

## Ex4 Unit Tests

Subscribe for email updates.

PINNED UNRESOLVED

UnitTest × + Add Tag

Sort replies by: Oldest first Newest first Most popular



Alex McCarthy · 5 days ago

★ APPROVED

Another week, another thread.

Same as before: please add your own unit tests to this thread, suggest improvements, or point out mistakes!

Don't post your code - just state the test case you used, and give the results.

These tests use the default octave formatting: type 'format' to restore the default format 'short'.

---

Test 1a (Feedforward and Cost Function):

**input:**

```
input_layer_size = 2;  
hidden_layer_size = 2;  
num_labels = 4;  
nn_params = [ 1:18 ] / 10;  
X = cos([1 2 ; 3 4 ; 5 6]);  
y = [4; 2; 3];  
lambda = 0;
```

```
nnCostFunction(nn_params, input_layer_size, hidden_layer_size, num_labels, X, y, lambda)
```

**output:**

```
ans = 7.4070
```

---

*Test 2a (Regularized Cost Function):*

**input:**

```
input_layer_size = 2;  
hidden_layer_size = 2;  
num_labels = 4;  
nn_params = [ 1:18 ] / 10;  
X = cos([1 2 ; 3 4 ; 5 6]);  
y = [4; 2; 3];  
lambda = 3;
```

```
nnCostFunction(nn_params, input_layer_size, hidden_layer_size, num_labels, X, y, lambda)
```

**output:**

```
ans = 16.457
```

---

*Test 3a (Sigmoid Gradient):*

**input:**

```
sigmoidGradient(1)
```

**output:**

```
ans = 0.19661
```

---

*Test 3b:*

**input:**

```
sigmoidGradient([2 3])
```

**output:**

```
ans =
```

```
0.104994 0.045177
```

---

Test 3c:

**input:**

```
sigmoidGradient([-2 0; 4 999999; -1 1])
```

**output:**

```
ans =
```

```
0.10499 0.25000
```

```
0.01766 0.00000
```

```
0.19661 0.19661
```

---

Test 4a (Neural Network Gradient (Backpropagation)):

**input:**

```
input_layer_size = 2;
```

```
hidden_layer_size = 2;
```

```
num_labels = 4;
```

```
nn_params = [ 1:18 ] / 10;
```

```
X = cos([1 2 ; 3 4 ; 5 6]);
```

```
y = [4; 2; 3];
```

```
lambda = 0;
```

```
[J grad] = nnCostFunction(nn_params, input_layer_size, hidden_layer_size, num_labels, X, y, lambda)
```

**output:**

```
J = 7.4070
```

```
grad =
```

```
0.766138
```

```
0.979897
```

```
-0.027540
```

```
-0.035844
```

```
-0.024929
```

```
-0.053862
```

```
0.883417
```

```
0.568762
```

```
0.584668
```

```
0.598139
```

```
0.459314
```

```
0.344618
```

```
0.256313
```

```
0.311885
```

```
0.478337
```

```
0.368920
```

```
0.259771
```

```
0.322331
```

---

Test 5a (Regularized Gradient):

**input:**

```
input_layer_size = 2;
```

```
hidden_layer_size = 2;
```

```
num_labels = 4;
```

```
nn_params = [ 1:18 ] / 10;
```

```
X = cos([1 2 ; 3 4 ; 5 6]);
```

```
y = [4; 2; 3];
```

```
lambda = 3;
```

```
[J grad] = nnCostFunction(nn_params, input_layer_size, hidden_layer_size, num_labels, X, y, lambda)
```

**output:**

$J = 16.457$

$grad =$

0.76614

0.97990

0.27246

0.36416

0.47507

0.54614

0.88342

0.56876

0.58467

0.59814

1.55931

1.54462

1.55631

1.71189

1.97834

1.96892

1.95977

2.12233

↑ 12 ↓ · flag



Tom Mosher COMMUNITY TA · 5 days ago

@Alex: I concur with your unit test results. Thanks for your work!

↑ 0 ↓ · flag



Paul T Mielke COMMUNITY TA · 5 days ago

Alex, I checked all your tests and the answers look good to me as well. Thanks very much for putting this together!

