

```
In [5]: 1 import pandas as pd
2 from sklearn.model_selection import train_test_split
3 from keras.models import Sequential
4 from keras.layers import Activation, Dense
```

```
In [6]: 1 import pandas as pd
2 data = pd.read_csv(r"C:\Users\HARISH BJ\Downloads\diabetes.csv")
```

```
In [7]: 1 data
```

```
Out[7]:
```

	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeFunction	Age	Outcome
0	6	148	72	35	0	33.6	0.627	50	1
1	1	85	66	29	0	26.6	0.351	31	0
2	8	183	64	0	0	23.3	0.672	32	1
3	1	89	66	23	94	28.1	0.167	21	0
4	0	137	40	35	168	43.1	2.288	33	1
...
763	10	101	76	48	180	32.9	0.171	63	0
764	2	122	70	27	0	36.8	0.340	27	0
765	5	121	72	23	112	26.2	0.245	30	0
766	1	126	60	0	0	30.1	0.349	47	1
767	1	93	70	31	0	30.4	0.315	23	0

768 rows × 9 columns

```
In [42]: 1 x = data.drop(columns=['Outcome'])
2 y = data['Outcome']
```

```
In [43]: 1 from sklearn.model_selection import train_test_split
```

```
In [44]: 1 x_train, x_test, y_train, y_test = train_test_split(x, y, test_size = 0.2, random_state = 42)
```

```
In [45]: 1 from keras.layers import Dense
2 model = Sequential()
3 model.add(Dense(32, activation = 'relu',input_shape=(x_train.shape[1],)))
4 model.add(Dense(16, activation = 'relu',))
5 model.add(Dense(1, activation = 'sigmoid'))
```

```
In [46]: 1 model.compile(optimizer = 'adam', loss = 'binary_crossentropy', metrics = ['Accuracy'])
2 model.fit(x_train, y_train, epochs= 30, batch_size = 30, validation_split = 0.2)
```

Epoch 1/30
17/17 [=====] - 1s 21ms/step - loss: 5.0186 - Accuracy: 0.5927 - val_loss: 3.3703 - val_Accuracy: 0.4146
Epoch 2/30
17/17 [=====] - 0s 8ms/step - loss: 2.2139 - Accuracy: 0.5519 - val_loss: 1.4200 - val_Accuracy: 0.6179
Epoch 3/30
17/17 [=====] - 0s 7ms/step - loss: 1.4807 - Accuracy: 0.5601 - val_loss: 1.5759 - val_Accuracy: 0.5610
Epoch 4/30
17/17 [=====] - 0s 6ms/step - loss: 1.1180 - Accuracy: 0.5560 - val_loss: 1.3528 - val_Accuracy: 0.5854
Epoch 5/30
17/17 [=====] - 0s 7ms/step - loss: 0.9987 - Accuracy: 0.5906 - val_loss: 1.2177 - val_Accuracy: 0.6098
Epoch 6/30
17/17 [=====] - 0s 6ms/step - loss: 0.8598 - Accuracy: 0.6477 - val_loss: 0.8554 - val_Accuracy: 0.5772
Epoch 7/30
17/17 [=====] - 0s 6ms/step - loss: 0.7600 - Accuracy: 0.6445 - val_loss: 0.8111 - val_Accuracy: 0.5854

```
In [47]: 1 test_loss, test_acc = model.evaluate(x_test, y_test)
2 print('Test_Accuracy:',test_acc)
```

5/5 [=====] - 0s 4ms/step - loss: 0.6467 - Accuracy: 0.7013
Test_Accuracy: 0.701298713684082

```
In [ ]: 1
```

```
In [ ]: 1
```