

NUEN 647
Uncertainty Quantification for Nuclear Engineering
Homework 2

Due on Wednesday, October 19, 2016

Dr. McClarren

Paul Mendoza

Contents

Problem 1	3
Problem 2	4
Problem 3	8
Problem 4	10
Problem 5	11

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 the 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.

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

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:

$$\begin{aligned} \rho_1 &= \frac{\frac{1}{100} \sum_{i=1}^{100} X_{1,i} X_{2,i} - \frac{1}{10000} \sum_{i=1}^{100} X_{1,i} \sum_{i=1}^{100} X_{2,i}}{\sigma_{X_1, A1} \sigma_{X_2, A1}} \\ 100 \sigma_{X_1, A1} \sigma_{X_2, A1} \rho_1 &= \sum_{i=1}^{100} X_{1,i} X_{2,i} - \frac{1}{100} \sum_{i=1}^{100} X_{1,i} \sum_{i=1}^{100} X_{2,i} \\ 100 \sigma_{X_1, A1} \sigma_{X_2, A1} \rho_1 + \frac{1}{100} \sum_{i=1}^{100} X_{1,i} \sum_{i=1}^{100} X_{2,i} &= \sum_{i=1}^{100} X_{1,i} X_{2,i} \end{aligned}$$

Similarly for the second 100 values:

$$\sum_{i=101}^{200} X_{1,i} X_{2,i} = 100 \sigma_{X_1, A2} \sigma_{X_2, A2} \rho_2 + \frac{1}{100} \sum_{i=101}^{200} X_{1,i} \sum_{i=101}^{200} X_{2,i}$$

The Pearson coefficient for all 200 values:

$$\begin{aligned} \rho_3 &= \frac{\frac{1}{200} \sum_{i=1}^{200} X_{1,i} X_{2,i} - \frac{1}{40000} \sum_{i=1}^{200} X_{1,i} \sum_{i=1}^{200} X_{2,i}}{\sigma_{X_1, A3} \sigma_{X_2, A3}} \\ 200 \sigma_{X_1, A3} \sigma_{X_2, A3} \rho_3 &= \sum_{i=1}^{200} X_{1,i} X_{2,i} - \frac{1}{200} \sum_{i=1}^{200} X_{1,i} \sum_{i=1}^{200} X_{2,i} \\ 200 \sigma_{X_1, A3} \sigma_{X_2, A3} \rho_3 + \frac{1}{200} \sum_{i=1}^{200} X_{1,i} \sum_{i=1}^{200} X_{2,i} &= \sum_{i=1}^{200} X_{1,i} X_{2,i} \end{aligned}$$

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

$$\begin{aligned}
& 200\sigma_{X1,A3}\sigma_{X2,A3}\rho_3 + \frac{1}{200} \sum_{i=1}^{200} X_{1,i} \sum_{i=1}^{200} X_{2,i} \\
& = \\
& 100(\sigma_{X1A1}\sigma_{X2A1}\rho_1 + \sigma_{X1A2}\sigma_{X2A2}\rho_2) + \frac{1}{100} \left(\sum_{i=1}^{100} x_{1,i} \sum_{i=1}^{100} x_{2,i} + \sum_{i=101}^{200} x_{1,i} \sum_{i=101}^{200} x_{2,i} \right)
\end{aligned}$$

Grouping and setting:

$$\begin{aligned}
\sum_{i=1}^{100} x_{1,i} &= X_{1,1} \\
\sum_{i=1}^{100} x_{2,i} &= X_{2,1} \\
\sum_{i=101}^{200} x_{1,i} &= X_{1,2} \\
\sum_{i=101}^{200} x_{2,i} &= X_{2,2} \\
\sum_{i=1}^{200} x_{1,i} &= X_{1,3} \\
\sum_{i=1}^{200} x_{2,i} &= X_{2,3}
\end{aligned}$$

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

Setting:

$$\begin{aligned}
\sigma_{X1A1} &= \frac{1}{100} \sum_{i=1}^{100} (x_{1,i}^2 - \mu_{X_{11}}^2) = \frac{1}{100} \sigma'_{X1A1} \\
\sigma_{X2A1} &= \frac{1}{100} \sum_{i=1}^{100} (x_{2,i}^2 - \mu_{X_{21}}^2) = \frac{1}{100} \sigma'_{X2A1} \\
\sigma_{X1A2} &= \frac{1}{100} \sum_{i=101}^{200} (x_{1,i}^2 - \mu_{X_{12}}^2) = \frac{1}{100} \sigma'_{X1A2} \\
\sigma_{X2A2} &= \frac{1}{100} \sum_{i=101}^{200} (x_{2,i}^2 - \mu_{X_{22}}^2) = \frac{1}{100} \sigma'_{X2A2} \\
\sigma_{X1A3} &= \frac{1}{200} \sum_{i=1}^{200} (x_{1,i}^2 - \mu_{X_{13}}^2) = \frac{1}{200} \sigma'_{X1A3} \\
\sigma_{X2A3} &= \frac{1}{200} \sum_{i=1}^{200} (x_{2,i}^2 - \mu_{X_{23}}^2) = \frac{1}{200} \sigma'_{X2A3}
\end{aligned}$$

