DATA MINING
ASSIGNMENT – 2
Pramod Maharjan (069-BCT-530)

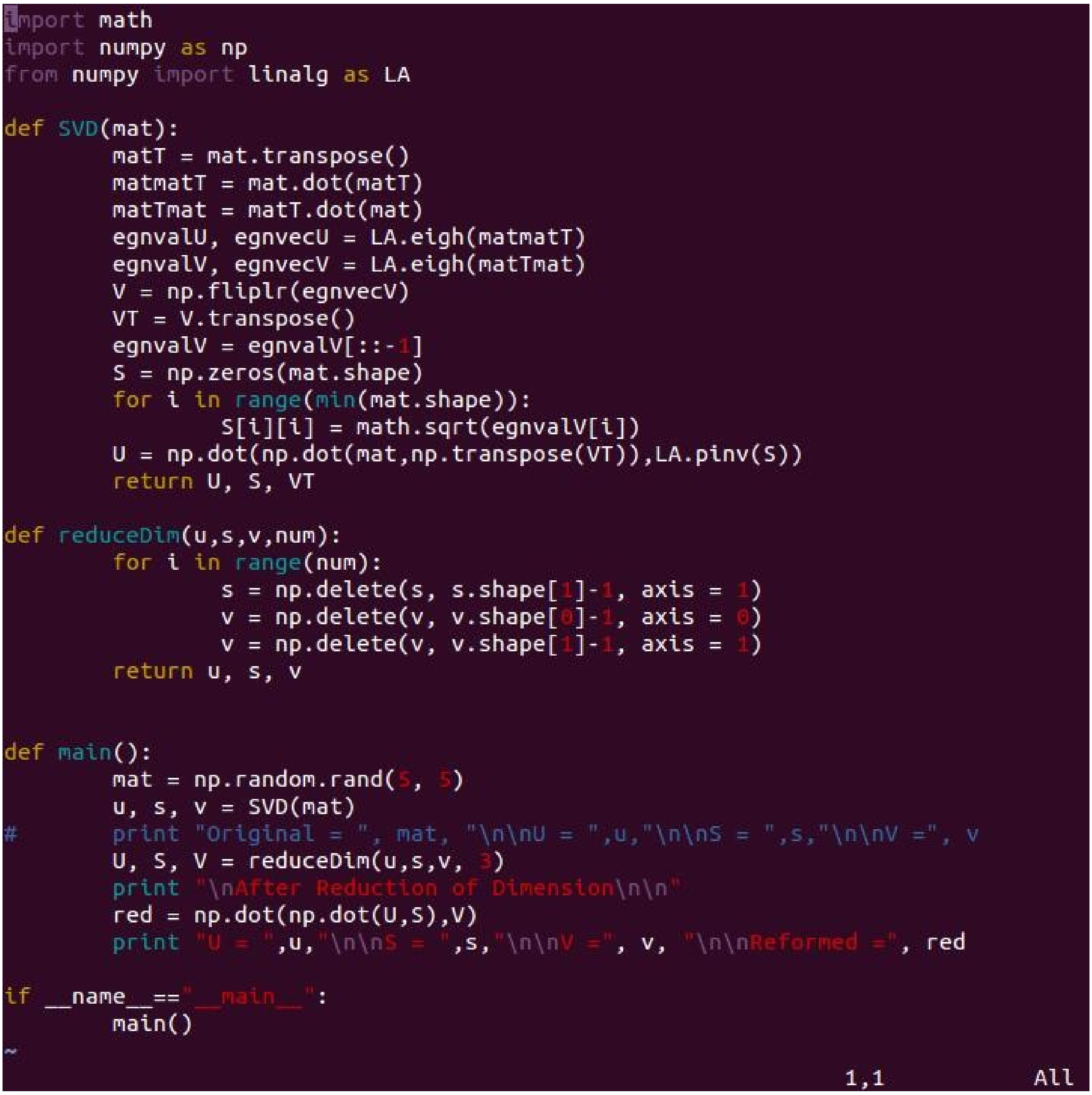
Singular Value Decomposition

SVD is based on a theorem from linear algebra which says that a rectangular matrix A can be broken down into the product of three matrices - an orthogonal matrix U, a diagonal matrix S, and the transpose of an orthogonal matrix V . The theorem is usually presented something like this:

