```python
!pip3 install keras-tuner
!pip3 install tensorflow


import pandas as pd
from google.colab import drive
from tensorflow import keras
from tensorflow.keras import layers
from kerastuner.tuners import RandomSearch
from sklearn.preprocessing import StandardScaler
drive.mount('/content/drive')
wb = '/content/drive/My Drive/DATA/ANN_Data.xlsx'




df = pd.read_excel(wb, sheet_name='Sheet1', header=0)


df.head()


scaler = StandardScaler()
X = df.drop(labels=['Damage','DI'], axis=1)
#y = dataset['DI']
X = df[['HYSD500','Fck','Depth of beam','Width of column','Density ', 'Pd','D','K0']]
y = pd.get_dummies(df['Damage']).values
X = scaler.fit_transform(X)


def build_model(hp):
    model = keras.Sequential()
    for i in range(hp.Int('num_layers', 1, 8)):
        model.add(layers.Dense(units=hp.Int('units_' + str(i),
                                             min_value=4,
                                             max_value=16,
                                             step=4),
                               activation=hp.Choice('act_' + str(i), ['relu','sigmoid','so
    model.add(layers.Dense(4, activation='softmax'))
    model.compile(
        optimizer=keras.optimizers.RMSprop(
            hp.Choice('learning_rate', [1e-2, 1e-3, 1e-4])),
        loss='binary_crossentropy',
        metrics=['accuracy'])
    return model


tuner = RandomSearch(
    build_model,
    objective='val_accuracy',
    max_trials=5,
    executions_per_trial=3,
    directory='project-21',
    project_name='Hyper tuning ANN with activation modification-2')


tuner.search_space_summary()
```

```python
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=0)


tuner.search(X_train, y_train,
             epochs=5,
             validation_data=(X_test, y_test))


tuner.results_summary()


best_model = tuner.get_best_models()[0]
best_model.build(X_train.shape)
best_model.summary()
```

✓  0s    completed at 3:43 PM                                    ● ✕