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

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

Note: $X_{1,3} = X_{1,1} + X_{1,2}$ 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:

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

Assuming that:

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

and

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

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

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

#####
##### Import packages #####
5 #####

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

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

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

for i in range(0,Ntimes):

20     Positive=True
     Negative=True

     while(Positive or Negative):

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

         if rho>0:
30             rho1=rho;X11=X1;X21=X2;
             Positive=False
         if rho<0:
```

```

    rho2=rho;X12=X1;X22=X2;
    Negative=False

35

    rho_Guess=(rho1+rho2)/2

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

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

Fun.PlotHistSave(Error,Ntimes)

print("Percent outside rho1 and rho2: "+str(100*CountOut/Ntimes)+"%")
50

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

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

```

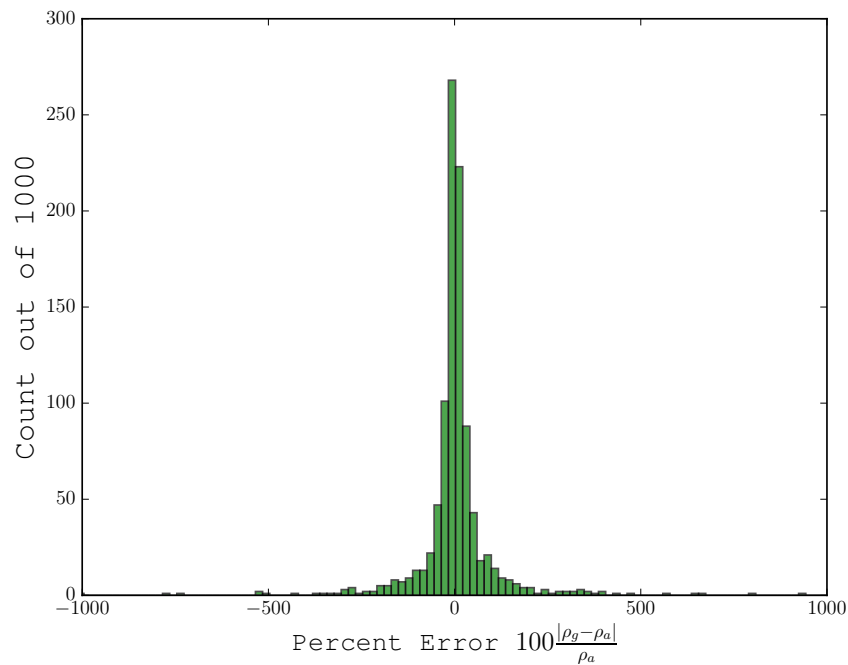


Figure 1: Histogram plot showing error ρ_g is the approximated guess at ρ_3 and ρ_a is the actual calculated ρ_3 .

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

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

#####
##### Import packages #####
5 #####

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

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

15 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])

20 rho=Fun.CalculatePearson(X1,X2)

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

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

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



```
##### Time To execute #####  
35 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.

The bivariate Gaussian copula is defined as:

$$C_N(u, v) = \Phi_\rho(\Phi^{-1}(u), \Phi^{-1}(v))$$

Where:

$$\Phi^{-1}(q) = \mu + \sigma\sqrt{2}erf^{-1}(2q - 1)$$

Evaluated at $q = 0$:

$$\Phi^{-1}(0) = -\infty$$

Also where:

$$\Phi_\rho(x, y) = \int_{-\infty}^x dx' \int_{-\infty}^y dy' \frac{1}{2\pi\sigma_x\sigma_y\sqrt{1-\rho^2}} \exp\left[-\frac{z}{2(1-\rho^2)}\right]$$

with

$$z = \frac{(x' - \mu_x)^2}{\sigma_x^2} - \frac{2\rho(x' - \mu_x)(y' - \mu_y)}{\sigma_x\sigma_y} + \frac{(y' - \mu_y)^2}{\sigma_y^2}$$

and

$$\rho = \frac{E[XY] - E[X]E[Y]}{\sigma_x\sigma_y}$$

Note: McClarren is a poo poo head

$$\Phi_\rho(x, y) = \int_{-\infty}^x dx' \int_{-\infty}^y dy' \frac{1}{2\pi\sigma_x\sigma_y\sqrt{1-\rho^2}} \exp\left[-\frac{z}{2(1-\rho^2)}\right]$$

Tail Dependence:

$$\lambda_l = \lim_{q \rightarrow 0} \frac{C(q, q)}{q}$$

Problem 5

Another Archimedean copula is the Joe copula with generator

$$\phi_J(t) = -\log(1 - (1 - t)^\theta),$$

and

$$\phi_J^{-1} = 1 - (1 - \exp(-t))^{1/\theta}.$$