 0., 0., ..., 0., 0., 0.],
       ...,
       [0., 0., 0., ..., 0., 0., 0.],
       [0., 0., 0., ..., 0., 0., 0.],
       [0., 0., 0., ..., 0., 0., 0.]])
```

```
vect[:10, :10].toarray()
```

```
array([[0., 0., 0., 0., 0., 0., 0., 0., 0., 0.],
       [0., 0., 0., 0., 0., 0., 0., 0., 0., 0.],
       [0., 0., 0., 0., 0., 0., 0., 0., 0., 0.],
       [0., 0., 0., 0., 0., 0., 0., 0., 0., 0.],
       [0., 0., 0., 0., 0., 0., 0., 0., 0., 0.],
       [0., 0., 0., 0., 0., 0., 0., 0., 0., 0.],
       [0., 0., 0., 0., 0., 0., 0., 0., 0., 0.],
       [0., 0., 0., 0., 0., 0., 0., 0., 0., 0.],
       [0., 0., 0., 0., 0., 0., 0., 0., 0., 0.],
       [0., 0., 0., 0., 0., 0., 0., 0., 0., 0.]])
```

```
vect.shape
```

```
(41157, 20000)
```

```
from sklearn.preprocessing import LabelEncoder
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.naive_bayes import MultinomialNB
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import precision_recall_fscore_support as score
```

```
le=LabelEncoder()
```

```
y = data['Sentiment']
y = le.fit_transform(y)
```

```
X_train, X_test, Y_train, Y_test = train_test_split(vect.toarray(), y, test_size=0.2, random_state=42)
```

```
def random_forest_hyper_params(n_decision_tree, depth):
    alg_RandomForest = RandomForestClassifier(n_estimators=n_decision_tree, max_depth=depth, n_jobs=-1)
    model = alg_RandomForest.fit(X_train, Y_train)
    predictions = model.predict(X_test)
    precision, recall, fscore, support = score(Y_test, predictions, average='weighted')
    print('nb decision tree: {} / Depth: {} | Precision: {} / Recall: {}'.format(
        n_decision_tree, depth, round(precision, 3), round(recall, 3)))
```

```
for ith_decision_tree in [5, 20, 50,100]:

    for depth in [10, 20]:

        random_forest_hyper_params(ith_decision_tree, depth)
```

```
nb decision tree: 5 / Depth: 10 | Precision: 0.444 / Recall: 0.317
nb decision tree: 5 / Depth: 20 | Precision: 0.421 / Recall: 0.345
nb decision tree: 20 / Depth: 10 | Precision: 0.535 / Recall: 0.294
nb decision tree: 20 / Depth: 20 | Precision: 0.522 / Recall: 0.357
nb decision tree: 50 / Depth: 10 | Precision: 0.636 / Recall: 0.295
nb decision tree: 50 / Depth: 20 | Precision: 0.579 / Recall: 0.35
nb decision tree: 100 / Depth: 10 | Precision: 0.649 / Recall: 0.295
nb decision tree: 100 / Depth: 20 | Precision: 0.598 / Recall: 0.345
```

```
from sklearn.svm import SVC
```

```
#from sklearn.model_selection import GridSearchCV

# Définir les paramètres que vous souhaitez tester
#param_grid = {
 #   'kernel': ['linear', 'poly'],
 #   'C': [0.1, 1, 10],
#}
```