↑ 0 ↓ · flag

Julien-Benjamin RUIZ · 4 days ago 

Apparently, there is a syntax error for this line of the first test unit : `nn_params = [magic(3)(); magic(4)()];`

I think that's because you cannot index directly a variable and declaring it in Matlab, as least this way.

Error given by Matlab 2012 : Error: ()-indexing must appear last in an index expression.

↑ 0 ↓ · flag

Colin Beckingham COMMUNITY TA · 4 days ago 

Check your syntax carefully - the line you quote works perfectly fine in Octave and is basic syntax so it should work equally well in Matlab. Perhaps a Matlab user could confirm this.

↑ 0 ↓ · flag

Julien-Benjamin RUIZ · 4 days ago 

I do not need to check my syntax ; I just copy-paste the unit test, so it should be running fine.

Edit : I should precise that, so far, that's the first time I got an error with the provided unit tests, especially for a variable declaration.

↑ 0 ↓ · flag

Colin Beckingham COMMUNITY TA · 4 days ago 

OK then it must be a Matlab oddity. You might want to post a note in the Matlab subforum for Mike Reardon to

comment on, he is the Matlab guru and can give you an authoritative answer. Can't help any more here I'm afraid.

↑ 0 ↓ · flag

Julien-Benjamin RUIZ · 4 days ago 

OK.

Thanks anyway ;)

↑ 0 ↓ · flag

 Tom Mosher COMMUNITY TA · 4 days ago 

Yes, I think I recall a similar Matlab issue in the previous session. Some valid Octave syntax is not valid in Matlab - so you might need to split that into separate lines.

↑ 0 ↓ · flag

 Alex McCarthy · 3 days ago 

Thanks for reporting this, Julien-Benjamin. I've updated the unit tests with syntax that should work with Matlab and Octave: can you try pasting them and confirming that they work?

↑ 0 ↓ · flag

---

+ Comment

Anonymous · 5 days ago 

My Part 1 was submitted and is correct.

For test2a: If I run the regularization on all numbers, including bias frames:

1074.6

Running the same code on ex 4:

0.381813 (should be 0.383770)

Any ideas? Also, Theta1 seems to have 401 parameters and not the 400 that the Instructions say it should have.

↑ 0 ↓ · flag



Tom Mosher COMMUNITY TA · 5 days ago



For the exercise example, Theta1 is 401 parameters. That is consistent with the instructions. 400 for the 20x20 pixels, plus one for the bias units.

The sizes are different for the unit test.

↑ 0 ↓ · flag



Tom Mosher COMMUNITY TA · 5 days ago



You may find this post (dimensional analysis) useful.

[https://class.coursera.org/ml-008/forum/thread?thread\\_id=1069](https://class.coursera.org/ml-008/forum/thread?thread_id=1069)

↑ 0 ↓ · flag

---

+ Comment

Manish Hatwalne · 4 days ago



I get this for first unit test -

```
nnCostFunction(nn_params, input_layer_size, hidden_layer_size, num_labels, X, y, lambda)  
ans = 97.511
```

Any suggestions?

↑ 0 ↓ · flag

+ Comment

Frank Heimerzheim · 4 days ago 

Very strange ...

For test 4a i get the correct cost, but my values for grad are exactly twice the result from the unit test. I checked my code some times, but can't find any error. Much less so any factor of 2.

Has someone seen anything like that and have any ideas for me?

Thanks!

↑ 0 ↓ · flag

Frank Heimerzheim · 4 days ago 

Found the error.

I've implemented a vectorized version of the algorithm. Strangely it worked nearly fine for the unit test, besides the factor of 2. For checkNNGradients it didn't work at all. So i strated from scratch, now with one testcase after the other and everything works fine.

Sorry for the confusion.

↑ 0 ↓ · flag



Tom Mosher COMMUNITY TA · 4 days ago 

Hi Frank,

If you can disclose what the error was, we can look at potential improvements to the unit tests to better catch that issue.

