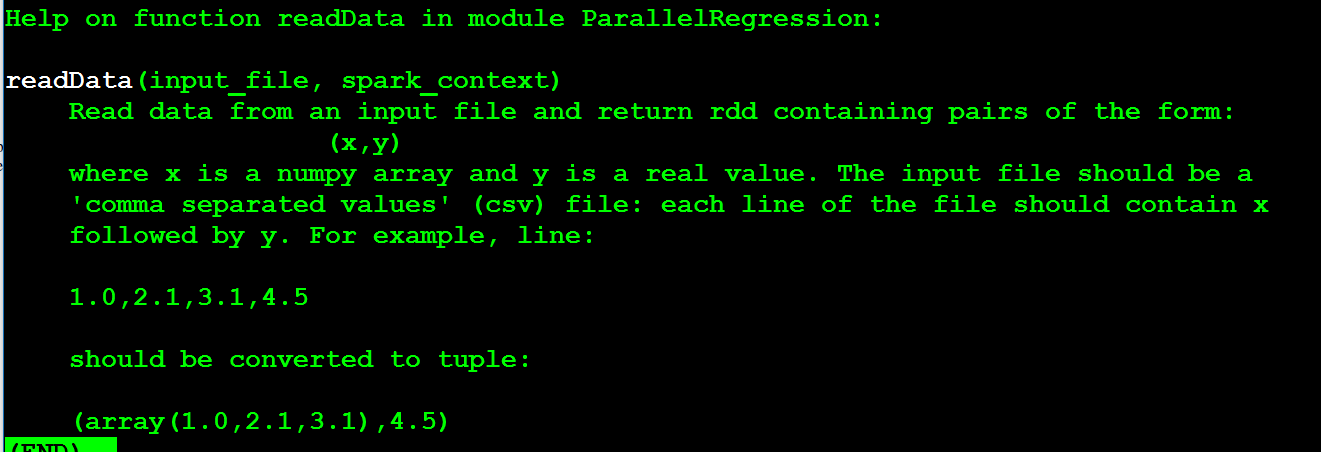
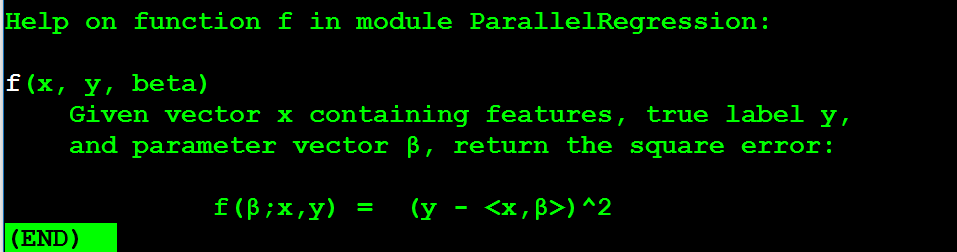
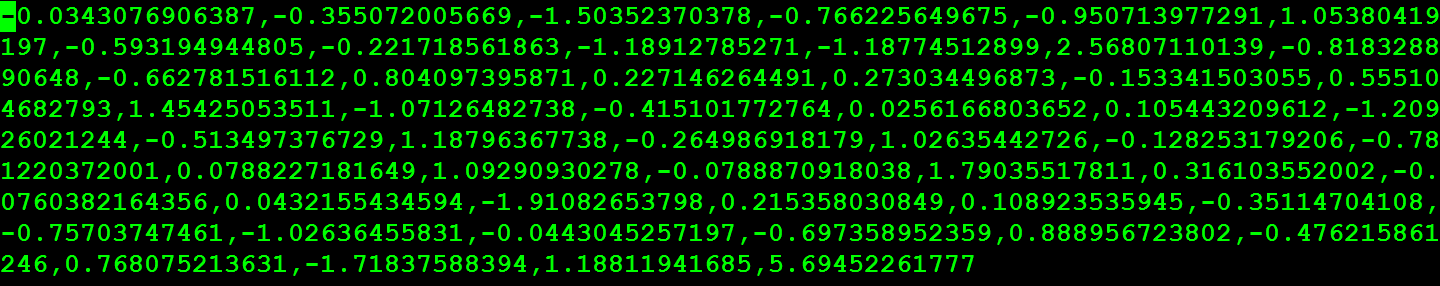
**Q0**

