Lecture A5 - Solution

Reinforcement Learning Study

2021-01-13

차 례

Exercise 1 - (김봉석)

Using ruinf() function in R, complete the following code block that generates 1,000 random numbers that follow $exp(3)$

```
N<-1000
u<-runif(N)
x<-(-log(1-u)/3)
head(x)
```

```
## [1] 0.32304390 0.04619330 0.15040797 0.04509487 0.32900170 0.05891449
```

```
import numpy as np
N=1000
u=np.random.uniform(size=N)
x=((-np.log(1-u))/3)
print(x[0:6])
```

in python

```
## [0.01709056 0.59020341 0.2774003  0.30996772 0.25788983 0.5962709 ]
```

Various random numbers, Uniform random numbers p.15 - (김봉석)

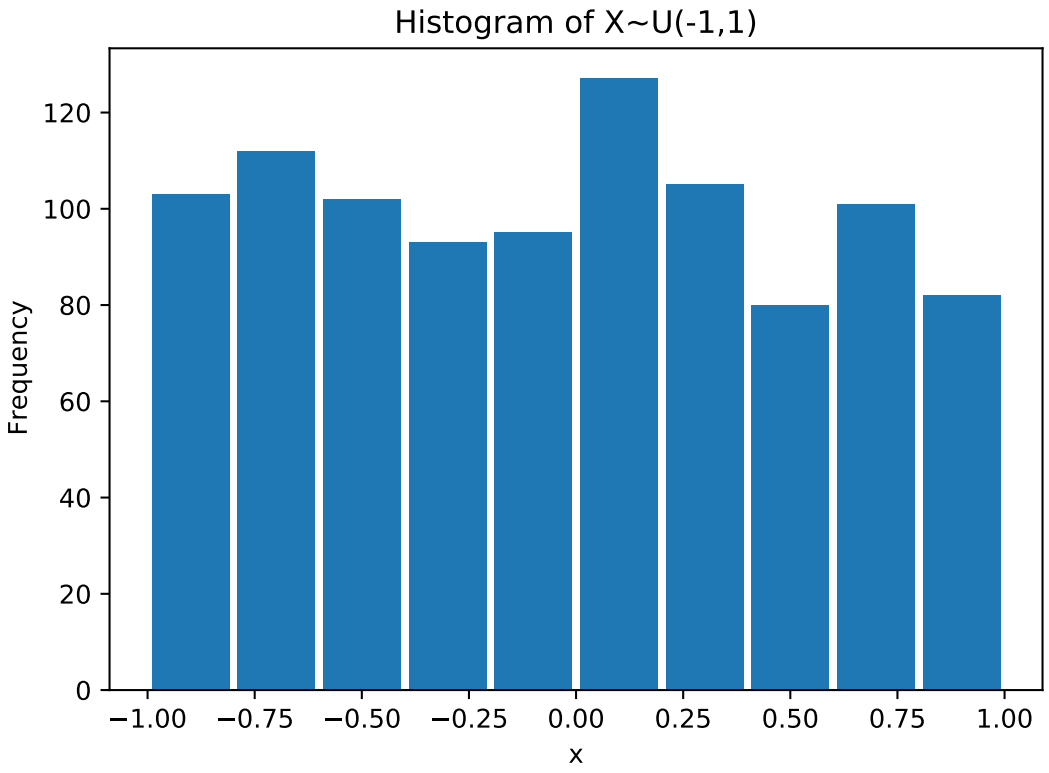
```
import matplotlib.pyplot as plt
import numpy as np

x=np.random.uniform(size=1000, low=-1,high=1)

plt.hist(x, rwidth=0.9)
```

```
## (array([103., 112., 102.,  93.,  95., 127., 105.,  80., 101.,  82.]), array([-9.99766628e-
01, -7.99827974e-01, -5.99889320e-01, -3.99950666e-01,
##        -2.00012013e-01, -7.33589311e-05,  1.99865295e-01,  3.99803949e-01,
##       5.99742602e-01,  7.99681256e-01,  9.99619910e-01]), <BarContainer object of 10 artists>)
```

```
plt.xlabel("x")
plt.ylabel("Frequency")
plt.title("Histogram of X~U(-1,1)")
```



