import pandas as pd

import numpy as np

from scipy.fft import fft

from sklearn.preprocessing import StandardScaler

def extract\_features(signal):

features = {

'mean': np.mean(signal),

'std': np.std(signal),

'max': np.max(signal),

'min': np.min(signal),

'kurtosis': pd.Series(signal).kurt(),

'skewness': pd.Series(signal).skew()

}

freqs = np.abs(fft(signal))[:len(signal)//2]

features['fft\_max'] = np.max(freqs)

features['fft\_mean'] = np.mean(freqs)

return features

def load\_and\_process(file\_path):

df = pd.read\_csv(file\_path)

samples = []

labels = []

for idx, row in df.iterrows():

signal = np.array(row[:-1])

label = row[-1]

feat = extract\_features(signal)

samples.append(list(feat.values()))

labels.append(label)

return np.array(samples), np.array(labels)

if \_\_name\_\_ == "\_\_main\_\_":

X, y = load\_and\_process('data/signal\_data.csv')

scaler = StandardScaler()

X\_scaled = scaler.fit\_transform(X)

np.savez('processed\_data.npz', X=X\_scaled, y=y)