↑ 0 ↓ · flag

Frank Heimerzheim · 4 days ago 

*I think there is no need for a unit test for test 4, as there ist the possibility to use checkNNGradients.*

*Beside of that thank you very much for the providing of the tests.*

↑ 0 ↓ · flag

 Alex McCarthy · 3 days ago 

*I agree that the unit tests are less useful this week, since this week's assignment had a lot of great built-in error checking that makes it much easier to feel confident about our answers. Thanks to the course authors for that! :)*

↑ 0 ↓ · flag

Bin Lang · a day ago 

*the is the reply for Manish's post, do I miss anything here, how come the row of  $a3 = 2$ ? there are 3 training set, size( $X$ ) = 3,*

$a3 =$   
1.00000 1.00000 1.00000 1.00000  
1.00000 1.00000 1.00000 1.00000

↑ 0 ↓ · flag

 Tom Mosher COMMUNITY TA · a day ago 

*Sorry, I don't understand your question.*

*The data for these unit tests was updated a few days ago, so Manish's question may be obsolete.*

↑ 0 ↓ · flag

[+ Comment](#)

Manish Hatwalne · 4 days ago 

In test 1a, I get this for a3 (hypothesis at output layer), is that correct?

```
a3 =
```

```
1.00000 1.00000 1.00000 1.00000
1.00000 1.00000 1.00000 1.00000
```

 0  · flag



Tom Mosher COMMUNITY TA · 4 days ago 

Yes.

 2  · flag

Filip Zivanovic · 4 days ago 

Ok, I'm getting the same values for a3, but if this is the cost function:

$$J(\theta) = \frac{1}{m} \sum_{i=1}^m \sum_{k=1}^K \left[ -y_k^{(i)} \log((h_\theta(x^{(i)}))_k) - (1 - y_k^{(i)}) \log(1 - (h_\theta(x^{(i)}))_k) \right]$$

and if :

$$h_{\theta}(x^{(i)})_k = a_k^{(3)}$$

than this part:

$\log(1 - (h_{\theta}(x^{(i)}))_k)$  can get a  $\log(0)$ , am I right or am I doing something wrong?

Thanks.

↑ 0 ↓ · flag



Tom Mosher COMMUNITY TA · 4 days ago

The  $a_3$  values you're getting are only shown as 1.0000 because of the display format you're using. If you use ...

format long e

... in your Octave console, you'll see the full precision of the values, and they (probably) won't be identically 1. Therefore  $\log(0)$  won't occur.

They're very close though. Perhaps Alex can be induced to change the range of values in the Unit Test.

↑ 1 ↓ · flag

Filip Zivanovic · 4 days ago

Thank you for your quick answer, you are right, it was the display format. I'm debugging my code, and I was just curious to know if my bug is related to that, but it's something else, obviously, still don't know what, but I'll find it, no need for changing the values.

↑ 0 ↓ · flag



Alex McCarthy · 3 days ago

I'm very happy to update the unit test input values. Can you recommend different inputs that provide less uniform hypothesis values?

↑ 0 ↓ · flag

 Alex McCarthy · 3 days ago 

I've scaled the values of Theta down by 10x as Tom recommended, so your a3 values should be something other than 1 when using the default format short.

↑ 0 ↓ · flag

 Tom Mosher COMMUNITY TA · 3 days ago 

Cool, thanks for the update.

Next: We have an issue again with use of square matrices in the unit tests Several students had code that passed the unit tests, but not the exercise or submit, because the square matrices don't facilitate detection of dimensional errors.

All of the Theta matrices in the unit tests are square.

This construct is almost as easy to type:

```
Theta = reshape(1;12,3,4)/10
```

↑ 0 ↓ · flag

 Alex McCarthy · 2 days ago 

Good point. Switched to using non-square values for Thetas and X, which also had the benefit of making the input and output shorter and more typeable.

Mind double checking the latest numbers?

