In [1]:

```python
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
import os
os.environ['TF_CPP_MIN_LOG_LEVEL'] = '3'

from keras.models import Sequential
from keras.layers import Dense, Activation
import seaborn as sns
import tensorflow as tf
```

In [2]:

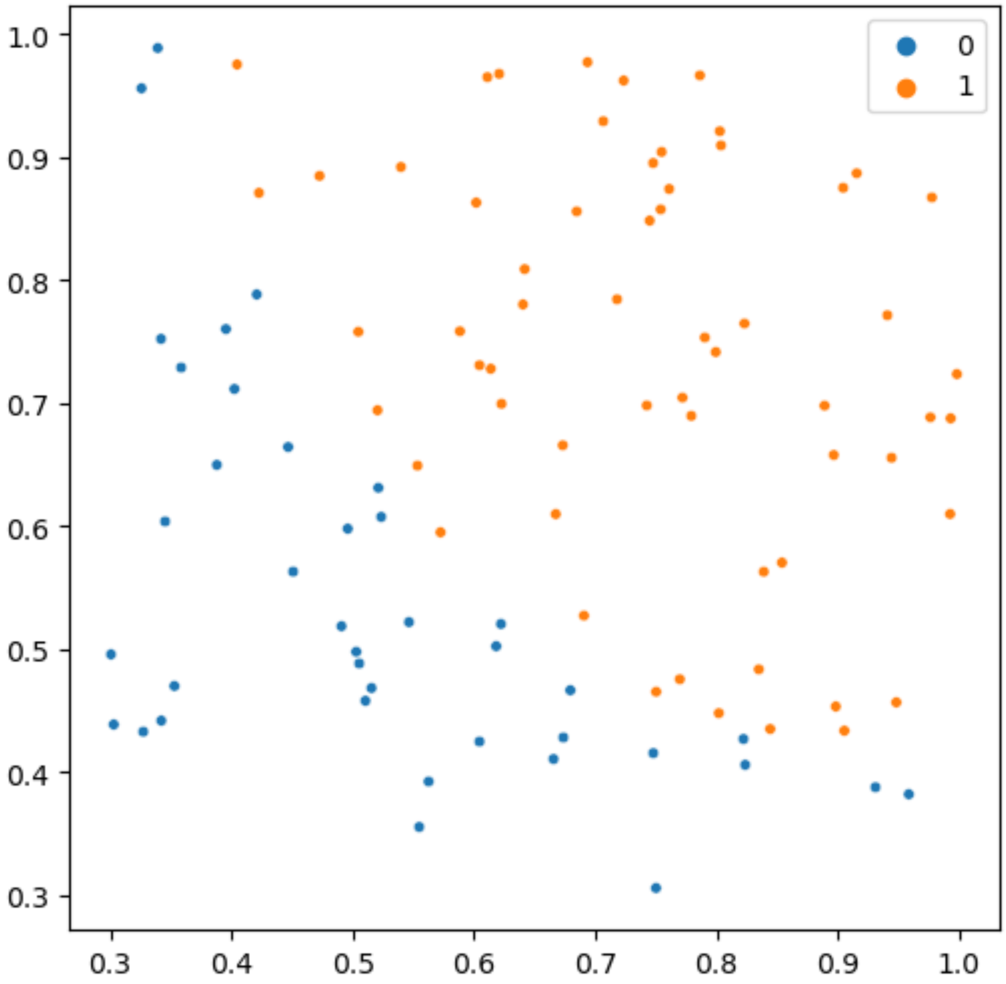
```python
data = pd.read_csv("".join([os.environ['HOME'], "/marks.txt"]))

 # X = feature values, all the columns except the last column
X = 0.01 * data.iloc[:, :-1].to_numpy(dtype=np.float64)

# y = target values, last column of the data frame
y = data.iloc[:, -1].to_numpy(dtype=np.int64)

plt.figure(figsize=(6, 6))

sns.scatterplot(x=X[:, 0],
                y=X[:, 1],
                hue=y,
                s=16);
```



