Introduction to Machine Learning Program Assignment #4

0510002 袁鈺勛

1. Forward Propagate

```python
def forward_propagate(X, theta1, theta2):
    m = X.shape[0]

    #Write codes here
    a1 = np.insert(X, 0, values = np.ones(m), axis = 1)
    z2 = a1 * theta1.T
    a2 = np.insert(sigmoid(z2), 0, values = np.ones(m), axis = 1)
    z3 = a2 * theta2.T
    h = sigmoid(z3)

    return a1, z2, a2, z3, h
```

2. Backward Propagate

```python
def backprop(params, input_size, hidden_size, num_labels, X, y, learning_rate):
    m = X.shape[0]

    #Write codes here
    theta1, theta2, a1, z2, a2, z3, h, J = cost(params, input_size, hidden_size, num_labels, X, y, learning_rate)

    delta1 = np.zeros(theta1.shape)
    delta2 = np.zeros(theta2.shape)

    for t in range(m):
        z2_colt = z2[t, :]

        d3t = h[t, :] - y[t, :]

        z2_colt = np.insert(z2_colt, 0, values = np.ones(1))
        d2t = np.multiply((theta2.T * d3t.T).T, sigmoid_gradient(z2_colt))

        delta1 = delta1 + (d2t[:, 1:]).T * a1[t, :]
        delta2 = delta2 + d3t.T * a2[t, :]

    delta1 = delta1 / m
    delta2 = delta2 / m

    delta1[:, 1:] = delta1[:, 1:] + (theta1[:, 1:] * learning_rate) / m
    delta2[:, 1:] = delta2[:, 1:] + (theta2[:, 1:] * learning_rate) / m

    grad = np.concatenate((np.ravel(delta1), np.ravel(delta2)))

    return J, grad
```

3. Accuracy

accuracy = 99.38%

```
user@user-K43SD:~/Homework/ML/HW4$ python3 0510002.py
Start time:  2019-01-03 23:52:25.103129
/usr/local/lib/python3.5/dist-packages/sklearn/preprocessing/_encoders.py:363: FutureWarning: The handling of integer data will change in versi
on 0.22. Currently, the categories are determined based on the range [0, max(values)], while in the future they will be determined based on the
 unique values.
If you want the future behaviour and silence this warning, you can specify "categories='auto'".
In case you used a LabelEncoder before this OneHotEncoder to convert the categories to integers, then you can now use the OneHotEncoder directl
y.
  warnings.warn(msg, FutureWarning)
End time:  2019-01-04 00:02:37.526653
Elapsed time: 00:10:12.42
accuracy = 99.38%
```