↑ 0 ↓ · flag



Paul T Mielke COMMUNITY TA · 2 days ago 🔍

*Alex, the current numbers all check with my code. Thanks!*

↑ 0 ↓ · flag



Tom Mosher COMMUNITY TA · 2 days ago 🔍

*I'm having trouble with a "non-conformant argument" error on the new Unit Test 2a. Maybe it's just me fumble-fingering my keyboard.*

↑ 0 ↓ · flag



Tom Mosher COMMUNITY TA · 2 days ago 🔍

*No, I don't think it's me. `a1` and `Theta1` don't seem to have compatible sizes.*

↑ 0 ↓ · flag



Paul T Mielke COMMUNITY TA · 2 days ago 🔍

*I just checked again and it worked for me. Test 2a, right? I'm using the Octave GUI, so I did it with cut and paste. You're sure you typed it correctly?*

↑ 0 ↓ · flag



Tom Mosher COMMUNITY TA · 2 days ago 🔍

*I'm not sure, since you say it worked. But I typed and re-checked several times, put a breakpoint in the function, and whos'd the sizes. I must be overdue for a nap.*

↑ 0 ↓ · flag



Tom Mosher COMMUNITY TA · 2 days ago

*Ah, I got it. I had the X matrix as (1 x 6) (instead of (2 x 3) because the elements are needlessly separated by commas and semicolons, when spaces and semicolons are more readable. I'm editing that.*

*To my squinty eyeballs, the commas and semicolons looked the same.*

0 · flag



Alex McCarthy · 2 days ago

*Thanks Tom, that is cleaner.*

*I seriously wish we could get paste working for you :/ You're running octave on windows, right? Do any of the following tips let you paste input into Octave? <http://stackoverflow.com/questions/131955/keyboard-shortcut-to-paste-clipboard-content-into-command-...>*

1 · flag



Tom Mosher COMMUNITY TA · 2 days ago

*What we really need is a way to upload/download unit test scripts.*

0 · flag

---

+ Comment



Tom Mosher COMMUNITY TA · 4 days ago

@Alex:

*The unit tests might give slightly more usable results if nn\_params was scaled down by 1/10.*

0 · flag



Alex McCarthy · 3 days ago

Good suggestion, done. Mind double checking the new outputs?

↑ 0 ↓ · flag

+ Comment

Filip Zivanovic · 4 days ago 

Well, when I run unit test 1a, I get:

```
ans = 90.877
```

So that's correct, but when I run ex4.m, I get:

```
Cost at parameters (loaded from ex4weights): 3.478115
(this value should be about 0.287629)
```

Any chance for one more unit test 1b?

Thanks.

↑ 0 ↓ · flag



Tom Mosher COMMUNITY TA · 4 days ago 

There are several test cases already available:

- Unit Test 1a has two training examples, two inputs, three hidden, and four outputs.
- Ex4 image example has 5,000 training examples, 400 inputs, 25 hidden, and 10 outputs.
- The "check NN gradients" portions of ex4 have yet another set (5 examples, 3 inputs, 5 hidden, 3 outputs).

What other combinations do you think would be useful?

↑ 1 ↓ · flag

Filip Zivanovic · 4 days ago 

*I think that you are right, in the meantime, I have found where I was wrong.  
Unit tests are more than helpful, it would be much harder without them.*

*Again, thank you.*

 0  · flag

---

+ Comment

Andrew Schroeder · 4 days ago 

*I can confirm that if one is using matlab you need to split the nn\_param creation into 2 steps or you will get the:  
Error: ()-indexing must appear last in an index expression.*

*If you want to modify the tests so they will work in matlab you can change all instances of :*

*nn\_params = [magic(3)(); magic(4)()];*

*to:*

*m3 = magic(3); m4 = magic(4);*

*nn\_params = [m3(); m4()];*

 2  · flag

 Alex McCarthy · 3 days ago 

*Thanks Andrew!. I've updated the unit tests with syntax you recommend: can you try pasting them and confirming that they work?*

 0  · flag

Andrew Schroeder · 3 days ago 

Confirmed that your updates fixed the matlab issue. Thanks.

↑ 1 ↓ · flag

---

+ Comment

Manish Hatwalne · 3 days ago 

This is what I am getting for 4a test -

```
3.184590
2.421557
2.475507
3.022740
1.589554
1.204653
1.229414
1.515271
-1.437125
-1.096236
-1.122432
-1.358537
1.000000
0.500000
1.000000
0.500000
0.985522
0.498816
0.985522
0.486705
0.555575
0.471025
0.555575
```

0.084550  
0.988421  
0.499837  
0.988421  
0.488584

*It seems my Theta2\_grad is correct, but Theta1\_grad is wrong, where should I look for potential error?  
I am doing this in a vectorized code, loop seems to convoluted to me.*