Amn = Umm Smn V T nn

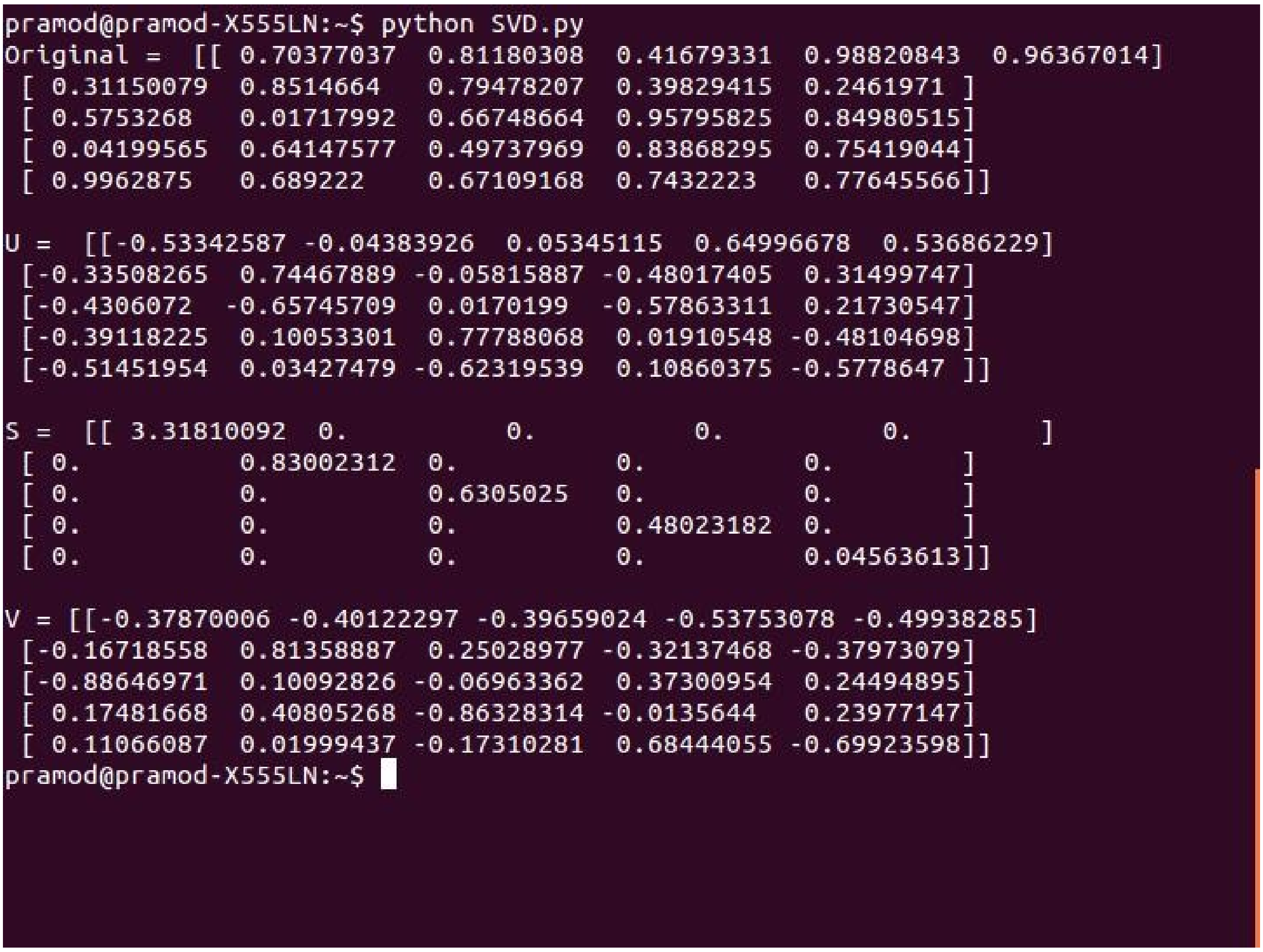
where UTU = I, V T V = I; the columns of U are orthonormal eigenvectors of AAT , the columns of V are orthonormal eigenvectors of ATA, and S is a diagonal matrix containing the square roots of eigenvalues from U or V in descending order.

