Neural Network Q1 (Forest_Fires)

PREDICT THE BURNED AREA OF FOREST FIRES WITH NEURAL NETWORKS

1. import Libs

In [1]:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.model_selection import train_test_split
from keras.models import Sequential
from keras.layers import Dense, Activation,Layer,Lambda
from sklearn.metrics import confusion_matrix
from sklearn.decomposition import PCA
from mlxtend.plotting import plot_decision_regions
from keras.models import Sequential
from keras.callbacks import History
history = History()
```

2. Import Data

In [2]:

```
forest_fire = pd.read_csv('forestfires.csv')
forest_fire
```

Out[2]:

	month	day	FFMC	DMC	DC	ISI	temp	RH	wind	rain	...	monthfeb	monthjan	m
0	mar	fri	86.2	26.2	94.3	5.1	8.2	51	6.7	0.0	...	0	0	
1	oct	tue	90.6	35.4	669.1	6.7	18.0	33	0.9	0.0	...	0	0	
2	oct	sat	90.6	43.7	686.9	6.7	14.6	33	1.3	0.0	...	0	0	
3	mar	fri	91.7	33.3	77.5	9.0	8.3	97	4.0	0.2	...	0	0	
4	mar	sun	89.3	51.3	102.2	9.6	11.4	99	1.8	0.0	...	0	0	
...	...	...	...	...	...	...	...	...	...	...	...	...	...	
512	aug	sun	81.6	56.7	665.6	1.9	27.8	32	2.7	0.0	...	0	0	
513	aug	sun	81.6	56.7	665.6	1.9	21.9	71	5.8	0.0	...	0	0	
514	aug	sun	81.6	56.7	665.6	1.9	21.2	70	6.7	0.0	...	0	0	
515	aug	sat	94.4	146.0	614.7	11.3	25.6	42	4.0	0.0	...	0	0	
516	nov	tue	79.5	3.0	106.7	1.1	11.8	31	4.5	0.0	...	0	0	

517 rows × 31 columns

3. EDA

In [3]:

```
forest_fire.isna().sum()
```

Out[3]:

```
month            0
day              0
FFMC             0
DMC              0
DC               0
ISI              0
temp             0
RH               0
wind             0
rain             0
area             0
dayfri           0
daymon           0
daysat           0
daysun           0
daythu           0
daytue           0
daywed           0
monthapr         0
monthaug         0
monthdec         0
monthfeb         0
monthjan         0
monthjul         0
monthjun         0
monthmar         0
monthmay         0
monthnov         0
monthoct         0
monthsep         0
size_category    0
dtype: int64
```

In [4]:

```
forest_fire.dtypes
```

Out[4]:

```
month             object
day               object
FFMC             float64
DMC              float64
DC               float64
ISI              float64
temp             float64
RH                 int64
wind             float64
rain             float64
area             float64
dayfri             int64
daymon             int64
daysat             int64
daysun             int64
daythu             int64
daytue             int64
daywed             int64
monthapr           int64
monthaug           int64
monthdec           int64
monthfeb           int64
monthjan           int64
monthjul           int64
monthjun           int64
monthmar           int64
monthmay           int64
monthnov           int64
monthoct           int64
monthsep           int64
size_category     object
dtype: object
```

In [5]:

```
forest_fire['size_category'].value_counts()
```

Out[5]:

```
small    378
large    139
Name: size_category, dtype: int64
```

In [6]:

```
forest_fire.describe().T
```

Out[6]:

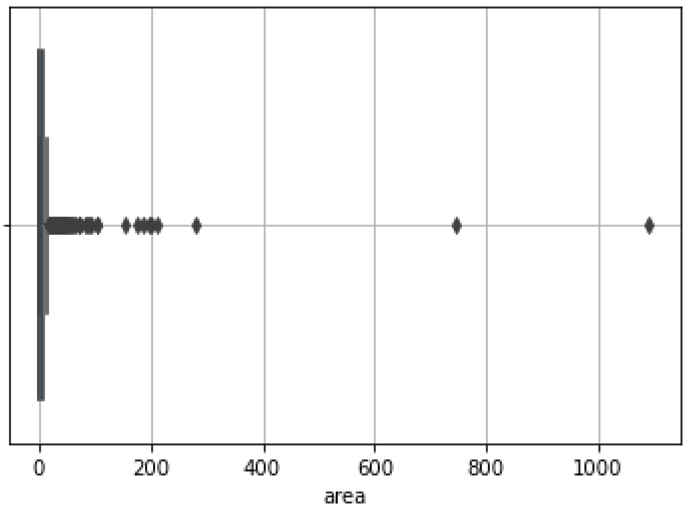
	count	mean	std	min	25%	50%	75%	max
FFMC	517.0	90.644681	5.520111	18.7	90.2	91.60	92.90	96.20
DMC	517.0	110.872340	64.046482	1.1	68.6	108.30	142.40	291.30
DC	517.0	547.940039	248.066192	7.9	437.7	664.20	713.90	860.60
ISI	517.0	9.021663	4.559477	0.0	6.5	8.40	10.80	56.10
temp	517.0	18.889168	5.806625	2.2	15.5	19.30	22.80	33.30
RH	517.0	44.288201	16.317469	15.0	33.0	42.00	53.00	100.00
wind	517.0	4.017602	1.791653	0.4	2.7	4.00	4.90	9.40
rain	517.0	0.021663	0.295959	0.0	0.0	0.00	0.00	6.40
area	517.0	12.847292	63.655818	0.0	0.0	0.52	6.57	1090.84
dayfri	517.0	0.164410	0.371006	0.0	0.0	0.00	0.00	1.00
daymon	517.0	0.143133	0.350548	0.0	0.0	0.00	0.00	1.00
daysat	517.0	0.162476	0.369244	0.0	0.0	0.00	0.00	1.00
daysun	517.0	0.183752	0.387657	0.0	0.0	0.00	0.00	1.00
daythu	517.0	0.117988	0.322907	0.0	0.0	0.00	0.00	1.00
daytue	517.0	0.123791	0.329662	0.0	0.0	0.00	0.00	1.00
daywed	517.0	0.104449	0.306138	0.0	0.0	0.00	0.00	1.00
monthapr	517.0	0.017408	0.130913	0.0	0.0	0.00	0.00	1.00
monthaug	517.0	0.355899	0.479249	0.0	0.0	0.00	1.00	1.00
monthdec	517.0	0.017408	0.130913	0.0	0.0	0.00	0.00	1.00
monthfeb	517.0	0.038685	0.193029	0.0	0.0	0.00	0.00	1.00
monthjan	517.0	0.003868	0.062137	0.0	0.0	0.00	0.00	1.00
monthjul	517.0	0.061896	0.241199	0.0	0.0	0.00	0.00	1.00
monthjun	517.0	0.032882	0.178500	0.0	0.0	0.00	0.00	1.00
monthmar	517.0	0.104449	0.306138	0.0	0.0	0.00	0.00	1.00
monthmay	517.0	0.003868	0.062137	0.0	0.0	0.00	0.00	1.00
monthnov	517.0	0.001934	0.043980	0.0	0.0	0.00	0.00	1.00
monthoct	517.0	0.029014	0.168007	0.0	0.0	0.00	0.00	1.00
monthsep	517.0	0.332689	0.471632	0.0	0.0	0.00	1.00	1.00

Checking Outlires

In [7]:

```
sns.boxplot(forest_fire['area'])
plt.grid()
```

```
C:\Users\shubham\anaconda3\lib\site-packages\seaborn\_decorators.py:36: Futu
reWarning: Pass the following variable as a keyword arg: x. From version 0.1
2, the only valid positional argument will be `data`, and passing other argu
ments without an explicit keyword will result in an error or misinterpretati
on.
  warnings.warn(
```

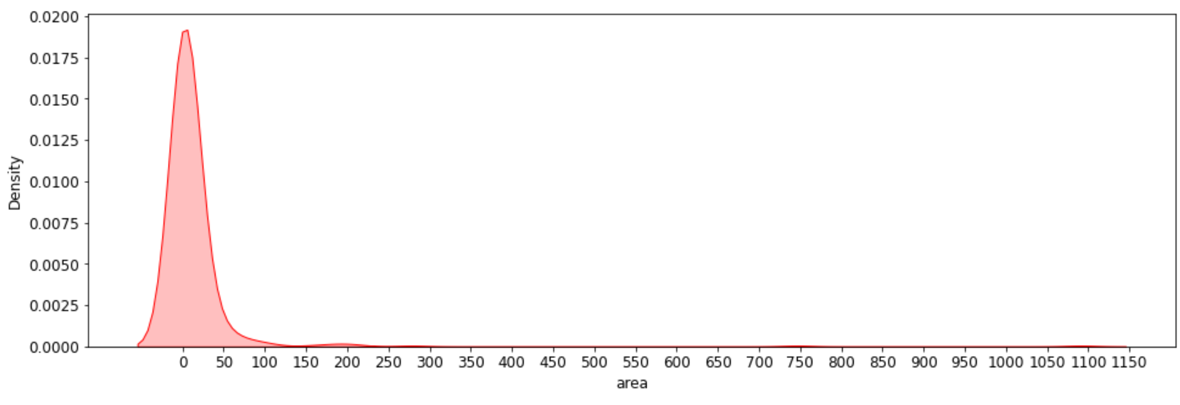


There are 3 Outlier instances in our data

In [118]:

```
plt.figure(figsize=(16,5))
print("Skewness =",forest_fire['area'].skew())
print("Kurtosis =",forest_fire['area'].kurtosis())
sns.kdeplot(forest_fire['area'],shade=True,color='r')
plt.xticks([i for i in range(0,1200,50)])
plt.show()
```

```
Skewness = 12.846933533934868
Kurtosis = 194.1407210942299
```



The Data is highly skewed and has large kurtosis value

Majority of the forest fires do not cover a large area, most of the damaged area is under 100 hectares of land

In [119]:

```
dfa = forest_fire[forest_fire.columns[0:10]]
month_colum = dfa.select_dtypes(include='object')
month_colum
```

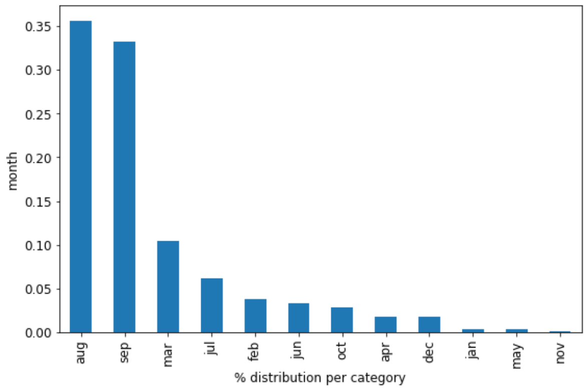
Out[119]:

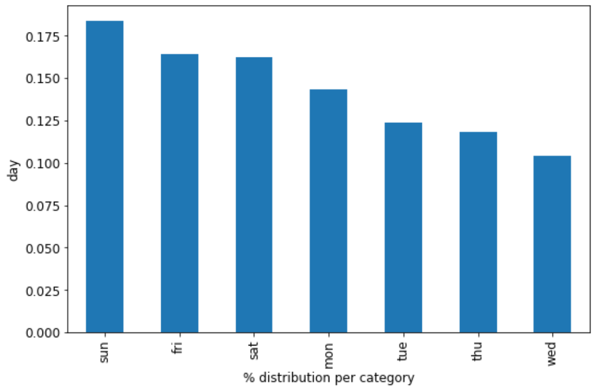
	month	day
0	mar	fri
1	oct	tue
2	oct	sat
3	mar	fri
4	mar	sun
...	...	...
512	aug	sun
513	aug	sun
514	aug	sun
515	aug	sat
516	nov	tue

517 rows × 2 columns

In [120]:

```
plt.figure(figsize=(16,10))
for i,col in enumerate(month_colum,1):
    plt.subplot(2,2,i)
    forest_fire[col].value_counts(normalize=True).plot.bar()
    plt.ylabel(col)
    plt.xlabel('% distribution per category')
plt.tight_layout()
plt.show()
```



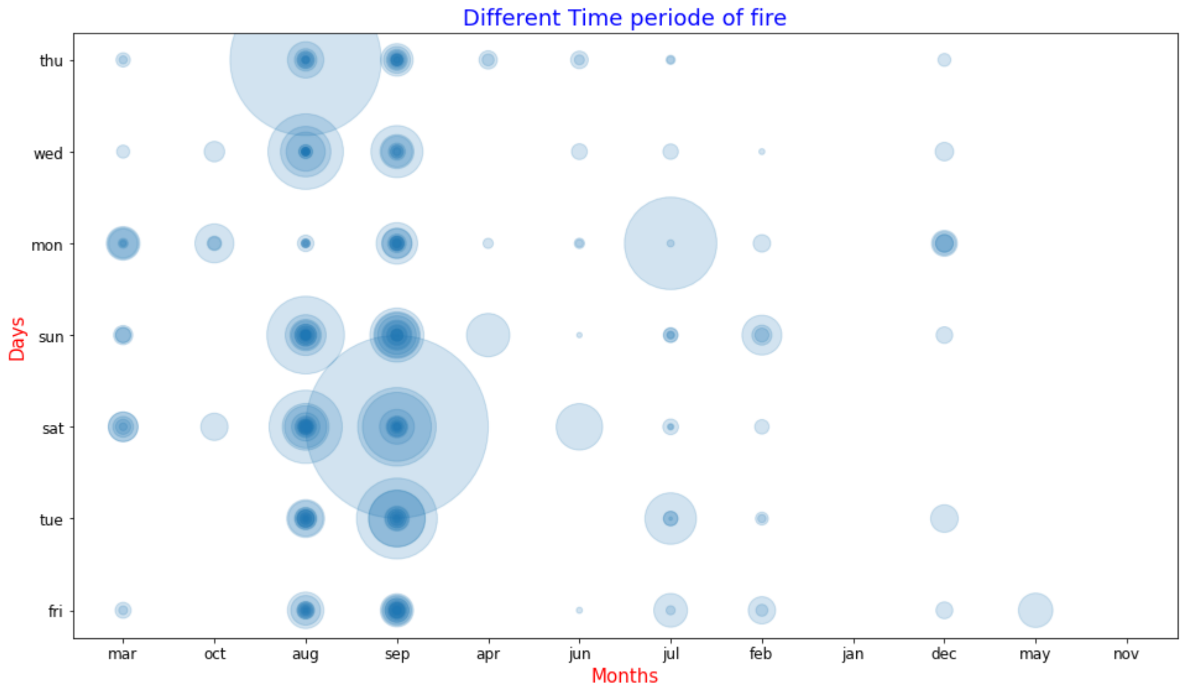


In [121]:

```
forest_fire.plot(kind='scatter', x='month', y='day', alpha=0.2, s=20*forest_fire['area'],fi
plt.xlabel('Months',color='red',fontsize=15)
plt.ylabel('Days',color='red',fontsize=15)
plt.title('Different Time periode of fire',color='blue',fontsize=18)
```

Out[121]:

```
Text(0.5, 1.0, 'Different Time periode of fire')
```



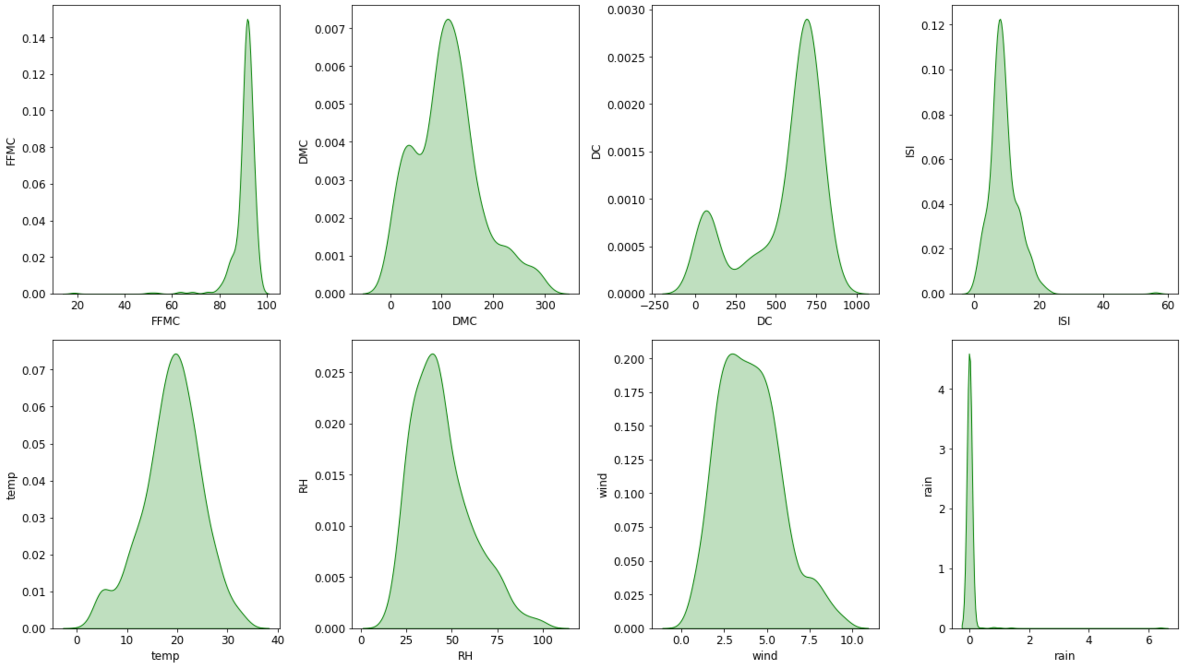
Majority of the fire accors in the month Aug and Sep || sunday and friday have recorded the most cases of fire

In [122]:

```
num_columns = dfa.select_dtypes(exclude='object')
```

In [123]:

```
plt.figure(figsize=(18,40))
for i,col in enumerate(num_columns,1):
    plt.subplot(8,4,i)
    sns.kdeplot(forest_fire[col],color='g',shade=True,legend=True)
    plt.ylabel(col)
plt.tight_layout()
plt.show()
```



In [124]:

```
pd.DataFrame(data=[num_columns.skew(),num_columns.kurtosis()],index=['skewness','kurtosis']
```

Out[124]:

	FFMC	DMC	DC	ISI	temp	RH	wind	rain
skewness	-6.575606	0.547498	-1.100445	2.536325	-0.331172	0.862904	0.571001	19.816344
kurtosis	67.066041	0.204822	-0.245244	21.458037	0.136166	0.438183	0.054324	421.295964

Finding Correlation

In [125]:

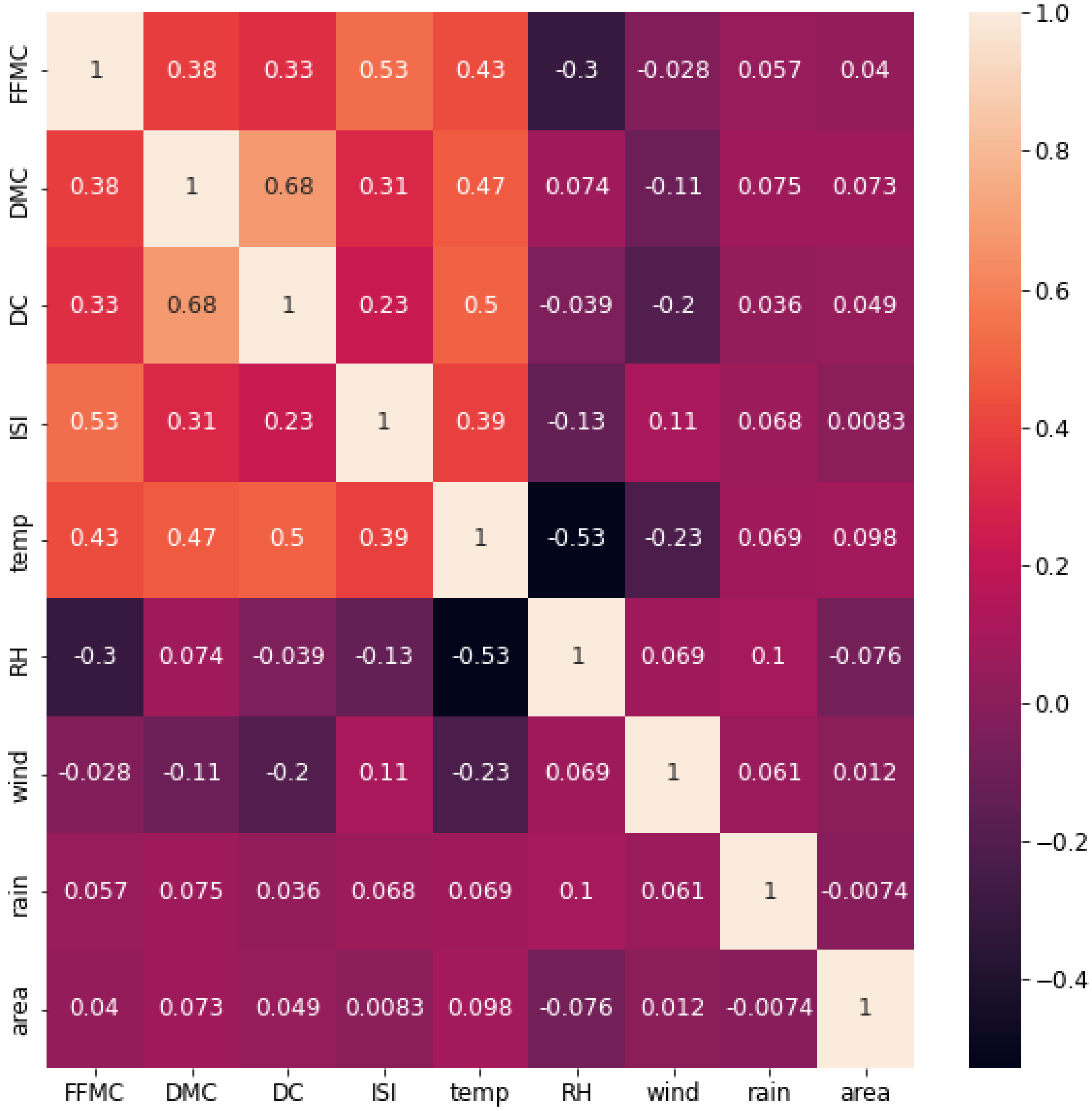
```
corr = forest_fire[forest_fire.columns[0:11]].corr()
```

In [126]:

```
plt.figure(figsize=(10,10))
sns.heatmap(corr,annot=True)
```

Out[126]:

```
<AxesSubplot:>
```



4. Model Building

In [127]:

```
mapping = {'small': 0, 'large': 1}
```

In [128]:

```
df1 = forest_fire.replace(mapping)
df1
```

Out[128]:

	month	day	FFMC	DMC	DC	ISI	temp	RH	wind	rain	...	monthfeb	monthjan	m
0	mar	fri	86.2	26.2	94.3	5.1	8.2	51	6.7	0.0	...	0	0	
1	oct	tue	90.6	35.4	669.1	6.7	18.0	33	0.9	0.0	...	0	0	
2	oct	sat	90.6	43.7	686.9	6.7	14.6	33	1.3	0.0	...	0	0	
3	mar	fri	91.7	33.3	77.5	9.0	8.3	97	4.0	0.2	...	0	0	
4	mar	sun	89.3	51.3	102.2	9.6	11.4	99	1.8	0.0	...	0	0	
...	...	...	...	...	...	...	...	...	...	...	...	...	...	
512	aug	sun	81.6	56.7	665.6	1.9	27.8	32	2.7	0.0	...	0	0	
513	aug	sun	81.6	56.7	665.6	1.9	21.9	71	5.8	0.0	...	0	0	
514	aug	sun	81.6	56.7	665.6	1.9	21.2	70	6.7	0.0	...	0	0	
515	aug	sat	94.4	146.0	614.7	11.3	25.6	42	4.0	0.0	...	0	0	
516	nov	tue	79.5	3.0	106.7	1.1	11.8	31	4.5	0.0	...	0	0	

517 rows × 31 columns

In [129]:

```
df1.drop(["month","day"],axis=1,inplace = True)
```

In [130]:

```
X = np.array(df1.iloc[:,0:28])
y = np.array(df1.iloc[:,28])
```

In [131]:

```
def norm_func(i):
    x = (i-i.min())/(i.max()-i.min())
    return (x)
```

In [132]:

```
X_norm = norm_func(X)
```

In [133]:

```
x_train,x_test,y_train,y_test= train_test_split(X_norm,y, test_size=0.2,stratify = y)
```

5. Create Neural Network Model

In [134]:

```
model = Sequential()
model.add(Dense(12, input_dim=28, activation='relu'))
model.add(Dense(8, activation='relu'))
model.add(Dense(1, activation='sigmoid'))
```

In [135]:

```
model.compile(loss='binary_crossentropy', optimizer='adam', metrics=['accuracy'])
```

Fit the model

In [146]:

```
model.fit(x_train, y_train, epochs=150,validation_split=0.33, batch_size=10,callbacks=[hist
```

```
Epoch 1/150
28/28 [==============================] - 0s 7ms/step - loss: 0.1501 - accura
cy: 0.9493 - val_loss: 0.1472 - val_accuracy: 0.9270
Epoch 2/150
28/28 [==============================] - 0s 5ms/step - loss: 0.1504 - accura
cy: 0.9457 - val_loss: 0.1410 - val_accuracy: 0.9489
Epoch 3/150
28/28 [==============================] - 0s 4ms/step - loss: 0.1475 - accura
cy: 0.9348 - val_loss: 0.1399 - val_accuracy: 0.9489
Epoch 4/150
28/28 [==============================] - 0s 5ms/step - loss: 0.1431 - accura
cy: 0.9529 - val_loss: 0.1402 - val_accuracy: 0.9343
Epoch 5/150
28/28 [==============================] - 0s 6ms/step - loss: 0.1437 - accura
cy: 0.9348 - val_loss: 0.1367 - val_accuracy: 0.9489
Epoch 6/150
28/28 [==============================] - 0s 5ms/step - loss: 0.1389 - accura
cy: 0.9493 - val_loss: 0.1358 - val_accuracy: 0.9489
Epoch 7/150
28/28 [==============================] - 0s 4ms/step - loss: 0.1395 - accura
cy: 0.9493 - val_loss: 0.1391 - val_accuracy: 0.9708
Epoch 8/150
28/28 [==============================] - 0s 4ms/step - loss: 0.1347 - accura
cy: 0.9493 - val_loss: 0.1345 - val_accuracy: 0.9489
Epoch 9/150
28/28 [==============================] - 0s 5ms/step - loss: 0.1357 - accura
cy: 0.9348 - val_loss: 0.1366 - val_accuracy: 0.9708
Epoch 10/150
28/28 [==============================] - 0s 5ms/step - loss: 0.1353 - accura
cy: 0.9565 - val_loss: 0.1290 - val_accuracy: 0.9489
Epoch 11/150
28/28 [==============================] - 0s 5ms/step - loss: 0.1333 - accura
cy: 0.9384 - val_loss: 0.1278 - val_accuracy: 0.9489
Epoch 12/150
28/28 [==============================] - 0s 4ms/step - loss: 0.1290 - accura
cy: 0.9601 - val_loss: 0.1264 - val_accuracy: 0.9489
Epoch 13/150
28/28 [==============================] - 0s 5ms/step - loss: 0.1268 - accura
cy: 0.9565 - val_loss: 0.1256 - val_accuracy: 0.9562
Epoch 14/150
28/28 [==============================] - 0s 4ms/step - loss: 0.1258 - accura
cy: 0.9601 - val_loss: 0.1236 - val_accuracy: 0.9489
Epoch 15/150
28/28 [==============================] - 0s 5ms/step - loss: 0.1290 - accura
cy: 0.9529 - val_loss: 0.1251 - val_accuracy: 0.9489
Epoch 16/150
28/28 [==============================] - 0s 5ms/step - loss: 0.1241 - accura
cy: 0.9638 - val_loss: 0.1209 - val_accuracy: 0.9489
Epoch 17/150
28/28 [==============================] - 0s 4ms/step - loss: 0.1224 - accura
cy: 0.9529 - val_loss: 0.1199 - val_accuracy: 0.9489
Epoch 18/150
28/28 [==============================] - 0s 4ms/step - loss: 0.1212 - accura
cy: 0.9529 - val_loss: 0.1186 - val_accuracy: 0.9562
Epoch 19/150
28/28 [==============================] - 0s 5ms/step - loss: 0.1202 - accura
cy: 0.9674 - val_loss: 0.1182 - val_accuracy: 0.9489
```