↑ 0 ↓ · flag



Tom Mosher COMMUNITY TA · 3 days ago 🔍

*Computing Theta1\_grad involves the sigmoid gradient function. Have you included that?*

↑ 0 ↓ · flag

Manish Hatwalne · 3 days ago 🔍

*Yes! I got it working! :-)*

↑ 0 ↓ · flag

---

+ Comment

Sagar Manohar · 3 days ago 🔍

*My Regularized cost function is yielding a value of 0.384430 instead of the given 0.383770 in PDF and my submission of the same is failing.*

*Also the unit test 2a given above gives me an answer of 18.607 instead of 17.504.*

What am I missing?

↑ 0 ↓ · flag



Tom Mosher COMMUNITY TA · 3 days ago

*The evidence suggests that the regularized portion of Cost is not correct.*

↑ 0 ↓ · flag

Sagar Manohar · 3 days ago

*After calculating cost without regularization,*

1. I calculated Theta1 and Theta2 values using reshape.
2. calculated element wise square using index '2:end'
3. used sum() twice.
4. added answer to previous cost(w/o reg)

Anything missing?

↑ 0 ↓ · flag



Tom Mosher COMMUNITY TA · 3 days ago

*You don't need to calculate Theta1 and Theta2 - that's already provided by the nnCostFunction.m template code.*

*Also, since Theta1 and Theta2 are matrices, you need to ignore the entire first column, not just the first element.*

*Try these commands in your workspace to see the difference:*

```
Q = magic(3)  
Q(2:end)  
Q(:,2:end)
```

↑ 0 ↓ · flag

Sagar Manohar · 3 days ago 

Ah okay thanks! That worked!

↑ 0 ↓ · flag

+ Comment

Dylan Caponi · 3 days ago 

**EDIT: FOUND MY ERROR!** Forward propagation gave the correct answer, but I wasn't updating a2 properly. I added bias units to a2 in the definition of z3 rather than updating a2 itself to have bias units. You still get the correct answer for feedforward but when you try to use a2 in backpropagation your column dimension will be wrong.

Original post:

Unit test 4a results match exactly but submitting exercise 4 yields "Sorry, your answer was incorrect" and ex4 produces this error:

??? Error using ==> horzcat

CAT arguments dimensions are not consistent.

Error in ==> checkNNGradients at 44

disp([numgrad grad]);

Any ideas as to why unit test would succeed but exercises do not?

When I display sizes I get:

`theta2_grad [3 x 5]`  
`theta1_grad [5 x 4]`  
`numgrad [38 x 1]`  
`grad [35 x 1]`

So it seems like a `theta2_grad` of `[3 x 6]` would create matching dimensions but I don't want to blindly add a column of bias units.

↑ 0 ↓ · flag



Tom Mosher COMMUNITY TA · 3 days ago



This issue pops up in every session of ML, with regard to the Ex4 exercise.

The "horzcat" error message indicates your `nnCostFunction()` appears to be returning the wrong size of value for the `Theta1` and `Theta2` gradients

↑ 0 ↓ · flag

Dylan Caponi · 3 days ago



Right, and I saw that when I printed out the sizes above. However, at what point would I want to reinsert bias units during backpropagation? There is no talk of that in the lectures or exercise guide.

↑ 0 ↓ · flag



Tom Mosher COMMUNITY TA · 3 days ago



You insert bias units during feed-forward.

For back-propagation, you mostly omit them.

↑ 0 ↓ · flag



Tom Mosher COMMUNITY TA · 3 days ago



A tutorial to clarify this is in-process...

↑ 0 ↓ · flag



Tom Mosher COMMUNITY TA · 3 days ago

Said tutorial is now available, in the Programming Exercise 4 sub-forum.

↑ 0 ↓ · flag

---

+ Comment



Patrick Ford · 2 days ago

For 2a, my Cost is too high. I'm removing the effect of the parameters on the bias term by setting that vector equal to 0 in both Theta matrices. After doing so, I get that the remaining elements in `magic(3)`, after being squared, sums to 196 and the remaining elements in `magic(4)`, after being squared, sum to 1118. When we then sum these (196+1118) and multiply by 3/4, the parameter penalty is 1034.5. Adding this to the 90.877 yields too high of a solution.