In [3]:

```
dense = tf.keras.layers.Dense(1, activation='sigmoid')
model = tf.keras.models.Sequential([dense])
model.compile(tf.keras.optimizers.Adam(learning_rate=0.05), 'binary_crossent
model.fit(X, y, epochs=200, verbose=0)
print(model.evaluate(X, y))
```

```
Metal device set to: Apple M2
4/4 [==============================] - 0s 6ms/step - loss: 0.3488 - accurac
y: 0.9091
[0.34881240129470825, 0.9090909361839294]
```

In [4]:

```
print(dense.get_weights())
w = dense.get_weights()[0].T[0]
print(w)
b = dense.get_weights()[1].T[0]
print(b)
```

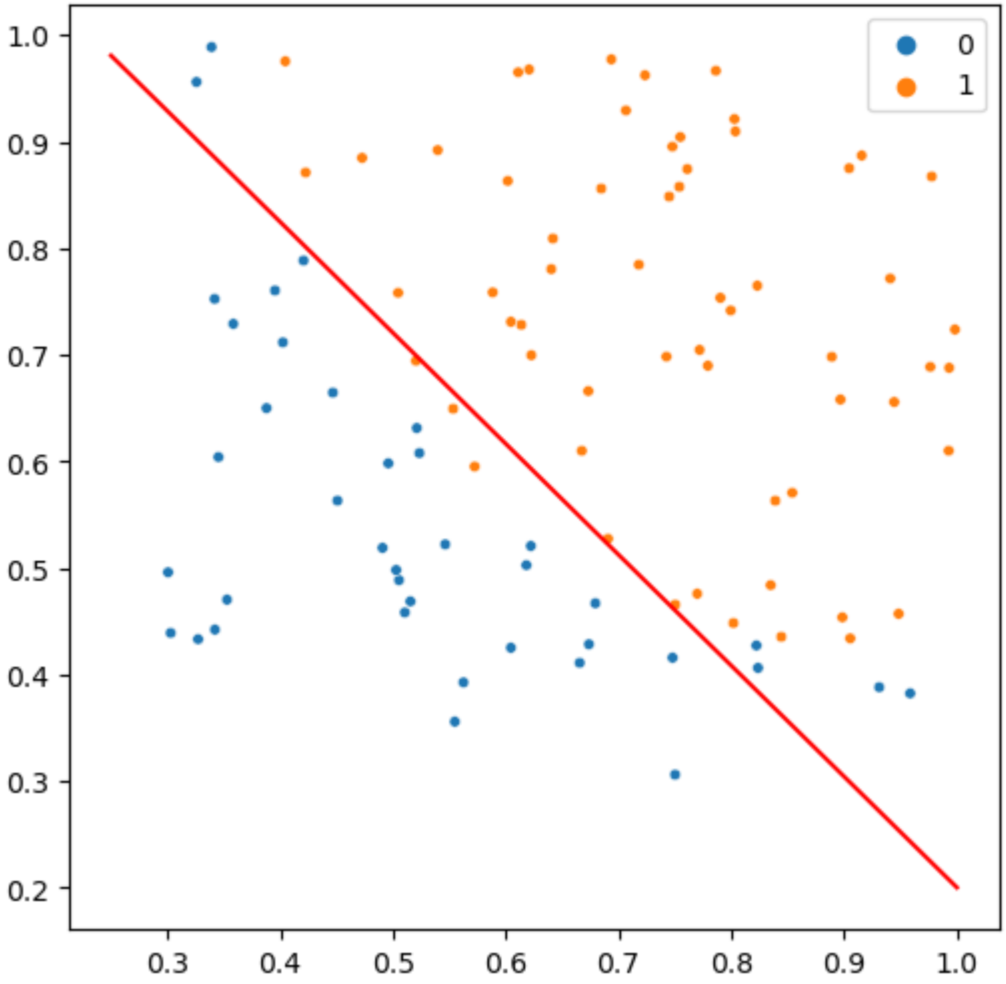
```
[array([[5.1874466],
       [4.983886 ]], dtype=float32), array([-6.1848087], dtype=float32)]
[5.1874466 4.983886 ]
-6.1848087
```

In [5]:

```
plt.figure(figsize=(6, 6))# Plotting our two-features-space

sns.scatterplot(x=X[:, 0],
                y=X[:, 1],
                hue=y,
                s=16);

x_points = np.linspace(0.25, 1.0)
y_points = -(w[0] / w[1]) * x_points - b / w[1]
plt.plot(x_points, y_points, c='r');
```



In [ ]: