```
from pandas import *
df1 = read_csv("synthetic.txt",sep=" ")
```

```
!ssh localhost -l hduser /home/hduser/Downloads/hadoop*/bin/stop-all.sh
!ssh localhost -l hduser /home/hduser/Downloads/hadoop*/bin/start-all.sh
```

```
stopping jobtracker

localhost: stopping tasktracker

stopping namenode

localhost: stopping datanode

localhost: stopping secondarynamenode

starting namenode, logging to /home/hduser/Downloads/hadoop-1.0.4/libexec/../logs/hadoop-hduser-namenode

localhost: starting datanode, logging to /home/hduser/Downloads/hadoop-1.0.4/libexec/../logs/hadoop-hdus

localhost: starting secondarynamenode, logging to /home/hduser/Downloads/hadoop-1.0.4/libexec/../logs/ha

starting jobtracker, logging to /home/hduser/Downloads/hadoop-1.0.4/libexec/../logs/hadoop-hduser-jobtra

localhost: starting tasktracker, logging to /home/hduser/Downloads/hadoop-1.0.4/libexec/../logs/hadoop-h
```

```
!ssh localhost -l hduser /home/hduser/Downloads/hadoop*/bin/hadoop dfs -copyFromLocal
    $HOME/Documents/classnotes/stat/stat_hadoop_kmeans/synthetic.txt /user/hduser
```

```
copyFromLocal: Target /user/hduser/synthetic.txt already exists
```

```
print open("mapper.py").read()
```

```
#!/usr/bin/python
import os,sys,itertools
import numpy as np
from numpy import linalg as la
os.environ['MPLCONFIGDIR']='/tmp'
import pandas as pd

centers = pd.read_csv("/tmp/centers.csv",header=None,sep=",")

def dist(vect,x):
    return np.fromiter(itertools.imap(np.linalg.norm, vect-x),dtype=np.float)

def closest(x):
    d = dist(np.array(centers)[:,1:3],np.array(x))
    return np.argmin(d)

comb = lambda x: str(x[0])+":"+str(x[1])

df = pd.read_csv(sys.stdin,header=None,sep="   ")
df['cluster'] = df.apply(closest,axis=1)
df['coord'] = df.apply(comb,axis=1)
df.to_csv(sys.stdout, sep='\t',index=False, cols=['cluster','coord'],
          header=None)
```

```
print open("reducer.py").read()
```

```

#!/usr/bin/python
import os,sys,itertools
import numpy as np
from numpy import linalg as la
os.environ['MPLCONFIGDIR']='/tmp'
import pandas as pd

def coords(x):
    return pd.Series(np.array(str(x).split(":"),dtype=np.float64))

df = pd.read_csv(sys.stdin,sep="\t",names=['cluster','coord'])
df2 = df['coord'].apply(coords)
df3 = df.combine_first(df2)
df4 = df3.groupby('cluster').mean()
df4.to_csv(sys.stdout, sep=',',header=None)
```