

```
In [2]: import pandas as pd
        from sklearn.model_selection import train_test_split
        from keras.models import Sequential
        from keras.layers import Activation,Dense
```

```
In [3]: import pandas as pd
        data = pd.read_csv(r"C:\Users\manju\Desktop\heart dataset.csv")
        data
```

```
Out[3]:
```

	age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	thal	target
0	52	1	0	125	212	0	1	168	0	1.0	2	2	3	0
1	53	1	0	140	203	1	0	155	1	3.1	0	0	3	0
2	70	1	0	145	174	0	1	125	1	2.6	0	0	3	0
3	61	1	0	148	203	0	1	161	0	0.0	2	1	3	0
4	62	0	0	138	294	1	1	106	0	1.9	1	3	2	0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
1020	59	1	1	140	221	0	1	164	1	0.0	2	0	2	1
1021	60	1	0	125	258	0	0	141	1	2.8	1	1	3	0
1022	47	1	0	110	275	0	0	118	1	1.0	1	1	2	0
1023	50	0	0	110	254	0	0	159	0	0.0	2	0	2	1
1024	54	1	0	120	188	0	1	113	0	1.4	1	1	3	0

1025 rows × 14 columns

```
In [4]: data.head()
```

```
Out[4]:
```

	age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	thal	target
<b>0</b>	52	1	0	125	212	0	1	168	0	1.0	2	2	3	0
<b>1</b>	53	1	0	140	203	1	0	155	1	3.1	0	0	3	0
<b>2</b>	70	1	0	145	174	0	1	125	1	2.6	0	0	3	0
<b>3</b>	61	1	0	148	203	0	1	161	0	0.0	2	1	3	0
<b>4</b>	62	0	0	138	294	1	1	106	0	1.9	1	3	2	0

```
In [28]: x = data.drop(columns=['target'])
         y = data['target']
```

```
In [22]: #data splitting
```

```
In [23]: from sklearn.model_selection import train_test_split
         x_train, x_test, y_train, y_test = train_test_split(x, y, test_size=0.2, random_state=42)
```

```
In [24]: from keras.models import Sequential
         from keras.layers import Dense
         model = Sequential()
         model.add(Dense(32, activation='relu', input_shape=(x_train.shape[1],)))
         model.add(Dense(16, activation='relu'))
         model.add(Dense(1, activation='sigmoid'))
```

```
In [32]: #comiling the model
         model.compile(optimizer='adam', loss = 'binary_crossentropy', metrics=['accuracy'])
```

```
In [34]: #model fit
         model.fit(x_train, y_train, epochs=30, batch_size=32, validation_split=0.2)
```

Epoch 1/30  
21/21 [=====] - 1s 22ms/step - loss: 0.5424 - accuracy: 0.7119 - val\_loss: 0.5882 - val\_accuracy: 0.7073

Epoch 2/30  
21/21 [=====] - 0s 15ms/step - loss: 0.5519 - accuracy: 0.7271 - val\_loss: 0.5791 - val\_accuracy: 0.6890

Epoch 3/30  
21/21 [=====] - 0s 8ms/step - loss: 0.5482 - accuracy: 0.7210 - val\_loss: 0.5768 - val\_accuracy: 0.6829

Epoch 4/30  
21/21 [=====] - 0s 8ms/step - loss: 0.5248 - accuracy: 0.7302 - val\_loss: 0.5869 - val\_accuracy: 0.6890

Epoch 5/30  
21/21 [=====] - 0s 8ms/step - loss: 0.4888 - accuracy: 0.7546 - val\_loss: 0.5881 - val\_accuracy: 0.7073

Epoch 6/30  
21/21 [=====] - 0s 8ms/step - loss: 0.5586 - accuracy: 0.7134 - val\_loss: 0.6780 - val\_accuracy: 0.6463

Epoch 7/30  
21/21 [=====] - 0s 8ms/step - loss: 0.5201 - accuracy: 0.7378 - val\_loss: 0.5769 - val\_accuracy: 0.7073

Epoch 8/30  
21/21 [=====] - 0s 10ms/step - loss: 0.4972 - accuracy: 0.7393 - val\_loss: 0.5662 - val\_accuracy: 0.7256

Epoch 9/30  
21/21 [=====] - 0s 16ms/step - loss: 0.4577 - accuracy: 0.7896 - val\_loss: 0.5458 - val\_accuracy: 0.6829

Epoch 10/30  
21/21 [=====] - 0s 8ms/step - loss: 0.4471 - accuracy: 0.7835 - val\_loss: 0.5512 - val\_accuracy: 0.7073

Epoch 11/30  
21/21 [=====] - 0s 8ms/step - loss: 0.4656 - accuracy: 0.7729 - val\_loss: 0.5676 - val\_accuracy: 0.7073

Epoch 12/30  
21/21 [=====] - 0s 9ms/step - loss: 0.4603 - accuracy: 0.7942 - val\_loss: 0.5456 - val\_accuracy: 0.7134

Epoch 13/30  
21/21 [=====] - 0s 9ms/step - loss: 0.4471 - accuracy: 0.7698 - val\_loss: 0.5288 - val\_accuracy: 0.7195

Epoch 14/30  
21/21 [=====] - 0s 8ms/step - loss: 0.4281 - accuracy: 0.8034 - val\_loss: 0.5294 - val\_accuracy: 0.7317

Epoch 15/30  
21/21 [=====] - 0s 12ms/step - loss: 0.4271 - accuracy: 0.8125 - val\_loss: 0.6107 - val\_accuracy: 0.7195

Epoch 16/30  
21/21 [=====] - 0s 9ms/step - loss: 0.4614 - accuracy: 0.7851 - val\_loss: 0.5508 - val\_accuracy: 0.7195

Epoch 17/30  
21/21 [=====] - 0s 13ms/step - loss: 0.4313 - accuracy: 0.8018 - val\_loss: 0.5079 - val\_accuracy: 0.7561

Epoch 18/30  
21/21 [=====] - 0s 10ms/step - loss: 0.4000 - accuracy: 0.8216 - val\_loss: 0.5142 - val\_accuracy: 0.7378

Epoch 19/30

```

21/21 [=====] - 0s 8ms/step - loss: 0.3947 - accuracy: 0.8216 - val_loss: 0.5064 - val_accuracy: 0.7744
Epoch 20/30
21/21 [=====] - 0s 8ms/step - loss: 0.3987 - accuracy: 0.8338 - val_loss: 0.5105 - val_accuracy: 0.7500
Epoch 21/30
21/21 [=====] - 0s 9ms/step - loss: 0.3917 - accuracy: 0.8293 - val_loss: 0.5021 - val_accuracy: 0.7683
Epoch 22/30
21/21 [=====] - 0s 10ms/step - loss: 0.3865 - accuracy: 0.8354 - val_loss: 0.5032 - val_accuracy: 0.743
9
Epoch 23/30
21/21 [=====] - 0s 8ms/step - loss: 0.3823 - accuracy: 0.8262 - val_loss: 0.5087 - val_accuracy: 0.7317
Epoch 24/30
21/21 [=====] - 0s 11ms/step - loss: 0.3872 - accuracy: 0.8323 - val_loss: 0.5916 - val_accuracy: 0.689
0
Epoch 25/30
21/21 [=====] - 0s 12ms/step - loss: 0.3906 - accuracy: 0.8247 - val_loss: 0.4886 - val_accuracy: 0.768
3
Epoch 26/30
21/21 [=====] - 0s 9ms/step - loss: 0.3710 - accuracy: 0.8399 - val_loss: 0.4916 - val_accuracy: 0.7561
Epoch 27/30
21/21 [=====] - 0s 8ms/step - loss: 0.3655 - accuracy: 0.8445 - val_loss: 0.5206 - val_accuracy: 0.7195
Epoch 28/30
21/21 [=====] - 0s 7ms/step - loss: 0.3700 - accuracy: 0.8415 - val_loss: 0.4839 - val_accuracy: 0.7744
Epoch 29/30
21/21 [=====] - 0s 8ms/step - loss: 0.3671 - accuracy: 0.8384 - val_loss: 0.4864 - val_accuracy: 0.7561
Epoch 30/30
21/21 [=====] - 0s 8ms/step - loss: 0.3750 - accuracy: 0.8430 - val_loss: 0.4901 - val_accuracy: 0.7805
Out[34]: <keras.src.callbacks.History at 0x1814e1d1bd0>

```

```

In [35]: test_loss, test_acc = model.evaluate(x_test,y_test)
print('Test Accuracy:',test_acc)

```

```

7/7 [=====] - 0s 4ms/step - loss: 0.4696 - accuracy: 0.7561
Test Accuracy: 0.7560975551605225

```

```

In [ ]:

```