```
Epoch 20/150
28/28 [==============================] - 0s 5ms/step - loss: 0.1199 - accura
cy: 0.9565 - val_loss: 0.1253 - val_accuracy: 0.9781
Epoch 21/150
28/28 [==============================] - 0s 5ms/step - loss: 0.1202 - accura
cy: 0.9638 - val_loss: 0.1152 - val_accuracy: 0.9562
Epoch 22/150
28/28 [==============================] - 0s 5ms/step - loss: 0.1132 - accura
cy: 0.9819 - val_loss: 0.1219 - val_accuracy: 0.9489
Epoch 23/150
28/28 [==============================] - 0s 4ms/step - loss: 0.1214 - accura
cy: 0.9457 - val_loss: 0.1122 - val_accuracy: 0.9562
Epoch 24/150
28/28 [==============================] - 0s 4ms/step - loss: 0.1126 - accura
cy: 0.9601 - val_loss: 0.1111 - val_accuracy: 0.9562
Epoch 25/150
28/28 [==============================] - 0s 4ms/step - loss: 0.1115 - accura
cy: 0.9710 - val_loss: 0.1122 - val_accuracy: 0.9489
Epoch 26/150
28/28 [==============================] - 0s 5ms/step - loss: 0.1139 - accura
cy: 0.9783 - val_loss: 0.1128 - val_accuracy: 0.9489
Epoch 27/150
28/28 [==============================] - 0s 4ms/step - loss: 0.1081 - accura
cy: 0.9638 - val_loss: 0.1172 - val_accuracy: 0.9781
Epoch 28/150
28/28 [==============================] - 0s 3ms/step - loss: 0.1135 - accura
cy: 0.9529 - val_loss: 0.1114 - val_accuracy: 0.9781
Epoch 29/150
28/28 [==============================] - 0s 3ms/step - loss: 0.1073 - accura
cy: 0.9783 - val_loss: 0.1057 - val_accuracy: 0.9708
Epoch 30/150
28/28 [==============================] - 0s 3ms/step - loss: 0.1043 - accura
cy: 0.9601 - val_loss: 0.1050 - val_accuracy: 0.9708
Epoch 31/150
28/28 [==============================] - 0s 3ms/step - loss: 0.1027 - accura
cy: 0.9746 - val_loss: 0.1054 - val_accuracy: 0.9562
Epoch 32/150
28/28 [==============================] - 0s 3ms/step - loss: 0.1043 - accura
cy: 0.9638 - val_loss: 0.1097 - val_accuracy: 0.9781
Epoch 33/150
28/28 [==============================] - 0s 3ms/step - loss: 0.1031 - accura
cy: 0.9710 - val_loss: 0.1032 - val_accuracy: 0.9708
Epoch 34/150
28/28 [==============================] - 0s 3ms/step - loss: 0.1006 - accura
cy: 0.9746 - val_loss: 0.1006 - val_accuracy: 0.9708
Epoch 35/150
28/28 [==============================] - 0s 5ms/step - loss: 0.0977 - accura
cy: 0.9746 - val_loss: 0.1006 - val_accuracy: 0.9708
Epoch 36/150
28/28 [==============================] - 0s 4ms/step - loss: 0.0974 - accura
cy: 0.9783 - val_loss: 0.0985 - val_accuracy: 0.9708
Epoch 37/150
28/28 [==============================] - 0s 4ms/step - loss: 0.0964 - accura
cy: 0.9674 - val_loss: 0.0981 - val_accuracy: 0.9708
Epoch 38/150
28/28 [==============================] - 0s 5ms/step - loss: 0.0954 - accura
cy: 0.9783 - val_loss: 0.0971 - val_accuracy: 0.9708
Epoch 39/150
28/28 [==============================] - 0s 5ms/step - loss: 0.0939 - accura
cy: 0.9746 - val_loss: 0.0996 - val_accuracy: 0.9562
Epoch 40/150
```

```
28/28 [==============================] - 0s 4ms/step - loss: 0.0935 - accura
cy: 0.9638 - val_loss: 0.0955 - val_accuracy: 0.9708
Epoch 41/150
28/28 [==============================] - 0s 5ms/step - loss: 0.0919 - accura
cy: 0.9746 - val_loss: 0.0940 - val_accuracy: 0.9708
Epoch 42/150
28/28 [==============================] - 0s 3ms/step - loss: 0.0919 - accura
cy: 0.9819 - val_loss: 0.0964 - val_accuracy: 0.9562
Epoch 43/150
28/28 [==============================] - 0s 3ms/step - loss: 0.0921 - accura
cy: 0.9746 - val_loss: 0.0931 - val_accuracy: 0.9708
Epoch 44/150
28/28 [==============================] - 0s 5ms/step - loss: 0.0967 - accura
cy: 0.9674 - val_loss: 0.1011 - val_accuracy: 0.9708
Epoch 45/150
28/28 [==============================] - 0s 5ms/step - loss: 0.0914 - accura
cy: 0.9783 - val_loss: 0.0911 - val_accuracy: 0.9781
Epoch 46/150
28/28 [==============================] - 0s 4ms/step - loss: 0.0891 - accura
cy: 0.9819 - val_loss: 0.0892 - val_accuracy: 0.9708
Epoch 47/150
28/28 [==============================] - 0s 5ms/step - loss: 0.0881 - accura
cy: 0.9746 - val_loss: 0.0886 - val_accuracy: 0.9708
Epoch 48/150
28/28 [==============================] - 0s 4ms/step - loss: 0.0850 - accura
cy: 0.9819 - val_loss: 0.0884 - val_accuracy: 0.9708
Epoch 49/150
28/28 [==============================] - 0s 5ms/step - loss: 0.0855 - accura
cy: 0.9710 - val_loss: 0.0876 - val_accuracy: 0.9781
Epoch 50/150
28/28 [==============================] - 0s 4ms/step - loss: 0.0833 - accura
cy: 0.9819 - val_loss: 0.0869 - val_accuracy: 0.9708
Epoch 51/150
28/28 [==============================] - 0s 5ms/step - loss: 0.0836 - accura
cy: 0.9783 - val_loss: 0.0914 - val_accuracy: 0.9562
Epoch 52/150
28/28 [==============================] - 0s 4ms/step - loss: 0.0809 - accura
cy: 0.9746 - val_loss: 0.0875 - val_accuracy: 0.9781
Epoch 53/150
28/28 [==============================] - 0s 5ms/step - loss: 0.0820 - accura
cy: 0.9891 - val_loss: 0.0858 - val_accuracy: 0.9708
Epoch 54/150
28/28 [==============================] - 0s 5ms/step - loss: 0.0803 - accura
cy: 0.9819 - val_loss: 0.0834 - val_accuracy: 0.9781
Epoch 55/150
28/28 [==============================] - 0s 5ms/step - loss: 0.0794 - accura
cy: 0.9819 - val_loss: 0.0832 - val_accuracy: 0.9781
Epoch 56/150
28/28 [==============================] - 0s 4ms/step - loss: 0.0796 - accura
cy: 0.9783 - val_loss: 0.0849 - val_accuracy: 0.9781
Epoch 57/150
28/28 [==============================] - 0s 4ms/step - loss: 0.0790 - accura
cy: 0.9891 - val_loss: 0.0816 - val_accuracy: 0.9708
Epoch 58/150
28/28 [==============================] - 0s 5ms/step - loss: 0.0756 - accura
cy: 0.9855 - val_loss: 0.0827 - val_accuracy: 0.9708
```

```
Epoch 59/150
28/28 [==============================] - 0s 4ms/step - loss: 0.0764 - accu
racy: 0.9819 - val_loss: 0.0810 - val_accuracy: 0.9708
```

```
Epoch 60/150
28/28 [==============================] - 0s 5ms/step - loss: 0.0759 - accu
racy: 0.9855 - val_loss: 0.0795 - val_accuracy: 0.9708
Epoch 61/150
28/28 [==============================] - 0s 5ms/step - loss: 0.0741 - accu
racy: 0.9891 - val_loss: 0.0845 - val_accuracy: 0.9635
Epoch 62/150
28/28 [==============================] - 0s 5ms/step - loss: 0.0782 - accu
racy: 0.9746 - val_loss: 0.0780 - val_accuracy: 0.9708
Epoch 63/150
28/28 [==============================] - 0s 5ms/step - loss: 0.0722 - accu
racy: 0.9855 - val_loss: 0.0774 - val_accuracy: 0.9781
Epoch 64/150
28/28 [==============================] - 0s 5ms/step - loss: 0.0723 - accu
racy: 0.9855 - val_loss: 0.0776 - val_accuracy: 0.9781
Epoch 65/150
28/28 [==============================] - 0s 5ms/step - loss: 0.0770 - accu
racy: 0.9746 - val_loss: 0.0775 - val_accuracy: 0.9781
Epoch 66/150
28/28 [==============================] - 0s 5ms/step - loss: 0.0773 - accu
racy: 0.9819 - val_loss: 0.0798 - val_accuracy: 0.9708
Epoch 67/150
28/28 [==============================] - 0s 5ms/step - loss: 0.0711 - accu
racy: 0.9964 - val_loss: 0.0793 - val_accuracy: 0.9708
Epoch 68/150
28/28 [==============================] - 0s 5ms/step - loss: 0.0719 - accu
racy: 0.9891 - val_loss: 0.0782 - val_accuracy: 0.9708
Epoch 69/150
28/28 [==============================] - 0s 5ms/step - loss: 0.0700 - accu
racy: 0.9891 - val_loss: 0.0774 - val_accuracy: 0.9708
Epoch 70/150
28/28 [==============================] - 0s 5ms/step - loss: 0.0685 - accu
racy: 0.9855 - val_loss: 0.0729 - val_accuracy: 0.9781
Epoch 71/150
28/28 [==============================] - 0s 4ms/step - loss: 0.0693 - accu
racy: 0.9891 - val_loss: 0.0739 - val_accuracy: 0.9708
Epoch 72/150
28/28 [==============================] - 0s 4ms/step - loss: 0.0666 - accu
racy: 0.9855 - val_loss: 0.0721 - val_accuracy: 0.9781
Epoch 73/150
28/28 [==============================] - 0s 5ms/step - loss: 0.0666 - accu
racy: 0.9855 - val_loss: 0.0743 - val_accuracy: 0.9708
Epoch 74/150
28/28 [==============================] - 0s 4ms/step - loss: 0.0665 - accu
racy: 0.9855 - val_loss: 0.0713 - val_accuracy: 0.9781
Epoch 75/150
28/28 [==============================] - 0s 4ms/step - loss: 0.0646 - accu
racy: 0.9928 - val_loss: 0.0757 - val_accuracy: 0.9708
Epoch 76/150
28/28 [==============================] - 0s 4ms/step - loss: 0.0659 - accu
racy: 0.9891 - val_loss: 0.0702 - val_accuracy: 0.9781
Epoch 77/150
28/28 [==============================] - 0s 4ms/step - loss: 0.0645 - accu
racy: 0.9855 - val_loss: 0.0694 - val_accuracy: 0.9781
Epoch 78/150
28/28 [==============================] - 0s 4ms/step - loss: 0.0623 - accu
racy: 0.9891 - val_loss: 0.0702 - val_accuracy: 0.9708
Epoch 79/150
28/28 [==============================] - 0s 5ms/step - loss: 0.0645 - accu
racy: 0.9891 - val_loss: 0.0719 - val_accuracy: 0.9708
Epoch 80/150
```

```
28/28 [==============================] - 0s 5ms/step - loss: 0.0641 - accu
racy: 0.9891 - val_loss: 0.0708 - val_accuracy: 0.9708
Epoch 81/150
28/28 [==============================] - 0s 5ms/step - loss: 0.0610 - accu
racy: 0.9891 - val_loss: 0.0698 - val_accuracy: 0.9781
Epoch 82/150
28/28 [==============================] - 0s 4ms/step - loss: 0.0660 - accu
racy: 0.9783 - val_loss: 0.0693 - val_accuracy: 0.9781
Epoch 83/150
28/28 [==============================] - 0s 5ms/step - loss: 0.0625 - accu
racy: 0.9891 - val_loss: 0.0686 - val_accuracy: 0.9708
Epoch 84/150
28/28 [==============================] - 0s 5ms/step - loss: 0.0595 - accu
racy: 0.9964 - val_loss: 0.0661 - val_accuracy: 0.9781
Epoch 85/150
28/28 [==============================] - 0s 5ms/step - loss: 0.0581 - accu
racy: 0.9891 - val_loss: 0.0668 - val_accuracy: 0.9781
Epoch 86/150
28/28 [==============================] - 0s 5ms/step - loss: 0.0618 - accu
racy: 0.9928 - val_loss: 0.0653 - val_accuracy: 0.9781
Epoch 87/150
28/28 [==============================] - 0s 5ms/step - loss: 0.0590 - accu
racy: 0.9928 - val_loss: 0.0663 - val_accuracy: 0.9781
Epoch 88/150
28/28 [==============================] - 0s 4ms/step - loss: 0.0588 - accu
racy: 0.9891 - val_loss: 0.0655 - val_accuracy: 0.9781
Epoch 89/150
28/28 [==============================] - 0s 5ms/step - loss: 0.0595 - accu
racy: 0.9891 - val_loss: 0.0666 - val_accuracy: 0.9708
Epoch 90/150
28/28 [==============================] - 0s 4ms/step - loss: 0.0577 - accu
racy: 0.9964 - val_loss: 0.0648 - val_accuracy: 0.9781
Epoch 91/150
28/28 [==============================] - 0s 3ms/step - loss: 0.0554 - accu
racy: 0.9928 - val_loss: 0.0649 - val_accuracy: 0.9781
Epoch 92/150
28/28 [==============================] - 0s 5ms/step - loss: 0.0576 - accu
racy: 0.9928 - val_loss: 0.0643 - val_accuracy: 0.9781
Epoch 93/150
28/28 [==============================] - 0s 5ms/step - loss: 0.0566 - accu
racy: 0.9819 - val_loss: 0.0638 - val_accuracy: 0.9781
Epoch 94/150
28/28 [==============================] - 0s 5ms/step - loss: 0.0540 - accu
racy: 0.9964 - val_loss: 0.0625 - val_accuracy: 0.9781
Epoch 95/150
28/28 [==============================] - 0s 5ms/step - loss: 0.0561 - accu
racy: 0.9891 - val_loss: 0.0627 - val_accuracy: 0.9781
Epoch 96/150
28/28 [==============================] - 0s 5ms/step - loss: 0.0558 - accu
racy: 0.9928 - val_loss: 0.0646 - val_accuracy: 0.9708
Epoch 97/150
28/28 [==============================] - 0s 5ms/step - loss: 0.0577 - accu
racy: 0.9891 - val_loss: 0.0681 - val_accuracy: 0.9781
Epoch 98/150
28/28 [==============================] - 0s 5ms/step - loss: 0.0548 - accu
racy: 0.9928 - val_loss: 0.0611 - val_accuracy: 0.9781
Epoch 99/150
28/28 [==============================] - 0s 5ms/step - loss: 0.0539 - accu
racy: 0.9928 - val_loss: 0.0651 - val_accuracy: 0.9854
Epoch 100/150
28/28 [==============================] - 0s 5ms/step - loss: 0.0541 - accu
```

