**Q1:**

**a)**

**b)**

Suppose there’s a vector and at u=v=0 we have

We don’t know if it’s greater or equal than 0. So l(u ,v) is not convex.

**c)**

If

For all i and j, u\* and v\* would minimize the RSE but since RSE is not convex, u\* and v\* can only be the local minima instead of global minima.

**d)**

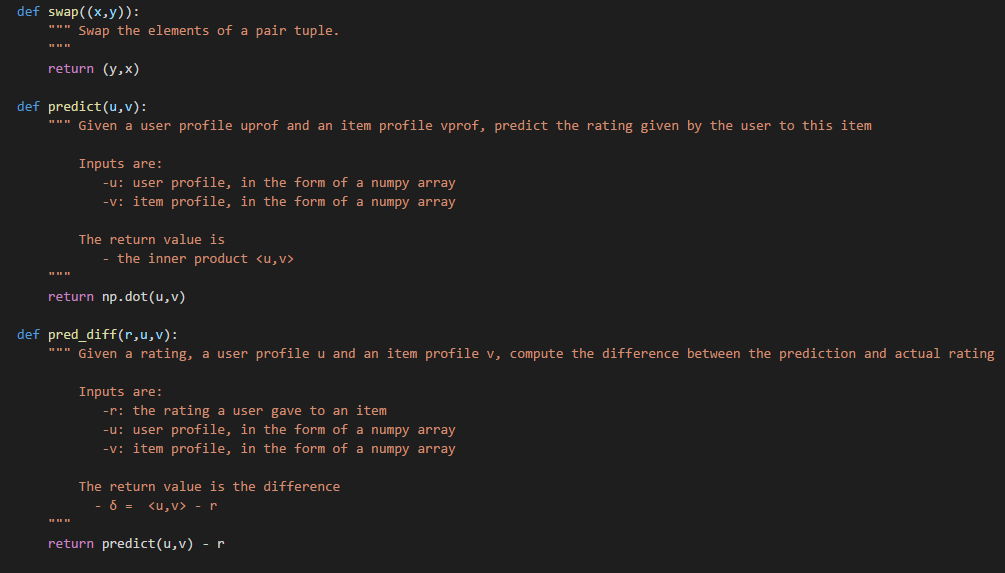
if ui = vj = 0 then

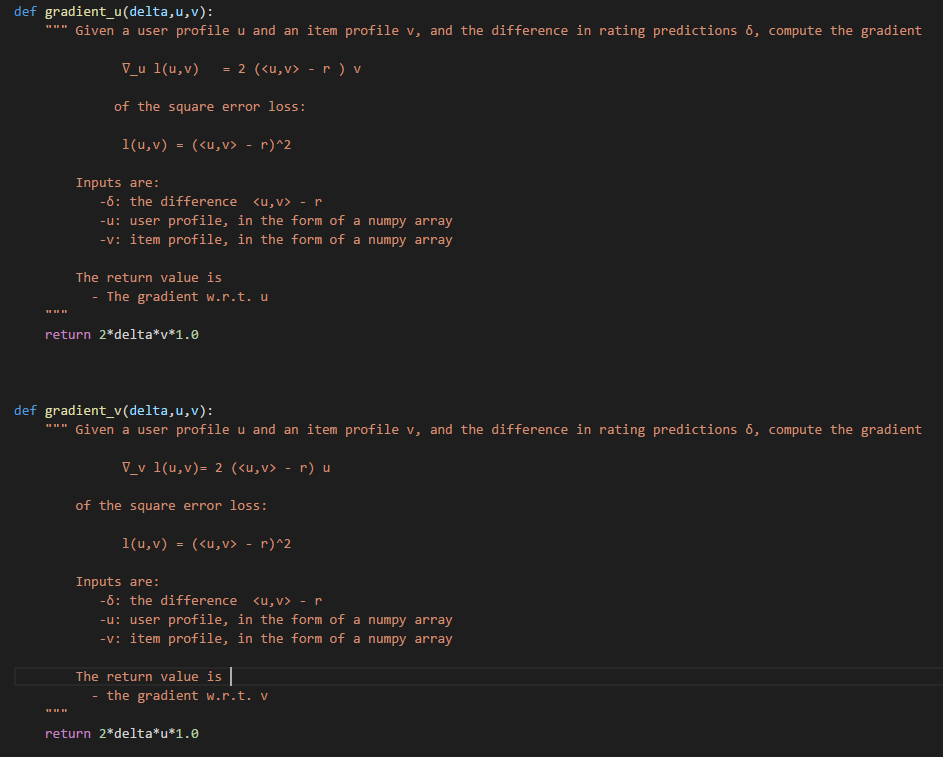
= 0

= 0

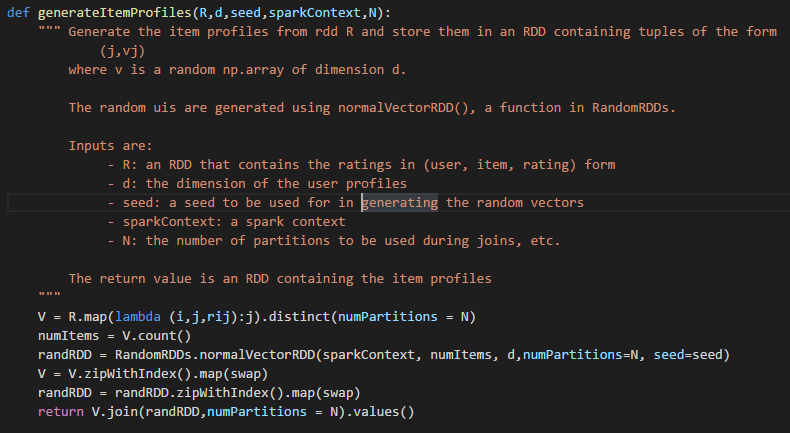
**Q2:**

**a)**

**b)**

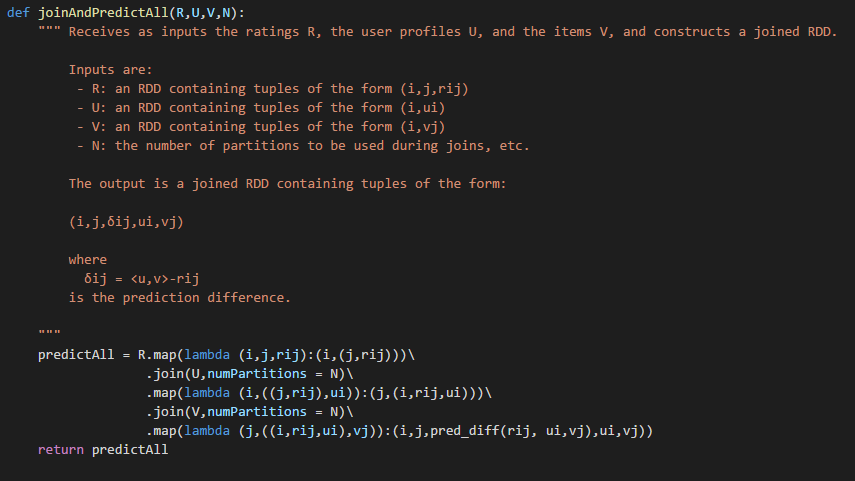


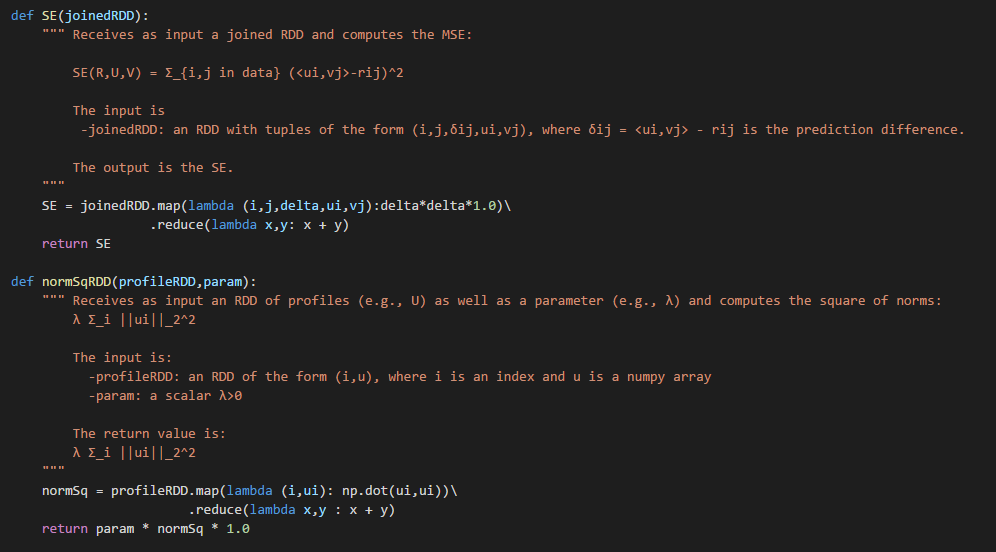
**c)**



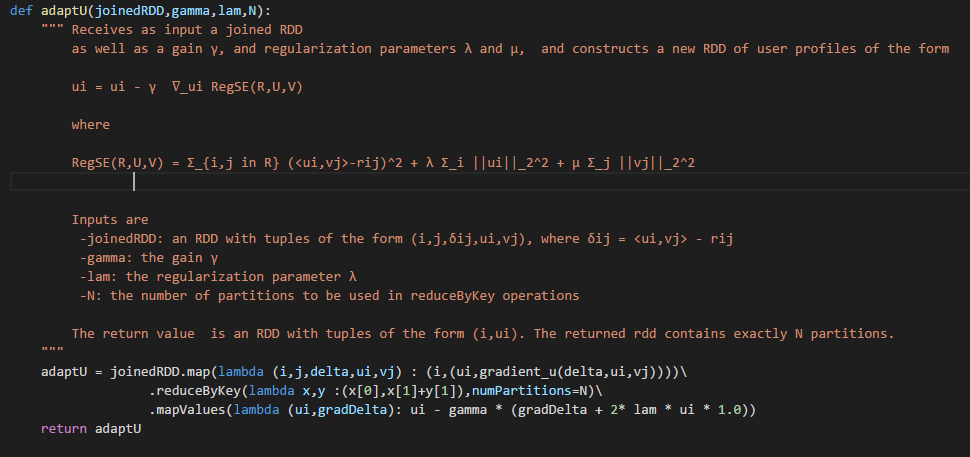