Can anyone diagnose my logical flaw?

↑ 0 ↓ · flag



Tom Mosher COMMUNITY TA · 2 days ago

I'll post the values for `a1`, `a2`, and `a3`, in a few minutes, after I get back from running some errands.

↑ 0 ↓ · flag



Tom Mosher COMMUNITY TA · 2 days ago

@Patrick: Note that the 2a unit test has been updated since you ran yours, the "`magic()`" square matrices are no longer included. You'll need to re-load the page and get the updated test.

↑ 0 ↓

· flag



Tom Mosher COMMUNITY TA · 2 days ago

Intermediate results for new Unit Test 2a:

```
debug> a1
a1 =

 1.00000  0.54030 -0.41615
 1.00000 -0.98999 -0.65364
 1.00000  0.28366  0.96017

debug> a2
a2 =

 1.00000  0.51350  0.54151
 1.00000  0.37196  0.35705
 1.00000  0.66042  0.70880

debug> a3
a3 =

 0.88866  0.90743  0.92330  0.93665
 0.83818  0.86028  0.87980  0.89692
 0.92341  0.93858  0.95090  0.96085

debug>
```

↑ 0 ↓ · flag



Tom Mosher COMMUNITY TA · 2 days ago

Gee, once again, we've got square matrices!

↑ 0 ↓ · flag

Jonne Zutt · 2 days ago

I have the exact same results for a3 (although transposed, but if the later code deals with that correctly it shouldn't matter).

I tried both with for-loops to compute J from that, as well as a vectorized implementation. Both give me  $J = 7.2127$  instead of  $J = 7.407$ .

The top 3 rows below are my values for the not selected classes, the bottom row for the selected classes. Summing all these values twice and dividing by m gives my (wrong) value of J. Any ideas?

2.195161	1.821258	2.569345
2.379763	1.968129	2.789980
2.567918	2.118596	3.013861
0.065446	0.108791	0.039936

↑ 0 ↓ · flag



Tom Mosher COMMUNITY TA · 2 days ago



If your `a3` is the wrong shape, then your `y_matrix` must be the wrong shape also, else you'd be getting a non-compliant argument error.

Note that this unit test has nothing to do with selecting classes. That's not the job of the cost function. There shouldn't be anything in your `nnCostFunction()` that selects the "best" classifier.

I don't understand where your table of values is coming from.

↑ 0 ↓ · flag



Tom Mosher COMMUNITY TA · 2 days ago



Oh, I get it now. That's the result of right-side of the cost calculation, everything to the right of the double-summation

symbol. I get (edit: some of) those same values.

The concept of "selected classes" is does not apply to the cost function.

Your first row and first column are correct. Most of the remaining values are incorrect.

Here's what I have for the cost factors prior to the double-sum.

```
debug> j1-j2
ans =
2.195161  2.379763  2.567918  0.065446
1.821258  0.150495  2.118596  2.272228
2.569345  2.789980  0.050348  3.240371
```

My 'j1' value is the left-side term, and 'j2' is the right-side term. The double-sum is 22.22, scaling by 1/m gives 7.407.

My "left side" involves -y\_matrix and log(a3), "right side" involves (1-y\_matrix) and log(1-a3).

Here's my j1 and j2:

```
ans = 1.821258
debug> j1
j1 =
0.00000  0.00000  0.00000  0.06545
0.00000  0.15050  0.00000  0.00000
0.00000  0.00000  0.05035  0.00000

debug> j2
j2 =
-2.19516  -2.37976  -2.56792  -0.00000
-1.82126  -0.00000  -2.11860  -2.27223
-2.56934  -2.78998  -0.00000  -3.24037
```

0 · flag

Jonne Zutt · 2 days ago

Ah, very interesting. My j1 (transposed) and j2 (transposed) look like:

j1' =

0.00000	0.00000	0.00000	0.06545
0.00000	0.00000	0.00000	0.10879

0.00000	0.00000	0.00000	0.03994
---------	---------	---------	---------

$j2' =$

-2.19516	-2.37976	-2.56792	-0.00000
-1.82126	-1.96813	-2.11860	-0.00000
-2.56934	-2.78998	-3.01386	-0.00000

We have the same values in case the values are non-zero.