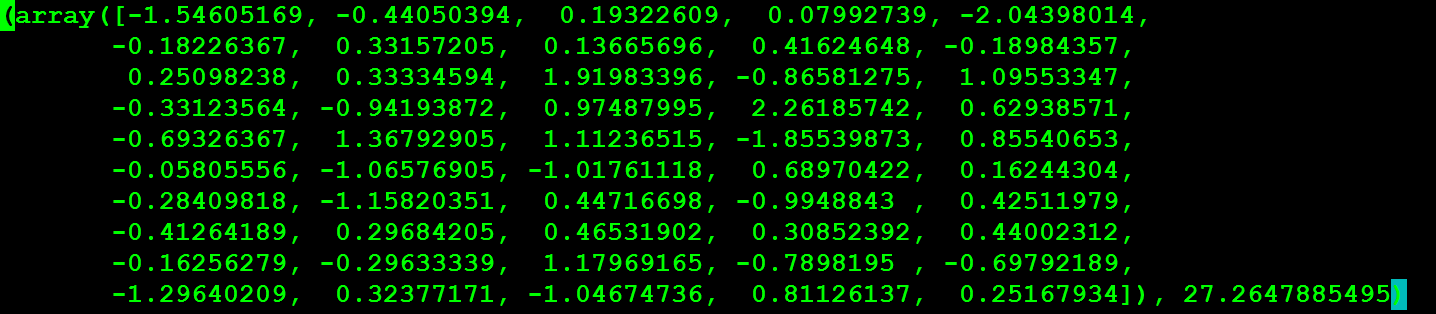
Below is the part print by help(PR.readData)

Below is the part print by help(PR.f)

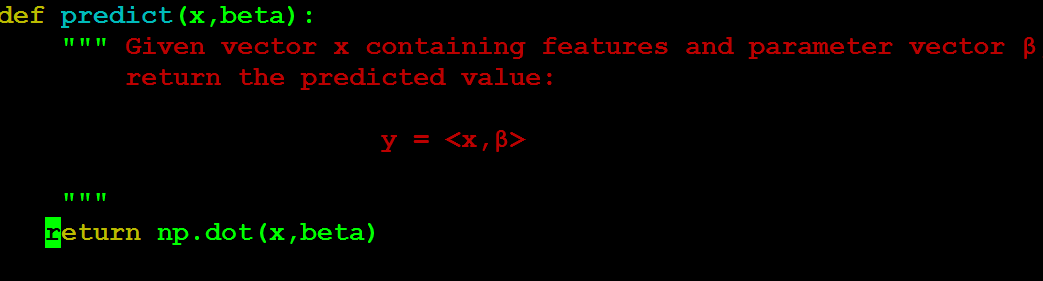
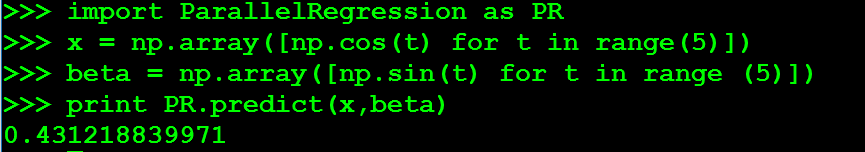
These two parts is printed because they are between """…""" symbol under the function. And they will show when you type help

Below is one example of data in small.test. It is a set of numbers.



Below is one example of data in the RDD. It’s a tuple with the form (x,y). It contains all the value except the last one as x and the last one as y

