

Вариант 3

```
def genData(size=500):  
    data = np.random.rand(size, 2)*2 - 1  
    label = np.zeros([size, 1])  
    for i, p in enumerate(data):  
        if (p[0]+0.2)**2 + (0.6*p[1])**2 >= 0.25:  
            label[i] = 0.  
        else:  
            label[i] = 1.  
    div = round(size*0.8)  
    train_data = data[:div, :]  
    test_data = data[div:, :]  
    train_label = label[:div, :]  
    test_label = label[div:, :]  
    return (train_data, train_label), (test_data, test_label)
```

Модель была выбрана таким образом:

```
model = models.Sequential()  
model.add(layers.Dense(25, activation='relu'))  
model.add(layers.Dense(25, activation='relu'))  
model.add(layers.Dense(1, activation='sigmoid'))  
model.compile(optimizer='rmsprop', loss='binary_crossentropy', metrics=['accuracy'])  
H = model.fit(train_data, train_label, epochs=150, batch_size=30, validation_data=(test_data, test_label), verbose=False)
```

Графики:

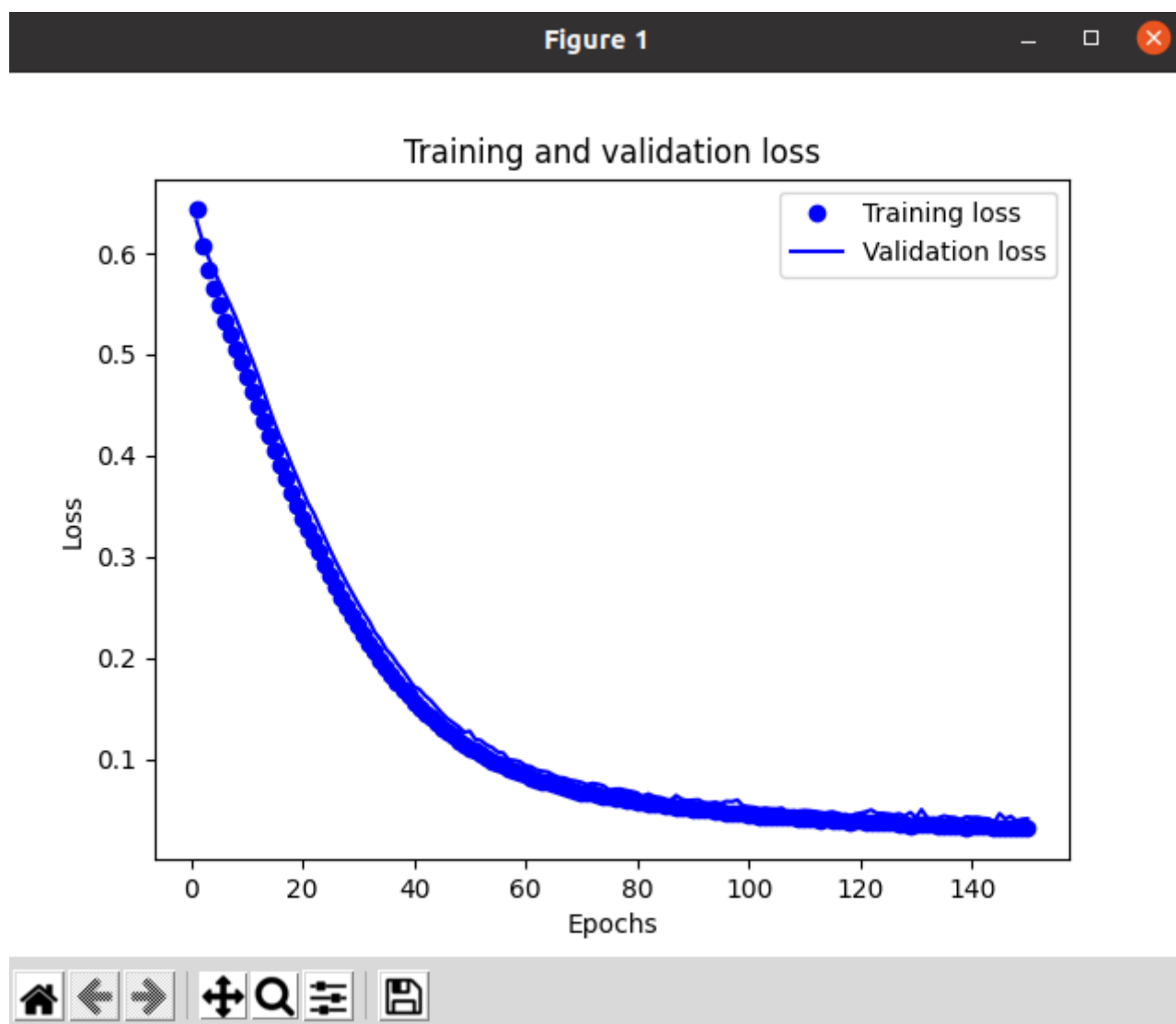


Рис.1 График ошибок

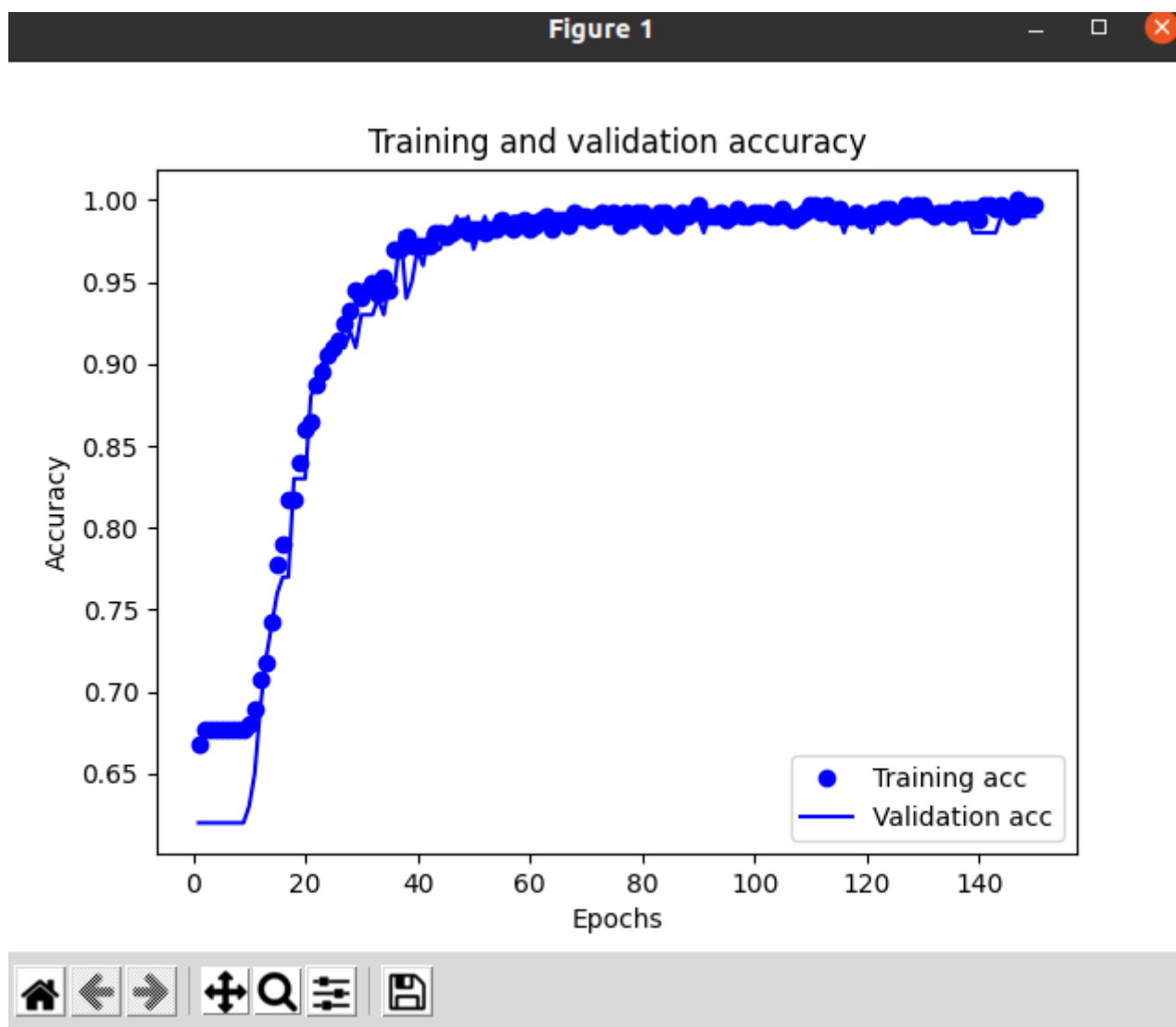


Рис.2 График точности

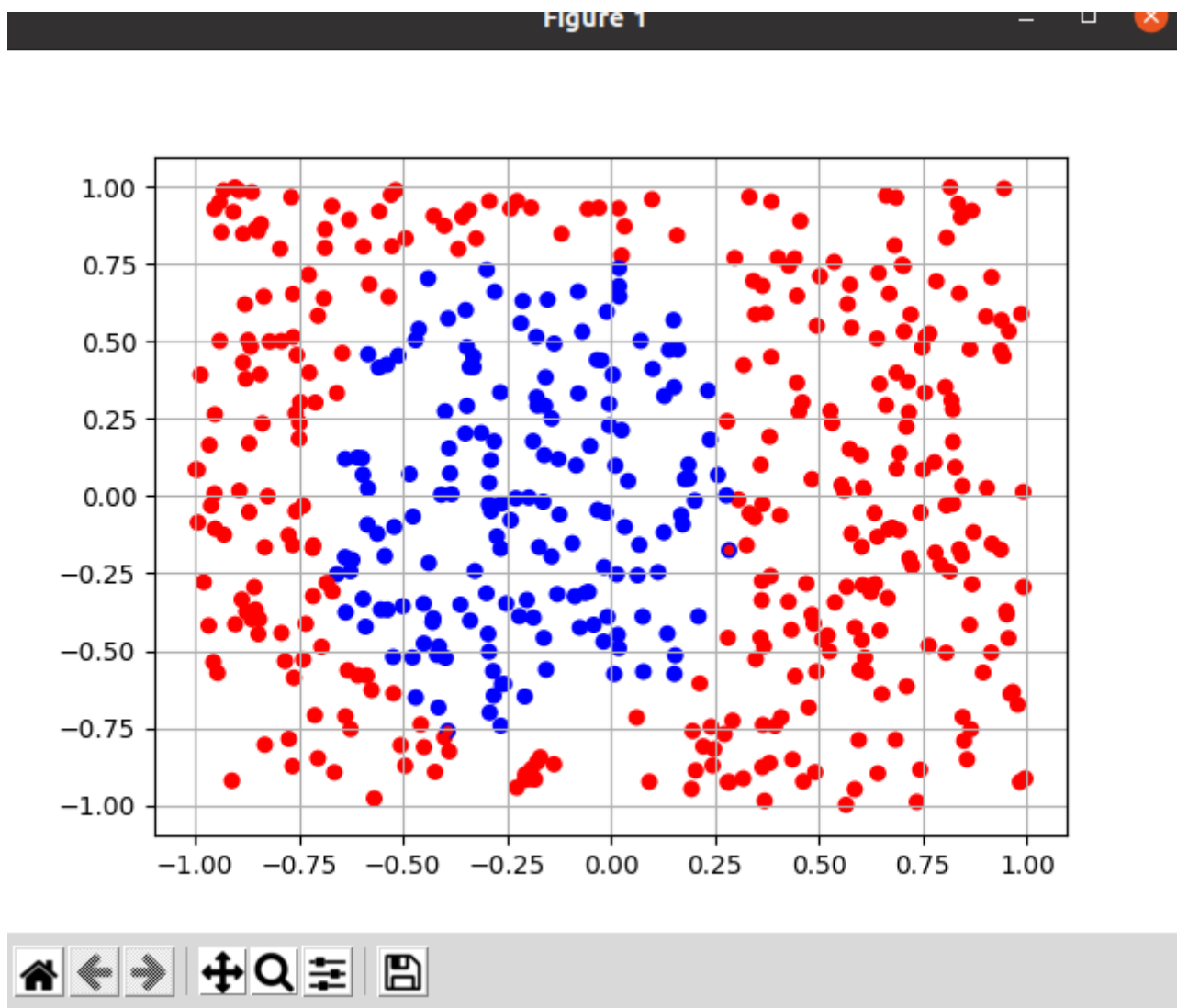


Рис.3 Результаты бинарной классификации

Из графиков можно понять, что модель обучается примерно за 40-60 эпох.

Точность примерно 99%, ошибки 4%:

```
4/4 [=====] - 0s 651us/step - loss: 0.0413 - accuracy: 0.9900  
[0.041297703981399536, 0.99000000095367432]
```