Various random numbers, Normal random numbers p.16 - (김봉석)

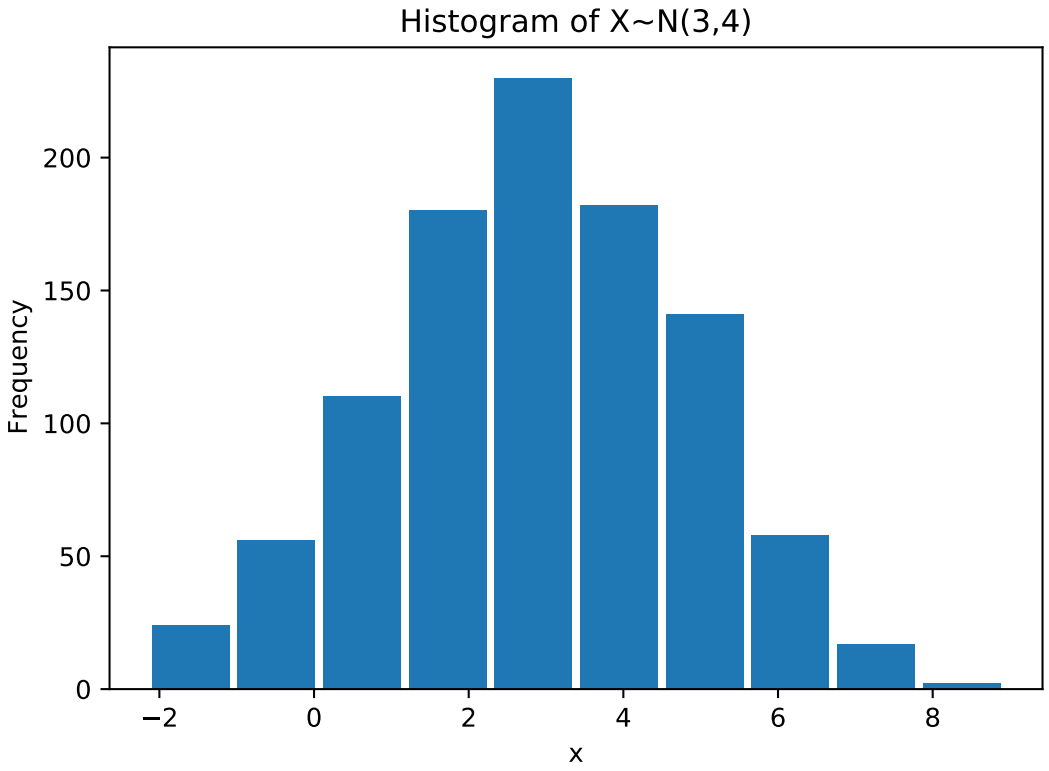
```
import matplotlib.pyplot as plt
import numpy as np

x=np.random.normal(size=1000, loc=3, scale=2) # loc = mean , scale =sd

plt.hist(x, rwidth=0.9)
```

```
## (array([ 24.,  56., 110., 180., 230., 182., 141.,  58.,  17.,   2.]), array([-2.15148606, -
1.04359189,  0.06430229,  1.17219646,  2.28009063,
##         3.3879848 ,  4.49587897,  5.60377315,  6.71166732,  7.81956149,
##         8.92745566]), <BarContainer object of 10 artists>)
```

```
plt.xlabel("x")
plt.ylabel("Frequency")
plt.title("Histogram of X~N(3,4)")
```



Various random numbers, Exponential random numbers p.17 - (김봉석)

