Forums / Programming Exercises / Programming Exercise 8

Help

One line implementation of Cost and Gradient computation

Subscribe for email updates.

☑ RESOLVED



Sort replies by: Oldest first Newest first Most popular

CostFunction × gradient × Elegant × + Add Tag

Jianwei Tao · 7 days ago %

For a one variable linear regression model, the cost is

$$J = \frac{1}{2} (\theta x - y)^2$$

So I guess the J in this exercise is

$$J = \frac{1}{2} \sum_{i,j} (X\Theta - Y)^2$$

As the sum is for non-trivial items of Y, so it shall be modified a little bit:

$$J = \frac{1}{2} \sum_{i,j} (X\Theta - Y)^2 . *R$$

This is very easy to be transformed into a OCTAVE representation.

For a one variable linear regression model, the gradient is

$$J' = (\theta x - y)x$$

So I guess the J' (the gradient) in this exercise is

$$Grad(\Theta) = (X\Theta - Y)X$$

$$Grad(X) = (X\Theta - Y)\Theta$$

As the sum is for non-trivial items of Y, so it shall be modified a little bit:

$$Grad(\Theta) = ((X\Theta - Y). *R)X$$

 $Grad(X) = ((X\Theta - Y). *R)\Theta$

These are also very easy to be transformed into OCTAVE representations

I tried and succeeded.

The regularized version of these equations are:

$$J = \frac{1}{2} \sum_{i,j} (X\Theta - Y)^2 \cdot *R + \frac{\lambda}{2} \sum_{i,j} \Theta \cdot *\Theta + \frac{\lambda}{2} \sum_{i,j} X \cdot *X$$

$$Grad(\Theta) = ((X\Theta - Y). *R)X + \lambda\Theta$$

 $Grad(X) = ((X\Theta - Y). *R)\Theta + \lambda X$

Is it simple and elegant?

You are not the first one to figure it out, but you are the first one to relase it out. Do not get too much excited, please.

Thank you very much for your comment, Shawn!

Of course I am not the first one. There are definitely countless people who have worked out better solutions. I just wanted to share my trivial efforts with co-learners, and I also hope that others could share their better approaches, and learn something interesting from them.

I am confused about the dimensions for Theta_grad and X_grad.

$$Grad(\Theta)=((X\Theta-Y).*R)\Theta$$

$$Grad(X)=((X\Theta-Y).*R)X$$

I thought the size of $((X\Theta-Y).*R)$ is 1682 by 944 (number of movies x number of people). How can we multiply both by Theta (size 944 by 10) and by X (size 1682 by 944)?

Dimitri Liakhovitski · 5 days ago %

Isn't there an error in the formulas above? Shouldn't it rather be:

Grad(
$$\Theta$$
)=(($X\Theta$ - Y).* R) X
Grad(X)=(($X\Theta$ - Y).* R) Θ

Sundeep Laxman · 5 days ago %

You're correct, Dimitri.

Jianwei Tao · 5 days ago %

Thank you, Dimitri. You are perfectly right. I have changed my writing fault.

I am coufused by $Grad(\Theta)$. The result of $((X\Theta-Y).*R)$ is number_movies * number_users,how could it time X (num_movies*num_features),the dimensions for Grad(X) must equal to X (num_movies* num_features)

A transpose will make it work. Of course, $Grad(\Theta)$ has the same size of Θ , Grad(X) has the same size of X.

Anonymous . a day ago %

One other thing....why compute X*Theta - Y over and over? Compute it once and use it for calculating J and both the gradients.

Raúl Solera Rallo · 21 hours ago %

I think there are a couple of mistakes taking into account the dimensions of each matrix:

- $Y = 1.682 \times 944$
- $\Theta = 944 \times 10$
- $X = 1.682 \times 10$

So, the first term of the equation sould be $X^*\Theta'$ to get a 1.682 x 10 * 10 x 944 = 1.682 x 944 matrix compatible with Y and R.

And after you get the 1.682 x 944 matrix: $M = ((X * \Theta' - Y). *R)$ then you have to transpose it to multiply it by X (as this is a 1.682 x 10 matrix).

So I think that the correct final equations should be as follows:

- $M = ((X * \Theta' Y). *R)$
- $Grad(\Theta) = M' * X + \lambda \Theta$
- $Grad(X) = M * \Theta + \lambda X$

Thank you for opening this post.



+ Comment



Saddique Khan · 4 days ago %

@Jianwei Tao

your work is tremendously amazing. There is an other mistake. I observed it. If it is not then clarify it. In Regularized cost function the last parameter need to sum two times because the simple J has 1x1 dimension and Last parameters have 1x3 dimensions. If you will sum it up again then it will also become 1x1 element. I found this mistake while I was simulating the regularized cost function. Thanks btw,

+ Comment

Kai Mysliwiec · 2 days ago %

The vectorized one-liners are pretty fast compared to using two nested for-loops. The slow version took over half an hour, the fast one less than 5 seconds to compute the final recommendations.

+ Comment



Daniel Rodrigo Ramírez Rebollo · 2 days ago %

Great implementation, I really like how you figured out how to just choose the rated movies.

Thanks for sharing.

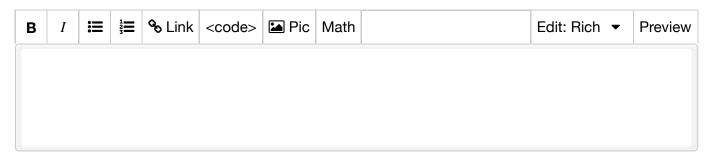
Best Regards from MExico



+ Comment

New post

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



This thread is marked as resolved. Staff are no longer monitoring this thread. If the problem is not fixed, please start a new thread here to let staff know that there is still a problem.

- Make this post anonymous to other students
- ✓ Subscribe to this thread at the same time

Add post