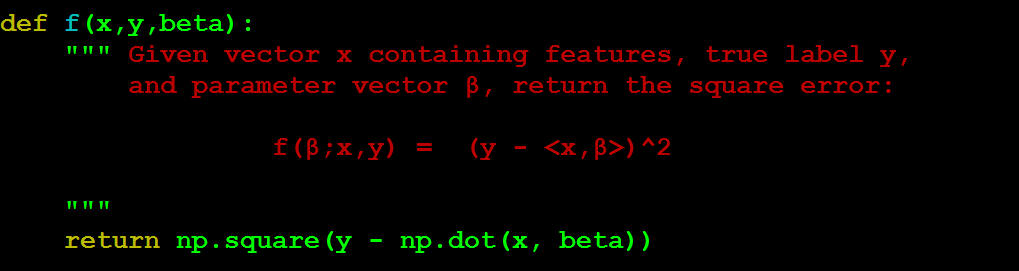
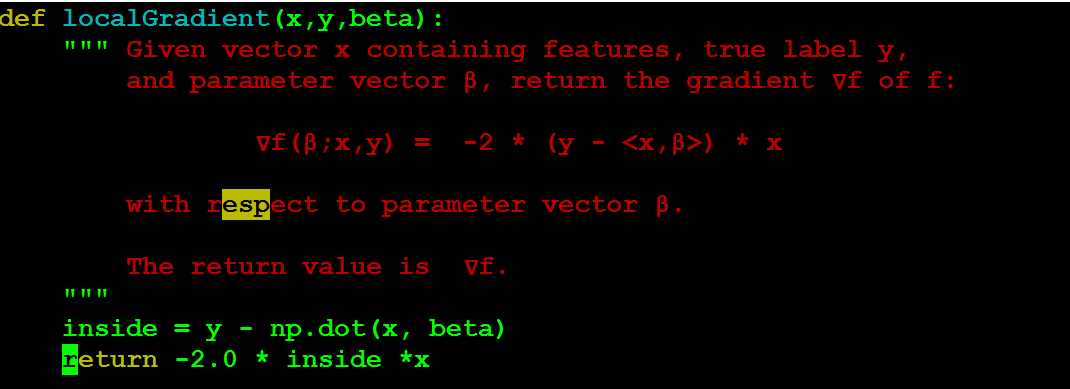
**Q1**

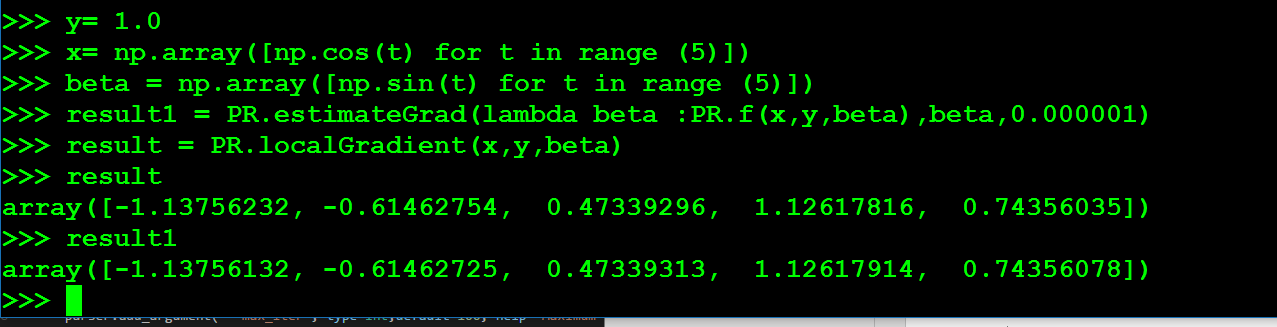
1. Below is my code for predict
2. Below is my executing of predict