```
racy: 0.9891 - val_loss: 0.0600 - val_accuracy: 0.9781
Epoch 101/150
28/28 [==============================] - 0s 5ms/step - loss: 0.0533 - accu
racy: 0.9928 - val_loss: 0.0625 - val_accuracy: 0.9708
Epoch 102/150
28/28 [==============================] - 0s 5ms/step - loss: 0.0551 - accu
racy: 0.9928 - val_loss: 0.0663 - val_accuracy: 0.9708
Epoch 103/150
28/28 [==============================] - 0s 5ms/step - loss: 0.0512 - accu
racy: 1.0000 - val_loss: 0.0592 - val_accuracy: 0.9781
Epoch 104/150
28/28 [==============================] - 0s 4ms/step - loss: 0.0512 - accu
racy: 0.9891 - val_loss: 0.0614 - val_accuracy: 0.9708
Epoch 105/150
28/28 [==============================] - 0s 5ms/step - loss: 0.0491 - accu
racy: 0.9928 - val_loss: 0.0627 - val_accuracy: 0.9708
Epoch 106/150
28/28 [==============================] - 0s 5ms/step - loss: 0.0531 - accu
racy: 0.9891 - val_loss: 0.0611 - val_accuracy: 0.9781
Epoch 107/150
28/28 [==============================] - 0s 5ms/step - loss: 0.0499 - accu
racy: 0.9964 - val_loss: 0.0587 - val_accuracy: 0.9781
Epoch 108/150
28/28 [==============================] - 0s 5ms/step - loss: 0.0499 - accu
racy: 0.9928 - val_loss: 0.0594 - val_accuracy: 0.9781
Epoch 109/150
28/28 [==============================] - 0s 5ms/step - loss: 0.0522 - accu
racy: 0.9928 - val_loss: 0.0605 - val_accuracy: 0.9781
Epoch 110/150
28/28 [==============================] - 0s 5ms/step - loss: 0.0500 - accu
racy: 0.9964 - val_loss: 0.0579 - val_accuracy: 0.9781
Epoch 111/150
28/28 [==============================] - 0s 4ms/step - loss: 0.0489 - accu
racy: 0.9891 - val_loss: 0.0598 - val_accuracy: 0.9781
Epoch 112/150
28/28 [==============================] - 0s 5ms/step - loss: 0.0522 - accu
racy: 0.9964 - val_loss: 0.0564 - val_accuracy: 0.9781
Epoch 113/150
28/28 [==============================] - 0s 5ms/step - loss: 0.0500 - accu
racy: 0.9928 - val_loss: 0.0582 - val_accuracy: 0.9708
Epoch 114/150
28/28 [==============================] - 0s 4ms/step - loss: 0.0497 - accu
racy: 0.9964 - val_loss: 0.0558 - val_accuracy: 0.9781
Epoch 115/150
28/28 [==============================] - 0s 4ms/step - loss: 0.0486 - accu
racy: 0.9855 - val_loss: 0.0598 - val_accuracy: 0.9854
Epoch 116/150
28/28 [==============================] - 0s 5ms/step - loss: 0.0479 - accu
racy: 0.9928 - val_loss: 0.0562 - val_accuracy: 0.9708
Epoch 117/150
28/28 [==============================] - 0s 4ms/step - loss: 0.0478 - accu
racy: 1.0000 - val_loss: 0.0558 - val_accuracy: 0.9781
Epoch 118/150
28/28 [==============================] - 0s 5ms/step - loss: 0.0450 - accu
racy: 0.9964 - val_loss: 0.0552 - val_accuracy: 0.9781
Epoch 119/150
28/28 [==============================] - 0s 5ms/step - loss: 0.0454 - accu
racy: 1.0000 - val_loss: 0.0548 - val_accuracy: 0.9781
Epoch 120/150
28/28 [==============================] - 0s 5ms/step - loss: 0.0460 - accu
racy: 0.9928 - val_loss: 0.0554 - val_accuracy: 0.9708
```

```
Epoch 121/150
28/28 [==============================] - 0s 5ms/step - loss: 0.0443 - accu
racy: 0.9964 - val_loss: 0.0544 - val_accuracy: 0.9781
Epoch 122/150
28/28 [==============================] - 0s 5ms/step - loss: 0.0466 - accu
racy: 0.9964 - val_loss: 0.0596 - val_accuracy: 0.9781
Epoch 123/150
28/28 [==============================] - 0s 5ms/step - loss: 0.0503 - accu
racy: 0.9891 - val_loss: 0.0555 - val_accuracy: 0.9781
Epoch 124/150
28/28 [==============================] - 0s 5ms/step - loss: 0.0435 - accu
racy: 0.9964 - val_loss: 0.0535 - val_accuracy: 0.9781
Epoch 125/150
28/28 [==============================] - 0s 5ms/step - loss: 0.0437 - accu
racy: 0.9964 - val_loss: 0.0534 - val_accuracy: 0.9781
Epoch 126/150
28/28 [==============================] - 0s 5ms/step - loss: 0.0450 - accu
racy: 0.9964 - val_loss: 0.0608 - val_accuracy: 0.9708
Epoch 127/150
28/28 [==============================] - 0s 5ms/step - loss: 0.0542 - accu
racy: 0.9819 - val_loss: 0.0549 - val_accuracy: 0.9781
Epoch 128/150
28/28 [==============================] - 0s 5ms/step - loss: 0.0453 - accu
racy: 0.9928 - val_loss: 0.0526 - val_accuracy: 0.9781
Epoch 129/150
28/28 [==============================] - 0s 5ms/step - loss: 0.0481 - accu
racy: 0.9928 - val_loss: 0.0524 - val_accuracy: 0.9781
Epoch 130/150
28/28 [==============================] - 0s 5ms/step - loss: 0.0462 - accu
racy: 0.9964 - val_loss: 0.0561 - val_accuracy: 0.9781
Epoch 131/150
28/28 [==============================] - 0s 5ms/step - loss: 0.0425 - accu
racy: 0.9964 - val_loss: 0.0558 - val_accuracy: 0.9781
Epoch 132/150
28/28 [==============================] - 0s 5ms/step - loss: 0.0418 - accu
racy: 0.9928 - val_loss: 0.0554 - val_accuracy: 0.9781
Epoch 133/150
28/28 [==============================] - 0s 5ms/step - loss: 0.0426 - accu
racy: 0.9928 - val_loss: 0.0521 - val_accuracy: 0.9781
Epoch 134/150
28/28 [==============================] - 0s 5ms/step - loss: 0.0411 - accu
racy: 1.0000 - val_loss: 0.0524 - val_accuracy: 0.9781
Epoch 135/150
28/28 [==============================] - 0s 5ms/step - loss: 0.0413 - accu
racy: 1.0000 - val_loss: 0.0538 - val_accuracy: 0.9781
Epoch 136/150
28/28 [==============================] - 0s 5ms/step - loss: 0.0411 - accu
racy: 0.9928 - val_loss: 0.0568 - val_accuracy: 0.9781
Epoch 137/150
28/28 [==============================] - 0s 6ms/step - loss: 0.0418 - accu
racy: 0.9928 - val_loss: 0.0508 - val_accuracy: 0.9781
Epoch 138/150
28/28 [==============================] - 0s 5ms/step - loss: 0.0395 - accu
racy: 0.9964 - val_loss: 0.0539 - val_accuracy: 0.9781
Epoch 139/150
28/28 [==============================] - 0s 5ms/step - loss: 0.0389 - accu
racy: 0.9964 - val_loss: 0.0509 - val_accuracy: 0.9781
Epoch 140/150
28/28 [==============================] - 0s 4ms/step - loss: 0.0391 - accu
racy: 0.9928 - val_loss: 0.0503 - val_accuracy: 0.9781
Epoch 141/150
```

```
28/28 [==============================] - 0s 5ms/step - loss: 0.0385 - accu
racy: 0.9928 - val_loss: 0.0608 - val_accuracy: 0.9708
Epoch 142/150
28/28 [==============================] - 0s 5ms/step - loss: 0.0423 - accu
racy: 0.9891 - val_loss: 0.0525 - val_accuracy: 0.9781
Epoch 143/150
28/28 [==============================] - 0s 5ms/step - loss: 0.0405 - accu
racy: 0.9964 - val_loss: 0.0506 - val_accuracy: 0.9781
Epoch 144/150
28/28 [==============================] - 0s 6ms/step - loss: 0.0397 - accu
racy: 1.0000 - val_loss: 0.0522 - val_accuracy: 0.9781
Epoch 145/150
28/28 [==============================] - 0s 5ms/step - loss: 0.0390 - accu
racy: 0.9964 - val_loss: 0.0501 - val_accuracy: 0.9781
Epoch 146/150
28/28 [==============================] - 0s 5ms/step - loss: 0.0414 - accu
racy: 0.9928 - val_loss: 0.0489 - val_accuracy: 0.9781
Epoch 147/150
28/28 [==============================] - 0s 5ms/step - loss: 0.0384 - accu
racy: 0.9964 - val_loss: 0.0563 - val_accuracy: 0.9708
Epoch 148/150
28/28 [==============================] - 0s 5ms/step - loss: 0.0359 - accu
racy: 1.0000 - val_loss: 0.0510 - val_accuracy: 0.9781
Epoch 149/150
28/28 [==============================] - 0s 5ms/step - loss: 0.0374 - accu
racy: 0.9964 - val_loss: 0.0516 - val_accuracy: 0.9781
Epoch 150/150
28/28 [==============================] - 0s 5ms/step - loss: 0.0387 - accu
racy: 0.9964 - val_loss: 0.0484 - val_accuracy: 0.9781
```

Out[146]:

```
<keras.callbacks.History at 0x24b1a2f9220>
```

evaluate the model

In [147]:

```python
scores = model.evaluate(x_train, y_train)
print("%s: %.2f%%" % (model.metrics_names[1], scores[1]*100))
```

```
13/13 [==============================] - 0s 2ms/step - loss: 0.0397 - accura
cy: 0.9927
accuracy: 99.27%
```

In [148]:

```python
model.history.history.keys()
```

Out[148]:

```
dict_keys([])
```

In [149]:

```python
hist= print(history.history.keys())
```

```
dict_keys(['loss', 'accuracy', 'val_loss', 'val_accuracy'])
```

summarize history for accuracy

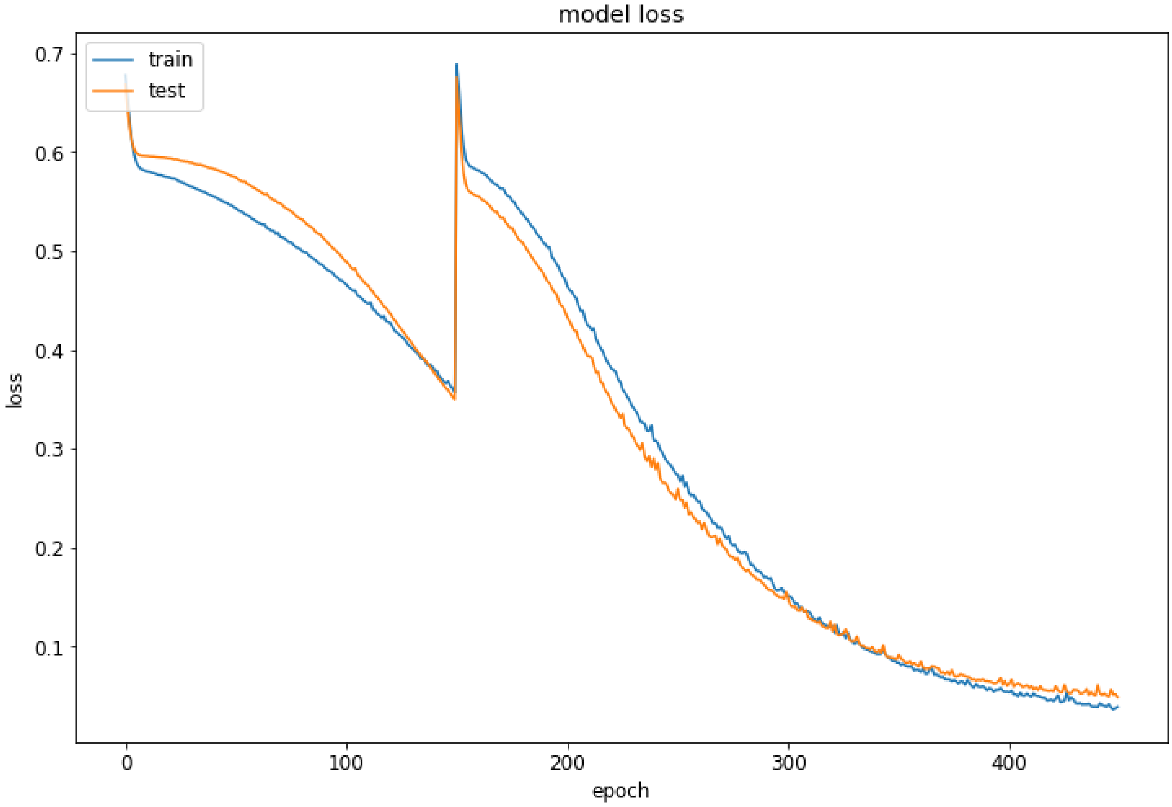
In [150]:

```
plt.figure(figsize=(12,8))
plt.rcParams['font.size'] = 12
plt.plot(history.history['accuracy'])
plt.plot(history.history['val_accuracy'])
plt.title('model accuracy')
plt.ylabel('accuracy')
plt.xlabel('epoch')
plt.legend(['train', 'test'], loc='upper left')
plt.show()
```

summarize history for loss

In [151]:

```
plt.figure(figsize=(12,8))
plt.plot(history.history['loss'])
plt.plot(history.history['val_loss'])
plt.title('model loss')
plt.ylabel('loss')
plt.xlabel('epoch')
plt.legend(['train', 'test'], loc='upper left')
plt.show()
```



Predicting values from Model using same dataset

In [152]:

```
def probToBinary(varProb):
    if varProb >= 0.5:
        return 1
    else:
        return 0
```

generating predictions for test data & creating table with test price & predicted price for test

In [156]:

```
y_predict_test = model.predict(x_test)


test_prediction = pd.DataFrame()
test_prediction['Test_Actual'] = y_test
test_prediction['Test_Probability'] = y_predict_test
test_prediction['Test_Predicted'] = test_prediction['Test_Probability'].apply(probToBinary)
print(test_prediction.shape)
test_prediction.head(10)
```

```
4/4 [==============================] - 0s 2ms/step
(104, 3)
```

Out[156]:

	Test_Actual	Test_Probability	Test_Predicted
0	1	0.899437	1
1	0	0.003865	0
2	1	0.981555	1
3	0	0.002311	0
4	0	0.609429	1
5	0	0.034691	0
6	0	0.004133	0
7	0	0.004449	0
8	0	0.001223	0
9	0	0.005232	0

generating predictions for train data & creating table with test price & predicted price for test

In [157]:

```
y_predict_train = model.predict(x_train)

train_prediction = pd.DataFrame()
train_prediction['Train_Actual'] = y_train
train_prediction['Train_Probability'] = y_predict_train
train_prediction['Train_Predicted'] = train_prediction['Train_Probability'].apply(probToBin
print(train_prediction.shape)
train_prediction.head(10)
```

```
13/13 [==============================] - 0s 2ms/step
(413, 3)
```

Out[157]:

	Train_Actual	Train_Probability	Train_Predicted
0	0	0.007990	0
1	1	0.999993	1
2	1	0.999912	1
3	1	0.999675	1
4	0	0.005066	0
5	0	0.001392	0
6	1	1.000000	1
7	0	0.178846	0
8	0	0.147317	0
9	1	1.000000	1

Predicting the probabilities of Forest Burned Area

END

In [ ]:

```

```