NUEN 647
Uncertainty Quantification for Nuclear Engineering
Homework 2

Due on Wednesday, October 19, 2016

Dr. McClarren

Paul Mendoza

Contents

Problem 1

Consider a covariance function between points in 2-D space:

$$k(x_1, y_1, x_2, y_2) = exp[-|x_1 - x_2| - |y_1 - y_2|]$$

Generate 4 realizations of a Gaussian random process with zero mean, $\mu(x, y) = 0$, and this covariance function defined on the unit square, $x, y \in [0, 1]$. For the realizations, evaluate the process at 50 equally space points in each direction. Plot the realizations.

Problem 2

Assume you have 100 samples of a pair of random variables (X_1, X_2) that have a positive correlation, call this set of pairs, $\mathbf{A_1}$. You then draw another 100 samples and call this set $\mathbf{A_2}$. The Pearson correlation between (X_1, X_2) in $\mathbf{A_1}$ is positive and hte Pearson correlation between (X_1, X_2) in $\mathbf{A_2}$ is negative. What can you say about the Pearson correlation for all 200 samples?

A normalized measure of the relation between two random variables, is the Pearson correlation coefficient, ρ. Oftentimes, this is simply called the correlation coefficient or correlation.

$$\rho(X_1, X_2) = \frac{E[X_1 X_2] - E[X_1]E[X_2]}{\sigma_{X1}\sigma_{X2}}$$

The expectation value for a series of realizations is defined:

$$E[g(x)] \approx \frac{1}{N}\sum_{i=1}^{N} g(x_i)$$

For the first 100 values:

$$\rho_1 = \frac{\frac{1}{100}\sum_{i=1}^{100} X_{1,i}X_{2,i} - \frac{1}{10000}\sum_{i=1}^{100} X_{1,i}\sum_{i=1}^{100} X_{2,i}}{\sigma_{X1,A1}\sigma_{X2,A1}}$$

$$100\sigma_{X1,A1}\sigma_{X2,A1}\rho_1 = \sum_{i=1}^{100} X_{1,i}X_{2,i} - \frac{1}{100}\sum_{i=1}^{100} X_{1,i}\sum_{i=1}^{100} X_{2,i}$$

$$100\sigma_{X1,A1}\sigma_{X2,A1}\rho_1 + \frac{1}{100}\sum_{i=1}^{100} X_{1,i}\sum_{i=1}^{100} X_{2,i} = \sum_{i=1}^{100} X_{1,i}X_{2,i}$$

Similarly for the second 100 values:

$$\sum_{i=101}^{200} X_{1,i}X_{2,i} = 100\sigma_{X1,A2}\sigma_{X2,A2}\rho_2 + \frac{1}{100}\sum_{i=101}^{200} X_{1,i}\sum_{i=101}^{200} X_{2,i}$$

The Pearson coefficient for all 200 values:

$$\rho_3 = \frac{\frac{1}{200}\sum_{i=1}^{200} X_{1,i}X_{2,i} - \frac{1}{40000}\sum_{i=1}^{200} X_{1,i}\sum_{i=1}^{200} X_{2,i}}{\sigma_{X1,A3}\sigma_{X2,A3}}$$

$$200\sigma_{X1,A3}\sigma_{X2,A3}\rho_3 = \sum_{i=1}^{200} X_{1,i}X_{2,i} - \frac{1}{200}\sum_{i=1}^{200} X_{1,i}\sum_{i=1}^{200} X_{2,i}$$

$$200\sigma_{X1,A3}\sigma_{X2,A3}\rho_3 + \frac{1}{200}\sum_{i=1}^{200} X_{1,i}\sum_{i=1}^{200} X_{2,i} = \sum_{i=1}^{200} X_{1,i}X_{2,i}$$

If we plug in the Pearson for the first 100 and second 100 for the right side of the equation,

$$200\sigma_{X1,A3}\sigma_{X2,A3}\rho_3 + \frac{1}{200}\sum_{i=1}^{200} X_{1,i}\sum_{i=1}^{200} X_{2,i}$$

$$=$$

$$100(\sigma_{X1A1}\sigma_{X2A1}\rho_1 + \sigma_{X1A2}\sigma_{X2A2}\rho_2) + \frac{1}{100}\left(\sum_{i=1}^{100} x_{1,i}\sum_{i=1}^{100} x_{2,i} + \sum_{i=101}^{200} x_{1,i}\sum_{i=101}^{200} x_{2,i}\right)$$

Grouping and setting:

$$\textstyle\sum_{i=1}^{100} x_{1,i} = X_{1,1}$$

$$\textstyle\sum_{i=1}^{100} x_{2,i} = X_{2,1}$$

$$\textstyle\sum_{i=101}^{200} x_{1,i} = X_{1,2}$$

$$\textstyle\sum_{i=101}^{200} x_{2,i} = X_{2,2}$$

$$\textstyle\sum_{i=1}^{200} x_{1,i} = X_{1,3}$$

$$\textstyle\sum_{i=1}^{200} x_{2,i} = X_{2,3}$$

$$200\sigma_{X1,A3}\sigma_{X2,A3}\rho_3 - 100(\sigma_{X1A1}\sigma_{X2A1}\rho_1 + \sigma_{X1A2}\sigma_{X2A2}\rho_2) = \frac{1}{100}\left(X_{1,1}X_{2,1} + X_{1,2}X_{2,2}\right) - \frac{1}{200}X_{1,3}X_{2,3}$$

Setting:

$$\sigma_{X1A1} = \tfrac{1}{100}\textstyle\sum_{i=1}^{100}(x_{1,i}^2 - \mu_{X_{11}}^2) = \tfrac{1}{100}\sigma'_{X1A1}$$

$$\sigma_{X2A1} = \tfrac{1}{100}\textstyle\sum_{i=1}^{100}(x_{2,i}^2 - \mu_{X_{21}}^2) = \tfrac{1}{100}\sigma'_{X2A1}$$

$$\sigma_{X1A2} = \tfrac{1}{100}\textstyle\sum_{i=101}^{200}(x_{1,i}^2 - \mu_{X_{12}}^2) = \tfrac{1}{100}\sigma'_{X1A2}$$

$$\sigma_{X2A2} = \tfrac{1}{100}\textstyle\sum_{i=101}^{200}(x_{2,i}^2 - \mu_{X_{22}}^2) = \tfrac{1}{100}\sigma'_{X2A2}$$

$$\sigma_{X1A3} = \tfrac{1}{200}\textstyle\sum_{i=1}^{200}(x_{1,i}^2 - \mu_{X_{13}}^2) = \tfrac{1}{200}\sigma'_{X1A3}$$

$$\sigma_{X2A3} = \tfrac{1}{200}\textstyle\sum_{i=1}^{200}(x_{2,i}^2 - \mu_{X_{23}}^2) = \tfrac{1}{200}\sigma'_{X2A3}$$

Where $A3$ and ρ_3 are for the series added to 200. Plugging these in, and multiplying both sides of the equation by 200.

$$\sigma'_{X1,A3}\sigma'_{X2,A3}\rho_3 - 2(\sigma'_{X1A1}\sigma'_{X2A1}\rho_1 + \sigma'_{X1A2}\sigma'_{X2A2}\rho_2) = 2\left(X_{1,1}X_{2,1} + X_{1,2}X_{2,2}\right) - X_{1,3}X_{2,3}$$

Note: $X_{1,3} = X_{1,1} + X_{2,1}$ and $X_{2,3} = X_{2,1} + X_{2,2}$ and that the right side of the equation simplifies to: $(X_{1,1} - X_{1,2})(X_{2,1} - X_{2,2})$. Then ρ_3 is:

$$\rho_3 = \frac{(X_{1,1} - X_{1,2})(X_{2,1} - X_{2,2}) + 2(\sigma'_{X1A1}\sigma'_{X2A1}\rho_1 + \sigma'_{X1A2}\sigma'_{X2A2}\rho_2)}{\sigma'_{X1,A3}\sigma'_{X2,A3}}$$

Problem 2 continued on next page. . .

Assuming that:

$$\frac{(X_{1,1}-X_{1,2})(X_{2,1}-X_{2,2})}{\sigma'_{X1,A3}\sigma'_{X2,A3}} \approx 0$$

and

$$\begin{aligned}\sigma'_{X1,A3}\sigma'_{X2,A3} &\approx 4\sigma'_{X1A1}\sigma'_{X2A1}\\ or &\approx 4\sigma'_{X1A2}\sigma'_{X2A2}\end{aligned}$$

The above would simplify to:

$$\rho_3 \approx \frac{\rho_1+\rho_2}{2}$$

Meaning, ρ_3 will usually be inside the interval $\rho_2 < \rho_3 < \rho_1$, I was curious, and wrote a script, to check to see if it would ever be outside. There could be an error with my script, but I found with the below script that a small percentage (less than 1% of the time), it would be outside the above interval. I also made a histogram plot...because I like wasting time.

Listing 1: Script for Problem

```python
#!/usr/bin/env python3

################################################################
#################### Import packages #########################
################################################################

import numpy as np
import time
start_time = time.time()
import Functions as Fun

################################################################
#################### Calculations ############################
################################################################

Error=[];Ntimes=1000;Nsamples=100;CountOut=0

for i in range(0,Ntimes):

    Positive=True
    Negative=True

    while(Positive or Negative):

        X1=np.random.uniform(-1,1,Nsamples)
        X2=np.random.uniform(-1,1,Nsamples)
        rho=Fun.CalculateRho(X1,X2)

        if rho>0:
            rho1=rho;X11=X1;X21=X2;
            Positive=False
        if rho<0:
            rho2=rho;X12=X1;X22=X2;
            Negative=False


    rho_Guess=(rho1+rho2)/2

    X13=np.append(X11,X12)
    X23=np.append(X21,X22)
    rho=Fun.CalculateRho(X13,X23)

    if(rho>rho1 or rho<rho2):
        CountOut=CountOut+1
    Error.append((abs(rho_Guess-rho)/rho)*100)

Fun.PlotHistSave(Error,Ntimes)

print("Percent outside rho1 and rho2: "+str(100*CountOut/Ntimes)+"%")

##################### Time To execute ################

print("--- %s seconds ---" % (time.time() - start_time))
```

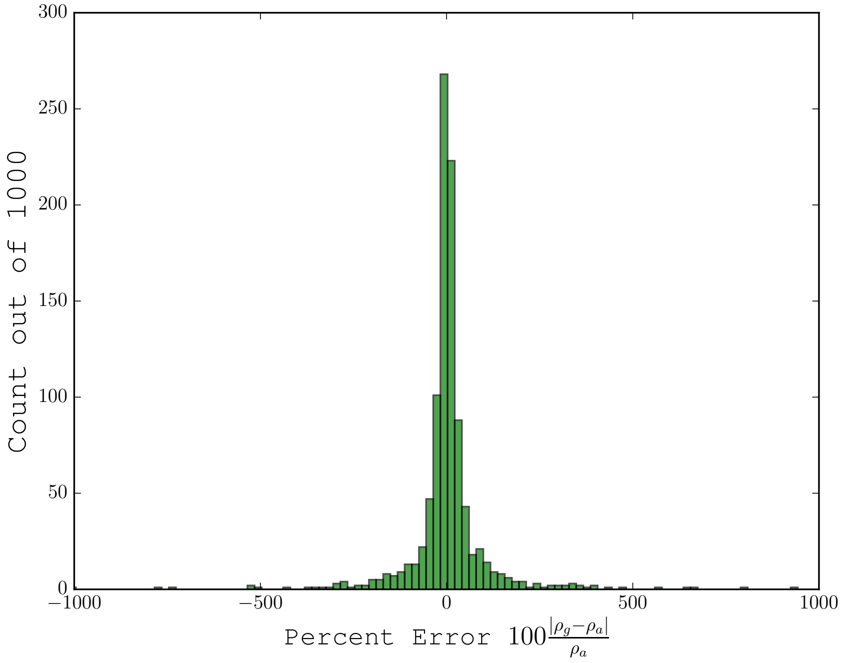

Figure 1: Histogram plot showing error ρ_g is the approximated guess at ρ_3 and ρ_a is the actual calculated ρ_3.

Problem 3

For the following data, compute by hand or via code you write the Pearson and Spearman correlations and Kendall's tau.

X_1	X_2
55.01	82.94
54.87	55.02
57.17	85.18
36.01	-84.27
35.88	-106.30
36.33	-119.65
43.49	-112.03
41.44	-71.69
54.43	-3.50
36.47	140.57

Listing 2: Script for Problem

```python
#!/usr/bin/env python3

################################################################
#################### Import packages ########################
################################################################

import numpy as np
import time
start_time = time.time()
import Functions as Fun

################################################################
#################### Calculations ###########################
################################################################

X1=np.array([55.01,54.87,57.17,36.01,35.88,36.33,
             43.49,41.44,54.43,36.47])
X2=np.array([82.94,55.02,85.18,-84.27,-106.30,-119.65,
             -112.03,-71.69,-3.50,140.57])

rho=Fun.CalculatePearson(X1,X2)

#Getting rank of each element, starting with 1
X1R=Fun.Rank(X1)
X2R=Fun.Rank(X2)

rhoS=Fun.CalculateSpearman(X1,X2,X1R,X2R)
tau=Fun.CalculateTau(X1,X2)

print("Pearson: "+str(rho))
print("Spearman: "+str(rhoS))
print("Kendall: "+str(tau))
```

Problem 3 continued on next page. . .

```
##################### Time To execute ################

print("--- %s seconds ---" % (time.time() - start_time))
```

Code output:

```
Pearson: 0.000672597071936
Spearman: 0.587878787879
Kendall: 0.5111111111111111
```

Problem 4

Demonstrate the tail dependence of a bivariate normal random variable is 0.

According to Link Stuff

Problem 5

Problem 5