ENGG1003 - Thursday Week 4

Using random numbers, and reading from spreadsheets

Steve Weller & Sarah Johnson

University of Newcastle

18 March, 2021

Last compiled: March 17, 2021 8:38pm +11:00

Lecture overview

1. Using random numbers
2. Reading from spreadsheets

1) Using random numbers

- reminder and recap
- random integers
- random floats

Random integers: simulating fair coin toss

Example 1

- generate array of 0s and 1s length N
- 0=heads, 1=tails
- expected number of 0s (heads)?

Coin toss simulation

- Python code for coin toss
- headsTails.py
- live demo

```python
import numpy as np

# generate random array of 0s and 1s, 0==heads & 1==tails
# N integers from [0,2) ie: 0 or 1
N = 100000
x = np.random.randint(0, 2, N)
print(x)

headCnt = 0;
for i in range(0,N,1):
    if x[i]==0:
        headCnt += 1

print('Expected number of heads: {}'.format(N/2))
print('Observed number of heads: {}'.format(headCnt))
```

Random floats: engineering tolerance

Example 2

- simulate values in a range
- need engineering application—part manufactured within a tolerance, calculate fraction outside range

Engineering tolerance simulation

- Python code for engineering tolerance
- engTolerance.py
- live demo

```python
import numpy as np

# generate random array of N floats in range [10,20]
N = 10000
x = np.random.uniform(10,20,N)
#print(x)

goodCnt = 0;
for i in range(0,N,1):
    if 11 <= x[i] <= 19:
        goodCnt += 1

print('Fraction of parts within tolerance: {}'.format(goodCnt/N))
```

Random floats: simulate dartboard

Example 3

- values in circle
- plot red inside, blue outside

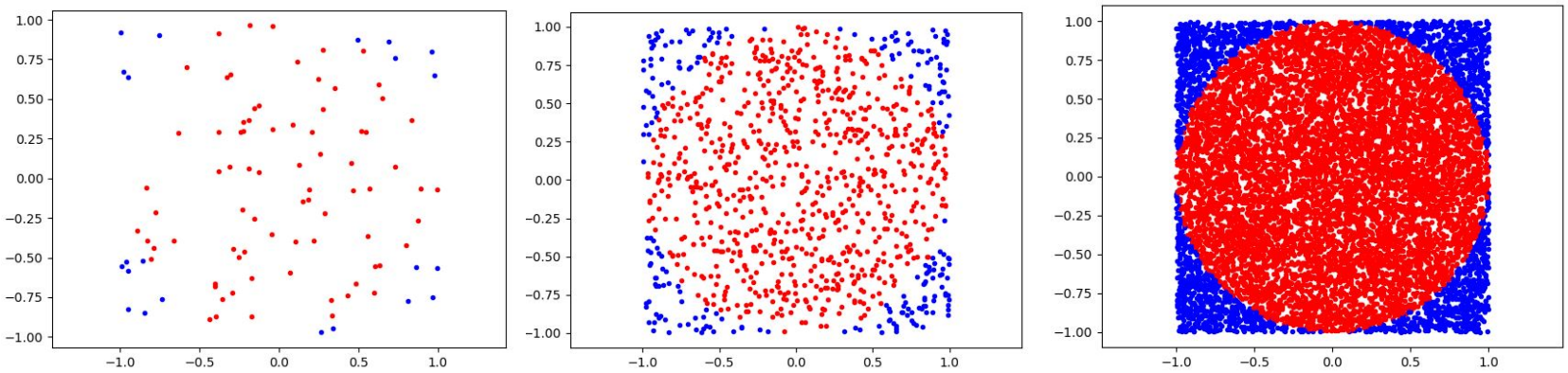
Dartboard simulation

- Python code for dartboard
- dartboard.py
- live demo

```python
import numpy as np
import matplotlib.pyplot as plt

# generate random array of (x,y) pairs covering
# square with edge length 2
N = 10000
x = np.random.uniform(-1, 1, N)      # N floats from [-1,1]
y = np.random.uniform(-1, 1, N)      # N floats from [-1,1]

insideCnt = 0;
for i in range(0,N,1):
    if x[i]**2 + y[i]**2 <= 1:
        plt.plot(x[i],y[i],'r.')
    else:
        plt.plot(x[i],y[i],'b.')

plt.axis('equal')
plt.show()
```

Dartboard simulation

- dartboard red/blue simulation output plots
- N=100,1000,10K



Random floats: estimate π

Example 4

- modify previous example to count points inside circle, hence...
- estimate π

Random floats: estimate π

```python
import numpy as np
import matplotlib.pyplot as plt

# generate random array of (x,y) pairs covering
# square with edge length 2
N = 10000
x = np.random.uniform(-1, 1, N)      # N floats from [-1,1]
y = np.random.uniform(-1, 1, N)      # N floats from [-1,1]

insideCnt = 0;
for i in range(0,N,1):
    if x[i]**2 + y[i]**2 <= 1:
        plt.plot(x[i],y[i],'r.')
        insideCnt += 1
    else:
        plt.plot(x[i],y[i],'b.')

R = insideCnt/N
print('Estimate of pi: {}'.format(4*R))

plt.axis('equal')
plt.show()
```

Live demo

2) Reading from spreadsheets