**Q2**



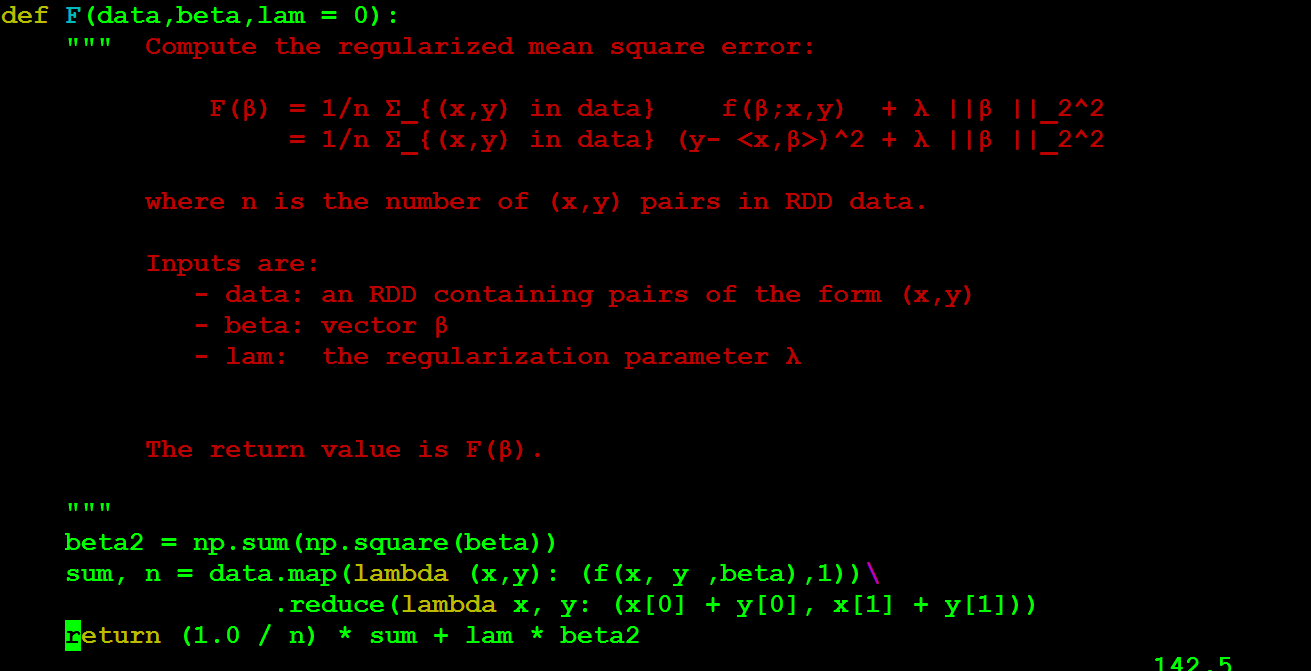
=

1. Below is my code for function f
2. Below is my code for localGradient

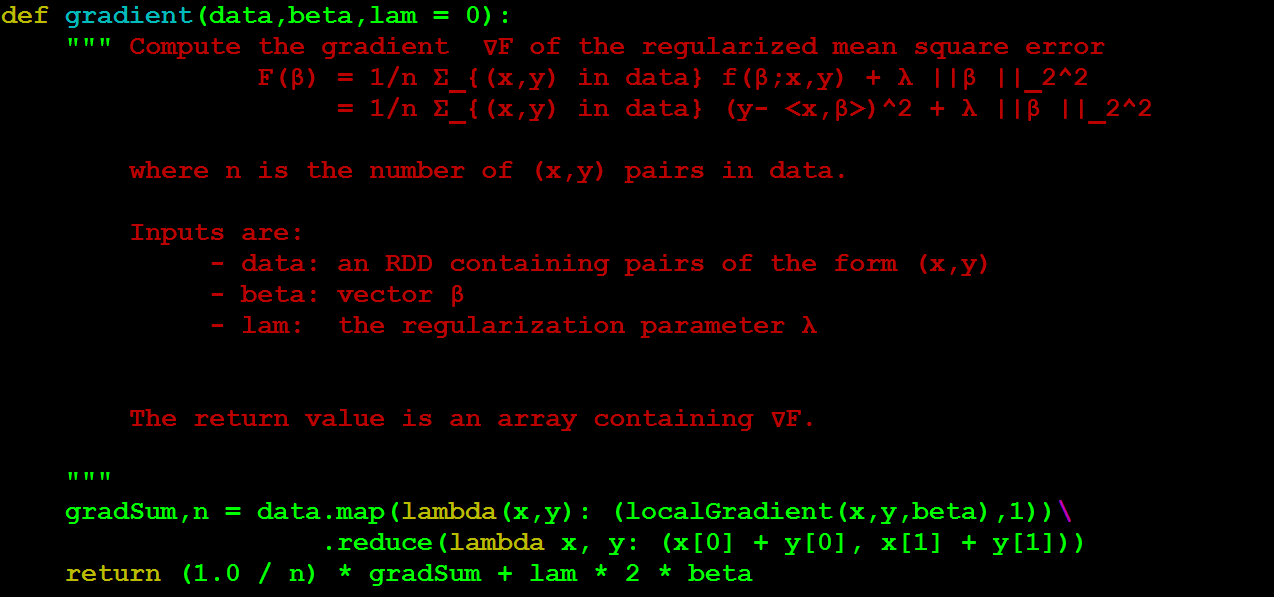
d) Below is the execution of localGradient and combine of f with estimateGrad

As you can see the result is basically the same. So, that the localGradient is correct since the correctness already been tested by localGradient agrees with the estimate produced by estimateGrad When 󠆷δ is small.

**Q3**

 a) Below is my code for function F

b) Below is my code for function gradient



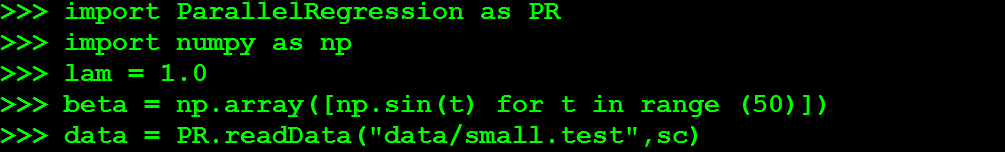
>>> import ParallelRegression as PR

>>> import numpy as np

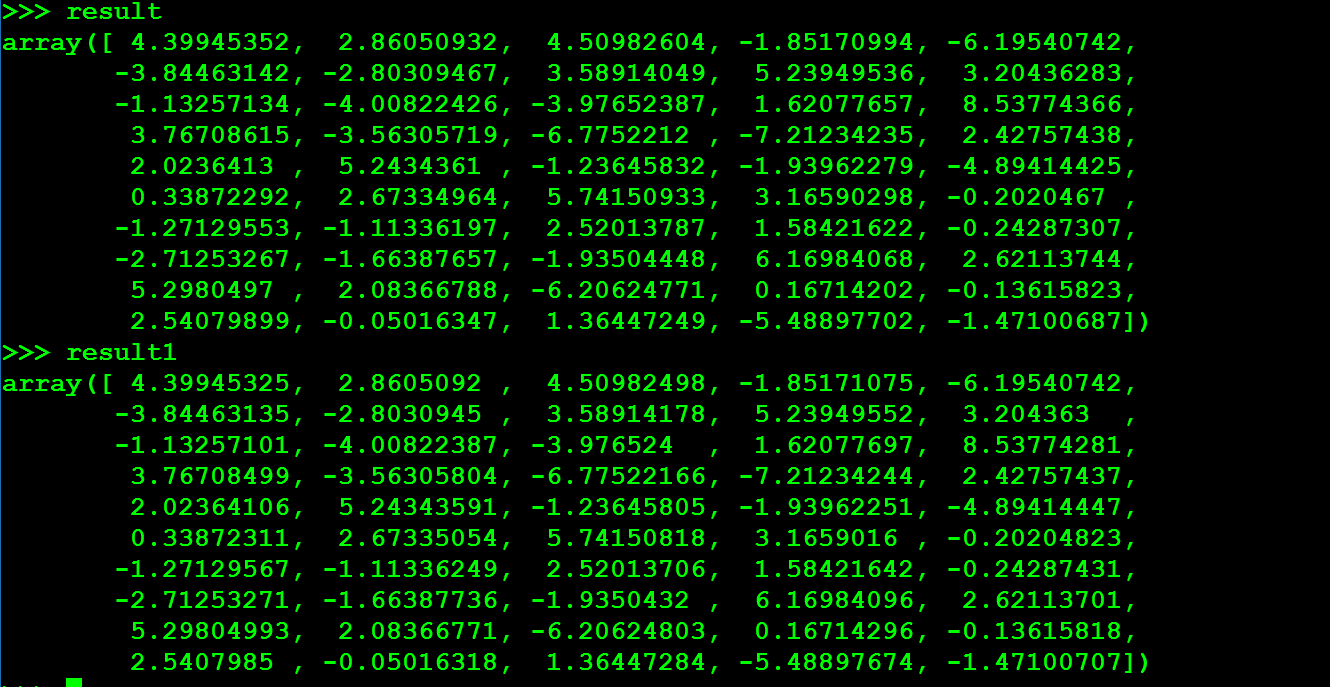
>>> beta = np.array([np.sin(t) for t in range(50)])

>>> lam = 1.0

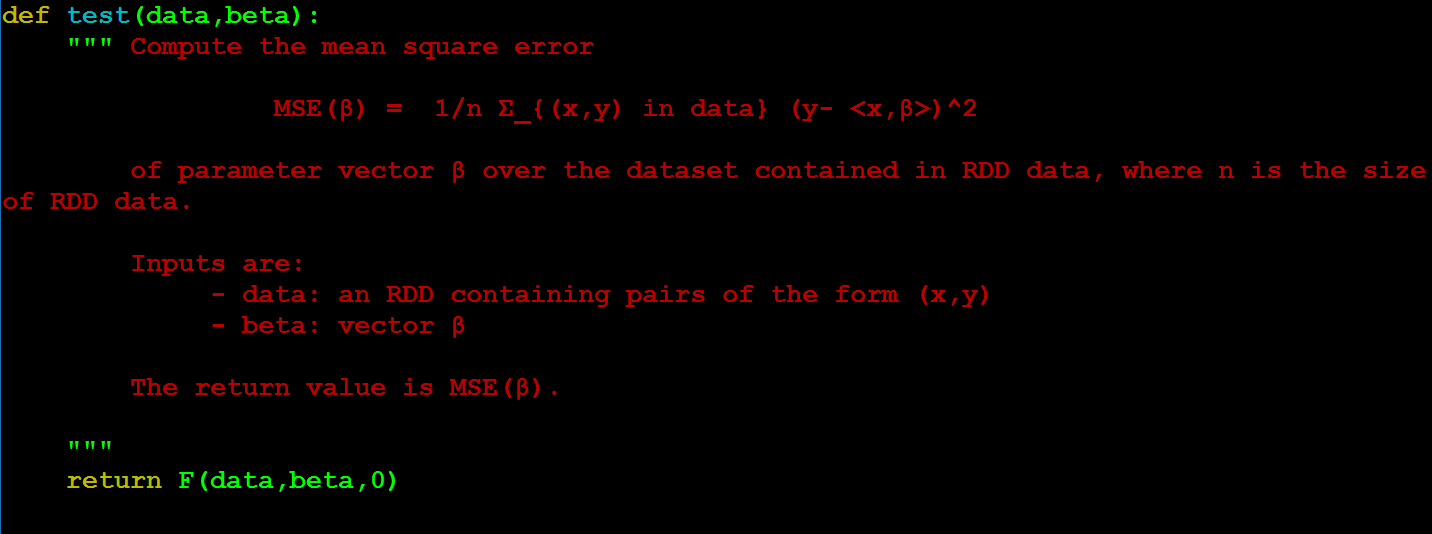
>>> data = PR.readData("data/smal.test", sc)

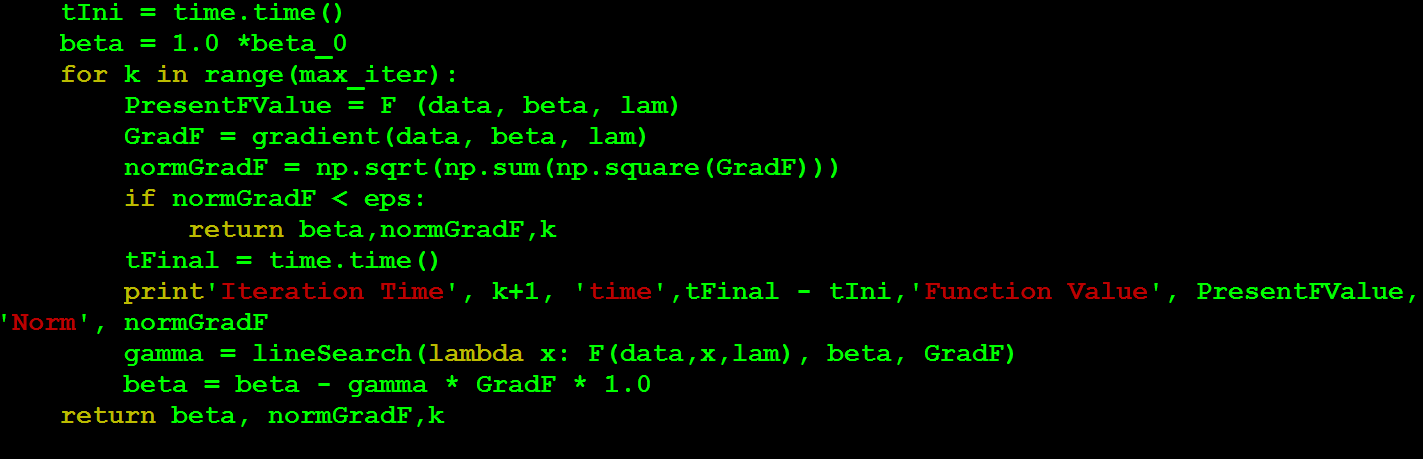
 res1 = PR.estimateGrad(lambda beta: PR.F(data, beta, lam), beta, 0.0000001)

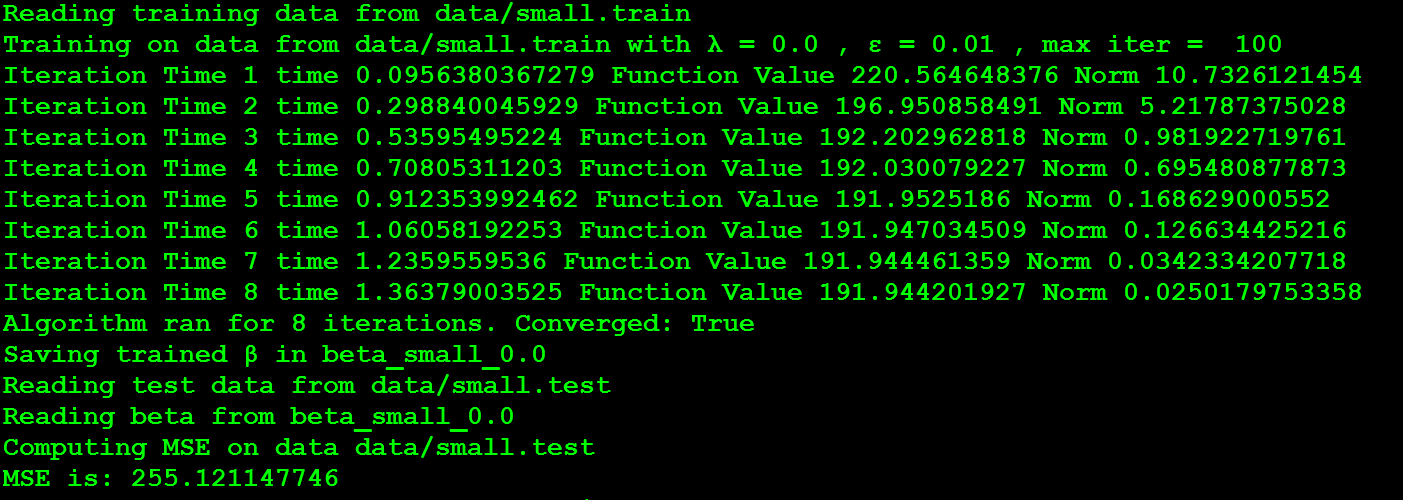
c) Below is my execucation for gradient and combine of estimateGrad with F

As you can see result. Result and Result 1 are basically the same. Which can prove my correctness of gradient is the gradient of function F.