That means my y-matrix is wrong.

Because in a3 the element corresponding to k=4 is the maximum, I should it should look like this:

0	0	0
0	0	0
0	0	0
1	1	1

(or transposed). But probably that's where my error is (as your zero values are more spread over the matrix).

Thanks for pointing me in the right direction,  
Jonne.

(edit: I think I know what's wrong, silly mistake, I wasn't using the input y - but instead used a computed prediction)

↑ 0 ↓ · flag



Tom Mosher COMMUNITY TA · 2 days ago

There should only be one '1' value set for each training example.

For the 2a unit test, here's my  $y\_matrix$ :

$y = [4 \ 2 \ 3]$

```
octave-3.2.4.exe:37> eye(4)(y,:)
ans =
 0   0   0   1
 0   1   0   0
 0   0   1   0
```

↑ 0 ↓ · flag

---

+ Comment

Jlee · 2 days ago 

For 4a unit test, I got correct values for  $\Theta_2\text{grad}$ , but different result for  $\Theta_1\text{grad}$  (Initial 6 elements of grad) as following:

0.264755  
0.351675  
-0.047155  
-0.061111  
-0.052656  
-0.069343

...

For  $\delta_2$  calculation, I did follow the equation in Step 3, which involves element-wise multiplication and sigmoid Gradient function. My sigmoid gradient function produces correct results.

What would be possible error?

↑ 0 ↓ · flag



Alex McCarthy · 2 days ago 

I'm not sure :/

What are your delta 2 values when iterating? Mine look like this for the three sample inputs in 4a:

d2 =

0.79393

1.05281

d2 =

0.73674

0.95128

d2 =

0.76775

0.93560

↑ 1 ↓ · flag



Tom Mosher COMMUNITY TA · 2 days ago

@Jlee

In Test 4a, since the a1 and a2 matrices are square, it's possible that you've got an incorrect transposition or order of operands when you're computing the gradients.

↑ 0 ↓ · flag

Jlee · 2 days ago

My delta 2 values are incorrect.

d2 =

0.26793 0.35824

0.34195 0.45110

0.18439 0.24569

I got correct J, sigmoid gradient, and Theta2\_grad outputs, thus I think a2, d3 and D2 operations are fine.

For d2 calculation, I did element multiplication between a3 and sigmoid gradient of z2 before multiplication with theta2 to match the dimension to match operation dimension.

I've been checking diverse possibilities, but I will try more.

Thanks Alex and Tom,

↑ 0 ↓ · flag



Tom Mosher COMMUNITY TA · 2 days ago



The key points for getting the correct d2 values:

- having the correct d3 values.
- omitting the use of the entire first column of Theta2.
- correctly-working sigmoidGradient() function.
- correct values for z2 - it doesn't include the use of sigmoid()

↑ 1 ↓ · flag

Jlee · a day ago



Thank you!!

I got correct result after editing Theta2 and using correct values for z2.

↑ 1 ↓ · flag

---

+ Comment



Cody · a day ago

for 4a, I calculated z2, a2, z3 and a3, when applying back propagation according to the formula:

$$\delta^{(2)} = (\Theta^{(2)})^T \delta^{(3)} * g'(z^{(2)})$$

*Theta2* is a 4\*3 matrix, thus *Theta2'* is a 3\*4 matrix

*delta3* is a 4\*1 vector, then *Theta2' \* delta3* makes a 3\*1 vector.

*z2=X\*Theta1'*, so it is a 3\*2 matrix, for each training example, in my mind, one row of *z2* is used, which is a 1\*2 vector, then *sigmoidGradient* also returns a 1\*2 vector.

So I get a [3\*1 vector] .\* [1\*2 vector], which doesn't make sense, what's the wrong step?

0 · flag



Cody · a day ago

By reading Tom's answer to Jlee, I think I know the problem. I should remove the first column of *Theta2*, and for *sigmoidGradient(z2)*, I think the vector needs a transpose too.