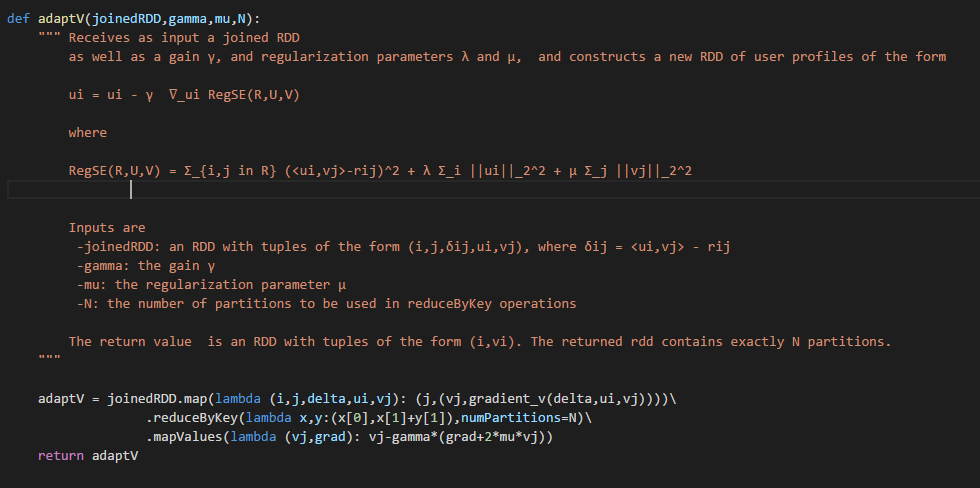
Since RSE is not convex so gradient descent would converge to a local minimum so the initial point can decide which local minimum to converge to. If we set all profiles to be zero. The gradient descent also converge to zero since zero is a stationary point so we would have no other information.

**Q3:**

**a)**

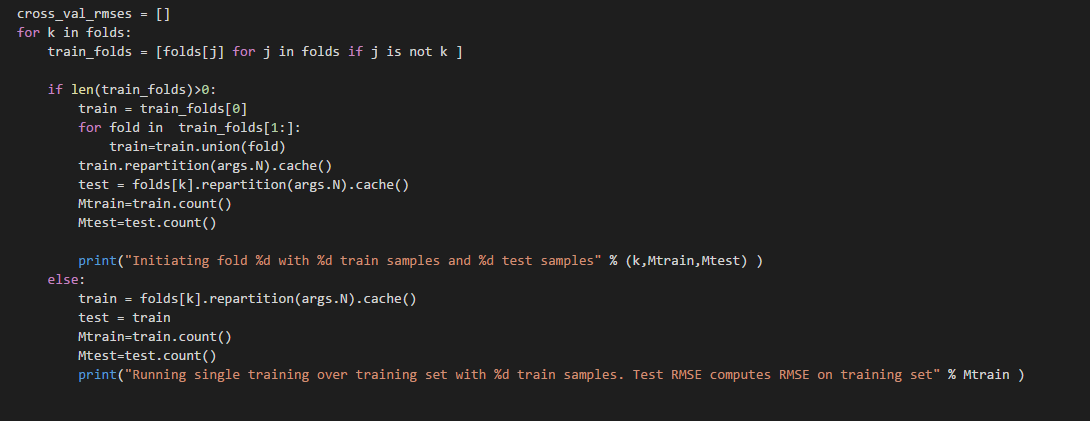
**b)**

**c)**



**Q4:**

**a)**



In the code, we can see that k-th fold can construct a training list of the other folds. Then we split the data into training RDD and testing RDD in N partitions. If the training list is empty, the training RDD and testing RDD would be the same. And after all we print the amount if data in training RDD and test RDD separately

**b)**

The step size is defined as

G is the gain and k p is gain exponent.

**c)**

When d is 6 we get the best result.

**d)**

gain = 0.1 power = 0.0

This is a constant step size which is much larger when gain is 0.001 and power is 0.2. It won’t converge at the end and the RMSP is huge

gain = 0.001 power = 1.0

The step size is smaller than the step size when gain is 0.001 and power is 0.2. So It converges very slow. With 20 steps, it doesn’t reduce the error significantly.

**e)**

****