The code for SVD is given below:



```
import math
import numpy as np
from numpy import linalg as LA

def SVD(mat):
        matT = mat.transpose()
        matmatT = mat.dot(matT)
        matTmat = matT.dot(mat)
        egnvalU, egnvecU = LA.eigh(matmatT)
        egnvalV, egnvecV = LA.eigh(matTmat)
        V = np.fliplr(egnvecV)
        VT = V.transpose()
        egnvalV = egnvalV[::-1]
        S = np.zeros(mat.shape)
        for i in range(min(mat.shape)):
                S[i][i] = math.sqrt(egnvalV[i])
        U = np.dot(np.dot(mat,np.transpose(VT)),LA.pinv(S))
        return U, S, VT

def reduceDim(u,s,v,num):
        for i in range(num):
                s = np.delete(s, s.shape[1]-1, axis = 1)
                v = np.delete(v, v.shape[0]-1, axis = 0)
                v = np.delete(v, v.shape[1]-1, axis = 1)
        return u, s, v


def main():
        mat = np.random.rand(5, 5)
        u, s, v = SVD(mat)
#       print "Original = ", mat, "\n\nU = ",u,"\n\nS = ",s,"\n\nV =", v
        U, S, V = reduceDim(u,s,v, 3)
        print "\nAfter Reduction of Dimension\n\n"
        red = np.dot(np.dot(U,S),V)
        print "U = ",u,"\n\nS = ",s,"\n\nV =", v, "\n\nReformed =", red

if __name__=="__main__":
        main()
~
```

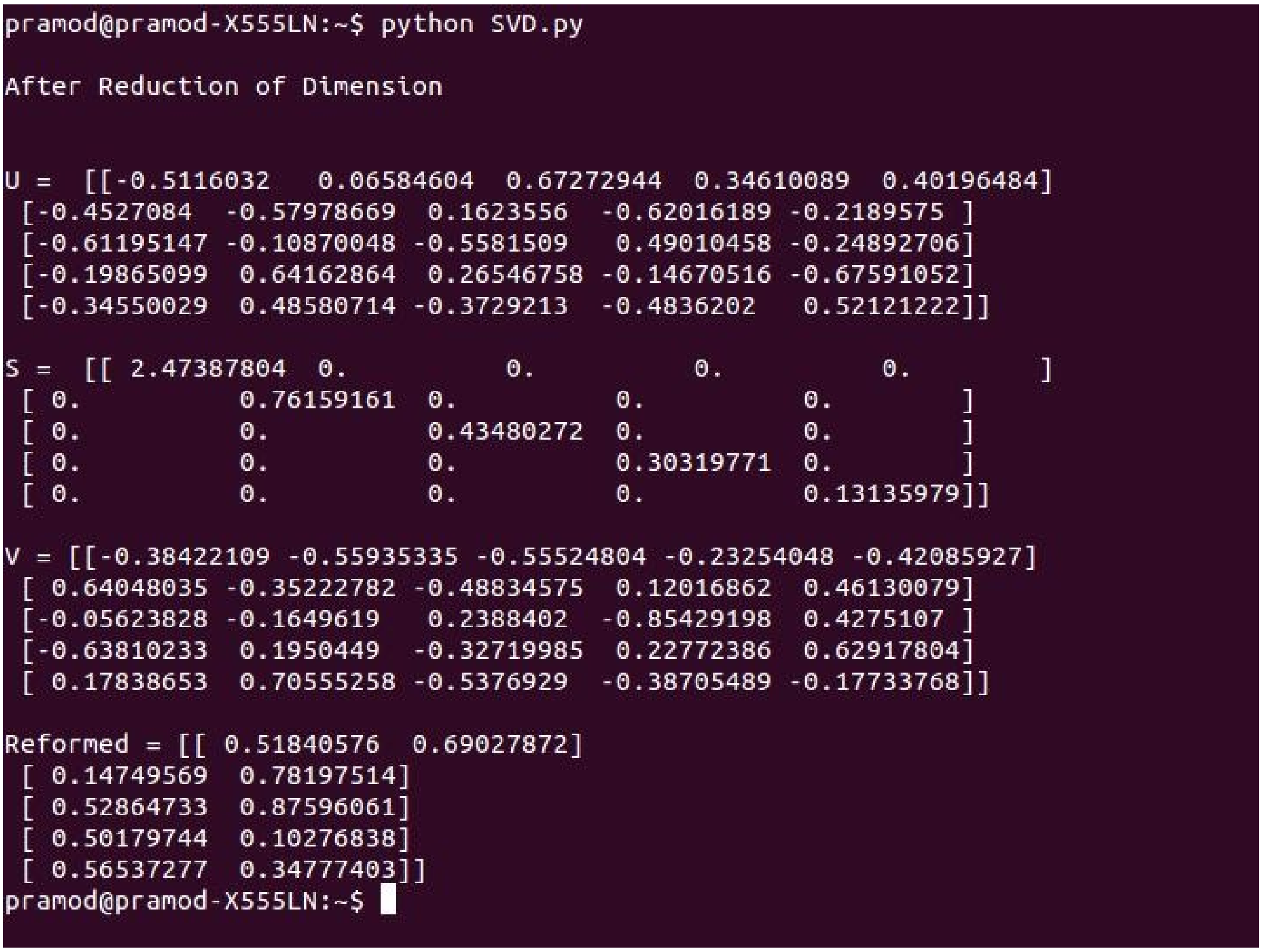
Matrix of size 5*5 is taken where 5 column represents the five dimension of data and rows represent the data samples and it is decomposed in matrices U S and V as below:



```
pramod@pramod-X555LN:~$ python SVD.py
Original =  [[ 0.70377037  0.81180308  0.41679331  0.98820843  0.96367014]
 [ 0.31150079  0.8514664   0.79478207  0.39829415  0.2461971 ]
 [ 0.5753268   0.01717992  0.66748664  0.95795825  0.84980515]
 [ 0.04199565  0.64147577  0.49737969  0.83868295  0.75419044]
 [ 0.9962875   0.689222    0.67109168  0.7432223   0.77645566]]

U =  [[-0.53342587 -0.04383926  0.05345115  0.64996678  0.53686229]
 [-0.33508265  0.74467889 -0.05815887 -0.48017405  0.31499747]
 [-0.4306072  -0.65745709  0.0170199  -0.57863311  0.21730547]
 [-0.39118225  0.10053301  0.77788068  0.01910548 -0.48104698]
 [-0.51451954  0.03427479 -0.62319539  0.10860375 -0.5778647 ]]

S =  [[ 3.31810092  0.          0.          0.          0.        ]
 [ 0.          0.83002312  0.          0.          0.        ]
 [ 0.          0.          0.6305025   0.          0.        ]
 [ 0.          0.          0.          0.48023182  0.        ]
 [ 0.          0.          0.          0.          0.04563613]]

V = [[-0.37870006 -0.40122297 -0.39659024 -0.53753078 -0.49938285]
 [-0.16718558  0.81358887  0.25028977 -0.32137468 -0.37973079]
 [-0.88646971  0.10092826 -0.06963362  0.37300954  0.24494895]
 [ 0.17481668  0.40805268 -0.86328314 -0.0135644   0.23977147]
 [ 0.11066087  0.01999437 -0.17310281  0.68444055 -0.69923598]]
pramod@pramod-X555LN:~$ 
```

