import pandas as pd

from sklearn.ensemble import RandomForestClassifier

# Chargez vos données dans un DataFrame

data = pd.read\_csv('votre\_fichier.csv')

# Calculez la matrice de corrélation

correlation\_matrix = data.corr()

# Définissez un seuil de corrélation (par exemple, 0.7)

threshold = 0.7

# Identifiez les paires de variables fortement corrélées

correlated\_pairs = set()

for i in range(len(correlation\_matrix.columns)):

for j in range(i):

if abs(correlation\_matrix.iloc[i, j]) > threshold:

correlated\_pairs.add((correlation\_matrix.columns[i], correlation\_matrix.columns[j]))

# Créez un modèle RandomForestClassifier pour évaluer l'importance des variables

model = RandomForestClassifier()

X = data.drop(columns=['variable\_cible']) # Assurez-vous de retirer la variable cible

y = data['variable\_cible']

model.fit(X, y)

# Choisissez la variable à conserver pour chaque paire

variables\_to\_keep = set()

for var1, var2 in correlated\_pairs:

importance\_var1 = model.feature\_importances\_[X.columns.get\_loc(var1)]

importance\_var2 = model.feature\_importances\_[X.columns.get\_loc(var2)]

if importance\_var1 > importance\_var2:

variable\_to\_keep = var1

else:

variable\_to\_keep = var2

variables\_to\_keep.add(variable\_to\_keep)

# Supprimez les variables redondantes du DataFrame

data = data.drop(columns=[var for var in data.columns if var not in variables\_to\_keep])

# Enregistrez le DataFrame résultant dans un nouveau fichier si nécessaire

data.to\_csv('nouveau\_fichier.csv', index=False)