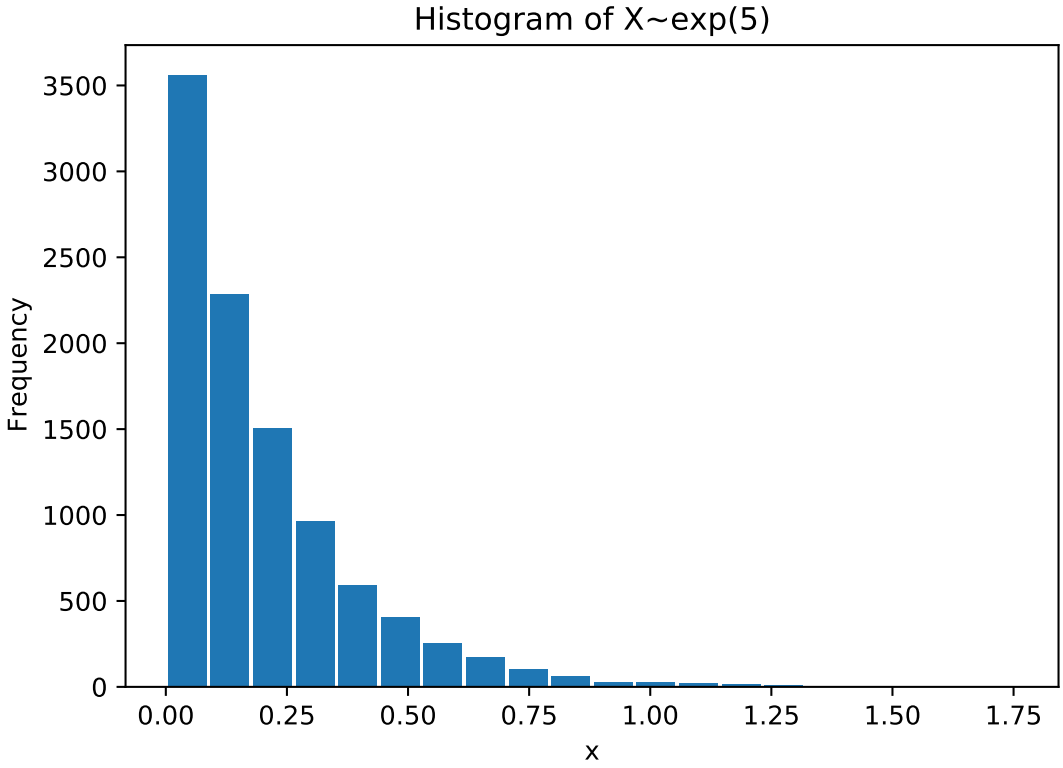
```
import matplotlib.pyplot as plt
import numpy as np

x=np.random.exponential(size=10000,scale=1/5) # meaning lambda=5

plt.hist(x,rwidth=0.9,bins=20)
```

```
## (array([3.557e+03, 2.285e+03, 1.507e+03, 9.610e+02, 5.890e+02, 4.070e+02,
##         2.550e+02, 1.720e+02, 1.000e+02, 6.000e+01, 2.900e+01, 2.600e+01,
##         1.900e+01, 1.300e+01, 8.000e+00, 3.000e+00, 4.000e+00, 1.000e+00,
##               3.000e+00,  1.000e+00]),  array([1.32920108e-05,  8.79952400e-02,  1.75977188e-
01, 2.63959136e-01,
##         3.51941084e-01, 4.39923032e-01, 5.27904980e-01, 6.15886928e-01,
##         7.03868876e-01, 7.91850824e-01, 8.79832772e-01, 9.67814720e-01,
##         1.05579667e+00, 1.14377862e+00, 1.23176056e+00, 1.31974251e+00,
##         1.40772446e+00, 1.49570641e+00, 1.58368836e+00, 1.67167030e+00,
##         1.75965225e+00]), <BarContainer object of 20 artists>)
```

```
plt.xlabel("x")
plt.ylabel("Frequency")
plt.title("Histogram of X~exp(5)")
```


Histogram of X~exp(5)
Frequency
3500
3000
2500
2000
1500
1000
500
0
0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75
x

Various random numbers, Poisson random numbers p.18 - (김봉석)

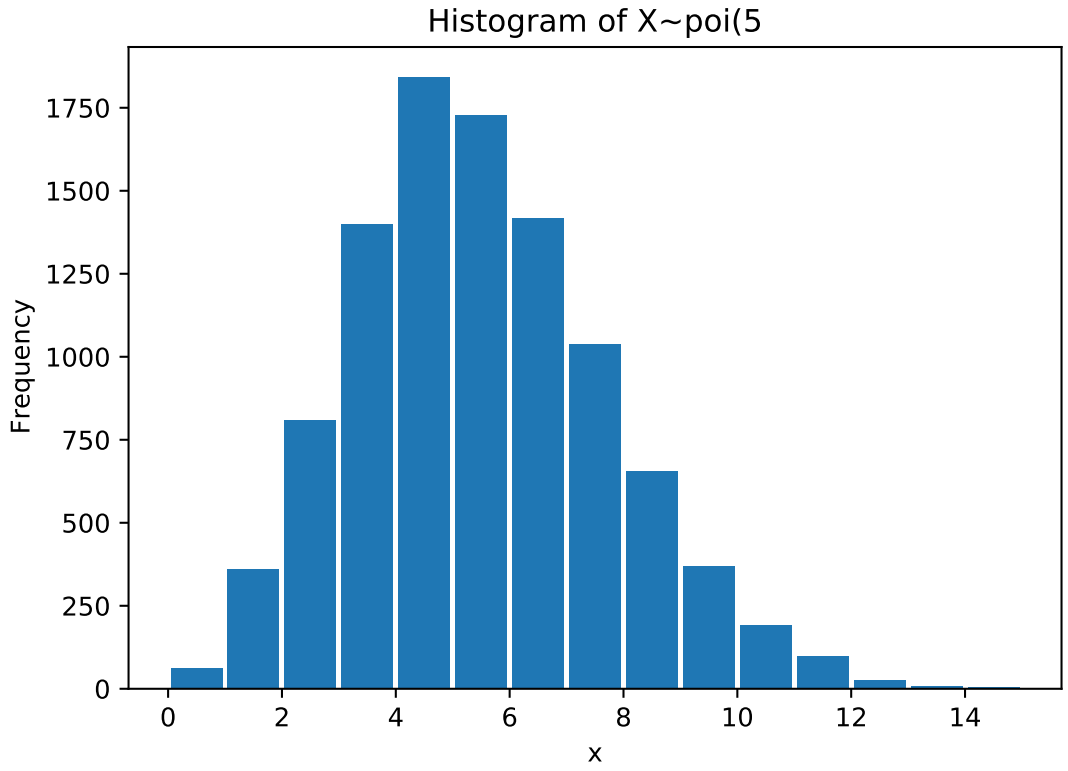
```
import matplotlib.pyplot as plt
import numpy as np

x=np.random.poisson(size=10000,lam=5) # meaning lambda=5

plt.hist(x,rwidth=0.9,bins=15)
```

```
## (array([  63.,  359.,  810., 1399., 1841., 1726., 1416., 1037.,  654.,
##       369.,  190.,   98.,   25.,    8.,    5.]), array([ 0.,  1.,  2.,  3.,  4.,  5.,  6.,  7.,  8.,  9., 10., 11., 12.,
##        13., 14., 15.]), <BarContainer object of 15 artists>)
```

```
plt.xlabel("x")
plt.ylabel("Frequency")
plt.title("Histogram of X~poi(5")
```



```
"Done, Lecture A5. Simulation 2 "
```

```
## [1] "Done, Lecture A5. Simulation 2 "
```