

Practical Implementation of Boston Dataset and prediction using deep neural network.

Step 1: Load the dataset

```
import pandas as pd

# Load the dataset from a CSV file
df = pd.read_csv('boston_housing.csv')

# Display the first few rows of the dataset
print(df.head())
```

Step 2: Preprocess the data

```
from sklearn.preprocessing import StandardScaler

# Split the data into input and output variables
X = df.drop('medv', axis=1)
y = df['medv']

# Scale the input features
scaler = StandardScaler()
X = scaler.fit_transform(X)

# Display the first few rows of the scaled input features
print(X[:5])
```

Step 3: Split the dataset

```
from sklearn.model_selection import train_test_split

# Split the data into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=42)

# Print the shapes of the training and testing sets
print('Training set shape:', X_train.shape, y_train.shape)
print('Testing set shape:', X_test.shape, y_test.shape)
```

Step 4: Define the model architecture

```
from keras.models import Sequential
from keras.layers import Dense, Dropout

# Define the model architecture
model = Sequential()
model.add(Dense(64, input_dim=13, activation='relu'))
model.add(Dropout(0.2))
```

```
model.add(Dense(32, activation='relu'))
```

```
model.add(Dense(1))
```

```
# Display the model summary
```

```
print(model.summary())
```

Step 5: Compile the model

```
# Compile the model
```

```
model.compile(loss='mean_squared_error',optimizer='adam',metrics=['mean_absolute_error'])
```

Step 6: Train the model

```
from keras.callbacks import EarlyStopping
```

```
# Train the model
```

```
early_stopping = EarlyStopping(monitor='val_loss', patience=5)
```

```
history = model.fit(X_train, y_train, validation_split=0.2, epochs=100, batch_size=32,  
callbacks=[early_stopping])
```