The matrix after reduction of dimension in 2 is given below:

(Similarly it cand be done for 100 data samples which need 100*5 sized matrix)



```
pramod@pramod-X555LN:~$ python SVD.py

After Reduction of Dimension


U =  [[-0.5116032   0.06584604  0.67272944  0.34610089  0.40196484]
 [-0.4527084  -0.57978669  0.1623556  -0.62016189 -0.2189575 ]
 [-0.61195147 -0.10870048 -0.5581509   0.49010458 -0.24892706]
 [-0.19865099  0.64162864  0.26546758 -0.14670516 -0.67591052]
 [-0.34550029  0.48580714 -0.3729213  -0.4836202   0.52121222]]

S =  [[ 2.47387804  0.          0.          0.          0.        ]
 [ 0.          0.76159161  0.          0.          0.        ]
 [ 0.          0.          0.43480272  0.          0.        ]
 [ 0.          0.          0.          0.30319771  0.        ]
 [ 0.          0.          0.          0.          0.13135979]]

V = [[-0.38422109 -0.55935335 -0.55524804 -0.23254048 -0.42085927]
 [ 0.64048035 -0.35222782 -0.48834575  0.12016862  0.46130079]
 [-0.05623828 -0.1649619   0.2388402  -0.85429198  0.4275107 ]
 [-0.63810233  0.1950449  -0.32719985  0.22772386  0.62917804]
 [ 0.17838653  0.70555258 -0.5376929  -0.38705489 -0.17733768]]

Reformed = [[ 0.51840576  0.69027872]
 [ 0.14749569  0.78197514]
 [ 0.52864733  0.87596061]
 [ 0.50179744  0.10276838]
 [ 0.56537277  0.34777403]]
pramod@pramod-X555LN:~$
```