

Rohan Nyati

500075940

R177219148

B5-Ai & ML

Experiment -4

The image displays two screenshots of a Jupyter Notebook interface. The top screenshot shows the notebook's title bar, menu bar, and toolbar. The main content area has the heading "Neural Networks" and a paragraph: "In the previous problem: Write down the optimization function. The goal is to learn w and b by minimizing the cost function J . For a parameter θ , the update rule is $\theta = \theta - \alpha d\theta$, where α is the learning rate." Below this is the heading "L1 loss function" and a code cell with the following code:

```
In [1]: import numpy as np

def L1(yhat, y):
```

The bottom screenshot shows the same notebook with the code cell completed. The code defines the L1 loss function:

```
In [1]: import numpy as np

def L1(yhat, y):
    loss = np.sum(np.abs(yhat-y), axis = 0)
    return loss
```

Below the L1 loss function code is the heading "Forward pass" and a code cell with the following code:

```
In [11]: def propagate(w, b, X, Y):

        m = X.shape[1]
```

Rohan_Nyati_Lab4 - Jupyter Notebook | localhost:8888/notebooks/Desktop/Sem%206/NN/NN_Lab/Rohan_Nyati_Lab4.ipynb#Samarth-Sundriyal

Forward pass

```
In [11]: def propagate(w, b, X, Y):  
    m = X.shape[1]  
    A = sigmoid(np.dot(w.T, X) + b)  
    cost = -1./m * np.sum(Y*np.log(A) + (1-Y)*np.log(1-A))  
  
    dw = 1./m * np.dot(X, (A-Y).T)  
    db = 1./m * np.sum(A-Y)  
  
    assert(dw.shape == w.shape)  
    assert(db.dtype == float)  
  
    grads = {"dw": dw,  
            "db": db}
```

db = 1./m * np.sum(A-Y)

```
assert(dw.shape == w.shape)  
assert(db.dtype == float)  
  
grads = {"dw": dw,  
        "db": db}  
  
return grads, cost
```

Optimization function

```
In [13]: def optimize(w, b, X, Y, num_iterations, learning_rate, print_cost = False):  
    costs = []  
  
    for i in range(num_iterations):
```

Rohan_Nyati_Lab4 - Jupyter Not... x

localhost:8888/notebooks/Desktop/Sem%206/NN/NN_Lab/Rohan_Nyati_Lab4.ipynb#Samarth-Sundriyal

jupyter Rohan_Nyati_Lab4 (autosaved) Logout

File Edit View Insert Cell Kernel Widgets Help Not Trusted Python 3

Run Code

```
db = grads["db"]

w = w - learning_rate * dw
b = b - learning_rate * db

if i % 100 == 0:
    costs.append(cost)

if print_cost and i % 100 == 0:
    print ("Cost after iteration %i: %f" %(i, cost))

params = {"w": w,
          "b": b}

grads = {"dw": dw,
         "db": db}

return params, grads, costs
```

Rohan_Nyati_Lab4 - Jupyter Not... x

localhost:8888/notebooks/Desktop/Sem%206/NN/NN_Lab/Rohan_Nyati_Lab4.ipynb#Samarth-Sundriyal

jupyter Rohan_Nyati_Lab4 (autosaved) Logout

File Edit View Insert Cell Kernel Widgets Help Not Trusted Python 3

Run Code

```
w = w - learning_rate * dw
b = b - learning_rate * db

if i % 100 == 0:
    costs.append(cost)

if print_cost and i % 100 == 0:
    print ("Cost after iteration %i: %f" %(i, cost))

params = {"w": w,
          "b": b}

grads = {"dw": dw,
         "db": db}

return params, grads, costs
```