0 · flag

Bin Lang · a day ago

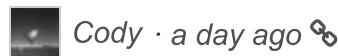
I am debugging my unit test 1a, this is my output hypothesis, is this correct?

ans =

```
0.88866 0.90743 0.92330 0.93665
0.83818 0.86028 0.87980 0.89692
0.92341 0.93858 0.95090 0.96085
```

0

· flag



Still, don't quite get the formula, I got delta3 and delta2 as follows:

*delta3:*

-----  
0.888659  
0.907427  
0.923305  
-0.063351

0.83818  
-0.13972  
0.87980  
0.89692

0.923414  
0.938578  
-0.049102  
0.960851

*delta2:*

-----  
0.79393  
1.05281

0.73674  
0.95128

0.76775

0.93560

is that correct?

And what's the point of 'accumulating the gradient' of step 4 in the pdf for the formula:

$$\Delta^{(l)} = \Delta^{(l)} + \delta^{(l+1)}(a^{(l)})^T$$

↑ 0 ↓ · flag

Jlee · a day ago 

Your delta2 and delta3 are correct.

Tom provides detailed explanation about backpropagation process including delta2 (d2) calculation in the following thread. This is very helpful.

[https://class.coursera.org/ml-008/forum/thread?thread\\_id=1146](https://class.coursera.org/ml-008/forum/thread?thread_id=1146)

↑ 0 ↓ · flag

 Cody · a day ago 

Thanks Jlee.

I read that post just now and found in the thread that a2 is 3x3, while I got 3x4.

But my J value is correct (all tests from 1~3 are correct), and due to this a2 dimension problem, for grad value, only the first two are correct.

↑ 0 ↓ · flag

 Tom Mosher COMMUNITY TA · a day ago 

@Cody:

*It's very easy to get the correct J value with the wrong size of objects, since you're summing everything together.*

*You're mixing two different implementations. The equations you posted are for the iterative (for-loop over each training example) implementation.*

*The tutorial addresses the vectorized method (all training examples at once).*

*That's why your delta3 is the wrong size.*

↑ 0 ↓ · flag



Tom Mosher COMMUNITY TA · a day ago



@Bin Lang:

See this post:

[https://class.coursera.org/ml-008/forum/thread?thread\\_id=1056#comment-4105](https://class.coursera.org/ml-008/forum/thread?thread_id=1056#comment-4105)

↑ 0 ↓ · flag



Cody · 11 hours ago



@Tom,

I read carefully about your tutorial and the pdf, and found my problem.

When calculating Delta1, it's  $d2*a1$ , where  $a1$  is actually the same as  $X1$ , while I used  $a2$  instead, and later used  $a3$  to calculate Delta2...

I got the correct result finally, thanks~

↑ 1 ↓ · flag



Tom Mosher COMMUNITY TA · 11 hours ago



Nice work!

↑ 0 ↓ · flag

---

+ Comment

Murtuza I B · 18 hours ago 

Test 4a (Neural Network Gradient (Backpropagation)):

I am confused with Theta2\_grad calculations.

My thinking is Theta2\_grad = Delta3 = (a3 - y).

Here y is logical matrix. So as per Unit Test case 4(a), y should be as below

0 0 0

0 1 0

0 0 1

1 0 0

So y is 4x3 matrix

And a3 is output, which is also 4x3 of values less than 1.

So, If do (a3 - y) then I get negative values for element where y has value 1.

But in result of test case, Theta2\_grad is all positive.

Where am I going wrong?

 0  · flag



Tom Mosher COMMUNITY TA · 18 hours ago 

If 'y' is a logical array, then every row must have at a single '1' value. Your first row doesn't.

Your concept of Theta2\_grad is incorrect.

I recommend you review the Ex4 Tutorial on backpropagation. It's available in the Forum.

 0  · flag

Murtuza I B · 6 hours ago 

Thank you Tom.

If you take transpose of y, it will be fine.

Anyway, after reading reviewing Ex4 Tutorial, I realized my mistake.

↑ 0 ↓ · flag

+ Comment

New post

To ensure a positive and productive discussion, please read our forum posting policies before posting.

B	I	≡	≡	🔗 Link	<code>	🖼 Pic	Math	Edit: Rich ▾	Preview

Make this post anonymous to other students

Subscribe to this thread at the same time

Add post