

```

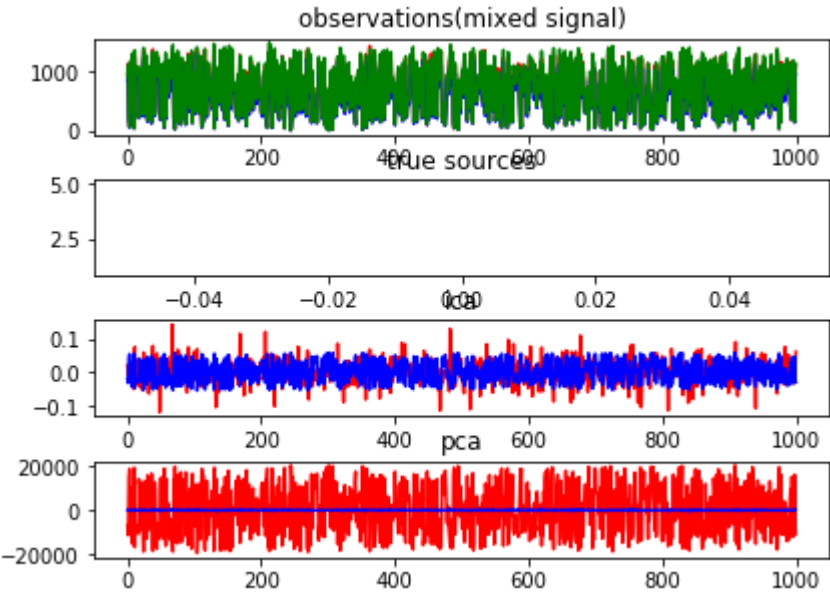
In [1]: import numpy as np
import pandas as pd
import seaborn as sns1
import numpy as np
import matplotlib.pyplot as plt
import scipy.stats as stats
import pylab as pl
import random
from scipy import signal
%matplotlib inline
from sklearn.decomposition import FastICA, PCA
df1_1=pd.read_csv('dist1_500_1.txt',sep=" ",header=None);
df1_1.dropna(how="all", inplace=True)
df1_2=pd.read_csv('dist1_500_2.txt',sep=" ",header=None);
df1_2.dropna(how="all", inplace=True)
df1=pd.concat([df1_1,df1_2])
df1+=0.2*np.random.normal(size=df1.shape)
df1/=df1.std(axis=0)
A=np.random.randint(12, size=(1000, 100))
X=np.dot(df1,A.T)
ica= FastICA(n_components=2)
df1_=ica.fit_transform(X)
A_=ica.mixing_
df1_.shape
df1_

pca = PCA(n_components=2)
H=pca.fit_transform(X)
# assert np.allclose(X, np.dot(df1_, A_.T) + ica.mean_)
plt.figure()
models=[X,df1,df1_,H]
names=['observations(mixed signal)','true sources','ica','pca']
colors=['red','blue','green']

for ii,(model,name) in enumerate(zip(models,names),1):
    plt.subplot(4,1,ii)
    plt.title(name)
    for sig,color in zip(model.T,colors):
        plt.plot(sig,color=color)

plt.subplots_adjust(0.09,0.04,0.94,0.94,0.26,0.46)
plt.show()

```



```

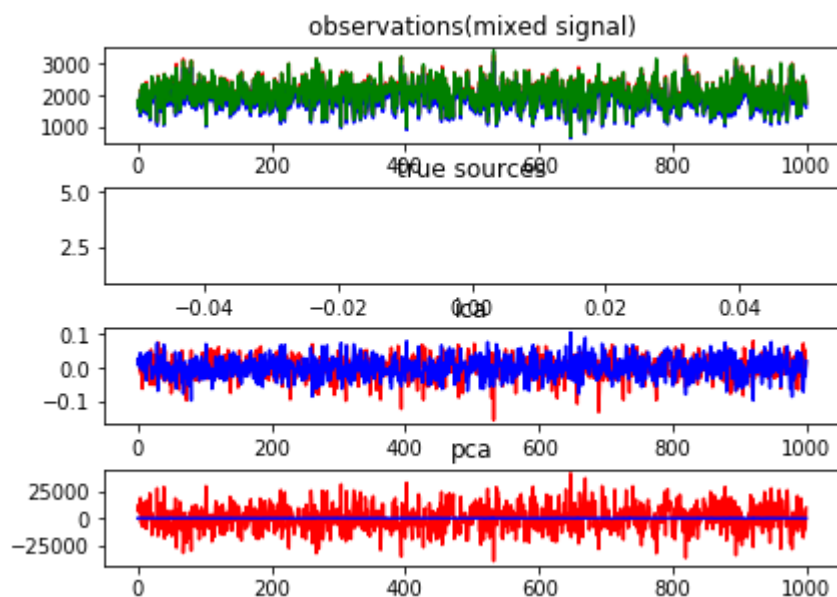
In [2]: df2_1=pd.read_csv('dist2_500_1.txt',sep=" ",header=None);
df2_1.dropna(how="all", inplace=True)
df2_2=pd.read_csv('dist2_500_2.txt',sep=" ",header=None);
df2_2.dropna(how="all", inplace=True)
df2=pd.concat([df2_1,df2_2])
df2+=0.2*np.random.normal(size=df2.shape)
df2/=df2.std(axis=0)
A=np.random.randint(12, size=(1000, 100))
X=np.dot(df2,A.T)
ica= FastICA(n_components=2)
df2_=ica.fit_transform(X)
A_=ica.mixing_
df2_.shape
df2_

pca = PCA(n_components=2)
H=pca.fit_transform(X)
# assert np.allclose(X, np.dot(df1_, A_.T) + ica.mean_)
plt.figure()
models=[X,df2,df2_,H]
names=['observations(mixed signal)','true sources','ica','pca']
colors=['red','blue','green']

for ii,(model,name) in enumerate(zip(models,names),1):
    plt.subplot(4,1,ii)
    plt.title(name)
    for sig,color in zip(model.T,colors):
        plt.plot(sig,color=color)

plt.subplots_adjust(0.09,0.04,0.94,0.94,0.26,0.46)
plt.show()

```



In []: