|  |
| --- |
|  |
| import numpy as np |
|  | import pandas as pd |
|  | from sklearn.neural\_network import MLPClassifier |
|  | from sklearn.model\_selection import train\_test\_split |
|  | from sklearn.preprocessing import StandardScaler |
|  | import joblib |
|  |  |
|  | df=pd.read\_csv('forestfires.csv') |
|  |  |
|  | df=df.drop(['X','Y','month','day'],axis=1) |
|  |  |
|  | def preprocessing(df,task): |
|  | df=df.copy() |
|  | if task=='Regression': |
|  | Y=df['area'] |
|  | elif task=='Classification': |
|  | Y=df['area'].apply(lambda x: 1 if x>0 else 0) |
|  |  |
|  | X=df.drop('area',axis=1) |
|  |  |
|  | X\_train,X\_test,Y\_train,Y\_test=train\_test\_split(X,Y,train\_size=0.60,shuffle=True,random\_state=0) |
|  |  |
|  | scaler=StandardScaler() |
|  | scaler.fit(X\_train) |
|  |  |
|  | X\_train=pd.DataFrame(scaler.transform(X\_train),columns=X.columns) |
|  | X\_test=pd.DataFrame(scaler.transform(X\_test),columns=X.columns) |
|  |  |
|  | return X\_train,X\_test,Y\_train,Y\_test |
|  |  |
|  |  |
|  | X\_train,X\_test,Y\_train,Y\_test=preprocessing(df,task='Classification') |
|  |  |
|  |  |
|  | nn\_classifier\_model=MLPClassifier(activation='relu',hidden\_layer\_sizes=(16,16),n\_iter\_no\_change=100,solver='adam') |
|  | nn\_classifier\_model.fit(X\_train,Y\_train) |
|  |  |
|  | model=joblib.dump(nn\_classifier\_model,'forestfiremodel.pkl') |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |