WORKSHEET-1

Pima Indians onset of diabetes dataset

```
#importing required libraries
import pandas as pd
from keras.models import Sequential
from keras.layers import Dense


# loading the dataset
data = pd.read_csv('pima-indians-diabetes.csv', delimiter=',')
```

```
data.head(10)
```

	6	148	72	35	0	33.6	0.627	50	1
0	1	85	66	29	0	26.6	0.351	31	0
1	8	183	64	0	0	23.3	0.672	32	1
2	1	89	66	23	94	28.1	0.167	21	0
3	0	137	40	35	168	43.1	2.288	33	1
4	5	116	74	0	0	25.6	0.201	30	0
5	3	78	50	32	88	31.0	0.248	26	1
6	10	115	0	0	0	35.3	0.134	29	0
7	2	197	70	45	543	30.5	0.158	53	1
8	8	125	96	0	0	0.0	0.232	54	1
9	4	110	92	0	0	37.6	0.191	30	0

```
#renaming columns
data.rename(columns={'6':'Noofpregnant','148':'Plasma glucose concentration a 2 hours in an o
                     '72':'Diastolic blood pressure (mm Hg)','35':'Triceps skin fold thicknes
                     '0':'2-Hour serum insulin (mu U/ml)','33.6':'Body mass index (weight in
                      ,'0.627':'Diabetes pedigree function'
                        ,'50':'Age (years)'
                        ,'1':'Class'},inplace=True)


# split into input (X) and output (y) variables
X = data.drop(['Class'], axis = 1)
Y = data[['Class']]
```

```
# define the keras model
model = Sequential()
model.add(Dense(12, input_dim= 8 , activation='relu'))
model.add(Dense(8, activation='relu'))
model.add(Dense(1, activation='sigmoid'))


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


# fit the keras model on the dataset
model.fit(X, Y, epochs= 150, batch_size=10 )
```

```
Epoch 1/150
77/77 [==============================] - 1s 945us/step - loss: 38.1498 - accuracy: 0.
Epoch 2/150
77/77 [==============================] - 0s 1ms/step - loss: 5.4594 - accuracy: 0.576
Epoch 3/150
77/77 [==============================] - 0s 1ms/step - loss: 1.2007 - accuracy: 0.649
Epoch 4/150
77/77 [==============================] - 0s 1ms/step - loss: 0.8383 - accuracy: 0.674
Epoch 5/150
77/77 [==============================] - 0s 1ms/step - loss: 0.7877 - accuracy: 0.674
Epoch 6/150
77/77 [==============================] - 0s 996us/step - loss: 0.8725 - accuracy: 0.6
Epoch 7/150
77/77 [==============================] - 0s 1ms/step - loss: 0.7793 - accuracy: 0.656
Epoch 8/150
77/77 [==============================] - 0s 2ms/step - loss: 0.7763 - accuracy: 0.665
Epoch 9/150
77/77 [==============================] - 0s 1ms/step - loss: 0.7043 - accuracy: 0.696
Epoch 10/150
77/77 [==============================] - 0s 962us/step - loss: 0.7880 - accuracy: 0.6
Epoch 11/150
77/77 [==============================] - 0s 982us/step - loss: 0.7198 - accuracy: 0.6
Epoch 12/150
77/77 [==============================] - 0s 990us/step - loss: 0.6738 - accuracy: 0.6
Epoch 13/150
77/77 [==============================] - 0s 964us/step - loss: 0.6499 - accuracy: 0.6
Epoch 14/150
77/77 [==============================] - 0s 1ms/step - loss: 0.6287 - accuracy: 0.668
Epoch 15/150
77/77 [==============================] - 0s 1ms/step - loss: 0.6362 - accuracy: 0.661
Epoch 16/150
77/77 [==============================] - 0s 997us/step - loss: 0.6743 - accuracy: 0.6
Epoch 17/150
77/77 [==============================] - 0s 998us/step - loss: 0.6349 - accuracy: 0.6
Epoch 18/150
77/77 [==============================] - 0s 1ms/step - loss: 0.6235 - accuracy: 0.670
Epoch 19/150
77/77 [==============================] - 0s 1ms/step - loss: 0.6334 - accuracy: 0.664
Epoch 20/150
77/77 [==============================] - 0s 988us/step - loss: 0.6151 - accuracy: 0.6
Epoch 21/150
77/77 [==============================] - 0s 1ms/step - loss: 0.6432 - accuracy: 0.645
```

```
Epoch 22/150
77/77 [==============================] - 0s 1ms/step - loss: 0.6833 - accuracy: 0.675
Epoch 23/150
77/77 [==============================] - 0s 1ms/step - loss: 0.6131 - accuracy: 0.689
Epoch 24/150
77/77 [==============================] - 0s 983us/step - loss: 0.5815 - accuracy: 0.7
Epoch 25/150
77/77 [==============================] - 0s 987us/step - loss: 0.5947 - accuracy: 0.6
Epoch 26/150
77/77 [==============================] - 0s 1ms/step - loss: 0.5890 - accuracy: 0.694
Epoch 27/150
77/77 [==============================] - 0s 1ms/step - loss: 0.5723 - accuracy: 0.686
Epoch 28/150
77/77 [==============================] - 0s 2ms/step - loss: 0.5774 - accuracy: 0.733
Epoch 29/150
77/77 [==============================] - 0s 1ms/step - loss: 0.5918 - accuracy: 0.693
```

```
# evaluate the keras model
_, accuracy = model.evaluate(X, Y)
print('Accuracy: %.2f' % (accuracy*100))
```

```
24/24 [==============================] - 0s 957us/step - loss: 0.5603 - accuracy: 0.7014
Accuracy: 70.14
```