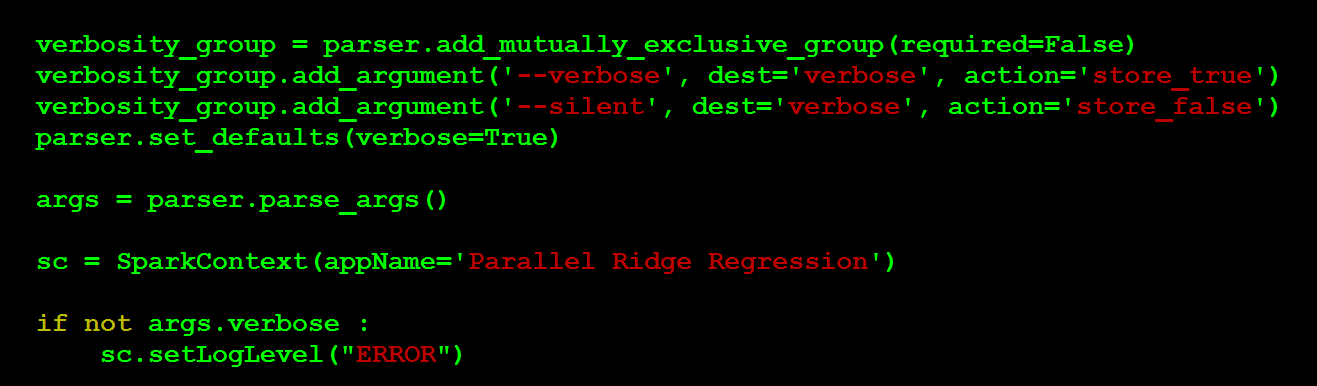
**Q4**

1. Below is my code for test
2. Below is my code for train



1. Below is the execution

--silent will reduce the verbose runtime. And in the code

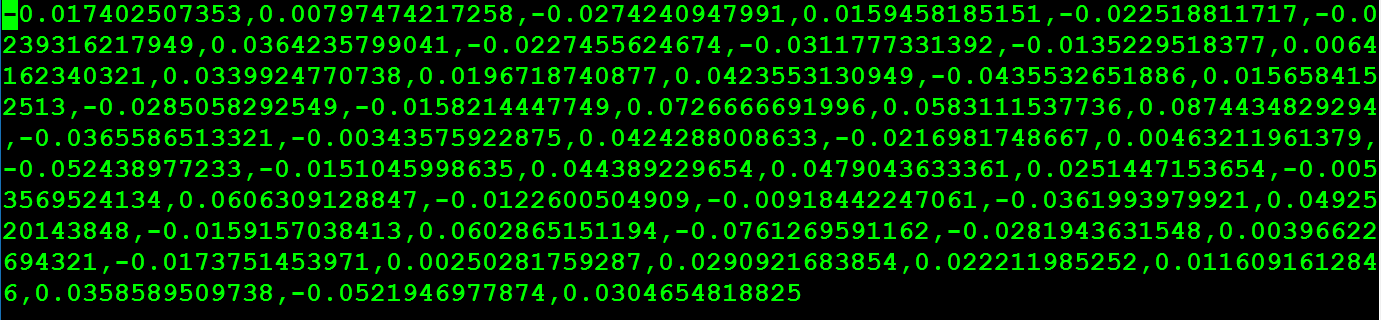
 We can see that there are two modes –silent and –verbose. We can only choose one of them. If we choose –silent the sc.setLogLevel will be ERROR Level which is different than the default value ALL

**Q5**

1. Below are the table for lambda = 0.0-20.0 for small.test and small.train

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Lam=0.0** | **1.0** | **2.0** | **3.0** | **4.0** | **5.0** | **6.0** |
| 255.121 | 233.008 | 229.538 | 228.312 | 227.731 | 227.407 | 227.203 |
| **7.0** | **8.0** | **9.0** | **10.0** | **11.0** | **12.0** | **13.0** |
| 227.067 | 226.972 | 226.900 | 226.847 | 226.804 | 226.771 | 226.744 |
| **14.0** | **15.0** | **16.0** | **17.0** | **18.0** | **19.0** | **20.0** |
| 226.721 | 226.702 | 226.685 | 226.671 | 226.659 | 226.648 | 226.639 |

Vector β that attains the smallest test MSE is shown below



b) Below are the table for lambda = 0.0-10.0 for big.test and big.train

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **0.0** | **1.0** | **2.0** | **3.0** | **4.0** | **5.0** |
| 4151.494 | 4000.099 | 3977.838 | 3971.328 | 3968.868 | 3967.833 |
| **6.0** | **7.0** | **8.0** | **9.0** | **10.0** |
| 3967.386 | 3967.214 | 3967.177 | 3967.208 | 3967.272 |